Proceedings of the

EASTERN JOINT COMPUTER CONFERENCE

December 9-13, 1957 Washington, D.C.

THEME: COMPUTERS WITH DEADLINES TO MEET

SPONSORS:

THE INSTITUTE OF RADIO ENGINEERS
Professional Group on Electronic Computers

THE ASSOCIATION FOR COMPUTING MACHINERY

THE AMERICAN INSTITUTE OF ELECTRICAL ENGINEERS
Committee on Computing Devices

Printed in the United States of America

Price $3.00

PROCEEDINGS OF THE EASTERN JOINT COMPUTER CONFERENCE

PAPERS AND DISCUSSIONS PRESENTED AT THE
JOINT IRE-ACM-AIEE COMPUTER CONFERENCE
WASHINGTON, D.C. DECEMBER 9-13, 1957

THEME: COMPUTERS WITH DEADLINES TO MEET

SPONSORS

The Institute of Radio Engineers Professional Group on Electronic Computers

The Association for Computing Machinery

The American Institute of Electrical Engineers Committee on Computing Devices

Published by

THE INSTITUTE OF RADIO ENGINEERS, INC.

1 East 79 Street, New York 21, N.Y.

ADDITIONAL COPIES

Additional copies may be purchased from the sponsoring societies listed below at $3.00 per copy. Checks should be made payable to any one of the following:

INSTITUTE OF RADIO ENGINEERS
1 East 79 Street, New York 21, N.Y.

ASSOCIATION FOR COMPUTING MACHINERY
2 East 63 Street, New York 21, N.Y.

AMERICAN INSTITUTE OF ELECTRICAL ENGINEERS
33 West 39 Street, New York 18, N.Y.

LIBRARY OF CONGRESS CATALOG CARD NUMBER: 54-9071

NATIONAL JOINT COMPUTER COMMITTEE

M. M. ASTRAHAN, Chairman
IBM Corporation

N. H. TAYLOR, Vice-Chairman
Lincoln Laboratories, M.I.T.

IRE Representatives

DANIEL HAAGENS, Underwood Corporation

L. NOFREY, Marchant Research, Inc.

W. S. SPEER, Norden-Ketay Corporation

N. H. TAYLOR, Lincoln Laboratories, M.I.T.

WERNER BUCHHOLZ, Ex-Officio
IBM Corporation

L. G. CUMMING, Headquarters
The Institute of Radio Engineers

ACM Representatives

S. FERNBACH, University of California

GILBERT W. KING, International Telemeter Corporation

ALAN PERLIS, Carnegie Institute of Technology

F. M. VERZUH, Massachusetts Institute of Technology

J. W. CARR III, Ex-Officio
University of Michigan

J. MOSHMAN, Headquarters
Council for Economic and
Industry Research, Inc.

AIEE Representatives

J. G. BRAINERD, University of Pennsylvania

FRED KALBACH, Burroughs Corporation

H. F. MITCHELL, JR., Sperry Rand Corporation

DAN C. ROSS, IBM Corporation

E. L. HARDER, Ex-Officio
Westinghouse Electric Corporation

R. S. GARDNER, Headquarters
American Institute of Electrical Engineers

Liaison with National Simulation Council

R. M. HOWE
University of Michigan

EASTERN JOINT COMPUTER CONFERENCE COMMITTEE

1957

S. N. ALEXANDER, Chairman
National Bureau of Standards

TECHNICAL PROGRAM COMMITTEE

HARRY H. GOODE, Chairman
University of Michigan

I. L. AUERBACH
Auerbach Electronics Corporation

A. A. COHEN
Remington Rand UNIVAC

J. A. HADDAD
IBM Corporation

R. M. HOWE
University of Michigan

E. C. JOHNSON
Bendix Aviation Corporation

R. E. SPRAGUE
Teleregister Corporation

LOCAL ARRANGEMENTS COMMITTEE

JOHN R. PROVAN, Chairman
U.S. Bureau of the Budget

W. HOWARD GAMMON, Assistant Chairman
Office of the Secretary of Defense

MARGARET R. FOX, Secretary
National Bureau of Standards

Advisory Staff

S. N. ALEXANDER
WALTER L. ANDERSON
DAVID S. BENDER
RALPH I. COLE

HOWARD T. ENGSTROM
EZRA GLASER
LOWELL H. HATTERY

EDWARD J. MAHONEY
GEORGE W. PETRIE
HAROLD K. SKRAMSTAD
MARK SWANSON

Vice-Chairmen

DELMER C. PORTS, Finance
Jansky and Bailey, Inc.

RICHARD T. BURROUGHS, Registration
IBM Corporation

MALCOLM B. CATLIN, Publicity
Council for Economic and Industry Research, Inc.

CLARKE RISLER, Hotel Arrangements
Remington Rand Division, Sperry Rand Corporation

L. DAVID WHITELOCK, Exhibits Liaison
U.S. Navy Department

ETHEL C. MARDEN, Women's Activities
National Bureau of Standards

PUBLICATIONS COMMITTEE

MORRIS RUBINOFF, Chairman
Philco Corporation

R. J. KONEFAL
Philco Corporation

TABLE OF CONTENTS

The Numericord Machine-Tool Director

GERALD T. MOORE[†]

WHAT is numerical control of machine tools? I would answer by saying that it is a system of machine-tool control in which the machining operation is guided by instructions in the form of coded numbers. These instructions may be inserted via punched cards, punched tape, magnetic tape, or other suitable means. A complete sequence of operations is predetermined and programmed in a coded form which is understandable to the controller or the director, as it is called.

As divided into their broad classifications, the two types of numerical machine-tool controls are:

1) Positioning controls. A sequence of positions of a tool is controlled, some operation occurring at each position before the tool continues to the next position. Here, it is generally unimportant by which route and at which speed the tool progresses from one position to the next. The tool is not in contact with the workpiece when moving between positions.

2) Path controls. The tool is made to follow a prescribed path over the surface of the workpiece at a prescribed, but not necessarily constant, velocity. Depending upon the particular control system, the path may be in two or three dimensions.

The numericord machine-tool director, about which I am going to talk, is a path control system. While it differs in some respects from other path control systems, a study of its functioning will serve to demonstrate the processes involved in path control.

When I speak of the machine-tool director system, I am not including the machine tool itself with its power servomechanisms and error-detecting and amplifying circuits. That is separate equipment. I am talking about the data-processing and digital-to-analog conversion equipment which is necessary to provide real-time continuous-control signals in response to the numerical instructions inserted into the director. In the Numericord system there is no physical interconnection between the director system and the machine-tool controls. The continuous-control signals are recorded on magnetic tape and are subsequently played back at the machine tool. The interposition of the recording and playback functions in the sequence of control makes possible the divorcing of the director system from the machine tool. Therefore, a magnetic tape may be repeatedly used to produce several identical parts on the machine tool. Meanwhile, the director system is recording tapes for other machine tools. Fig. 1 shows the director system.

† Concord Control, Inc., Boston, Mass.

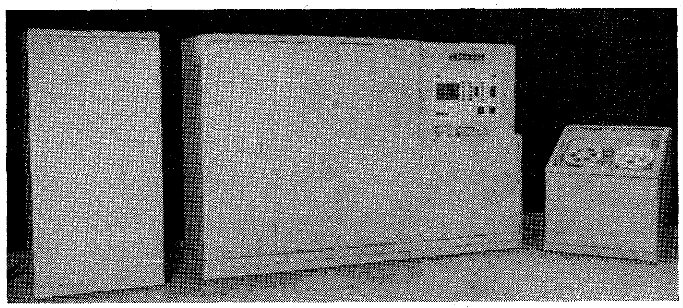


Fig. 1—The director system.

The Numericord director has punched paper tape as its input. Coded numbers on this tape prescribe the path of the cutting tool center in five axes. Thus, for example, a milling machine on which the milling head has two rotational degrees of freedom as well as orthogonal X, Y, and Z degrees of freedom may be controlled. The punched paper tape does not specify the path at all points, therefore it is necessary for the director to interpolate between specified points. That is, the continuous-control signals for the five axes must direct the cutting tool along some path between the points specified on tape.

The amount of data that is required on the input tape for any numerical system depends upon the interpolation method used in that system, and, in general, the amount of data decreases with increasing complexity of the interpolator. If you were to define positions on paper tape successively at one-thousandth intervals on the workpiece, no interpolation between defined points would be necessary at all in order to attain a reasonable degree of accuracy. On the other hand, if the director will interpolate linearly between defined points, that is, if the director directs the machine tool to cut a straight line between defined points, then the paper-tape input need define only the end points of all straight-line cuts. Here, however, a curve must be defined as a series of straight-line segments, the number of the segments depending upon the prescribed accuracy. If, for instance, you were required to cut half of an inside circle of six-inch diameter with a two-inch diameter cutter maintaining an accuracy of one thousandth, linear interpolation would require that the paper tape input specify 78 straight-line cuts. More elaborate interpolation schemes are possible, which pass higher degree curves through a number of specified points. Depending upon the type of cutting to be done, these systems may reduce considerably the amount of data required at the director input at the expense of a greater amount of equipment within the director. All considerations being taken, the Numericord designers were led to the choice of a linearly interpolating system.

The interpolator has five output lines, one for each axis. On each of the output lines discrete "command" pulses appear. One pulse represents a fixed increment of displacement at the machine tool, the amount of displacement being referred to as the quantization level of the system. The Numericord director has a quantization level of one eighth of a thousandth of an inch. Thus, the occurrence of 8000 command pulses in succession on the X-axis output line would drive the tool one inch over the workpiece in the X direction. Since the power servomechanisms at the machine tool respond to analog signals and not to pulses, a pulse-to-analog conversion must take place. This occurs in the "decoder." The output from the decoder consists of five command synchro signals. These are recorded on magnetic tape. When played back at the machine tool, the signals provide the command positions for five servodrives.

A block diagram of the director system is shown in Fig. 2.

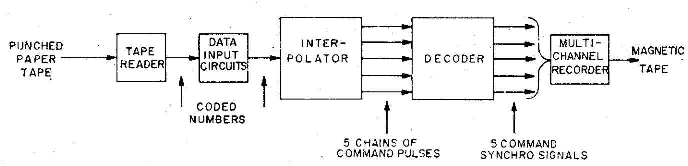


Fig. 2—Block diagram of a Numericord system.

Data Input

The terminal point of each straight-line segment of the programmed tool path is specified on the paper tape by coding the distance in each axis from the terminal point of the previous straight-line segment. In addition to the incremental distances for the five axes, a time-of-cut, or command time, is specified for each straight-line segment, and a direction-of-cut or sign code is inserted at the beginning of each dimension. One line of tape is required for each coded decimal digit or sign. The arrangement of data is such that the command time appears first. Three lines of tape are allocated to command time so that the command time is three decimal digits long. The command time is followed by a sign and seven decimal digit codes for the X axis, then by a sign and seven decimal digit codes for the Y axis, and so on for each of the five axes. The seven decimal digits indicate hundreds of inches, tens of inches, units, tenths, hundredths, thousandths, and tenths of thousandths of inches. The seventh digit is either a zero or a five so that distances are programmed in multiples of a half of a thousandth. The director will handle a maximum distance of 399.9995 inches in all axes. Of course, when the fourth and fifth axes are used to control rotations, a conversion must be made from angular degrees to linear inches so that the programming can be done in inches.

The command-time and command-distance information for a single straight-line cut comprise one "block" of paper-tape information. At the director, each block is read serially, line by line, each line being translated into a four-digit binary code and stepped into four magnetic-core stepping registers. The first digit of the four-digit code is weighted five, the second is weighted two, and the third and fourth are each weighted one. The binary code for seven, therefore, is the binary 1100. Each of the four stepping registers is associated with one of the four binary digit columns in the translated number. One register is designated the "five" register. Into it goes the most significant binary digit. The next register is designated the "two" register, and it receives the second most significant digit. The third and fourth registers are the "one-A" and "one-B" registers respectively.

Table I demonstrates the coding.

TABLE I
Coding

	5		2		1_A		1_B		
0	0×5	+	0×2	+	0×1	+	0×1	=	0
1	0×5	+	0×2	+	0×1	+	1×1	=	1
2	0×5	+	1×2	+	0×1	+	1×1	=	2
3	0×5	+	1×2	+	0×1	+	1×1	=	3
4	0×5	+	1×2	+	1×1	+	1×1	=	4
5	1×5	+	0×2	+	0×1	+	0×1	=	5
6	1×5	+	0×2	+	1×1	+	0×1	=	6
7	1×5	+	1×2	+	0×1	+	0×1	=	7
8	1×5	+	1×2	+	0×1	+	1×1	=	8
9	1×5	+	1×2	+	1×1	+	1×1	=	9

As each character is read, the appropriate code is set into the shift registers and advanced one position into the registers. When one block of tape has been read, the registers are full, and the tape reader stops. The numbers read first are stored in the magnetic cores farthest down the stepping registers. The stepping registers have between them a group of four cores (one core per register) to store each coded decimal digit of the three command-time digits; they have a group of four cores to store each coded decimal digit of the seven command-distance digits for each axis, and they have a group of four cores (some of which are redundant) to store each of the signs. So there is a total of 12 command-time cores, 140 command-distance cores, and 20 sign cores.

Fig. 3 shows the stepping register with the command time, 200 seconds, and with the command distance, —250,9645, stored as an example. Note that the arbitrary choice was made to use the same code for minus as for two. A limitation on the command-time code is that not more than one command-time core contains a binary "one." The reason for this will be seen as we progress. The allowable command times are 200, 100, 50, 20, 10, 5, 2, 1, and 0.5 seconds.

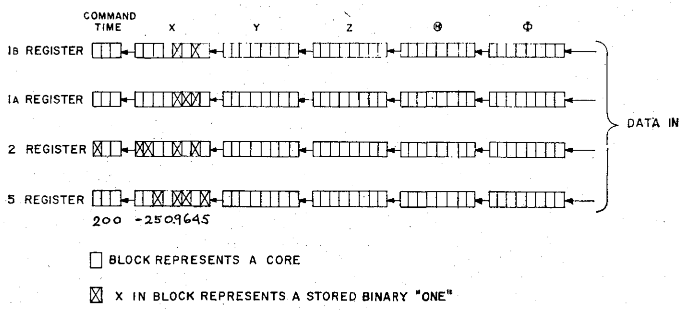


Fig. 3—Magnetic core shift register.

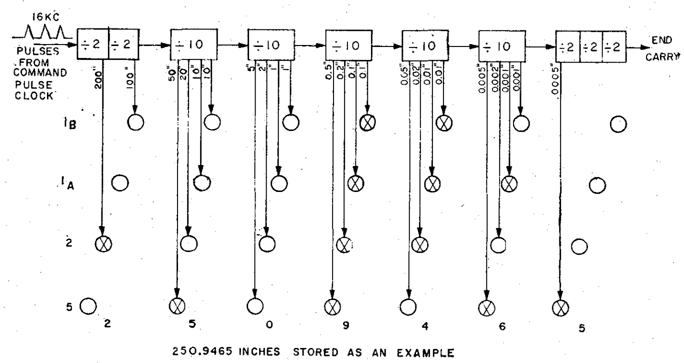


Fig. 4—Arrangement of interpolation counter in relation to storage cores of a typical axis.

Interpolation

Upon the occurrence of an internally generated signal, the contents of the dimersion-storage cores of the four stepping registers are transferred in parallel to a second set of storage cores. Simultaneously, the sign and command time cores are reset, causing pulse outputs from cores which had been storing ones. The stepping registers are thus freed to receive another block of information while the coded dimension numbers of the previous block are available to be operated upon in the second set of cores.

Let us consider of what this operation must consist. We are attempting to convert a coded number into a corresponding number of discrete pulses at a rate of one pulse per eighth of a thousandth, or a rate of four pulses for our least programmable distance, one half of a thousandth. Thus, let us interrogate nondestructively the core in which the half-thousandth bit is stored. If this core contains a binary "one," a pulse output will occur each time we interrogate it; if we interrogate it four times during the processing of the block, four output pulses will occur. We see that we can weight the binary digit stored in any core by fixing the number of times that that core is interrogated during the programmed command time. The cores storing binary "ones" weighted at 0.001 inch will be interrogated eight times during any command time; the cores storing binary "ones" weighted at 0.002 inch will be interrogated 16 times during a command time, a 0.005-inch core 40 times, and so forth. The outputs of the cores for one axis are buffered onto a common output line so that the command pulses on that line are a result of contributions from all the cores which have "ones" stored in them in that axis.

It would appear that the weighting functions for the core-stored dimension could be generated by a counting chain, and this is just what is done. A counting chain composed of cascaded binary and decade scalers is used. Fig. 4 shows how this counting chain, called the interpolator counter, is arranged with respect to the core storage for one axis. The decade circuits consist of four flip-flops connected so that, for 10 pulses entering a decade, there occur 5 carry and 5 noncarry transitions of the first flip-flop, 2 carry and 2 noncarry transitions of the second flip-flop, and 1 carry and 1 noncarry transition from each of the third and fourth flip-flops. The noncarry transitions of each flip-flop trigger an interrogate pulse which results in a command pulse out, if a binary "one" is contained in the magnetic core being interrogated. Although Fig. 4 shows the cores of only one axis, each flip-flop in the interpolator counter interrogates the five corresponding cores of the five axes. With a divide-by-four circuit beyond the half-thousandth flip-flop, the half-thousandth core is interrogated four times for every end carry. The end carry signals the end of the straight-line motion.

It is interesting to note two properties of the interpolation counter without which this system of linear interpolation would not work.

1) No two cores of any axis are interrogated simultaneously, and hence, the command-pulse contributions of the various cores appear as separate discrete pulses on the command-output line. This is because each oscillator pulse propogates down the chain as carry transitions until the first flip-flop ready for a noncarry transition is reached. The noncarry transition of that flip-flop does not result in any action farther down the chain. Hence, only one noncarry transition can occur anywhere in the chain for each input pulse to the chain.

2) It can be shown that, regardless of what pattern of "ones" and "zeros" exists in the storage system, that is, regardless of what number has been stored, the resulting pulse distribution is such that the displacement vs time for any axis never varies from a perfect ramp by more than one quanta.

Our command-pulse clock-oscillator frequency in the Numericord system is 16 kc. Referring again to Fig. 4, we see that 3.2-million oscillator pulses are required for each end carry. At an input rate of 16 kc, it would require 200 seconds to cycle through or cause an end carry. However, if we feed in our 16-kc clock pulses farther down the chain, it will require less time for the counter to cycle through. The Numericord system feeds clock pulses to nine gates, only one of which is open at a time. So pulses are fed to one of nine input points along the interpolator counter. The nine command times available are, as I previously mentioned, 200, 100, 50, 20, 10, 5, 2, 1, and 0.5 seconds. These gates are controlled by flip-flops, one of which is set by the pulse output of the appropriate command-time storage core.

The end carry from the interpolation counter clears the storage register and resets the command-time flip-flop so that no more oscillator pulses are admitted to the counter. When the clearing action is complete, the next command dimension is dumped in the storage register from the stepping register. Simultaneously, the nine command-time storage cores are reset resulting in an output pulse from the one core holding a "one." This pulse sets the appropriate command-time flip-flop. Clock pulses are now entered into the counter at a point in accordance with the new command time. Command pulses continue to appear on the output line after the end carry, but at a rate determined by the new command time and the new command distance. So there may be a discontinuity in command-pulse rate and, therefore, a discontinuity in command velocity at the end of one straight-line cut and the beginning of the next.

In programming an excessively large velocity step in any axis, the programmer may use a special code which will automatically reduce the clock rate as the end of the cut leading into the velocity step is approached. So the velocity step occurs at a much lower clock-pulse level and consequently results in a much lower velocity step at the machine tool. After the velocity step has passed, the clock rate rises to normal.

You can see in Fig. 5 that when clock pulses are entered into the counter at a point other than through the 200-second gate, not all cores will be interrogated. For instance, a 50-second command time results in the 100-inch and 200-inch cores not being interrogated. Therefore, a restriction must be placed on the programmer so that the programmed distance for 50 seconds is not more than 99.9995 inches in any axis. In fact, the restriction for any command time is such that the maximum vector component of feedrate in each axis is two inches per second.

Decoding

Since a synchro signal is well suited to the control of position, the Numericord system was designed to produce command synchro signals from the command-pulse outputs of the interpolator. This is done in the electronic phase-shift decoder. This decoder produces six 200-cps square wave outputs, one for each axis, plus one for a reference. The axis signals are phase shifted with respect to the reference by an amount proportional to the command distance. The decoder output is similar to the output of a rotary command synchro where the stator windings are excited by two or three phase reference voltages, and the phase of the rotor signal with respect to any one of the stator-phase voltages is proportional to the mechanical angle of the rotor.

The mechanism by which the phase shift is produced in

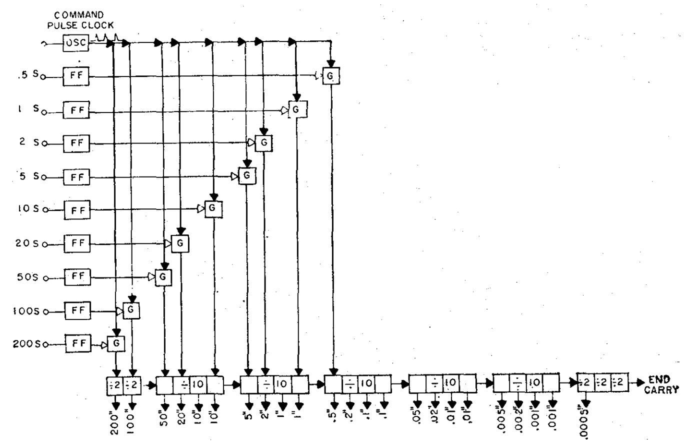


Fig. 5—Interpolation counter with command-time gates and flip-flops.

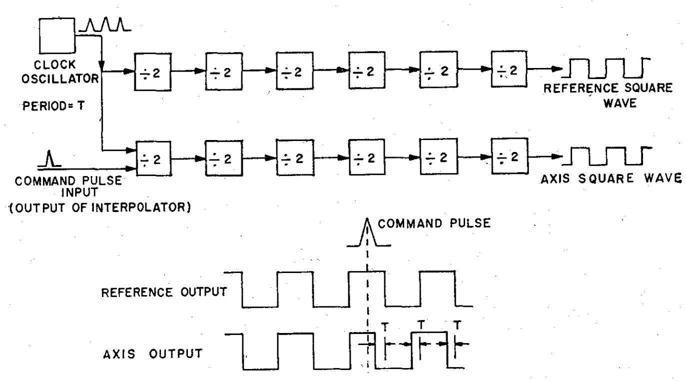


Fig. 6—Sample decoder.

response to a command pulse is best understood through reference to Fig. 6. Here, two binary-counting chains of equal length are shown. Both have inputs from a common high-frequency oscillator, the carrier-clock oscillator. One of these counting chains has an additional input which is the command-pulse line. If pulses only appear on the clock-oscillator input line, and if both counting chains initially start with all flip-flops reset to a common state, then the square-wave signals appearing on the plates of the last flip-flop in each chain will be of the same frequency and will be in phase with each other. Now, if a command pulse appears on the command line at a time between the occurrence of two clock pulses, the first flip-flop of this claim will become 180 degrees out of phase with the first flip-flop of the other chain. One less clock pulse will be required for this chain to cycle through a complete count than is required for the other chain. Therefore, the transition of the last flip-flop occurs sooner for this chain, by the amount of time between clock pulses. If no more command pulses appear, the two chains continue counting, and the phase of the square-wave signal of the last flip-flop in the second chain remains advanced by T microseconds with respect to the first chain. (T = clock-pulse period.) The first chain is the reference chain, and the square-wave signal from the plate of its last flip-flop is the reference-output signal and is essentially the signal which excites the stator of the feedback synchro. The second chain is the axis chain and its output is the phase-shifted signal to be compared with the phase of the rotor signal on the feedback synchro. The reference signal must, of course, be filtered to a sine wave and converted to 2 or 3 phase.

When the direction of machine-tool travel is negative, the command pulses are made to delete incoming clock pulses, one for each command pulse. So the phase of the axis counter lags the reference by further fixed increments with every command pulse.

In the Numericord system, there are five counting chains in addition to the reference chain so that five motions may be simultaneously controlled. Each chain is a combination of cascaded binary and decade scalers so that the total reduction of frequency is by a factor of 800 in each axis. The clock frequency is 160 kc and, therefore, the nominal output frequency is 200 cps. Since command pulses may be entered (added or subtracted) at the high-frequency end of the scalers at a rate of up to 16,000 pps, it is possible to modulate the output phase at a rate of 7200 degrees per second, or, in other words, to modulate the frequency at a rate of 20 cps. Each command pulse shifts the phase by one 800th of a cycle or by 0.45 degrees. Eight-hundred pulses or 0.1 inch of command causes the phase to shift one cycle.

The carrier-clock oscillator and command-clock oscillator are not synchronized. It could happen that a command pulse and a carrier-clock pulse could appear simultaneously at the input to a decoder axis counting chain if the precaution were not taken to avoid this. A circuit which we call the chronizer circuit prevents this from happening. In Fig. 7, we see that a command pulse sets a flip-flop to the "one" state, which after a short delay, opens a gate. The next carrier-clock pulse that occurs passes through this gate and resets the flip-flop. A pulse is produced at a fixed interval after the reset transition of the flip-flop. The time that this pulse can occur with respect to the time that carrier-clock pulses occur is determined by the setting of the delay. The delay is adjusted so that the pulse occurs between two carrier-clock pulses. The pulse will either be added to the carrier-pulse chain entering the axis counter, or it will generate a gating potential of sufficient length to prevent the next carrier pulse from entering the counter, depending on the state of the add-subtract flip-flop.

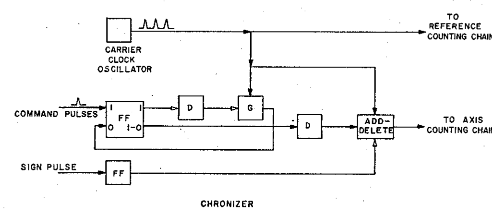


Fig. 7—Chronizer.

I have attempted to explain the operation of the essential feed-forward elements of the Numericord system. There are many auxiliary features, an explanation of which time does not permit. There is an indication scheme by which a continuous decimal-digit display of the actual phase between any axis and the reference is presented to the operator. There are area alarms which point to specific areas in the equipment when a fault occurs.

You can appreciate that the director is a very special purpose type of computer, and I think you will appreciate that much computation may be necessary in the initial paper-tape preparation. These computations include determining tool-center offsets, since it is the contour of the point of tangency between the tool and the workpiece that is of interest, whereas it is the path of the center of the tool that must be programmed. The computations also include determining the straight-line segments necessary to approximate a specified curve with a given degree of accuracy. A general-purpose computer lends itself to these computations, while the real-time problem of interpolating and rate generating is the special province of the director.

Design of a Numerical Milling Machine System

Y. C. HO† AND E. C. JOHNSON†

I. General Considerations

A. Basic Concept of Numerical Control

ADVANCED data-processing and control techniques can be used in many ways to effect substantial improvements in present manufacturing processes. A good example is the application of numerical control to machine tools. As currently used in this connection, numerical control describes loosely the concept of operating machine tools from information recorded on punched cards, punched tape, or magnetic tape. The recorded information may or may not be in digital form. If it is not, however, the control record is generally produced from numerical data by equipment which is considered to be part of the system. Hence the use of the word "numerical."

The underlying objective in applying numerical control to machine tools is improvement in the over-all process of producing finished parts from basic design information. Improvements commonly sought include greater accuracy and reproducibility of the part, increased machine productivity, reduction in tooling costs, reduction in skilled manpower required, and over-all shortening of the manufacturing cycle. Sizable gains can be realized through attention to any of several specific problem areas. However, maximum benefits are to be expected only if the entire manufacturing process is considered as an integrated system.

The proper starting point for such an approach may well be in the design stages which immediately precede manufacturing. Actually, serious thought has been given to the use of modern data-processing techniques in mechanizing the design process itself.[1] However, little progress has been reported to date except where the design process is at least partially analytical already, as with certain types of cams. For the most part, therefore, in contemporary numerical machine-tool systems it is presumed that a more or less conventional engineering drawing of the part to be made is available.

A further stage of manual effort, referred to here as process planning, likewise appears in present systems. This is concerned with the development of data pertaining to the metal-cutting aspects of the job: the manner in which the part is to be mounted on the machine, the sequence of cuts to be made, the amount of metal to be removed in each cut, the cutter size and shape, the feed rates, etc. Since most of this information is derived from the part drawing, utilizing past experience, it might be expected that it would be possible to mechanize this task. Practically, however, mechanization has not yet been found feasible because of the large number of decisions to be made and the difficulty of defining suitable criteria. The input to a numerical manufacturing system proper therefore consists of two basic kinds of information: geometrical data and machining instructions.

At the output end of the system is a machine for physically producing the part. The result most ideally is a part on which all machining operations are complete. It may even be desirable to consider that automatic inspection of the part is included as well. As a practical matter, however, it may be necessary to accept far less, recognizing that some finishing operations may be required either by hand or by machines essentially unrelated to the numerical process. Likewise, inspection may be entirely separate. With the latter reservations in mind, a numerical manufacturing system of the type being discussed can be represented symbolically as in Fig. 1.

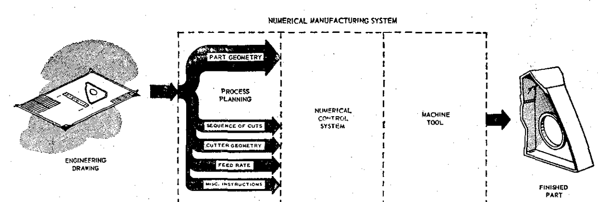


Fig. 1—Symbolic representation of a numerical manufacturing system.

B. Path Control of a Milling Machine

In some machining techniques such as drilling and broaching, the shape of the cutter determines the shape of the work piece. Control of such machines is largely a matter of positioning the cutter to the proper location between operations. Milling and turning, however, are techniques in which the shape of the part surface may have little or nothing to do with the shape of the cutter. The surface is determined rather by the relative motion between cutter and work piece.

The amount of control information necessary in such a process depends not only on required accuracy in the usual sense, but also on surface finish. Generally speaking, surface finish has to do with uniformity in relatively small areas; accuracy has to do with absolute location. Irregularities permitted by the surface-finish requirement are frequently a factor of ten or more smaller than the errors which are tolerated from accuracy considerations alone.

† Bendix Aviation Corp., Detroit, Mich.

[1] G. R. Price, "How to speed up invention," *Fortune*, p. 150; November, 1956.

Hence basic resolution provided in the motions of the machine must usually be many times better than implied by the accuracy requirement.

Coarse control information would be filtered to some extent by unavoidable lags in the machine drives. Such smoothing might be completely satisfactory at high velocities, or feed rates, but completely ineffective at low rates. In addition, the dynamic performance of the machine drives should be high for other reasons—to maintain low following errors on complex contours at reasonable speeds, and to resist load forces generated by the cutter over a wide-frequency range. Consequently dynamic filtering of coarse control information by the machine drives cannot generally be relied upon to provide the necessary smoothness of motion.

Another factor tending to increase the amount of information required in the control of a milling machine is the relative inefficiency of the scanning procedure used conventionally to generate such shapes as the one illustrated in Fig. 2. Little if any advantage can be taken of the similarity of cutter motions on successive passes. To make matters worse, the spacing of the passes may be dictated by the need to keep the scallops produced by the cutter within the limits allowed by a tight surface-finish requirement.

The problem is somewhat analogous to that encountered in the production of a television image which is pleasing to the eye. The large amount of redundancy between adjacent lines in a frame is recognized. Theoretical considerations clearly indicate that substantial improvement is possible.

However, techniques which would avoid this redundancy and still be practical on an economic basis are yet to be demonstrated.

Thus the amount of information required to control a milling machine, using conventional cutters and procedures and with no built-in smoothing other than that provided by the usual dynamic lags on the drives, is quite high. The *rate* of information flow depends on machine feed rates. These may range from fractions of an inch per minute to several hundred inches per minute. A theoretical upper limit to the usable information rate might be established in terms of the dynamic capability of the machine drives. There would seem to be little point, for example, in changing the position command at a rate appreciably higher than that at which the machine drive servos can respond. Larger steps at a less frequent rate may give just as satisfactory results. Taking full advantage of this principle may involve more expense in terms of additional hardware than could ordinarily be justified. The "pulse-multiplication" scheme described in Section III approximates this, however.

C. Computational Problems

As mentioned earlier, the process-planning stage, while perhaps theoretically amenable to modern data-processing techniques, practically has not yet been found to be so. However, several other problems of a computational nature definitely are subject to mechanization.

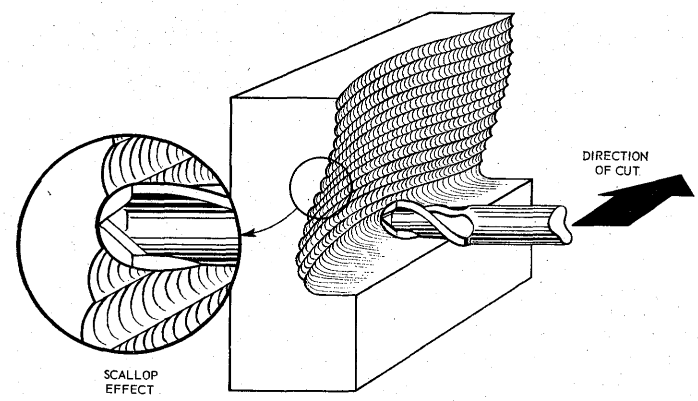
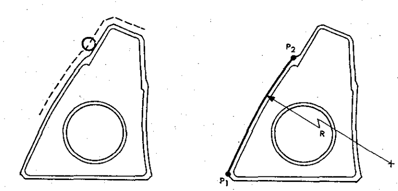


Fig. 2—Scanning nature of three-dimensional contour milling.



(a) (b)

Fig. 3—Illustration of two major computational problems. (a) Cutter-center offset. (b) Interpolation or path generation.

One of the most significant of these is the cutter-offset problem. As previously noted, complex surfaces are generated in milling by relative motion between cutter and work piece. In the usual case, it is the cutter *axis or center* that is directly controlled. The surface of the work piece, however, is produced by the *periphery* of the cutter, Fig. 3(a). A translation from part-design information is therefore necessary to determine a cutter-center path which will produce the desired surface.

Another major problem is that of path generation, or interpolation. Fig. 3(b) illustrates a very common situation in which a portion of the part to be made is a circular arc. End points are located and the radius is specified. Somewhere in the manufacturing process there must be the ability to establish from this kind of information an essentially continuous sequence of cutter-center positions sufficient to generate the curve.

Although other problems such as optimum control of feed rate can be considered, the cutter-offset and interpolation problems seem to be the most fundamental. In general, either of the problems can be difficult and tedious to handle manually. The economic success of numerical machining appears to hinge on the development of effective techniques for dealing with these problems automatically.

D. System Organization

A numerical manufacturing system can be divided into any number of physically distinct functional elements, provided suitable means are available for communication between elements. The last element in the system, as presently conceived, is a machine tool of more or less conventional configuration (assuming that there is no radical departure in metal-removal techniques). A control element of some description will necessarily be direct-coupled to the machine. This element may be a full-scale computer-control device for performing all data-processing operations at the machine, or it may be little more than a data-receiving device. Between the machine control unit and the input end of the system, many variations are possible.

Although computing techniques are common at present, a basic incompatibility exists between the instantaneous time scales required for some calculations, such as cutter offset, and the rate of control information required by the machine to achieve uniform and continuous operation. A computer which is fast enough to meet the peak information rate on a real-time basis would be far more powerful than necessary on the average. Isolation of the difficult computational tasks from machining proper is therefore advisable. This implies that there should be at least one element in addition to the machine and its control unit.

Other principles can be formulated from the data-handling viewpoint which under some circumstances can serve as useful guides. For example, consider the two computational problems discussed in the preceding part. That of cutter offset generally does not produce additional information; the cutter-center surface is not usually significantly more complex than the part surface. The process of interpolation, on the other hand, is basically one of generating additional data. Other things being equal, therefore, cutter offset should be performed before interpolation in order to avoid having to transmit and operate on an unnecessarily large volume of data.

Other considerations lead to basic conflicts. It may be feasible to perform the purely data-processing operations at a rate which is significantly faster on the average than the rate at which metal can be removed. A single data-processing facility could then service a number of machines. In this case, all possible data processing should be done away from the machine and its directly connected control unit. This approach is attractive from the viewpoint of simplifying the equipment required at the machine. However, as emphasized previously, the control of milling-machine motions requires a very large amount of data. In the approach just described, all of this information would be transmitted to the machine, none of it being generated in the machine control unit. The result may be extremely bulky control records and expensive and complex equipment for their generation.

E. A Solution to the Systems Problem

From a functional point of view, present numerical-control systems differ mainly in the manner in which they are subdivided physically, the operations assigned to each element, and the data-transmission links between elements. Because of the numerous conflicting considerations, obviously no system can be optimum in an absolute sense. The system illustrated pictorially in Fig. 4, however, meets to a high degree the over-all objectives outlined in Section I-A.

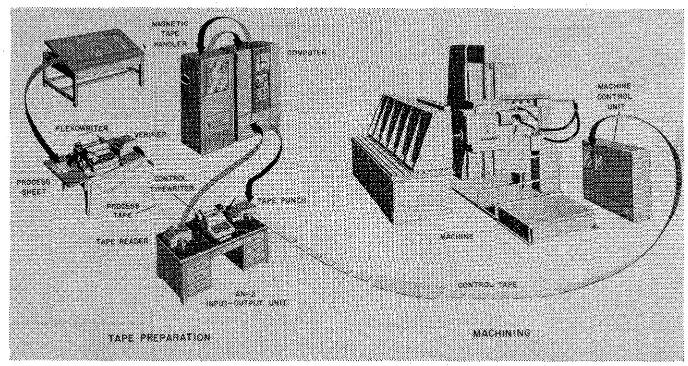


Fig. 4—Pictorial diagram of the Bendix numerical milling machine system.

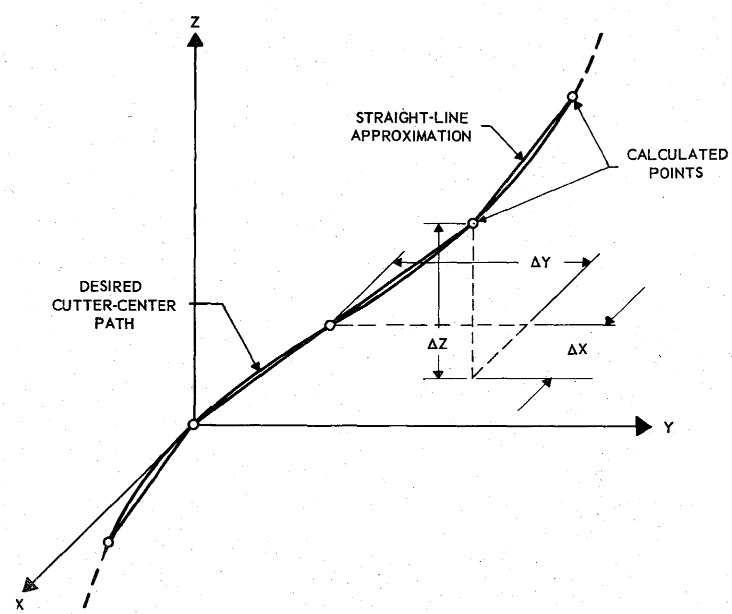


Fig. 5—Straight-line approximation to a curve.

This system comprises two major groups of equipment. One consists of the machine and its directly connected control unit. The other involves a small, general-purpose, digital computer for data preparation. The basic conflict between the required large volume of control information and extensive computation at the machine is resolved by building into the machine control unit the ability to perform an elementary straight-line curve-fitting or interpolating operation. With coordinate differences provided along an arbitrary cutter-center path, as indicated in Fig. 5, the machine control unit in effect generates the connecting straight lines.

The remaining computation—compensation for cutter geometry and gross interpolation of curves and surfaces—is performed by the data-preparation equipment. Linear interpolation at the machine is relatively economical, is compatible with the peak rates of control information de-

Fig. 6—Computer, magnetic-tape unit, and input-output unit.

Fig. 7—Flexowriter-Verifier for preparation of process tapes.

manded by the machine, and, most important, very greatly reduces the amount of information which must be transmitted to the machine control unit by way of the control record. The volume of information is reduced to the point where punched tape of conventional configuration is feasible as the control record. The system is digital throughout up to the point of producing operating signals for the machine drives.

II. Data Preparation

The heart of the data-preparation portion of the system is the Bendix G-15D general-purpose computer. An automatic programming system called COMPAC (COMprehensive Program for Automatic Control) enables the computer to accept raw dimensional data and machining instructions in a language familiar to the process planner. This section gives a brief description of the equipment and discusses the design of the automatic program.

A. Equipment

1) Computer: The Bendix G-15D (Fig. 6) is a medium-speed computer, operating serially with a magnetic drum of 2160-words capacity.[2] Two noteworthy features make it particularly suitable for this application. First, the G-15D instruction has a micro-programming structure consisting of seven independent parts. The effectively two-address nature of the instruction facilitates minimum-access coding to achieve maximum computing speed. Second, input and output can proceed simultaneously with computation through built-in buffer-storage registers. Both features were exploited in the design of the program.

2) Auxiliary Equipment: Three pieces of auxiliary equipment supplement the computer in the data-preparation system. They are a Friden Flexowriter-Verifier, a special input-output unit, and a magnetic-tape unit.

[2] H. D. Huskey and D. C. Evans, "The Bendix G-15 general purpose computer," *Proc. WESCON Computer Sessions,* pp. 87-91; August, 1954.

Information describing the part and its machining process is introduced into the computer by way of a punched tape (the process tape). The production of this tape is accomplished by the Flexowriter, an electric typewriter equipped with a tape reader and a tape punch (Fig. 7). Input data as furnished by the process planner on a handwritten manuscript, or process sheet is copied on the Flexowriter. As a by-product of this typing, a punched tape is produced which contains in coded form the process information. An additional tape reader, shown in the background of Fig. 7, acts as a verifier to check the correctness of the tape in a separate typing operation. The tape then serves as direct input to the computing system. The Flexowriter may also be used to duplicate process or control tapes.

The input-output unit (Bendix AN-2) consists of a punched-tape reader, a tape punch, and a control desk housing some electronic circuitry. The reader accepts process tape in standard Flexowriter code and translates it to straight binary. The punch produces control tape directly as output from the computer. The special format used on the control tape may also be read back into the computer by means of the input-output unit.

The magnetic-tape unit (Bendix MTA-2) supplements the internal memory of the computer and provides permanent storage for all computer programs. Both the input-output unit and the magnetic-tape unit are shown in Fig. 6 with the computer.

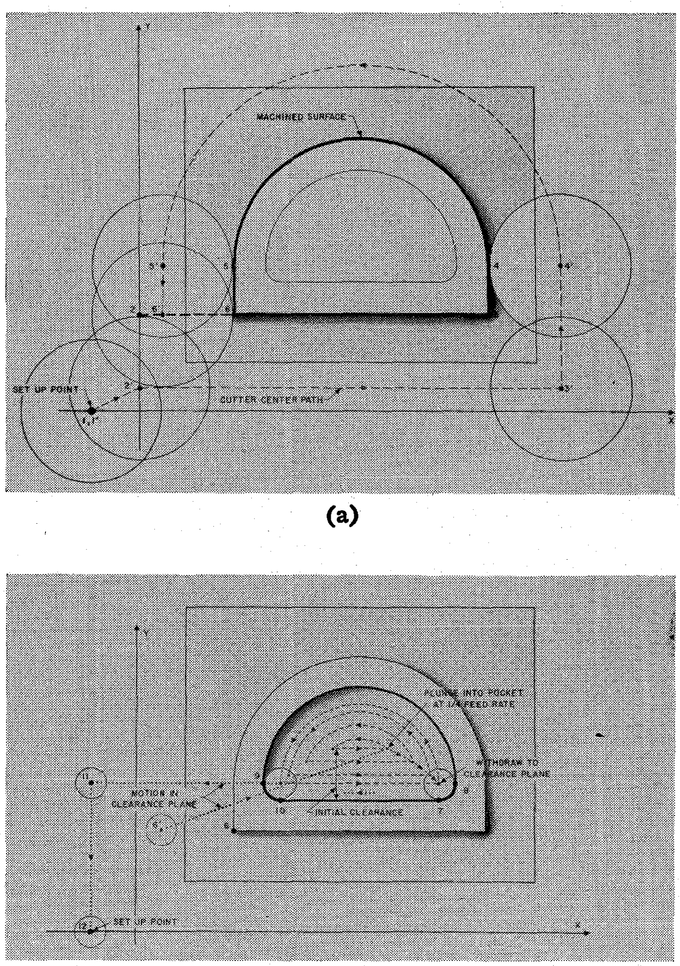
B. Program Design

The basic objective of COMPAC is the translation of engineering information which can be interpreted by human beings into information of a form which can be recognized by the machine control unit. Two essential functions involved in this process are coding and computing. With respect to the former, the problem in program design is to determine the type of numerical-control instructions to be built into the program and the format of the input data. Here, convenience of use from

	Setup Point X	Setup Point Y	Setup Point Z	/	Clearance Plane z_0	Tolerance	Type of Approx.	Date	Part No.
	IN.	IN.	IN.		IN.	IN.		9-6-57	E.I.A. PART
								Planner	Tape No.
(1)	-001.5000	000.0000	000.7500		001.2500	000.0002	1	Y.C. HO	Sheet 1 of 1

	End Point of Section X	Y	Z	/	Initial Clearance	Circle Radius	Arc Length	Corner	Type of Point	R	Feed Rate	Tool Diam.	R	Aux. Func.	R	Final Cut	Rough Cut Passes	No. of Sections	R
	IN.	IN.	IN.		IN.	IN.				00	IN./MIN.	IN.	40		41	IN.			01
	PRINT																		
	20 (TAB)	45 (cr)																	
											030.0000	004.5000	40						
														05	41				
(2)	000.0000	003.0000	000.7500				0	1	-2	00									
(3)	011.0000	003.0000	000.7500				0	1	2	00									
(4)	011.0000	004.5000	000.7500				0	1	2	00									
(5)	003.0000	004.5000	000.7500			-004.0000	1	1	2	00									
(6)	003.0000	003.0000	000.7500				0	1	2	00									
											030.0000	-001.0000	40						
														04	41				
																000.2500	03	04	01
(7)	009.5000	004.0000	000.7500		001.6000		0	-0	-2	00									
(8)	010.0000	004.5000	000.7500		000.5000	-000.5000	1	0	2	00									
(9)	004.0000	004.5000	000.7500		001.6000	-003.0000	1	-0	2	00									
(10)	004.5000	004.0000	000.7500		000.5000	-000.5000	1	0	2	00									
(11)	-001.5000	004.5000	001.2500				0	0	0	00									
(12)	-001.5000	000.0000	000.7500				0	0	0	00									
	END																		

Fig. 10—Format of the Bendix COMPAC-1 process sheet.



(a)

(b)

Fig. 11—Part to which the process sheet shown in Fig. 10 applies. (a) Profiling outside of D. (b) Pocket milling inside of D.

discussion of the processing for this part has been presented elsewhere.[3]

The design of this program provides a highly systematic method of introducing the part design data. The process planner needs to know little of actual computer programming beyond the functions performed by each of the eight operation codes. A somewhat different approach includes a greater number of more basic operation codes. Although potentially more flexible, such an approach appears to demand more of the process planner in an area at present unfamiliar to him. At the expense of some flexibility, the COMPAC programming system tends to offer easier and more inclusive instruction codes in preference to codes of a microprogramming nature.

The choice of entering data together with operation codes is prompted by somewhat similar considerations. Valuable storage space is saved when only a small amount of information is stored in the computer at one time. Although this necessitates the occasional repetition of identical data points in the same program, it eliminates the possibilities of error by the process planner in attempting to identify specific points with numerical-control instructions not entered simultaneously.

An over-all flow diagram of the COMPAC system is shown in Fig. 12.

2) Computation Techniques: The preceding describes the functional aspects of the program. In this part the mechanization and execution of these functions are discussed. The four major steps in this computation process are outlined below.

[3] E. C. Johnson, "Bendix tape preparation system," *Proc. EIA Symp. on Numerical Control*, p. 63; September, 1957.

the process planner's viewpoint is of prime importance. In computing, on the other hand, the problem is to make most effective use of the computer, considering such characteristics as storage capacity and computing speed. These two problems are invariably conflicting and require certain compromises. Over-all design of the program is discussed from both viewpoints.

1) Instruction Codes and the Process Sheet: The form of the instruction code is shown in Fig. 8. It consists of a variable-length data field and a fixed-length operation code. The data field may contain up to 600 decimal digits with signs. The operation code is a two-digit number with sign. A negative sign indicates that the following operation requires use of new routines stored on magnetic tape. The computer then proceeds to load the appropriate program as specified by the code. Once the program is loaded, operation codes again are given a positive sign until another search on magnetic tape is required. This scheme allows for practically unlimited expansion of the program. The present description, however, is limited to the eight operation codes comprising COMPAC-1, which are handled by routines stored exclusively on the drum of the computer. The meaning of these codes is explained below.

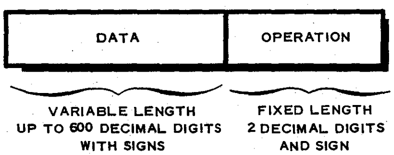


Fig. 8—Form of COMPAC instructions.

Profile milling—"00": Data preceding the 00 code describe a circular or straight section of the profile of a part. The computer calculates the cutter-center path required to produce this profile, and punches the results in the form of control tape. Computation of an appropriate intersection with the cutter-center path of the following section, and automatic deceleration at a corner are included.

Pocket milling—"01": Data preceding this code specify the number of subsequent profile-milling codes that define the boundary of the enclosed pocket or profile, the number of roughing cuts required to clean out the pocket, and the amount of material to be removed in the final cut. The computer produces a control tape that completely machines the pocket on the basis of information defining its boundaries only. All computational features provided in profile milling are carried over to pocket milling as well.

Feed rate and tool diameter—"40": A new feed rate or tool size to be used in subsequent computations is indicated by information preceding the 40 code.

Auxiliary function—"41": Data preceding this code define special control-tape characters that control on-off functions at the machine. Two examples are shown in Fig. 9.

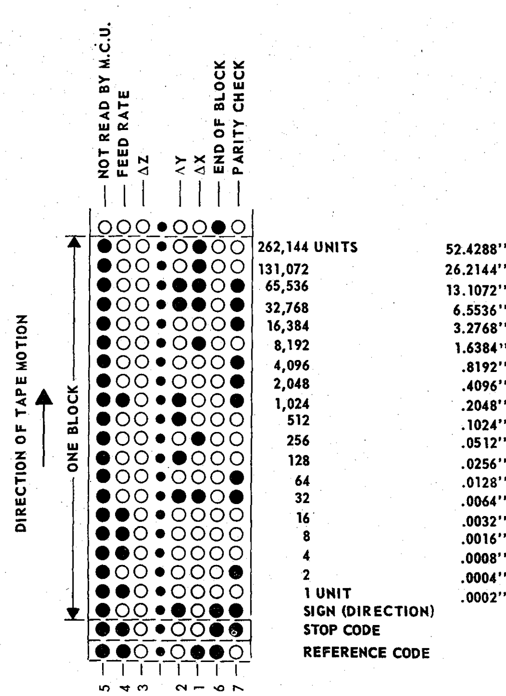


Fig. 9—Control-tape format illustrating command for motion along X axis of $+100$ inches and along Y axis of -20 inches.

Print—"17": The print code initiates the type-out of cutter-center coordinates at the end points of circular and straight sections during computation. The results are presented in inches with the decimal point inserted. This feature is used primarily as a means of monitoring the tape-preparation process.

Print suppress—"34": This code cancels the "print" code.

Print chord—"45": The print-chord instruction causes cutter-center coordinates corresponding to intermediate points on circular arcs to be typed out. The number preceding the code determines the frequency. For example, 15 45 causes every 15th point to be printed.

End—"0X": The end code causes the computer to cease reading process tape, complete the computations, and punch a machine stop code on the control tape.

These instructions (data and operation codes) are entered by the process planner on a work sheet referred to as a process sheet. Its format is shown in Fig. 10. The process sheet is merely a disguised form of coding sheet for the program. The act of filling in the blanks automatically supplies the data and operation codes in the proper format.

The order of the instructions defines the cutting sequence. One line of dimensional information is supplied for each section, either straight or circular, making up the boundary. The information on the process sheet of Fig. 10 applies to the part shown in Fig. 11. Fig. 11(a) shows the outside profiling specified by the first 9 lines on the process sheet proper, and Fig. 11(b), the pocketing-milling portion defined by the last 9 lines. A more detailed time Δt is computed for the movement. If this cutting time is shorter than the reading time at the control unit for the next block of tape, the cutting time is automatically increased. From the adjusted Δt, the feed-rate number is computed:

$$F = k \frac{2^N}{\Delta t} \cdot$$

The system constant k is proportional to the clock frequency in the machine control unit. The exponent N is the number of binary digits in the longest of the three incremental motions Δx, Δy, and Δz, expressed in basic machine units. The reason behind this particular form is discussed in Section III.

3) The control tape employs a serial bit, parallel-word, variable-length format (Fig. 9). Since normal computer tape-output codes are parallel-bit, serial-word, output blocks must be assembled before punching. Parity bits and end-of-block codes are automatically inserted by the AN-2 input-output unit.

Approximately 350 instructions are used in this portion of the program. An extra 100 instructions are required for the type-out of cutter-center coordinates.

Supervisory routine: The supervisory routine provides all control links between the routines mentioned above. Two major paths are those of profile milling and pocket milling. The latter actually is a loop over the former. The parameter being varied each time around the loop is the effective cutter radius. This is decreased from some starting value (initial clearance) to the actual cutter size on the final pass around the loop. The total number of instructions involved is approximately 700. Temporary storage of data and intermediate results requires another 300 locations.

3) Computing Speed: All input-output operations during tape preparation are time-shared with computation. Each line of the process sheet requires approximately seven to ten seconds to be read, processed, and the corresponding block of control tape punched. Additional points on a circle require about 3.5 seconds to compute and punch.

4) Miscellaneous Service Routines: Several special service routines have also been developed for use with the system. One of these enables the computer to read a control tape prepared by the COMPAC program and type out the accumulated cutter-center locations and feed rates. General-purpose automatic-programming systems are likewise available for the G-15D which facilitate the handling of data not otherwise directly suitable as input to the present COMPAC program.

III. The Machine Control Unit

A. Block Diagram of the Control Unit

Section II discusses the data-preparation aspects of the numerical-milling system. The operations involved here lead to machine control information recorded on a punched tape in the form of variable-length blocks as illustrated in Fig. 9. The four channels of computed information (1-4) contain binary numbers which specify the number of pulses, or basic units of motion, required along each machine axis and the speed at which the movement is to take place. These numbers are referred to as the x, y, and z distance commands and the feed-rate number, respectively. The tape thus produced by the data-preparation equipment serves directly as input to the machine control unit.

Functionally, the present control unit is similar to the one developed at M.I.T.[4] It consists of two principal sections: the interpolator and the servosystem. The principal job of the interpolator is the resolution of the coded distance commands into discrete pulses to drive the digital servos. The rate of pulse generation is controlled in accordance with the feed-rate number as furnished by the control tape. The servos then convert these pulses into actual machine motion.

In a typical application, each pulse represents a unit motion of 0.0002 inch. At this scale factor, the control unit being described is capable of calling for a maximum speed of 240 inches per minute along each axis (20,000 pulses per second). The maximum length of a block of information on the control tape is 22 lines, providing for machine strokes of over 400 inches. These features are obtained using serial computing techniques in the interpolator, and essentially parallel logic in the servosystem.

Fig. 18 is a detailed block diagram of the control unit. The interpolator consists largely of recirculating registers for storage of distance commands and feed-rate numbers as read from the control tape. Two additional recirculating registers are used as serial counters with appropriate control logic for the generation of pulses. The digital servosystem is identical for each axis. It involves a reversible counter which keeps continuous track of the difference between the number of command and feedback pulses. A digital-to-analog decoder is used within the servo loop to develop the error signal. Other logic of a supervisory or operational nature is also contained in the unit.

B. The Digital Interpolator

The variables to be controlled for a given block of tape are the distances to be moved along each axis and the velocities at which these movements are to take place. In the interpolator to be described, control of the velocities is best thought of in terms of control of the time interval allowed for the specified motions. More accurately, therefore, the variables actually being controlled are Δx, Δy, Δz, and Δt. The method used in controlling cutting interval, Δt is discussed first.

1) Cutting-Interval Control: One of the easiest ways to measure time in digital systems is by counting an accu-

[4] J. O. McDonough and A. W. Susskind, "A numerically-controlled milling machine," *Proc. Joint Computer Conf.*, pp. 133-139; December, 1952.

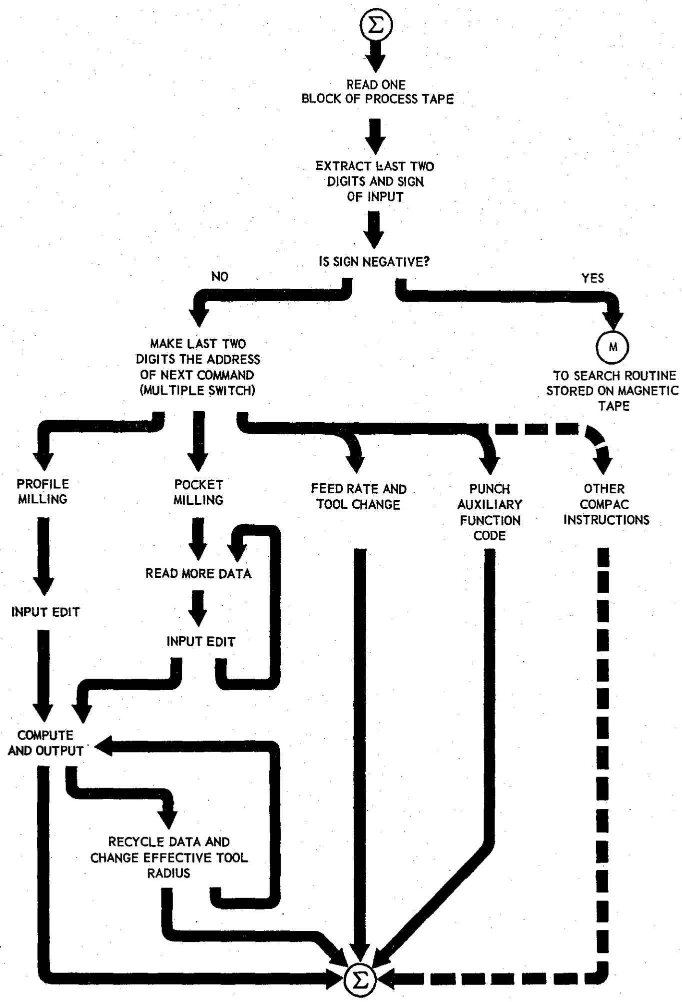


Fig. 12—Over-all flow diagram of COMPAC system.

Input-edit routine: In the input-edit routine, the information shown on the process sheet is converted to binary form and arranged in a fixed format at predetermined locations for further processing by later portions of the program. About 150 instructions are used in this routine.

Geometric routines: There are two basic geometric routines used in COMPAC-1. One finds the intersection between two straight lines; the other finds the intersection between a straight line and a circular arc. The latter is used to find the intersection between two circular arcs with a simple modification of the input data. These two routines are the basis for practically all computations.

1) Computation of an offset point from a point on the part surface, Fig. 13.
2) Computation of end points of a cutter-center section, Fig. 14.
3) Computation of an automatic slowdown block at the end of the cutter-center section, Fig. 15.
4) Computation of the center of a circle, Fig. 16.
5) Computation of intermediate points on a circle, Fig. 17.

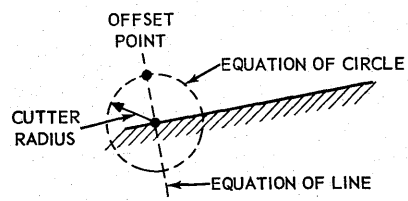


Fig. 13—Computation of an offset point from a point on the part surface.

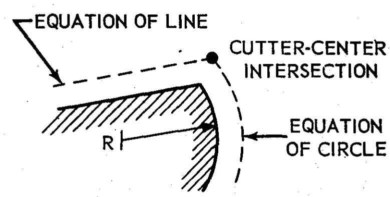


Fig. 14—Computation of end points of a cutter-center section.

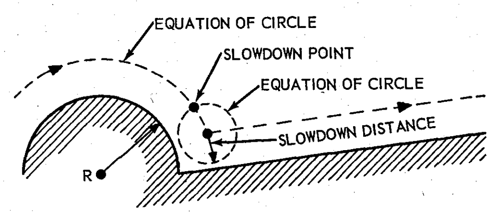


Fig. 15—Computation of an automatic slowdown block at the end of the cutter-center section.

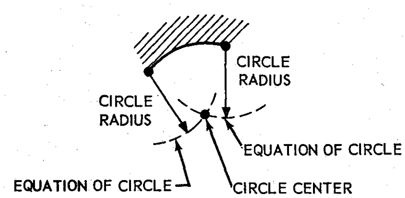


Fig. 16—Computation of the center of a circle.

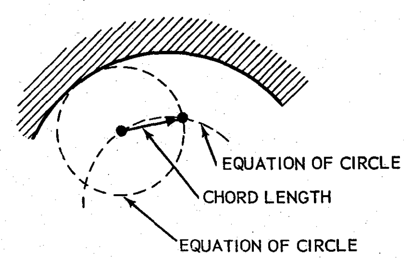


Fig. 17—Computation of intermediate points on a circle.

Approximately 300 instructions are required in this portion of the program.

Output routine: After a new point is determined on the cutter-center path, further processing is necessary before a block of output tape can be punched. The following principal steps are involved.

1) The differences Δx, Δy, and Δz, between the current tool location and the desired new end point are computed.

2) On the basis of the specified feed rate and the length of the cut $\Delta s = \sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta z)^2}$, total cutting

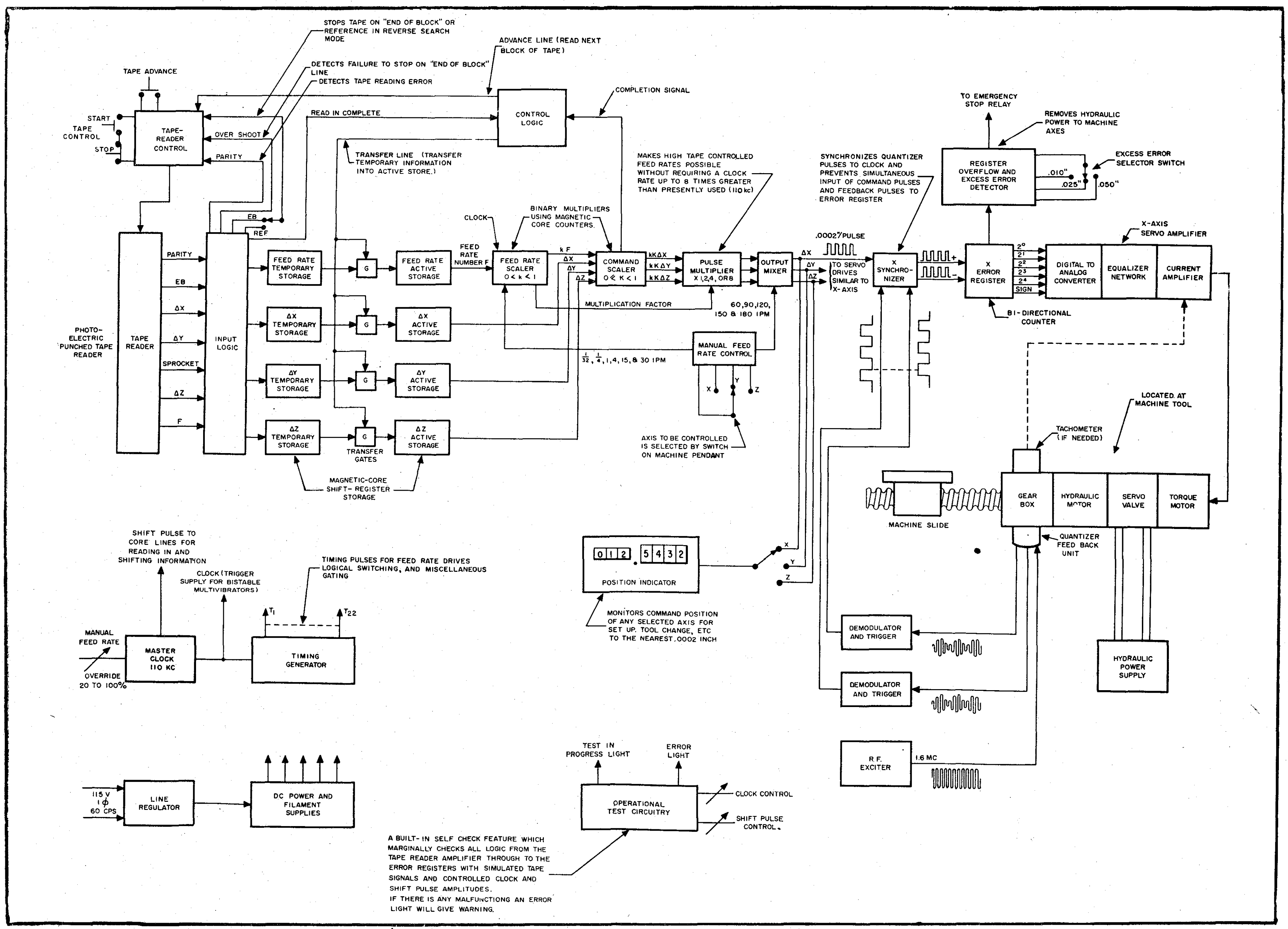


Fig. 18—Block diagram of machine control unit.

rately controlled pulse rate. Consider the serial counter labeled command scaler in Fig. 18. A count up to overflow starting with zero will determine an interval directly proportional to the clock frequency. If the number of stages in the command scaler is changed, the time to overflow will vary by powers of two. If, in addition, the rate of counting is changed, a further variation in the time to overflow can be achieved. If the time to overflow of the command scaler is defined as the cutting interval, then by combining the two above mentioned techniques, a very large range of cutting intervals is possible.

In the Bendix system, the length of the counter is arranged to be equal to the length of the binary number representing the largest of the three movements to be produced. For example, a movement in space of $\Delta x = 6.5336$ inches (2^{15} pulses), $\Delta y = 0.8192$ inches (2^{12} pulses), and $\Delta z = 0.0000$ inches (zero pulses) determines a command scaler of 16 stages. The rate of counting in the command scaler is controlled according to the feed-rate number. This is calculated from:

$$F = k\frac{2^N}{\Delta t}$$

as described in Section II-B, 2. The reason for this particular form is now clear: the feed-rate number merely controls the input pulse rate to the command scaler so that it overflows in Δt seconds.

The control of pulse rate is accomplished by a circuit commonly known as a binary operational multiplier. This device has two inputs, one of which is a pulse train and the other a binary number. The output is another pulse train, scaled down in the rate proportional to the fraction represented by the binary number. The basic principle of the multiplier is well known.[5] However, the circuit used in this system is novel in that it utilizes serial computing techniques.

Fig. 19 shows schematically the multiplier which varies the counting rate of the command scaler according to the feed-rate number specified. Two delay-line type recirculating registers are used. One is operated as a serial counter (that called feed-rate scaler in Fig. 18) by recirculating its contents through a half adder into which one timing pulse is added each cycle. The first "one" produced by the adder during each circulation is termed the noncarry pulse. This pulse is used to gate the contents of the other recirculating register which contains the feed-rate number, most significant bit first. The resulting pulse train then serves as input to the command scaler.

2) Cutting-Distance Control: The command scaler, being another serial counter, also generates one noncarry pulse each time an input pulse is added. If these noncarry pulses are used to gate the output of the Δx, Δy, and Δz recirculating active-storage registers, the gated outputs over a complete overflow period (2^N input pulses) will be Δx, Δy, and Δz pulses respectively. The pulses generated for each axis are essentially uniformly distributed throughout the counting interval due to the linear interpolating properties of the binary operational multiplier. The control of distances is thus accomplished.

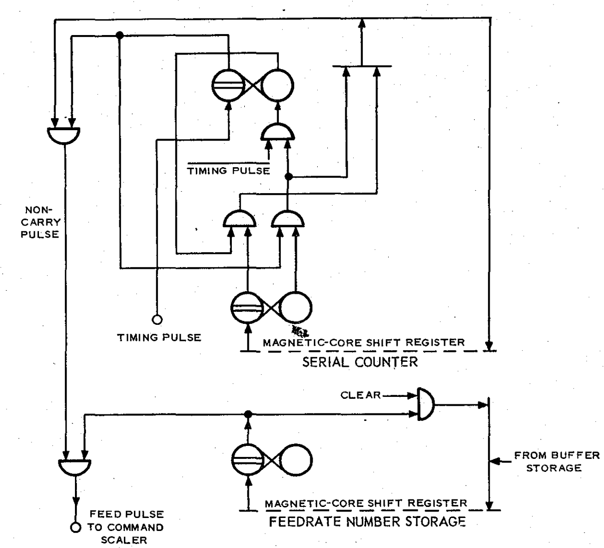


Fig. 19—Binary-operational multiplier using magnetic-core shift registers.

3) Pulse Multiplication: The previous two parts describe the use of two cascaded binary operational multipliers for control of the number and rate of pulses generated in any given interval. In this scheme a maximum of one command pulse is produced on each circulation of the command scaler. Actually, since any binary combination is possible in the distance commands, the output pulse rate may be as low as half of the recirculation rate.

The recirculating registers used in this system are 22 bits long. A complete circulation therefore requires 22 clock pulses. In order to achieve the high command-pulse rate of 20,000 per second previously mentioned, without an unduly high clock rate, a means for pulse multiplication is provided in the control unit. As described below, multiplication becomes effective only at relatively high pulse rates.

The action of the pulse multiplier can best be described by way of an example. Consider a case in which the feed-rate number is 1023 (2^{10}-1) and Δx is 1.6384 inches (10,000,000,000,000 or 2^{13} pulses). The input pulse rate to the command scaler will be 5000 per second at a clock frequency of 110 kc. The output pulse rate from the command scaler will be scaled down by two due to the configuration of the distance command as a binary number. Thus the maximum rate that could be achieved would be 2500 pulses per second, corresponding to a feed rate of 30 inches per minute.

If the feed-rate number turns out to be greater than 1023, as a result of a request for a higher feed rate, the pulse multiplier becomes operative. Suppose, for example,

[5] M. A. Meyer, "Digital techniques in analog systems," IRE Trans. on Electronic Computers, vol. EC-3, pp. 23-29; June, 1954.

the feed-rate number turns out to be 2047 (2^{11}-1). Overflow detection is then carried out one stage earlier in the command scaler, thus terminating the process when only half as many pulses are generated. The cutting interval, or time to overflow, is thereby halved.

The pulses now are all multiplied by two before being transmitted to the servosection. The resultant number of pulses therefore remains the same as in the previous example. Since the cutting interval is halved, the feed rate is effectively doubled.

The feed-rate number is still treated as a 10-bit number with the binary point automatically shifted one bit to the left. For feed-rate numbers of 12 or 13 bits, the outputs from the command scaler are multiplied respectively by four or eight. With this technique, command pulse rates as high as 20,000 per second can be produced with a clock rate of only 110 kc. A smooth transition to these high feed rates is automatically provided in the scheme.

When the desired feed rate is sufficiently high as to require pulse multiplication, the number of pulses generated must be a multiple of two, four, or eight. Consequently, the distance commands must be rounded off to 0.0004, 0.0008, or 0.0016 inch, respectively. To prevent errors of this type from accumulating from one block of control tape to the next, the computer program described in Section II keeps track of such round-offs. It continuously applies corrections from one block to the next so that at any one point the computed distance commands cannot be off by more than the current round-off quantity. Situations in which this round-off procedure would tend to produce errors in the part surface also result in transient servo errors. At the high feed rates involved, the servo errors can be expected to be many times larger than the round-off errors. Likewise, under steady-state conditions, the pulse-multiplication technique does not generally produce a significant effect. Few servos, for example, have dynamic characteristics capable of detecting the difference between a uniform command pulse rate of 20,000 per second and a pulse train consisting of 2500 groups per second of eight pulses each.

C. The Digital Servo

Once the distance commands have been transformed into discrete pulses, they must be further converted into machine motion which is essentially analog in nature. Several approaches are possible by which the required transformation can be made. In one method the discrete pulse trains are first converted to analog commands. These then are compared with feedback signals produced by analog devices as in conventional servosystems. Such a scheme is described by Moore.[6] The approach employed in this system, as well as numerous other applications of this type, utilizes digital feedback instrumentation. For each unit distance traveled by the machine, a pulse is generated by the feedback instrument. A running difference is kept between the number of command pulses and the number of feedback pulses received by the servosystem. This difference is then used as the error signal to drive the servomotor.

1) Feedback Instrument: The feedback instrument used in the system is a rotary electromagnetic device called a quantizer. It has one rotor winding and two stator windings spaced 45 electrical degrees apart. The windings of the rotor and stator, having the equivalent of 250 poles, are formed by etching conductors and copper-clad plastic disks. The rotor is excited from a high-frequency source of about 1.6 mc. The signals induced in the stator windings depend on their position with respect to the rotor winding. They are thus amplitude modulated by movement of the rotor. The two stator outputs are fed to demodulators and wave-shaping circuits from which they emerge as rectangular, or two-level, waves.

2) Synchronizer: The synchronizer, which receives the rectangular signals from the demodulators and wave-shaping circuits, converts them into pulses representing unit motions. The zero crossings of one signal are used to define unit rotation of the quantitizer shaft. The relationship between the direction of zero crossing and the level of the other signal indicates the direction of rotation.

Command pulses produced by the digital interpolator are also fed to the synchronizer and combined in the opposite sense with quantizer pulses. Simultaneous occurrence of command and feedback pulses is detected, and if necessary one is delayed by a clock period. The synchronizer outputs then consist of two trains of pulses, one train representing error in the positive direction and one in the negative direction.

3) Error Register: The error register is an 11-stage reversible binary counter, composed of flip-flops, which counts continuously the pulses supplied by the synchronizer. The count in the register therefore represents at all times the accumulated difference between the number of command pulses and the number of feedback pulses. The first five stages of the register are decoded by means of a binary-weighted resistive summing network to produce a proportional voltage. The last stage of the register indicates the sign of the error and is similarly included in the network with appropriate weight. Large counts are not decoded but clamped at a fixed level.

Provision is made to detect overflow of the counter and thereby cause an emergency shut down of the system. Excess-error detection at adjustable levels of 0.010, 0.025, and 0.050 inch is also provided.

4) Servodrive: The analog error voltage from the resistor decoding network serves as input to the servoamplifier. In all present applications of the system, the machine slides are powered by high-performance electrohydraulic drives. The drives consist of piston-type rotary hydraulic motors controlled by servo valves from a constant-pressure hydraulic supply. The valves in turn are stroked by elec-

[6] G. T. Moore, "The Numericord machine-tool director," this issue, p. 6.

trically operated "torque motors." As a combination, the specially developed valve and torque motor have a bandwidth in excess of 250 cps.

D. Supervisory Control and Other Features

1) Control Logic: Control tapes are read one block at a time into buffer storage by means of a photoelectric tape reader operating at the rate of 150 lines per second. Four magnetic-core shift registers serve as temporary storage for the three distance commands and the feed-rate number, and another four registers serve as active storage. Under normal circumstances, overflow of the command scaler causes clearing of all counters and storage registers in the interpolator and the transfer of input information from temporary storage into active storage. Reading of the next block of control tape into temporary storage then commences.

As information is read in from the tape, it is automatically checked for parity. Detection of an error causes an appropriate indication, and switches the machine to manual control at the conclusion of the cutting specified by the previous block. Other control circuits are provided to detect excess errors in the servosystems, failure of the clock source, and malfunctioning of the tape reader, the power supplies, and other components. These errors are indicated on the control panel and automatically cause suspension or complete shutdown of the machine system until the errors are corrected. On-off operations at the machine can be initiated by the special auxiliary-function tape codes at any time.

2) Manual and Tape-Control Synchronization: The interpolator is designed so that at any time it is possible to switch from tape to manual control and vice versa without loss of synchronization. This is accomplished by terminating the flow of input pulses to the command scaler under manual control, thus preventing the further generation of noncarry pulses in the command scaler. Consequently, no output pulses will be delivered to the servos. The command scaler, as well as all storage registers for the distance numbers and the feed-rate number, retain their contents through recirculation until automatic operation is resumed. The feed-rate scaler associated with the feed-rate register is kept running at all times. Command pulses are obtained for manual control by detecting its overflow at different stages. The clock frequency is adjustable from 20 to 100 per cent of nominal without disturbing the operation of the unit. Since a change in clock frequency changes the over-all speed of operation proportionally, a wide range of manual feed-rate override is thereby provided.

3) Position Indicator Counter: Command pulses produced by the interpolator can be accumulated in a separate reversible decimal counter. The contents of this counter, having a capacity of 999.9998 inches, is displayed directly in inches, using decimal indicator tubes. The counter can be switched to any desired axis for monitoring purposes or for manual positioning.

4) Marginal-Checking Circuitry: The control unit contains built-in circuitry by which tape-controlled operation of the unit can be simulated for test purposes. Simulated operation is obtained by furnishing dummy inputs by way of the tape-reader amplifiers to the temporary storage registers and causing a transfer into active storage. The interpolator begins production of command pulses for the servosystems on each axis. The simultaneous overflow of all error registers with the command scaler in the proper state will reinitiate the test cycle as long as the test switch is on. Otherwise the process halts, and an error signal appears. This built-in circuitry tests 80 to 90 per cent of the logic in the control unit. A more exhaustive test can be made with test tapes which in addition check the tape reader and the remaining portion of the servosystems. Supply-voltage levels, the clock-wave shape, and the shift current for the magnetic-core circuitry can be varied to provide for marginal checking under the above test conditions.

E. Components and Construction

The machine control unit uses vacuum-tube flip-flops, triggers and cathode followers, together with germanium diodes, in a dc gating system. Storage registers and scalers are made up of magnetic-core shift registers with associated vacuum-tube read-write circuitry. The unit contains approximately 400 tubes, 2800 diodes, and 200 magnetic cores. Plug-in type construction and etched circuitry are used throughout. A view of the control unit is shown in Fig. 20 with the cabinet doors removed.

IV. Machine

Fig. 21 illustrates an application of the control equipment just discussed to a contour milling machine of the moving-column type.[7] Position of the cutter is controlled along three mutually perpendicular axes with respect to a stationary work piece mounted on the vertical angle plate. Motion perpendicular to the plane of the work (along the cutter axis) is provided by the spindle head assembly. The spindle in turn is carried by a slide which moves vertically on the column. Longitudinal motion is provided by the column itself, carrying the operator's platform with it.

All three axes of the machine are equipped with high-performance hydraulic servodrives of the type described in Section III-C, 4. The longitudinal axis, having the longest stroke of 172 inches, is driven through a rack-and-pinion arrangement. Backlash is effectively eliminated by means of a dual-drive system in which two motors, connected hydraulically to the same valve, work against each other through separate gearing to the rack. The vertical and transverse axes, having strokes of 52 and 18 inches respectively, are driven by single motors acting through precision ball-nut lead screws. A preloaded double-nut arrangement

[7] Machine designed and built by the Kearney and Trecker Corp., Milwaukee, Wis.

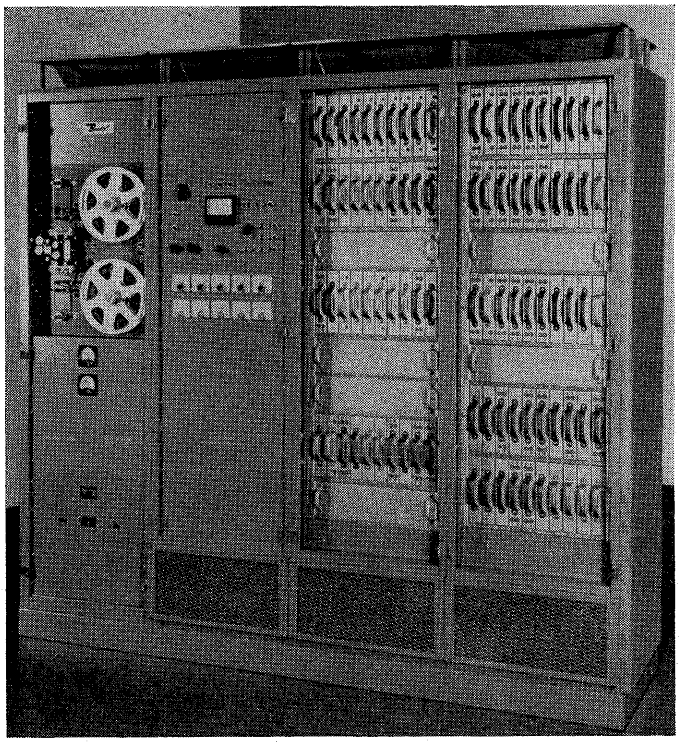


Fig. 20—Photograph of machine control unit.

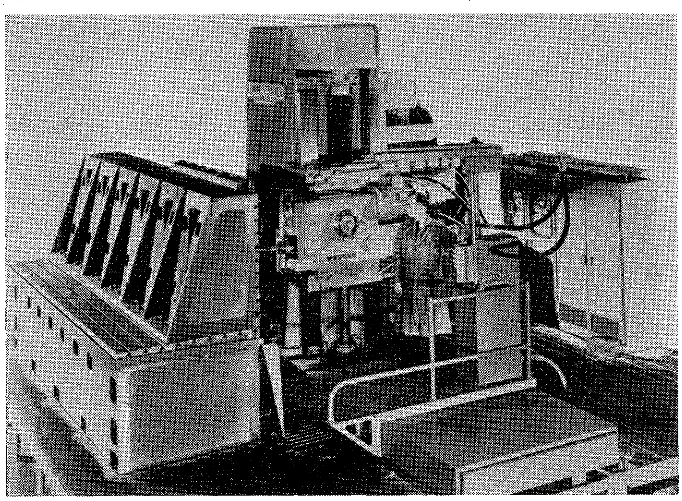


Fig. 21—Application of control equipment to a large three-axis contour milling machine.

is designed to eliminate backlash between the machine slide and the lead screw.

Feedback information is provided to the machine control unit by means of quantizers, according to the system outlined in Section III-C. In the case of the longitudinal axis, the quantizer is coupled to the slide by means of a high-precision rack and pinion. For the two shorter axes, the instruments are geared directly to the drive screws. Gear ratios between quantizers and machine motion are such as to produce 5000 pulses per inch of travel.

Dynamic response is unusually good for a machine of this size as a result of the hydraulic drives and the care taken during design to achieve a high-stiffness, low-mass mechanical structure. For small signals, the bandwidth of the longitudinal system is in the range of 7 to 10 cps in spite of a moving structure which weighs over 35,000 pounds. A sudden stop from 30 inches per minute (without programmed slowdown) results in an overshoot on the order of 0.010 inch. The bandwidth of the other two axes is considerably higher, with correspondingly faster response to severe transients.

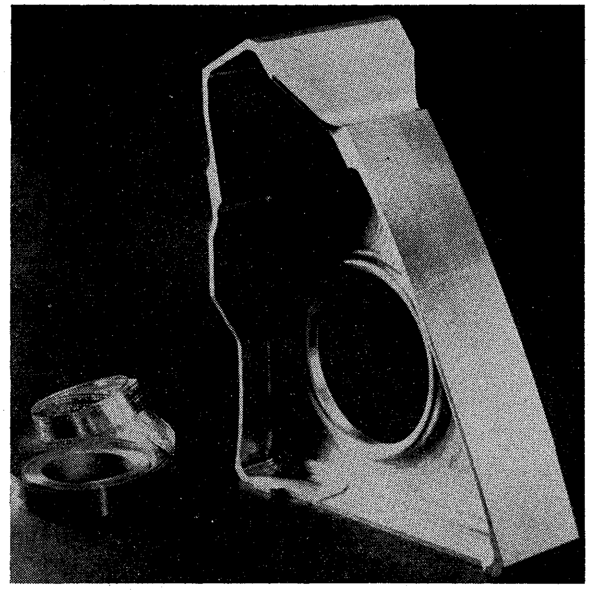


Fig. 22—Aircraft fitting machined under numerical control.

Maximum speed of the machine slides is nominally 180 inches per minute, although feed rates are usually limited to 100 inches per minute during actual cutting. Initial tests indicate that the machine is capable of repeatability within 0.001 inch over the entire range of strokes.

V. Illustration of System Performance

The part illustrated in Fig. 22 is an aircraft fitting produced from a forged blank by a machine functionally similar to that just described. The process consisted of both roughing and finishing cuts on the outside of the flange, on the inside of the flange on both sides of the web, and in an area between the flange and the center circle on each side of the web. Single cuts were taken in order to machine mounting pads on one edge of the flange, and to finish the inside of the large hole near the center.

Preparation of the process sheets for this part by an experienced process engineer would require approximately 10 hours, starting with a suitable working drawing. When advantage is taken of the symmetrical-cutting feature of the machine, six sheets of the COMPAC-1 format are necessary for full definition of the machining process. These result in a process tape 46 feet long, which requires 2.4 hours to punch and verify with the Flexowriter. Computer time is approximately 1.2 hours. This latter time includes the reading of the process tape, all necessary computing, the punching of the control tape, and the print-out of coordinates and feed rates at frequent intervals along the cutter-center path. The print-out operation accounts for about 25 per cent of the total computing time. The control tape for the part is 70 feet long.

TABLE I
PRODUCTION PROCESS TIMES FOR TRACER-CONTROLLED AND NUMERICALLY CONTROLLED MACHINES

	Tracer Control	Numerical Control
Fixture design and manufacture	126 hours	71 hours
Template design and manufacture	84 hours	—
Process-sheet preparation	—	10 hours
Process-tape punching and verifying	—	2.4 hours
Computing time	—	1.2 hours
Total tooling time	210 hours	85 hours
Set-up time for first part	3.75 hours	1 hour
Machining time	1.75 hours	1 hour
Total time machine occupied	5.5 hours	2 hours

The times involved in the production process are summarized in Table I. Corresponding data for the conventional method utilizing a tracer-controlled machine are also given. This comparison shows a substantial reduction in both tooling and machining times for the numerical approach. It also emphasizes the fact that the operations directly concerned with numerical processing now require a relatively small fraction of the time consumed by the more conventional steps still left in the production process, such as the design and manufacture of the holding fixture (which incidentally was not produced by numerical control).

Incorporation of linear interpolation with the machine, permitting final control-tape preparation directly by a small general-purpose computer, represents a compromise having many desirable operating features. Sufficient data is not yet available to permit a full scale economic evaluation of the system. However, the above example illustrates the reasonableness of processing time, computing time, and tape lengths characteristic of this basic approach. Productivity of the controlled machine in terms of speed, accuracy, and surface finish has been well demonstrated in actual field use.

VI. ACKNOWLEDGMENT

Development and production of the equipment described in this paper would not have been possible without the joint efforts of a large number of people. Over-all responsibility for the system, as well as development of computer programs, the machine control unit, and servodrive components has been carried by the Research Laboratories Division of Bendix Aviation Corporation. Bendix Computer Division has contributed materially in the development of magnetic-core circuitry, and in production of the control units. Design and production of the machine mentioned in this paper was entirely the responsibility of the Kearney and Trecker Corporation. The authors gratefully acknowledge the contributions of numerous personnel in all of these organizations.

Discussion

Question: Doesn't pulse multiplication effectively change the quantizing level of the system?

Mr. Ho: The answer to that is a qualified yes. If you are multiplying by 2, 4, or 8, then the distance you can specify is rounded off to the nearest 2, 4, or 8 pulses. In other words, if you are cutting at 240 inches per minute, you can specify distance only as close as 1.6 mils. We say this isn't too big a handicap. If you are cutting at such a high speed, the servo error is going to be many times greater than that. Also, in the automatic tape preparation program, we keep track of these errors and correct from one block to the next so that at any time we are never off more than the 2, 4, or 8 pulses. When we finally come to stop after cutting, the automatic program applies a correction and returns the tool to the accurate position again. As far as the tool engineer is concerned, he doesn't have to worry about this. In fact, he doesn't know it exists.

Question: What are the checking and printing features of the automatic program?

Mr. Ho: There are several loading checks on the automatic program. For example, if a tool engineer describes a cutter-center path which would result in a situation impossible for the machine to cut, an error stop would be indicated by the automatic program.

The print feature allows the position of the cutter to be checked as the tape is being prepared. Now, this print feature can be suppressed by including another instruction on the process sheet. On the circular arcs or other curves' points, we can print out intermediate points at any frequency specified, for example, every 15 points. But the end point of every section, straight line or curve, is always printed out if print-out is requested. Print-out slows down the computer somewhat. Every print-out takes approximately ten seconds.

The computer output tape is a sandwich construction utilizing aluminum foil between two mylar layers. The process tape made by the Flexowriter can be verified by the verifying attachment to the Flexowriter, essentially by typing the same information for the second time and comparing with the first tape.

Question: Is there any compensation made for variation of cutter size due to wear?

Mr. Ho: No. Maximum travel of the machine is 170 inches, but the control unit can accommodate straight-line cuts up to approximately 409 inches on each axis.

Question: What is the drum storage capacity in the G-15 computer?

Mr. Ho: This machine has a storage capacity of approximately 2000 words.

Question: What are the components in the computer system which prevent the computer from running in real time?

Mr. Ho: The most difficult problem in numerical control is cutter-center offset. Especially in three dimensions, the cutter-center offset problem generally is very complicated and leads to all sorts of exceptions. A completely general tape-preparation system which could keep up with the machine and supply information as demanded would require a very large and very fast computer. In general, we find a physical separation for buffering purposes is almost essential between the cutter-center offset problem and the machine control problems. We feel that if you have a computer that is fast enough to supply information to the machine in real time under peak demand conditions, then most of the time the computer will be sitting there doing nothing. Only when really complicated situations come up that require full use of the computer, would it be utilized efficiently.

Logical Organization of the DIGIMATIC Computer

JACK ROSENBERG†

System Specifications

PRECEDING the design of the original DIGIMATIC computer and control system, Electronic Control Systems Inc., (ECS) performed technical and economic surveys to determine the requirements for successful entry into the commercial automatic machine-tool market. The results may be summarized as follows.

1) A three-axis milling machine, with any two of the three slides simultaneously controlled, was the most logical tool to be adapted.
2) Cost and complexity of the control apparatus directly associated with the machine tool must be minimized, and reliability maximized. This dictated the elimination of the computing or interpolation function from the tool, with only the control function remaining.
3) From the standpoint of flexibility, accuracy, reliability, cost, and bandwidth, the only form of precomputed memory consistent with the above was magnetic tape recorded in digital incremental form.
4) The programming process by which part-drawing data is converted to magnetic-tape commands in a special-purpose computer must be rapid, accurate, economical, and easily accomplished by regular machine-shop personnel.
5) To satisfy the bulk of shop needs (about 95 per cent of commercial parts), the computer must generate (interpolate) linear or true circular paths with an accuracy of ±0.001 inch, without accumulation of error, from data already present on drawings or from those readily obtainable from drawings.

Mathematical Requirements—Straight Lines

As in most engineering projects, the most difficult task was defining the problem; once this was accomplished, the technical solutions were evolved fairly readily. Since this paper deals mainly with the computer portion of the system, it is possible to derive the input data, its code, and the input device from 4) and 5) above. Part drawings normally define a piece by the decimal coordinates of the start and end of each segment of the contour, and for circular arcs, usually include the radius and decimal coordinates of the center. Thus, the computer should accept contour breakpoints in decimal form; if code conversion is necessary prior to interpolation, it should be accomplished automatically.

The input mechanism, for reliable use by shop people, should be extremely simple and provide means for verification. A decimal keyboard device (such as that on an adding machine) with print-out tape is familiar to nearly all such personnel.

Let us begin by examining the case of a straight-line segment in the X-Y plane. The general equation of such a line

$$y = mx + b, \tag{1}$$

must be solved by the computer, and given the start and end points of the desired segment (which lies on this line).

Since it is fair to assume that a workpiece will contain continuous contours, the constant b may be eliminated from (1) due to the fact that the cutting tool has been brought to the proper start point for this path as a result of the completion of the previous segment.

Now m represents the slope, or tangent of the angle between the desired line and the X axis. By definition, the tangent is equal to the change in Y from the start point (P_1) to the end point (P_2) divided by the change in X over the same interval. It is exactly expressed as follows:

$$m = \frac{y_2 - y_1}{x_2 - x_1}. \tag{2}$$

Eq. (1) has therefore been reduced to the form

$$y = \frac{y_2 - y_1}{x_2 - x_1} x + b, \tag{3}$$

and for the reason indicated above, b need not be computed. This expression is illustrated graphically in Fig. 1. The line to be traversed has now been expressed in terms of the data normally furnished on a part drawing.

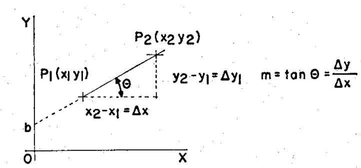


Fig. 1.

Engineering Solution—Straight Line

Quantities y_1, y_2, x_1, and x_2 are constants over the entire line, and are known before the interpolation begins. Since differences $y_2 - y_1$ and $x_2 - x_1$ need be calculated only once for the entire segment between P_1 and P_2, these subtractions may be performed on a relatively slow mechanical adding machine without any undue time penalty, inasmuch as subtraction time is less than keyboard entry time.

† Electronic Control Systems, Inc., Los Angeles, Calif.

Some keyboard device to receive data is unavoidable, so it may as well be an adding machine.

Therefore we will insert quantities y_2 and y_1 (in that order) into an adding machine, then order it to find the difference, and automatically store y_2 and y_1 in an electrical register (relays). Next, x_2 and x_1 will be entered, and $x_2 - x_1$ will be stored in another relay register. Let us call these differences Δy and Δx, respectively.

An electronic counter can be used to divide input pulses by an integral quantity which may range from one up to the counter capacitance; in such service it is usually termed a predetermined counter. Most commercial predetermined counters using a binary or modified binary code, are preset mechanically and require a considerable interval after recognition to be reset to an initial condition and to be ready to accept further input pulses.

However, a high-speed all-decimal predetermined counter can be designed around the magnetron beam-switch tube. Each target may be connected to a coincidence gate. When the total count in the multistage counter has proceeded from 0 to the integer by which the input pulses are to be divided, recognition will occur, an output pulse will be generated, and the entire counter rapidly reset to 0 via the spades. One microsecond is sufficient for reset; thus, the input pulse rate may be as high as 1 megacycle and still permit reliable counting and resetting. We term the process dividing, and such a counter, a divide counter. Fig. 2 shows the logic of a divide counter.

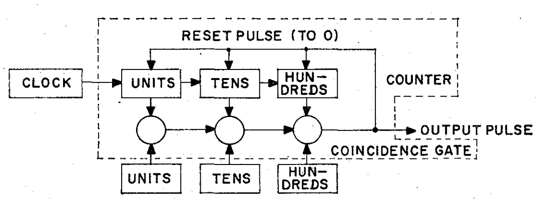


Fig. 2—Divide counter, register containing quantity by which the clock pulses will be divided.

Suppose we feed pulses from a common clock into two divide counters, one dividing the clock rate by Δx, the other by Δy, as pictured in Fig. 3. If the clock frequency f is constant, a train of uniformly spaced pulses will emerge from each divide counter; the output frequency of the first counter will be $\frac{f}{\Delta x}$, that from the second counter will be $\frac{f}{\Delta y}$, and the ratio between these rates will be

$$\frac{\frac{f}{\Delta x}}{\frac{f}{\Delta y}} = \frac{\Delta y}{\Delta x}, \tag{4}$$

which is exactly the slope of the line connecting P_1 and P_2. Assuming each output pulse represents a motion of 0.001 inch by a machine-tool slide, the interpolation process must

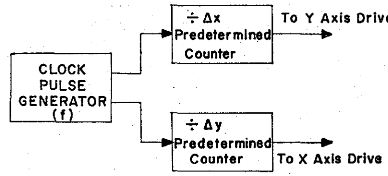


Fig. 3—Straight-line generator.

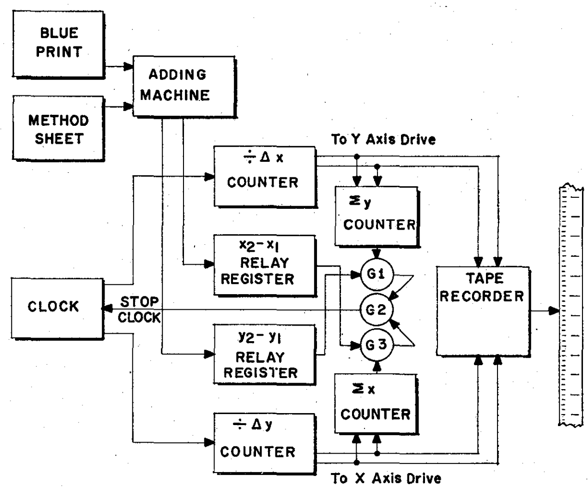


Fig. 4—Straight-line generator.

continue until the Y-axis drive receives a pulse total of Δy, the X-axis drive a total of Δx. Two additional counters and three recognition gates as shown in Fig. 4 monitor this process and turn off the clock when recognition signifies P_2 has been reached. Simultaneously the pulses from the divide counters are recorded on appropriate tracks of an eight-track magnetic-tape recorder for later use in controlling the machine-tool table.

Since each of the two output pulse trains will have uniform pulse spacing, they will be periodic, and therefore optimum for driving servomechanisms at constant velocities. The description of the DIGIMATIC computer as a linear interpolator requires but one further embellishment to be complete. For vector machine table feed rates to be controlled automatically, the two output pulse trains are sampled by analog pulse rate discriminators, combined in quadrature, and the resultant voltage compared to that commanded by a feed-rate potentiometer calibrated in inches per minute. The difference voltage operates a clock frequency control system in a closed loop, so that the desired feed rate results.

Mathematical Requirements—Circles

The engineering solution given above describes a means for generating a line of constant slope by producing two pulse trains whose frequencies are constant, and are related so that

$$\frac{f_y}{f_x} = m. \tag{5}$$

Now let us examine the mathematical characteristics of a circle. The general equation for a circle with its center at point (a,b) in the XY plane is

$$(x - a)^2 + (y - b)^2 = r^2. \tag{6}$$

We can derive the exact expression for the instantaneous slope at any point by first taking differentials

$$2(x - a)dx + 2(y - b)dy = 0. \tag{7}$$

Separating variables we reduce it to

$$(y - b)dy = -(x - a)dx, \tag{8}$$

and finally the slope is expressed by

$$\frac{dy}{dx} = -\frac{x - a}{y - b}. \tag{9}$$

Engineering Solution—Circles

In the solution for first-degree equations, predetermined dividing counters were utilized to divide the common clock-pulse source by two constant quantities. Referring to the expression for slope of the circle given in (9), it can be seen that if quantities $x - a$ and $y - b$ can be made available at all times, the predetermined counters mentioned earlier can be used to divide the clock-pulse rate f by these varying quantities, and a circle can be generated by continuously changing the slope to fit the above equation.

Inasmuch as the slope changes in sign four times during the generation of a complete circle, it becomes necessary to keep track of the magnitude and sign of the quantities $x - a$ and $y - b$. This can be accomplished by the use of reversible counters which can add and subtract pulses, instead of simple monodirectional counters which would have sufficed for performing the pulse-summing operation in Fig. 4. We call such counters "sum" counters, and we have obtained economy of components and sufficient counting speed by using decimal glow-transfer counter tubes of the Erikson type. Since it is necessary to present the quantities $x - a$ and $y - b$ in parallel decimal form to the inputs of the divide counters, reversible gas-tube GS10C, which has all of its ten cathode electrodes brought out to the tube socket, was chosen.

In the case of the circle, the input commands to the divide counters have to be switched from the output terminals of relay registers, as described for a straight line, to the output terminals of sum counters. In addition, direction gates must be added between the output of the sum counters and the tape-recording channels, and must be commanded by the sign of the appropriate sum counter, to make the logic completely consistent with the mathematical requirements of a circle. The logic of a circle generator, which receives as input information the coordinates of the start point, end point, center, and instructions as to whether the circle is to be generated in a clockwise or counter-clockwise direction, is given in Fig. 5.

To convert the straight-line generator of Fig. 4 to a circle generator of Fig. 5, it is necessary to provide a 100-pole double-throw relay, which switches the 50 input terminals of each 5-decade divide counter from a relay register to a sum counter. Some changes in the method of handling input data must also be incorporated, to permit the sum counters to be initially preset to the actual quantities $x_1 - a$, $y_1 - b$ at the start of the circular arc.

Mention should be made of the singularities which occur four times during the generation of a complete circle. The instantaneous slope of the circle twice goes to zero and twice becomes infinite. The former cases correspond to the quantity $x - a$ becoming equal to zero, and the latter cases occur when $y - b$ becomes zero. If the clock frequency is divided by zero at these times, an infinite output rate should be produced by the predetermined counters. However, as soon as one additional pulse at this infinite rate is emitted and added into the previous total of zero in the reversible counter, the appropriate total changes from zero to either plus one or minus one, and the predetermined counter is then asked to divide by a finite integer. The problem can be resolved by setting up a system of logic which causes the predetermined counter to emit one pulse without receiving a pulse from the clock, if it is ready to be preset by the total in the reversible counter and finds this total to be zero. By thus generating a pulse with no input from the clock, we can simulate an infinite ratio of output to input at the point of singularity. Fig. 5 assumes the use of this type of predetermined counter.

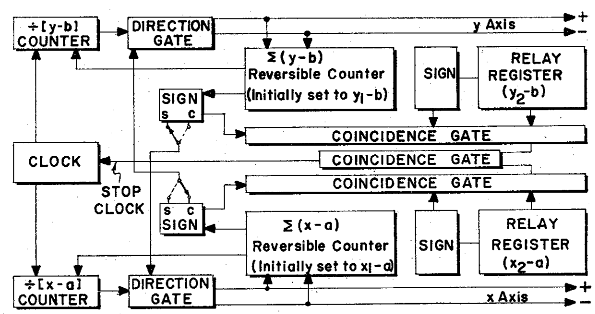


Fig. 5—Circle generator. *Note 1*—In sign block of reversible counters, the letter s refers to the true sign, the letter c refers to the complement of s. *Note 2*—With switches on the sign outputs of the above counters set as shown by the solid lines, the circle will be generated in a clockwise direction. If they are set as shown by the dashed lines, the circle will be generated in a counter-clockwise direction.

To illustrate the operation of this all-decimal interpolator, Fig. 6 shows the path described on graph paper by the occurrence of pulse outputs in the case where the center of the circle is at (0,0) and the radius is 10 increments. Although the appearance of the outline is not smooth, in practice a smooth contour is machined because of the smoothing action of the servomechanisms which drive the slides of the machine-tool table. Furthermore, the example was chosen for simplicity, since a circle with such a small radius is somewhat academic. Standard milling tools are available only in diameters of 1/16 inch (0.0625 inch) and higher.

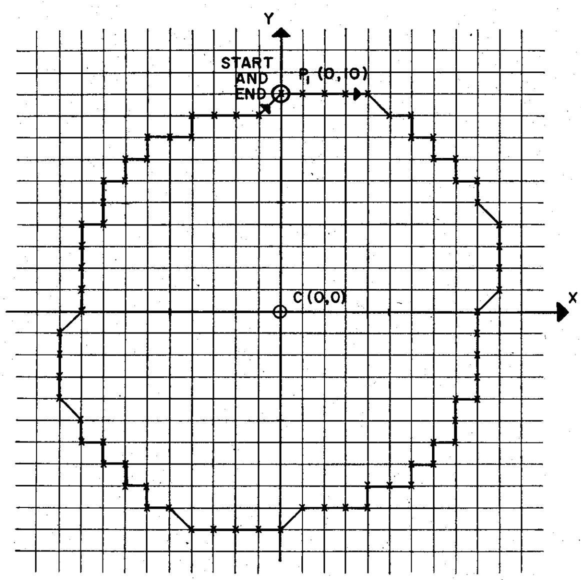


Fig. 6—Interpolated circle. (Radius 10 increments.)

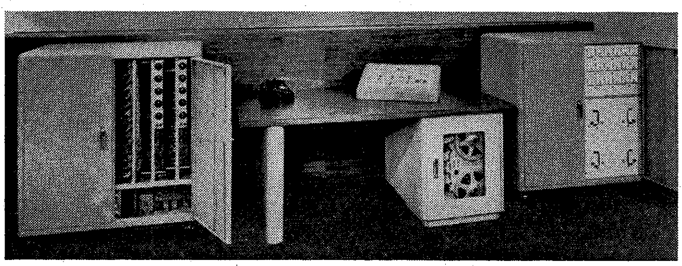


Fig. 7—Model 120 DIGIMATIC computer.

Thus additional smoothing will be provided by the large size of the periphery of the cutting tool compared to the value of an increment, which in our case is 0.001 inch.

The DIGIMATIC Model 120 computer, which follows the logical principles described above, is pictured in Fig. 7. The relay register, clock, divide counters, and sum counters are housed in the cabinet at the left. The desk contains the magnetic-tape handler and some input distribution circuits, while the control console (including adding machine) may be seen on the desk top. The cabinet on the right contains only power supplies.

A close-up of the control console is shown in Fig. 8. Fig. 9 is a photograph of a triangle, circle, and parabola which was machined from tape prepared by the 120 computer.

Generation of Other Curves

The technique of using two divide counters to generate a curve of continually changing slope can also be applied to other second-degree curves. As an example, the equation for a parabola (principal axis parallel to the X axis) is

$$(y - k)^2 = 2p(x - h). \tag{10}$$

In this case k, p, and h are constants. The instantaneous slope is

$$\frac{dy}{dx} = \frac{p}{y - k}. \tag{11}$$

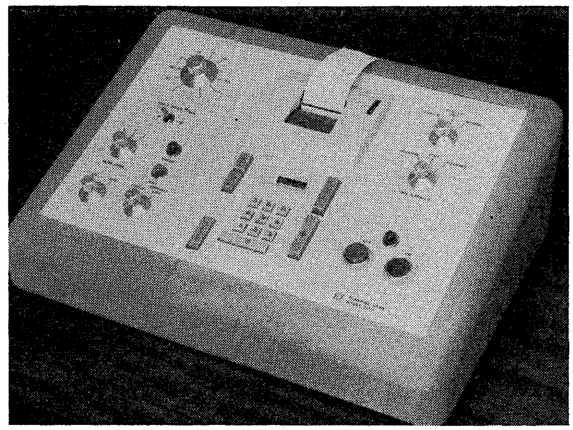


Fig. 8—Model 120 DIGIMATIC computer.

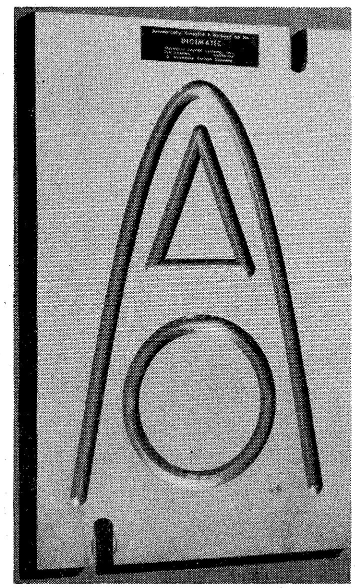


Fig. 9—Geometric contours produced by model 120 DIGIMATIC computer.

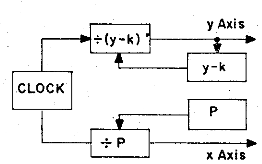


Fig. 10—Parabola generator.

It may be seen that it differs from the slope of a circle only in the respect that the numerator is a constant instead of a variable. Fig. 10 indicates the logic of an interpolator for this kind of parabola. In practice, we have generated parabolas by entering the information as if a circle was desired, and preventing x-axis command pulses from reaching the X sum counter (by removing a driver tube).

The case of an ellipse can be analyzed as follows. The general equation is

$$\frac{(x - h)^2}{a^2} + \frac{(y - k)}{b^2} = 1. \tag{12}$$

From this the slope is derived

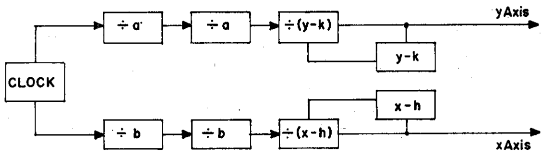


Fig. 11—Ellipse generator.

$$\frac{dy}{dx} = -\frac{(x - h)b^2}{(y - k)a^2}. \tag{13}$$

Fig. 11 shows how this could be implemented. Since two additional constants occur, *a* and *b*, additional divide counters would be necessary, as well as memory registers to retain the quantities for presentation to the extra divide counters.

This computing philosophy can be extended to accommodate higher order equations. As an example, consider this fourth-degree curve

$$y = 5(x - 10)^4. \tag{14}$$

It has the instantaneous slope

$$\frac{dy}{dx} = 20(x - 10)^3, \tag{15}$$

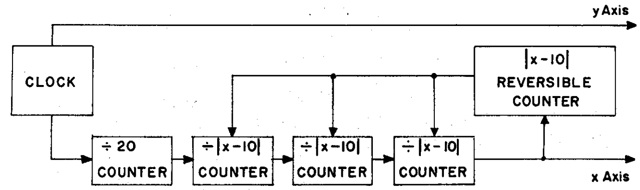


Fig. 12—Higher order curve generator.

which can be generated by the computer logic given in Fig. 12.

Conclusion

A special-purpose, high-speed, digital computer has been described which operates completely in the decimal system, and contains built-in programs for straight lines and circles, selection being made by the turn of a switch. The input data required is inserted very simply, and is either already present or may be simply derived from information on part drawings. This computer could be described as a function generator, which generates analytically expressed relationships in a digital manner. Similar computing principles are being applied to control problems outside the machine-tool field.

Discussion

Question: What is the reliability of the Electronic Control Systems Director and mean time to failure?

Answer: Well, the most unreliable component of our Director is the human operator and unfortunately he breaks down pretty often. The entry of numerical data from a planning sheet is a fairly tedious process and we find that monotony is the chief producer of errors. It is a little hard to answer this accurately, since we have been operating a laboratory system, not an industrial one.

I will say that the mean free time is beyond one day. We have some peculiarities in the Los Angeles climate which control this. We use standard telephone-type relays and whenever there is a strong dust storm from the desert, it plays hob with the relays and we have to clean them out. In general, the mean free time is more influenced by electromechanical than electronic components.

Question: What is the least count, least programmable increment of the ECS system? What is the maximum feed rate of any axis?

Answer: In our present system, one pulse equals one thousandth of an inch, so this is the least programmable increment. Our computer controls the vector feed rate, so that the maximum feed rate in one axis is not necessarily the actual cutting feed rate. To put it another way, our maximum clock-pulse frequency is 500 kc and if you will go through the mathematics and logic of our system, it works out that for most typical cuts on our prototype Bridgeport Mill we must limit the feed to 15 inches per minute. The manufacturer of the original machine provided power feeds up to this rate.

We designed our DIGIMATIC computer to produce feed rates up to 15 inches per minute. There are some cuts that we could make much faster. However, the machine-tool servos could not follow them. To put it another way, the feed rate must depend on the type of material and depth of cut. Our new Director, now near completion, will interpolate at eight times real time for feed rates up to 50 or 100 inches per minute.

Question: Do you verify the tape before you use it for direction of the machine tool, and if "yes," by what means?

Answer: At the present time, we make a pulse count. Our magnetic tape contains six control channels, a plus and minus channel for each of the feeds, *X*, *Y*, and *Z*, and for each segment of the cut we perform a pulse count for each axis. This is a fairly tedious process, so what we usually do, unless the part must be made to unusually close tolerances, is simply machine a part. It would be possible to play the tape back into the sum counters of our computer. We did not provide for it in the present system. We will in the forthcoming ones.

Question: Do you have a stored program for the interpolation process, or is it made by hardware?

Answer: We have only two programs for this computer, both of them built in. The selection of linear or circular interpolation is controlled by a number of relays which change the input lines to our divide counters. In the case of straight lines, the relays connect the divide counter inputs to numerical relay storage. In the case of circles, these inputs are connected to the sum counter outputs.

This is about the only meaning that I can place on the question.

The Master Terrain Model System

JOSEPH A. STIEBER†

Synopsis

THE master terrain model system is an automatic data reduction system which will extract three-dimensional contour information from maps of various projections, aerial stereo photographs, or existing master models, and store these data in a universal format. From these stored commands, the system will automatically drive a fabrication unit and produce three-dimensional terrain models.

The master terrain model system was conceived to fill the serious need of the armed forces for a faster and less costly method of constructing terrain models. The proposed system will replace the present manual process of construction with a completely automatized system. The system will further provide a valuable means for the permanent storage of master model data and thus supplant present unwieldy storage methods and eliminate the loss of valuable models due to natural deterioration and handling.

The system, which is now in the stage of prototype construction, will consist of a map scanning unit, a recording and playback unit, and a contour cutting mechanism, all units being programmed and controlled by digital computers. The digital computer control system will interject positioning corrections into the scan data for the various map projections, so that the scanned three-dimensional information can be recorded and stored in a universal spherical format. The computer system will additionally provide a feedback error correction medium to the scanning drive and also supply an interpolation of pulse analog signals to produce faired curved surfaces.

The Naval Training Device Center has for many years been concerned with the development of better methods of producing master terrain models, reproduction models, and the surfacing of these models with mapping or photographic intelligence.

A long range program, covering many projects, has produced new techniques and equipments which have contributed greatly to the present success of many operational and training devices used by our military forces.

The development of an automatic system for the production of terrain models has been evolved through many years of research by engineers of the Center. Many methods were tried to translate contour elevation data into digestible information for recording and machine consumption. Among these have been systems using color coding, magnetic, electrochemical, photographic, and line counting techniques. Of those investigated, the photographic processes coupled with copper etching techniques appeared to offer the best possibility of meeting the desired performance characteristics from a standpoint of simplicity, speed of scanning, and accuracy.

When the determination of prime feasibility was accomplished, a contract was let to Technitrol Engineering Company to undertake a design study of one-year duration for further development of an over-all completely automatic system. As a result of the success of this design study, a second contract was let to Technitrol Engineering Company for the finalized design and construction of a prototype model of the system. This prototype is presently nearing completion and preliminary tests indicate that all specification requirements will be met successfully.

System Approach

The magnitude and complication of the initial system designed required a design study of one-year duration to adequately prove preliminary engineering concepts. The speed, accuracy, and flexibility requirements of the system presented many complicating factors which had to be solved one at a time.

A basic problem was the development of a method of translating two-dimensional positions plus a third-dimensional code into computer words.

Let us first study the physical aspects of the problem and the terms to be used. We will be dealing with maps of various projections, scales, and miscellaneous dimensional terminology all of which must be reduced to a common denominator which can be assimilated by a data reduction system. A three-dimensional model is in effect a scale map which has been deformed into the third dimension to simulate the exact contours of the earth which it represents. The problem here is to convert two-dimensional maps into three-dimensional models. A typical master model, as shown in Fig. 1, presently is produced by manual methods. Thus, a model of this type requires many months of arduous labor to produce. The objective of the master terrain model system is to produce these same models in a matter of hours with greater accuracy and with consequent savings in cost.

Map Coding Methods

The concept of an automatic terrain model system starts with the preparation of data from a flat map plate for machine acceptance. If we consider the familiar multicolor map, it has been printed from as many as a dozen separate color plates to consolidate the colors and mapping intelli-

† U. S. Naval Training Device Center, Port Washington, N.Y.

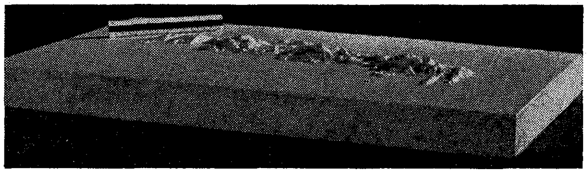


Fig. 1—Typical master model.

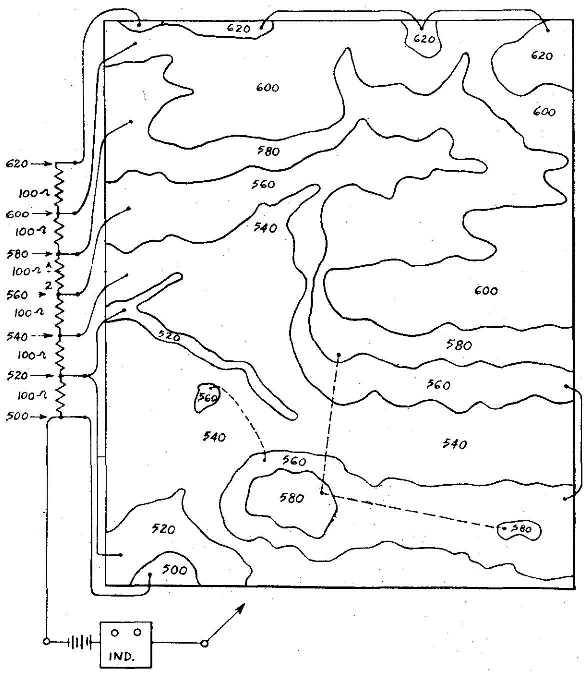


Fig. 2—Metal-map coding.

gence on a single sheet. One of these color plates (called a brown plate) has been singled out for our use because this plate contains the contour lines which represent earth contours and which will provide the three-dimensional information we require for the system. If this contour plate is used to photographically etch a copper-clad laminate sheet as is done in printed circuit techniques, a resulting map in metal will be produced which will provide terrain levels insulated from each other by the line thickness which has been etched away. If this metal map is now coded as shown in Fig. 2 by using electrical connections to supply appropriate voltage levels in ratio to map elevations, the resulting prepared metal map will provide the third dimension when scanned electrically.

Automatic Scanning

The scanning mechanism consists of a single stylus which is driven across the "metal map" by means of a servodrive. The stylus makes electrical contact with the map plate and picks up the analog signal code which represents terrain levels. The speed of the scan is proportional to the variable pulse drive in the X direction and a shift of 0.01 inch takes place in the Y direction at the end of each X scan. Thus, continuous map profile data are obtained in incremental steps of 0.01 inch and these data are converted into the binary system and passed on to the recording stage of the system.

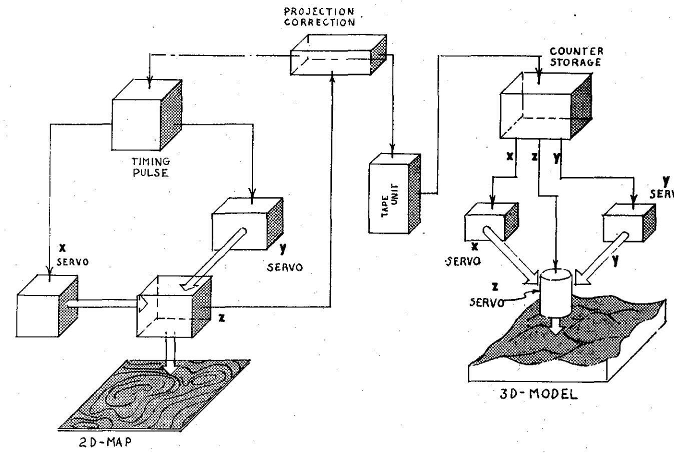


Fig. 3—Block diagram of system.

The Storage System

The 1-inch magnetic tape storage is used in the system primarily to provide a permanent storage medium for model information. This method of storage is destined to replace present methods of storing heavy molded models. A single reel of tape, representing a portion of the earth's surface at a set scale, may be used to reproduce three-dimensional models at various horizontal scales and various vertical scales or vertical exaggeration. Also, the same tape may be used to reproduce spherical model sections or flat models to various map projections. Thus, this single tape may be used to reproduce any one of possibly 25 master terrain models.

The Over-All Master Model System

Fig. 3 shows the diagram of the system. The prepared metal map is placed in a scanning mechanism and the timing generator causes the X and Y servos to position a scanner head to appropriate scan the map plate. The X and Y servo-positioning information along with the Z-code information is passed into a computer to convert the map projection coordinates into spherical coordinates which match the curvature of the earth's surface. The data are then passed on to the memory device which is a magnetic recording system. The magnetic tape is then used to drive the model cutting mechanism through a counter storage medium feeding the appropriate three-dimensional servodrives. The cutting tool is a high-speed routing tool which is positioned in three dimensions by the servosystem and thus produces a three-dimensional model.

Design Factors of the System

(See Fig. 4.) The three-dimensional models will be constructed from Hydro-cal plaster blocks approximately $30 \times 30 \times 3\frac{1}{2}$ inches thick. Hydro-cal plaster has been

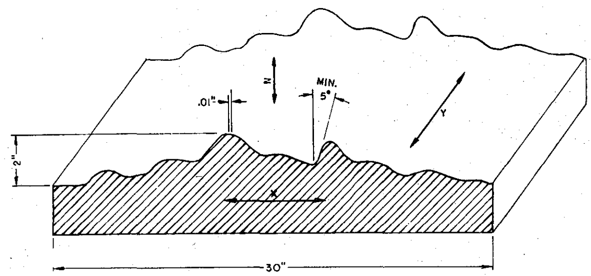


Fig. 4—Typical model section.

used in the system in combination with inert agents and lubricating agents to produce the model blocks. Because of the dimensional stability of plaster, its good machining properties at very high speeds, and the ease of reducing residue to small dimensions, it provides a good material for the system.

Profile cuts will be taken at each 0.01 inch. A cutting speed of 30,000 to 50,000 rpm is required to maintain deflection within tolerance limits and reduce the size of residue chips to micro dimensions for ease of residue evacuation.

Maximum cutting depth is 2 inches, and accuracy is maintained to 0.01 inch in any dimension.

Digital coding is used to obtain more accurate computation for projection corrections, interpolation of profile coordinates, and storage recording. All the standard map projections will be incorporated into the system, both in the map scanning portion, and in the model input section. Thus, flat models to various map projection coordinates will be produced as well as curved models simulating scale curvature of the earth.

The Control System

(See Fig. 5.) The control system uses serial type SEAC circuits. A 1-inch wide magnetic tape is used for information storage utilizing binary pulse coding of 3 channels, thus allowing 4 separate runs on the tape to use 12 channels. Ten bits are used on the tape for each position or word. The scanning/recording portion of the control system is programmed by means of a timing pulse generator. This generator impresses a timing pulse channel on the magnetic tape and pulses the X and Y counter units. The counter units in turn operate the X and Y servos for scanning the map plate and feed the binary positioning code for the X and Y channels on the tape. A single channel on the tape is used for both X and Y positioning code, since a single profile X scan produces no Y change until the end of the scan run. Then the Y increment shift takes over on the same channel using 4 pulses of time on the channel to effect the 0.01-inch physical shift of the axis and a new X profile scan then proceeds on the same channel. While this XY scanning operation is proceeding, the Z-code information is being recorded on a third channel of the tape.

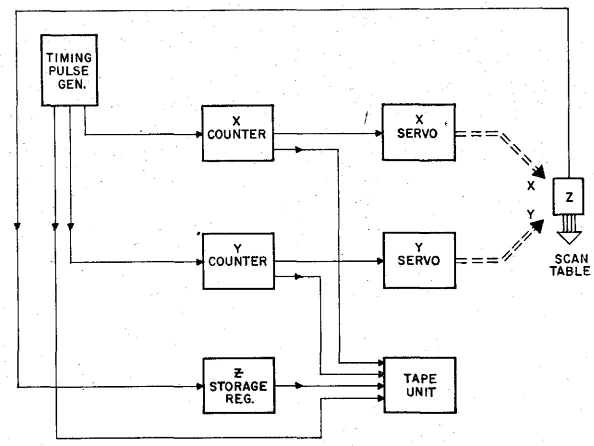


Fig. 5—Recording system block diagram.

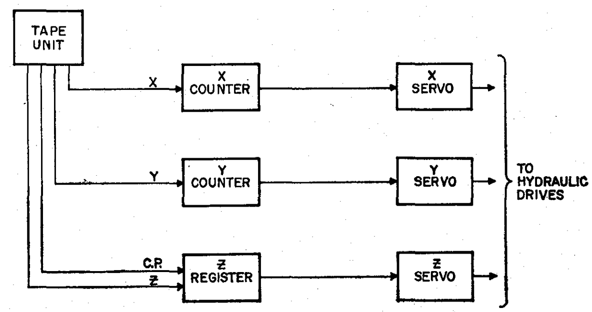


Fig. 6—Reproduction system block diagram.

(See Fig. 6.) The tape output drive system is actuated by the three-channel pulses of the tape unit. The XY channel activates the X and Y counters alternately which energizes the X and Y servodrives. The timing pulse channel controls the register and the Z-dimension channel activates the Z servosystem concurrently with the XY systems.

The Machine Section

The mechanical portion of the equipment consists of a heavy base table which carries the three-dimensional drives, the milling head housing, and the residue exhaust system. The hydraulic power supply unit provides power to the hydraulic motors at approximately 3000 psi.

The three-dimensional servocontrol system controls hydraulic motors which turn ball-screw actuators to position a milling head carriage in the X and Y positions. The third Z motion is accomplished by hydraulically activating a piston which positions the milling head drive in vertical motion. The milling head, which rotates a tapered fluted tool at high speed, produces a profile cut of approximately 0.01-inch width and varying in depth as the drives produce the profile motion.

The combination of precise tolerance "ball-screw actuator drives" and the fast acting hydraulic motor drives provides accuracies of speed and acceleration in excess of the original requirements.

Conclusions

The research program outlined in the foregoing paragraphs has been geared to meet the over-all requirements of the master model program. This research program has been divided into three phases as follows.

Phase I

This phase was devoted to a detailed analysis of the problems in the form of a design study. This included studies relating to mapping and cartographic techniques, data programming, data storage, machining or forming methods, and systems study to correlate these units into a workable design.

Phase II

Preparation of design drawings, specifications, and the construction of a prototype model of the design to prove feasibility of the over-all system were covered here.

Phase III

The final phase included the addition of complete computer control to the prototype model and reworking of components as necessary to carry the project through the various research phases, which were proceeding in close conformity to the original program schedule.

This project is believed to be the first step in a series of completely automatic devices for the correlation of cartographic and photographic intelligence to maps and models.

It is expected that this equipment when used for model production will consist of scanning units located in a centralized location, with model production units at various activities concerned with model usage.

Discussion

In answering the questions on the model terrain system, Mr. Stieber had with him R. E. Hock, of Technitrol Engineering Co., Philadelphia, Pa., which is reducing this technique to practice.

J. S. Seely (Southern Railway System): Can overhanging cliff configurations be handled?

Mr. Hock: We can handle anything up to approximately 85 degrees; anything greater than that is so vague in our mapping that we don't worry too much about it.

E. L. Harden (Westinghouse Electric Corp.): Is there interpolation between contour lines or does the cutter make steps corresponding to the contour lines?

Mr. Hock: In one method of coding, which is to etch away the areas between the contour lines, we coat the map with a conducting paint, a resistive paint in which you get contour smoothing between the lines. We actually convert this smooth analog voltage to digital steps between lines so that if you are cutting an exaggerated scale, you do get digital changes between your contour lines. In the method in which you are etching away the lines and leaving the areas, there is no easy way of interpolating between contour lines.

Mr. Ebeling (Otis Elevator Co.): What is the maximum map scale ratio?

Mr. Hock: On the XY axis, they are in the ratios of one half, 1, 2, and 4, in the Z axis, one half, 1, 2, 4, and 8. It was easiest to obtain binary values at some later date. These will probably be converted to decimal.

Mr. Maetra (RCA Labs., Princeton, N.J.): What is the limit of the smallest change delta Z that can be recorded? Is this comparable with the precision that an operator can obtain manually?

Mr. Hock: The smallest increment of movement in any axis is five thousandths of an inch and it is better than manual accuracy.

Question: How do you take into account the cutter center offset correction? It would appear that no correction is made, in which case the three-axis part will be in error.

Mr. Hock: The cutter is a tapered tool, tapering down to fifteen thousandths at its tip and has a fifteen-to-one aspect ratio. So, the offset is approximately seven and one-half thousandths. Now, we are trying to obtain accuracies of plus or minus one one-thousandth of an inch so we have almost really taken up our accuracy in the cutter offset.

However, we do ignore it, as you have stated.

Question: When will the system be producing three-dimensional maps?

Mr. Hock: We actually have the Z axis at the plant. We are working on the servosystem at the present time and the X and Y axes are under construction at a subcontractor. However, we have a prototype of the XY axis which is a converted milling table that gives us limited travel of approximately three by six inches and this we hope to be operating in January, 1958.

Mr. Winslow (ABMA): In the preparation of the original map for photographic work, do you use a color or line to indicate the contour line?

Mr. Hock: The original map is a black line map and it is a transparent negative with the contour lines in black and from this we produce the etched—either the area map or the lined map.

Question: What are the common contour intervals used?

Mr. Hock: These vary from ten feet up to several hundred feet, depending upon the horizontal scale of the map. The elevations are stored as earth-centered spherical coordinates.

Question: Do you use the output computer to convert to flat projection?

Mr. Hock: Actually this conversion of the coordinate system is the next step in the program. Presently, we are working on using the existing projection in reading and storing in that projection and cutting in that projection.

A Coordinated Data-Processing System and Analog Computer to Determine Refinery-Process Operating Guides

C. H. TAYLOR, JR.†

General Description of the System

THE coordinated data logging and computer equipment, which has been constructed for Esso Standard Oil's Belot Refinery, Havana, Cuba, measures and records the true value of 101 process variables and 11 operating guides. The measured variables are gas flows, liquid flows, level, pressure, oxygen percentage, and temperature. The inputs to the logger representing flows, level, and pressure are in the nature of 3 to 15 psig signals. The oxygen percentages and temperature signals exist as dc mv signals. Unique in the system is the incorporation of an analog computer which calculates 11 operating guides. These are computed at the end of a readout cycle from the logger. The operating guides computed are the following:

1) Carbon burning rate,
2) Catalyst-circulation rate,
3) Catalyst-to-oil ratio,
4) Ratio of feed-to-reactor catalyst hold-up,
5) 430° FVT conversion-corrected,
6) Percentage weight of hydrogen in coke,
7) Per cent weight of carbon make on total feed,
8) Heat duty of top pump-around system,
9) Heat duty of mid pump-around system,
10) Regenerator superficial velocity,
11) Reactor superficial velocity.

The logger can be adjusted to give a complete readout of all process variables every 10 minutes, 30 minutes, or hourly. At the end of the hourly logging cycle, information required by the computer is fed into the computation circuits which proceed to calculate the 11 operating guides noted above. As the computation for each guide is completed, the true value is logged on the output typewriter. Thus, a given set of guides is based on information given in the logging cycle immediately preceding the computation period. Scheduled logging is supplemented by "on demand" logging and can be initiated at any time by the operation of a push button.

Two electrically actuated automatic typewriters with separate tape punch have been furnished to record the outputs of the automatic logger and analog computer. The tape punch produces a 5-channel punched-tape output suitable for the actuation of IBM equipment. In order to provide for proper utilization of this tape, the following information accompanies the data:

1) A symbol to identify "on demand" readouts.
2) Tabulating card-advance and card-eject signals for each group of 15 data points.
3) Tabulating card number for each 15 data points.
4) Time to the nearest minute for each tabulated card.
5) Unit-identification number for each tabulated card.
6) An identification character on each card in order to differentiate preset hourly readings which are of interest to accounting and other groups for manual or more frequent readouts demanded by the operator but not essential to tabulating card computations.
7) Two additional identification characters on each card in order to identify ultimate data users.

The equipment is housed in two cabinets in the following manner. One cabinet contains the transducing equipment for the process variables, and the programming equipment for the data logger; the second cabinet houses the computing circuits. Both cabinets are arranged to permit a continuous air purge. In addition, the computer cabinet was provided with an air-conditioning system in order to dissipate the electrical heat generated by the computing elements. All equipment has been designd to conform with specifications for a Class I, Group D, Division 2 area as defined in article 500 of the 1953 National Electrical Code.

It should be noted that among the process variables being logged are three gas flows which have been pressure and temperature compensated. All flows are printed as hourly averages and 24-hour totals. Five of the temperatures are printed as hourly arithmetic averages, and 61 are printed as instantaneous values. All necessary linearization of thermocouple inputs, the conversion of customers 3-15 psig pneumatic signals to digital signals, and the extraction of square-root functions have been provided in order to check the over-all accuracy of the automatic logger. These points are printed out on the log sheet before each readout cycle. A dead weight loaded precision pneumatic comparator has been provided to check the pneumatic-to-digital transducer which is used for the pneumatic-signal inputs.

The accuracy for the logged-process variables is as follows:

1) Temperature ± 0.25 per cent.
2) Pressure ± 0.5 per cent.
3) Compensated flows ± 1.0 per cent.
4) Integrated and averaged flows ± 1.5 per cent from 10 to 25 per cent of full-scale flow, ± 0.75 per cent

† Fischer and Porter Co., Hatboro, Pa.

from 25 to 50 per cent of full-scale flow, and ± 0.5 per cent from 50 to 100 per cent of full-scale flow.

The accuracy of computation for the operating guides is ± 2 per cent.

Use of AC Computer Signals

Because the computer must be capable of operating at a 100 per cent duty cycle, that is to say, 24 hours per day for 365 days per year, the signals handled by the computing circuits are in the nature of 60-cycle ac voltages. This was done in order to produce a high degree of reliability in the equipment. Most conventional computers today utilize signals which are dc voltages. In order to obtain the best possible degree of accuracy and minimum drift, it is necessary that all dc signal voltages be checked for proper calibration before the problem is solved. This becomes a frequent maintenance procedure in the case of a continuously-operated piece of equipment. In order to eliminate the need for a periodic check of all signal-voltage accuracies and amplifier balance, the signal voltages in the computer are obtained from a group of transformers. These have been designed to operate on a primary voltage of 220 volts, 60 cycles. In practice, the transformers are operated on 115 volts, 60 cycles. We have therefore supplied twice as much iron in the laminations as actually required. This insures that the transformers operate on the linear portion of their magnetic-characteristic curves. This is done so that we are certain that the coupling flux within the core does not approach the saturation level and that the secondary voltages contain a minimum of distortion and bear a constant proportionality to the primary voltage. The transformers are wound with a turns-ratio accuracy of ± 1 per cent. The signal voltages are then padded to the desired accuracy by means of adjustable rheostats connected in the secondary circuits. In order to further insure repeatability between the units handling a multiplicity of signals, all transformers are loaded equally. Furthermore, they were constructed with laminations stamped from a uniform batch of iron in order to further insure repeatability among units. The result is a signal-source module of five volts, 60 cycles. When signals of greater magnitude are required, additional transformers are used with their primaries connected in parallel, and their secondaries connected in series.

For example, if a signal of, say, 22.00 volts were required, we would supply five transformers with their primaries in parallel, and their secondaries in series. This, it can be seen, results in a "voltage stick" which is 25 volts long. We take the signal from the low end of the voltage stick to a point 22 volts from the end.

This reasoning was applied since a computer operating with signals which were obtained as described above will require no standardization or reference to a secondary voltage standard, nor will it require any special regulation of line voltages. It is felt that this results in a more reliable system of equipment, and restricts computer down-time to a program of preventive routine maintenance.

Types of Computer Inputs

The information fed into the computer is derived from several sources as follows:

1) Logged during a readout cycle and stored as a shaft position prior to the start of the computation period,
2) Derived during the computation period and stored as a subroutine for use later in the computing cycle,
3) Fed into the computer manually.

Fig. 1 shows a representative segment of an information time-flow diagram. As can be seen, the computer solves the 11 equations in sequence, beginning with (1) and proceeding to (2), (3), etc. In order to solve a particular relationship, it is necessary to obtain the values for all variables involved in the equation. As the diagram indicates, these may consist of parameters which have been logged during the previous readout cycle, variables which have been manually set, or those which have been stored as a subroutine during a previous computation. When all of the necessary data have been assembled, the information is fed into the computer, and when the output device has come to balance, the desired solution is printed out on the logger typewriter. The computer programming circuits then proceed to assemble the data necessary to solve the next equation.

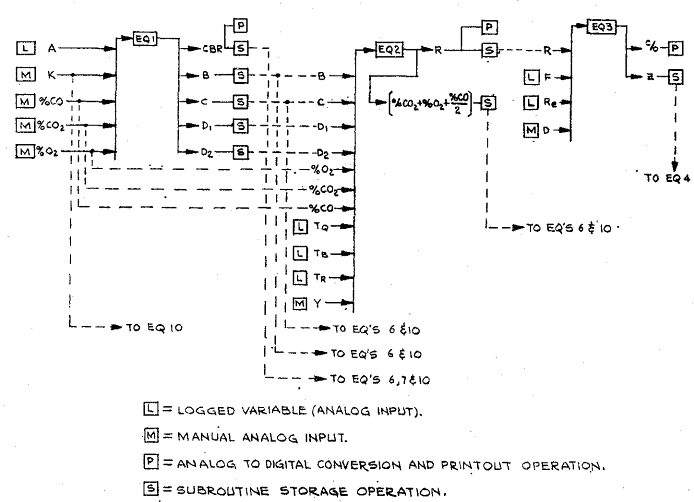


Fig. 1—Information time flow diagram.

When this information is available, it is fed into the computer circuits which proceed to solve the next relationship, etc. When the last operating guide, or (11), has been solved and typed out, the computer circuits are disengaged from the logger for a period of one hour at which time the next computation period is initiated.

A Time-Shared General-Purpose Computer

In order to perform the necessary computation with a minimum of equipment, it was decided to assemble a group of computing elements consisting of algebraic summation elements, coefficient modules, and electronic multiplication or division elements, and to program their input and output circuits to solve for the value of each operating guide serially. This means that a given computer building block may be used repeatedly with a minimum duplication of equipment. This time sharing of computer "elements" makes it possible to assemble a so-called general purpose computer and, in effect, makes it a specific purpose machine through programming. To determine the actual number of computing blocks required, a study of the relationships to be calculated was made, and a block diagram for each relationship was established. The equation requiring the greatest number of elements then determined the diversification and number of modules to be included in the computer. Since all other relationships are of a simpler nature, we use the building blocks over and over again to solve these less complicated equations.

Fig. 2 shows a typical computer block diagram. The relationship to be solved, in this case reactor superficial velocity, is shown at the top of figure. The input-signal information is shown at the left-hand side of the figure. The block diagram indicates how the various signals are modified and combined in order to solve the desired equation.

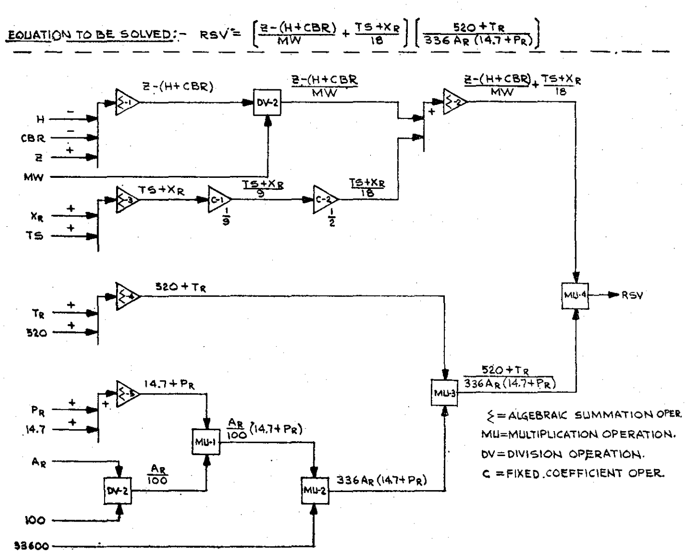


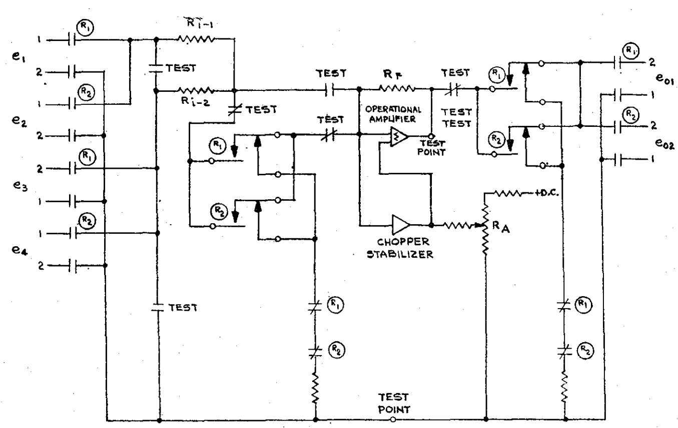
Fig. 2—Typical computer block diagram.

The references shown in the diagram refer to the analog signals as ac voltages with respect to ground, that exist at the various points in the circuit. The equation is generated, so to speak, by starting with single terms which are combined by addition, subtraction, multiplication, and division until the desired results are achieved at the output (shown at the right-hand side of the figure). It was necessary that we know the variation in magnitude for each of the individual input signals so that we could assign appropriate signal voltages to these inputs during the process of scaling the computer parameters.

Types of Computer Building Blocks

In order to time share the various computer elements, it was necessary to switch a given module between the several circuits in which it was to be used. To illustrate how this was achieved, diagrams are included showing how the computer relay programming is used to accomplish this switching.

Fig. 3 shows the circuits for a typical time-shared summation amplifier. In this case, we desire first of all to add voltage E1 to E3 in one equation and then, at some later time, to add voltage E2 to E4. It will be noted that the input signals are connected to the amplifier by means of computer programming relays with contacts R1 and R2 as shown. The signals are switched by Form D (or make before break) contacts which insure that the input to the amplifier is not open-circuited during the switching operation. The stage gain of this particular type of amplifier is made equal to unity by suitable adjustment of the feedback resistor R_F and the input resistances R_{I-1} and R_{I-2}. Furthermore, it can be seen that, to minimize any dc drift within the amplifier itself, a chopper-stabilizer module has been included. Should it become desirable to check the dc output level as referred to the input, a test switch has been provided as shown. This switch decouples the amplifier from any existing input signals and returns the summing junction to ground through a suitable resistance. A dc volt meter can then be placed from the output of the amplifier to ground, and potentiometer RA manually adjusted to reduce the dc unbalance to zero.



$e_{o1} = (e_1 + e_3)$

$e_{o2} = (e_2 + e_4)$

R_1-R_2 = COMPUTER PROGRAMMER RELAY CONTACTS

Fig. 3—Summation amplifier.

Fig. 4 illustrates how these same principles are applied to an operational amplifier being used as a coefficient module. That is to say that the input voltage is multiplied by a fixed constant depending upon the position of the sliding contact on a multiturn potentiometer associated with the desired constant. Here again, it will be noted that the switching operation is accomplished by Form D contacts (or make before break) so that neither the amplifier-feedback loop nor the input-summing junction is left open-circuited during the switching operation.

The chopper-stabilizing amplifier is again included, as well as the test switch, should it be desirable to check the dc level of the output with respect to the input.

Further reference to the diagram will show that the polarity of the input signal determines the alegbraic sign of the term it represents. Thus, if we desire to add two signals together, we arrange to feed voltages of like polarity into the summing amplifier. If we desire to take the difference between two voltages, the signals are fed 180 degrees out of phase. This simply means that the secondary leads from the signal transformer are reversed, resulting in the necessary phase reversal. It should also be noted that the operational amplifiers used invert the signals fed to their inputs; that is to say, if a signal with polarity 1-2 is fed into an amplifier, the output voltage will have phase 2-1. It was therefore necessary to study the relationship being solved in order to determine the phase relationships between the various input signals.

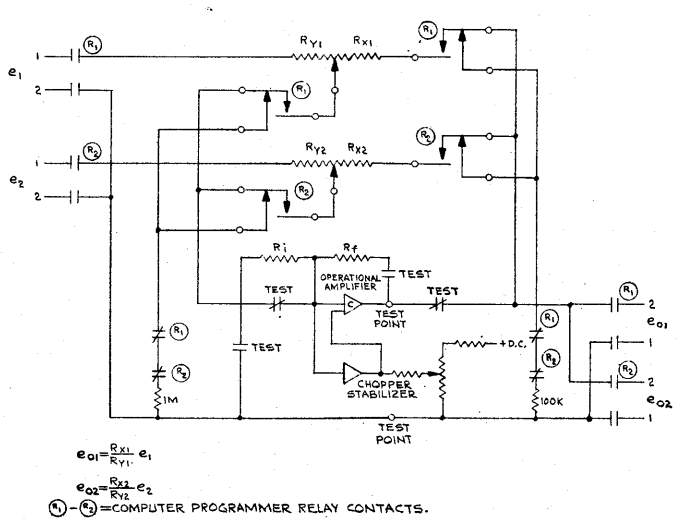


Fig. 4—Coefficient amplifier.

Fig. 5 illustrates the method of handling an electronic multiplier-divider element using ac analog signals. Since the device is a true multiplier, it was necessary to avoid the sine-squared relationship which results when two ac signals are fed to the inputs of the module. It will be seen that one signal is converted to a dc voltage by means of a full-wave rectifier and forms the input applied to terminal No. 2. This voltage is then modulated by the signal applied to input No. 1. The output signal is proportional to the rms product of the input signals in the case of multiplication, and to the ratio of the rms amplitudes of the inputs in the case of division. Contacts on the computer programmer relays determine whether the block will be used for multiplication or division. Auxiliary operational amplifiers C_D and C_M were provided in order to give the proper output voltage relationships. All amplifiers within the multiplier-divider were chopper stabilized in order to minimize any dc drift within the element itself. The circuit shown was arranged to provide a multiplication and division element with unity gain. In order to increase reliability and minimize drift, it should be noted that all computer building blocks were provided with chopper stabilization and also

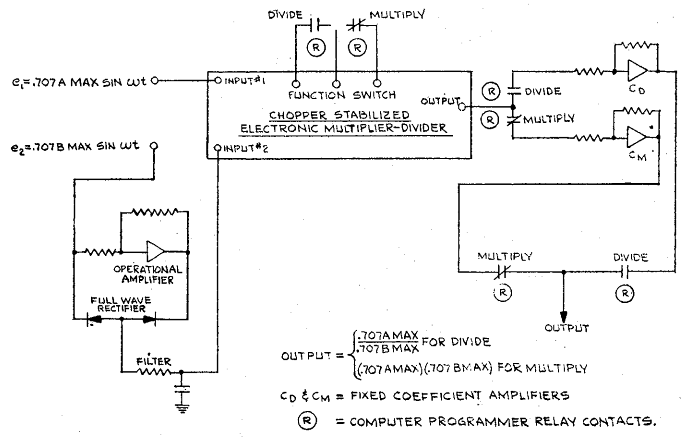


Fig. 5—Multiplication and division circuit.

means for manually adjusting any dc unbalance which may occur in operation.

Memory Systems

Information stored in the computer is derived from 3 sources:

1) Information stored as a shaft position within the logger itself.
2) Data which exist within the logger instantaneously during a readout cycle, but which must be maintained for use as a computer signal.
3) Information stored as a subroutine during the computation period.

Process variables which are read as integrated quantities such as compensated flows and hourly average temperatures exist as shaft positions in the logger. Fig. 6 shows the circuit for a typical compensated gas flow. In this arrangement the position of the contact on the accumulator slide wire is a function of the flow which has been pressure and temperature compensated. During the logging cycle, the flow accumulator slide wire is connected to the logger readout device. During the computation cycle, however, an ac voltage from a computer signal transformer is impressed across the terminals of the slide wire, and the voltage developed from the contact to the low end of the slide wire is returned to the computer as an ac signal for use in the computation circuits. It can thus be seen that the accumulator slide wire is time shared between the logger circuits and the computer circuits. Moreover, there are computer parameters which are derived from signals that exist in the logger only instantaneously as the logger programmer scans the customer input information. For example, instantaneous temperatures, as given by thermocouples, must be stored at the time that they are read during the logging cycle and must be maintained in memory in a form which can be used by the computer. This was accomplished by means of a small electromechanical servosystem. The input signal from the thermocouple is fed to the logger readout device. A retransmitting or follower potentiometer mounted on the shaft of the logger readout device transmits a dc voltage to a storage servosystem located in the computer cabinet. The shaft of this servo then

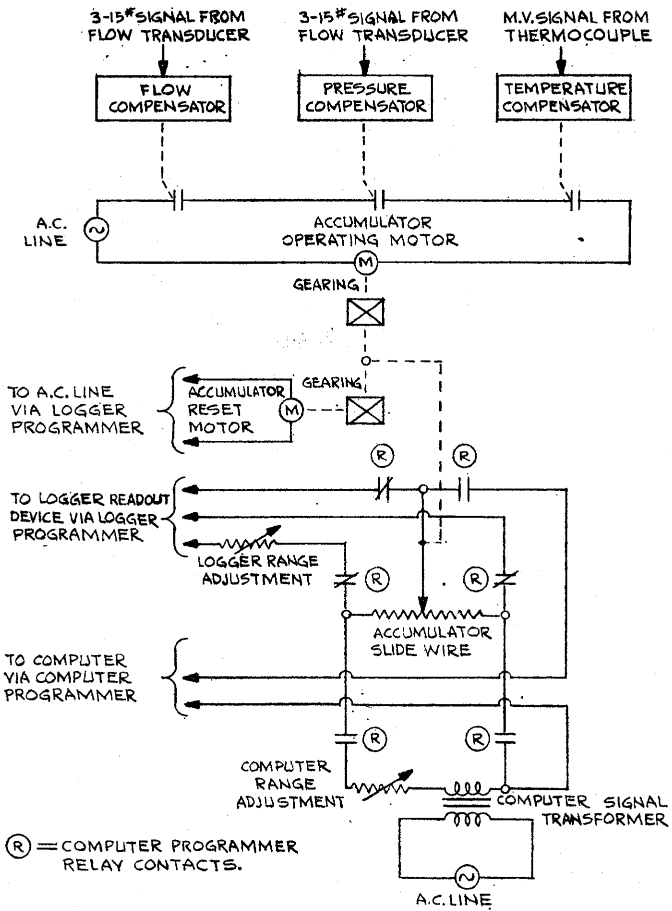


Fig. 6—Typical compensated-flow circuit.

assumes a position proportional to the value of the input temperature being measured. A follower potentiometer mounted on the shaft of the storage servo is then used as a computer input information by impressing across its terminals an ac voltage from a computer signal transformer.

Subroutine storage within the computer itself is achieved in a similar manner. The ac voltage representing the quantity to be stored is fed to a storage servo whose shaft again assumes a position proportional to the stored quantity. Retransmitting or follower potentiometers mounted on the shaft of the storage servo are then used in subsequent computations.

Signal Sources

The signal-source transformers were treated in a manner similar to that of the computer building blocks themselves. A single transformer is time shared between the circuits comprising several of the relationships to be handled by the computer. Fig. 7 illustrates this point. The diagram shows a single signal-source transformer which has been time shared between four equations. The signal to be used for (1) is generated across potentiometer P_{1B} which is shown as a manually-set variable. The voltage to be fed for (2) is generated across potentiometer P_{2B} in series with rheostat P_{2C}. Rheostat P_{2C} has been included in this case to provide zero suppression for the instance where the signal does not go to zero at the minimum range of its variation. For (3), potentiometers P_{3B} and P_{3C} are used in an arrangement similar to that shown for (2). In this case, potentiometer P_{3B} is shown as a retransmitting or follower potentiometer mounted on servostorage mechanism S_{V1}. The signal to be used in (4) is generated across P_{4B} which is shown as a potentiometer mounted on one of the compensated flow accumulators.

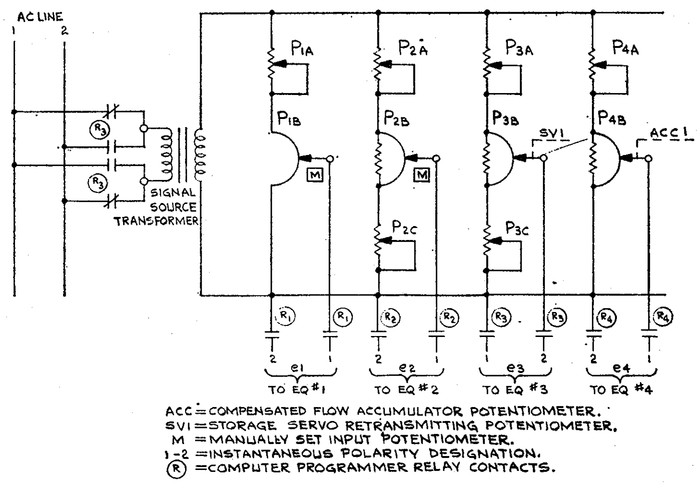


Fig. 7—Representative signal-source diagram.

Potentiometers P_{1A}, P_{2A}, P_{3A}, and P_{4A} have been included as range adjustments set so that the voltage developed across the signal potentiometers is of the correct magnitude, depending upon the scaling desired for a particular variable. Computer programming relays operate contacts R1, R2, R3, and R4 which are associated with equations (1) through (4), respectively. The relay contacts in the primary circuit of the signal-source transformer are used to reverse the phase of the transformer secondary voltage. As mentioned previously, an instantaneous polarity of 1-2 is given the designation of a positive signal in the computer, whereas an instantaneous polarity of 2-1 designates a negative signal in the computer. Reference to the diagram will show that the signals used in (1), (2), and (4) are shown to have a negative sign, while the signal generated for (3) is shown as a positive signal.

In this manner, a single signal-source transformer has been time shared with a corresponding reduction of three in the number of transformers to be supplied. This results in a saving of physical space within the computer cabinet as well as in a reduction in the cost of the necessary components.

Tie-In Between Computer and Logger

Fig. 8 indicates how the various circuits previously described were arranged to complete the coupling circuits between the automatic data logger and the computer. It should be noted that the addition of the computer in no way affects the operation of the data logger—it was designed to act as a supplementary or auxiliary device.

As previously stated, the logger programmer scans the

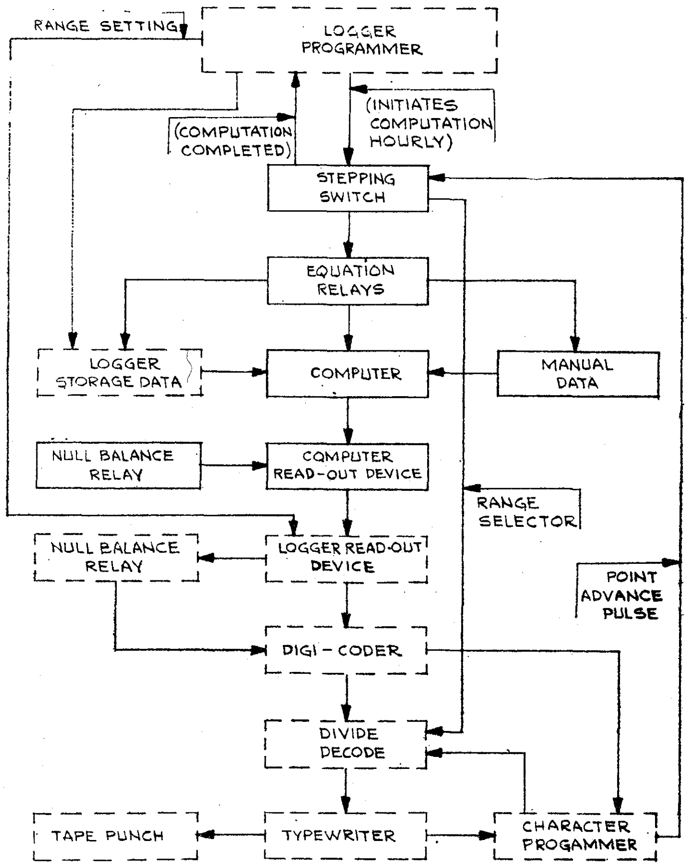


Fig. 8—Coordination between logger and computer.

process analog-input information at least once an hour. At the completion of the hourly logging cycle, the logger programmer sends a signal to a stepping switch located in the computer programmer. This signal advances the stepping switch to position no. 1 and initiates the computation cycle. Position no. 1 on the stepping switch operates a group of equation relays associated with the first relationship to be solved. Contacts on these relays then proceed to select for the computer input information which has been derived from logger storage data as previously described, and manual input data. The equation relays also serve to connect the computer building blocks in the arrangement necessary to solve this first relationship. The output from the computation circuits is fed directly to the computer readout device. This unit is an electromechanical self-balancing ac potentiometer. A retransmitting or follower potentiometer sends a dc signal back to the logger readout device causing the two servos to track. Since the logger readout device is a multirange self-balancing potentiometer, it is necessary that range information be supplied to it from the logger programmer during the computation cycle. A digicoder, coupled through gearing, is used as an analog-to-digital converter which sets up a digital output proportional to a shaft-position input from the logger readout device. The digicoder output is then fed through a divide-and-decode network to the logger typewriter. The characters present in the output are scanned by the character programmer which actually operates the solenoids on the typewriter mechanism. Tape punch information is generated as the typewriter is recording the value of the computer solution.

During the time that the solution to (1) is being typed out, a feedback-point advance pulse is sent back to the stepping switch in the computer programmer which then advances to position no. 2. The equation relays associated with operating guide no. 2 are than energized and the cycle repeats as described above for (1). The programming continues until a solution is obtained for each of the 11 relationships to be computed. At the completion of (11), a feedback pulse is generated by the computer stepping switch and returned to the logger programmer. The stepping switches associated with the logger point programmer then return to the home position where they remain until the initiation of the next logger cycle.

System Characteristics of a Computer Controller for Use in the Process Industries

W. E. FRADY† AND M. PHISTER‡

INTRODUCTION

BEFORE the detailed design of a computer may be begun, it is necessary to set up some fairly detailed specifications which define the operating characteristics of the proposed system. It is the purpose of this paper to show how these specifications were developed for the RW-300 digital-control computer, the first computer designed specifically for process control. The specifications for this computer were developed as a result of a number of studies carried out on specified industrial processes over a period of almost three years.

THE PROCESS CONTROL PROBLEM

The functional and environmental specifications for a digital system arise explicitly or implicitly in answer to a number of questions which can be raised about the job the system is to perform. The job to be done by a process-control computer is that of making adjustments in process variables to attain some specified process objective in the face of variations (in raw material characteristics, ambient conditions, etc.) over which no control can be exercised. In the course of answering questions about the process-control jobs we will show how the basic specifications for the RW-300 were developed. A description of the computer will complete this paper.

Inputs

What is the source of the information, and how may it best be translated into the language of the machine? The data entered into a process-control computer is fundamentally of two different kinds: data from process instruments (measuring temperature, flow, pressure, liquid level, chemical composition, viscosity, density, etc.), and data supplied by the operator as requests for special operations from the system, or as changes to be made in system operation. The instrument data is fundamentally analog in character, normally in the form of an electric or a pneumatic signal. The signal may continuously represent the quantity being measured by the instrument (as a thermocouple voltage continuously represents the temperature), or it may represent the physical quantity being measured only at intervals (as the output of a chromatograph represents a composition by a peak voltage or by the integral of a slowly varying voltage). In addition to these analog instrument signals, there may be digital signals designating the mode of operation of the instrument.

† The Ramo-Woodridge Corp., Los Angeles, Calif.
‡ The Thompson-Ramo-Wooldridge Products Co., Los Angeles, Calif.

Although the analog input information could be transcribed manually by an operator and inserted into the computer control system in digital form, this would be an inconvenient slow operation, subject to human errors of transcription.

The digital information inserted by the operator is most conventiently presented in decimal form, so that the operator can prepare it easily. However, it may be desirable to permit the operator to initiate special requests and changes by pressing a button or operating a switch.

What kind of information must be represented, numerical or alphabetic? As can be inferred from the answer to the first question, the information read into the computer is fundamentally numerical in nature. The precision of the input data is limited by instrument precision. A precision of better than 1 per cent of full scale is unusual in common process instruments.

What is the rate of flow of information from the source? Data from the continuous type instruments are available at any time. From instruments like the chromatograph, data is available only periodically. Typically, a chromatograph signal might be sampled periodically every ten seconds over a period of five or ten minutes. Although part of the instrument data is available continuously, the variation in process variables is usually slow enough that the computer control system need to sample the instruments no oftener than once every five minutes or so. The number of instruments which must be so sampled varies from process to process and from problem to problem. A typical complex process may require as few as 25 inputs or as many as 250 inputs.

Data Processing

What must be done to the information? Must it be altered; if so, how? A process-control computer must be able to handle many different kinds of computations and manipulations of process input data. Typical of the kinds of data processing required are data interpretation, calculations for optimal control, data logging or printout of process information, and checks for hazardous process conditions or for instrument failures. By *interpretation of data* we mean the translation of readings from process instruments into numbers corresponding to the physical quantities these readings represent. This interpretation may be as simple as the application of a scale factor. It may include a linearizing operation like one which must be applied to a thermocouple to translate voltage into a temperature. Or it may require the solution of a set of simultaneous linear algebraic equations, as are required to compensate for interferences between a number of chemical compounds analyzed by a mass spectrometer. The *calculations required for control* in general require the solution of very nonlinear algebraic equations. These equations vary widely from process to process, and there seem to be no characteristics common to all of them. The *data-logging* operation requires that information be printed out in a digital-decimal form. The *checking of process instruments* and process conditions for malfunctions requires in general the comparison of observed or calculated data with certain standard values established by the operator. These comparisons and the decisions based on these comparisons comprise the checking operation.

How much time is available for processing the information? The data interpretation and control calculations need to be carried out at a frequency determined by the dynamics of the process and by the rate at which significant changes take place if the variables are slow and the time constants involved in a process are fairly long. A control calculation once every five minutes to once every hour or half hour is sufficient for most processes. The data-logging calculation may also vary depending on the state of the process and the desires of the operator. A complete logging operation of all process variables and related data once every five minutes is most frequently required. The alarm-checking operation may again be one which should be carried out very frequently or relatively infrequently depending on the variable in question. Some process checks must be done once a second; others can be carried out as infrequently as once every fifteen minutes to an hour.

Outputs

What must be the output rate of the process information? The rate at which output adjustments are made on the process is again a function of the dynamics of the process and the rate of change of the uncontrollable variables. In most processes an adjustment once every fifteen minutes to a half hour is ideal. In some processes more frequent or less frequent adjustments may be desirable.

What is the purpose of the output information? What form should it be in to accomplish this purpose most effectively? There are two principal forms of output information, just as there are two forms of input information. The first is data calculated by the computer which must be used to adjust process variables. The second is data in fundamentally digital form which must be supplied to the operator. The data supplied to the process may itself be of two forms. It may be necessary for the computer to send a digital, on-off signal to turn an instrument on or to effect some calibration or to open or shut some valve; it will be necessary to make adjustments in instrument settings by means of analog signals which correspond to digital numbers calculated by the computer as the proper setting for the instruments. Added to this is the more conventional digital output system required for data-logging purposes. As was mentioned before, this requires decimal output data. Typically, the number of process outputs is *at most* about half the number of inputs.

Reliability and Maintenance

What effect would a machine error have on the information flow, and how would it affect the operation being performed? May the operation be interrupted for emergencies or for regular periods of preventive maintenance? Conventional instruments used in the process industries are typically very reliable. Operation over periods of several years without preventive maintenance and without failure is not uncommon. However, every process instrument has some probability of failure, and the process engineer, in designing a control system, takes this into account and assures that the control system is fail-safe, that is, that a failure of some instrument or even some combination of instruments will not result in a disaster. A digital-control computer, by its very complexity, is not likely to be as trouble free as conventional and very simple pneumatic instruments. However, as might be expected in view of the fail-safe precautions commonly taken, the problem involved in incorporating a digital-control system into a process is different in degree rather than in kind from the problem of installing the conventional instrumentation. The entire control system must still be fail-safe, regardless of the reliability of the computer.

Nevertheless, the practical effectiveness of a control system of this kind is very much dependent upon its reliability, and upon the ease with which it is repaired when failures do occur. High reliability and great ease of maintenance are very important. In addition, it must remain practical to do preventive maintenance with the least possible interference with normal computer operation.

Environment

What are the environmental conditions under which the system must operate, and what effect should these have on the system characteristics? Environmental conditions for computer control systems in process industries can be expected to vary widely from installation to installation and to be very difficult. Wide temperature variations, from below freezing to somewhat above 100°F, are frequently encountered as are wide variations in humidity. Corrosive gases and vapors of one kind or another are often present in the air. Large quantities of dust are not at all unusual. Vibration due to the proximity of heavy machinery can be anticipated. Electrical power supplied to the computer may be locally generated, with the result that line voltages and frequencies change by 10 or 20 per cent of their nominal values. And it can be expected that computer use and maintenance will be in the hands of inexperienced operators who are unimpressed by delicate or fragile equipment.

System Specifications

The answers to the above questions resulted in a set of rough system characteristics which served to guide the designers when detailed decisions were made about the computer's operation. These system specifications may be stated very briefly as follows.

Because of the complexity and variety of the problems to be handled by a process-control computer, and because of the importance of flexibility in these applications, the basic machine should be a stored-program computer. The computer should have available a very large memory for storage of programs of instructions. The important requirement that the system be fail-safe suggests that the computer outputs, used to adjust process variables, be employed with conventional process controllers which compensate for the second-by-second variabilities in process conditions. The computer itself therefore need be of only moderate speed.

An input-output system capable of handling some 250 analog inputs, 100 analog outputs, and a similar number of one-bit inputs and outputs is necessary. In order that the process input-output system be suitably matched to computer speed and to typical process dynamics, all inputs should be made available to the computer and all outputs adjusted by the computer at least once a minute.

A conventional decimal input-output system is also required, but since it is not necessary to read in or print out large volumes of data, relatively low-speed devices are satisfactory.

Because of the inherent lack of precision in instrument input and output equipment, it might appear that the great precision obtainable from a digital computer would be unused in process control applications. It is certainly true that the ten-decimal-digit word length useful in some scientific computations is not required here. However, a computer precision somewhat greater than that supplied by the instruments is very desirable to make scaling problems easy for the programmer. An input conversion system accurate to one part in 256 or 512 (8 or 9 bits) would be adequate, and a word length two or three times that is appropriate. Since most of the computer's operations do not require human intervention, the binary number system is suggested, with decimal output and input conversions handled by the computer when necessary.

The reliability and environmental specifications emphasize the importance of mechanical and electrical ruggedness and ease of maintenance.

Detailed Characteristics

The RW-300 computer controller is designed to fulfill all of the requirements above as economically as possible. Its detailed characteristics were worked out by planning a hypothetical computer, putting it into typical industrial process-control systems, and then evolving a more detailed set of specifications while altering the computer's characteristics to meet new demands. This whole operation was carried out, of course, with still another objective in mind: that of providing an ultimate system which would be cheap enough to permit reasonable payoff periods in these applications.

The result, which will be described in the following sections in some more detail, is a transistorized (for reliability) general purpose, stored-program digital computer. It has a magnetic drum memory (for large capacity at low cost operable over wide temperature variations) and operates in a serial mode with fixed-length binary words. It contains analog and digital input and output facilities, the number of which can be increased or decreased without affecting the internal logic of the basic computer.

The computer uses an 18-bit binary word for numbers, consisting of a sign bit and 17 magnitude bits. The magnitude of each number is less than one. This word length is approximately twice the word length of the analog-to-digital and digital-to-analog conversions and is compatible with the accuracy of industrial instrumentation. An infrequent number of cases arises in process control requiring the use of double-precision arithmetic operations. Because of their infrequent occurrence it was not necessary to provide automatic double precision. These operations can be programmed through the use of the other instructions in the machine's repertoire. The instruction system contains 19 instructions of the one plus one address type. That is, each instruction specifies the address of one operand and the address of the next instruction to be executed. Instructions are stored in the magnetic drum memory as two adjacent words. Thus, 36 bits are used to store an instruction. The first word of an instruction pair contains the operand address and the execution time of the particular operation. The second word contains the next instruction address and the instruction code. The execution time field gives the programmer the capability of specifying the number of bits used in the multiplication, division, shifting, and digital-input instructions. These 19 instruction codes can be divided into three categories. The first category contains the basic arithmetic operations of add, subtract, multiply, and divide. The second group contains the conditional transfer or program-branching type of operations. These are transfer on a zero number, transfer on a negative number, transfer on overflow, and compare magnitude. The stop instruction is also put in this category. The third category of instructions contains the data handling operations for loading and storing either of the two principal arithmetic registers, transferring information between these two registers, shifting of numbers in the registers, merging and extracting numbers (logical add and multiply), and the digital input-output instruction.

The time required for execution of the various instructions is as follows. This includes reading the instruction from memory, obtaining the operand, and completing the operation where a minimum access time is allowed for obtaining both the instruction and the operand.

Add and subtract	0.91 milliseconds
Multiply	2.99 milliseconds
Divide	2.99 milliseconds
Branch	0.65 milliseconds
Load	0.65 milliseconds
Store	0.78 milliseconds.

These speeds of operation and the command repertoire are compatible with data-processing requirements and information-flow rates encountered in process-control systems.

The internal structure of the RW-300 can be divided into the following operational units: a magnetic drum memory, a control unit, an arithmetic unit, a digital input-output unit, an analog input-output unit, a test and maintenance panel, and an operator's control panel.

Memory

The addressable memory consists of 64 tracks of 128 words each. One of these tracks is reserved for a memory-loading program which will load the memory from punched paper tape. This track is unalterable by the programmer as a safety precaution. A second track of the 64 is used for a 16-word circulating register for fast access. The remaining 62 tracks of 7936 words are used for general storage. It is possible to write into only eight of these tracks at a time under computer control, thus giving 1024 words of variable storage in addition to the circulating register. The eight tracks are selected by means of an accessible connector plug between the writing circuits and the memory unit. The normally used 8 tracks have both a reading and a writing head which are separated by 32 words such that a number may be read from a track, operated upon, and stored back into its original address without waiting an entire drum revolution. The memory capacity for program constants and variable storage was determined on the basis of programming several typical industrial processes. Additional memory capacity was added over and above that determined in the study process since additional use of the computer once installed in a process will surely be made, thus requiring greater storage capacity for both program and constants.

Control Unit

The control unit of the computer consists of registers and counters to store information concerning the instruction and the sequence of steps in the execution of the instruction. Two circulating registers on the magnetic drum are used to store the operand address and next instruction address. The track selection portions of these addresses are transferred to a selection register at the proper time. Sector address coincidence is determined by serially comparing the sector address portions of the instruction to a sector-identification track on the drum. A third circulating register on the drum is time shared with the arithmetic unit but is used to store the execution time of an instruction when used in the control unit. A second flip-flop register is used to store the actual instruction code. Two flip-flop counters are used to distinguish the digit times in a word time and to sequence the steps involved in executing an instruction.

Arithmetic Unit

The arithmetic unit of the computer consists of two main circulating registers, the time-shared register mentioned above, and an adder. The principal arithmetic register contains the results of instructions and holds one of the operands in the majority of instructions. The second arithmetic register is used to hold the multiplier and remainder for multiplication and division. Because the contents of the second register are readily interchanged with the principal arithmetic register, it can be used as a one word time, fast access, temporary storage. The time-shared register is used for multiplication and division. It is not addressable.

Digital Input-Output Unit

The digital input-output unit contains the basic facilities for reading in 6 bits from a paper-tape reader and for putting out 6 bits for a paper-tape punch and/or typewriter. A large number of digital inputs and outputs other than the paper tape and print inputs and outputs is possible without changing the basic computer. The total number of addresses available for digital input or output devices is 64 and the maximum word length for these inputs and outputs is 18 bits. These digital input-output facilities are more than adequate for the alarm output and operator input instructions encountered in process control.

Analog Input-Output Unit

The analog input-output system does not require programmed control from the computer. All analog-input quantities are converted to a digital number and stored in specific addresses in the memory. All analog outputs are read automatically from the memory and converted to analog quantities for control. Input quantities are obtained from the memory by the computer as are any other numbers stored in the memory. The number of analog inputs and outputs required for process control varies considerably between applications. The maximum number of inputs could be as high as 512, and the number of outputs could be as high as 256. Changes in the number of analog inputs and outputs used does not change the basic computer. The input-output system is a cyclic system in that all inputs and all outputs are read during a fixed length of time and then the cycle is repeated. This makes the latest information available to the computer at any time where the maximum delay in an input is a matter of a few seconds. Ten binary digits are used in converting analog-to-digital and digital-to-analog. The basic full-scale analog inputs fall into two categories. These are low-level signal inputs and high-level signal inputs. The standard low-level input is 0 to 10 millivolts while the higher-level inputs are standardized at 0 to 10 volts. The latter may be obtained from instruments which have current outputs sufficient to give 0 to 10 volts full scale. The analog outputs are standardized at 0 to 5 milliamperes, although higher current or voltage outputs can be specified without loss in accuracy. A single converter is time shared for all analog inputs and all analog outputs, and switching of inputs and outputs is done both with relays and electronic switches.

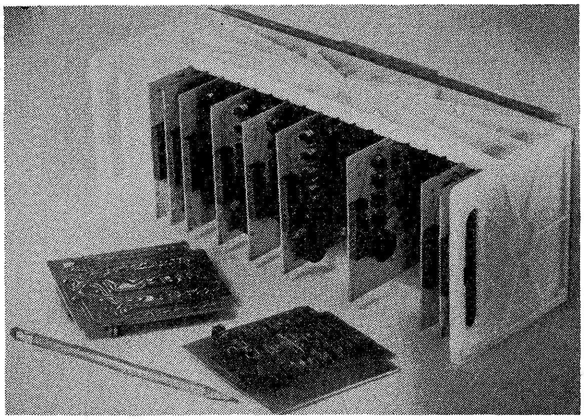


Fig. 1—Computer module and insert cards.

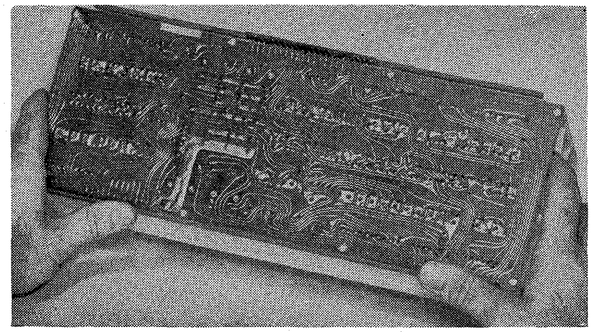


Fig. 2—Module bottom etched wiring.

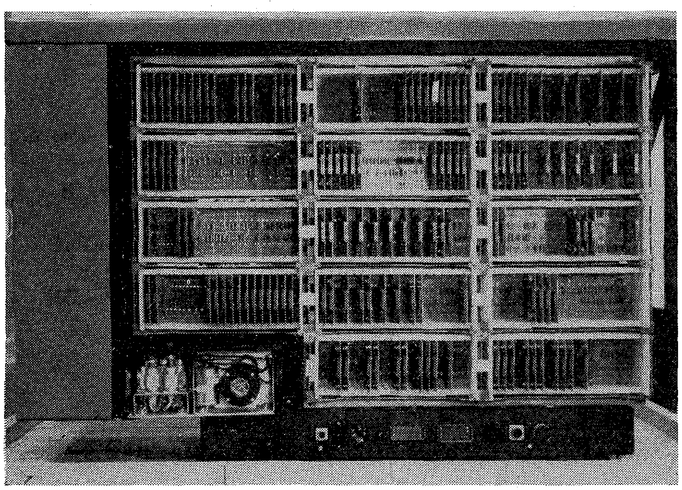


Fig. 3—Computer subframe holding 14 modules.

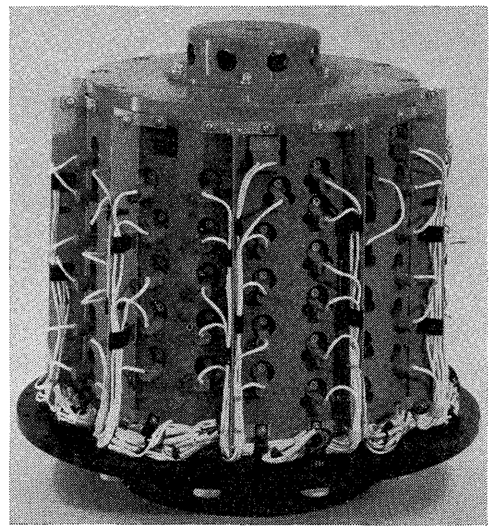


Fig. 4—Magnetic drum memory (cover removed).

Test and Maintenance Panel

The test and maintenance panel contains the facilities for code checking of programs, marginal testing, and for on-line trouble shooting of the machine. The normal automatic sequence in executing an instruction is to first obtain the instruction from the memory, then to obtain the operand, and finally to perform the indicated operation. It is possible to interrupt this sequence of operations from the test and maintenance panel and to do these operations in two steps at a manually controlled rate. The first step, under the control of the operator, picks up the instruction from the memory address and through the aid of a built-in oscilloscope the operator may inspect the address of the operand, the address of the next instruction, the instruction to be performed, and the execution time. A second step permits the computer to pick up the operand and to perform the indicated operation. The operator may inspect the results of the operation and note any changes in the next instruction address. The oscilloscope also allows the operator to look at the contents of the arithmetic registers and other important logical points in the computer. Neon lights are also provided which indicate the contents of the control registers and certain designated flip-flops.

Controls to adjust the computer voltages and clock-pulse amplitude for marginal checking are located on the maintenance panel. Digital input switches on the test and maintenance panel can be used for program checking (break-point switches) so that portions of programs may be checked without running through the entire program.

Operator's Control Panel

The operator's control panel contains the push buttons and indicating lights to start the computer, to stop the computer, to turn the power on and off, to resume at the point of stopping in the program, and to load the memory from punched paper tape.

The high reliability, ease and speed of maintenance, and environmental immunity specifications are by far the most important requirements for any process-control computer. RW-300 reliability is achieved by using only high quality components, maintaining rigid quality control on the use of these components, and derating all components a considerable amount. Conservative circuit-design techniques with wide tolerances, for component variation, load variation, and voltage variation, are used throughout. The RW-300 computer is almost entirely transistorized. Both germanium and silicon diodes and transistors are used in the machine.

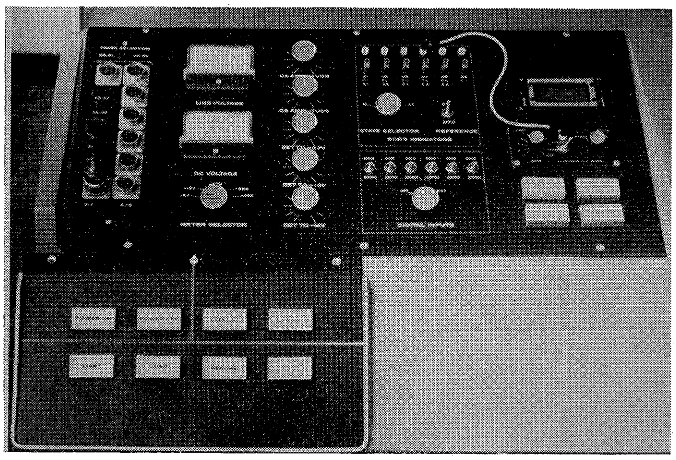


Fig. 5—Test and maintenance panel (top), and operator's control panel (lower left).

The construction of the machine is unitized and modular in nature for rapid location and replacement or repair of units, subunits, and components. The building block of the basic computer is a module as shown in Figs. 1 and 2. These modules plug into frameworks where the interconnection between the modules is minimized. Each module contains several flush-etched wiring inserts where the etched wiring on the module between the inserts represents the majority of wiring in the machine. Components are also mounted on the module bottoms. Fig. 3 shows the computer subframe which houses the arithmetic and control unit, a portion of the memory unit, and a portion of the digital input-output unit. This modular construction greatly reduces the maintenance time both in locating a trouble and repairing the trouble. The use of etched-wiring techniques increases the reliability of the machine over those using conventional wire and solder techniques. Each insert and each module contains test points which are accessible when the machine is in operation.

The magnetic drum, as shown in Fig. 4, is a sealed unit so that corrosive gases and vapors, dust, etc. will not enter. The construction of the drum is similar to that used in drums for airborne applications so that temperature variations and mechanical shock and vibration will not alter its operation. Fig. 5 shows the RW-300 computer with the test and maintenance panel exposed. Fig. 6 is an over-all picture of the RW-300 showing its size, which is that of a conventional office desk.

Fig. 6—The RW-300 digital-control computer.

Conclusions

Specifications for the RW-300 were developed as a direct result of studies of industrial processes. Because the computer was designed specifically for application to process control, it will make available to this application the flexible, sophisticated computing and control ability necessary to implement integrated control systems.

Optimized Control through Digital Equipment

E. J. OTIS†

By optimum control of a process we mean the achievement of a series of objectives in the production of a specific product. The objectives to be achieved are primarily of an economic nature, although they frequently are expressed in terms of the process input and output. In other words, the problem is one of maintaining product characteristic and quality with the minimum use of raw materials and at a minimum production cost. Furthermore, the problem includes maintaining continued product output of specified characteristics in the face of changing plant conditions and variations in raw materials, as long as the cost of the product is within the limits specified by a competitive market.

† Daystrom Systems, Div. of Daystrom, Inc., Lo Jolla, Calif.

At present, the control of a process is inadequately relieved by resorting to the use of "minor control loops" and a human operator to close the major loop around them. (See Fig. 1). These minor loops consist of a transmitter-recorder-controller combination which measures a process variable and maintains it at a desired level. The control of this variable is effected with respect to a set point without regard to the state of the total process or to the value of any of the other variables. In some cases the value of one variable and its excursions is used to control another variable in a configuration called "cascade control." However, this type of control, although more sophisticated than the simple minor loop, can only be used in the few cases where a simple, known, and non-time-varying relationship exists between two variables while the over-all control remains effectively composed of the series of minor loops. The integration of each minor loop into the whole control system is then effected through the operator, who observes the process state on different recorders and adjusts (adapts) the controller to process conditions.

In other words, by adjusting a set point, after he has observed the state of the process as presented by numerous recorders, the operator's skill and knowledge of the process are resorted to in closing a second loop around the "minor control loops." Through this major loop the interactions of the different variables are now integrated into the control system.

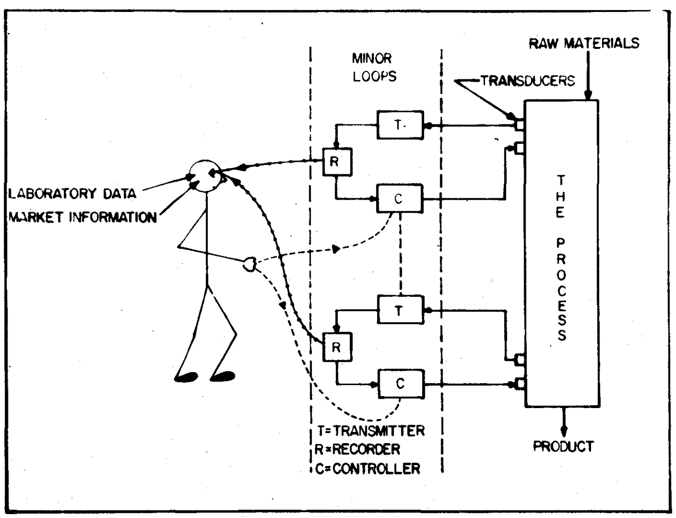


Fig. 1—Process control today.

Because of the complex nature of today's processes, however, a human operator can not keep track of all the necessary variables and their interrelationships, as well as their effects upon product cost and quality. And, although he is more intelligent than any machine that can be conceived, the human operator fails when it becomes necessary to digest large amounts of data and respond with adequate speed to an increased system complexity.

Furthermore, even though he might achieve an apparently satisfactory combination of settings for the operation of the process, he rarely, if ever, knows within a reasonable period of time whether this is the best combination, *i.e.*, whether the process is at maximum efficiency.

It is becoming evident that the operator can not attend the major loop efficiently because of increasing process complexity. An attempt has therefore been made to close the control loop through equipment rather than through the operator. This is not meant to replace the operator but rather to place him in parallel so that the control equipment can function effectively and sufficiently fast in the making of decisions. These decisions may be routine; they are almost invariably numerous. The operator can therefore be relieved of making many routine decisions, and can intelligently monitor the process and provide over-all direction in parallel with the computing-controlling equipment. (See Fig. 2.)

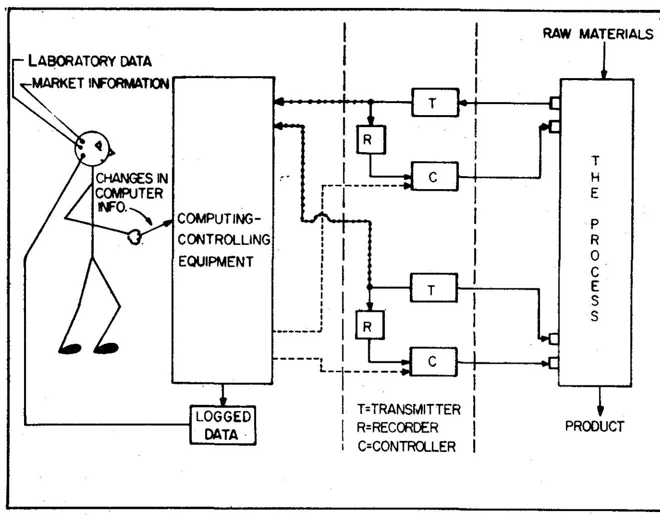


Fig. 2—Optimized process control.

Since the computing-controlling equipment is placed in parallel with the operator, it must be capable of communicating with both the operator and the process. Thus, it must be able to accept signals directly from the transducers and/or transmitters, as well as from the operator, and its output must be recognized by, and be intelligible to, the controllers and to the operator.

After considerable study of various processes, a system has been designed to close the loop around minor-loop controllers. This system centers in a general-purpose digital computer and, as is the case with any other computing system (industrial or military), the peripheral equipment becomes important. So besides the computer we have system components such as input multiplexer, analog-to-digital converter, computer input-output equipment, and control output.

Although many hours could be spent in discussing the characteristics of these component systems, at this time we shall content ourselves with indicating their main design criteria.

Equipment Characteristics

Before discussing the particular characteristics of component systems, let us consider three of the major decisions that influenced the design of the over-all system.

First of all, it was decided to maintain minor-loop analog controllers for two very practical reasons. 1) They do provide continuous control of the different variables. In order for the computer to duplicate such control, it being time shared among all the variables, it would have to be designed to operate at a tremendously high speed—one well beyond that considered today to be *reliably* attainable. 2) In case of computer failure, the process would be held to the most recent set points, changes of which could still be made manually.

The second major decision concerned the data processing equipment. Here, the decision to be made was whether to use a general-purpose or a special-purpose computer. This decision depended upon the consideration of numerous factors. The main reasons for deciding to use a general-purpose computer were its flexibility (required to control a complex process under varying process conditions) and its capability of adapting—or better yet, self-adapting—its program to varying conditions in the process. When discussing the computer, we shall see how this capability to adapt itself enables it to provide meaningful control signals.

Last, and most important, the reliability of the equipment, which must be capable of operating continuously, must be carefully considered before installation in a process plant. This requirement for reliability dictates the use of solid-state components throughout and a minimum use of electromechanical devices. Keeping in mind the retention of the minor loops, the usage of a general-purpose computer, and the employment of solid-state components properly derated in circuits designed and constructed to withstand an adverse plant environment (temperatures to 120°F, high humidity, and a generally corrosive atmosphere), we can now look at the component systems required to close the major loop.

System Input Section

The input section (Fig. 3) supplies the control system with the data that determine the operating conditions of the process, the present settings of variables to be controlled and, of course, the program indicating the control variables and the method of control.

These inputs are derived from two sources, the process variable measurements, and the operator.

The process variable measurements include those which yield information on the state of the process. These can be temperatures, flows, and levels, as well as physical and chemical characteristics of raw materials used, and the characteristics of the end product.

Types of analyzers (instruments which are used to measure the physical and chemical characteristics of both raw materials and end products) will be chosen for the particular process. While, control variables such as temperatures, flows, levels, etc., are derived from transducers where the signal level can be as low as 10-50 mv full scale. Since these signals are electrical analogs, they must be quantized to at least one part in one thousand. The input multiplexer must therefore be able to handle such low-level signals and to restrict noise levels in excess of 10 μv. If the multiplexer can not handle such low-level signals, an amplifier is required for each input. However since we have found that there is a large number of inputs (300-1000), the cost would be prohibitive. Therefore the use of amplifiers must be avoided.

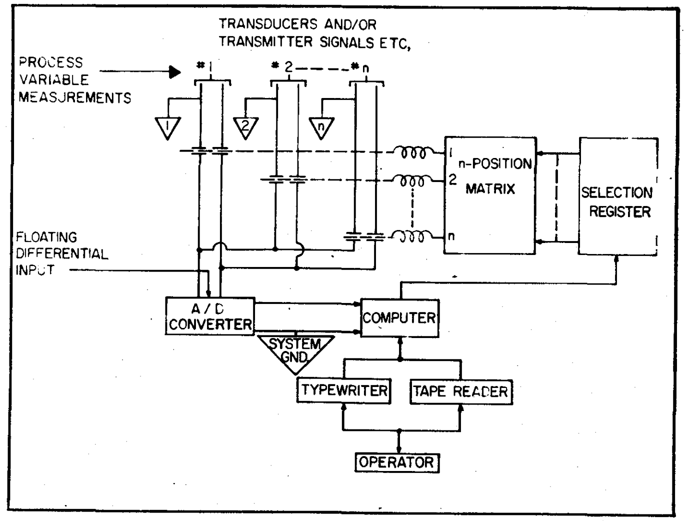


Fig. 3—System input section.

Furthermore, in order to circumvent ground-loop problems, the input multiplexer must be able to switch both sides of the signal line. The most reliable component that can switch both sides of a low-level signal line and still not introduce unpermissible noise, is the mercury-wetted relay. This component has displayed long life (billions of operations), lack of contact bounce, and a very stable contact resistance. The Clare HG2A series has been found satisfactory both from the point of view of reliability and of drive requirements. However, like any other component handling analog signals, there are noise problems that have to be contended with. Noise introduced by these relays is composed of 1) a transient portion which lasts for approximately 3-5 milliseconds and 2) a dc portion. The transient portion of the noise is caused by flux build-up and decay, while the dc portion is caused by thermocouple effects resulting from thermal gradients existing between the two sides of the relay.

The transient portion of the noise, although minimized by using appropriate circuits, still exceeds the permissible 5-10 μv level. However, as will be shown below, this transient is rejected through the use of an appropriate analog-to-digital converter.

The dc portion of the noise is minimized by appropriate packaging and by providing a "shorting-bar" wiring arrangement inside the relay. In this manner we have found that the noise level can be reduced to below 10 μv (actually of the order of 1 to 5 μv). This enables us to maintain the required signal accuracy of 0.1 per cent or better.

These relays are driven directly from transistor circuits utilizing the RCA 2N217 germanium transistor in a random-access switch arrangement. This feature, permitting us to connect any input variable at random to the system as directed by the computer, has been incorporated for three very important reasons: 1) It permits switching to a standard input for recalibration purposes at any desired time, 2) it enables the equipment to sample discrete variables at time intervals shorter than the scanning cycle (this is necessary in order to scan quantities that must be integrated with respect to time, such as flows), and 3) under a self-adaptive computer program, the equipment can sample more frequently those variables that become critical under varying process conditions.

The analog-to-digital converter must also possess special characteristics determined by the signals generated in a process plant. Besides the fact that the desired signals are very low in level, they usually contain electrical noise produced by plant power equipment, such as motors. In addition, noise pick-up is frequently found when long-signal leads (100-500 feet) are required. Because of the noise and ground loop problems mentioned above, the converter itself has been designed with the following features:

1) It provides input isolation required because of differences in ground potentials throughout the plant with respect to the control system ground. This ground potential difference can be as high as 500-1000 volts during transient conditions such as storms. Furthermore, this differential-isolating type of input rejects common noise.
2) It can resolve 10 mv (or higher signals), with accuracies to 0.01 per cent.
3) It integrates over the sampling period of 1/5 to 1 second. This sampling rate of 1 to 5 conversions per second might at first appear to be slow, but the speed has been selected to conform to the type of signal present in a process plant where there is noise pick-up. If rapid conversion were required, a tremendous burden would be placed upon the computer if it were to average many of these readings. Using an integrating converter, however, the computer is freed to perform correction and control computations while the converter is digitizing a reading.

The converter is completely transistorized and known as the DADIT (Daystrom Analog-to-Digital Integrating Translator). It integrates over the entire period of time that the input is connected to it (1/5 second) and is not affected by transient noise present in the relay multiplexer (5-msec duration). Actually, the longer the integrating period, the better the noise rejection. Although the multiplexer and the analog-to-digital converter are matched in speed to cope with the "noisy" type of signal that they are expected to switch and quantize, they might not function fast enough if readings are to be made from a number of variables at intervals closer than 1/5 second apart. By the same token, since a "faster" converter would not provide adequate filtering, and hence meaningful signals, two or more converters are used when it is necessary to solve the above-mentioned problem.

The multiplexer and analog-to-digital converter, supplemented by any special equipment used to tie in product and material analyzers, form the input to the system. It is these units which monitor the condition of the process as well as its performance.

In addition, the system is given instructions by the operator, such as pertinent results of laboratory analyses concerning a new raw material, market information, or other commands the operator wishes to introduce. The operator may even want to enter new programs. For this purpose a paper-tape reader and a typewriter have been provided. The reason for including a typewriter rather than a keyboard is so that a record containing instructions given to the computer will be always available. The typewriter is used for the manual introduction of data as well as for the control of the computer.

Computer

Once the data have been converted to the digital form, they are no longer affected by component noise and drift, and from then on we deal with digital accuracies.

The computer, in receiving these data, must first operate upon them, linearize and scale factor the numbers in order to account for different transducer characteristics and, in general, convert a set of numbers to meaningful quantities representing physical measurements as required by the computer program or for logging sheets. The values of some measurements have to be compared with alarm set points while others have to be integrated and correlated in order to provide information concerning the amount of output, the process efficiency, and the determination of product quality. This part is easy to program and is designed to yield information concerning the state of the process at the present time.

The computations that determine the control signals are complex (because we are dealing with complex processes) and nonlinear time-varying in their characteristics. They are therefore difficult to define in popular mathematical terms, and the control program is largely biased by knowledge of the process, the interactions of different control variables, and their effects upon the output (product).

The process is usually described with a set of nonlinear differential equations. The object is to make these equations linear for a particular process state, although they are over-all nonlinear. This can be achieved by creating relationships which define the coefficients of these equations in terms of the measured variables (process state), the control set points, and the process output. The system then can, for a set of inputs and outputs, select the appropriate set of coefficients, and by so doing, create a piece-wise linear approximation of a nonlinear process.

The computer can then solve these equations with respect to an optimizing criterion and create the control changes that will yield optimum performance. Actually the computer is expected to iterate through many solutions, choose the most attractive one, and introduce it into the process. It will then observe the actual results obtained as compared to the expected results, and will compensate for discrepancies by readjusting sets of coefficients. In this fashion the computer continuously adapts its mathematical model of the process to the actual case, accounting for different plant conditions that occur, many of which could not have been anticipated at the time of system design.

It is this capability of self-adaptation that dictates the use of a general-purpose computer.

The computer must be fast with respect to the process which, in computer language, is fairly slow.

The computer designed for this purpose is a 50-kc serial-binary machine. Word length is 20 bits plus sign and parity. This provides a computational accuracy of one part in 10^6, or well in excess of the accuracy of the measurements. It has a coincident-current memory that can be as large as 16,384 (2^{14}) words. It is a single-address machine with a speed of 1.3 msec for an addition, and 10.1 msec for a multiplication, including look-up. It has the full complement of arithmetic operation commands and branch and shift commands, as well as special ones used for controlling the input and output functions.

The computer, through its "analog-input" command, can select a particular input, connect it to the analog-to-digital converter, and then proceed with another computation while the signal is being converted. One-fifth second later (at the 5 integration/second rate) it causes a new input to be connected to the converter, accept the integrated reading of the previous input, and proceed with the new computations.

Again the computer, like all the circuits in the input section, is completely solid-state, uses approximately 3500 transistors, (RCA 2N217), and 3000 diodes.

System Output

The output of the system serves the dual purpose of introducing the required control changes to the process and of informing the operator of the state of the process, the changes made, and any other information derived from the input variables. (See Fig. 4.)

It is very difficult to generalize concerning the design of actuators since they depend upon the characteristics of the actual equipment. Sometimes an on-off control will suffice, while in other instances variables are controlled in discrete steps. A digital stepping motor, such as the Sigma Cyclonome stepping motor (magnetically indexed rotor) and the Digitork, announced by The Teller Company, frequently can be used for this purpose.

The accuracy with which set points are set depends upon the process and the particular variable. It is definite, however, that today equipment can control much narrower ranges than an operator can.

The output for the operator is derived through tape punches and typewriters. The computer can select one to eight punches if the system so requires. The reason for using punches rather than typewriters directly is based on two considerations: 1) The particular punch used is faster than a typewriter (Western Electric punch type BRPE #1, 60 characters per second), and 2) they are considered to be more reliable than the relatively complicated typewriter. The typewriter can be "down" yet the system can continue to operate since the data are accumulated on punched-paper tape.

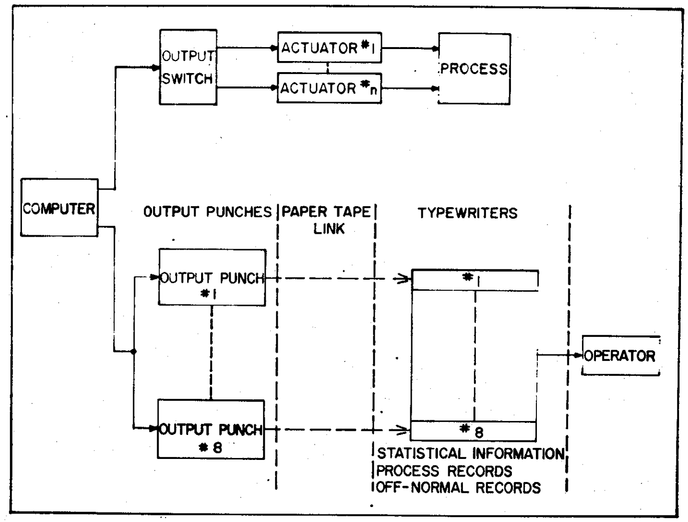


Fig. 4—System output section.

At this point, a word concerning reliability is again in order. The responsibility placed upon the control equipment described above is tremendous. In order to meet the requirements, this equipment must operate without failure for many months, if not for years. Extreme care therefore must be exercised in choosing the components to be used and in designing its circuits. There is room neither for components that have not been tested over long periods of time, nor for critical circuits. Normal maintenance must be performed while the equipment is operating, and check problems should be run through automatically at predetermined intervals of time in order to ascertain proper functioning. Finally, rapid troubleshooting procedures and faulty component location are musts.

In conclusion, rising processing costs are forcing industry to adopt automation. Optimized control will make new processes economically feasible and, therefore, practical. The challenge is here, and the future will show how well we are meeting it.

Acknowledgment

The author gratefully acknowledges the helpful assistance in preparing this paper given him by Eric Weiss, David Taylor, Charles Taylor, Bill Waddell, and Wilbur Erickson, all of Daystrom Systems, Division of Daystrom, Inc.

Discussion

Question: In your opinion, is Daystrom's Heath analog computer adaptable for use in the system described? If so, wouldn't this simplify the system by eliminating the analog-to-digital converter?

Mr. Otis: While you are right in remarking that an analog computer would eliminate the use of an analog-to-digital converter, it would not meet the requirements in terms of reliability as well as computational accuracy, time sharing, and flexibility that we were able to design into the present system.

Question: Is the system described for a specific installation?

Mr. Otis: No, it isn't, although the first two are committed and will go in specific installations. The design is based on a study of many types of processes.

Question: Would you please repeat the arithmetic operation?

Mr. Otis: I assume by this you mean the speed of the machine. It is 1.3 milliseconds for addition and 10.1 milliseconds for a long multiplication and division. Shift commands fall into the 1.3 milliseconds or what we call the short commands.

Question: Is the coincident current memory the only memory? Is there any drum?

Mr. Otis: No, the coincident current memory is the only memory in this system. However, the machine is designed to work with a tape deck.

Question: In the presence of interaction in a process, how is optimum adjustment of the variables achieved? What criterion is established for behavior of multivariable systems?

Mr. Otis: It is a difficult question to answer in the sense that the criterion can only be established if you have a particular process that you are talking about. But, as I indicated in the paper, you usually have many sets of simultaneous equations and in solving them together, you take into consideration the interaction of the different variables.

It is this interaction between the different control variables that this equipment introduces into the control system. We want to take into consideration this interaction. The criterion is usually an economic one. You want to produce a product of a specific quality at minimum cost. This might be achieved by the use of minimum input power, the use of minimum raw materials, maximum catalyst life, or minimum processing time, whichever the criterion might be. You might have three or four criteria and go down the line until you find one that you will give you the best answer.

Question: Would you please repeat the type of your memory in your computer and mention its capacity?

Mr. Otis: It is a coincident current memory, transistor driven, and depending on the system we can have from 1000 to 16,000 words in this memory.

In other words, the computer has the circuits addressing up to 16,000 or 2^{15} words.

Real-Time Presentation of Reduced Wind-Tunnel Data*

M. SEAMONS†, M. BAIN†, AND W. HOOVER†

Introduction

THE effective use of wind-tunnel testing in determining aerodynamic properties of a body is very much dependent upon the reliability and speed with which wind-tunnel data can be reduced. The ability to provide reduced aerodynamic coefficients in real time, or on-line, greatly increases the operating efficiency of the wind tunnels and thereby reduces expensive wind-tunnel time required for each test. This paper describes a system for presenting reduced wind-tunnel data in real time for the two wind tunnels at the Jet Propulsion Laboratory (JPL).

The requirements for data-handling equipment and data-reduction procedures for wind tunnels throughout the country are quite diverse, and depend upon the wind-tunnel design and the type of tests for which they are used. The supersonic wind tunnels involved in this description are used for force tests, pressure tests, and miscellaneous research studies, and include a variety of force-balance systems. Consequently, the problems associated with on-line data reduction for these tunnels can be considered as representative of the problems associated with tunnels generally.

Real-time reduced-data presentation requires a system consisting of three major parts: 1) the instrumentation necessary to convert force, moment, pressure, and angular measurements into a form compatible with available computing equipment; 2) an operational system including the computer program and methods for accomplishing the data processing; and 3) a system for presenting reduced coefficients within a specified time interval in a form allowing use of test results to control the test program. An earlier data-reduction system providing reduced data on a daily basis has been described in the literature.[1] Most elements of the new on-line system have now been developed. The completion and installation of the new system will be accomplished step by step and will not involve wind-tunnel downtime. Prime considerations in developing the new on-line system have been economy in capital investment, high reliability, and flexibility in handling the variety of test types.

* This paper presents one phase of research carried out at the Jet Propulsion Lab., California Inst. Tech., under Contract No. DA-04-495-Ord, sponsored by the Dept. of the Army, Ordnance Corps.

† Jet Propulsion Lab., California Inst. Tech., Pasadena, Calif.

[1] W. R. Hoover, J. J. Wedel, and J. R. Bruman, "Wind-tunnel data reduction using paper-tape storage media," *J. Assn. Computing Mach.*, vol. 3, pp. 101-109; April, 1956.

Wind-Tunnel Testing

Wind-tunnel tests are conducted by immersing an accurate scale model in the wind-tunnel air stream and recording a set of readings related to physical quantities such as forces, moments, pressures, or angles. Most tests requiring on-line presentation of reduced data are force tests or pressure tests. A point of force-test data consists of a set of forces and moments with respect to some reference system, the angles of attack and roll, the pressures necessary to specify tunnel operating conditions, and the free-stream Mach number. A point of pressure-test data consists of a set of readings giving the pressure at points on the body surface.

At JPL, force and moment measurements are usually made using two types of balance systems. The six-component external balance system resolves three independent components of force and three independent components of moment about a reference system fixed with respect to the tunnel. The strain-gauge system isolates forces and moments about a reference system fixed in the model. The readings from the two balance systems are four-digit numbers. The external balance readings require a code digit to indicate the set of balance constants needed in the data-reduction process.

Pressure tests are conducted by measuring the pressure at fixed positions on the body surface. Measurements are made by connecting each point with a pressure transducer which transmits a signal to the automatic pressure readout system.

The system is capable of digitally recording as many as 192 pressure readings and of automatically ratioing each reading to a prescribed pressure.

The purpose of force-data reduction is to convert test readings from the balance coordinate system to a coordinate system suitable for engineering purposes. This process involves a sequence of changes of scale, rotations, and translations to obtain dimensionless aerodynamic coefficients. A general data-reduction scheme has been evolved utilizing a sequence of vector by matrix products which includes the transformations for both the external and internal balance systems. In reducing the data, two classes of constants exist, those fixed for a complete test and those changing as the test progresses. The air-off zero correction is a point taken with air off and at zero angle of attack and indicates the balance condition for zero forces and moments. The static tare is the change in balance readings under air-off conditions caused by movement of the model center of gravity during model pitch angle rotations. Moment transfer distances are dependent upon the model configuration.

Pressure-test data reduction consists of computing ratios for all model pressures to a set of known pressures, in addition to computing profile averages, local Mach numbers, drag, and other aerodynamic forces. In general, pressure-test data reduction is much simpler than the force-test reduction used at JPL.

System Requirements

One of the primary considerations in the presentation of wind-tunnel data is the selection of the proper form of raw-data record. Accumulated experience with the paper-tape storage system has verified the advantages previously claimed,[1] and the real-time data-reduction system will retain this feature. The large volume of data output from wind-tunnel operations requires fast and reliable data accumulation equipment which does not limit the operational speed of the wind tunnel. The system must accept information from either of two wind tunnels and convert it to a standard form for data reduction and presentation; also, the system must be flexible enough to handle all categories of force tests presented by a wide variety of engineering requirements.

Real-time presentation of reduced wind-tunnel data requires that all of the final data be presented before normal model or tunnel conditions are changed for a succeeding run. It is desirable that all data-handling units be integrated and utilized in such a manner that real-time results used for monitoring the test correspond to the final coefficient tabulations and plots required for engineering reports.

System requirements necessitate tabulating and plotting directly from the raw-data record. This first inspection serves to determine whether raw data appear reasonable and sufficient (curves are smooth and defined); also, it provides sufficient information to allow testing to proceed on the basis of raw-data presentation in the event of computer breakdown.

The operating rate of the wind tunnel is 15 seconds per point; this is an average time required to change an independent variable in the test procedure, allow the tunnel to reach a stable condition, digitize the balance readings, and punch a point of raw data into tape. A run of data includes a series of related points; for each run of data, there is an additional two minutes available, thus increasing the average time available for computing purposes on a run basis to 25 seconds per point.

The basic computer system to be used is an ElectroData Model 202. To increase the efficiency of the system for use on data-reduction problems, several modifications were required. A second photoreader and input order were added to the computer, giving the system two independent input stations which are internally controlled and can be actuated either manually or by programmed command words. A second teletype punch with independent output commands was added to the system. Presentation of wind-tunnel data in both tabulated and plotted form from paper tape requires two output tapes with different data organization for use in the plotter and tabulator, respectively.

Instrumentation

Force Tests

The six components of force sensed by the external hydraulic balance are measured on an automatic servo-controlled beam balance. The balance position is converted to digital form by means of a shaft-position digitizer. The force components obtained from the internal strain-gauge balance are measured by a strain-gauge bridge with servo follow-up which also drives a shaft-position digitizer.

The digitizers employed are the double-brush decimal encoders manufactured by the Coleman Engineering Company. These devices require a set of readout relays before the digital data can be transmitted to the tape punch. In order to ease the interpretation of raw data, it is necessary that the output readings be recorded in both plus and minus values. The conversion of digitizer readings to correct plus and minus decimal numbers is a relatively complex operation requiring a sequence of relay closures. Since the readout relays are time-shared among the digitizers, the relay cycling time would be prohibitively long and would slow the scanning rate if it were not for a novel method of readout devised at JPL which requires only one relay closure time per digitizer, approximately 15 msec.

The scanner records the digitizer and keyboard readings on a punched paper tape using a Teletype BRPE-2 60-digit-per-second punch. The slow-speed scanning of data-source words is accomplished by telephone-type stepper switches while the high-speed scanning of individual digits is accomplished electronically using magnetron beam switching tubes and transistor switches.

The data are recorded on punched tape and then verified on a tabulator; in addition, selected words in the scan are plotted on an automatic tape-controlled plotter (the raw-data plotter indicated in Fig. 1). The raw-data plotter can plot as many as twelve components on a single 30- by 30-inch sheet of paper. After the data have been displayed on the tabulator and plotter, the tape can be read on a high-speed tape reader which is controlled by and transmits information to the computer. After data reduction, the final data tapes are read by tape readers located near the computer's output punches, and these readers are controlled by and transmit information to the final data tabulating and plotting facilities. The tabulating machine is a Burroughs Sensimatic used as a word-at-a-time printer. The data for a word are assembled in a magnetic-core memory which together with associated circuitry controls the type bars of the Sensimatic. The final data are plotted by three Electronic Associates, Inc., Model 1100D Variplotters. These plotters use 11- by 17-inch graph paper and employ programmable symbol printers.

Pressure Tests

The data from pressure tests are recorded on a multipressure measuring system developed at JPL.[2] This system scans as many as 192 pressure sources by means of pressure selector tubes. The unknown pressures are channeled to a single pressure transducer; the transducer reading is converted to a binary-encoded four-digit decimal number in a high-speed analog-to-digital converter, and the data are subsequently recorded on punched paper tape. The 192 pressures can be scanned in approximately 40 seconds. For a lesser number of pressure tubes the scan time is correspondingly shorter. The output of the pressure transducer can be scaled to any arbitrary value, thus obviating much of the data reduction. As an example, if it is required that all model pressures be recorded as a ratio to stagnation pressure, it is a simple matter to calibrate the system by feeding stagnation pressure into the machine and adjusting the encoder to read 1000. The accuracy of the multipressure measuring system is better than 0.2 per cent of full scale. This accuracy is achieved by the use of a single pressure transducer, allowing detailed observation of any transducer zero drift or calibration shifts. In addition, the method of scanning provides for the gauge to be connected to a vacuum system before each reading, thus eliminating a major source of gauge error, the hysteresis effect.

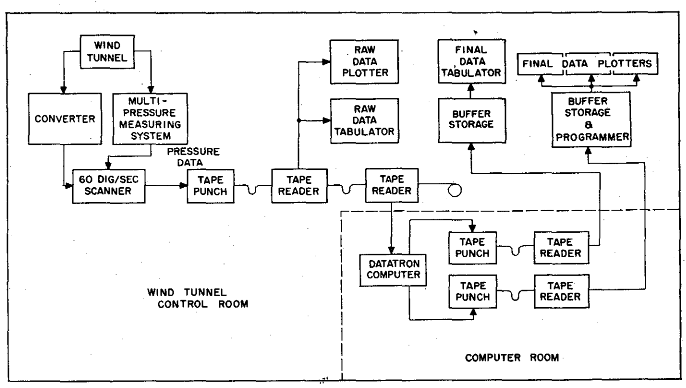


Fig. 1—Block diagram of wind-tunnel data-reduction system.

After being punched the tape is fed to a high-speed tape reader controlling the Burroughs Sensimatic printer. Selected pressures can be plotted on the Electronic Associates Variplotter for the presentation of pressure contour curves.

Data Reduction of Force Tests

The computer program is written to conform to a standard reduction procedure which converts information from the coordinate system of the force balance to aerodynamic coefficients in any or all of four coordinate systems which may be specified in a test. The same program is used to reduce data from either internal or external balance systems; the procedure accounts for characteristics which are functions of the tunnel, suspension, balance, and readout system.

The basic program for reduction of force tests consists of eleven steps, each written as a separate routine or subroutine. Essentially, each of the steps is written as a matrix-by-vector multiplication, and any change required usually consists of a minor alteration of a matrix at a particular step. In some instances it is necessary to bypass or reprogram a step completely; either type of change

[2] M. B. Bain, "A multipressure measuring system," IRE Trans. on Instrumentation, vol. I-6, pp. 18-22; March, 1957.

affects only an isolated portion of the entire program. This program structure minimizes the amount of pretest programming and checkout prior to the test date.

Preparation of a test for reduction consists mainly of setting up an efficient method for handling run parameters such as configuration constants, roll angles, Mach numbers, and deflection constants. As standard practice, all combinations of constants are prestored in computer memory and programming changes are provided to select and check proper constants as the run number changes. Infrequently, the schedule of constant changes overtaxes the limited memory capacity and a second photoreader is used to introduce required changes.

Fig. 2 is a flow diagram outlining the sequence of operations on a point of raw data as executed by the computer program. Assuming that the basic program, the required program alterations, and all fixed constants have been prestored in the computer, the input is actuated for read-in of the first data point from the accumulation system. The raw data are permuted to a fixed order, and a check is made to determine whether the point is an air-off-zero point. (If the point is an air-off-zero it is flagged and stored in memory for later reference.) Introduction of an air-off-zero point indicates the necessity for changing run parameter constants; selection of such constants is made either from memory or from an external photoreader tape. A coded word structure is used to flag input of points from a static-tare run; pitch and rolling-moment data from these points are used to form a static-tare table which is in general applicable to a series of related runs. Upon read-in of an air-on point, the static tare for the raw-data angle of attack is determined and corrections for air-off zero and static tare are applied to the raw data. The corrected raw data are then reduced through the body-axis coordinate system. A set of code words is used to determine to which of the coordinate systems the reduction is to be carried and the form of the aerodynamic coefficients, which are punched on two separate output tapes; one tape is for tabulation on a Burroughs word printer and the other drives automatic plotters for final data plots.

The pointwise reduction program as written reduces six-component external balance data to body-axis coefficients in 15 seconds. Results are presented with no increase in inaccuracy and with all anomalies accounted for. Extension of the reduction to additional coordinate systems requires approximately 2 seconds for each system desired. These times include the operations necessary to scale and otherwise adjust all output to forms acceptable to the listing and plotting equipment.

In the proposed system, the total data-handling time for a typical point of tunnel data will be approximately 23 seconds. Operating times of system blocks are as follows. 1) Digitizer, scanner, and punch require 1½ seconds per point; during this period the independent variable for the next point may be set in the tunnel. 2) Raw-data tabulation and plotting will require 12 seconds; if the computer is ready to accept a point of data when readout is completed, the raw data point will first be read into the computer and then into the raw-data tabulating and plotting system in such a manner that reduction and raw-data presentation operate concurrently. 3) Computer read-in, reduction, and punchout of final tabulating and plotting tapes require 18 seconds per point. 4) Tabulating and plotting of final results require 3 seconds per point.

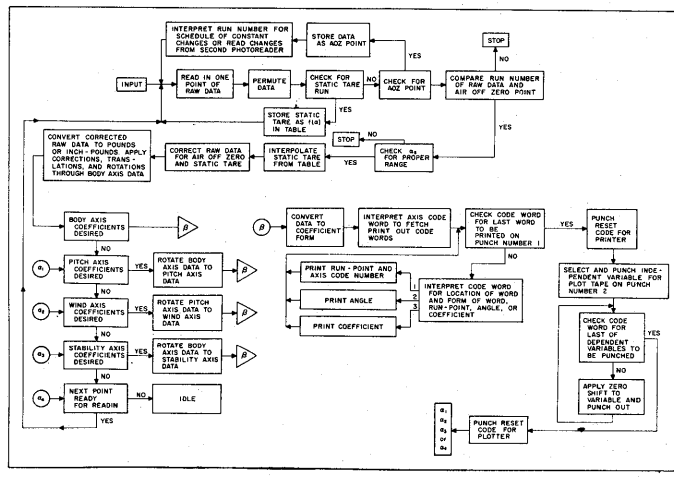


Fig. 2—Flow diagram of reduction program operations.

In the actual data presentation process the system will always lag tunnel output by one or two points; however, model pitch-down time incurred during the course of each run will allow the system to "catch up" by the time the run is actually completed. The times listed are basic times; varying requirements of test procedures may necessitate compromise between the amount and rate at which data are taken and final results presented.

Conclusions

The system described for the real-time presentation of reduced wind-tunnel data at JPL provides the broad flexibility required by the varied test programs conducted. The system utilizes components now in operation with the present data-processing system. Much of the necessary computer programming has been accomplished and the over-all system design has been completed.

The system will provide a real-time tabulation and plot of reduced data for most standard tests and will have the ability to provide reduced data with a time delay of one run for most of the possible variations on the standard reduction.

The Mechanization of Letter Mail Sorting

I. ROTKIN†

WHY mechanize the storing of letter mail? After all, billions of letters are mailed and delivered every month for just a few cents each.

The answer lies in the rapid increase of the volume of mail. Fig. 1 shows how rapidly this volume is increasing. Note that the ordinate is logarithmic and the curve is almost a straight line. Fig. 2 shows that the mail volume is increasing even more rapidly than the population. From 6 pieces of mail per person per year in 1847, it has climbed to 350 pieces of mail per person as of 1955, and we can expect this number to be doubled by 1980. About 150,000 people are involved already in the sorting of letter mail. At the present rate of increase, it will soon be difficult to find enough suitable people in the country to sort mail manually.

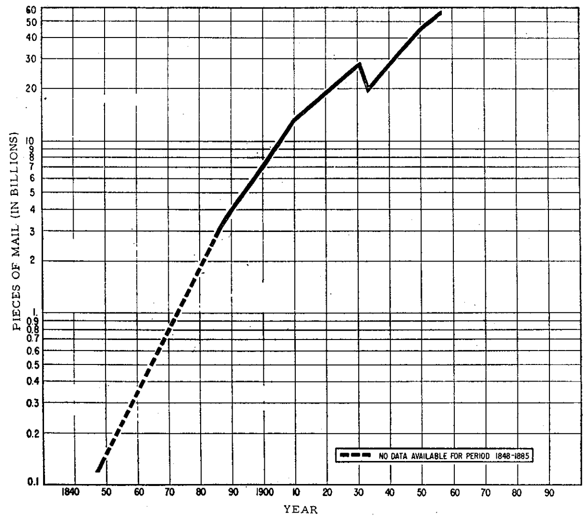


Fig. 1—Mail volume growth per year.

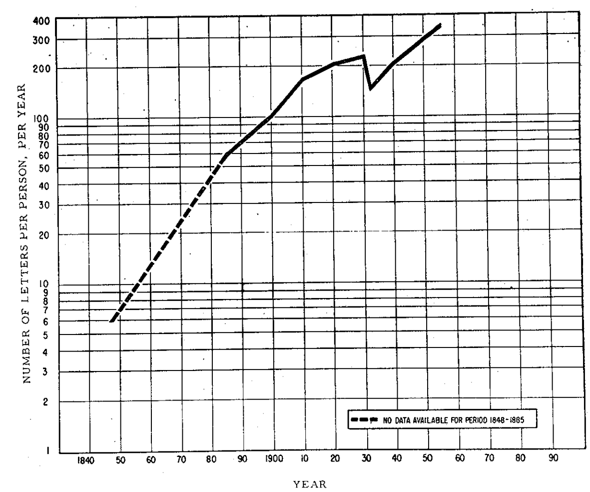


Fig. 2—Ratio of mail volume to population.

Thus the Post Office Department is forced to mechanize, simply to be able to accommodate the exponentially increasing volume of mail. However, it also can expect other advantages, such as speed, accuracy, reliability, and economy of operation, and all of this with no increase in personnel. The personnel now employed in the sorting of letter mail need not fear layoffs, because it is the publicly expressed policy of the Department that no one will be laid off as the result of the adoption of sorting equipment.

So much for policy matters. Now to get down to engineering.

At the National Bureau of Standards Post Office Project, we started with these basic assumptions:

1) The manual system of sorting letter mail has been in use and under study for many years. It is very unlikely that it can be improved appreciably. Therefore, any major improvement must be the result of introducing mechanization.
2) The U. S. postal system is too large to mechanize all at once. Besides, this would not be a prudent way to proceed even if it were possible. Therefore, mechanization must be introduced progressively.
3) At least initially, the mechanization of a post office should not affect the nature of its output or input. If a post office sorts outgoing mail to 2500 destinations and incoming mail to 600 letter carrier routes before mechanization, it should continue to do so after. This insures that the mechanization of one post office will not force reorganization of the whole postal system.
4) We cannot hope to do more than help the Post Office Department get started in this field of mechanizing letter-mail sorting. That Department will introduce many improvements as a result of operating experience. Hence we must not delay the mechanization of post offices while we strive for perfection in equipment or systems. Rather, we must freeze designs and procedures as soon as practicable to speed the day of actual operation.
5) The greatest improvement potential lies in those operations that take the most man hours per letter. This leads to the conclusion that manual letter-mail sorting is the operation most deserving of attention, because each letter is individually sorted three to eleven or more times. The average must be about six, including the last sort in which the postman arranges the letters in the order of his walk.

Having established this "axiomatic" base, we started our study in three directions simultaneously:

† National Bureau of Standards, Washington, D.C.

1) An examination of the physical characteristics of letter mail,
2) A study of the mail flow through present sorting offices, and
3) A study of existing equipment for sorting mail.

Here only a brief summary of the third is given.

There are systems in use in other countries in which the human operator reads the mail, interprets the address, and indicates the proper destination in the output of the machine to which each envelope should go. But a machine helps him do the purely mechanical work by bringing letters to him and taking letters from him, each to its designated output bin. Such a machine makes it possible for a human being to sort letters about twice as fast on the average as he can do manually and to sort, not to a maximum of fewer than a hundred output destinations, but to three or four hundred. Such machines have been in use in Holland since about 1935, in Belgium since about 1950, and in England since the Spring of 1956.

However, these machines would not eliminate multiple sorting of mail for large post offices in the U. S., because these post offices sort to more than three or four hundred destinations. Moreover, the operators must be of even higher caliber mentally than manual sorters, since they sort to three times as many destinations at double speed. This is a serious drawback, because it limits severely the fraction of the population capable of doing this work. Finally, these machines are not designed to take advantage of any future standardization of the mail. It was therefore concluded by the National Bureau of Standards personnel that these machines would not constitute a final solution to the mechanization of letter-mail sorting in the United States, although they may find their place in the smaller post offices of this country just as they have in foreign countries.

A more promising system is the one that is being developed in Canada under the leadership of Dr. M. M. Levy. Dr. Levy has proposed that each letter be standardized by having its address converted into a dot code by the first human being who reads the address in the post office. In this way, subsequent readings of that address can be carried out by machines, and no further manual reading of addresses is required until the postman is about to deliver the letter to its final destination. Thus, the number of human readings for sorting is reduced from an average of about six to one. In manual sorting today there are about six sorts on the average, including that of the postman ordering his route. Now although the conversion of the address into the dot code may take longer than the ordinary sorting of a letter manually, it does not take six times as long; the difference represents the savings in manpower achieved by the adoption of this system. The exact saving is not known as of this time, although we hope to have such information in a comparatively short time.

There is a further potential advantage in this type of standardization. A large fraction of the mail has its origin in business houses or firms which specialize in direct mail advertising. Such firms may be induced to imprint the dot code on the envelope themselves. In those cases, the Post Office Department would not have to use any human readers. We have also urged the Post Office Department and firms in the business-machine field to adopt a standardized type font based on a 5×7 mosaic for machine reading. The use of such a font in a standard size for addressing business mail would also eliminate the need for human readers.

Having established our axiomatic base and studied what was available from others, we proceeded to develop our own system using the Canadian idea. We did not adopt the Canadian equipment, because it was not suitable for our post offices. All history is omitted here, and we shall describe our system as we visualize it today.

Fig. 3 is an artist's conception of one section of a mechanized post office. In the foreground are the code-printing stations. Mail that has been cul.ed, faced, and cancelled is brought to these stations. Human operators read the addresses and operate keyboards to rewrite the addresses in a standardized, abbreviated form. These standardized addresses are printed on the back of the envelopes in a dot code not very different from Teletype code. The printing is done with phosphorescent ink to enhance contrast during subsequent mechanical reading.

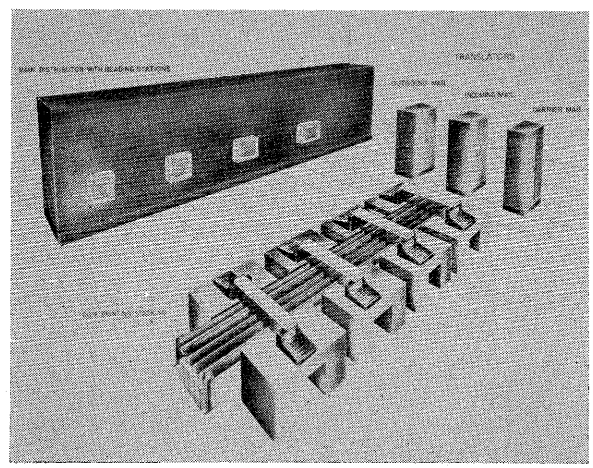


Fig. 3—Artist's conception of one section of a mechanized post office.

Provision is made at this stage for a rough sort of the mail by classes. It may prove operationally advantageous not to code immediately all of the mail that reaches these stations, and therefore provision is made to put some of the mail aside into several categories. For example, since in most large post offices between a third and half of the mail is local, it may be operationally advantageous, in order to speed up the outgoing mail, to put the local mail aside for later detailed address coding. Similarly, it may be advantageous to separate out mail requiring special handling, such as air mail, special delivery, and registered mail.

Also, provision must be made to eliminate from further operations defective mail, *e.g.*, mail which has insufficient postage or an incomplete or improper address.

One of the classes will consist of the mail that has been coded for immediate sorting. This mail is transported bulk-wise to the boxes shown mounted on the large machine in the rear. Each of these boxes represents an automatic code-reader input for the large machine which is the distributor. In each code reader, the dot code on the envelope is illuminated by ultraviolet light and then read in darkness by photocells. The resulting electrical address signals are then fed to one of the three translators represented by vertical rectangles on the right side of the illustration. One of these operates for outgoing mail, another for incoming mail, and the third operates when the mail for each carrier is arranged in the normal order of his walk. The appropriate translator accepts the address signals, and translates them (in terms of its stored scheme of distribution) into electrical signals representing the proper output bin on the distributor. These output signals are sent to the distributor in synchronism with the envelope itself. The distributor then delivers the envelope to the designated output bin.

This, in very broad outline, is the system of mail sorting that the National Bureau of Standards now has under development. What are the problems associated with this system?

The chief problem, of course, is to develop criteria, methods, and data for judging the relative merits of various sorting schemes. To this end, we have investigated the sorting operations of several post offices such as Washington, Baltimore, Philadelphia, Los Angeles, and San Francisco. We may add Chicago and St. Louis to this list later. We study the relative volumes and physical characteristics of the mail from all sources in each of these cities. We study the mail flow pattern through each post office, and we determine the destinations to which it must sort. There is much too much mail in each post office for us to conduct a complete piece count, so we have developed statistical sampling procedures. These data enable us to determine the system parameters for any mechanized scheme of doing the work of each of these post offices. Our comparisons are thus real instead of conjectural.

In addition to the over-all systems problem, there are the many equipment and procedures problems. Some of these will be examined in the order in which they would arise in following an envelope through a mechanized sorting operation. Thus we start with the code printer.

First, what kind of a code shall we use? It must require as little operator time as possible, be easy for operators to learn and to remember, and result in a high speed of operation with a very small error rate. An early report on code development was made to the Post Office Department.

Second, what type of keyboard shall the operator use? We expect to use ordinary typewriter keyboards, initially at least. But we are not sure that these are best; and tests are being run to see whether the direct digital keyboard used by the Canadians or a rather elaborate multichoice keyboard proposed by the Dutch may not be superior. These are human engineering problems and in order to answer them, we have been forced to run tests with human beings. The same test series is being used to determine the human engineering aspects of the codes. These tests are expensive. They take a long time to run. They must be very carefully designed, and the results must be very carefully interpreted so as not to be misleading.

Third, a more straightforward engineering problem, in what form shall the abbreviated address appear on the envelope? Our choice has been to use phosphorescent dots in a code similar to Teletype tape. We arrived at this choice for three reasons:

1) Only optical sensing equipment can be relatively distant from the surface on which the pattern is imprinted and still retain sharp focus;
2) Only in an optical pickup can sensitivity be independent of the distance between the pickup device and the surface from which it is reading; and
3) The use of phosphorescent ink allows us almost complete immunity from anything that may be printed on the envelope in the way of extraneous matter such as advertising, and it gives us a very good signal-to-noise ratio. Phosphorescent dyes are not used today in envelopes nor in the inks used on envelopes. Furthermore, we have proposed that such use be forbidden in the future. There is really no reason for anyone to want to use them, so that this prohibition should not work any hardship on anybody.

Next we come to the translator. It is technically possible to develop an electronic device which is so fast that it can translate address signals into output signals for a whole post office by time-sharing its services to several distributors. The alternative is to develop a device cheap enough to be used with a single distributor, but required to make translations at a correspondingly slower rate. Which shall we use?

One contractor has developed laboratory versions of both kinds of equipment. We believe the future belongs to the more potent equipment, because it will be more compact and economical. However, for the present, we are perfecting the more modest version; so that, in the event of breakdown, only one translator is made idle instead of a whole post office.

The translator under intensive development is somewhat similar, but by no means identical, to the punched metal-card device used by the Bell Telephone System in its long-distance automatic switching. It will be small, cheap, and fast enough. The author hopes that J. Rabinow, the inventor, will prepare a detailed history of the development of this device. It would make a fascinating story.

Finally we come to the distributor. First, there is the question of its general design. There are two ways of dedesigning such equipment. One way is used by the Belgians and the British in their designs. We shall call this an external control system. It makes use of a main distributor frame, which is essentially a mechanical device for pushing envelopes, but which carries no intelligence whatever. The instructions to this device, as to when and how it shall divert envelopes into the proper output bins, come from separate control equipment which must run in strict synchronism with the main distributor. Such devices have been built which work satisfactorily. However, in order to insure strict synchronism between the distributor proper and its control, the construction must be very precise and stable, and the flexibility of layout is limited. We have chosen, therefore, to adopt a modification of another system employed by the Dutch and others. This we call the self-control system. In this, each envelope has associated with it, a mechanized form of the output bin address as it moves through the machine. Each output bin also has a mechanized address associated with it. The two work together like key and lock. There is no auxiliary control unit. Thus, it makes no difference how far this envelope is carried, nor in what direction, because its instructions travel with it.

Another question that arises in connection with the distributor is: For how many output destinations should it be designed? From the point of view of the postmaster who has a job to do, it is desirable to design this equipment to handle the same destinations that he now sorts to manually. This means that the introduction of the equipment would not force any changes in the mail transportation system. So far as other post offices are concerned, they would not know any difference after the equipment was installed. However, this implies, for a place like Chicago, for example, sorting to about 5000 destinations. It may not be feasible, for cost or space reasons, to build so large a distributor.

These problems have been described very briefly and in rather general terms. We are attacking them much more specifically, using, mathematical models, computer simulation, statistical studies, and engineering trials. It is hoped that each of these problems will be the subject of a separate paper at a later date by some member of the project staff. These papers will be prepared for internal use of the project and the Post Office Department. If they prove to be of interest to others, we are sure the Post Office Department will permit distribution.

The National Bureau of Standards Post Office Project staff is not doing all these things by itself. We are getting invaluable help from the Post Office Department in gathering and interpreting postal data. Most of the very ingenious engineering features of the sorting equipment are due to the Rabinow Engineering Company, our sorting equipment contractor. Prof. Harry H. Goode of the University of Michigan, who is chairman of the Technical Program Committee of this Conference, has been our invited critic, and a very conscientious and useful one. It is he who has encouraged us to make earlier and greater use of mathematical models. Many others have also helped in other ways.

This has been a very quick review of the work of the Post Office Project at the National Bureau of Standards. It has run very lightly over a large and complex field. We hope that, in time, ways will be found to cover the ground more thoroughly for the benefit of those who may be interested in greater detail.

Discussion

Question: Do you care to say anything about mechanical facing of mail incidental to mechanical mail sorting?

Mr. Rotkin: Since this is not a mail handling specialist group, I had better explain that facing means arranging the envelopes so they all face the same way. It is done at present for the convenience of the canceling machine as well as that of the sorter. This is not part of the work that the National Bureau of Standards is doing. We pick up the mail after it has been faced and canceled. It is perfectly feasible from a technical point to do this mechanically and the Post Office Department is working on it in conjunction with at least two contractors.

Question: What type of distributor is visualized for the final route sorting of the carrier mail?

Mr. Rotkin: The same kind that would be used for distributing outgoing and incoming mail. It is our objective to have one distributor do the work of all three kinds of sorting, either by time sharing or by just dividing the distributor into sections appropriate to each sort.

It is more likely to be by time sharing because of the way the work comes into the post office. It is quite feasible to do this. The number of sorts involved is comparable. For example, in Washington, we sort to about 2500 or 2600 outgoing destinations. We sort to about 500 or 600 carrier routes and each carrier makes several hundred stops. I don't know the exact number but it is in the neighborhood of 500.

In Chicago, for example, they sort to about 5000 or 6000 outgoing sorts; for incoming mail, they sort to about 5000 or 6000 carrier routes and each carrier also has somewhere in the neighborhood of 500 stops.

Therefore, the same equipment can be used for all of the sorts and the only difference required would be in the kind of instructions the machine gets from what I call a translator.

Question: Will it be necessary to standardize letter sizes to attain optimized mechanization?

Mr. Rotkin: It is not necessary; it certainly would be helpful. We are trying at present to determine what the largest letter will be that we will accept into this mechanization sorting system. This is one of the reasons for measuring physical characteristics of the mail. Letters come in all sizes from very, very small to quite gigantic things called flats. It obviously does not pay to design a machine that will handle anything as large as flats because it doesn't occur in a high enough percenetage of the cases.

We don't know whether this will be the proper cutoff point, but tentatively, we are working on the assumption that $6 \times 12 \times 3/8$ is big enough. I expect that after we have done our statistical work, we will find out this is too big, and then we can afford to make our cutoff for a smaller size of envelope.

Question: Is character reading a part of the system?

Mr. Rotkin: It is not a necessary part. If we could have good character reading, it would be helpful. This is why we would like to have people standardize the style and size and format for addresses for business purposes. It would make the problem of character reading almost child's play compared to what it is today with the wide variety of sizes, type fonts, and general arrangements of addresses.

Preparations for Tracking Artificial Earth-Satellites at the Vanguard Computing Center

D. A. QUARLES, JR.†

Introduction

It is considered appropriate to remark at the outset that part of the computer programming system subsequently to be described has been used extensively in performance of tracking calculations for artificial earth-satellites. This operational use commenced with the tracking at the Vanguard Computing Center, Washington, D.C., of the first Russian satellite last October 5, the day after the satellite was launched, and has continued during the interim.

To perform with great speed the calculations which would be required in the tracking of any artificial earth-satellites launched during the period of the International Geophysical Year ending December 31, 1958, an IBM Type 704 Electronic Data-Processing Machine was installed at the Vanguard Computing Center. The purpose of this paper is to describe the flow of operations and calculations performed using the programming system developed to handle these tracking calculations on this stored-program 704 computer.

Various stages involved in the data processing will be discussed; stages commencing with receipt of raw observational data and extending to provision of computed orbital information specifying the predicted future motion of a satellite, and extending eventually to provision of a comprehensive history of its past motion. Deadlines pertain to processing observational data and to distributing predicted positional information.

It is emphasized that the only calculations considered in this paper pertain to satellite tracking and orbit determination. It is the preparation for such calculations only—not for calculations pertaining to satellite or launching-vehicle structural design, not to launching-vehicle trajectories, and not to studies of the earth and its atmosphere—that the Vanguard Computing Center has been responsible. Furthermore, the Vanguard Computing Center, which is owned, staffed, and operated by the IBM Corporation under a contract with the Navy, is largely concerned with problems involved in processing positional information obtained from a satellite's continuous radio transmission. Problems involved in acquisition and processing of optically or visually obtained observations of a satellite, enabling continued use of new observational data for orbit determination if the satellite outlasts the life of the batteries which power its radio transmitter, are primarily the concern of another group, also using a 704 computer, and located in Cambridge, Mass. Naturally, however, both the Cambridge and Washington groups are interested in using observations obtained by both radio transmission and optical or visual methods.

The responsibility for establishing the orbit computation procedures rests with the working group on orbits which includes: Dr. J. W. Siry and J. J. Fleming of the Naval Research Laboratory; Dr. Paul Herget, Director of the Cincinnati Observatory; Dr. G. M. Clemence and Dr. R. L. Duncombe of the Naval Observatory. The detailed mathematical formulations were developed chiefly by Dr. Herget with assistance from Dr. Peter Musen, formerly of the Cincinnati Observatory.

The author of this paper has had responsibility, together with those working under him in New York, N.Y., for planning and preparation (synonymously "programming") of the system of instructions used by the 704 computer for accomplishing the desired orbital calculations, and for the programming of certain special calculations. These special calculations were performed during the development of the mathematical formulation, and guided the course of this development.

IBM staff members of the Vanguard Computing Center, assisted by the New York system development group, will handle the operation of the programming system on the 704 computer. The operation of the system is highly automatic, with many minor decisions being made by the computer according to rules provided to it. Nevertheless, from the processing of raw observations, to the computation of orbital characteristics and predicted positional information, the operation permits certain major alternative techniques to be used at various intermediate stages of the calculations. Decisions regarding some of these major alternatives can be supplied in advance to the computer, enabling its subsequent automatic handling of the desired choices of these alternatives. However, if these decisions are not made in advance or if it is desired to alter any of these decisions during the course of the calculations, the programming system is designed to make this conveniently possible by manual intervention. On hand to assist in any such decisions and to interpret the calculated results will be Dr. Herget and others responsible for the formulation.

General Description of Programming System

A very general description of the structure of the programming system will now be given before discussion of mathematical techniques employed and of their flow of operation. From the outset of the planning, it became apparent that the system should be designed to permit convenient choice not only in the methods of computation

† IBM Corp., New York, N.Y.

used, but also in the order of their use. To enable great flexibility in these respects, it was decided to make the system card-controlled. That is to say, the choice and arrangement of control cards in an input deck would, barring manual intervention, govern the choice and ordering of any operations subsequently performed by the computer. The terminology "macro-operation" was adopted to denote a collection of subroutines linked together to perform a broad orbit-computation function.

The component parts of a macro-operation were frequently needed in other macro-operations. Any such commonly-needed component part was then usually written as a subroutine with sufficiently general specifications to enable its use in the various macro-operations in which this type of operation was required. This frequently-adopted practice of programming by subroutines, rather than directly by macro-operations, had several significant advantages. Not only would this practice eliminate much duplication of programming effort which would otherwise have occurred, but also it would tend to keep localized, perhaps to a single subroutine, effects of a minor change in formulation. It may be remarked that for a research endeavor in a new field, such as the establishment of the formulation for tracking artificial earth-satellites, some changes in the formulation should not be unexpected.

It was decided to assign one auxiliary storage magnetic tape of the 704 computer to serve as a system tape, each block of information on this tape comprising instructions, constants, and control information required by just one macro-operation. Thus each control card (the contents of which are fixed) possibly followed by one or more input-data cards (the contents of each of which are variable) correspond to one macro-operation. In general, when a macro-operation control card is read at the card reader, a system subroutine retained in high-speed magnetic core storage causes (with a minimum of tape motion) the appropriate macro-operation information to be read from the tape into high-speed storage immediately prior to its use. In general, the complete set of instructions required for each macro-operation consists not only of those instructions transferred for the macro-operation from magnetic tape to high-speed storage, but also of utility subroutines (*e.g.*, input-output subroutines) commonly needed by various macro-operations, and retained in the high-speed storage. The auxiliary magnetic drum storage is used to store certain input parameters which are subject to change, and also output of certain macro-operations which may serve as input for one or more subsequent macro-operations. Comparatively small amounts of output are directly printed and/or punched on cards. Larger amounts of output are written in binary-coded-decimal form on magnetic tape for subsequent printing on a magnetic tape-to-printer peripheral device not connected to the 704 computer, and operated independently of it. In particular, one such tape, called a "log tape," was assigned to preserve a detailed chronological history of calculations performed, including not only input and output quantities but also many intermediate results. The system is designed to enable the observational data, subsequent to its preparation in decimal punched-card form, to be supplied as input to the computer, optionally from these punched cards directly, or from binary-coded-decimal tape prepared on a card-to-tape peripheral device. To check for occurrence of random machine error, the internal calculations are generally performed in duplicate with comparison of check sums, while transference of information between high-speed magnetic core storage and auxiliary magnetic tape or magnetic drum storage is checked by check sums, and special checks are made for input-output operations. In addition to the printer, magnetic tapes, and card punch, the use of which has already been indicated, the cathode-ray tube display and cathode-ray tube recorder are available as output devices. Programming has also been done to enable optional use of these devices to provide plotted output in direct-visual and/or filmed form.

Flow of Operations and Computational Methods

The flow of operations performed in processing the data will now be discussed, together with brief descriptions of the computational methods involved. As a starting point, suppose that some of the continuous radio transmission from a satellite has just been received by one of the specially-designed receiving stations, called "Minitrack stations," during a single transit of the satellite over the station. This transmission is subdivided at the Minitrack station into individual observations, up to about thirty in number and evenly spaced in time. Each observation then consists of phase-difference readings, one for the east-west-oriented radio antenna of the station, and one for the north-south-oriented antenna, both corresponding, after small adjustments, to a single instant of time. A typical time interval for these observations is one second. The associated information, comprising the phase-difference readings, first and last times for the readings, and certain information identifying and specifying characteristics of the observing station, will for convenience be called a "message." Each message is transmitted in triplicate to a control center at the Naval Research Laboratory, and thence, still in triplicate form, via teletype to the Vanguard Computing Center. A device at the Vanguard Computing Center automatically converts the teletype tape to decimal punched cards.

This message is then ready to be processed by one of the macro-operations of the programming system for the 704 computer. This macro-operation, which is always the first macro-operation to process any of the messages, includes four principal subroutines. The first subroutine is designed to load the message into the high-speed storage of the computer, transforming information from compact form on the input-data cards (or, optionally, input binary-coded-decimal tape) into a more suitable form for its subsequent use. The second subroutine performs an editing function, comparing the triplicate items of the message for exact agreement. This second subroutine reduces the size of the message if permissible when, for a given item, at least two out of three of the corresponding items do not agree exactly. The third subroutine performs several adjustments to the data due to certain characteristics peculiar to the Minitrack station which received the data, due to passage of time during the recording of a single observation, due to radio refraction, and in order to convert the phase readings to directional information. Then, each adjusted and converted observation provides the approximate direction of the satellite from the observing station at a certain instant of time. It is to be noted that no direct measurement of the distance of the satellite from the observing station is yet available. In fact, satellite distances will be derived as output of orbital calculations. Only after such calculations, shortly to be discussed more specifically, does such a distance serve as an input quantity. The processing, thus far, also checks to assure that at least three individual observations remain so that the fourth subroutine may have reasonable assurance of being able to perform its function. This function is to fit a least-squares parabola to each of the two sets of direction components which were derived collectively from east-west and north-south phase-difference readings. The principal output of this macro-operation, then obtainable, consists of a single "smoothed" direction of the satellite from the observing station, expressed in a local coordinate system and corresponding to an instant of time. This instant of time is centrally located with respect to the time range of the set of raw observations which, it is recalled, were obtained from a single Minitrack station during one transit of the satellite over this station. This principal output is provided in both punched-card and printed form. Certain subordinate output quantities, such as any discrepancies between the triplicate messages and standard errors of the least-squares parabolic smoothing operations, also are printed. Also, as in the operation of all macro-operations, a detailed record of input, intermediate output, and output is preserved on the log tape. In the subsequent discussion of the flow of calculations, it will be understood that preservation of information on a log tape, and printing of output information which is required for more rapid surveillance, are included in all macro-operations, whether or not explicitly mentioned (see Fig. 1).

Several methods have been programmed in order to enable computation of a preliminary orbit from such observations. (Here, and subsequently, the qualification "parabolically smoothed" is understood when referring to an observation.) One of these methods makes use of two observations which, roughly speaking, are suitably widely spaced in time, to obtain a preliminary circular orbit. Other available methods develop a preliminary elliptic orbit from three or four observations which are neither too closely nor too widely spaced in time. If the observations were too closely spaced in time, inaccuracies in the observations could seriously reduce the accuracy of the result, while the particular methods used would be invalidated in the case of observations spaced too widely.

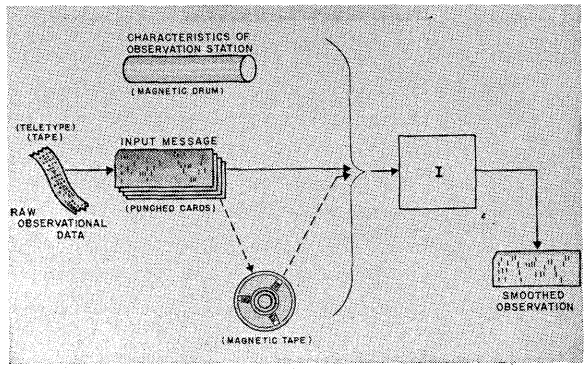


Fig. 1—Processing of Minitrack (or Minitrack-simulated) input. Macro-operation I.

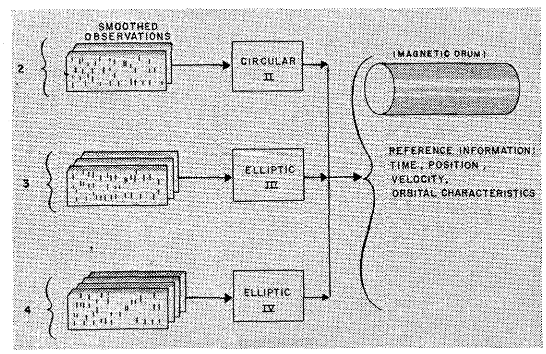


Fig. 2—Preliminary orbit determination. Macro-operations II, III, IV.

These methods are based on iterative techniques due to Gauss, and are applied after transformation of coordinates of observations, expressed in their original local systems, to an inertial coordinate system. In each case, a reference position vector, and a corresponding velocity vector, are obtained for the satellite at some instant of time. Then, from this reference time and corresponding position and velocity vectors for the satellite, the programming provides for computation of certain quantities serving to characterize the orbit in question in different ways. For example, such computed quantities are the period of the satellite's revolution, the inclination of its orbit plane, and, in the case of the elliptic orbits, also such quantities as the semi-major and semiminor axes of the orbit, and the perigee (*i.e.*, closest to earth) position of the satellite. The programming for each of these preliminary orbit computation methods constitutes a separate macro-operation (see Fig. 2).

After having obtained a preliminary approximate orbit, other methods have been programmed to enable its improvement and, as the orbital characteristics would constantly be subject to change, also its updating. One macro-operation consists primarily of a procedure for numerical integration of the differential equations relating, by Newton's law, the components of forces acting upon the satellite, to the components of its acceleration. Each of the three force components contains an expression for the force on the satellite due to gravitational attraction of the nonspherical earth, and an additive term for force due to an admittedly fairly rough estimate of atmospheric-drag force acting upon the satellite. As is well known, one of the reasons for launching artificial earth-satellites is the desire to gain further information about the structure of the upper atmosphere. Until more is learned about the upper atmosphere, in particular about variation of atmospheric density with altitude, the components of drag force in these differential equations may be subject to significant inaccuracies. The other contribution to the force components, due to gravitational attraction of the earth, may also introduce significant error because of our inadequate knowledge of local variations in the earth's gravitational field. There are, of course, further errors in the numerical integration procedure caused by inaccuracy in initial values of the satellite's position and velocity, due to replacement of derivatives by finite differences, and due to growth of error during the numerical integration computations. Though its accuracy is limited by such errors, this numerical integration macro-operation is expected to be very useful, not only because its output consists of predicted positions of a satellite spaced in time by an arbitrarily-chosen time interval, but useful also, in combination with a method of differential correction, for orbital improvement and updating (see Fig. 3).

The macro-operation for this differential correction method obtains corrections to position and velocity vectors corresponding to a reference time which may be periodically updated. These corrected vectors may be used, as before, to obtain new orbital characteristics. The input for this differential correction procedure consists of observations (preferably including, because of changing orbital characteristics, the latest available) and predictions, at the same observational times, obtained from output of numerical integration by 6-point Lagrangian interpolation. So-called equations of condition are computed, one set for each observation, after making an improved adjustment to the observations for refraction. The differential corrections are then readily obtained from least-squares solution of these equations of condition. The processes of prediction by numerical integration and differential correction may be performed iteratively in attempt to bring predictions and observations in close agreement (see Fig. 4).

A more complicated alternative technique, which is also planned to be used for prediction and orbital adjustment, is based upon three further macro-operations. In brief, one of these macro-operations uses a modification of Hansen's lunar theory to compute Fourier series representations of orbital characteristics, including perturbations due to the oblateness of the earth, in terms of a variable representing time. A second macro-operation treats separately drag perturbations in these orbital characteristics by numerical integration. By evaluations of these Fourier series and additive perturbations, adjusted orbital characteristics and, thence, derived predicted positions, may be

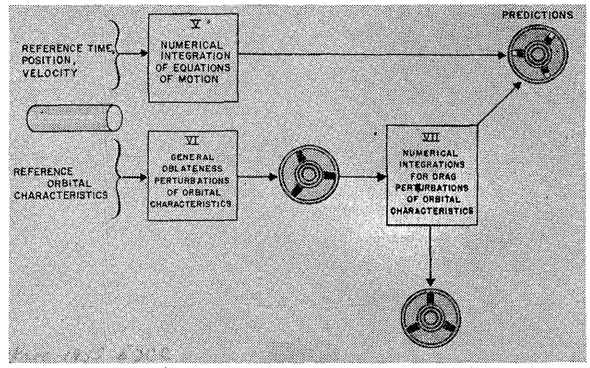


Fig. 3—Predicted positions (inertial vectors). Macro-operations V, VI, VII.

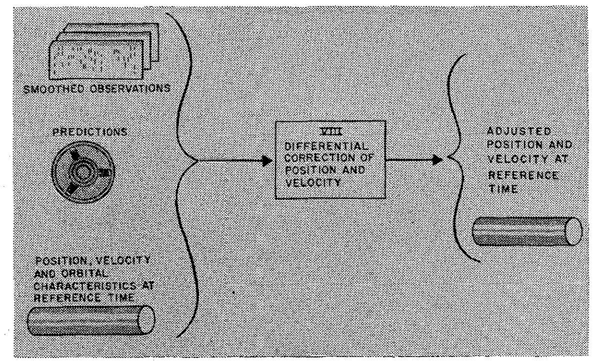


Fig. 4—Orbital adjustment (improvement and/or updating). Macro-operation VIII.

obtained. The third macro-operation computes differential corrections to the orbital characteristics by a technique similar to that used for correcting the reference position and velocity vectors. This technique for prediction and orbital adjustment, using these macro-operations iteratively, is expected to be a valuable alternative until the later stages of a satellite's "life." Near the end of the flight of a satellite, the drag force would become large enough to invalidate the method used for the oblateness perturbations.

At such time, it is planned to continue predicting and orbital adjusting by the first-described techniques for numerical integration and differential correction (see Figs. 3 and 5).

Two separate macro-operations are available for transforming predictions from the inertial-vector form into forms more convenient for use by the general public. One of these macro-operations provides as output, for each prediction of a specified time span, the time, latitude, and longitude for the subsatellite position, height, and zenith-angle-acquisition information. Also provided as output is a list of any official Minitrack or optical stations to be alerted due to the satellite's proximity, but, in the case of an optical station, only if the favorable observation condition of twilight exists. The other of these macro-operations provides, as output, positional information for a specified set of times and relative to a specified station (see Fig. 6).

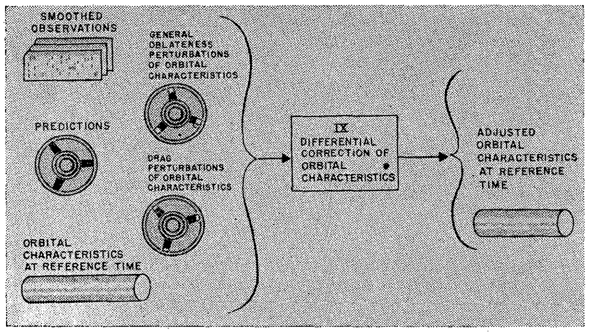


Fig. 5—Orbital adjustment (improvement and/or updating). Macro-operation IX.

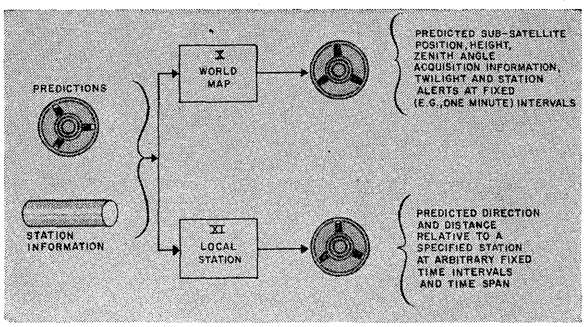


Fig. 6—Predictions (transformed for convenient use). Macro-operations X, XI.

Conclusion

Several remarks seem appropriate at this point. It was earlier stated that certain deadlines pertain to processing observational data and to distributing predicted positional information. Satellites, of the types presently being considered, would complete a single revolution around the earth in approximately an hour and a half. Allowing for communication and distribution delays in incoming data and outgoing predictions, the speed of a satellite's motion requires very rapid calculations to be performed in order to enable use of the methods described in providing, sufficiently in advance, alerts and predicted positional information to observers around the world. A machine with significantly slower speed or significantly smaller storage than the 704 computer would not have been able to do the same job adequately. In particular, in the very early stages of a satellite's flight, the Vanguard Computing Center will be concerned with developing and distributing orbital information within a matter of minutes of the receipt of observations which first enable preliminary orbit determination. However, until at least one revolution of a satellite has occurred, observations may be so sparse that better than a rather inaccurate determination of the satellite's actual motion would be prevented.

The magnitude of the programming system's development, and of associated programming for special calculations, may be measured by an estimated expenditure of between 6 and 7 man-years of work involving the writing of approximately 25,000 instructions. The size of the programming system is a result of a complex mathematical formulation whose programming included many different types of Fourier series manipulations, and a result of the system's flexibility in enabling convenient use of alternative computing techniques. To guard against possibility of machine breakdown at an inopportune time during a satellite's flight, another IBM center with a 704 computer will be kept prepared with operational information by means of a transceiver and telephone on an emergency stand-by basis.

Acknowledgment

The author wishes to express his appreciation to those whose efforts and assistance have made possible the development of the programming system. These persons include Dr. G. E. Collins and R. T. Mertz, who have been associated with this project from the beginning of this development, and also Miss L. Y. Chang, Mrs. N. G. Copeland, Israel Krongold, A. R. Mowlem, J. B. Secrist, Jr., Dr. R. W. Southworth, and N. R. Wagner.

Bibliography

[1] Herget, P. "The Computation of Orbits." Published privately, 1948.

[2] Herget, P. and Musen, P. "General Theory of Oblateness Perturbations for Vanguard Satellites" (memorandum), March 23, 1957. Presented by Dr. Herget in abstract form at the American Mathematical Society Symposium on Orbit Theory, April 4, 1957.

Discussion

It should be noted that these answers were taken from the latest technical information available through March 26, 1958.

Chairman M. Rubinoff (Philco Corp., Philadelphia, Pa.): Is a triangulation system used or planned to be used to increase accuracy of observation? This question is by E. A. Keller.

Mr. Quarles: In attempting to answer this question, I shall restrict my attention to the standard Minitrack receiving units being used by the Navy in performance of radio-tracking responsibilities. The disposition of these stations was aimed at providing a very high probability of receipt of transmission from a United States satellite by one station during each earth-circuit of the satellite. At originally expected altitudes for such a satellite, this disposition provided an expectation that simultaneous observations by two or more stations would be exceptional rather than usual. Consequently, there was an expectation of being able to triangulate simultaneous observations from these stations only comparatively rarely. The occurrence of higher than originally expected altitudes has made somewhat less exceptional the recording, and expectation for future, simultaneous observations. It is planned to investigate accuracy by triangulation of any such simultaneous observations, possibly leading to increased accuracy of observation. However, on the basis of these considerations, triangulation does not occupy a very significant role in the present operational use of the Minitrack system.

The method of obtaining a directional observation inherent in the Minitrack recording involves what might be termed "triangulation" from displaced antennas, based upon phase difference. I assumed that the intent of the question was to gain information about "triangulation" in the usual sense of this word.

Chairman Rubinoff: L. Elrod of Westinghouse asks, "will it be possible to modify the program while actual tracking is in process?"

Mr. Quarles: It is expected that modifications to the program will probably only be made after studying the results obtained and would not actually be made during the course of one continuous operation on the machine. However, it is possible to exercise a number of programmed options by manual positioning of sense switches, and to modify the arrangement of macro-operation control cards and content of data information to be used at a later stage, during calculation.

Chairman Rubinoff: The third question is, "Is there any provision in the subject program for using observational data from sources other than official tracking stations such as amateur optical or radio measurements?"

Mr. Quarles: Let me emphasize first that it is not required that the observational data be from the Minitrack stations. On the first slide it was noted that the data could be either Minitrack data or simulated Minitrack data.

To go one step further, it wouldn't even be necessary to enter the data in simulated Minitrack form if one had sufficiently reliable data obtained by whatever means. For example, one could bypass the whole operation of the first macro-operation which edits and smoothes the data, and simply enter as a smoothed observation the direction of a satellite from an observer at a certain time, if one had such information. So, it is quite flexible as to the types of data that could be used and in fact several kinds of data have been used. However, the qualification "sufficiently reliable" stated above makes virtually necessary what might be termed professional electronic or optical equipment and professional operating personnel, and seriously limits the usefulness of amateur measurements. Of course, in the absence of professionally obtained data, amateur measurements take on increased importance.

Chairman Rubinoff: A question from R. Isaacs of Philco, "What reports, if any, of the programming system are available?"

Mr. Quarles: The first actually published report of the programming system will be this paper as presented, supplemented in publication by questions and answers. Later, I expect there will be more detailed reports of the programming system available.

Chairman Rubinoff: A question by Mr. Sumpter of the Department of Defense, "Why use cards for input to the computer? Why not go directly from teletype tape to magnetic tape, then into the computer?"

Mr. Quarles: As described in my paper, there is an option of providing input observational data to the computer either directly from punched cards or from magnetic tape prepared on a device which operates independently of the computer. This option of the use of magnetic tape was provided in order to enable more rapid computer processing in the event that a large volume of input data were to be supplied to the computer at any one time. Whether or not this option is exercised, the preliminary preparation of cards has an advantage of enabling, together with the printed triplicate messages also produced from the teletype operation, convenient partial editing, selection, or rearrangement of the data prior to the more extensive editing and processing performed by the computer.

Also, there isn't presently, so far as I know, a commercially available device for converting directly from teletype tape to magnetic tape. Furthermore, there has not seemed sufficient need for such a device in this computer application to render it an important consideration. More rapid provision to the computer of this input data does not at present seem important. Using either of the options described, the operation typically requires a small amount of time for supplying the input data to the computer in comparison with times for calculation and for development of output data on magnetic tape. In particular, the maximum possible time saving due to bypassing preliminary preparation of punched cards by using a hypothetical teletype-tape-to-magnetic-tape device would not under present, or presently expected, conditions of operation seem to justify the sacrifices of conveniences of cards described above.

Chairman Rubinoff: From D. J. Nemanic of Remington Rand Univac, "Can unknown factors such as density of the atmosphere be estimated from the differential corrections to the predicted orbit?

Mr. Quarles: Yes, it certainly is possible to estimate some of these factors on the basis of the orbital calculations in general—not only from the differential corrections. One of the main purposes of the whole project is to make improvements in our knowledge about the atmosphere by examining and studying the deviations of the predictions from the actual observations.

In particular, the calculations have indicated that the density of the atmosphere is greater than originally had been expected at the altitudes attained by the artificial earth-satellites. However, much study of orbital calculations is expected to be required before reasonably reliable information about density variation at high altitudes is known.

B. Zendle (National Bureau of Standards): Has it been possible to determine the mass of the Russian earth satellites, thereby verifying the mass values announced by the Russians? If so, how, and to what degree of accuracy?

Mr. Quarles: Though I have not been concerned with this question, it is my understanding that the presently limited information about the density of the atmosphere, and consequently about the drag acting upon a satellite, would have prevented any independent, accurate verification of the masses announced by the Russians. Later, when it is possible to determine atmospheric drag with greater accuracy, such independent verifications should be possible with reasonably good accuracy from the laws of motion of a satellite which depend upon both the drag and the mass.

S. M. Selig (Chemical Corps Eng. Command, Army Chemical Center, Md.): What are the computer facilities that the Russians have set up equivalent to Minitrack?

Is the accuracy of their predictions for Sputnik I due to more sophisticated equipment than the IBM-704, or to better programming and better fitting or Fourier series and other essential computations that you mentioned?

Mr. Quarles: I do not know the answer to this question. However, on the basis of any predictions by the Russians which I have seen, I do not have reason to believe that they are using anything but comparatively unsophisticated techniques.

M. A. Hyman (Philadelphia, Pa.): Approximately how many points were calculated by numerical integration for each elliptical trajectory? What can be said about the accumulation of errors during calculation of an average trajectory? How long did the computer require for each trajectory?

Mr. Quarles: The typical time interval in the numerical integrations performed to date is one minute, and hence the order of one hundred steps per earth-circuit. A thumb rule which may be applied to estimating the accumulation of errors by the method of numerical integration which has been programmed is that the error is approximately the three halves power of the number of steps, divided by eight, units in the last place of the digital precision employed. The programming of the numerical integration enables optional use of single- or double-precision floating-point calculations, providing precision equivalent approximately to eight or sixteen significant decimal digits, respectively. Including binary-coded-decimal tape output as well as binary-tape intermediate output, the single-precision computation requires approximately six minutes per day of predictions using a one-minute time interval, whereas the double-precision computation requires about six times as long. Both of these computations are reduced by about four minutes per day of such predictions at the sacrifice of the binary-coded-decimal tape output.

W. H. Jenkins (ElectroData): You mentioned 25,000 instructions. Does this complete the program? Are these operations debugged? If so, you should have some accurate times for the processing from the input cards or tape until the final result of "where to look."

What are the limits of these times? (For example, three to five minutes, or ten to thirty minutes.)

Mr. Quarles: The figure of 25,000 instructions was intended to cover completed instructions and modifications which were in progress. It is expected that some additions and modifications to the system will be planned and developed later due to the research character of the project.

With the exception of some current comparatively minor modifications, all of the macro-operations described are debugged and all are in use. In view of the great variety of ways in which the macro-operations have been combined in actual operation, permitted by the flexibility of the design of the programming system, it is very difficult to give meaningful over-all time figures without extensive qualification. Even for individual macro-operations there usually are several modes of operation with significantly different times, as indicated by the answer to the preceding question. (See the answer to the next question.)

J. Otterman (University of Michigan, Ann Arbor, Mich.): Has consideration been given to carrying out macro-operation No. I on a separate smaller computer?

With all Minitrack stations in operation what is the percentage of times the computer will be idle (or reserve operation time)? Can the system handle data simultaneously on two satellites?

Mr. Quarles: No serious consideration has been given to carrying out macro-operation I on a separate smaller computer.

In the present manner in which the programming system is operated, 24-hour-per-day utilization of the 704 would permit simultaneous handling of the tracking calculations for from six to ten satellites. It should be possible to increase the number of satellites which could be "simultaneously" tracked by a significant amount when further experience has been obtained and/or by reducing the volume of output information.

E. H. Weiss (Applied Physics Lab., Johns Hopkins University, Baltimore, Md.): You mentioned that "other checks" are used to ascertain the accuracy of input and output. What are some of those checks and how reliable are they?

Mr. Quarles: I was specifically referring to the data transference checks such as: check sums used in connection with transference of information between magnetic tape or magnetic drum and magnetic core storage; "echo checking" of printed output; check sums and/or double-punch-blank-column checks for certain input cards. These checks have been found to be very reliable.

However, macro-operation I, for example, also contains checks which would eliminate very unreasonable data, in addition to various other editing checks as indicated in the paper.

W. W. Youden (National Bureau of Standards, Washington, D.C.): How accurate were your predictions?

Mr. Quarles: The present accuracy of predictions obtained with this programming system varies with the satellite in question. It has been possible to obtain considerably better accuracy for predictions of Vanguard I than for any of the other satellites launched to date. In part this is felt to be due to the availability of more better-calibrated Minitrack stations for recording observational data, but probably primarily due to the greater perigee distance and consequent lower distortion of the orbit due to atmospheric drag. More specifically, predictions made for Vanguard I, and commencing shortly after its launching, have been accurate to within a small fraction of a minute of time.

Use of a Digital Computer for Airborne Guidance and Navigation

S. ZADOFF† AND J. RATTNER†

Introduction

RECENT developments in computer instrumentation have permitted a vast increase in speed and complexity with no increase in the size of the large-scale digital computers designed for scientific computation.

These developments have also made possible a new application for digital computers, namely, "real-time" computation in the field of control systems.

By way of definition, a digital computer is said to operate in "real time" when it is an integral part of a physical control system. One of the requirements for real-time operation of a digital computer is rapid computation consistent with changes in the input physical quantities and the output data rates required by the system.

Historically, the analog computer has been used in control applications. However, the analog computer is intrinsically limited in its ultimate accuracy, whereas digital-computer accuracy can be increased with little change in size or basic complexity.[1] Problem-handling capacity can also be increased for the digital machine with little or no change in its size although this may imply a change in rate. This latter is far from true for the analog computer since its complexity is in one-to-one correspondence with that of the problem it solves.

From this, it follows that there is a point of diminishing returns by way of weight and size in the use of analog computers over the digital type as problem complexity or accuracy needs increase.

The development of simple, reliable logic techniques has reduced the number of vacuum tubes in many computers. Magnetic elements and transistors are on the verge of totally replacing those vacuum tubes still required. The net decrease in size and weight produced by these components is further enhanced by their lesser power requirements. It should be noted that this progress is far from stabilized.

These component developments affect the size and weight of analog computers also, but the increases in speed and reliability in the digital field combined with demands for more complex real-time computers have made the digital machine eminently practical for this purpose.

General Discussion

In real-time computation the problem to be solved is generally described by a system of nonlinear differential equations. The analog computer is a direct physical approximation of these equations. When using digital techniques, an equivalent set of difference equations is set up

† Sperry Gyroscope Co., Great Neck, N.Y.

[1] J. Von Neumann, "The General and Logical Theory of Automata" in "The World of Mathematics," Simon and Schuster, New York, N.Y., vol. 4, p. 2070 ff.; 1956.

and solved by the digital computer. It is necessary that the solution of the difference equations be asymptotic to the solution of the differential system and that the same stability criteria must hold.[2]

Finally, the computation must be performed in a time commensurate with the response characteristics of the physical system.

In addition to solving systems of difference equations, the digital computer can be used as a function generator and a decision device in the control application.

Evidence of the progress of digital computation in the field of automatic control is its use in airborne systems. The Cytac system is an example of an airborne guidance and navigation system using a digital computer in a control loop.

Cytac is a long-range, all-weather, ground-controlled navigation and tactical bombing system. It was developed and tested by Sperry Gyroscope Company under a contract with Rome Air Development Center and Wright Air Development Center.

The system is built around a hyperbolic radio-navigation aid which was also developed by Sperry[3] and is now known as Loran-C. Loran-C is essentially an extension of the principles of the standard Loran system which is presently in use in the Atlantic and Pacific as a long-range aid to marine navigation. Fig. 1 shows a typical configuration. A master station, S_m, transmits radio-frequency pulses at a uniform repetition rate. The two slave stations transmit similar pulses synchronized to the master. A receiver in the service area measures the time differences of arrival of the master pulses and each of the slaves to obtain a fix at the crossing of the corresponding lines of position. Measurements are made only on the ground-wave portion of the received signal.

Loran-C achieves long range by using the low-frequency transmission within the internationally allocated band of 90 to 110 kc. Loran-C is a two-step system and obtains high precision by making a measurement of the phase of the radio-frequency cycles within the received pulses, achieving an instrumental accuracy of 20 to 30 mμsec. The system is fully automatic with respect to both signal acquisition and time-difference measurement and indication.

The output of the time-difference measuring receiver is continuous fix information in hyperbolic coordinates. In the Cytac system the digital computer is used to combine the inherent long-term accuracy and stability characteristics of radio-derived data with the accurate dynamic character of air-derived data in the form of airspeed, compass, and altimeter indications, to provide navigation information having the best qualities of each. This was essentially the first application in which such techniques were used in long-range navigation and guidance.

[2] H. J. Gray, Jr., "Numerical methods in digital real time simulation," *Quart. Appl. Math.*, vol. 12, pp. 133-140; July, 1954.

[3] W. P. Frantz, W. N. Dean, and R. L. Frank, "A precision multi-purpose radio navigation system," 1957 IRE National Convention Record, pt. 8, pp. 79-85.

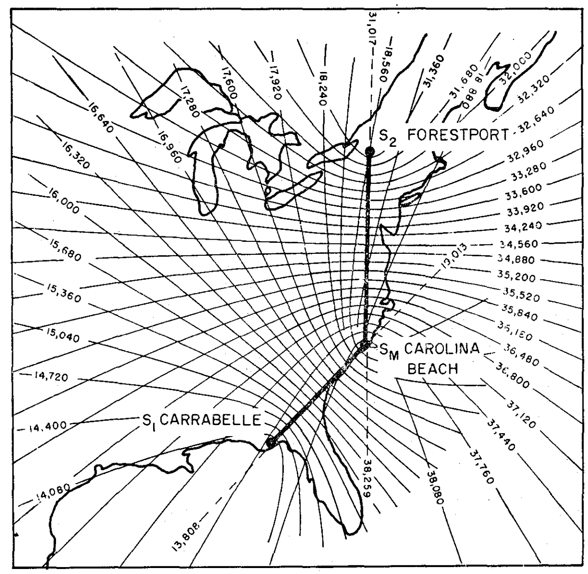


Fig. 1—Typical configuration of Loran-C stations.

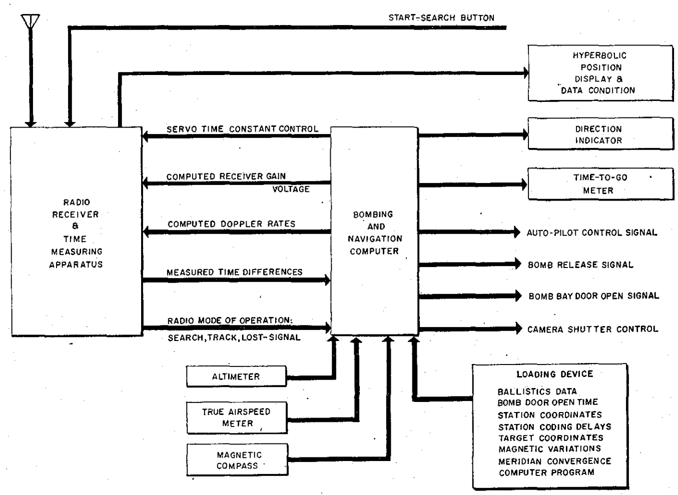


Fig. 2—Block diagram of Cytac system.

Fig. 2 is a block diagram of the Cytac system. The time differences of arrival of radio signals are measured in the radio receiver and time-measuring apparatus and are fed out in the form of shaft rotations. These are converted to digital data by the computer input equipment. The computer also accepts instrument panel information including compass heading, altitude, true airspeed readings and mode of operation of the radio system (*i.e.*, acquisition, track, or lost signal). These inputs are in the form of shaft rotations, which are also converted by analog-to-digital converters at the computer input, or are in the form of relay settings. After operating on these inputs, the computer converts the digital information back to analog or relay data to be used by the other equipment.

The computer output consists of an analog voltage to an autopilot to guide the aircraft toward a correct bomb release point and a signal to control the bomb-bay doors and the actual bomb release. Additional computer functions include:

1) Dead reckoning in the event of lost radio signals
2) Computation of time-difference rates
3) Computation of expected radio receiver gain settings
4) Control of certain servo time-constants and gains in the radio receiver during receiver switch from acquisition mode to track mode of operation
5) Generation of timing signals for control of reconnaissance-camera equipment.

Before take off, the program is stored on the magnetic drum memory of the computer, together with the constants required in the equations and the target coordinates.

System operation is initiated by pressing a "start" button on the pilot's control panel. This puts the radio equipment into its acquisition mode of operation in which it proceeds to locate the radio pulses in time and lock on to the received pulses. The computer notes that signals have been acquired, and normal operation begins.

In the event of lost signals the radio system gives the computer an indication of this condition. The computer then ignores further radio time-difference data and dead reckons on the basis of the last reliable radio information. It continues to feed computed Doppler rates to the radio system so that, if the lost signal is due to a temporary interruption of signal transmission or reception, the radio equipment will be in position to track the pulses as they are received again. This eliminates the need for the receiver reverting to the acquisition mode under these conditions. On the other hand, if the radio-data indication does not show good signals within a reasonable time, the computer directs the radio equipment back to its full acquisition mode.

Fig. 3 is a functional block diagram of the computer operation. The two-step measurements of time difference obtained from radio pulse-envelope measurement and radio-frequency phase measurement within the pulses are converted from shaft rotations to binary form and introduced at block 1. The coarse pulse-envelope time differences and fine radio-frequency phase differences are compared and a pair of consolidated time-difference numbers are obtained. The time-difference numbers corresponding to the center lines between the master and slave stations are subtracted from these numbers to provide a set of numbers suitable to geometric computation.

A conversion from hyperbolic to rectangular coordinates is performed in block 2.

A dead-reckoning computation of the aircraft position in rectangular coordinates is made in block 3, based on air-speed, altitude, and heading. Heading is derived from a Sperry J-2 Gyrosyn® gyromagnetic compass, and suitable corrections for magnetic variations and meridian convergence are provided in block 6. Magnetic heading is then converted to rectangular coordinates and the dead-reckoned rectangular coordinates are compared with the radio-derived data in the same coordinates in block 4. A portion of the difference is fed back through smoothing block 5 to correct the dead-reckoning computation of block 3.

By feeding back only a portion of the difference, the equivalent of an exponential smoothing factor[4] is obtained which reduces the effects of random variations. The correction is applied to the dead-reckoned, apparent wind vector which is substantially invariant for short periods of time. Aircraft steering control is derived from the dead-reckoned solution. Since the long time-constant smoothing is applied to a quantity substantially independent of aircraft heading, it does not materially affect aircraft stability.

On the basis of the corrected, smoothed position and velocity, further computation of the steering from present position to target is done in block 8. The time-to-go and the bomb-release point are computed in block 9. The distance to the stations and the velocity relative to the stations (or the Doppler rates) are computed in block 10. Bomb-ballistics data for a range of airspeeds and altitudes appropriate for each mission are stored in the computer, and exact ballistics for actual airspeed and altitude are derived from the stored data in block 7 for use in the steering and bomb-release computation.

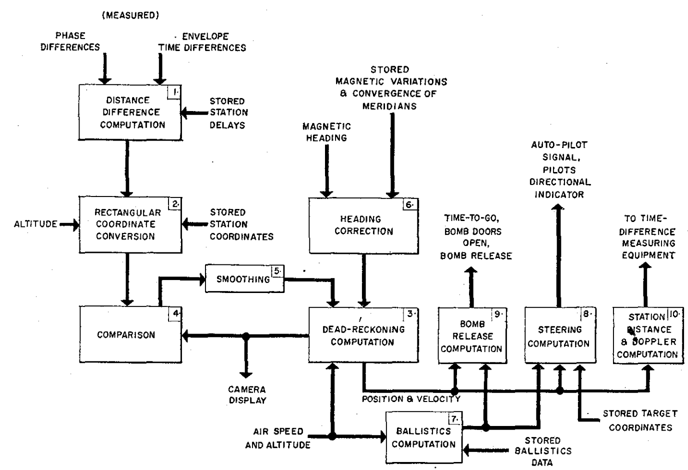


Fig. 3—Block diagram of computer operation.

The next diagram, Fig. 4, shows the two major control loops of the system and their interaction. One of these is a fast control loop comprised of an autopilot, airframe controls and surfaces, heading, airspeed and altitude measurements, and dead-reckoning computation. This control loop is a conventional autopilot arrangement except for the fixed delay introduced by the dead-reckoning computer. It will be noted that this control loop does not contain any long time-constant or narrow-band circuits.

The second loop is a slow control loop which includes the autopilot, airframe controls and surfaces, radio time-difference field, time-measuring apparatus and transfer or smoothing function, computer coordinate transformation, and dead-reckoning computation and computer smoothing. This secondary slow control loop has a narrow-band circuit in the time-measurement smoothing and a very-

[4] R. E. Spero, "Effectiveness of two-step smoothing in digital control computers," Proc. IRE, vol. 41, pp. 1465-1469; October, 1953.

narrow-band circuit in the computer smoothing. Were it not for the fast control loop, severe stability problems would be encountered. Because of the action of the fast loop, however, only noise and signals due to errors in the dead-reckoning computation pass through the narrow-band circuits. Errors in the dead reckoning may be caused by wind changes and also by errors in heading, airspeed, and altitude measurements. Insofar as the problem of stability is concerned, only heading, airspeed and altitude-measurement errors, and wind changes are of importance. Since these errors are small, the secondary slow control loop will remove any cumulative effect of such errors, but will have a minor effect on the airframe stability.

A tertiary control loop is provided by the computed Doppler rates which are generated by the dead-reckoning computation and fed back to the time-measuring apparatus. This is provided primarily for the purpose of providing a memory function in the time-measuring apparatus during a lost-signal condition, and does not affect the basic stability or smoothing considerations.

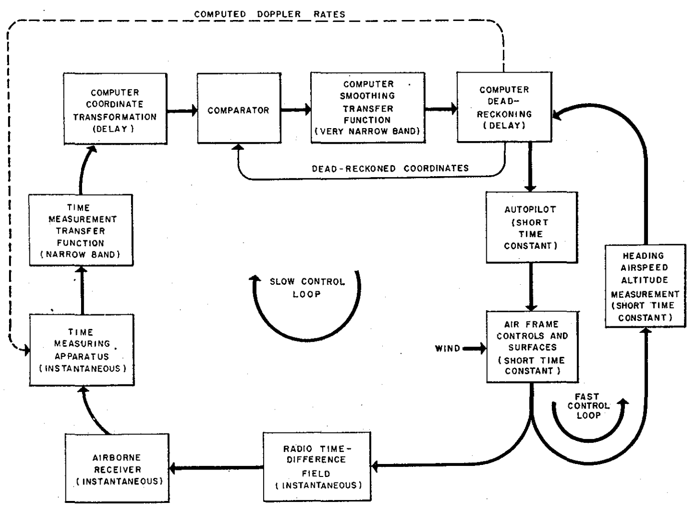


Fig. 4—Control loops of system.

Analysis performed at the beginning of the Cytac development program indicated the need for a computer to perform certain calculations with stipulated accuracy and speed. In order to meet the time schedule for the experimental instrument, an airborne digital computer was procured from a contractor.

This computer is capable of performing all required functions. The computer was programmed for a bombing mission and the pilot could select any of three preset targets. The operating range, distance from all transmitters, and speed and altitude of the aircraft were satisfactorily executed by the computer. In addition, other auxiliary functions for control of radio-system components were programmed for simultaneous solution. However, doubts were raised as to whether this computer was the best possible for the Cytac system. It seemed likely that a digital computer developed specifically for this system and utilizing the latest techniques and components would prove more satisfactory.

The digital computer used in the Cytac system is an optimum-programmed, serial, magnetic drum-storage computer. The drum storage contains 31 order channels, 7 number channels, and one channel for modifiable orders. Read and write are performed by separate heads, the write head being disconnected from the write amplifier for those channels which contain nonerasable program and problem constants. The drum also has one channel for high-speed access.

A novel feature of the drum is the varying of the spacing between the read and write heads of the numerical storage channels. This is advantageous in reducing access time.

The wordlength is 16 binary bits plus sign. There are 64 words per channel providing a total memory capacity of 2496 words. The arithmetic unit consists of an accumulating register, a shift register, an operand register, and the add-subtract matrix. The three registers are dynamic circulating registers.

The arithmetic orders include addition, subtraction, multiplication, division, and square root.

The input-output equipments operate through the input-output unit which automatically writes onto and reads from the drum, utilizing a separate set of heads on those numerical storage channels selected for input and output. A manual control console which is necessary for test procedures and for loading the drum contains the usual display lights and required switches. There is also an array of auxiliary equipment for testing, monitoring, display, problem preparation, and output recording which are not part of the control system.

The computer was designed with the goal of a fast, small volume, low-weight computer which would be just adequate to perform the required functions in about one second. In consequence, the computer was difficult to program and code. This was justifiable only because the code would remain unchanged and be retained on the drum once it was debugged. For those unfamiliar with the problems of an optimum-programmed computer, it may be said that optimum programming requires extensive juggling of orders and intermediate storage positions to achieve adequate results.

Those parts of the computer actually part of the airborne-control loop weigh about 300 pounds and occupy about 6 cubic feet. It is estimated that this computer could easily be reduced to less than 150 pounds and 3 cubic feet by using more modern instrumentation.

While the computer was being utilized, a study was made to find a more suitable computer. This study concluded that a machine using magnetic-decision elements as a basic unit and a magnetic drum as a storage device would be better for the Cytac system.

The set of equations for the Cytac system was selected as that best suited to the characteristics of the digital computer. These equations and their programming were designed to minimize the effects of short wordlength and truncation inherent in the computer. Several alternate sets of equations developed during the Cytac study were discarded because of the difficulty of programming them for the computer. The set of equations used gave satisfactory control of the aircraft and indicated satisfactory bomb release during several test runs made with this computer in a B-29 aircraft.

The computer program was also tested, prior to this time and apart from the actual system, on the Florida Automatic Computer (FLAC) at the Patrick Air Force Base. Furthermore, complete simulation trials were made at a Reac installation at the Sperry Gyroscope Company laboratories using the Cytac-digital computer and the radio time-measuring equipment which had already been thoroughly debugged and flight tested apart from the computer.

The final program used about 1000 orders, 150 constants, and 50 temporary storage addresses. Five channels of 320 words were allocated for test-program storage. The actual length of the computation cycle was set for about one second whereas the time required was about 0.8 second. The one-second period was chosen since it was consistent with airframe-stability requirements and the smoothing factors desired.

This project demonstrated that a digital computer could be utilized as a very flexible part of a control system with reliability and size to make it a practical component of an airborne system.

A typical analog computer arrangement which might have fulfilled all the functions provided by the digital computer would have required more than 25 servoamplifiers, 21 assorted synchros, 45 potentiometers (many of which would require special or high-precision windings), 20 servomotors, 8 tachometers, 20 differentials and assorted gear trains, supporting hardware and electronics, and power supplies. Using present-day techniques, this equipment could also be expected to weigh at least 150 pounds and fill more than three cubic feet, with the question of ultimate accuracy left unanswered. On the other hand, a digital computer designed for the same problem with present-day techniques would require no more space or weight and would definitely be capable of meeting the accuracy requirements. Servicing of either type of computer would not be a pleasure, and reliability and serviceability of each would be on the same order of magnitude.

Conclusion

At the time that the Cytac program began, neither the equations to be solved nor all the functions to be performed had yet been stipulated. Faced by a short development time, an existing general-purpose computer for the job seemed most advisable since it provided ease of making changes in programming and addition of control functions with no extra equipment development. Optimum programming permitted the achievement of computation time commensurate with airframe-stability requirements with a magnetic drum-memory computer.

On the other hand, where there is sufficient time to develop a computer best suited for the job, a special-purpose machine may turn out to be fastest, lightest, and smallest, with a resulting loss of flexibility in making program changes with the ease provided by a general-purpose machine.

Some Experimentation on the Tie-In of the Human Operator to the Control Loop of an Airborne Navigational Digital Computer System

CORWIN A. BENNETT†

Introduction

ONE of the human operator's most important tasks in contemporary bombing and navigational systems is crosshair error correction or "tracking." Due to navigational or intelligence errors, the system's crosshairs may not fall on the target or other reference point. When the operator recognizes this error he sends correcting signals, by means of a hand control, to the computer which then corrects the display.

Typically, bombing and navigational systems have used analog computers to process the operator's control signals. However, when a digital computer is utilized, the operator is faced with the new problem of seeing the results of his corrections periodically on the display at the solution rate of the digital computer.

With this "sampled-data" tracking, when the operator

† IBM Corp., Owego, N.Y.

moves the target across the display it seems to "jump" from point to point. The apparent discreteness is an inverse function of the inertia in the system. Since the operator's control signals are accepted by the computer only at sample times, part of them are ignored. This becomes particularly noticeable at low solution rates. Since the complexity of the digital computer is determined in part by its solution rate, it is necessary to minimize this rate. On the other hand, if the sampling of the operator's control produces poorer tracking performance with lower solution rates it should be maximized. The problem faced by the engineering psychologist is to determine a computer solution rate at which neither of these two goals—equipment simplicity and tracking performance—is unduly sacrificed.

Descriptive Experimentation

A series of experiments was carried out over a period of three years to provide systems engineers with design requirements for digital tracking. While initially the question of required solution rate was the sole object of investigation, later study was devoted to related sampled-tracking problems and to possible ways of circumventing stringent equipment requirements.

Fig. 1 shows the digital control loop studied in most of these experiments. The operator's near-continuous control signals are sampled by analog-to-digital converters. These numbers are processed by the digital computer which, among other things, integrates the signals. This integration means that a rate of crosshair movement is proportional to a displacement of the control which is known as a rate or velocity-tracking control. The computer's outputs are converted back to analog form and displayed as periodic display changes. Feedback is then provided through the operator.

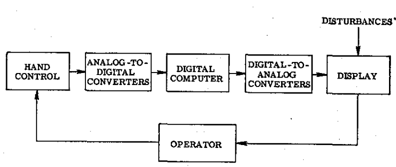
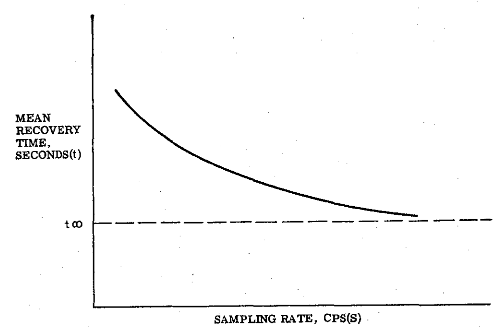


Fig. 1—Digital rate-control loop.

In the experimentation such a loop was simulated by means of an analog computer and relays, a spring-loaded joystick, and a laboratory oscilloscope. The simulation was such that sampling in time (at the "solution rate") was carried out, but sampling in amplitude (at the "quantization level") was not. While quantization could be critical for tracking, the systems converter resolution was such that with a rate control no great problem existed.

Laboratory technicians and engineers served as subjects in each of the experiments. The actual running of a given experiment would last just a few days, although weeks of preparation and equipment "debugging" were generally required. Time records of error were made to obtain performance measures. The usual performance measure was "recovery time"—the time it took the operator to place the target under the crosshair to a given tolerance for a specified initial error. Conventional statistical analyses and significance test were performed on the data.

Fig. 2 shows a typical curve for the relationship between recovery time and solution rate. As the solution rate is decreased, recovery time increases; as the solution rate increases, performance improves, and recovery time approaches that obtained under analog-tracking conditions asymptotically. Statistical tests were applied to determine a specific solution rate which could be considered as yielding performance that was equivalent to analog conditions. In most cases this rate turned out to be on the order of 10 cps—a number to which engineers could design in order to insure no loss of tracking performance with the digital system.



$$t = \frac{K_1}{K_2\sqrt{S}} + t_\infty$$

Fig. 2—Recovery time as function of sampling rate.

One parameter which was discovered to be highly critical in this initial experiment was the display-control ratio, or gain or sensitivity of the hand control. This may be defined as the displacement of the display (or one of its derivatives) for a given control-displacement. Numerous investigators have shown the control sensitivity to be a significant determinant of tracking performance. For example, our results showed a U-shaped relationship between recovery time and sensitivity; that is, as the sensitivity is raised or lowered from optimum sensitivity, performance deteriorates. Furthermore, we demonstrated that, as the sampling rate is decreased, the optimum sensitivity is decreased. This required us to predetermine optimum sensitivities for all experimental conditions prior to testing. For the systems engineer, whether dealing with a digital system or not, the practical implication is apparent. Since, within limits, sensitivity or scale factor is one of the easiest equipment changes to make, an optimum value should be selected for any given situation.

One way of reducing the complexity of the digital computers (other than by reducing the solution rate) is to allow more solution periods for processing each set of inputs. Computer delays or "transmission-type delays" were therefore studied in one experiment.

Fig. 3 shows the results of one solution-period delays, as compared to no delay on tracking performance—re-

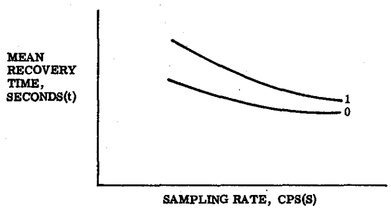


Fig. 3—Effects of transmission delays on recovery time.

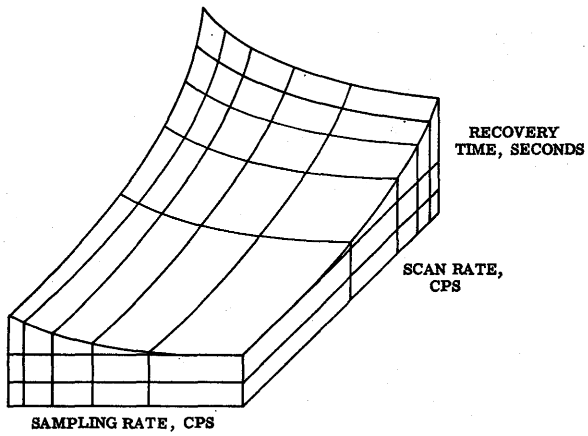


Fig. 4—Recovery time as function of both sampling and scanning rate.

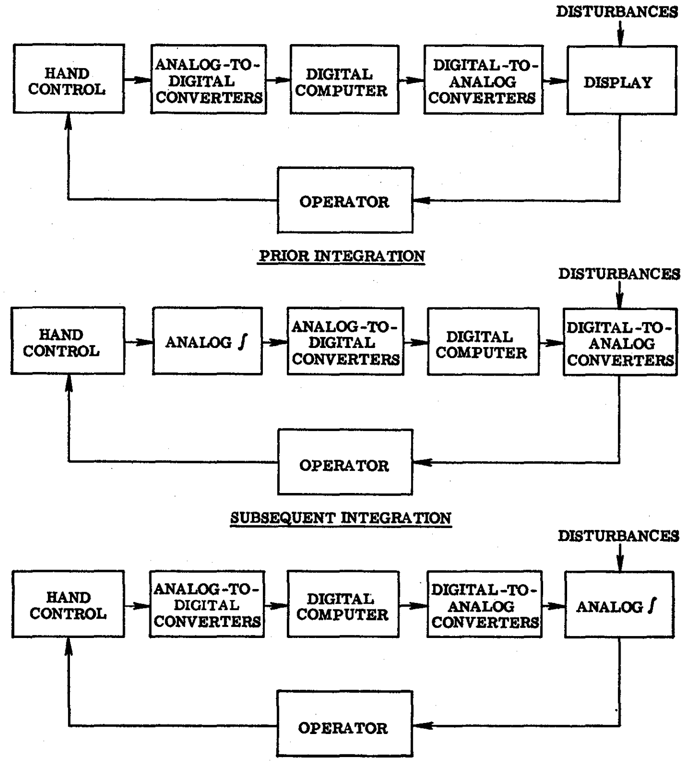


Fig. 5—"Normal," "prior," and "subsequent" intergration loops.

covery time increases with the addition of delays. Similar results have also been found for other types of delays such as the "exponential delays" associated with equipment inertia. Analysis of the two-delay conditions, studied here, indicated that the performance differences would disappear at about 20-cps solution rate.

One further parameter studied was that of scan rate. In the navigation system using a radar display for tracking as well as digital computation, a second kind of time sampling, that of scanning, takes place. Two alternative hypotheses for the possible combined effects of scanning and sampling were suggested. First, the effects of sampling and scanning on tracking might be completely independent. Second, it might be that if the scan rate of the display were very low it would be unnecessary to have high solution rates to optimize performance, that is, there would be an interaction between sampling and scanning effects.

Fig. 4 shows the results of this study. In brief the effects of sampling are of the same nature regardless of scan rate. Both scanning and sampling degrade tracking and they do so independently. Thus, within equipment limitations, both rates should be maximized.

In this study the two rates were unsynchronized. We thought that if we synchronized the scanning and sampling, or set up some phase relation between them, that the scanning could prove a useful signal of sampling time. Thus, at low rates, the operator might benefit from knowing when sample time was going to occur. This did not prove true. The addition of any auditory signal preceding sample time did not prove useful either in the range of practical rates. It is obvious, however, that at very low rates, say 0.01 cps, such a signal would be essential.

Improvements Phases of Investigation

Summarizing the results described, sampling of the operator's control loop reduces the tracking performance, and the lower the solution rate the poorer the performance. The engineer is thus faced with the problem of building a digital computer to operate at higher rates than would be otherwise necessary.

In considering why time sampling degrades tracking with a rate control, some possible solutions suggest themselves. First, at low rates, a noticeable number of the signals the operator imparts to his control are not seen by the computer at all, since they do not occur at sample time. If, however, the hand-control signals were to be integrated analogwise before sampling, all signals would have an eventual effect on the display. This is labeled "prior integration." Second, the discontinuities in events on the display may be causing the trouble. If the integration were performed after the reconversion to analog form, there would be no discontinuity in the positional information on the display. This has been called "subsequent integration." These loops are shown in Fig. 5, with the essential differences from normal" or "digital" integration being the locus of the integration in the control loop. Comparisons were made between the performance yielded for the three loci of integration conditions. In sum, not only did the two analog-integration conditions not prove superior to the digital-integration loop, they proved to be inferior. Just why this happened is not clear, but in any case they offered no solution to the practical problem of building a low solution-rate computer.

Another line of attack to the problem of too high solution rates is to change the control from a rate control to a position control, that is, drop the integration completely. Many data exist for analog tracking which show position control superior to rate. Furthermore, since a position control is not time dependent, it should prove less sensitive to the solution rate. Observations indicate that a digital position control is less sensitive to solution rate, but is poorer over-all in the practical tracking range. The killing blow to digital-position control is quantization or equipment resolution. Whereas with a rate control, amplitude sampling is not particularly critical, with a position control, the analog-to-digital converters must have at least as much resolution as the final precision required in tracking.

A final proposal to the too high computer rates is one that occurred to us quite early in our program of investigations and again at the end of it.

As shown in Fig. 6, an analog control loop within the over-all digital-system loop can be instrumented by feeding information simultaneously to the display, and sampling it for the digital computer. For the operator's purposes, the system is analog, and for other computational purposes, it is digital. With such a "mixed" loop we hope that the sampled-data-tracking problem is "solved" by giving the operator an analog loop, although this instrumentation will probably create problems of its own.

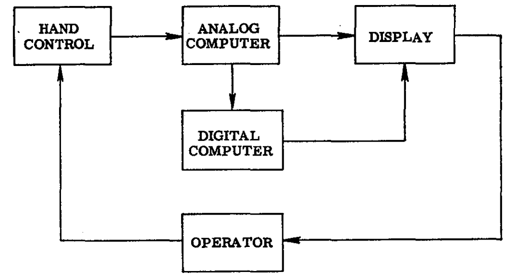


Fig. 6—Analog tracking with a digital computer.

Multiweapon Automatic Target and Battery Evaluator

D. E. EISENBERG†, A. E. MILLER†, AND A. B. SHAFRITZ‡

THE Multiweapon Automatic Target and Battery Evaluator, to be referred to from now on as the MATABE, is a large-scale, real-time, automatic-control computing system developed and built under the sponsorship of the U. S. Army Signal Corps, originally for field installation in the AN/GSG-2 antiaircraft defense system. In conducting the defense of a city against an attacking force of aircraft, the MATABE has the capability of evaluating the changing tactical data provided by the system, automatically determining and initiating individual battery assignments, and recording the detailed history of the raid. Specifically, the MATABE makes more than 136,000 calculations in six-tenths of a second in determining which, if any, of as many as 32 targets should be engaged by a given one of 20 batteries within the defense system.

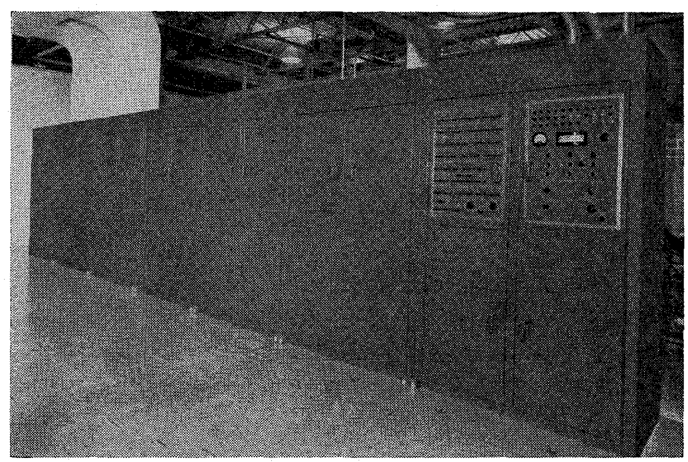


Fig. 1—The MATABE.

Fig. 1 is a picture of the MATABE. The heart of the MATABE system is a digital, binary, parallel, fixed-point, single-address, magnetic drum, tailored, electronic-control computer. Surrounding this portion of the system is an electromechanical input-output system that provides approximately 2500 signal connections to other equipment in the defense system. Table I lists the characteristics of the MATABE.

This paper will describe the information and control sources in the system, as well as the computational organization of the MATABE. Special attention will be drawn to the operational indications of possible malfunctions, since the determination of such situations by a monitor, in the midst of a complex, dynamic environment, is of the utmost importance.

† Burroughs Corp., Paoli, Pa.
‡ Auerbach Electronics Corp., Narberth, Pa.

TABLE I

PRINCIPAL CHARACTERISTICS OF THE MATABE

Word length	20 bits, including sign bit
Instruction rate	5-μsec intervals
Maximum pulse-repetition frequency	2 mc
Addition time, including access time	5 μsec
Multiplication time (two 20-bit numbers)	65 μsec
Divide time (two 20-bit numbers)	80 μsec
Square-root extraction time	160 μsec
Magnetic register storage (nondestructive access)	400 bits (twenty 20-bit registers)
Magnetic drum storage	150,000 bits
Number of different instructions	64
Number of instructions on the drum	3770
Electronic package complement	511
Electron tube complement	2200
Crystal diode complement	13,803
Relay complement	1100
Signal connections to external equipment	approximately 2500
Total power required	29 kw
Total cooling required	9 tons
Volume of cabinets	450 cubic feet

The logical design of the MATABE differs sufficiently from that of other large-scale computers. Consequently, a thorough explanation of any one of its operational aspects would consume more time than is available for a single paper. The MATABE was designed to apply a great quantity and variety of real-time data to a highly complex program, and to perform this function rapidly and accurately. To accomplish this capability, a great many self-checking features and unique computational techniques were incorporated in the design of MATABE. These special features and their integration into the overall MATABE function will be described in this paper; the many details involved in each of the special features will not.

Relation of the MATABE to the System

The integration of the MATABE into the defense system can be better appreciated by looking at Fig. 2. The MATABE is paralleled by a group of human operators who shall be referred to as tactical monitors. Their function is to monitor the MATABE when it is being used as a tactical tool for target evaluation. The battery and target information sent to the MATABE is the same sort of information that is sent to the tactical monitors. In addition to the control lines used by the monitors in operating the MATABE, a monitor type of communication between the MATABE and these operators is also required. In passing, it should be noted that the information content of the displays to the human beings is very similar to the information content of the channels to the MATABE, although obviously in different form. Thus, we find the monitors and the MATABE aware of the deployment and status of the batteries defending the city, as well as the changing configurations of targets and other information associated with evaluating the attacking air threat.

Battery Eligibility for Evaluation

When a battery has completed an engagement, it indicates this fact to the tactical monitors and to the MATABE which stores the information that this particular battery is in a position to initiate an engagement. This, in effect, places the battery's eligibility for an evaluation in a "waiting line." When it becomes time for the battery to be evaluated, the MATABE, on the basis of current strategy, determines which, if any, of as many as 32 targets this battery should engage. This decision is stored in the input-output system of the MATABE which can then communicate with the battery to effect the specific engagement, while the heart of the MATABE is available to evaluate another battery.

There is storage within the MATABE to retain indications of whether each of the 20 batteries is eligible for an evaluation. The electronic portion of the MATABE uses this information to determine which of the 20 batteries should next receive an evaluation. It should be noted that this "waiting" for an evaluation seldom constitutes more than a few seconds. The battery commander requesting an evaluation is seldom aware of any wait at all.

In addition to the 20-battery eligibility circuits, there is also a twenty-first eligibility circuit which represents a fictitious battery used by the MATABE to standardize its computational routines and to effect certain types of error-checking features.

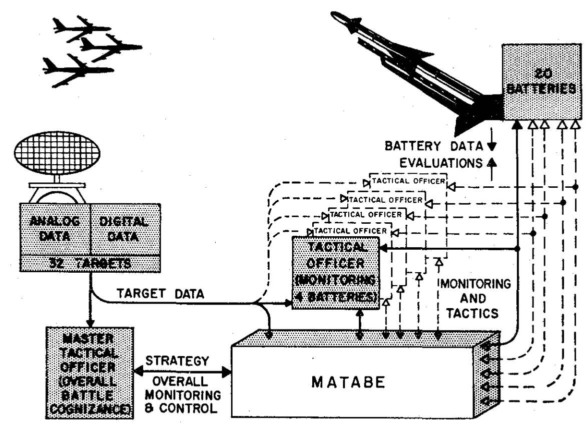


Fig. 2—Functional integration of the MATABE into the AN/GSG-2 system.

Basic MATABE Computations

It is appropriate, at this time, to describe some of the basic types of computation made by the MATABE. We shall concern ourselves only with 22 of these types of computation. The first is a scheduling program that determines which of the other 21 routines the MATABE should embark upon. Twenty of the remaining 21 programs are evaluations for the 20 batteries; the twenty-first program is an evaluation for the fictitious test battery. The test-battery evaluation is basically the same as the other battery evaluations except for error-checking features which will be mentioned later. References to the organization within an evaluation program will thus apply to the test-battery program as well as to the 20 battery programs.

Let us consider how the MATABE reads and uses its program. There are 3772 instructions stored around the drum. Each of these instructions is read from the drum each revolution. Each time an instruction is read from the drum, a decision is made to perform the instruction, or to ignore it. Whether a given instruction is performed may depend upon previous computation results and upon the contents of various registers, particularly the battery register (which identifies the battery being evaluated) and the revolution register (which identifies the number of drum revolutions that have occurred thus far in the program). The type of operation called for by each instruction also may be dependent upon the information currently stored in these registers; for example, some of the instructions carry out different operations in different drum revolutions. In this way, rather than by using jump instructions, the MATABE differentiates between, and carries out, its required programs.

Each of the MATABE instructions is carried out within a 5-μsec interval. The operations which correspond to the more complicated instructions in a general-purpose computer (such as multiplication and division) are compiled by using several of these 5-μsec instructions. The data that are required from the drum are obtained within a 5-μsec period. Thus, during a 5-μsec interval, an instruction and a corresponding address are read from fixed tracks on the drum, and the associated operation specified by the instruction is carried out. If this instruction were the add instruction, and the specified address were one of the twenty 20-bit random-access magnetic-core registers, the contents of this specified register would be added to the accumulator. The same register would again be available during the next 5-μsec interval.

When data are to be read from the drum, the information is available from the location on the drum which corresponds in time to the instruction specifying the use of the data. For example, if during a 5-μsec interval, we wish to read the information on the temporary storage tracks, the instruction would enable the data then available from the temporary storage tracks to be read into the accumulator. Thus, there is no addressing system associated with the drum.

In summary, there are 3772 instructions stored around the drum. The instructions effective in a particular revolution of the drum are determined primarily by three factors: first, the contents of the battery register; second, the contents of the revolution register; and third, the results of previous calculations in the revolution.

Returning to the discussion of the types of programs, Fig. 3 illustrates the evaluation and scheduling programs. The MATABE repeats a cycle of operation for each battery evaluation. The drum revolutions are numbered 0 to 34. One revolution is used for each of the 32 target computations in a battery evaluation. Revolution 33 is used for several special checking operations, including the computation of a figure of merit for the engagement of a fictitious test target by the battery. The figure of merit for the engagement of the test target by any of the real batteries is made zero during all evaluations except that of Battery 21. Revolution 33 is also used to specify to output equipment the choice of engagement selected by the MATABE for the battery under evaluation. Revolution 34 is used in performing the scheduling program, and revolution 0 the setup routine which differentiates the various evaluations. Note that some drum revolutions are actually only part of an actual, physical revolution of the drum.

As a typical example, consider the battery 17 has been selected for evaluation. The setup routine manipulates the various constants associated with battery 17, making them ready for the actual evaluation to take place. During each of the next 32 drum revolutions, a figure of merit is computed. This figure corresponds to the predicted results of the engagement by the battery of the target corresponding to the drum revolution number. The highest figure of merit obtained in these 32 revolutions is retained. For example, during the twenty-eighth drum revolution after the setup revolution in an evaluation for battery 17, a figure of merit is computed for the engagement of target 28 by battery 17 at this time. If a previously computed figure of merit was higher than that for target 28, the latter would be ignored; if the figure of merit for target 28 was higher than the previous maximum, the prior figure retained would be replaced by that for target 28.

The thirty-third drum revolution is used for checking purposes which will be discussed in a few moments. The end of revolution 33 is used to indicate, to the input-output equipment, the number of the target which the MATABE has determined the battery should engage, or a number indicating that the battery should not engage any target at this time. In the latter case, the MATABE makes the battery eligible for an evaluation, after a suitable delay.

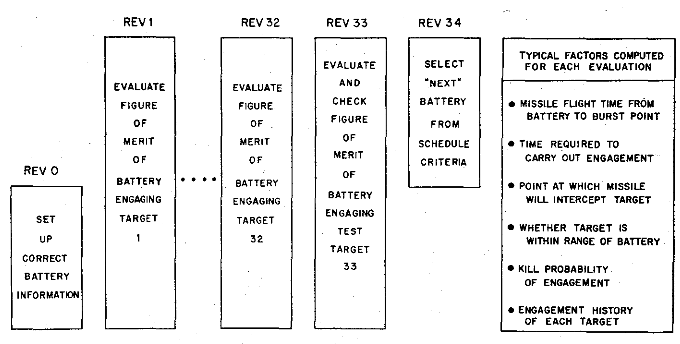


Fig. 3—MATABE evaluation cycle.

MATABE Input Functions

Let us now focus our attention on the input portion of the MATABE system. Consider the case in which a battery has been selected by the scheduling program from those eligible for evaluation. From among the 20 sets of control and information lines associated with the 20 batteries, the MATABE must select those which correspond to the battery selected. At the same time, the MATABE must proceed to obtain correct information about the attack as it is seen by the defense system at this time. This target information is brought into the MATABE in both digital and analog form. Furthermore, the electronic portion of the MATABE does not indicate different input addresses to differentiate between any two real targets; that is, during the calculation of the figure of merit, the same input addresses for target information are repeated exactly, once per drum revolution, during a given battery evaluation program. The differing figures of merit that result for different targets are caused by the input information that is available at the input addresses. Therefore, the input-switching system of the MATABE must present, to the electronic portion of the MATABE, the information associated with a given target at the time that this information is needed by the electronic portion. Thus, information on target 1 must be switched into the MATABE so that it is available during drum revolution 1, target 2 during drum revolution 2, and so on, up through target 32 during revolution 32.

During the time that specific target information is being made available in a given revolution, the MATABE samples the digital sources of information and converts the analog data into digital form. Possible changes in the digital data, that might occur during a sampling, are taken care of by means of multiple sampling of the same source, or by using gray code on the input information. The input switching is realized by means of a relay ring counter which is stepped once per drum revolution.

Error-Checking Features

Prior to considering MATABE output functions, it is appropriate to turn to the questions concerning error checking. First, the MATABE is programmed in such a way as to prevent the same battery from being evaluated twice in a row. Thus, in an evaluation for a particular battery, an error caused by something peculiar to that battery will not occur in the next evaluation. Because of this feature, those troubles which concern only one battery are isolated. If, however, the MATABE makes an error in two consecutive evaluations, such an error is considered to be *significant*. Since the MATABE can perform evaluations for any of the batteries independently of the others, the ability to isolate nonsignificant errors is obviously very important in the over-all operation of the machine. Thus, if the part of the MATABE which is associated with a specific battery should become faulty, the MATABE can continue to perform evaluations for the other batteries successfully. In the design of the machine, this isolation was carried down to the level of individual fusing for the battery circuits.

Let us now look, in further detail, at the error-checking facilities within a particular evaluation. Reading and writing associated with the drum include a parity check. The ability of the instruction and address registers to set up properly is checked periodically. During each drum revolution, a diagnostic program is carried out. If any mistake is discerned, the whole evaluation can be automatically discarded, and the battery automatically made eligible for another evaluation.

During drum revolution 33, the ring-counter switching is checked, since it should now be in a position corresponding to target 33, the fictitious target. The information that is read through the switching system is compared with the correct value for target 33 (which is stored in another part of the MATABE) in order to determine whether this switching has been adequate. The analog-to-digital conversion is also checked at this time.

In the case of battery 21, the fictitious battery, a merit number is computed for the battery 21/target 33 combination. This number and the subcalculations derived to produce the number are known by the computer, so the whole computation can thus be checked. Since battery 21 receives an evaluation the majority of the time, the MATABE spends quite a bit of its time checking itself. At this time one might ask why the MATABE is so fast; why must battery evaluations be carried out in only six-tenths of a second if the MATABE is to spend most of its time checking itself by means of fictitious battery evaluations? The reason for this apparent luxury is that, while this great speed is not needed in carrying out a specific battery evaluation, the speed is necessary in order to avoid a considerable delay caused by a waiting line. In other words, the MATABE is designed to handle a peak-load problem.

MATABE Output Functions

We are now in a position to consider the output system of the MATABE and the man-machine relationship associated with monitoring the operation of the MATABE. The batteries go about performing their engagements and, at the end of each of these, indicating to the MATABE their eligibility for another engagement. As was mentioned, this eligibility is also indicated to the tactical monitors. At this point it is well to differentiate between two types of tactical monitors as related to the MATABE. There are six of these monitors. One of these, the master tactical monitor, is concerned with the over-all battle situation, and with the operation of the MATABE. Each of the other five is associated with monitoring the operation of four batteries. These latter five monitors receive information indicating that batteries are eligible for evaluation and also receive information relating to the operational effectiveness of batteries. The master tactical monitor is concerned with the decision to use the MATABE, as well as with its over-all operation. Thus, if the MATABE were making significant errors—that is, errors in two successive evaluations—the master tactical monitor would receive this information.

Now consider a situation in which a specific battery is eligible for an evaluation. This information is sent to the MATABE as well as to the appropriate tactical monitor. The tactical monitor is advised of the battery's eligibility by the lighting of an eligibility lamp on his console. When the MATABE proceeds to carry out the evaluation for this battery, the eligibility lamp on the tactical monitor's console is extinguished as the MATABE starts the evaluation. Normally, the evaluation is completed and an engagement initiated. However, if there were something faulty with a part of the MATABE concerned with this battery, the resulting error would cause the MATABE to automatically put the battery back into eligibility so that it could be considered, at least, one evaluation period later. These events would be indicated to the cognizant tactical monitor by the extinguishing of the eligibility lamp for that battery for a period of six-tenths of a second, followed by the reillumination of the lamp. However, since most of the time the MATABE would be evaluating battery 21—that is, there are no real batteries available or requiring evaluations—the battery involved in the trouble would be evaluated every other time. The eligibility lamp would thus be lighted for six-tenths of a second and then extinguished for six-tenths of a second. The master tactical monitor would not directly know of this trouble. From his point of view, the MATABE would be operating well, in an over-all sense. However, the cognizant tactical monitor would notice the flashing lamp which would immediately indicate to him the fact that the MATABE was not operating well for his particular battery. Note also, that if an error is of a transient nature—if it is not repeated when the battery is reevaluated—the MATABE causes no indication requiring action or consideration.

During the time that the MATABE is processing battery evaluations, it is also recording a running history of the raid. This record of the raid can be used to help evaluate the performance of the defense system and of the MATABE itself. The significant data for each evaluation are punched out in a 5-bit teletype code while the MATABE is performing the succeeding evaluation. Interlaced

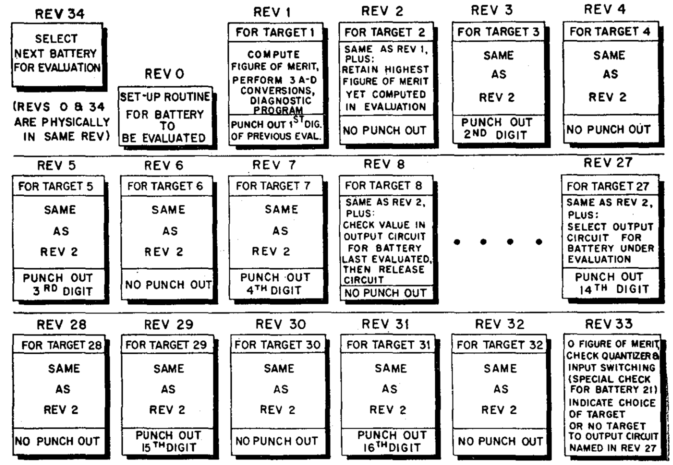


Fig. 4—Instruction program for MATABE battery evaluation cycle.

in the complete cycle of instructions for each battery evaluation is a set of instructions indicating the moment to punch out each digit of data concerning the previous evaluation. The manner in which this is accomplished is indicated in Fig. 4, which also effectively summarizes the entire MATABE computational routine. The various aspects of the MATABE's position in the over-all system stem from the sequence of instructions in drum revolutions 0-34.

We hope that the foregoing has given a fairly broad, but comprehensive, picture of the operational aspects of the Multiweapon Automatic Target And Battery Evaluator which is probably the first large-scale digital computer whose design was tailored to effect a real-time control function.

Control of Automobile Traffic—A Problem in Real-Time Computation*

D. L. GERLOUGH†

Introduction

Automobile traffic in urban areas has been characterized as one of our most serious engineering problems.[1] Fig. 1 to Fig. 3 show respectively for one particular area the recent increases in population, motor vehicle registrations, and motor vehicle registrations per person. Some urban areas have approached the traffic problem by the construction of a network of superhighways or freeways. In other areas a solution has been sought through improved traffic controls on existing streets. But in many cases traffic continues to grow more rapidly than corrective measures can be provided. The question is frequently asked, therefore, "Can we improve traffic movement by some control system which takes ad-

* The opinions expressed herein are those of the author.

† Inst. of Transportation and Traffic Eng., University of California, Los Angeles, Calif.

[1] H. E. Wessman, "What are contemporary demands on the engineering curricula—civil engineering," *J. Eng. Educ.*, vol. 43, pp. 298-302; January, 1953.

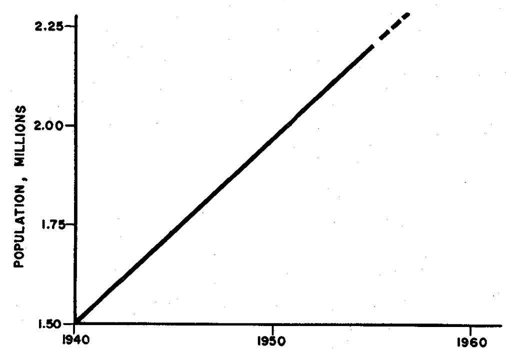


Fig. 1—Population growth, City of Los Angeles.

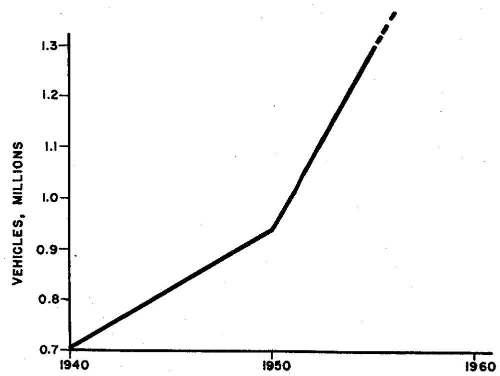


Fig. 2—Motor vehicle growth, City of Los Angeles.

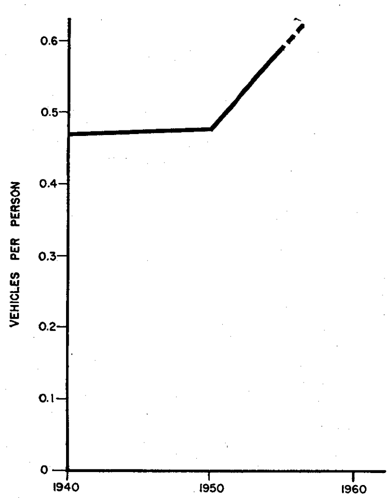


Fig. 3—Increase of vehicles per person, City of Los Angeles.

vantage of the abilities of the large-scale automatic computer?" It is the purpose of this paper to discuss some of the possibilities and difficulties involved.

Area Control of Traffic

In approaching this problem the engineer sees a control spectrum. At one end, control is accomplished by supplying advisory or mandatory instructions to the driver, who, in turn, executes control orders. At the other end of the spectrum, control might be made fully automatic with the driver playing no part as long as the vehicle remains within the control system. For the purpose of this discussion these extremes will be designated "the manual system" and "the automatic system."

Manual System

In the manual system, control consists of gathering information on present traffic, comparing present traffic with stored information on past traffic behavior, and supplying information to the drivers as to how to proceed. An example of a crude form of this type of control takes place annually on New Year's Day in Pasadena, Calif., where extremely large crowds gather for the Tournament of Roses Parade and football game. For several years it has been the practice of the Chief of Police to take to the air in a blimp or helicopter carrying police radio equipment. On the ground, police cars are stationed at strategic control points. By observation from the air it can be determined which thoroughfares are overloaded, and which, if any, can carry additional flow. This information is used as a basis of radio commands to the various control points to cause diversion of traffic from overloaded to underloaded thoroughfares. (The police officers give instructions to the drivers who control the cars. Here instructions are mandatory.)

The City of Los Angeles for nearly two years has had a helicopter which is used primarily for freeway control during the rush hour periods. Observations are made of tie-ups or potential tie-ups and corrective action is taken. Where a tie-up occurs, information is sent to other drivers via radio advising them to take a different route.

Several cities have been experimenting with closed-circuit television as a means of obtaining information on traffic behavior. It is not inconceivable that information in the form of maps, etc., might be transmitted to the driver by television.

Thus, with manual control, one of the principal techniques is diversion of traffic from overloaded to lesser loaded thoroughfares.

Another technique is the control of traffic signals on an area-wide basis. In Denver and Baltimore, there have been approaches made to the control of traffic signals in the city as a whole on the basis of the traffic actually present. These approaches have, however, been based on a limited number of sampling points, a limited number of control possibilities, *e.g.*, signal cycle lengths, and communication with the driver solely on the basis of conventional traffic signals.

To obtain the maximum benefit from control of traffic on an area-wide basis with the manual system, it will be necessary to have many sampling points, several forms of communication with the driver, a large stored background of information on traffic behavior within the area, and a large central computing facility. Stored information must include anticipated origins and destinations of traffic as a function of time of day and day of week. Unusual patterns on occasions of special events must also be known. Characteristics of the complete street network must be stored in the forms of lists of parameters. Most important of all, there must be information in the form of equations, curves, or simulation procedures which will permit the computation of the flow behavior on a given thoroughfare under varying conditions. The central computer will evaluate the existing situation and select the appropriate control measures.

Traffic signals of the conventional type will still constitute an important communication channel between the system and the driver, but other forms of communication will play an increasingly important role. There may be wide usage of changeable signs to convey special messages to the driver at appropriate times. For instance, neon signs, similar to those used on some of the Eastern turnpikes to inform drivers of snow, ice, etc., may be used to inform the driver of changes in turning regulations, direction of flow on one-way streets, closing of streets, etc. In many locations even a series of neon signs may prove to be too inflexible, and a sign made up of individual lights may be needed. This sign could display a moving message similar to that used to convey the news at Times Square in New York, or more likely as a sign of similar type construction but with the message not moving. Such signs can be remotely controlled by a computer. Radios can become an increasingly important method of communication to the driver, and it is conceivable that the use of radio might be mandatory for the driver just as radio is mandatory for the flyer who wishes to make use of certain airports and certain air navigational facilities.

Traffic can be sensed by the techniques to be described in connection with the automatic system.

To summarize: In control by the manual system, operation is manual only in that the actual driving of the vehicle is manual. Selections of routes, etc., are performed by the central computer. Benefits will come through the diversion of traffic to various routes so that the load is spread more uniformly, and through the use of extremely flexible signal timing.

Automatic System

In an automatic system the driver does not have direct control of the vehicle and there are many ways in which marked improvements in traffic flow may be obtained. Fig. 4 shows the form of curve relating the number of vehicles per mile in a traffic lane and the number of vehicles which can flow per hour in that lane.

It will be noticed that at the peak capacity there are only about 2000 cars per hour traveling per lane at a density of around 100 cars per mile. In other words, under present traffic situations, the amount of unused space in the traffic stream is appreciable. The drop off from the maximum flow occurs by virtue of the fact that drivers must main-

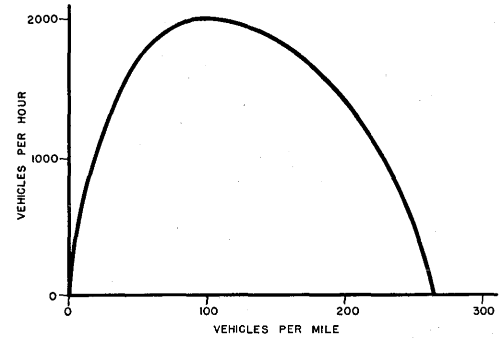


Fig. 4—Form of density-volume relationship for single traffic lane.

tain increasingly larger spacings as speeds increase. If some method were devised whereby vehicles could be operated close together without danger of collision, it might be possible in an extreme case to operate as high as 264 vehicles per mile with no change in spacing as speeds increase. For instance, Le Tourneau[2] has indicated a technique for coupling of long trains of vehicles to be operated on normal streets and roadways. It seems doubtful, however, that the motorist would ever accept an actual coupling of vehicles, but if a method were achieved by which vehicles could be operated more closely at least on roads of a freeway type, considerable economy would result. Close operation necessarily implies, however, operation of all vehicles at approximately the same speed. It is possible that drivers might accept traveling at a uniform speed if the benefits achieved thereby were quite clear.

Zworykin and his associates, on the other hand, have demonstrated by means of a model a method by which vehicles follow a buried conductor and in which speeds can be different for different vehicles, passing being permitted.[3] The guidance principle of the buried conductor has been demonstrated for full-scale usage by a vehicle designed for use in the arctic.[4]

From the standpoint of optimum control it would be desirable to maintain a continuous record of each vehicle in the system. The computing task involved, however, would be so large as to dwarf several SAGE systems, and thus it does not appear that this would be feasible. Instead, as much of the control as possible should be carried in the individual vehicles. One might visualize, then, that the ultimate achievable control system would contain some sort of guidance, by buried conductor or otherwise, including collision prevention and automatic provision for passing as the vehicles come too close together. There should be provision for automatically selecting the optimum routes to various portions of the area on the basis of the traffic pres-

[2] "Trackless cross-country freight train has all-wheel drive," *Elec. Eng.*, vol. 75, p. 95; January, 1956.

[3] "Possibilities of electronic control of automobiles explored by Dr. Zworykin," *Elec. Eng.*, vol. 72, pp. 849-850; September, 1953.

[4] C. O. O'Rourke, "Electronic trail-finding," *Control Eng.*, vol. 4, pp. 117-119; May, 1957.

ent and the amount of traffic going to each zone. Each driver on entering the system could, for instance, set a destination indicator in his vehicle. This could be a tap switch which would select a signal to be emitted from his vehicle and picked up by appropriately placed scanners on the roadway. These scanners would count the number of vehicles going to a given zone, and a computer would select the appropriate routing accordingly.[5] As routings were computed, optimum exits for each zone would be established for the current amount of traffic. On approaching the designated exit for the particular zone of destination, the vehicle's emitted signal would be sensed, and the vehicle would be automatically guided to the deceleration lane leading to the exit. Here the automatic control would cease and manual control would begin. The manual system has the advantage of a much lower cost in that it makes use of the computing and control facilities of the human operator; it does not necessitate reconstruction of the existing highways to provide the facilities necessary for fully automatic control. It can thus be accomplished at an earlier date and accomplished in a stepwise fashion.

The principal benefit from the automatic system is, then, increased flow (*i.e.,* increased capacity) on a given facility. A fringe benefit will be the decrease in tension on the part of the drivers on being freed of the driving task within the freeway system.

System of the Future

If, then, one may be permitted prevision, an urban traffic system in the year 19XX may be something like this: Long distances will be traveled on a system of freeways where control will be conducted in the automatic mode. Entrances and exits of these freeways will connect with one-way streets where parking is prohibited; these streets will serve as the carriers for intermediate distances. On these intermediate streets control will be conducted in the manual mode; drivers will receive instructions by means of traffic signals, special signs, and radio. Between these intermediate thoroughfares there will be "local" streets on which there will be no central control. That is, the driver will have complete control subject only to conventional traffic signals.

The automatic-control equipment will consist of units carried by each vehicle, sensing units located at appropriate points throughout the street network, and a central control unit containing a computer.

Vehicular Units

Each vehicular unit will contain: 1) the destination indicator composed of a signal generator, a selector switch, and the appropriate radiation equipment, 2) automatic tracking and control equipment to permit following a conductor in the pavement or other guidance, including facilities for passing and for collision prevention (while collision prevention will be mainly in the automatic mode, provision can be made to permit its use in the manual mode as well), and 3) radio equipment, either a standard AM receiver or a special receiver for control messages.

[5] To avoid confusion and disruption of such a system by visiting vehicles, visitors would be required to stop prior to entering the system to pick up a map and code sheet so that they could properly adjust their destination indicators.

Sensing Units

Sensing units will have the ability to determine for each passing vehicle its presence, speed, and destination. The destination will be ascertained, as previously stated, by sensing a driver-selected signal emitted from the vehicle. The sensing unit will have the ability to accumulate data for later transmission via digital data link on receipt of an interrogation signal.

Central Control Unit

The central control unit will have a programming device which periodically interrogates the various sensing units. Origin and destination information for the traffic in the system will be continuously accumulated with appropriate updating.

There will be stored, probably on some random-access large-capacity medium, information on past origin-destination movements; information to be stored could well include such items as time of day and rate of onset for particular flow patterns, and the optimum handling of these patterns. Special provisions for emergency situations such as diversion of traffic from disaster areas could be provided for in advance. To aid in the compilation of this stored information it would be desirable for the computer to possess learning ability. One computer can serve both the automatic and manual portions of the system, or there can be a separate computer for each portion with intercommunication between the two.

The computer will continually compute control parameters on the basis of the origins, destinations, volumes, and speeds of existing traffic by means of analytic relationships or simulation routines. These parameters will provide a basis for searching the stored body of knowledge in order to find the appropriate listing of optimum control procedures. These procedures will be read from storage to the control transmitter which will cause them to be executed. As vehicles pass various exits of the system exit data will be fed back to the computer as a check on performance.

Development of System

Such a system cannot, of course, spring into existence full grown. It must be built in a piece wise fashion over a number of years. While much of the computer technology is presently at a stage which would permit the immediate start of design, much research and development will be required in other phases of the problem.

One thing which needs to be decided early is the form of guidance to be used. This information should be made available at the earliest possible date to the designers of new freeways and automotive equipment. It is visualized that there might be a long transition period in which some vehicles would be equipped with guidance facilities and others would not. It would be necessary to set a date after which no new vehicles would be sold without guidance facilities and a still later date beyond which no vehicle would be allowed to use a freeway-type road unless so equipped.

Systems of intermediate streets should be developed as rapidly as possible and can provide immediate relief to certain existing situations.

Research Needed

The area requiring the most investigation is the formulation of relationships describing traffic flow and indicating the measures for optimization. While progress is being made in theoretical investigations by Lighthill and his associates at the University of Manchester,[6] Richards,[7] Prager and Newell at Brown University,[8] Edie and others at the Port of New York Authority,[9] the staff of the Chicago Area Transportation Study,[10] and Pipes at the University of California,[11] there is at present no comprehensive theory of traffic flow. To bridge this lack of theory, development of traffic simulation techniques has been undertaken at the University of California by the writer and others,[12] Goode and others at the University of Michigan,[13] Wong,[14] and the staff of the Road Research Laboratory in England.[15]

Paradoxically, while traffic is a very important and complex engineering problem, the amount of high-grade technical talent applied to this problem has been exceedingly small in comparison to the technical skills required for the development of a single large-scale weapons system. There are few agencies conducting continuing research in problems related with the possible use of computers in large-scale traffic control systems. To the best of the writer's knowledge, all efforts to date have been supported by rather limited budgets. If there is to be any major change in the handling of traffic, such as that visualized in this paper, there must be early recognition of the need, and the appropriation of adequate funds by both public agencies and commercial interests so that the needed research and development may be accomplished in time to permit an evolutionary change.

[6] M. J. Lighthill and G. B. Whitham, "On kinematic waves, II. A theory of traffic flow on long crowded roads," *Proc. Roy. Soc. A, London,* vol. 229, pp. 317-345; May 10, 1955.
S. C. De, "Kinematic wave theory of bottlenecks of varying capacity," *Proc. Cambridge Phil. Soc.,* vol. 52, pt. 3, pp. 564-572; July, 1956.

[7] P. I. Richards, "Shock waves on the highway," *Oper. Res.,* vol. 4, pp. 42-51; February, 1956.

[8] W. Prager, "On the Role of Congestion in Transportation Problems," Div. Appl. Math., Brown Univ., Providence, R.I.; March, 1955.
—, "Problems in traffic and transportation," *Proc. Symposium on Operations Research in Business and Industry,* Midwest Res. Inst., Kansas City, Mo.; April, 1954.
G. F. Newell, "Statistical analysis of the flow of highway traffic through a signified intersection," *Quart. Appl. Math.,* vol. 13, pp. 353-369; January, 1956.
—, "Mathematic models for freely flowing highway traffic," *J. Oper. Res. Soc. Amer.,* vol. 3, pp. 176-186; May, 1955.

[9] L. E. Edie, "Expecting of multiple vehicle breakdowns in a tunnel," *Oper. Res.,* vol. 3, pp. 513-522; November, 1955. Discussion and author's closure, vol. 4, pp. 609-619; October, 1956.
E. S. Olcott, "The influence of vehicular speed and spacing on tunnel capacity," *J. Oper. Res. Soc. Amer.,* vol. 3, pp. 147-167; May, 1955.
L. C. Edie, paper in preparation for presentation at annual meeting of Highway Res. Board, January, 1958.

[10] R. L. Creighton, "Speed volume relationship on signalized roads." *C.A.T.S. Res. News,* vol. 1, pp. 6-11; June 21, 1957.

[11] L. A. Pipes, "A Proposed Dynamic Analogy of Traffic," Special Study, Inst. Trans. and Traffic Eng., Univ. of Calif., Los Angeles, Calif.; July 11, 1950.
—, "An operational analysis of traffic dynamics," *J. Appl. Phys.,* vol. 24, pp. 274-281; March, 1953.
D. L. Gerlough, "Automatic computers for traffic control," *Munic. Sig. Eng.,* vol. 17, pp. 40-42, 60-62; July-August, 1952.

[12] D. L. Trautman, H. Davis, J. Heilfron, E. C. Ho, J. H. Mathewson, and A. Rosenbloom, "Analysis and Simulation of Vehicular Traffic Flow," Inst. Trans. and Traffic Eng., Univ. of Calif., Los Angeles, Calif., Res. Rep. 20; December, 1954.
J. H. Mathewson, D. L. Trautman, and D. L. Gerlough, "Study of traffic flow by simulation," *Proc. Highway Res. Board,* vol. 34, pp. 522-530; 1955.
D. L. Gerlough and J. H. Mathewson, "Approaches to operational problems in street and highway traffic," *Oper. Res.,* vol. 4, pp. 32-41; February, 1956.
D. L. Gerlough, "Simulation of freeway traffic by an electronic computer," *Proc. Highway Res. Board,* vol. 35, pp. 543-547; 1956.

[13] H. H. Goode, C. H. Pollmar, and J. B. Wright, "The use of a digital computer to model a signalized intersection," *Proc. Highway Res. Board,* vol. 35, pp. 548-557; 1956.

[14] S. Y. Wong, "Traffic simulator with a digital computer," *Proc. WJCC,* pp. 92-94; 1956.

[15] Several unpublished technical memoranda.

Discussion

J. L. Jones (Chrysler Corp.): From your paper, I received the impression that most of the work done has been on traffic pattern recognition to which an already known solution may be applied. If this is true, has any work been done on a mathematical model to which analystical processes may be applied?

Do you advocate that automotive manufacturers consider future inclusion of a "traffic control radio" as standard equipment? If so, what should the salient features of such equipment be?

Mr. Gerlough: Work is being done on mathematical models, but it is progressing slowly. Many of the investigators are in universities and have not had budgets to cover this type of work. In recent months there has been some interest shown by one of the automobile manufacturers, and it is hoped that this will result in an increasing rate at which mathematical studies progress.

Yes, I would advocate such a radio as standard equipment. The specifications should be worked out by some national committee which should include representatives of automobile manufacturers, highway people, and the FCC.

Physical Simulation of Nuclear Reactor Power Plant Systems*

J. J. STONE, JR.†, B. B. GORDON†, AND R. S. BOYD†

ONE method of control of a heterogeneous boiling reactor uses the steam pressure in the reactor vessel to control the height of the water reflector surrounding the core. As the steam pressure increases, the reflector height and the reactor power level decrease. Thus, as the steam load varies, the pressure varies and forces the reactor power to follow the load changes.

Fig. 1 illustrates, diagrammatically, the reactor system as controlled by the height of the water reflector. The upper portion of the pressure shell collects the steam produced by the boiling within the reactor-core assembly and delivers this steam to the load attached to the system.

Water coolant in the lower portion of the pressure shell covers the reactor-core assembly. Boiling of this water within the core produces steam, and the flow of steam upward through the core results in a circulation of water up through the reactor, and then, after passing through ports in the annular reflector tank, the water flows down past the core along the inner surface of the main pressure shell.

The annular reflector tank surrounding the reactor core is partially filled with water. This water acts as a reflector for neutrons produced by the core, and as the level of this water decreases, the reactivity decreases. Openings around the top of the reflector tank admit steam to the upper surface of the water in the reflector tank. The water in this annular tank connects, via a pipe, with an external surge tank in which a reference gas pressure is maintained. Any excess steam pressure in the reactor over that required to maintain the water in the reflector system at equilibrium will cause the following sequence:

1) Flow of water to the surge tank,
2) Decrease in reflector level,
3) Decrease in reactivity,
4) Tendency for a decrease in reactor power, and
5) Return of the steam pressure to its equilibrium value.

Initially the system was studied by an all-electronic simulation. This simulation required making assumptions concerning the magnitude of frictional forces in the hydraulic system. It was assumed also that inertial and frictional terms in the equations of motion of the water in the reflector system were determined primarily by the size of the connecting pipe. To determine the validity of these assumptions, a physical simulation of the hydraulic portion of the system was undertaken.

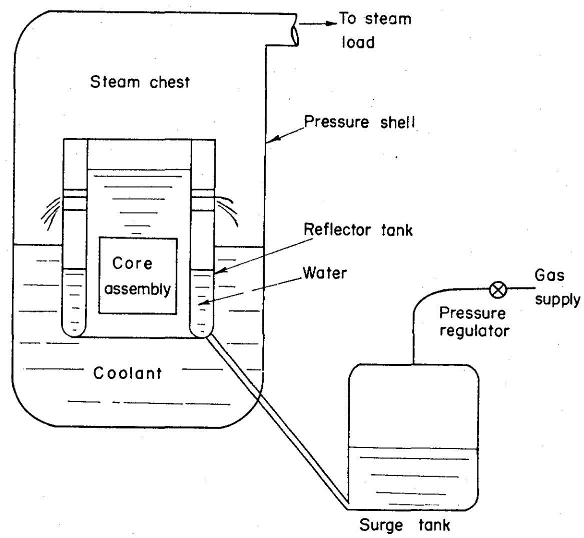


Fig. 1—Diagrammatic sketch of heterogeneous boiling reactor with reflector control.

A full-scale physical mock-up was constructed as shown in Fig. 2. This hydraulic simulator consists of a reflector tank, a surge tank, and a connecting pipe, together with pressure accumulator tanks coupled with compressors. Fig. 3 is a schematic of the hydraulic simulator.

The accumulator tanks (labeled *A* in Fig. 3) are 16 cubic feet ASME-approved 500-psi air pressure vessels each mounted above, and connected to, a 3-hp, 500-psi air compressor. These tanks provide 500-psi air to the reflector tank and surge tank as needed.

The surge tank (labeled *S* in Fig. 3) is a 36-inch diameter tank so constructed to allow for hydraulic coupling pipes up to 6 inches in diameter, and to have various connecting ports for mechanical control valves and relief valves. An inlet air pressure regulator is used to reduce 500-psi air from the accumulator tank to 300-302 psi in the surge tank. The outlet air pressure regulator is used to release air from the surge tank when the pressure increases above 295 psi in the surge tank.

The reflector section consists of the simulated reflector vessel (labeled *R* in Fig. 3), two pneumatic control valves with a controller, a capacitance-type water level indicator, and the necessary safety relief valves. The reflector tank has a cross-sectional area of five feet^2, and the water level can be raised two feet from the low portion without any interference from inlet air or water connections. One-half inch pneumatic control valves are used on the inlet and

* Work performed under AEC Contract W-7405-eng-92.
† Battelle Memorial Inst., Columbus, Ohio.

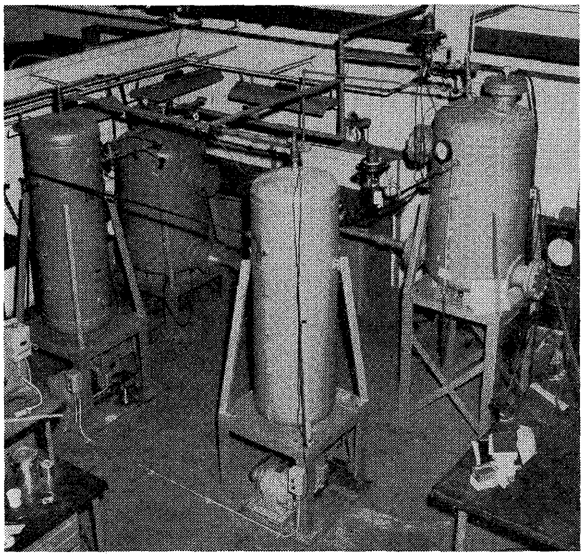


Fig. 2—Full-scale hydraulic-system mock-up.

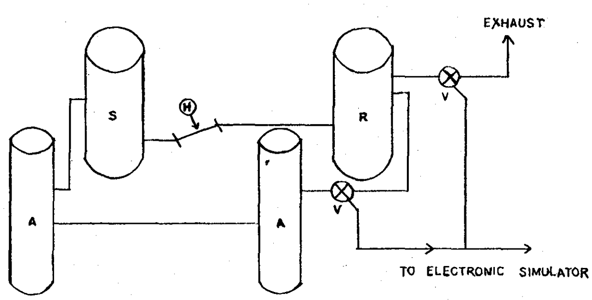


Fig. 3—Schematic of hydraulic simulator. R = simulated reflector vessel, S = surge tank, A = pressure vessels, H = hydraulic coupling, V = pressure control valves.

outlet lines to the reflector tank, and are both controlled by the same pneumatic signal from a pressure controller.

Safety relief valves are provided on each tank and are set at 505 psi on the accumulator tanks and 340 psi on the surge and reflector tanks.

The design parameter of the physical simulation is the hydraulic coupling (H in Fig. 3). It is necessary to install some sort of damping in this portion of the system to provide stable operation. The purpose of this investigation is to determine an acceptable means of damping this system.

An analog computer was used in the analysis and evaluation of this reactor. The computer was used to solve the equations describing the reactor kinetics, reflector reactivity, and steam pressure under various load conditions.

A standard group of nuclear kinetic equations were employed in the study of this reactor. These are

$$\frac{dP}{dt} = \left[\frac{(1-\beta)k - 1}{l}\right] P + \sum_{i=1}^{i=6} \lambda_i c_i + S_0$$

$$\frac{dc_i}{dt} = -\lambda_i c_i + \frac{\beta_i k}{l} P$$

where

P = reactor power, btu/sec,

$\beta = \sum_{i=1}^{i=6} \beta_i$,

β_i = fraction of neutrons produced each mean lifetime that are delayed in the ith group,

l = mean lifetime, 10^{-4} sec,

λ_i = decay constant for ith delay group, sec^{-1},

S_0 = term proportional to neutron source,

c_i = term proportional to concentration of ith delay group,

k = effective multiplication factor.

The Battelle analog facility has a self-contained "nuclear kinetic feedback unit" to solve these equations. The use of this unit requires only two operational amplifiers, and saves considerable setup time.

For the purpose of this evaluation, it was assumed that boiling commences at the point where the water temperature reaches the saturation temperature and increases, linearly, in intensity as the water temperature increases beyond this point. The rate of change of water temperature was computed as the difference between power produced by the reactor and power used to convert water to steam. From these relationships the rate of steam production was determined.

The rate of change of the weight of steam in the steam chest is proportional to the difference between the rate of steam production and the rate of steam used to satisfy the power demand. The pressure in the steam chest was determined from the weight of steam and the volume of the steam chest. This volume varies inversely as the height of the reflector, since an increase in the volume of water in the reflector leaves less volume to be occupied by the steam. A voltage proportional to this computed pressure was fed to the hydraulic mock-up as the pressure demand signal.

Two main factors affect δk in this system. These are reflector worth, which is a function of reflector height, and steam-void fraction, which is a function of power level and pressure. The functions used were obtained from experimental data.

The electronic and hydraulic portions of the system were then coupled together (as shown in Fig. 4) to complete the simulation. A pressure demand signal from the computer was used to determine the set point of the controller. The actual pressure in the simulated reflector tank was compared with the set-point pressure, and the error determined the pneumatic signal to the control valves. If the pressure in the reflector tank were lower than the demand pressure, the control valve to the accumulator would open to increase the pressure. Conversely, if the pressure in the reflector tank were too high, the control valve to the atmosphere would open to exhaust the pressure. The control valves were adjusted so that at set-point pressure both would be slightly open.

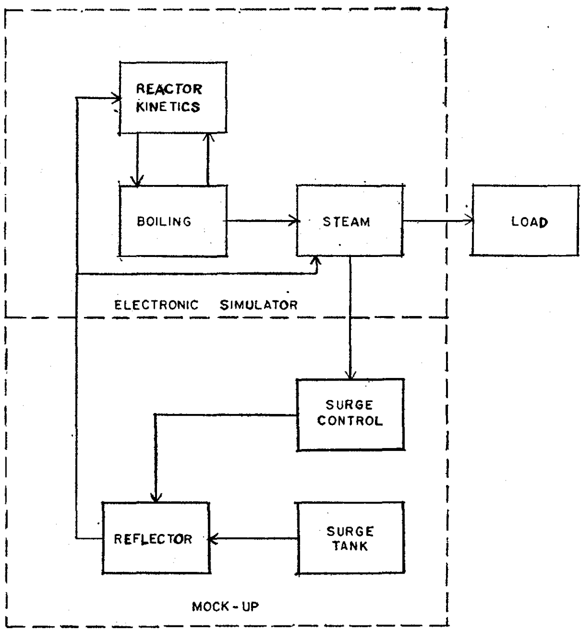


Fig. 4—Block diagram of physical simulation.

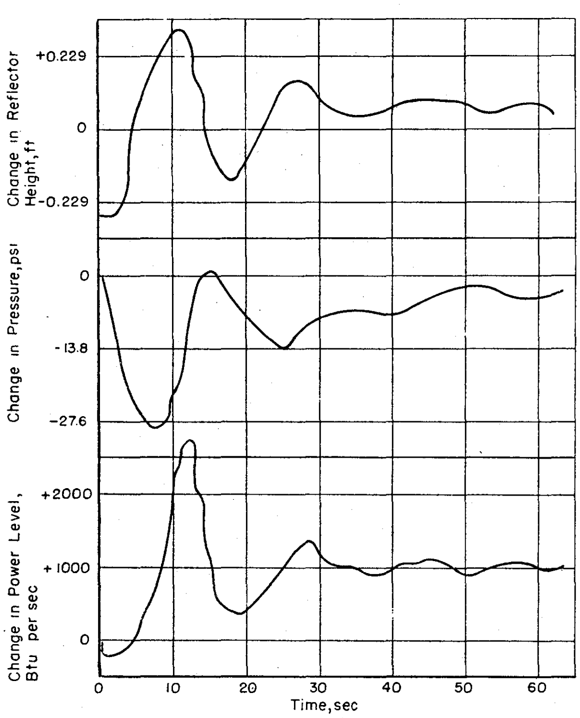


Fig. 5—Responses of undamped system.

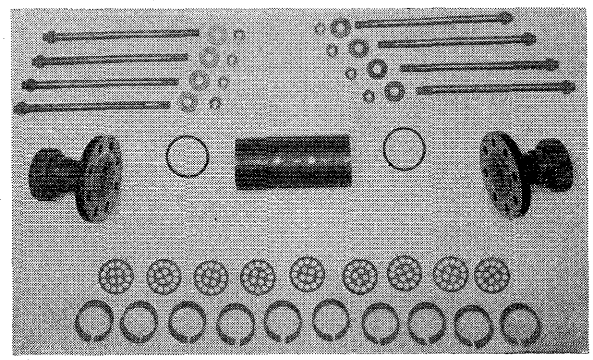


Fig. 6—Expanded view of the hydraulic coupling.

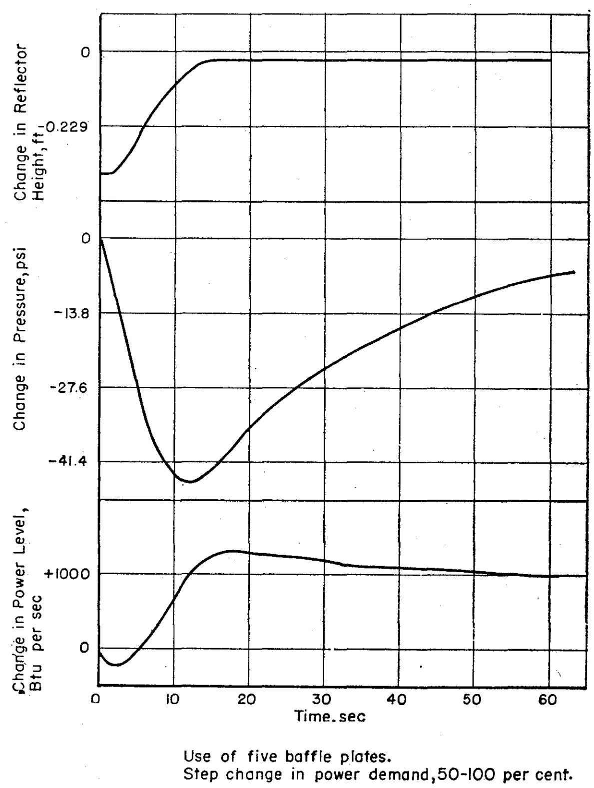


Fig. 7—Responses of damped system.

The pressure controller used in this simulation employed proportional-plus-reset (integral) type control. Both the proportional band and reset rate were set at their minimum values. In addition, because of an undesirable time lag between the pressure in the simulated reflector tank and the demand pressure, an anticipation circuit was included. The output of this circuit was added to the pressure demand signal to produce the controller set-point signal. This was effectively rate control. This circuit was adjusted for optimum response of the controller.

In order to complete the loop, a signal proportional to the height of the reflector had to be fed back to the computer. The water level was indicated by a capacitance-type height gauge. The output of this instrument was sent to an electronic recorder. The signal from a precision potenti-ometer geared to the recorder drive mechanism was used as the height indication required in the electronic simulation.

To examine this system, the simulated reactor is brought up to power manually. The power demand signal is adjusted to design point power. When the demand pressure to the simulated reflector reaches the operating level, the system is put on automatic control. The system thus far described tends to oscillate. Fig. 5 shows the responses of power, pressure, and reflector height for the undamped system.

To establish the required frictional forces for stable operation, the following configurations in the hydraulic coupling were attempted:

1) Various concentrations of steel wool,
2) Four 2-inch 90-degree elbows,
3) A 1-inch orifice in a 2-inch pipe,
4) A system of baffle plates.

Stable operation was achieved with the use of the baffle plates. This coupling is shown in an expanded view in Fig. 6. It consists of a section of 3-inch-ID tubing 9 inches long with inserts and spacers to damp the flow of water through the tube. The inserts are made of 14-gauge brass and have 16½-inch diameter holes drilled in each insert. These inserts can be placed in the coupling in various combinations of spacing up to 18 inserts.

The optimum responses occurred with the use of five inserts. These results are shown in Fig. 7.

With this stable system, the effects of changing the void coefficient and the incremental moderator worth of the reactor were examined.

Discussion

N. Irvine (Convair): Since you bring the system up to rated output manually, I take this to imply that control is most applicable over a limited range. What are the difficulties of control from start?

Mr. Boyd: Our study involved control over the operating ranges of 100 per cent of power to 10 per cent of power. However, when a nuclear reactor is brought up from zero power it becomes very important that this so-called start-up is very carefully handled.

We presented an illustration showing the nuclear kinetic equations which indicated that the rate of change of power was proportional to power. This was solved using a special unit involving only two operational amplifiers. Most engineers are familiar with the fact that two amplifiers in a loop will tend to be unstable, and consequently it's very easy for the output of this system to go exponential. This is true of reactors if there is too much disturbance or error in the initial start-up; where the power is very low, the reactor power could be exponential. This is considered in reactor technology as the period. It turns out that the period is nothing more than the amount of time it takes the reactor power to increase by a factor *e*. At very low power, a very short period could cause a reactor to go supercritical in a very short time.

Consider a period of half a second or even less, in which case an operator having to react to the situation might not be able to react fast enough to control the reactor. Consequently, at reactor start-up, which is specifically mentioned here, control is done manually. For our simulation, which is direct analog, the feedback unit will not operate at extremely low power levels because of this tendency for the power exponential.

O. Updike (University of Virginia, Charlottesville, Va.): When the reflector liquid leaves the reactor vessel, discharging to the surge tank, it is saturated and any lowering of temperature should cause some steam to "flash off." Could you go into more detail as to how this flashing was handled in the simulation?

Mr. Stone: In direct answer to the question, this condition was not considered in the simulation, so no detail could be gone into there. However, the liquid from the reactor does move out of the reactor toward the surge tank when the pressure in the reactor is above the equilibrium value, or when it has just risen from the condition at which it was being maintained to some higher pressure. This would imply first that the saturation temperature has gone up; then, that the water going from the reflector tank towards the surge tank is at a saturation temperature for the pressure. The drop in pressure, primarily in the pipeline, would perhaps be the result of 1) the change from a higher to a lower elevation and 2) velocity heads for the flow velocities. The resulting pressure changes in either case were not of a major amount in the pipe itself; consequently we did not feel that they constituted a problem.

The simulator was conducted with the amount in the pipe itself; consequently, we didn't have to consider flashing in the experiment itself.

H. T. DeFrancesco (Westinghouse Electric Corp., Baltimore, Md.): What amount of time was required in the study and programming phases of the simulation?

Mr. Gordon: Approximately three man months went into the study and programming phase of the simulation but by far the larger portion of that was in the study. The actual programming phase required about one man week.

Application of Computers to Automobile Control and Stability Problems

ROBERT H. KOHR†

CURRENT advances in automobile stability and control studies can be credited largely to modern electronic computers. Solution of the lateral control problem, like problems from other areas of automotive technology, has been deterred by the extreme complexity of the automobile. Some understanding of static or steady-state stability was obtained in the late 1930's, but the solution of dynamic response to steering input was not obtained until the advent of modern high-speed computing equipment.

The Automobile Stability and Control Problem

Stability and control of the automobile, sometimes called the handling problem, is concerned with providing an automobile with the proper steering behavior. It is really two problems: 1) providing a vehicle with directional sense so that it will run straight of its own accord, and 2) providing sufficient steering control so that the automobile may be easily steered along some desired path.

In its simplest sense, the study of automobile stability and control is the study of the lateral motions induced in a car by the steering inputs of the driver. The entire system of automobile steering response, shown in Fig. 1, consists of a driver, a steering gear, and an automobile. The first block is the driver whose information input sets the system in operation. This may be the desire to go straight, to pass another car, or to turn a corner. In response to the information input, the driver decides to do something and as a result applies a torque to the steering wheel which turns the steering wheel to a given position. This action, working through the steering gear, turns the car's front wheels to some angle and finally the car begins to change its path down the road. The automobile's change in path is called its lateral response. Besides the flow of effects forward from the operator to the lateral response, there are also several feedback loops. For example, the lateral response is fed back to the steering gear as a torque, and to the operator in the form of visual inputs and lateral acceleration. There is also steering-wheel torque feedback from the steering gear to the operator.

Although the human operator steers his automobile by a combination of steering torques and steering displacements, it is possible to study the responses to these two types of steering inputs separately. The stability and control characteristics associated with a fixed steering wheel or the response produced by a steering-wheel displacement are called the "fixed control" characteristics. Conversely, the characteristics associated with a free steering wheel, or the response produced by a steering torque, are called the "free control" characteristics. The remainder of this paper is concerned with the block labeled "automobile" and particularly its lateral fixed-control response to front-wheel steering inputs.

† General Motors Corp., Detroit, Mich.

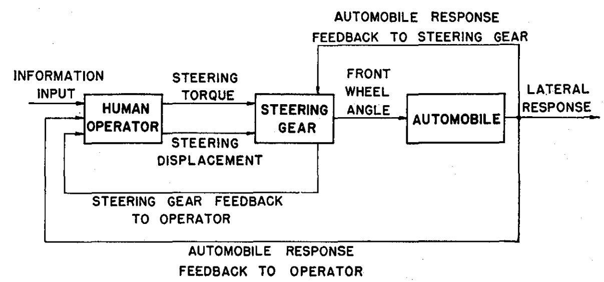


Fig. 1—Block diagram of car control system.

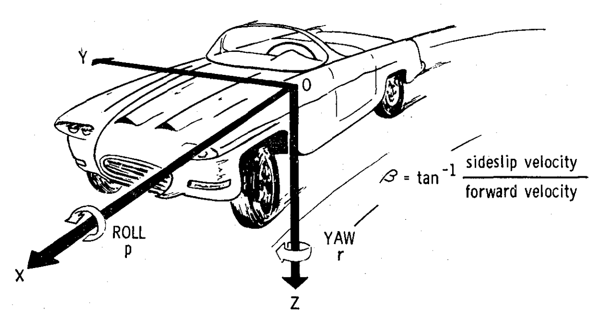


Fig. 2—Axis system for simplified automobile.

Development of Equations of Motion

The automobile motions, of concern in studying lateral response, are shown in Fig. 2. They are yawing, rolling, and sideslipping, and are defined as follows:

1) Yawing is the angular velocity about the vertical reference axis (OZ) and is denoted by the symbol r.
2) Rolling is the angular velocity about the fore-and-aft horizontal reference axis (OX) and is denoted by p.
3) Sideslipping pertains to the side velocity and is described by the sideslip angle β.

The axis system is effectively fixed in the unsprung mass (wheels and tires) so that Y and Z axes do not roll with the sprung mass (body, frame, and engine), but remain parallel and perpendicular to the road surface respectively. The car is considered to have a constant forward velocity along the X axis, and for the small angles of sideslip usually encountered, the sideslip angle β is defined as the side velocity along Y divided by the forward velocity along X.

A first attempt was made in 1953 to describe the lateral motions of a car by use of several differential equations.[1] Later that same year, General Motors enlisted the aid of the Cornell Aeronautical Laboratories. Cornell's extensive experience with stability and control problems in the aircraft field, particularly their advanced instrumentation techniques, was brought to bear upon this problem.[2] The result of this joint effort was a verified set of equations which describe a car's lateral response to steering inputs. The set of three simultaneous linear differential equations which represent a car's handling motions is shown in Fig. 3.

In each of these equations, the mass, inertia, and acceleration terms are given on the left, while the forces and moments that cause the acceleration are given on the right. The forcing terms on the right each consist of the product of some motion variable and a "stability derivative." For example, in the side-force equation, δ is the front-wheel steer angle put in by the driver, and Y_δ is the stability derivative with the dimensions of force per unit angle of front-wheel steer. The stability derivatives appear in each equation and are composed of various car parameters, like tire lateral stiffness, weight distribution, suspension characteristics, and various other terms. In all, there are 20 car parameters included in the equations of motion. In deriving these equations, it was assumed that the tires, springs, and shock absorbers all behave linearly and that the car is operating on a flat road with no wind blowing so that the only force input to the system is caused by the driver's turning of the front wheels of the car.

Measurement of Vehicle Parameters

The initial step in the experimental program consisted of determining the actual values of the mass, chassis, and tire characteristics of a particular automobile. The yawing moment of inertia, Iz, was measured by hanging the car on four cables and swinging it as a multifilar pendulum. By measuring the frequency of the yaw oscillation, it was possible to compute the yawing moment of inertia. The rolling moment of inertia, Ix, and the product of inertia linking the yawing and rolling motions, Ixz, were both determined by oscillating the car on knife edges about a horizontal axis. Ix was found from the rolling oscillation frequency and Ixz from the yawing moment produced by the rolling oscillation. The total weight and the longitudinal center of gravity locations were obtained by a simple weighing process.

Chassis characteristics, such as roll-spring rates, rear-axle roll steer, and front-wheel camber due to body roll, were determined by standard General Motors Proving Ground tests. The damping characteristics of the shock absorbers were determined with a stroking machine, and the damping produced by the shock absorbers in the suspension system became a simple geometrical calculation.

[1] R. Schilling, "Directional control of automobiles," *J. Indus. Math. Soc.*, vol. 4, pp. 64-77; 1953.

[2] W. F. Milliken, Jr., "Dynamic Stability and Control Research," Cornell Aeronautical Lab., Report No. CAL-39; 1951.

Side force equation

$$MV(\overset{\circ}{\beta} + r) + M_s h \overset{\circ}{p} = Y_\beta \beta + Y_r r + Y_\delta \delta + Y_\phi \phi$$

Yawing moment equation

$$I_z \overset{\circ}{r} - I_{xz} \overset{\circ}{p} = N_\beta \beta + N_r r + N_\delta \delta + N_\phi \phi$$

Rolling moment equation

$$I_x \overset{\circ}{p} + M_s h V(\overset{\circ}{\beta} + r) - I_{xz} \overset{\circ}{r} = L_p p + L_\phi \phi$$

Fig. 3—Lateral equations of automobile motion.

Variable of Motion	Sensing Instrument
1) Left front-wheel position, δ_L	Angular potentiometer
2) Right front-wheel position, δ_R	Angular potentiometer
3) Steering-wheel position, δ_{SW}	Angular potentiometer
4) Lateral acceleration, η_Y	Statham lateral accelerometer
5) Roll attitude, ϕ	Minneapolis-Honeywell attitude gyro
6) Pitch attitude, θ	Minneapolis-Honeywell attitude gyro
7) Angular yaw velocity, r	Doelcam rate gyro
8) Angular roll velocity, p	Doelcam rate gyro
9) Forward velocity, V	Fifth-wheel-generator set

Fig. 4—Measured variables and associated transducers.

Tire side force and moment characteristics were supplied by the tire manufacturer who obtained the data by running the tire on a moving drum. These tire characteristics were determined for wide variation in tire pressure and in the load carried by the tire.

In addition to providing numerical data to insert in the equations of motion, these tests demonstrated that the assumption of linearity for the various car parameters was valid for lateral motions of a reasonable magnitude.

Verification of the Equations

The equations of motion were verified by response tests made with an instrumented 1953 Buick. The instrumentation that was required to measure the car's lateral response is shown in Fig. 4. Both left and right front-wheel positions were measured and averaged to yield the effective front-wheel angle. Since it was not convenient to measure the sideslip angle, the total lateral acceleration along the Y axis was measured with a lateral accelerometer. The pitch attitude is not a lateral degree of freedom, but its measurement was made to determine whether any coupling occurred between the vertical and lateral motions. The outputs of the various motion-sensing instruments were recorded on an oscillograph.

In the early stages of the experimental work, it was assumed that information would be recorded in the frequency range of 0 to 10 cycles per second. After the first shakedown runs, it was discovered that both engine vibrations, and wheel vibrations at the natural frequency of the wheel on the tire, were being picked up by the various transducers. These unwanted vibrations were removed with low-pass filters utilizing a tuned galvanometer so that only frequencies from 0 to 3 cps were recorded.

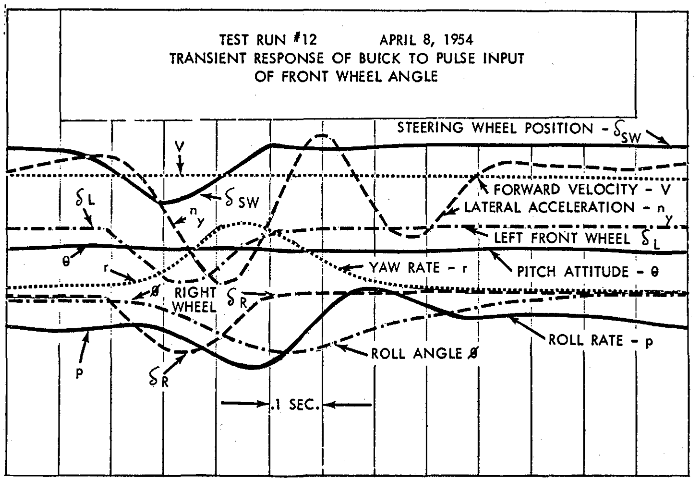


Fig. 5—Typical test response.

Response tests were conducted by stabilizing the car's forward speed, then performing a steering maneuver and recording the resulting car responses. The record of a typical test is shown in Fig. 5. This record shows the transient responses induced by a pulse steering input. All the motion variables are shown here. It is of interest to note that there was no appreciable change in the car's pitch attitude during this test. Numerous tests were made using both step and pulse steering inputs. The "step" and the "pulse" steering inputs used in this test work are only approximations to the mathematically pure step and pulse inputs used by servo engineers. The use of these two inputs with different harmonic content made certain that all of the frequencies of interest were introduced as inputs at some time during the experimental program. In addition to varying the inputs, response data were obtained in which the stability derivatives were varied from normal. With these response data in hand, a comparison was made between the actual measured responses and the responses predicted by the differential equations.

Comparison of Theory and Experiment

This comparison was made on the basis of frequency response, that is, the steady-state response of the automobile to a sinusoidal input of front-wheel steer angle. The theoretic frequency response was obtained by applying the Laplace transformation to the system of equations, and then replacing the Laplace operators by $j\omega$ where ω is the frequency of front-wheel oscillation and j is $\sqrt{-1}$.

The determination of yawing, rolling, and sideslipping frequency responses was then a matter of algebraic computation which was quickly accomplished by use of a digital computer.

The experimental frequency responses were determined from transient responses like the one shown in Fig. 5. This procedure is based on the use of the Fourier integral which, under certain conditions, enables a time function of a system $f(t)$ to be transformed into a complex frequency function $F(j\omega)$. The Fourier integral may be expressed as

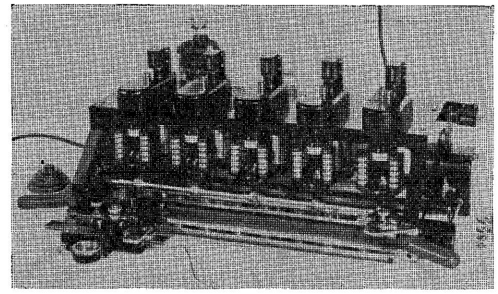


Fig. 6—Rolling sphere harmonic analyzer.

$$F(j\omega) = \int_0^\infty f(t)e^{-j\omega t}dt.$$

Since this integral must be evaluated along the interval from zero to infinity, it is necessary to know the behavior of $f(t)$ for an infinite time. This is most easily arranged by applying a disturbance to the system such that $f(t)$ reaches a steady value in some finite time, T. Under these conditions, the frequency function $F(j\omega)$ may be broken into its real and imaginary parts as:

$$F(j\omega) = R + jI$$

where

$$R = \int_0^T f(t) \cos \omega t \, dt - \frac{f_T}{\omega} \sin \omega T$$

$$I = -\int_0^T f(t) \sin \omega t \, dt - \frac{f_T}{\omega} \cos \omega T.$$

These integrals may be evaluated in a number of ways.[3] One convenient method utilizes the rolling-sphere harmonic analyzer shown in Fig. 6.

This method simply involves following the curve to be analyzed with a cross hair eyepiece that is attached to the analyzer. As the eyepiece is moved along the curve from the starting point (initial conditions zero) to the point where steady state is reached, it actuates a number of rolling ball integrators. Each integrator is equipped with a recording dial, and after the eyepiece has completed the traverse of the curve, the individual dials produce readings which are proportional to the real and imaginary components of the Fourier integral. The machine used in this work produces five harmonics for one traverse of the transient curve.

In order to obtain experimental response data for comparison, it was necessary to analyze the input to the system (steering angle), and the various system responses. Once this was done, a comparison was made between theoretic and experimental responses.

Fig. 7 shows the excellent agreement between the predicted yawing velocity response and that actually obtained on the road. Good agreement was also obtained for the rolling and sideslipping motion. It should be pointed out that the use of the frequency-response technique has two

[3] J. M. Eggleston and C. W. Mathews, "Applications of Several Methods for Determining Transfer Functions and Frequency Response of Aircraft from Flight Data," NACA Report 1204; 1954.

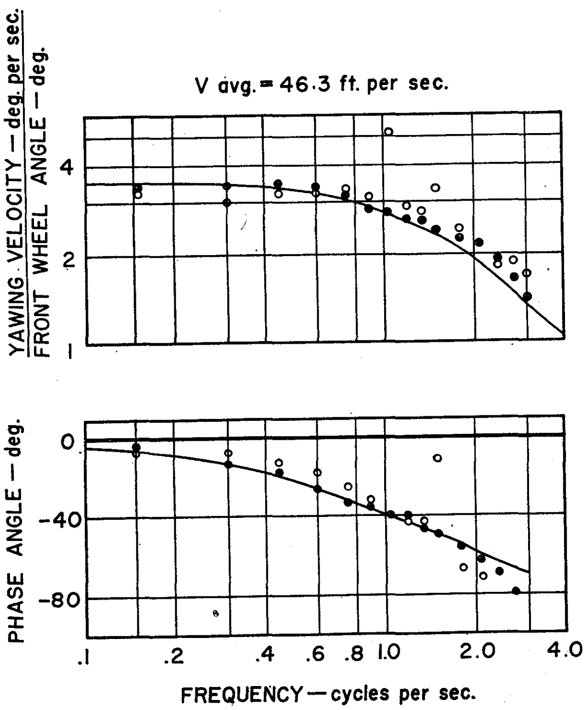


Fig. 7—Theoretic and experimental frequency-response, yawing velocity.

important advantages: 1) it provides for easy removal of dynamic effects produced by the filters in the recording channels, and 2) it provides a more general solution than does a transient response. It may be noted here that the first experimental frequency responses that were obtained did not match the equations exceptionally well, and, through use of the frequency-response plot, it was possible to determine some additional terms which were added to the equations of motion.

Computer Simulation of Vehicle Lateral Responses

When the equations of motion had been verified, they were then used to study the effects of the various car parameters on its lateral response. Both digital and analog computers have been used in this work. The analog computer has been used only to determine the transient response, while the digital computer has been used to determine the transient response, the frequency response, and the roots of the characteristic equation.

Analog Computer Studies

The mechanization of the differential equations on the analog computer follows the standard procedure of summing the various quantities which determine the various accelerations in the system, and then integrating acceleration to velocity and velocity to displacement.

The block diagram in Fig. 8 shows the general procedure that was used. Any of the motion variables, for example, the yaw acceleration r, is found by summing, horizontally, the products of the term in each box and the corresponding vertical input. The yaw acceleration is thus found as

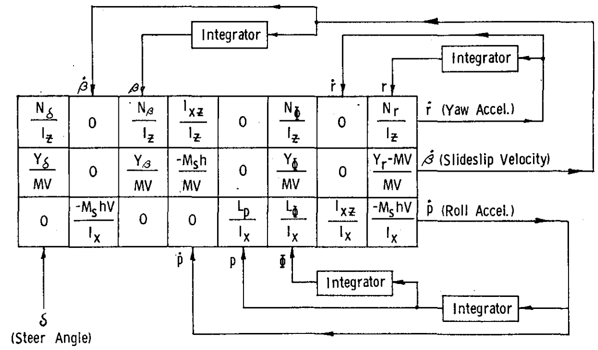


Fig. 8—Block diagram of automobile lateral-motion simulation.

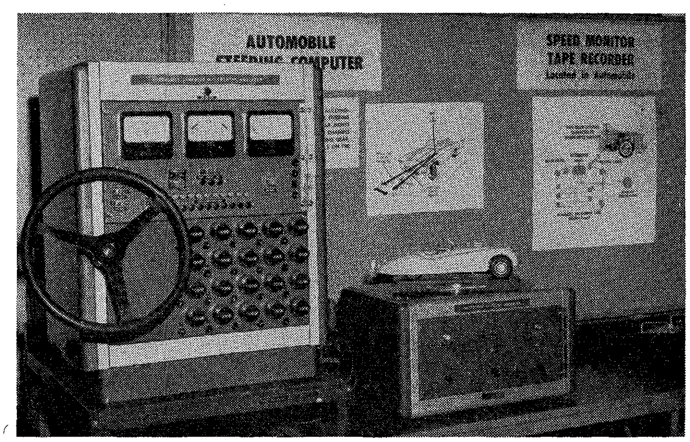


Fig. 9—Real-time automobile-handling simulator.

$$\dot{r} = \left(\frac{N_\delta}{I_z}\right)\delta + (o)\dot{\beta} + \left(\frac{N_\beta}{I_z}\right)\beta + \left(\frac{I_{xz}}{I_z}\right)\dot{p} + (o)p + \left(\frac{N_\phi}{I_z}\right)\phi + (o)\dot{r} + \left(\frac{N_r}{I_z}\right)r, \tag{1}$$

and is exactly the yawing moment equation that is given in Fig. 3. The rate of change of sideslip and the rolling acceleration are calculated in a similar manner. Each row then represents a summation, with the quantity in each block being the gain factor applied to the various motion variables.

The complete simulation of the equations on the analog computer requires only fourteen amplifiers. Although no nonlinear equipment is used at the present, future work will require the addition of some function generators and multipliers.

Real-Time Simulator

In addition to analog computer studies which are often run in "slow time," a real-time simulator has been of material value in demonstrating the lateral motions caused by steering inputs. This simulator, shown in Fig. 9, is composed of a small, special-purpose analog computer which can be "steered" by turning a steering wheel attached to the computer. This motion causes "steering angle" voltages to be introduced into the analog computer circuit. The computer then solves the lateral equations of motion of the automobile. The voltages proportional to

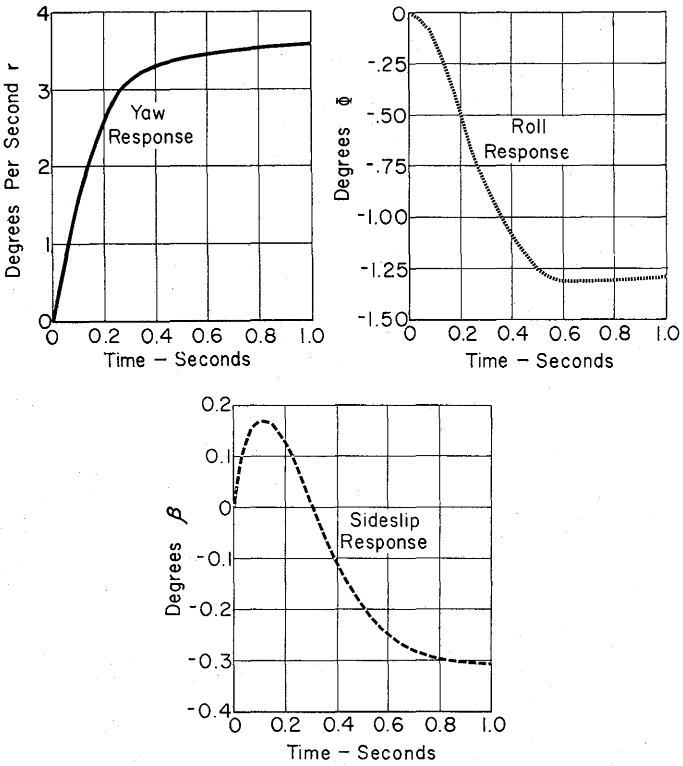


Fig. 10—Yaw, sideslip, and roll response of a 1953 Buick to a 1-degree step-steering input at 30 mph.

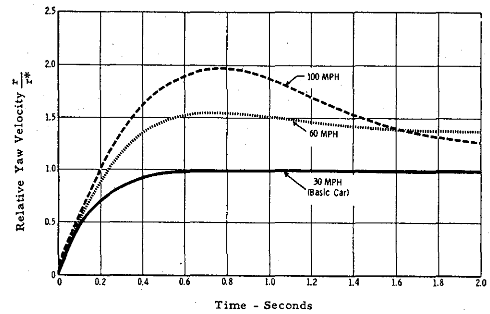


Fig. 11—Effect of speed on yaw.

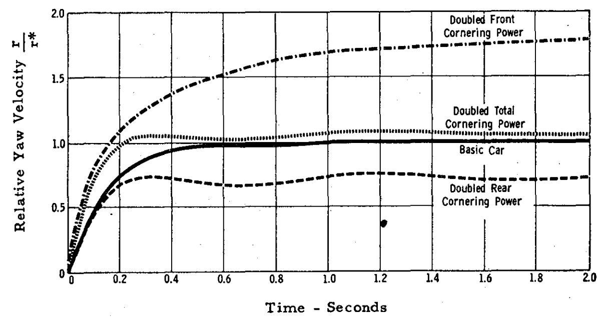


Fig. 12—Effect of tire cornering power on yaw.

yaw, roll, and sideslip are applied to a small servodriven model car. The model car can yaw and roll, and demonstrates sideslip by a pointer mounted on the hood to show the direction in which the car is actually going. The computer includes a "skid" meter, a "velocity" meter, and a meter which indicates the applied steer angle. By use of a ganged potentiometer which has an element in the input or feedback of certain of the amplifiers, it is possible to set the forward speed of the car at any point between 30 and 100 mph.

Digital Computer Studies

The digital computer was used to check the transient response obtained by the analog computer and also to obtain the frequency response and the roots of the characteristic equation of the automobile system. The transient response was determined by a Runge-Kutta method with values of the motion variables determined at each 1/40 of a second of problem time. The frequency response problem required the solution of three simultaneous, complex equations which result when the complex operator $j\omega$ is inserted in the equations. This solution utilized a matrix subroutine which converts a 3×3 complex matrix into a 6×6 real matrix. The roots of the characteristic equation were determined by first finding the characteristic equation by direct expansion and then solving the equation by a Newton-Raphson method.

A typical solution of the lateral response equations is shown in Fig. 10. The transient responses shown are the yaw, roll, and sideslip of a 1953 Buick traveling at 30 mph when the front wheels are steered suddenly to produce a step-steering input of 1 degree. The yaw and roll develop smoothly, but in the opposite sense, while the sideslip starts in one direction and then reverses. If these curves were normalized so that the steady state of each response tended toward unity with increasing time, the yaw would be shown clearly leading the roll and sideslip.

Fig. 11 shows the effect of increasing car speed on the lateral response. The curves in this figure have been normalized with respect to the steady-state yaw response of the car traveling at 30 mph. As the speed increases from 30 to 60 to 100 mph, the yaw response develops an overshoot. This is due to a decrease in the yaw damping of the tires as the car's speed increases.

One particularly important vehicle parameter is the lateral stiffness or "cornering power" of the tires. Fig. 12 shows the effect of changes in the tire cornering power on the yaw response. When the cornering power of the front tires is doubled, the car is placed in an "over steer" condition characterized by the large steady-state response which is reached quite slowly. Doubling the rear cornering power reduces the steady-state response of the system, while doubling both the front and rear cornering power produces a quicker response with little change in the steady-state value.

All the other parameters in the system have been explored and their effects on the various lateral responses have been cataloged.

Results of the Study

As a result of the extensive examination of the solutions of the equations of motion, an understanding of the effects of the various car parameters on the car's lateral response has been obtained. In addition to this somewhat specialized information, a number of general observations regarding the car's lateral response has been made.

The first effect observed pertains to the coupling of the lateral response motions. Fig. 13 shows the relative strength of the coupling between the three types of vehicle motion. Yaw and sideslip are strongly coupled, but there is only a weak coupling linking the roll to the yaw and sideslip. However, this weak coupling is often the reason that a particular automobile is stable or unstable, particularly in yaw.

A second interesting aspect of the automobile pertains to its modes of motion when it is in a state of free lateral oscillation. Determination of the modes of motion was made with the anticipation that there would be a well-defined "directional" mode of motion consisting of yawing and sideslipping, and also a "rolling" mode of motion. The modes were found by calculating the roots of the characteristic equation of the system, then inserting these roots in the lateral-motion equations. This study showed that there are, generally, no distinct modes of motion, but under certain conditions it is possible to find modes of motion which are primarily "directional" and "rolling" in nature.

A third interesting result of this work is that the equations of motion predict lateral-motion responses which are compatible with the on-the-road observations of the "handling" engineer. Early work in this field has often concerned itself with the so-called "oversteer" or "understeer" characteristics of a given car.[4] These characteristics are shown in Fig. 14. Because of the nature of the pneumatic tires, they must run at slip angles β_1 and β_2 in order to produce side forces. If β_1 is greater than β_2, the path followed by the car curves away from the side force. This is called understeer. Conversely, if β_2 is greater than β_1, the path of the car curves toward the side force and this is called oversteer. The directional stability of an understeering car tends to increase with increasing speed, while the

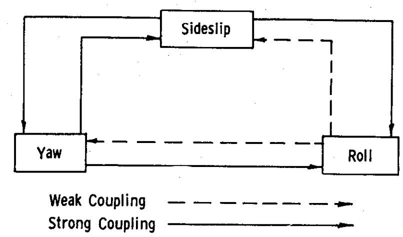


Fig. 13—Relative strength of coupling of automobile's motions.

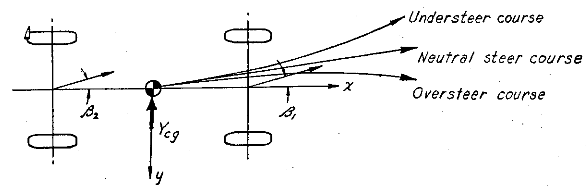


Fig. 14—Understeer and oversteer characteristics.

directional stability of the oversteering car tends to decrease. In fact, in the case of oversteer, the "handling" engineers found a critical speed at which the car becomes directionally unstable and at this speed only the most alert action of the driver will keep the car on the road. The equations of motion contain terms from which the "oversteer" or "understeer" characteristics may be computed, and also permit the calculation of a speed at which an "oversteer" car becomes unstable.

Conclusion

The result of this research program to date, then, consists of a verified linear model of the "automobile," shown in Fig. 1. This work represents only the first step in an over-all systems analysis of the general automobile-steering problem. In addition to a study of the nonlinearities encountered when the amplitudes of the lateral motion are large, further work will necessarily concern itself with the steering gear and the human operator. In the last analysis, the determination of desirable handling qualities, from the driver's point of view, is the most important result that will come from this research effort.

[4] M. Olley, "Road manners of the modern car," *Proc. Inst. Auto. Eng.*, vol. 15, no. 5, pp. 147-182; 1947.

Discussion

O. J. Osofsky (Republic Aviation Corp.): What are the criteria for entering a skid?

Mr. Kohr: A skid occurs when a tire is in a region where it won't develop any more side force no matter how much more you turn it. If you would plot a curve of cornering force vertically vs slip angle horizontally, the place where the curve peaks and gets flat is generally called the skid region.

Mr. Osofsky: Has any work been done on the motion of an automobile while in a skid?

Mr. Kohr: To the best of my knowledge, no, there has not. The real difficulty is in determining the transient which occurs in going from linear to nonlinear motion. If we knew the complete nonlinear equations of motion we could then predict what the car would do in the skid when the front or rear end broke away first or if they happened to skid simultaneously.

R. J. Mead (McDonnel Aircraft): Can tire "thump" and power-steering dead-zone effects be studied easily?

Mr. Kohr: Tire thump is an acoustic phenomenon which enters the body and which has apparently nothing to do with steering and control. Consequently, we have not considered it in our studies.

As far as power-steering dead-zone effects are concerned, I don't foresee any difficulty in simulating these on either the analog or digital computer, although we have not included any such effect to date.

H. F. Meissinger (Hughes Aircraft Co., Culver City, Calif.): Do you consider emergency handling problems? For example, torque required to hold the car on course under blowout, etc.

Mr. Kohr: As you may recall, the equations that I showed were those which pertain when the force put into the system is only that introduced by the driver, *i.e.*, there are no road forces considered. If we could describe the system mathematically, we might be able to study it. The way this is done currently is as follows. If the car is suspicious in terms of stability during blowouts, it is taken out on a test and a forced blowout is made at a time unknown to the driver. The car's motion and stability following the blowout is then observed.

An Analog-Digital Simulator for the Design and Improvement of Man-Machine Systems*

H. K. SKRAMSTAD†, A. A. ERNST†, AND J. P. NIGRO†

UNDER sponsorship of the Aero Medical Laboratory, Wright Air Development Center, the National Bureau of Standards is designing a simulator facility for research on man-machine systems. The facility is specifically intended to enable experimentation with man-machine control systems such as those for air-traffic control, ground control of interceptors, and command systems in general. The facility is to be equipped for the dynamic representation and simulation of control systems which have human operators as elements of the closed loop. The present work has included the construction of a prototype of the essential elements of a complete simulation facility. This paper presents some design considerations peculiar to this class of simulator, describes the present prototype, and indicates refinements which might profitably be incorporated in the final facility.

The prototype facility incorporates the basic capabilities required for the final simulation facility. The construction of the prototype was intended to serve three important purposes. The first was to deal with technical and economic problems which could be neither properly anticipated nor adequately resolved simply by means of a "paper study." The second was to achieve a better basis on which to specify the facility and estimate the costs of implementation, operation, maintenance, and expansion thereof. The third was to test the utility of the specified facility through application of the prototype to the resolution of one or more of the important issues being faced in current development of systems. The desired capabilities have been provided by combining analog and digital techniques in a manner intended to minimize the requirements for equipment. The equipment includes general-purpose computers, both analog and digital, which may be used independently for the more conventional kinds of simulation and for data reduction.

Simulators have become recognized as an essential tool for research on and development of automatic systems; and the present effort is directed toward extending the application of this tool for research and development of semi-automatic systems. A simulator which permits dynamic as well as operational analysis of a system enables prediction and optimization of the performance of that system in a laboratory. The alternative to simulation is to build the proposed system and carry out extensive testing and modification programs. In the case of complex man-machine systems, the latter procedure is generally prohibitively expensive and time-consuming, and the results are often inconclusive due to the inability to control important experimental conditions in the field. However, the problem of predicting human performance in man-machine systems confronts laboratory simulation as well as system design and field testing. Aside from human factors, the chief difficulty has been to obtain detailed and accurate mathematical statements of the system functions to be performed by the human operators. Simulation forces an objective and quantitative examination, in the laboratory, of the information flow between the man and the machine; it also requires that the human operator be placed in the closed loop if maximum confidence is to be placed in experimental results. This complication, together with the general complexity of the systems of interest, gives rise to a number of requirements which are peculiar to the design of a facility for the simulation of semi-automatic systems.

There are a number of distinctive requirements which characterize a simulator for research on man-machine systems and make it different from one designed for the study of automatic systems. The simulator must be capable of operating in real time, since human operators are to be included as parts of closed loops in the model of the system. Inclusion of the human operator also introduces requirements for coupling the operator to the simulator in such a manner that his performance will be comparable to that in the system being simulated. It is thus necessary to provide appropriate means for presenting information to the human operator and for the acceptance of his responses. Since the configurations of both displays and controls will be as variable as the operations to be studied, the general-purpose computers must be equipped for convenient connection to a variety of special-purpose equipment which more or less duplicates the operator's work space. Human variability precludes exact repeatability of measurements and requires extensive use of statistical methods in analyzing experimental results. Furthermore, special provisions must be made for the automatic handling and reduction of the large amount of data which are produced by statistical procedures. It is also desirable to incorporate equipment for effective qualitative monitoring of experiments so that exploratory or pilot runs may be employed to limit the amount of data which must be quantitatively analyzed. Since practical considerations limit

* This research was supported in whole or in part by the U. S. Air Force under Delivery Order No. (33-616) 56-10, monitored by Aero Medical Lab., Directorate of Res., Wright Air Dev. Center, Wright-Patterson AFB, Ohio.
† National Bureau of Standards, Washington, D.C.

the amount and capacity of simulation equipment, it is often necessary to partition a complex system and simulate only a portion thereof. Such partitioning must be accomplished so as to permit a valid integration of partial results. Since the human operator is extremely nonlinear, superposition does not apply. Hence, care must be taken, when partitioning the system, so that features of the human environment, not directly a part of the control loop under study but possibly affecting that loop significantly, are included in the simulation. Man-machine systems of any appreciable complexity usually include one or more sampled-data control loops; therefore, the facility must include the capability for simulating sampled-data systems. Finally, these requirements must be met while keeping the cost and complexity of the equipment at a minimum.

While the NBS staff has had considerable experience with the simulation of automatic control systems, it possessed but a superficial understanding of the principles and methods of experimental psychology which apply to the simulation of man-machine systems. Therefore, it was necessary to draw heavily upon the experience of groups such as the Psychology Branch of the Aero Medical Laboratory and the Engineering Psychology Branch of the Naval Research Laboratory. The assistance of these groups has been invaluable in converting the requirements of the bioscientists into specifications for simulation equipment. Their contributions are best manifested in the specialized operator work-spaces which have been constructed. Both an interceptor cockpit and an assembly of ground-display equipment incorporate a number of alternative techniques for presenting information to the operators and accepting their responses. The individual features of these work-spaces have been adopted from a number of different systems which are either presently in operation or in the final stages of development, and these features are among those which the psychologists feel to be especially significant. For example, it has been pointed out by the psychologists that the statistical deviations encountered in experimental psychology can be greatly reduced by having each subject serve as his own standard of reference. The accomplishment of this purpose in a system simulator requires the capability for rapidly changing system parameters which include the nature of operators' displays as well as the more conventional boundary conditions and stability parameters.

The foregoing requirements are reasonably well satisfied by the prototype which has been set up at the National Bureau of Standards. The effort to achieve maximum flexibility, economy, and convenience of operation has strongly influenced the functional organization of the simulator, the choices of the techniques employed, and the manner in which they have been integrated. The major pieces of the equipment which comprise the prototype can be functionally divided into five categories as follows: 1) general-purpose electronic computers which operate upon a mathematical model of the system under study; 2) operators' work-spaces which are appropriately fitted for integration of the man with the system being simulated; 3) equipment for centralized control and monitoring of the experiment; 4) equipment for automatic recording of data on the chosen evaluation criteria; 5) devices required for interconnecting the several major components of the prototype.

It was initially decided that both analog and digital computers would be required in order to handle effectively the mathematical models of complex systems. Considerable importance was attached to the apportionment of computing tasks between the analog and digital computers. It was believed that significant economies could be obtained with little loss of flexibility by arbitrarily exploiting the rather complementary advantages of the two types of computers. We have used analog computing equipment for those tasks which we believe can be handled best by analog computers, such as accepting continuous control information, solving complex dynamic equations in real time, and activating conventional display devices. The digital computer has been used for tasks which we believe can best be done by digital computers, such as the control of the experiment, the precise generation of open-loop data, calculations of high precision, handling of variables requiring large dynamic range, storage of data, and statistical analyses of the recorded data. Since the minimum sampling period which can be used in simulation cannot be shorter than the basic cycle of computation of the digital computer, that computer should not be used for calculations which can readily be done elsewhere in the simulator. Hence, all calculations involving the solution of differential equations in real time, such as the solution of the dynamic equations of motion of the airplane, are done on analog equipment. Thus, there is no need for approximating such equations by finite difference methods as would be required if calculated digitally. The requirements upon the speed and size of the digital computer and converter equipments are reduced enormously by confining the solution of differential equations to the analog computer. The price of this economy is the loss of the extreme digital accuracy. However, this loss is believed to be trivial, since there are very few applications which are known to require better than analog accuracy of the dynamic calculations, since the accuracy of solution has but a minor effect upon closed-loop performance.

The ground-controlled interceptor problem has been chosen for the first experiments, since it is an important example of a high-performance sampled-data system which employs human operators as components in a variety of control loops. A relatively comprehensive simulation of this problem has already afforded a fairly severe measure of the capabilities of the prototype. Indications are that the present facility has the capabilities required for the effective study of many other systems of interest, such as air-traffic control, ground-controlled approach and landing, weapon assignment, and certain classes of missile guidance. However, the specificity of the operator's work-

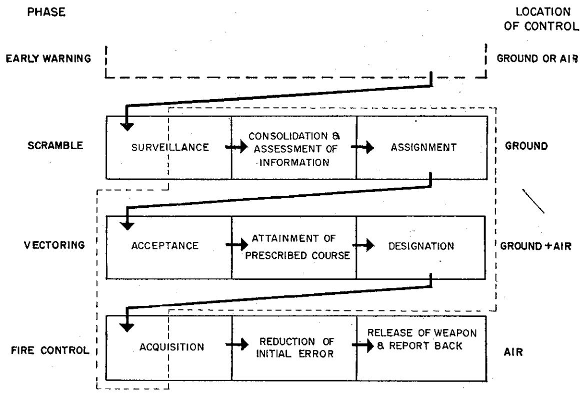


Fig. 1—Ground-controlled intercept system.

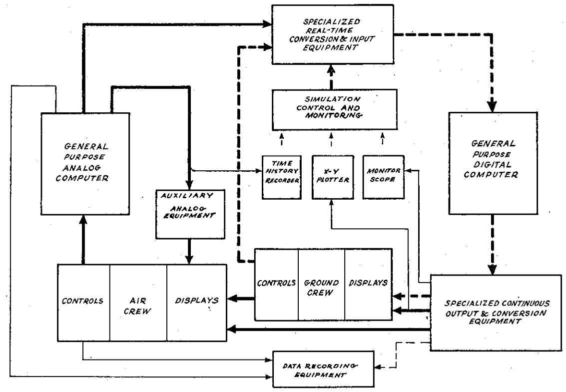


Fig. 2—Man-machine systems research simulator.

spaces limits the number of immediate applications of the present facility, and the ease of adaptation to other applications is closely related to the complexity required of the displays and controls to be added.

For our current experiments, we are simulating a class of systems which encompass a five hundred-mile-square air space populated by a number of target, interceptor, and civilian aircraft which are under the surveillance of search radars. In these systems, coordinate data from the radars are placed in an electronic store either automatically or by means of manual tracking. An operator then correlates these with aircraft identity and other descriptive data. Means are provided to enable him to evaluate the air situation and assign target-interceptor pairs. The assigned interceptor might be manually vectored by voice or data link, manually tracked and automatically vectored, or placed under fully automatic control of a ground computer. Ground-control instructions are presented to the pilot of a high-speed interceptor who controls his aircraft so as to arrive at the correct position and heading for attack upon the selected target. The ground-controlled interceptor system is shown diagrammatically in Fig. 1. The parts of the system that can be simulated on the experimental facility are outlined by the dotted lines in the figure. Following the arrows in the diagram, this includes surveillance, consolidation and assessment of information, assignment of interceptors to specific targets, the acceptance of the assignment by the interceptor, the interceptor pilot's attainment of the prescribed course, the designation of the target to the interceptor pilot and the acquisition of the target by the pilot.

The experimental setup is shown in block diagram form in Fig. 2. Dotted lines represent the flow of digital information and solid lines the flow of analog information. The system includes a general-purpose electronic analog computer, a general-purpose digital computer (SEAC), a ground crew with associated displays and controls, an air crew with its displays and controls, and specialized equipment for computer inputs and outputs, for control of experiments, and for recording of data.

First, let us consider the digital loop on the right side of the diagram. This includes the items connected by heavy dotted lines—the general-purpose digital computer, the specialized continuous output and conversion equipment, the ground crew with associated input displays and output controls, and the real-time conversion and input equipment. This loop carries all information going to and from the ground crew, with the exception of voice communication and analog vectoring information to the cockpit. Data on raid size and on the position, identity, size, speed, altitude, and combat status of all aircraft are stored in the digital-computer memory, one 44-bit word containing the complete set for each aircraft. The displays of the ground crew comprise two large cathode-ray oscillographs on which the aircraft-position pips are presented on a continuous synthetic PPI display, and an electronic status board on which the description of any aircraft may be displayed. The ground crew's control equipment includes a joy stick by means of which a small circle may be placed around any selected pip, a light pencil which may be placed upon any desired pip on the scope, a descriptor key set, and means for selecting the desired mode of operation. The ground crew's control console is shown in Figs. 3 and 4. In Fig. 3, the status board is seen between the scopes, the key set at the lower left, the joy stick at the lower right, and the selector switches in the center foreground. In Fig. 4, the light pencil is being held up to the master scope. Provision is made so that on the second scope, any desired part of the entire area displayed on the first or master scope can be magnified up to 32 times. The part magnified is selected by using the joy stick to move a circle of adjustable area about the master display. Information stored in the digital computer concerning the status of any aircraft can be presented on the status board by using the light pencil, the joy stick, or the flight number designator switches. The data thus displayed can be modified as required by means of the keyset, and the modified data can then be introduced into the digital computer by use of a master key. The keyset can be alternatively used to brighten the pips which represent aircraft falling within the designated ranges of altitude, speed, size or identity.

Let us now consider the analog loop at the left of the diagram of Fig. 2. This loop includes the general-purpose analog computer, auxiliary analog equipment, and the air

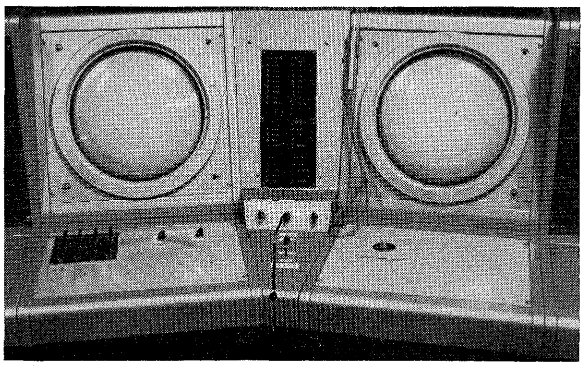


Fig. 3—Ground-control console.

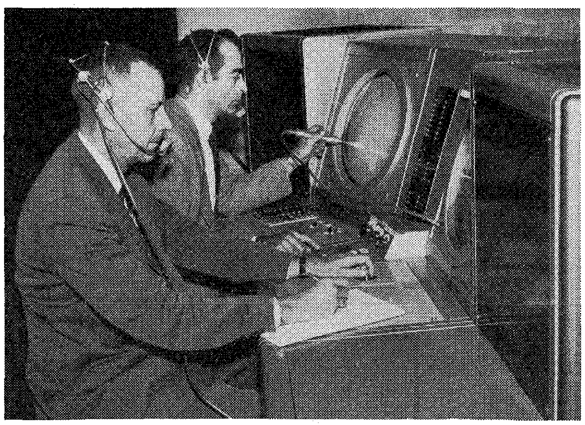


Fig. 4—Ground crew at console.

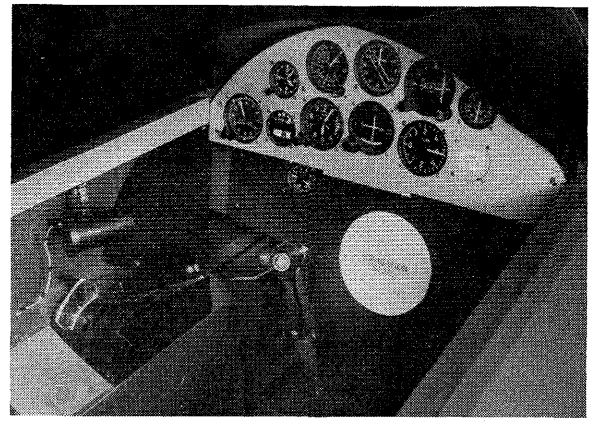


Fig. 5—Interceptor cockpit.

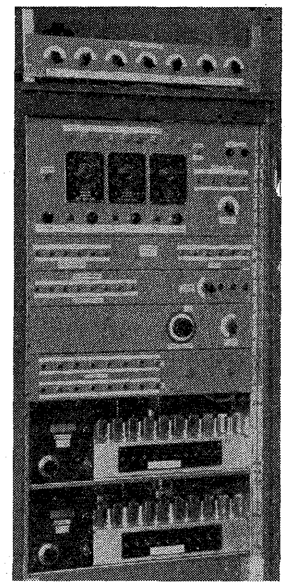


Fig. 6—Experimenters' control panel.

crew together with its input displays and output controls. The lower left-hand box represents the cockpit of a modern, high-speed, interceptor aircraft, with its instrument panel, joy stick, throttle, and rudder pedals. A view of the interior of the cockpit showing the instrument panel and controls is shown in Fig. 5. Displacement of the controls applies dc input voltages to the analog computer. The dynamical equations of motion of the aircraft are solved in the analog computer in real time. Voltage outputs proportional to such quantities as angle of roll, pitch, and rate of climb are converted by the auxiliary analog equipment to syncho rotations, which are used to drive the flight instruments in the cockpit. Both flight and navigation instruments are shown; the latter are used to present vectoring information from the ground control. Navigation information may be presented on either of the compasses, the ILS indicator, or upon the cathode-ray tube.

Let us now turn to the box in Fig. 2 labeled "simulation control and monitoring." The experiment is specified and modified according to the settings of a series of selector switches and knobs on a control panel. The settings of these switches determine such items as the radar scan period, the tactics, the mathematics of the intercept control computation, the navigation constants, the control lags, the limiting quality of data to be used in the control computation, and the data-smoothing procedures. A view of the experimenters' control panel is shown in Fig. 6.

Let us now consider the combined analog-digital loop in which the interceptor is controlled to arrive at the correct position and heading for attack upon the target. The velocity components of the interceptor aircraft obtained from the analog computer are converted to a ground-reference system, integrated over one scan period of the radar to yield incremental position components, and converted to digital form. A specialized equipment called a "format synthesizer" receives the following inputs in digital form: 1) the position increments of the interceptor; 2) the information stored in the positions of the switches and knobs on the experimenter's control panel; and 3) the output information from the ground crew, such as the positions of the joy stick and the switches in the key set. The format synthesizer organizes the data into computer words, and serializes them for direct entry into the digital computer. In the simulation of the ground-aircraft control loop, the digital computer has three jobs to do: 1) update the position of the controlled interceptor once during each sampling period of the system; 2) generate the trajectories of all preprogrammed aircraft; and 3) provide interceptor course data in simulation of the control computer.

Readout from the digital computer, SEAC, is concurrent with calculations and proceeds in an uninterrupted series of cycles. Output information is then routed to the various display devices where the selection of the desired information is accomplished. This readout procedure makes possible the display of information when desired without requiring special instructions for the digital computer. The specialized output equipment consists of a "staticizer" which is connected to twelve tanks of the acoustic memory in each of which are circulating eight words, each containing 44 binary bits which represent all of the data on one aircraft. Every ninth word is read out, which causes all words in the tank to be read out in succession at 432-microsecond intervals. At the option of the experimenter, the output cycle may be set to any number of tanks from one to twelve. The shortest output cycle will consist of 8 words repeated about 280 times per second, and the longest output cycle will consist of 96 words repeated approximately 24 times per second. In each word, two blocks of 10 bits each are connected, respectively, to two digital-to-analog converters. The resulting series of analog voltages go to a switching system which distributes them in a predetermined sequence among display equipments, plotting boards, and graphical recording equipment.

When the staticizer is reading out from that part of the memory which stores the position coordinates of aircraft, these voltages are connected to the plan-position indicators which display the location of all aircraft to the ground crew. The remaining 24 bits (those not connected to the digital-to-analog converters) contain the complete description of whatever aircraft is represented by the associated pair of coordinates. These bits are used for the selected gating of the aircraft to be displayed as well as for activating the electronic status board.

A study of the evaluation of the air situation and manual assignment of aircraft as interceptors or targets may be accomplished through use of the master key, while the equipment is alternatively switched between the "interrogation" and "assignment" options of operation. When the simulator is used for ground control of an interceptor which has been assigned for attack on a specific target, the digital computer operates on the position data of these aircraft, provides simulated degradation and smoothing, calculates the information necessary for attaining the prescribed course, and relays this information to the interceptor where it is displayed on the pilot's instrument panel. If manual tracking and automatic control is to be simulated, the ground crew manually tracks the target and the interceptor with the joy stick, and the digital computer uses this data to calculate the information necessary for attaining the course. The pilot receives the information directly from the computer as before.

The advantage of using a digital computer, when the dynamic range of the variable is large, is illustrated here. Assume that the analog voltage which represents the interceptor velocity is converted to a 10-bit code in which the least significant digit is made equal to one yard per second. With an interceptor velocity of 500 yards per second, a displacement of 1,000,000 yards or about 500 miles would be produced in about 33 minutes. To maintain the above resolution and permit this not unreasonable range, at least 20 bits must be provided in the position registers. It would not be possible to integrate over long periods of time to the resolution required here by means of electronic analog computers, because of noise and amplifier drifts.

Considerable attention was devoted to providing for convenient and effective utilization of the facility. One feature that has been emphasized is the ability to modify experimental procedures and vary system parameters rapidly and conveniently at the control panel. The experimenter, who may be either psychologist or engineer, does not have to be greatly concerned with programming difficulties, since changes in the experiment are made by turning knobs or throwing switches on the control panel which modify computer instructions, change system parameters, and control the selection of program subroutines. This avoids the need for frequent reprogramming of the digital computer, with the attendant experimental delays and costly programming staff.

Due to the variability of human performance, a facility intended for research on man-machine systems is likely to produce large quantities of experimental data. The provisions for data handling and data reduction require particular attention in order that an experimental investigation may be properly directed and that conclusions may be reached promptly. The requirements for data reduction may be divided into three categories: 1) that which must be performed concurrently with the experimental run to enable necessary monitoring and control; 2) that which should be accomplished between runs of a single experiment; and 3) that which is performed after the experimental investigation has been completed. It is desired to minimize the requirement for concurrent data reduction since the combining of simulation and data reduction would serve to increase substantially the requirements for size and speed of the computing equipment. Therefore, special provisions have been made for the recording of appropriate measurements as the experiments proceed. These data are recorded on magnetic tape in digital form. Data may thus be re-evaluated as new methods of analysis are developed, and the results may be compared with equivalent analysis of subsequent experiments.

For data in categories 2) and 3) above, experience has shown that the bottlenecks in the reduction process are due to computer input rates rather than to computation speed, since a large volume of data is produced which requires relatively little manipulation. In most computers, as with SEAC, provisions for reading from auxiliary storage are the fastest of the conventional input means. Among the available storage media, magnetic tape offers the advantage of being separable from the computer for purposes of both preparation and filing. Hence, the auxiliary magnetic-tape storage system of the computer was chosen as the input medium. The basic requirement is that optimum use of the tape be made, consistent with the tape-reading speed of the computer itself. It is also desirable that the data be organized on the tape in appropriate computer language so as to avoid the need for using computer time for data reorganization. Since both simulation input and direct recording must proceed concurrently, a second format synthesizer is used for the direct-recording system. Recorded data takes the form of information blocks which correspond to an 8-word tank, having capacity for 16 variables of 20 bits each. The recording frequency has been made variable from ½ to 32 cycles per second, in binary increments. It is expected that the computer will be programmed to call for one or more tanks of data at a time, but since the exact number cannot be predicted, space for stopping and restarting the tape is allowed between the recording of each tank of data. As a result, the information density is somewhat less than optimum, but still permits reading 20-bit data points into SEAC at the rate of 1200 points per second.

It should be re-emphasized that the configuration which this research tool might take ultimately should not be identified with the organization of the equipment which constitutes the prototype facility at NBS. We have been assembling only the essential elements of a facility in order to provide specific information about the functions and performance characteristics required in a versatile system simulator, to clarify some of the equipment problems relating to a general study of man-machine systems through the use of simulation, to enable a more judicious specification and a better estimate of cost, and to serve as the basis on which the detailed design and procurement of such a facility could proceed. Although we have used the ground-controlled intercept as an example, the facility is equally well adapted to other systems of interest, such as air-traffic control, missile assignment and guidance, and other complex systems involving human beings in the control loop. To simulate other systems, the basic equipment connecting the analog and digital computers would remain the same. The changes required would involve modification of the human operators' input displays and output controls, the patching of the analog computer, and the programming of the digital computer.

As has been indicated, one of the chief purposes for building a prototype simulator was to gain essential experience. Although the assessment of the prototype is not yet complete, some of the more important points thus far noted are discussed below.

The analog computer which is incorporated in the prototype is relatively modest. However, it is adequate since it is believed to be both appropriate and sufficient to limit the dynamic simulation to the flight realm which is pertinent to the control system under examination. While it is always desirable to have plenty of computing units, it is of particular interest to note the kinds of computing elements needed for the present class of simulation and the minimum number of each of such elements:

Operational amplifiers and integrators	50
Multipliers	15
Servoresolvers and nonlinear units	6
Electronic function generators	10
DC voltage to synchro output	12 or more

It might be noted that the number of synchro outputs is governed largely by the display requirements in the operator's work-spaces.

The SEAC has been found to have sufficient operating speed and memory capacity for a rather wide range of simulator applications. However, it must be remembered that this capability was achieved only by very circumspect use of the digital computer in combination with the analog computer. It is believed that a computer with somewhat less capability than SEAC could be profitably employed for system simulation; however, it presently appears that a well-balanced facility would incorporate a digital computer with substantially greater capability than SEAC. It is preferable that the computer employ binary arithmetic; have a word length of about 36 bits; and be equipped for completely concurrent input as well as output.

The concurrent output presently employed in the prototype appears to be very satisfactory in concept. The principle of having each display console set up to take only the desired information from an endless-belt flow of data is in contrast to the principle of externally directing the computer to supply the specified data. The endless-belt approach enables the modular addition of display consoles to an output "bus bar" of uniformly high output efficiency. Implementation of this approach, however, requires that each such console be equipped with its own selection logic and also requires that more bits of coded data be available in each of the output sets. An output set of two 36-bit words would be preferred to the single 44-bit word available from SEAC.

Perhaps the greatest room for improvement exists in the present provisions for real-time input to the digital computer. Before discussing this matter, it should be noted that there appears to be no real need for the digital computer to be able to read from the memory locations where output data is stored, and, conversely, no need for the computer to be able to enter data into the storage reserved for real-time input. These conditions greatly facilitate the provision of means for truly concurrent input and output. The present input equipment enters data through the regular input-shift register of the SEAC and is therefore not concurrent. This circumstance arose because it was expedient to adapt the solution of the direct recording problem to the real-time input problem. The chief shortcomings are that all input rates are slaved to the selected system sampling rate without means for aperiodic or sporadic input, which is very costly of time. It would be preferable to equip the digital computer with a static store which has a capacity of about 1000 bits, into which data may be entered by means outside the computer (either manual or automatic) and which the computer reads in parallel just as an equivalent amount of electrostatic or magnetic core memory.

In addition to changes of equipment, there are a number of refinements which can be made in the present procedures for using these equipments and for programming both the analog and digital computers. Since the optimum choice of procedures is strongly influenced by the nature of the equipment employed, a discussion of procedural refinements is not felt to be worthwhile at this time.

In closing, it should be observed that the suggested improvements must be considered as being tentative until more experience is gained with the prototype and until the preferred means of implementing the facility can be more definitively specified.

ACKNOWLEDGMENT

The authors wish to acknowledge the assistance of Leonard Taback, J. Howard Wright, Jr., Emile S. Sherrard, David Friedman, and many others of the staff of the Data Processing Systems Division for their contributions to the design and fabrication of the prototype facility. They also wish to acknowledge the assistance and guidance of Lt. Richard L. Deininger of the Aero Medical Laboratory, WADC.

Discussion

W. McLean (North American Aviation): Please list reports or references describing hardware use (analog-to-digital conversion, etc.) and results obtained to date.

Dr. Skramstad: There have been no reports published as yet describing the hardware used. However, it will be described in a future report that will be submitted to the Wright Air Development Center.

S. Kwiatkowski (Avro Aircraft): In aircraft application, how many channels of analog-to-digital conversions and vice versa were required?

Dr. Skramstad: We have two channels for analog-to-digital and two channels for digital-to-analog conversion.

Mr. Kwiatkowski: How much time was spent in making the installation operational?

Dr. Skramstad: The equipment is now all built and most of the components are operating, but the installation is not yet operational. We are still in the debugging stage. We hope that it will be in operation very shortly.

Facilities and Instrumentation Required for Real-Time Simulation Involving System Hardware

A. J. THIBERVILLE†

THE design and testing of the aeroelastic flight control system for the B-58 has been accomplished with the aid of numerous studies conducted on the analog computers at Convair, Fort Worth, Texas. These studies have evolved from simple two-degree-of-freedom perturbation equations with simulated autopilots to the full-scale longitudinal and lateral total equations tied into the actual flight control system used in the airplane itself.

In this presentation, the problems involved in testing the flight control system in the longitudinal mode are considered. The analog computer was used to solve the airframe equations of motion and to simulate any needed elements of the autopilot and power control system for which actual hardware was not available.

The need for this study manifested itself in a threefold manner:

1) To completely test and wring out, under all conditions, the flight control system to be used in the airplane,
2) To ascertain the degree of control that a pilot would possess under adverse conditions, and
3) To familiarize the flight crews with the handling and instrumentation of the ship in preparation for actual flight.

The satisfaction of each of these three needs has been realized and is responsible in part for the success of the flight test program of the B-58.

The discussion will be begun by presenting a flow chart or block diagram of the different components considered in the study.

First, the purpose and operation of each block will be described in general, and then some details will be examined by discussion of the associated equipment and the listing of individual problems connected with each.

† Convair, Fort Worth, Texas.

A schematic drawing of the components and plan of intelligence used in the autopilot system test is shown in Fig. 1. Except for the air data computer, which was simulated by the analog equipment, the autopilot existed and operated as it would in actual flight. The central unit assembly, the heart of the system, correlated incoming commands, guidance, and reference signals and, by proper channeling to the power control system units, accomplished control of the airplane.

The power control linkage package responded to commands from the pilot and autopilot and effected corresponding motion of the control surfaces. In this instance, the control surfaces, which are called elevons in a delta wing airplane, act only as elevators.

The cockpit simulator, naturally enough, was used to house the pilot, his flight instruments, and controls.

Incidentally, the pilot was by no means excluded from this study. He was considered just as much a part of the complete system as any other piece of equipment listed and might as well have had a response study run on him which would have shown accuracy, resolution, and tolerance values equally as well as any of the studies seen on any of the predominating servos. He was the unit completing the closed loop system.

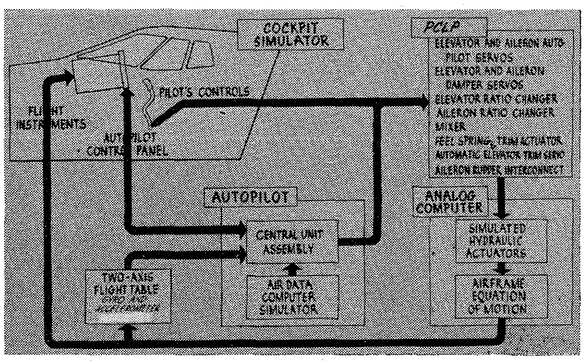


Fig. 1—Autopilot system's test.

The two-axis flight table, driven by Θ, the pitch angle of the airframe, supported the flight control system's gyro and accelerometer packet which then in turn fed the autopilot.

The analog computer consisted, with the exception of twenty-four channels of Reeves diode function generators, mainly of Electronic Associates' 16-31-R equipment.

The following equipment was used:

- 193 amplifiers
- 227 potentiometers
- 11 servos
- 10 electronic multipliers
- 6 resolvers
- 24 diode function generators
- 7 relays
- 4 recorders
- 16 diode limiters
- 2 plotting boards.

A study of the equations of motion of the airframe will serve as an aid in examining the details of these various components.

In Fig. 2, in addition to the auxiliary equations, the three main longitudinal equations are presented, namely, lift, drag, and pitching moment, from which the angle of attack, the velocity or Mach number, the pitch angle, and the altitude can be obtained. Because these are total equations containing the nonlinear coefficients which vary with flight conditions, *i.e.*, Mach number, altitude, angle of attack, etc., the representation of the airframe could be flown through the full range of speeds and altitudes for which it was designed.

$$\text{Lift: } \dot{\alpha} = \dot{\theta} - q\frac{s}{mV}\{C_L + C_T\sin(\alpha+\epsilon)\} + \frac{g}{V}\cos(\theta-\alpha)$$

$$\text{Drag: } \dot{u} = -q\frac{s}{m}\left\{C_{D_{\min}} + K_\delta\delta_e(\delta_e + 2\Delta\delta_e) + \frac{C_{D_L}}{C_L{}^2}\left[C_L - \Delta C_L + \frac{\partial\Delta C_L}{\partial\delta_e}\delta_e\right]^2 - C_T\cos(\alpha+\epsilon)\right\} - g\sin(\theta-\alpha)$$

$$\text{Pitch Moment: } \ddot{\theta} = q\frac{s\bar{c}}{I_{YY}}\left\{C_{m_0} + C_{m\alpha(\bar{c}/4)}(\alpha-\alpha_{L_0}) + C_{m\delta_e(\bar{c}/4)}\delta_e + \delta_{RS}C_{m\delta_{RS}(\bar{c}/4)} + C_L(CG-.25) + C_T\frac{d}{\bar{c}}\right.$$

$$\left. + \left[C_{m\dot{\alpha}(\bar{c}/4)} + \frac{\partial C_{m\dot{\alpha}}}{\partial CG}(CG-.25)\right]\frac{\bar{c}}{2V}\dot{\alpha} + \left[C_{mq(\bar{c}/4)} + \frac{\partial C_{mq}}{\partial CG}(CG-.25)\right]\frac{\bar{c}}{2V}\dot{\theta}\right]$$

AUXILIARY EQUATIONS

$$\text{Lift Coefficient: } \sum C_L = C_{L\alpha}(\alpha-\alpha_{L_0}) + C_{L\delta_e}\delta_e + C_{L\delta_{RS}}\delta_{RS}$$

$$\text{Normal Acceleration: } n_a - 1 = q\frac{s}{w}[C_L + C_T\sin(\alpha+\epsilon)] - (X_a - CG)\frac{\bar{c}}{g}\ddot{\theta} - 1$$

$$\text{Hinge Moment: } HM_e = 2M_{AC}q[C_{H_0} + C_{H\alpha}\alpha + C_{H\delta_e}\delta_e]$$

$$\text{Thrust: } T = [T_{\text{MIL}_{SL}} - \Delta T_{\text{MIL}_h}][\text{M.F.}T_{\text{RPU}} + T_{\text{NPU}}]$$

$$\text{Indicated Velocity: } V_{\text{IND}} = V\left[1 + \frac{h}{300{,}000}\right]\sigma^{1/2}$$

$$q = 1/2\rho V^2 \qquad M = \frac{V}{a} \qquad T = qsC_T$$

Fig. 2—B-58 longitudinal total equations.

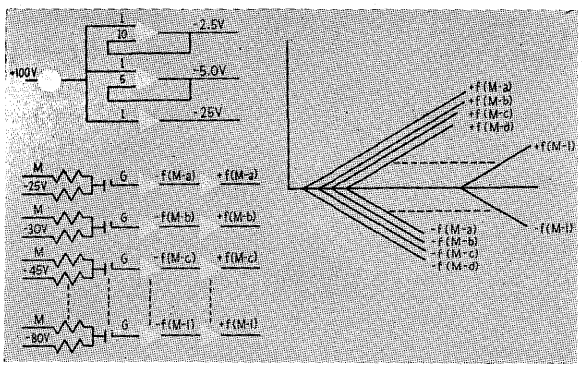


Fig. 3—Fixed break point method.

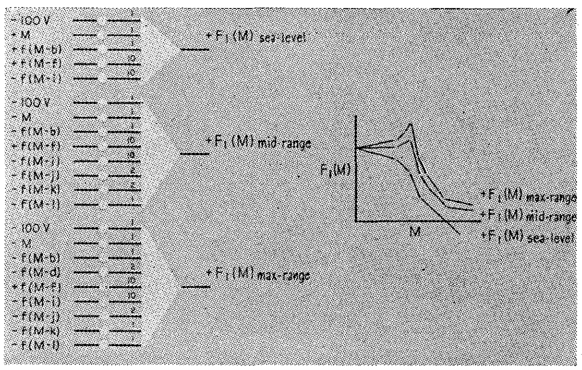


Fig. 4—$F_1(M)$.

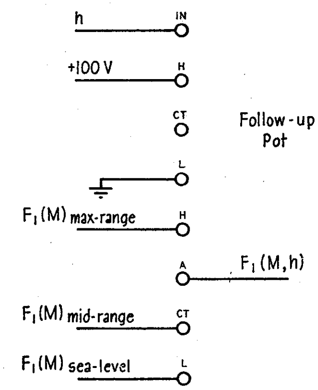


Fig. 5—$F_1(M,h)$.

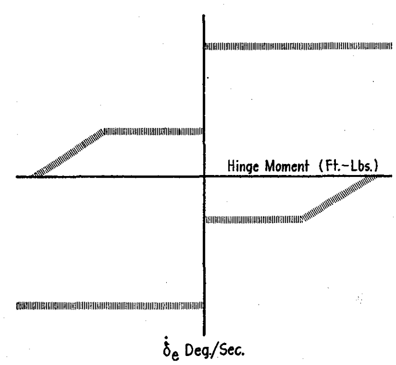


Fig. 6—Elevon rate limit.

To do so, it was necessary to generate some 31 nonlinear functions, eight of which were functions of two variables so that a total of 43 generators was needed. Because only 24 channels of diode function generators were accessible, 19 coefficients which were functions of Mach number were selected and simulated with what is known as the fixed break point method, as shown in Fig. 3.

This simulation was effected by selecting the group of functions which needed break points at a minimum number of points within the Mach range. By using resistors, diodes, and amplifiers, these individual break points, with both positive and negative constant slopes, were then generated on the computer patch board.

All of the break points, either positive or negative, needed to constitute the individual functions were then gathered together on amplifiers, as shown in Fig. 4, by using gain pots to adjust the slopes.

This method proved to be expensive in pots, amplifiers, and time but entirely adequate to fulfill the need since other means were not available.

To extend nonlinear coefficients to functions of both Mach number and altitude, the assumption was made that these coefficients were linear, in altitude, over two ranges; that is, from sea level to some midrange and then again from this midrange to the maximum ceiling. This approach led to three functions, one for each altitude, which varied with Mach number alone.

These functions were then placed respectively on the low, center tap, and high end of a multiplying pot of a servo, as shown in Fig. 5. The servo, driven by (h), the altitude, then produced the function $f\ (M,h)$ on the arm of the multiplying pot. By having three multiplying pots on each servo, all eight functions of two variables on three servos were accommodated.

An elevon rate limit was needed, which, as can be seen in Fig. 6, is somewhat of an awkward constraint to be imposed on the elevon rate since it is a function of the hinge moment as well as the direction of motion.

This limiting rate was accomplished by a circuit consisting mainly of diodes, pots, and high-gain amplifiers (Fig. 7). Its operation depended upon two general parts. One was a pair of electrically symmetrical function generators, which generated the elevon rate limit as a function of hinge moment. The other was a pair of symmetric absolute value summing amplifiers. The inputs to these summers were the rate limits, the elevon commands, and a feedback from the time lag. These signals caused the outputs to be the difference between the rate and rate limit when the limit was exceeded. This difference was then fed into the time lag out of phase with the rate and therefore subtracted from the rate the amount that the limit was exceeded.

As long as the rate was within its limits, the outputs remained at zero and thus allowed the δ_e command to be affected only by the time lag.

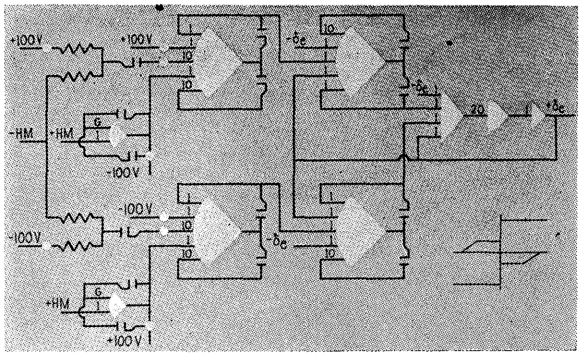


Fig. 7—Elevon rate limit circuit.

Ordinarily, diode limiters tend to "give" when hit by a large voltage. This circuit did not depend upon this type of limiting but rather upon the cancellation of one voltage by another. This cancellation resulted in sharp limits with no "giving" or leaking. When trouble did occur, it was usually immediately obvious because the high-gain amplifiers became overloaded.

For the most part, the rest of the equations were handled on the computer in a fairly routine manner. Except for provisions which allowed the operators to drop the pod from the plane, lower landing gears, shift the cg position, and those stated previously, reliance was mainly placed on dependable and familiar methods.

In tying analog equipment to the hardware, some difficulties of instrumentation were encountered since the flight control system accepted a 400-cycle amplitude modulated signal rather than a dc signal. These inputs were shown in Fig. 1 as the information from the air data computer to the central unit assembly.

After some experience with the difficulties involved with modulators was gained, a method was used in which transmitting synchros, excited with a 400-cycle signal, were coupled directly to dc resolvers. With proper scaling, the computer variables were made to drive the resolvers and thereby position the flight system's synchros.

The information from the power control linkage package was taken from the arms of dc excited Helipots, which were coupled to the hardware, and fed into the computer as analog voltages.

These signals consisted of:

1) Right and left elevon position,
2) Throttle and throttle servo output,
3) Resolution surface output,
4) Stick position.

Tying the computer to the cockpit simulator presented a variety of problems; however, their solutions were all provided for in the simulator's original design. Actual instruments or close approximations to the instruments were installed on the flight panel in order to give the pilot as realistic a presentation of his flight information as possible.

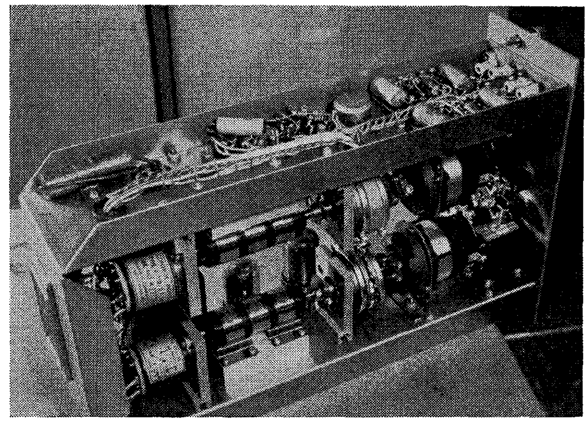


Fig. 8—Cockpit servo unit.

Some indicators were simulated simply by altering the face of a microammeter. Instruments having a "long scale," 250-degree rotation of the needle were preferred. Those used in this project were Hickok 500 microammeter, some with zero at the left and some with zero at the center.

The artificial horizon was especially complex and presented special problems. Because no way was devised to simulate this instrument, a real artificial horizon was purchased from Lear, Inc., and adapted for the specific usage.

Nine of the instruments chosen for the pilot's panel were driven by synchro repeaters (Fig. 8). Servo units, driven by signals from the analog computer, were built to position a synchro transitter, which in turn controlled the cockpit instrument, a synchro repeater.

Components were selected to give adequate performance at a reasonable cost. The Kearfott R-110 motor was chosen because it was powerful enough and quite small, although not small enough to be in the expensive miniature class.

All servos, except one, the altimeter, revolve one revolution or less, so one-turn follow-up pots were used. A type-J Helipot with its shaft extending front and back was selected. The motor, gear box, follow-up pot, and synchro were mounted colinearly to save space, to reduce cost, and to improve accuracy. If it had not been possible to arrange these components in a line, it would have been necessary to use idler gears to transmit torque around 90-degree corners, thus losing accuracy and space and increasing cost.

Magnetic amplifiers were used for power amplification to drive the motor since they are compact, efficient, and inexpensive. They controlled the flow of power from the power line instead of the flow being furnished from an expensive rectified power supply and thereby sharply reduced the necessary size of the dc power supplies. The Kearfott R-601 magnetic amplifier was chosen because of its small size, simple control circuits, and low cost. Its response was good up to 20 cps, which was sufficient for the instruments. Also, this cutoff "corner" frequency of 120 radians was safely remote from the motor cutoff frequency of 20 radians so that no unusual stability problems resulted from the use of this amplifier.

A two-stage dc preamplifier was used ahead of the magnetic amplifier. A summing circuit compared the inputs with the outputs and produced an error voltage, which was then amplified.

The Summers Model 85A flight table was, as shown in Fig. 9, a two-axis, electromechanical system which simulated airframe motion on a one-to-one time scale. The equipment consisted of a control console and a two-axis flight table, which contained two independently controlled platforms, each driven by counter rotating electric motors.

The table could accommodate a 40-pound payload with a maximum height of 11 inches. It was capable of unlimited angular displacement in both yaw and roll and could attain an angular velocity of 5 radians per second in yaw and 6 radians per second in roll. The angular acceleration attainable for yaw and roll was 50 radians per second squared and 10 radians per second squared, respectively. It proved adequate in every respect, except for the fact that it was capable of but two degrees of freedom rather than three. Had it been a three-degree-of-freedom table, the study would undoubtedly have been extended to the combination of both lateral and longitudinal equations of motion.

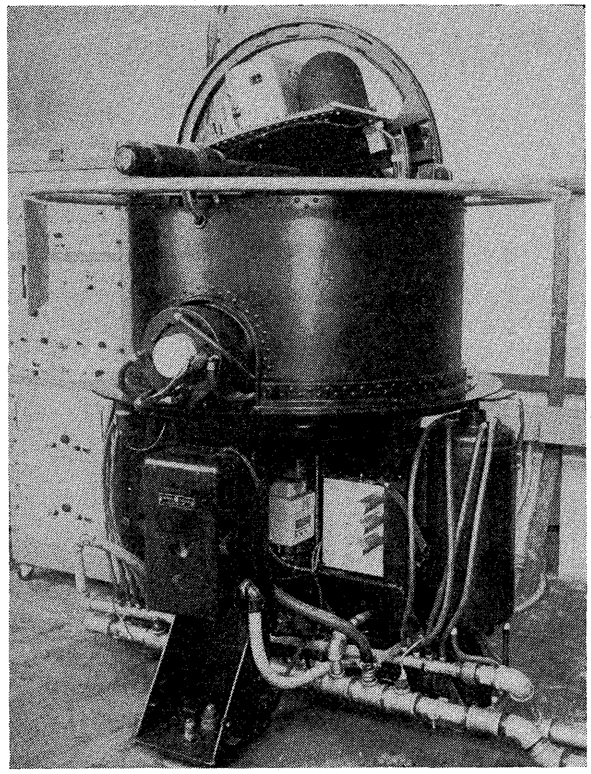


Fig. 9—Two-axis flight table.

Problems in Flight System Simulation

E. J. McGLINN†

Introduction

IN THE SUMMER of 1951, the Research Laboratories Division of Bendix Aviation Corporation initiated the development of a high-performance three-axis flight systems simulator for the Office of Naval Research, Department of the Navy. This effort was completed late in 1953 with the satisfactory construction of analog equipment designed primarily for the simulation of high-speed air-to-air missiles in real time. This equipment consists of a flight table and an associated simulator electronics unit. The three-axis flight table provides for the evaluation and testing of the actual control, stabilization, and guidance system equipment under the influence of angular motions of the simulated missile. The flight table contains three independently controlled gimbals as shown in Fig. 1 (opposite). The simulator electronics unit (computer) contains the significant elements required for representing high-speed dynamics on a one-to-one time scale. The computer (Fig. 2), when augmented by an analog facility, allows the flight table to be used in the complete trajectory simulation of a missile in three dimensions. (The Bendix three-dimensional flight systems simulator is described in more detail by Edwards and McGlinn.[1])

After completion of the simulator, the Bureau of Aeronautics established an operating program at the Bendix Research Laboratories in Detroit, Mich. For more than two years the simulation facility engaged in a wide variety of physical simulation and equipment evaluation studies. Early in 1957, the simulator was transferred to the Naval Air Missile Test Center at Point Mugu, Calif.

Although the flight table was especially designed for fast air-to-air missiles, the simulation program at Bendix also required that it be used in the real-time simulation of subsonic manned aircraft and missiles. A major difficulty in

† Bendix Aviation Corp., Detroit 35, Mich.

[1] C. M. Edwards and E. McGlinn, "The use of the Bendix flight table," *Proc. Natl. Simulation Conf.*, pp. 6.1-6.7; 1956.

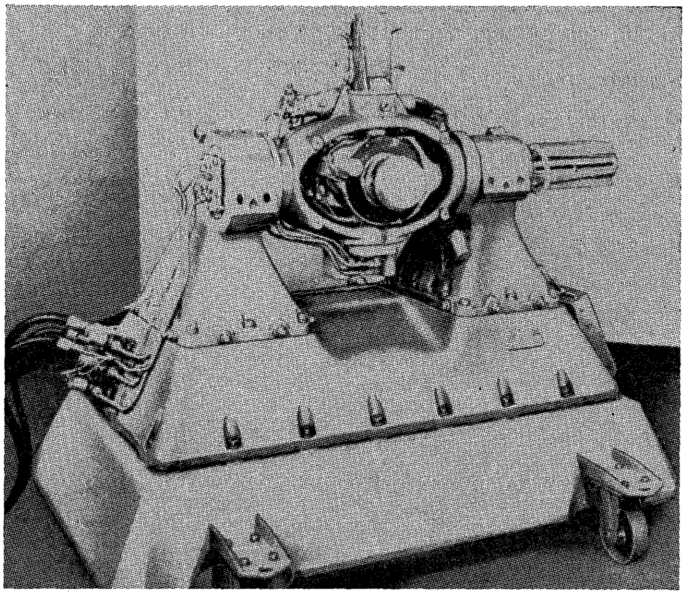


Fig. 1—Bendix three-axis flight table.

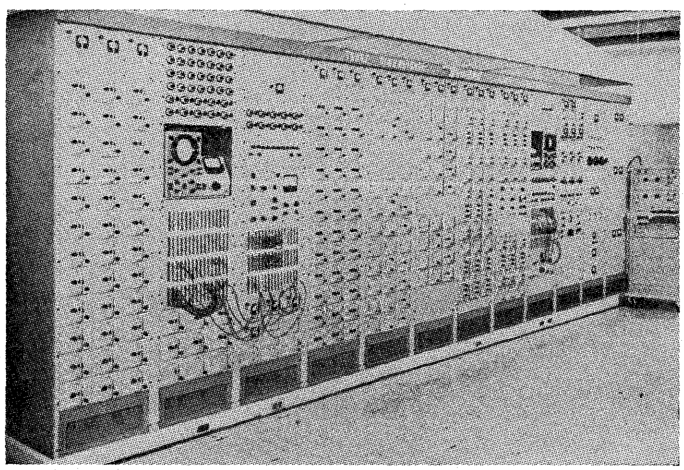


Fig. 2—Simulator electronics unit.

such a simulation was the low-speed performance required of the flight table. A technique of position control is described which extended the range of the flight table into the lower dynamic performance regions.

Another major problem connected with simulation of guided missiles in three dimensions was the target-missile geometry (*i.e.,* computation of the direction cosines of the line-of-sight vector). The large initial range and small miss distances required placed severe requirements on these computations. A technique involving continuous rescaling was chosen for the simulation work accomplished on the Bendix three-dimensional flight systems simulator.

Simulation of the Missile System

The simulated missile system block diagram is shown in Fig. 3. (Security regulations prevent a more detailed system description.)

As a reasonable approximation of the anticipated flight condition, the missile was assumed to fly at a constant

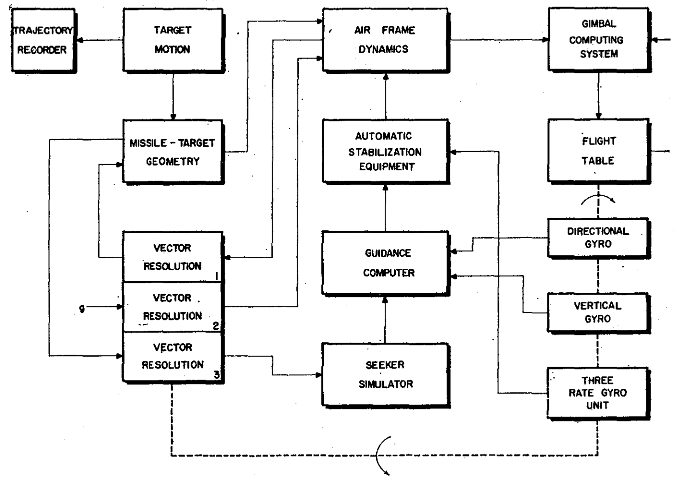


Fig. 3—Simulation block diagram.

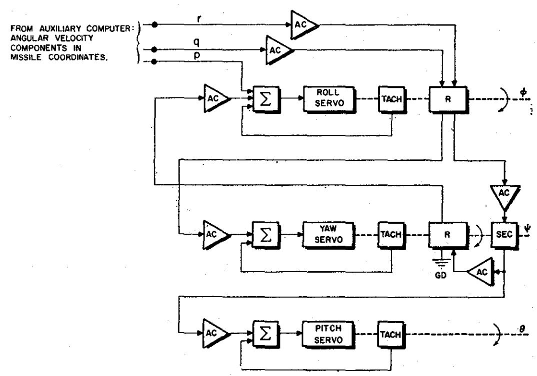


Fig. 4—Gimbal computing system—schematic diagram.

Mach number. The aerodynamic coefficients used in the simulation were varied with altitude since, with a constant Mach number, the forward velocity varies with altitude. Subsystem transfer functions were included in this study and portions of the actual missile components were tested in their respected locations in the simulation loop to establish their compatibility in the over-all system. The simulation studies included the effects of heading errors, gyro reference errors, boresight errors, steady winds, wind shear, wind gusts, and target noise.

Two sections of this simulation were rather critical with respect to the flight system simulator. These were the flight table and gimbal computing system and the missile-target geometry. These portions of the simulation and the methods used to accomplish them provided the primary problem in this particular simulation effort.

The Flight Table and Computing System

The gimbal computing system, as shown in Fig. 4, is a nonorthogonal transformation of missile body angular rates to the correct rates for driving the three flight table servos. This system is composed of analog equipment using 1000-cps suppressed carrier signals. The performance characteristics of the system were as follows:

Static accuracy	0.15 per cent of full scale
Bandwidth	50 cps
Drift	0.5 degree/minute (maximum).

The flight table, which contains the three servos, is an integral part of the gimbal computing system. (A complete discussion of the implications of this feature and the performance requirements for flight tables may be found in Blanton.[2])

The flight table is driven by three rate servos with the characteristics indicated in Table I.

TABLE I

	Roll	Yaw	Pitch
Maximum acceleration (rad/sec^2)	2500	500	500
Maximum velocity (rad/sec)	50	15	15
Attitude range (for noninterference of load—degrees)	Continuous on all gimbals		
Frequency for 90-degree phase shift (cps)	100	45	45
Transient time constant (sec)	0.002	0.003	0.003
Positional accuracy (degrees)	0.2	0.2	0.2
Input signal	1000-cps suppressed carrier voltage corresponding to a velocity command on all gimbals		
Steady state velocity error	Less than 0.015 per cent on all gimbals		
Maximum load	50 pounds (with the center of gravity not more than 2.5 inches from the center of rotation of the yaw gimbal; at reduced pressures, the center of gravity can be as much as ten inches from the yaw axis)		
Load dimensions for noninterference	8-inch diameter and 15-inch length		

These servos were, of course, designed for high-speed missile simulation. The use of this equipment in the simulation of a slower air vehicle might therefore be questioned, since accurate low-speed servo performance is limited by loop gain, static friction in the hydraulic motor and drive, and the low-speed capabilities of the tachometer.

In the design of the flight table servos, the principal factor that limited the low-speed performance was the tachometer. However, a careful choice of available tachometers provided excellent low-speed performance. A maximum load velocity variation of 0.5 per cent was achieved at a load speed of 0.3 radian per second. (A more complete discussion of the servo is contained in Bailey and Feder.[3])

Therefore, it is apparent that the low-speed properties of the flight table are really quite outstanding, and that the burden for accurate computation at any speed might be placed on the other computing elements in the system. Servo drift, however, must be considered. The flight table servos are electromechanical integrators that are exceptionally free of the type of drift normally attributed to the electronic integrator. Instead, the electromechanical integrator is subjected to a "counter" drift because of the tachometer. In effect, the tachometer feedback contains positional information which acts in the same manner as stiction in the servo. A more complete discussion of the effects of this type error is contained in Jones.[4] In this work, it was concluded that the errors of the two types of integrators, electronic and electromechanical, have a ratio on the order of

$$\frac{E_e}{E_m} = k \frac{T_s}{T}$$

where E_e is the error of the electronic integrator, E_m the error of the electromechanical integrator, k the ratio of the average drift of the electronic integrator to the aximum value of the tachometer positional error, T_s the total duration of the solution, and T the amount of time during which the input to the electromechanical integrator is very near the threshold. A good estimate for the value of k is 0.02. The value of T/T_s for the simulation of subsonic air vehicles involving lengthy flights is on the order of 2/3, and at the extreme approaches unity. Thus, for this type of simulation with the present configuration of the flight table, the more accurate integration would be accomplished with electronic integrators.

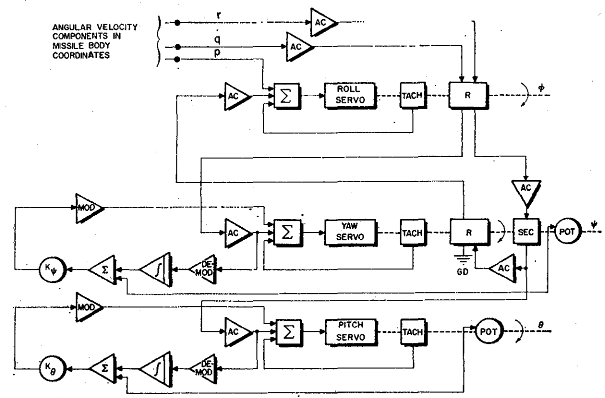


Fig. 5—Modified computing system.

Therefore, to improve the integrating capability of the system at low speeds, and hence the flight table resolution, an electronic integration and a position feedback loop were added to the yaw and pitch gimbal servos of the gimbal computing system, as shown in Fig. 5. Although system bandwidth was decreased by a factor of approximately two with the position feedback circuit added, the response was more than adequate for the simulation study. In addition, this improvement was accomplished with a minimum of effort using standard flight simulator electronic components.

The improvement was immediately apparent in the simulation operation since the drift problem was all but eliminated. A trajectory with a duration on the order of three minutes was compared to a digital check solution and a maximum deviation of less than 2 per cent was obtained.

[2] H. E. Blanton, "Performance Requirements for Flight Tables," Wright-Patterson Air Force Base, Ohio, WADC Tech. Rep. No. 54-250, pt. 10; September, 1954.

[3] K. V. Bailey and M. S. Feder, "Design of a high performance hydraulic control system," *Proc. Natl. Simulation Conf.*, pp. 7.1-7.6; 1956.

[4] T. Jones, Jr., "The propagation of errors in analog computers," Master's thesis, Mass. Inst. Tech., Cambridge, Mass.; May, 1952.

Missile Target Geometry

Probably the most critical portion of any three-dimensional missile simulation, with regard to the electronic computer, is the missile-target geometry, or the kinematics of the problem. The mathematical structure of the simulated system was

$$l = \frac{x}{R}$$

$$m = \frac{y}{R}$$

$$n = \frac{z}{R}$$

where

$$R = \sqrt{x^2 + y^2 + z^2}$$

$$x = \int u dt + x_0$$

$$y = \int v dt + y_0$$

$$z = \int w dt + z_0.$$

In this formulation, R represents the missile-to-target range; x, y, and z, the relative components of the line-of-sight expressed in inertial coordinates; u, v, and w, the translational velocities; and l, m, and n, the direction cosines of the line-of-sight in inertial coordinates.

An extensive preliminary study was conducted to obtain a geometry system which would be suitable for this simulation and compatible with the high resolution capacity of the flight table. Square root and division loops, with an accuracy of better than 5 per cent over a range of 100:1, were considered necessary because the required miss distance was relatively small with respect to the initial range.

A number of methods were evaluated and found unsuitable. The system adopted was one involving a continuous rescaling process during the computation. The equations simulated were:

$$10kR = [(10kx)^2 + (10ky)^2 + (10kz)^2]^{1/2}$$

$$l = \frac{10kx}{10kR}$$

$$m = \frac{10ky}{10kR}$$

$$n = \frac{10kz}{10kR}$$

$$k = \int_0^T [a(10kR) + b] dt + \alpha \qquad \text{for } 0 \leq t \leq T$$

where

$$[a(10kR) + b]_{t=0} = 0$$

$$k = 1 \qquad \text{for } t \geq T.$$

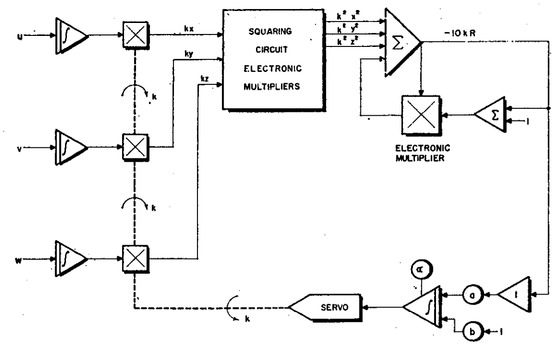


Fig. 6—Missile-target geometry simulation diagram. *Note:* a, b, α are adjusted for best results. $a = b \cong 0.2$, $\alpha \approx 0.10$.

The solution of these equations is shown in Fig. 6. Electronic multipliers were used throughout since rapid response was necessary when target noise was introduced. A rescaling servo was driven by the function k as indicated. With the computer in the initial condition state, k approximately equaled 0.1 (where unity indicated one machine unit, or full scale). When computing, k increased as R decreased according to the defining equation and when R equaled 0.1, k was approximately 1. The unscaled circuit provided good accuracy to 0.1 machine unit, while the rescaled circuit was accurate to approximately 0.01 machine unit.

Resolution of the direction cosine information into missile axes was accomplished by the resolver computing section of the flight systems simulator. For the over-all system, accuracy was approximately 1 per cent for $0.1 < R < 1$, and better than 5 per cent for $0.01 < R < 1$.

Conclusions

It has been demonstrated that a real-time simulator with a high dynamic range can be used for applications which require performance on the lower portions of the dynamic range. Such a simulation, using the Bendix three-dimensional flight systems simulator, has been described. Problems encountered in the flight table and gimbal computing system were solved by careful selection of critical components (*e.g.*, the feedback tachometer) and the addition of an electronic integrator and a position feedback loop to the yaw and pitch gimbal servos of the system. Problems presented by the missile-target geometry were minimized by use of electronic multipliers and a continuous rescaling process during the computation.

Acknowledgment

The successful culmination of the simulation studies performed with the aid of the Bendix flight systems simulator has been the result of the combined effort of many people under the guidance of C. M. Edwards. The author especially wishes to thank F. B. Lux, who suggested the rescaling circuit for the kinematics section, W. H. Baur for his effort in synthesizing the computer setup, and J. Kaiser for his valuable assistance in the simulation program. Other Bendix Research Laboratories Division personnel also contributed support to this program.

Analog, Digital, and Combined Analog-Digital Computers for Real-Time Simulation

C. G. BLANYER† AND H. MORI†

Introduction

A LARGE, important area of use for automatic computing machines is the study of the dynamic operation of physical systems, especially guidance and control systems. If the mathematical model under study describes in substantial detail an existing or contemplated piece of equipment and if stimuli are employed that correspond to those experienced by the equipment when in use, the process of study is referred to as simulation. The term simulation cannot be used to express a precisely defined area of work because widely varying interpretations are given to it. As used here, the term refers to an area of computation comprising analysis of large, complex systems that are represented in sufficient detail to include many of the distinguishing characteristics of practical equipment, even though few, if any, parts of the actual system are linked with the computational components.

Although rapid solution of a problem is advantageous in itself, the requirement for real-time speed in computation becomes necessary only when the simulation includes actual equipment. Additionally, the existence of identifiable signals related closely to physical variables in the real equipment provides both concrete and abstract advantages, particularly if the computer signals occur in the same time as in the actual equipment.

Simulation can be performed with either analog or digital computers. Furthermore, the many variations of the traditional form of these machines tend to blur the distinction between them. Most of the variations have resulted from attempts to combine the advantages of the two methods.

The word analog itself indicates that analog computers inherently involve a measure of simulation. Although simulation indeed has formed a large area of activity for these machines, the investigation of simple dynamic relationships including only a few variables and the education of engineers in the field of dynamics have been important uses of analog computers. Because the principle of operation of digital computers is the reduction of all relationships to a few simple forms that can be performed with extreme accuracy, a wide variety of operations, including bookkeeping and certain logical manipulations as well as simulation studies, can be accomplished with these machines. In general, they are most useful in fields where many arithmetic operations must be performed with great accuracy.

† Hydel, Inc., Cambridge, Mass. Formerly at the Dynamic Analysis and Control Lab., M.I.T., Cambridge, Mass.

Although both types of computers have been used extensively for simulation, most installations leave much to be desired from the point of view of convenience, accuracy, and efficiency. The disparate characteristics of the two types of machines indicate that either improvements in each should take widely different forms, or that features selected from each type should be combined to form the most satisfactory simulator. One major difficulty associated with improvement in simulation computers is the lack of a clear understanding of real requirements for a highly effective simulator, together with a realistic comprehension of what can be accomplished with various computing methods.

The purpose of this paper is to review the characteristics of analog and digital computers and of several variations of these basic machines, to point out the areas in which improvement is most needed, and to outline some of the methods being investigated for achieving such improvements. An attempt is made to describe the characteristics of an efficient real-time computer and to evaluate the demands that these characteristics would make on the computer designer.

Characteristics of Analog Computers

The distinguishing characteristic of analog computation is well known; physical quantities, such as voltages or shaft angles, are constrained to obey relationships analogous to variables existing in a real system. This process is feasible because many different physical systems obey mathematical laws of identical form. However, the ability of a system to perform certain elementary mathematical operations is not sufficient to make it useful for simulation. Generation of complex functions and communication with external equipment and personnel must be provided. Furthermore, combination of operations into large groups must be practicable. Certain abstractions that correctly belong in the field of human engineering determine, to a large extent, the utility of a computation scheme.

Before computer characteristics can be evaluated effectively, appropriate criteria must be established. Three useful criteria are the accuracy and the speed with which a computation is performed and the degree of difficulty associated with an operation. Error, or imperfect accuracy, includes any deviation of a response from the required result. Errors, which may exist statically or appear only in dynamic operation, almost invariably accumulate when functions are combined. Finite resolution results from unavoidable quantization of variables or from noise which limits the minimum discernible variation in a signal. Although speed can be measured quantitatively, the appropriate criterion in simulation is whether the dynamic error in an operation permits or prevents real-time operation. The term "difficulty" (as used above) applies particularly to operations that reduce significantly the convenience of computation, or that result in great effort or expense.

The irreducible minimum error in a simple analog computation is set by the accuracy that can be maintained in the basic circuit elements. Designers have concluded that the characteristics of materials limit this basic accuracy to roughly one part in 10^4, although the stability of available circuit elements and their freedom from parasitic effects have increased steadily during the past decade.[1] As a result, the relatively simple mathematical operations of summation, multiplication by a constant, integration, and the inverses of these operations, usually can be performed to an accuracy of three significant figures. Typical noise levels limit resolution to one part in 10^4 of full scale. Lags associated with these components are inconsequential, with certain exceptions noted later. Errors common to electronic analog integrators are of a type that may be almost negligible in the simulation of closed-loop systems. However, these errors can prevent open-loop operation of typical integrators in many circumstances, particularly if long computing times are involved. The unavoidable noise in computer circuits frequently prohibits the explicit use of differentiation. However, in many simulation problems, a lag can be associated with the differentiation without introducing unacceptable dynamic errors. Unless an indirect method of computation can be utilized, integration and differentiation with respect to variables other than time can be accomplished only with all-mechanical devices that are both slow and inconvenient.

The operations of multiplication and division of variables and function generation are difficult to perform in most analog installations. Static accuracy varies between 1/10 per cent for multiplication and 1 per cent for the generation of complex functions. Dynamic performance, even of the electronic methods, usually is not completely satisfactory. The use of servomechanisms to convert voltages into shaft positions makes possible various combinations of these operations in output elements; however, resolution as fine as 0.01 per cent in output potentiometers can be achieved only in linear elements. Furthermore, severe rate and acceleration limits impair servo performance. A typical unit requires a few tenths of a second to reach full scale at maximum speed and may require an equal time to accelerate from zero to full speed.

Programing and setup procedures for analog computers are readily intelligible to engineers. A computer diagram is very similar to the familiar block diagram. In many installations, the interconnections of individual circuit elements to form functional units must be planned; in others, only the functional units themselves need be connected. Although setup typically consists of making physical connections between units by means of patch cables and of setting controls, examples exist of installations employing remote, as opposed to manual, connections and settings.[2] Automatic or especially convenient manual methods of changing a few settings and a few parameters may be provided. The physical manipulation of connectors and controls included in setting up and operating the machine is time consuming, but forms a psychological aid for the operating personnel.

In summary, an analog computer is characterized by similarity to real systems, both in detail and in an over-all sense. Most simple operations of algebra and the calculus can be performed with approximately three-place accuracy and four-place resolution, although only a single independent variable is permissible in most integration and differentiation. Nonlinear operations, which are relatively difficult to perform, limit achievable resolution and tend to be at least an order of magnitude less accurate than linear operations. Speed is no problem in the simpler operations, however, the dynamic performance of servos and many nonlinear devices is marginal in large-scale real-time simulation. Over-all solution accuracy typically is 1 per cent in the simulation of a closed-loop system of medium complexity. This degree of accuracy suffices for many engineering applications.

Characteristics of Digital Computers

In a digital computer, bits of information are carried in groups by means of a spatial arrangement of signals or a time arrangement of pulsed signals. Because essentially all mathematical operations can be approximated by successive applications of addition and comparison, solutions to differential equations and other manipulations required for the simulation of systems can be obtained.

The basic operations must be combined to form functional relationships to a much greater extent than in analog computers; in most digital computers, these basic operations, which are separated in time by repeated communication with a memory, must be performed sequentially. Because the degree of approximation permitted in nonexact functions can be varied, the speed and accuracy of operation are not fixed quantities. Furthermore, the use of sequential operations causes speed and accuracy to be functions of problem complexity. If the details of the operational methods to be employed are left to the programer, the effectiveness of the speed-accuracy compromise depends, to a great extent, upon the skill of the programer in the fields of numerical analysis and logical manipulation, and upon the time allowed for program preparation.

Resolution, which usually can be measured in tens of binary digits, forms essentially the only limit to the exactness of basic operations such as addition and multiplica-

[1] G. A. Korn and T. M. Korn, "Electronic Analog Computers," McGraw-Hill Book Co., Inc., New York, N.Y.; 1956.

[2] H. E. Harris, "New techniques for analog computation," *Instr. and Automation,* vol. 30, pp. 895-899; May, 1957.

tion. If a problem requires a large number of multiplications, the time required for multiplication may determine, to a significant degree, the resolution that can be retained.

Operations of the calculus are performed by approximate arithmetic procedures, with the programer exercising some choice of the specific method to be used. Again, results are repeatable precisely and indefinitely, although the errors introduced by approximations and truncations may be serious if rapidity of operation consistent with real-time simulation is required. Functions may be generated by solution of a formula, consultation of a stored table of values, or interpolation between widely spaced stored values. The evaluation of integration presented here also is appropriate for function generation, except that the latter process may require a large part of the available fast-access storage space.

The performance of general-purpose digital computers is difficult to assess, owing to the wide variety of computing methods that can be used and the close relationship between problem complexity and the speed-accuracy combination. However, the accuracy of typical installations might be estimated at a few per cent for applications such as real-time simulation of a missile in three dimensions.

Because of the method by which information is read in and stored, a large mass of detailed data, such as that involved in the description of an arbitrary function, can be inserted or changed quite rapidly. This advantage is offset partially by the fact that instructions for relatively simple operations also consist of a large mass of details. Programing essentially consists of an exercise in numerical computation with a limited choice of methods and language determined by specific design features of the computer.

Special routines such as the Differential-Equations Psuedo-Code Interpretor (DEPI)[3] and the Numerically Integrating Differential Analyzer (NIDA)[4] have been developed. These methods reduce the amount of work associated with programing and result in a setup procedure that is similar to that for an analog computer. These routines, of course, do not alter the capabilities of the machine, nor do they eliminate the necessity for knowledge of coding in specific machine language. However, they do reduce the problem of treating differential and integral relationships in terms of numerical analysis. The net result of using these routines, however, may be a decrease in speed of computation because the simplified code must be translated by the machine into digital-machine instructions and, moreover, these sets of instructions usually are not efficient.

The concepts embodied in special programs have been extended to special machines that are programed permanently to permit convenient solution of integro-differential equations. The most widely known of these is the Digital Differential Analyzer[5] or DDA. In such machines, special sections of the memory, designated integrators, are arranged to be processed in sequence by an arithmetic unit in accordance with a simple integration formula. These so-called integrators also can be used as servos, limiters, and generators of simple functions. Typical DDA machines offer no substantial advantage over general-purpose computers in speed or accuracy, and their performance may be poorer owing to the extremely simple methods of integration employed. However, most systems engineers undoubtedly would find these special machines easier to use.

Comparison of Computers

The task of comparing mechanisms that are basically as different (and diversified) as analog and digital computers perhaps can be accomplished best by concentrating on the results of operations instead of on the operations themselves and by commenting on the techniques necessary to utilize them. The following evaluation is based on the characteristics of existing computing equipment rather than on the inherent features of the two types of machines.

The accuracy of digital computers accounts for their superiority in algebraic operations, particularly in critical processes such as summation of large, nearly equal quantities. All-algebraic loops usually must be avoided in analog machines owing to stability problems. The range of problems that are possible with analog computers is restricted because most analog installations are capable of performing only a few multiplications of variables, with marginal accuracy. On the other hand, the time required for multiplication, either by constants or variables, may become significant in digital computers if many of these operations are necessary.

Digital integration on a real-time scale is quite difficult. Complex integration formulas require many operations and raise questions of convergence. Simple methods require a high density of points in time, furthermore, they introduce serious limitations on the rate of change of variables. For example, even an electromechanical analog servo or servo integrator may be capable of changing its output two or even three orders of magnitude faster than a digital-counter type of integrator with equal resolution.

The ease with which arbitrary functions can be set up and changed in a digital computer is an advantage that is particularly significant in systems studies involving empirically derived relationships. This advantage extends to a variety of analytic functions as well. However, a speed problem may exist in real-time simulation if many functions are to be represented.

Analog computers have superior characteristics for programing and communication. The advantages lie not

[3] F. H. Lesh and F. G. Curl, "DEPI: An Interpretive Digital Computer Routine Simulating Differential-Analyzer Operations," Jet Propulsion Lab., Pasadena, Calif., Memo. No. 20-141; March 22, 1957.

[4] H. H. Anderson and J. R. Johnson, "Numerical Integration of Differential Equations on the 704 EDPM," Eng. Lab., IBM Corp., Poughkeepsie, N.Y., Tech. Rep., Code: 011.070.592; January 11, 1956.

[5] M. L. Klein, F. F. Williams, H. C. Morgan, and S. Ochi, "Digital differential analyzers," *Instr. and Automation,* vol. 30, pp. 1105-1109; June, 1957.

only in the close parallelism between the system and machine diagrams—an advantage shared in part by DDA machines—but also in the parallelism between control systems and analog-computer elements. The physical aspects of communication, such as connection of real components and read-out of data, cause complexity or speed difficulties in digital computers.

Although special programs and machines have increased the usefulness of digital computation in simulation, numerous factors tend to limit the speed of digital machines, and therefore, analog computers are superior in many applications requiring high operating speed. The margin of difference is difficult to assess quantitatively. However, some conception of the problems involved can be gained from consideration of the operations performed in a large-scale simulation. A recent three-dimensional guidance study performed on the large-scale analog computer at M.I.T. involved the following individual mathematical operations (exclusive of scale-factor changes):

Additions of two quantities:	65
Multiplications by a constant:	102
Multiplications and divisions by variables:	51
Integrations:	23
Generation of analytic functions:	25
Generation of arbitrary or piece-wise linear functions:	11.

This simulation was conducted in real time (a solution time of approximately 20 seconds). In a similar situation, the programing and operation of a digital computer to obtain a check solution for this analog machine became so tedious that the check solution had to be simplified to an extent that rendered it almost useless.

Certain abstractions assist the analog user. First, the tangible features of these machines aid in orienting the engineer and often help him to feel at ease with the mechanical aid. Second, the computer may be one of the few contacts with the realities of the physical world for the systems analyst who does not work closely with components. An example in system synthesis occurred recently at M.I.T.[6] A design of a time-variable control system, that was ingenious theoretically but unfeasible practically, was reconsidered after troubles in the simulation procedure illustrated the difficulties involved.

Trends in Computational-System Improvement

Activities in the development of computer components, control systems, and programing methods have included the refinement of commonly used techniques and the trading of methods between digital and analog computers. Also, a number of schemes have been devised for making more efficient use of existing facilities.

Much of the development effort in analog systems has been devoted to providing faster means for setting up and checking computers. Arrangements exist whereby parameter setting and function-generator calibration instructions are reduced to taped commands that can be fed to the machine automatically. Automatic digital print-out of data is being included in many installations.

In the digital field, the effort to increase the rate at which operations are performed probably will go on indefinitely. Because memory storage has been a bottleneck in attempts to increase both capacity and speed, much work has been concentrated in this area. Most installations now provide several means of storage characterized by a variety of capacities and access time. An example of an advanced system is a photographic technique for high-scanning-rate storage that is under development at M.I.T.[7] This scheme utilizes a projection system and a rotating mirror to sweep digital information stored on a photographic medium past a row of stationary photoelectric transducers. Preliminary studies indicate that a reading rate greater than 10^7 bits per second is obtainable.

Better methods for taking advantage of the capabilities of existing facilities are being devised. For example, highly accurate solutions can be obtained with relatively low accuracy equipment by the solution of variational equations describing deviations from a known solution. This technique is particularly valuable with analog computers. Basic equations can be solved with very small error by digital or other means; standard conditions are employed in this solution for parameters and relationships that are to be varied. Equations describing the dependent variables of interest then can be linearized about the standard solution. If this technique is applied to a systems study, such as an investigation of ballistic missile trajectories, the variational equations can be generated as linear equations in time with coefficients that vary as integral powers of time. Because the equations describe relatively small deviations, the effects of errors are reduced by one or two orders of magnitude. The equations, in general, include a large number of multiplications and similar functions that may prove difficult for many computers. However, the situation is one in which an increase in the quantity of available equipment increases the accuracy by reducing truncation errors in the expansions. In a recent simulation study using these variational techniques at M.I.T., an equivalent over-all accuracy of four significant figures was obtained.

Techniques for more efficient utilization of digital computers usually include methods for reducing the detail required in programing, and frequently involve procedures based on the block-diagram concept. Examples of trends in this direction are the DDA machines and the DEPI and NIDA programs referred to previously.

Advanced digital computational schemes, as well as

[6] P. Goldberg, R. V. Morris, and L. H. Walker, "Multicondition Terminal Control and Its Application to Aircraft Landing," Dynamic Analysis and Control Lab., M.I.T., Cambridge, Mass., Rep. No. 109; September 30, 1957.

[7] D. M. Baumann, "A High-Scanning Rate Storage Device for Computer Applications," presented at Twelfth Natl. Meeting, Assn. for Computing Machinery, Houston, Texas; June 19-21, 1957.

convenient programing techniques, are being devised. Improved integration formulas and other computational algorithms have been incorporated into an all-digital system for an operational flight trainer. This work,[8] which was done at the University of Pennsylvania, was supported by studies of the rate at which discrete computer solutions are required to ensure stability. A study of faster integration schemes for DDA's has been conducted at Harvard.[9] The results indicate that the use of complex integration schemes instead of the simple methods presently used in DDA's will improve the speed of computation considerably. An extension of DDA techniques has resulted in the Incremental Computer,[10] which solves a single basic equation that can be rearranged to perform many different mathematical operations.

The increased use of analog-style programing for general-purpose digital computers can be expected to continue because the versatile general-purpose machines are becoming common pieces of equipment to satisfy demands other than those of simulation studies, and because block-diagram organization possesses undeniable advantages for engineers. The extension of the combination of digital circuitry and analog form is resulting in the appearance of two types of hybrid systems that differ mostly in the degree to which the principle of combination is carried.

One hybrid device operates internally on a digital basis, but accepts analog inputs and delivers analog outputs. An example is the DIANA system[11] that was under development at M.I.T. This system performs function generation and multiplication by digital techniques. Functions of a single variable are stored on a magnetic drum and read out at command from a coincidence detector that compares an analog input with the stored values of the independent variable. The function is delivered to the output through an analog amplifier in which the gain is determined by the digital signal. In a more extensive project, a complete combined system is being designed at M.I.T. for use as an operational flight trainer.[12] This real-time analog-digital simulator employs conventional analog integrators, but performs all other computation by floating-point digital techniques. Programing requires no familiarity with the logical structure of the computer. Another method under study at M.I.T. involves the unique combination of digital equipment and a time-shared electronic analog multiplier.

A combination of individual analog and digital computers incorporating the best features of both systems has proved useful. In such a combined analog-digital simulator, each subsystem should be handled by the type of computer to which it is most readily adaptable.[13] This procedure has been implemented at the National Bureau of Standards in the simulation of a closed-loop sampled-data system involving a human operator and control equipment.[14] The problems of accuracy and drift in a long-time kinematic problem suggested the use of a digital computer, and the need for a large number of solutions to gather statistical data (owing to the nonrepeatability associated with a human operator) indicated the desirability of analog computation. The simulation of this system by either analog or digital computers separately could have been done, but with difficulty. In the combined system finally selected, the analog machine is used for dynamic-equation solution in real time and for activation of display devices. The use of digital computation is restricted to operations requiring precise and accurate computation, logical decisions (including control), and the statistical analysis of resultant data.

A combined analog-digital arrangement is being studied at the Ramo-Wooldridge Corporation to reduce the amount of time required to perform a three-dimensional missile simulation.[15] Simulation of the missile is being accomplished on the analog machine, and the kinematics has been assigned to a digital computer. One technique being employed is the combination of analog and digital integration, in order to produce a fast, accurate system, by integrating the high-frequency signal components by analog techniques, and the low-frequency components by digital methods. The combined simulation has been estimated to require twice as much equipment, but only one fifth the time of the equivalent operation performed completely by digital methods. An all-analog computation was ruled out in this instance by the accuracy requirements. Although real-time simulation was not a necessity here, high computation speed was desirable because a considerable amount of statistical work was contemplated.

Another combination of an analog computer and a digital computer is being attempted at Convair.[13] A rather complex weapons system is being simulated in real time because physical system components are to be included. An all-analog simulation is impractical in this situation owing to the accuracy requirements. An all-digital simulation would prevent real-time operation.

Considerable activity is occurring in this area; the general trend is to use digital equipment with a fixed program for the portions of a simulation that demand extreme accuracy and to reserve an analog computer for high-

[8] W. H. Dunn, C. Eldert, and P. V. Levonian, "A digital computer for use in an operational flight trainer," IRE Trans. on Electronic Computers, vol. EC-4, pp. 55-63; June, 1955.

[9] R. L. Alonso, "A Special Purpose Digital Calculator for the Numerical Solution of Ordinary Differential Equations," Computation Lab., Harvard Univ., Cambridge, Mass., Prog. Rep. No. AF47; 1957.

[10] W. J. Moe, "A Digital Computer System for Airborne Applications," presented at Natl. Conf. on Aeronautical Electronics, Dayton, Ohio; May 14-16, 1956.

[11] P. A. Hurney, Jr., "Combined analogue and digital computing techniques for the solution of differential equations," *Proc. Western Joint Comp. Conf.*, pp. 64-68; 1956.

[12] E. W. Pughe, Jr., "Logical design of a real-time analog-digital simulator," M.S. thesis, Elec. Eng. Dept., M.I.T., Cambridge, Mass.; 1957.

[13] J. H. McLeod and R. M. Leger, "Combined analog and digital systems—why, when, and how," *Instr. and Automation*, vol. 30, pp. 1126-1130; June, 1957.

[14] H. K. Skramstad, "Combined Analog-Digital Simulation of Sampled Data Systems," presented at the AIEE Summer General Meeting, Montreal, Can.; June 24-28, 1957.

[15] J. H. McLeod, Jr., "The simulation council newsletter," *Instr. and Automation*, vol. 30, p. 695; April, 1957.

frequency dynamic simulation involving extensive parameter variation.

The Future of Simulation Computers

The temptation to describe a Utopian computation scheme is difficult to resist when the advantages and disadvantages of existing simulation equipment are outlined. However, specifications of this nature are best modified by economic and practical factors which may play a large part in determining future trends in machine computation. Such factors are considered briefly here.

A computer for simulation purposes should not be designed to unnecessarily stringent specifications. Errors of approximately 1 per cent in the final solution can be tolerated for many simulation problems, although a machine capable of repeating solutions to approximately 0.1 per cent is desirable. Certain operations probably should be performed to four-place accuracy to achieve this goal. Variable accuracy would be desirable in a computer for simulation if accuracy could be traded for simplicity. Although over-all accuracy to four places may be highly desirable in a few applications, the computer should be designed for ease of preparation for less accurate computation. Complexity in preparation and equipment for the high-accuracy computation probably could be tolerated.

As indicated earlier, the speed requirement can be stated simply in that real-time operation is required. In any case, speed should be sufficient to permit the operator to compare successive solutions mentally. Provision for a variable time scale is useful for checking and exploratory purposes.

Preparation, setup, and control of the computer should be characterized by convenience and flexibility. Furthermore, rapid, convenient programing should be possible without the need for special skills. Setup for a new problem should require a short time in comparison with the computing period; little is gained in reducing to fifteen minutes the setup time for a two-week program if multiple-shift computer operation is not used. Adjustment of parameters in a system as well as changes in subsections of a system should be convenient operations. Means for incorporating real equipment and human operators in a systems study on a computer should be provided.

A few additional comments on operations and methods may be worthwhile. The machine must be capable of performing multiplications, divisions, and common types of nonlinearities easily and in quantity. Purely arithmetic computations must be possible without special techniques. Machine capabilities should include dynamic operations such as integration and differentiation on more than one independent variable. Automatic scale factoring should be provided to reduce the drudgery of preparation and checkout; components should be checked automatically, and means for repeating a preset static check should be available. Both digital and analog read-out displays should be provided to preserve the advantages of multiplace accuracy as well as the graphical form of data presentation. If substantial sections of the machine are unused during studies, the installation should be arranged for simultaneous use on more than one problem.

Machine setup and physical arrangement should be based as thoroughly as possible on concepts that are familiar to the engineer. The block-diagram or flow-chart representation of relationships has proved its value repeatedly, not only in engineering, but also in unrelated fields in which organization is important. The widening range of subjects with which an engineer must become acquainted suggests that additional burdens of learning should not be placed on technical men to enable them to use computational aids. Not only should specialized machine languages be avoided, but optimum machine exploitation should not require detailed knowledge of electronic circuitry.

Comparison of these specifications with the capabilities of analog and digital computers indicates that a computing facility composed of both types of equipment suitably interconnected would be of great immediate utility. Functions could be assigned to the type of mechanism best suited to the specific operation; in addition, individual operations might be performed by combined types of equipment separated on a time basis or some other basis. For example, slowly varying components of signals might be calculated by digital means while the analog equipment is being utilized essentially for time-wise interpolation. Of course, any such combination schemes place a heavy burden on analog-to-digital conversion mechanisms. Because many of the troubles associated with digital simulation could be alleviated by improved conversion equipment, the development of fast, convenient converters probably should be emphasized in simulation-computer development programs.

To establish firm goals toward which computer designers should be working, accuracy and speed requirements should be determined carefully. As indicated previously in this section, accuracy to four and five or more significant figures is not vital to much successful simulation. Experience in a large analog computing facility has shown that over-all agreement between analog and digital check solutions can be maintained to approximately ½ per cent for complex systems by careful and laborious trimming of the analog. On the other hand, the effort required to achieve these small errors is not applied during large parts of the simulation because the desired information on system behavior is obtainable without it.

Undeniably, real-time speed is mandatory for the class of simulation including environmental component testing, human-response investigation, and similar activities. However, in many cases, inclusion of real-system equipment in a simulation is a mistake even if the possibility exists and is attractive. An analogy exists between this situation and the process of obtaining an answer to a problem without understanding the method of solution. In a variable-time-scale computer, operators frequently adjust the length of a problem to suit personal preference (limited by machine capabilities, of course) without much regard for the realism of the time scale employed. This suggests that factors other than physical tie-in, or the desire to obtain data very rapidly, are operative in the selection of an appropriate solution time. In wide varieties of dynamic systems, the real-time scale may be so short or so long that real-time simulation is either absurd or highly undesirable. Recent work at M.I.T., for example, has included simulation of systems in which transients are completed within microseconds at one extreme and last for many years at the other extreme. Study of systems with such diversified characteristics should not be excluded from consideration in planning simulation computers. Thus, the goal in speed for such computational systems probably should be based on the preference of operators; rapidity of computation that is fitted to an engineer's direct memory span and other mental processes appears to be consistent with the needs of most simulation where real-time scales are desirable.

Although analog equipment approaches fairly closely the accuracy required for most simulation purposes, substantial improvement in this respect seems doubtful. On the other hand, the tenfold or hundredfold increase in speed necessary for satisfactory utilization of general-purpose digital computers for simulation probably cannot be accomplished efficiently by the brute-force speedup of individual operations. At least one combination of characteristics of the two basically different machines would be highly satisfactory; that is, a system arranged and handled in analog form, in which operations are performed simultaneously by digital techniques. Parallel operation is probably the most efficient mechanism for obtaining the speeds required. By means of this technique, a simulator can be developed in which sections, if not individual operations, are independent and can be isolated; an increase in complexity or accuracy would involve the utilization of more equipment and would increase the cost, but would not entail a loss in speed.

Any scheme that results in specialization of a computer leads to economic repercussions, some of which can be severe. The tremendous number of digital computers in use may be attributed to the extremely wide range of activities in which these machines are useful, if not vital. Any computer that is limited to simulation work by inherent design features is restricted to a smaller market, a smaller volume of use, and substantially less support both in operation and in crucial developmental activities. Although advanced programing procedures may simplify the use of digital computers in the field of simulation, they are not likely to remove the obstacles inherent in the application of ordinary general-purpose digital machines to real-time simulation. If this assumption is correct, the future of simulation computers depends upon two things: the ingenuity and activity of the users of such machines in extending the field of systems analysis to the immense areas in which these methods are applicable, and the ability of designers to devise machines that retain the versatility of general-purpose digital computers while fulfilling the unique requirements of simulation computers.

Acknowledgment

The authors wish to thank Dr. William W. Seifert of M.I.T. for furnishing information and ideas for the preparation of this paper. The consultation and advice of colleagues at M.I.T. are also acknowledged.

Discussion

A. S. Baron (Westinghouse Electric Corp., Baltimore, Md.): Can you furnish the names and the locations of the hybrid installations to which reference was made?

Mr. Mori: We have a number of copies of this paper available; it has a complete bibliography and the references are in there.

H. McKenna (General Motors Research): What analog computer are you talking about when you quote 1 per cent accuracy for a simulated system? How many amplifiers in this system and is the system linear or nonlinear?

Mr. Mori: The computer to which I referred mostly was the home-built analog computer at the Dynamic Analysis and Control Lab of M.I.T. This was originally built in conjunction with the three-axis flight table. A part of it consists of ac carrier equipment. There are 15 electromechanical rate servos or integrators, each of which can have five output elements; in conjunction with this we have about 130 dc amplifier operational amplifier positions. The entire system has been estimated as the equivalent of 400 to 600 amplifier positions. It is very definitely nonlinear. Multiplications and so on are done in the servo part of the system.

H. F. Meissinger (Hughes Aircraft Co., Culver City, Calif.): In reference to the variational approach mentioned, does this mean the behavior of the system is predicted on the basis of linear (or higher order) extrapolation from a solution obtained on the computer?

Mr. Mori: You obtain a solution first on digital computer or by hand computation methods or in some cases you can get an analystic solution if you leave out some of the complications. Then you must develop some variational equations from this. You can expand terms in Taylor series, or similar operation. These variational equations, which describe deviations from basic equations, then are placed on the analog computer and simulated.

E. L. Harder (Westinghouse Electric Corp.): How long did the digital check solution take?

Mr. Mori: On a simulation of a transonic aircraft that we studied, it took one programmer four months.

The Place of Self-Repairing Facilities in Computers with Deadlines to Meet

LOUIS FEIN†

Introduction

FROM the title of this paper, one might expect to find in it exclusively a detailed account of self-repairing facilities in computers. However, the objective here is to identify the *place* of self-repairing facilities in computers with deadlines to meet. The objective is not to discuss details of self-repair. How does self-repair fit in with the host of techniques used and proposed for attaining the performance required of equipment with severe deadlines to meet?

In order to find a place for any activity with respect to related activities, one must set up a framework, a classification schema, allowing for the proper placement or classification of this activity. Thus, this paper contains a classification of used and proposed techniques for attaining high computer reliability, here called performability. In addition, a concept of deadline is quantitatively defined. Finally, the place of self-repair as a technique is identified. Economic implications of these techniques are not discussed here.

Deadline Situations and Some Practical Examples

"Computers with deadlines to meet" is the theme of this conference. We suspect that what motivated the organizers of this conference was a more specific concern about computers with *severe* deadlines to meet—those computers that must operate in real time, for example. In fact, this example seems to predominate here. But it is clear that all operating computers have some kind of deadline to meet; they are not all severe, but they are deadlines. We recognize that these deadline situations are important, yet the notion of deadline is still rather qualitative. One is thus motivated to characterize such deadline situations quantitatively; *i.e.,* this characteristic situation parameter that entirely describes the deadline situation in terms of its quantitatively defined elements should be isolated, named, and defined, and should have a unit selected for it. Results should, of course, be harmonious with intuition and experience. It would be attractive to be able to assign a number to the severity of a deadline situation both to measure the severity of the deadline and to make a quantitative comparison among deadline situations having different degrees of severity.

A deadline situation contains several important elements. The elapsed time from start to end of a computation is significant in determining the severity of a deadline, and it is intuitively clear that the amount of work to be done during this available time, and the capability and reliability of the equipment are also significant parameters whose values contribute to how we feel about the "toughness" of a deadline. This deceptively simple description identifies those situation characteristics that should be named and defined, and for which units should be selected. Thus, the concept of deadline can be handled quantitatively, in terms of time (the variable usually considered exclusively), and in terms of a relation between relevant characteristics of the computer and relevant characteristics (the application parameters) of the job to be done by the computer. (One of the fundamental problems in the computer field is to identify, name, define, and select units for the characteristic parameters of computers, as well as the characteristic parameters of representative applications, and finally to find functional relation among them. It will be noted later that in the case under discussion here, we indeed have a simple illustration of computer parameters like power and up-time, of an application parameter, like the time available and required to do the job, and of a functional relation among these parameters, namely the deadline ratio.)

It is sufficient for our purposes to borrow some basic concepts from elementary mechanics. One is inclined to name the characteristic that would measure the severity of a deadline the "deadline ratio" or "deadline coefficient," and to define it as the ratio of the time T_j required by a given equipment to do a job, divided by the time T_L available or scheduled to do this job, *i.e.,*

$$\text{deadline ratio} = T_j/T_L.$$

Thus, when a job is to be done within the capabilities of the equipment and within a given time (T_L), the situation is characterized by a deadline ratio less than or equal to one; without these conditions, it would be greater than one.

It will be found useful to rewrite the expression for the deadline ratio in terms of machine and time parameters. The work W_{max} that can be done by a given equipment in available time T_L may be written as the product of the equipment power utilized on this job P_u and T_L:

$$W_{\max} = P_u \cdot T_L,$$

where P_u is the time rate at which work is actually being done by the computer.

The amount of work W required to be done on the job requires time

† Consultant, 431 Ferne Ave., Palo Alto, Calif.

$$T_j = \frac{W}{P_u} + t_d; \qquad t_d = \text{down time.}$$

Thus,
the deadline ratio

$$= \frac{T_j}{T_L}\left(\frac{W}{P_u} + t_d\right) \Big/ T_L = \frac{W}{P_u T_L} + \frac{t_d}{T_L}$$

$$= \frac{W}{W_{\max}} + \frac{t_d}{T_L} = \text{work ratio} + \text{down-time ratio,}$$

For a situation wherein the work to be done in available time T_L is to be the maximum work that can be done by a machine operating at a given power (P_u), the down time t_d must be zero, and $W = W_{\max}$; *i.e.,* the deadline ratio is one.

This is characteristic of many so-called "real-time" situations. In this context, the following remarks are addressed to some problems of the design of equipment useful in situations whose deadline ratio is one or very close to one for both short and sustained periods of time. Let us first note some practical examples of such situations as well as the relevant activities of designers in this aspect of the field.

A most significant military deadline situation with deadline ratio equal to one is the situation in which the SAGE system is designed to operate. Here the value of T_L is large indeed. The central component of the SAGE air-defense system is a digital computer. Radar and other data are accepted and processed by the computer and a complete description of the air situation is prepared and presented to operators. The equipment is interrogated and gives commands to the external environment. In this application, as already noted, the system requirement is for *continuous* operation.

Every branch of the service has requirements for equipments that will operate without error during offensive or defensive operations. Inasmuch as the frequency and duration of these operations are, on the whole, unpredictable (*i.e.,* T_L may have a wide range of values) the problem of designing equipment to meet these requirements is, in a sense, no different from the problem of designing for continuous operation.

Some business-type applications impose requirements that are also severe. If payrolls are delivered late, or if some checks are incorrect, there will be grief, consternation, discontent, etc. When computing equipment is called upon to handle these deadline situations, one is again faced with the task of getting a prescribed amount of work done every day. In commercial-account banking applications there is the double requirement that a prescribed minimum amount of work must be done with the equipment each banking day, and some of the work must be done within certain prescribed hours of the day. For example, laws in certain areas require that checks that are to be refused payment must be returned by a specified time of the working day. Statement day is important to some of us. Here again one can imagine that it is imperative for the equipment to provide a certain minimum of error-free operation for a prescribed period prior to statement day.

In industrial control applications where an operating staff normally controls the factory processes inside the factory, many situations are characterized by deadline ratios of one for a certain period within the working day, which may be twenty-four hours in some cases. Arguments concerning the use of computers in industrial control revolve about the problem of obtaining equipment that can provide the required error-free operation. As long as such equipment is not available, it seems that the proper function of the computer can not be to control a plant directly but only to provide information to operators, managers, foremen, etc. Then if the computer fails, the factory, not being connected to the computer, will not blow up, and the people who are connected to the machine will remain intact and can continue operation and control of the plant on the basis of experience, meter readings, etc.

It is clear that there is and will be a large market for computers in situations characterized by deadline ratios of one for longer and longer durations. The characteristics of computers usable in these situations will also, incidentally, provide a basis for minimizing the cost of maintenance of these computers.

Performability

From the point of view of sustained performability[1] (usually called reliability), today's electronic-computer systems have been and are erratic, on the whole. The few who warned about the situation were shouted down, on the basis of the presumed initial high cost of implementing facilities designed to increase performability and maintainability. This charge might be proven about a substantial segment of the electronic industry. It would be interesting to know the cost to the Department of Defense for electronic equipment that was not economically maintainable or failed to work at all, in the last ten years. Some published estimates indicate that it has taken from 10-100 times the initial cost to maintain military equipment. The tide is turning, however. What is of deepest significance is the fact that it is presently quite in fashion to be concerned about performability. Whereas people until a year or two ago smiled benignly on the misguided zealots who insisted on the necessity for performability considerations in the design stage, today there is a 50:50 chance of their getting a hearing—in the military, that is. Commercial organizations are only now beginning to pay attention to the importance of performability considerations, mostly because of the belatedly recognized need for equipments to handle situations with deadline ratios close to one, and also because of the now officially recognized high cost of maintenance. Nevertheless, these computer manufacturers' sales departments still insist on and win the point that an

[1] It is recommended that the term "perfectly reliable" be reserved for *components* or *modules* of a system that are "failure-free" and that a *system* producing "error-free" results be designated "perfectly performing."

extra initial cost for performability will make their computers less competitive. It may also be noted in passing that the trend toward the concern for performability is partly a reaction to a situation where electronic equipment could not be operated economically and sometimes not at all, even in situations with deadline ratios much less than one.

This is not to say that there has not been any activity in the performability field, for there has. No doubt, many will recall the differences of opinion that raged in the early days—and newcomers to this field fight the battles all over again—among proponents of marginal checking à la Whirlwind, proponents of built-in checking for "everything" à la Univac and Raydac and later, Norc, proponents of duplication à la Binac, and proponents of programmed checking à la everyone who didn't have other facilities. Diagnostic and exercising routines and other techniques were later added to the arsenal of techniques for increasing performability. In recent years, more and more attention has been given to the development of reliable "long-life" components.

These techniques are categorized under various names: preventive maintenance, built-in checking, accuracy control, programmed checking, etc. Unfortunately, except for the effort to obtain perfectly performing equipments by having perfectly reliable components, connectors, and connections, none of the techniques that were developed and adapted for use in computers were part of an intgrated program to attain maximum performability. Even superficial observation will show that these techniques are each bits and pieces. By themselves, they help to do part of the required job.

To develop this argument, it may be useful to consider a detailed operational definition of performability of equipment. First, consider this definition. A perfectly performing equipment is one that produces error-free results during the period that the equipment is operated at its utilized power P_u. It follows from this definition that one would consider equipment perfectly performing if no errors resulted, even though components failed, the design was poor, cross talk prevailed, etc. One would consider that an equipment was not performing perfectly if errors resulted in spite of no component failure but because of an external transient, for example. Thus, the key consideration in the determination of performability from the user's viewpoint is the freedom from error in the results—in the output. It is *not* primarily in those faults of components or design or whatever that are usually responsible for errors. If we assume, on the other hand, that one has a well-designed equipment, that steady-state errors all result solely from components that failed, and that transients do not appear too frequently, then we can restate our definition of performability to include the designer's (and maintenance) viewpoint as follows: (ideally), a perfectly performing equipment is one that produces error-free results during the period that the equipment is operated at its utilized power with t_d, the down time, equal to zero, *i.e.*, uninterrupted continuous operation. It should be noted especially that we do *not* concentrate attention here on detailing the activities one engages in during the period designated "down time," but merely on how much time is involved in executing the activities necessary for getting the equipment back on the air. Thus, the implication of this definition is that one must consider that an equipment is performing perfectly—even if errors are actually made within the equipment—when one can do whatever is necessary to correct these errors (before they show up at the output) in zero time. One would have to consider that the equipment is not performing perfectly at utilized power if errors were not made, but there was a delay in obtaining results for some external reason.

If we detail and classify the activities necessary to put the equipment back on the air (the maintenance procedures), it will 1) allow us to write a detailed operational definition of performability from both the user's and the designer's viewpoints, 2) help to classify the activities that workers in the field have been engaged in, and indicate how to evaluate these as contributions to an integrated program to maximize performability, and 3) help to identify, classify, and set criteria for evaluation of any proposal for increasing performability of computer equipment.

Lost Time—Lost Work

Assuming a fault-free design and a fault-free program, down time on a computer consists of the time taken to detect errors, to locate the fault causing the error, to repair the fault, and to prepare for restarting the computer so that it can continue from the point of error. If there is reconstituting to be done, because the error was detected and the machine stopped during the execution of an instruction beyond the one in which the failure occurred, then one might have not only "lost time" but "lost work." If some work must be redone, then the time to redo this work must be added to the lost time due to error detection, failure location, and failure repair. To take this situation into account, we will modify our definition of deadline ratio (or deadline coefficient).

$$\text{Deadline ratio} = \frac{W}{W_{\max}} + \frac{t_d}{T_L} + \frac{W_r}{W_{\max}};$$

$$W_r = \text{amount of rework.}$$

$$= \underset{\text{ratio}}{\text{work}} + \underset{\text{ratio}}{\text{down time}} + \underset{\text{ratio.}}{\text{rework}}$$

A perfectly performing equipment operating at a given power P_u on a job will be characterized by having

$$W/W_{\max} \leq 1 \text{ and } \frac{t_d}{T_L} + \frac{W_r}{W_{\max}} = 0.$$

Thus, a detailed operational definition of perfect performability is the following: A practically perfect performing equipment is one that produces error-free results during the period that the equipment is operated at its utilized power with the total time taken for error detection, failure diagnosis, and failure repair vanishingly small, and where the amount of rework is vanishingly small.

Maximizing Performability by Already Developed Techniques

We are now in a position to identify exactly what part of an integrated program for maximizing performability previously developed techniques are designed to handle. One would like to be able to anticipate, if not prevent, component failures causing errors during operation of a computer. Thus, pre-aging and preselecting components, marginal checking, and test and exercising routines are all examples of techniques for *prevention, anticipation,* and *extrapolation* wherein one attempts to treat or test components under prescribed conditions in order to extrapolate the behavior of the components at a later time. In light of the fact that a large effort today is going into the development of very "long-life" reliable components, one should note that these preventive techniques assume "unreliable" components.

Others of the techniques mentioned earlier fall into one of the three classes: *error detection, failure diagnosis,* and *failure repair.* Parity checking, echo checking, duplicate equipment, and programmed transfers on alternate paths are all designed to *detect* errors in transfer or storage of information. Duplicate equipment, check arithmetic or logical units that work with appropriate algorithms, and program checks are used to *detect* errors in arithmetic or logic. Circuits have been built and programs written to *detect* errors in control and timing and also in storage-address selection.

Failure-diagnostic techniques have included the use of programs, component indicators, module indicators, and routine dynamic trouble-shooting techniques. While the *repair* of components that failed is almost universally done by manual replacement, automatic substitution for faulty subsystems has been done occasionally.

Other Techniques For Performability

We now come to a consideration of those approaches and techniques which if successfully implemented could indeed provide equipments with almost perfect performability and thence make them candidates in situations with deadline ratios equal to or close to one, *i.e.,* for computers with severe deadlines to meet. It has already been mentioned that perfectly reliable components can implement an adequately designed equipment with perfect performability, since then results should be error-free and the lack of component failure would result in no requirement for work rerun.

Error-correcting codes and facilities have the same property. If errors, although they do occur within the system, are corrected almost instantaneously and if little or no rerun is necessary, then systems with error-correcting facilities satisfy the criterion for practically perfect performability. Note that when the errors that are corrected are caused by components that failed, one can live with these in the equipment.

In the work of Moore and Shannon [1], and also of von Neumann [2], on the design of reliable systems from unreliable components, techniques are discussed that also have this characteristic in common with two other techniques mentioned below, namely that the correctness of the results is indifferent to faulty components that remain in the system. Thus, the equipment satisfies the criterion for perfect performability in spite of faulty components. Such circuits we shall call "indifference" circuits.

Other indifference circuits [3] are those characterized by a quad of diodes substituted for a single diode wherein the circuit behaves even with some diode failure exactly as a nonfaulty single diode would behave,

Another indifference circuit [4] has enough component redundancy so that output values coincide with the value of a majority opinion from among individual circuits performing identical functions. Here again the system is immune to some component failures, and erroneous results are not produced even though some components may have failed.

Self-Repair

In every approach mentioned above, except for the one of trying to get perfectly reliable components in the first place, it is assumed that there will be component failure, and an attempt is made to design around this by selecting a kind of redundancy that makes the system immune to component failure, *i.e.,* the faulty components remain in the equipment. Usually, we replace faulty components. The question arises as to whether or not we can satisfy the criterion for almost perfect system performability and still replace, rather than live with, faulty components. A self-repairing facility with the characteristics described below will do just that.

Consider a computer made up of a number of standard module types. Build in sufficient redundancy in equipment and code so that the error-detection function is performed; *i.e.,* storage, transfers, arithmetic and logic operations, address selections, timing, and control are checked for accuracy instantaneously. Error detection is made an integral part of the design. Design each module in such a way that the dynamic operation of each module is continually and automatically monitored, and failure is instantaneously sensed and indicated. When a failure is diagnosed, automatically switch in an appropriate "hot" spare module that is in the rack, and automatically restart at the beginning of the micro-operation that was being executed when the error was detected. If the switch fails, arrange for a spare module to be switched into a predetermined position by having the switch that failed so designed. (It may be noted incidentally that the "tilt" indicator on the package or switch that failed would indicate to a maintenance "boy" what modules needed replacement.) A self-repairing equipment, wherein the error detection, failure diagnosis, and failure-repair time (switching time) add up to a small value and the amount of rework (for the micro-operation) is likewise small, would come sufficiently close to satisfying the criterion for a practically perfectly performing equipment and would thus provide a place for self-repairing facilities in computers with severe deadlines to meet.

We do not believe that any one or the other of these approaches will ultimately prevail. We expect to see all of them employed, even in different parts of the same computer system, and used where there are severe deadlines to meet. In some applications, one technique will be more feasible technically; in other applications, one or the other of the techniques will be more economical.

Bibliography

[1] Moore, E. F., and Shannon, C. E. "Reliable Circuits Using Less Reliable Relays," *Journal of the Franklin Institute,* Vol. 26 (September, 1956), pp. 191-208 (October, 1956), pp. 281-297.

[2] von Neumann, J. *Probabalistic Logic and the Synthesis of Reliable Organisms from Unreliable Components.* ("Annals of Mathematical Studies," 34.) Princeton, N.J.: Princeton University Press, 1956, pp. 43-98.

[3] Creveling, C. J. "Increasing the Reliability of Electronic Equipment by the Use of Redundant Circuits," PROCEEDINGS OF THE IRE, Vol. 44 (April, 1956), pp. 503-505.

[4] Schneider, S., and Wagner, D. H. "Error Detection in Redundant Systems," *Proceedings of the Western Joint Computer Conference,* February 26-28, 1957.

Organizing a Network of Computers to Meet Deadlines

A. L. LEINER†, W. A. NOTZ†, J. L. SMITH†, AND A. WEINBERGER†

Introduction

A DEADLINE requirement on a data-processing job usually implies a high over-all problem-solving speed. When the basic computing components available at any time have inadequate top-speed limits, the system designer must attempt to produce the required problem-solving speed by organizing the components together in more powerful logical combinations.

One mode of approach is to try to hook together several computers into an interconnected data-processing network. In using such a network, the data-processing job is broken up into different pieces, and all the different computers in the network work on different pieces of the job simultaneously. Under ideal circumstances, the whole processing job could be split up into pieces that are completely independent. In actual practice, however, this ideal circumstance of complete independence usually cannot be attained because the results of one set of interim computations will, as a rule, be needed as input data for another set, and the commencement of one phase of the operations must await the conclusion of another. These interactions between the individual programs on the different computers in the network produce a higher order of system complication, and thus give rise to new design problems that are not ordinarily met with in the usual single-machine type of system.

This paper discusses techniques for attacking three important types of such design problems. The first type is concerned with the over-all efficiency of the workload-sharing scheme under which the computers operate, that is, the amount of productive time lost when one computer stands by idle, waiting for one of its collaborating computers to produce a certain item of data that the first machine needs in order to continue with its program. The paper shows how queueing-theory techniques can be applied to evaluating average time losses due to these causes, and to finding optimum parameters to use in programming mutual data exchanges.

The second type of design problem is concerned with the basic logic involved in keeping the different computers in logical step with each other so that each machine carries out its portion of the over-all job in proper order. For a complex system where a great deal of interlacing of steps takes place between the different units, systematic methods are needed for establishing clearly what the necessary and sufficient order of precedence among the machines is at every step. It is shown how problems of this nature can be attacked using matrix methods.

The third type of problem concerns the design of instruction systems for multicomputer networks that are easily within the coding capability of the average human programer. With a well-designed system the programer should not need to figure out detailed quantitative timing relationships, and should not have to do more than indicate the proper order in which certain critical stages in the over-all process are to take place. This paper describes the logic of a multiple-control system incorporated into a new dual-computer network now being designed at the National Bureau of Standards by the authors.

† National Bureau of Standards, Washington, D.C.

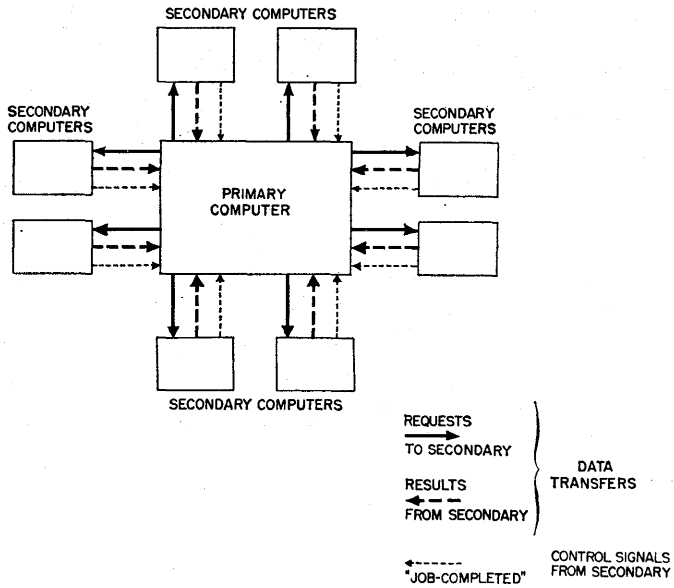


Fig. 1—Schematic of multicomputer network system.

Before discussing these three types of design problems, it is necessary to consider some of the various possible schemes for sharing workloads in a multicomputer network.

Varieties of Workload-Sharing Systems

Fig. 1 shows a schematic plan of a model multicomputer network system. The *primary computer* located in the center shares its workload with one or more *secondary computers* connected to it via data-transmission lines. Three types of intercommunication between the primary and secondary computers are indicated. They correspond respectively to 1) *outgoing data* from a primary to a secondary computer, 2) *incoming data* from a secondary, and 3) *control signals* from a secondary to a primary computer.

In order to illustrate the basic principles more clearly, let us examine in detail a simple special case of such a network system that contains only two independent computers, that is, a primary and a single secondary (see Fig. 2). The distinction between the primary and the secondary computer is based upon the rules by which the initiative for carrying out interchanges between the two types of machines is assigned. Although both types are considered as independent, internally programmed computers, the primary is the one that initiates the joint transactions between the two machines. It does this by transmitting, from time to time, a *request* (or job assignment) which orders the secondary to carry out some specified job. The secondary thereupon carries out the indicated job, while the primary in general continues, performing computations that do not need the data (results) currently being worked on by the secondary.

Let us now consider several possible dynamic models for such a system that differ in the manner in which the secondary computer behaves after it has completed the job assigned to it.

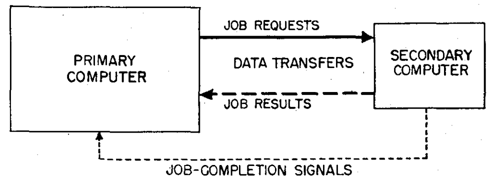


Fig. 2—Schematic of dual-computer network.

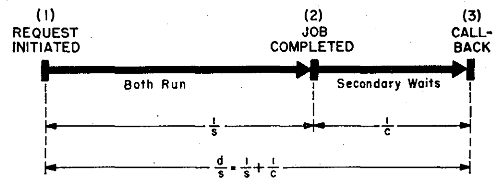


Fig. 3—Sequence of events using explicit job-completion signals.

Case 1: Use of Explicit Job-Completion Signals

With this model, when the secondary computer finishes its assigned job, it halts and transmits a "job completed" signal to the primary. The primary computer, however, may not be at a stage in its program where it can accept the offered data, in which case a delay ensues, during which the primary continues to run while the secondary remains idle, awaiting further orders from the primary. As soon as the primary reaches a suitable point in its program following receipt of a "job completed" signal, it issues a "call-back" order to the secondary and then immediately accepts the data that is returned to it from the secondary. Following acceptance of the result, the secondary remains idle until the primary transmits another request to it. This three-phase cycle of request signal, job-completed signal, and call-back signal, etc., continues indefinitely (see Fig. 3).

The simple model just described is a fairly satisfactory approximation for many practical job-sharing applications. For some types of usage, however, certain additional complications need to be taken into account. These complications arise when the nature of the program being carried out is such that the primary computer program cannot be formulated so as to schedule its job requests to the secondary far enough ahead in the primary program. As a result it may become impossible for the primary to be kept busy doing useful work during the entire time that the secondary is working to fulfill the request. In the preceding model it was implicitly assumed that the primary program could always be arranged so that the primary computer had an indefinite amount of useful work to do while the secondary was working on the last job request, that is, that the primary would always continue working at least until the secondary returned its job-completed signal. When, after issuing a request, the primary can proceed only a limited distance through its program before the lack of receipt of a reply from the secondary constitutes an absolute barrier to further progress, it no longer becomes profitable to assign the prime initiative for the return of the data to the secondary computer. There-

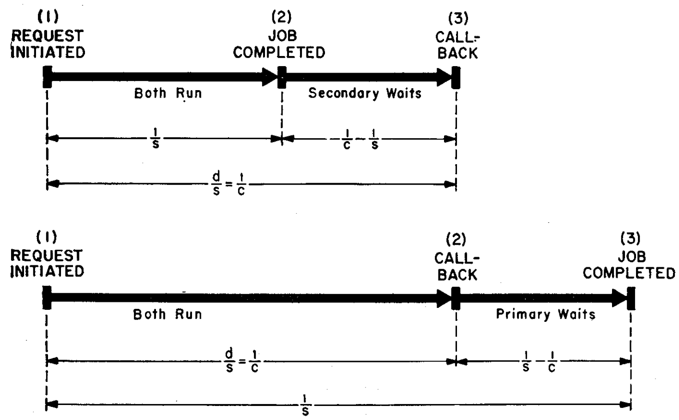


Fig. 4—Alternative sequences of events using estimated job-duration intervals.

fore, let us now consider a model in which no explicit job-completed signal is issued by the secondary.

Case 2: Use of Estimated Job Durations

With this model, the primary computer issues its job request to the secondary in the same manner as before. The primary continues running for a certain period of time and then issues a call-back signal to the secondary. This running period (or call-back interval) has as its upper bound the time needed for the primary to exhaust its current supply of working data. In order to approach optimum efficiency, the programmer will attempt to make this *average* call-back interval not too different from his best estimate of the *average* secondary job duration. It is assumed, however, that exact matching of an individual call-back interval to its corresponding job duration is not possible, and that in the long run, fixed percentages of overestimates and underestimates will be found to occur randomly. If the secondary is not yet finished with the job at the time of the call-back, the primary waits idle until the secondary is ready. As soon as the secondary finishes the job, the primary immediately accepts the returning data and resumes operation. Of course, in some instances the secondary will succeed in finishing the job before the primary issues its call-back signal. In this event the secondary waits idle until the primary calls back the job-result and issues a new job request. Under either circumstance one or the other computer spends a certain part of its time idle. Therefore, at any instant we should expect to find either the primary computer halted, the secondary halted, or neither one halted, but never both halted simultaneously (see Fig. 4).

Multiple Overlapping Assignments

Up to this point we have considered only systems in which the primary program issued requests and call-backs in strict alternation; that is, the primary program was so formulated that a second job request was never made before calling back the results from the previous job request. Let us now consider models in which the primary computer may load a sequence of several job requests on to the secondary computer in succession before calling for the results of a previous request. The introduction of these multiple overlapping job assignments for the secondary computer introduces a new consideration into the picture, namely the *storage capacity* of the secondary computer.

In the previous examples, the dynamics of the system were determined wholly by the rates at which the two computers issued their request signals, job-completed signals, and call-back signals, respectively. Although certain values for these parameters evidently resulted in much longer idle periods than other choices, it was clear that those idle periods always terminated themselves spontaneously in the natural course of events. In the present case, however, precautions must be taken against permanent tie-up of both programs. These situations can arise because the capacity of the secondary computer for accepting requests and storing finished results is necessarily limited. Because of this, the primary must not issue more than some specific number n of requests in succession (which load the secondary storage) without also issuing some intervening call-backs (which unload the storage). More precisely, if the total number of requests issued by the primary since the last idle waiting period of the secondary exceeds the total number of call-backs during the same period by more than n, both computers must halt in a mutual deadlock. The secondary must halt without accepting the new request because it has no more room for accepting new data until some of its previously calculated results are removed by the primary. The primary must halt because it cannot proceed with its program to possible future call-back orders without first disposing of its current request. Thus, an insoluble deadlock ensues that cannot be resolved without outside intervention. Hence the need exists for observing a program convention that recognizes a specific upper bound, n, on the cumulative excess of requests over call-backs. This bound is not in any way dependent upon computer speeds or timing relationships, but is entirely a function of the readily determinable storage capabilities of the secondary computer. In the following analysis, observance of this programming convention is assumed.

Case 3: Multiple Assignments, with Job-Completion Signals

This case is similar to Case 1 except that the primary may transmit a series of requests to the secondary without intervening call-backs, as long as the restriction on the maximum number, n, of current undelivered jobs in the secondary is observed. The primary calls back all jobs in the same sequence in which they were originally requested, but the actual number of current jobs stored (unprocessed or undelivered) in the secondary may fluctuate from time to time between 0 and n. Because of the assumption about job-completion signals, the primary never stands by idle waiting for the secondary to finish with a particular job.

Case 4: Multiple Assignments, with Estimated Job Durations

This case corresponds to Case 2 except that up to n current assignments may be in storage in the secondary. Because no explicit job completion signals are available (or, at any rate, helpful) the primary computer may from time to time try (or need) to call back a particular job result from the secondary before the job is ready. In this event, the primary waits idle until the secondary is finished with that job. During this idle period, of course, the primary program is halted and hence no new requests or call-backs can be issued.

Efficiency of Workload-Sharing Systems

The foregoing four job-sharing schemes can all be comprised in a single abstract dynamic model suitable for numerical analysis. For this purpose the following *states* are defined, in which the system can reside from time to time. These states are completely definable for our purpose merely by specifying two parameters, x and y, as follows. Let $x =$ the number of jobs in the secondary that have not yet been completed, and $y =$ the number of jobs in the secondary that have already been completed, but have not yet been called for by the primary.

It is understood that the programming convention mentioned earlier is observed; hence $x + y \leq n$.

Now define three parameters, r, s, and c, that specify the dynamics of the frequency of transition between states. Let $r =$ the request rate, that is, the average number of requests per unit time made by the primary *during the periods when it is running and able, under the rules, to issue requests.* (Obviously, the request rate when the computer is halted or not permitted to make requests is zero; these periods are not included in computing the value of r. Similar remarks apply to the remaining parameters, s and c.) $s =$ the average number of jobs completed per unit time by the secondary during the periods when it is running, and $c =$ the average number of call-back signals issued per unit time by the primary during the periods when it is running and able, under the rules, to issue call-backs.

Among the major items of interest is the fraction of the time that each of the two computers runs. A universal objective function that an "optimum" system should tend to maximize is not easy to define, but a reasonable set of performance parameters for provisional use would appear to be: $W_1 =$ the fraction of the time that the primary computer is running, $W_2 =$ the fraction of the time that the secondary computer is running, $R = W_1 + W_2 =$ the net *productivity* of the system.

Under ideal circumstances (for example, in processing a problem that can be so formulated that no interchanges whatever are required between the two computers) both machines can run continually, and $W_1 = W_2 = 1$, $R = 2$. Hence the quantity $E = \frac{1}{2}R$ might be considered a direct measure of the over-all *efficiency* of the system relative to the same two computers operating concurrently in complete independence.

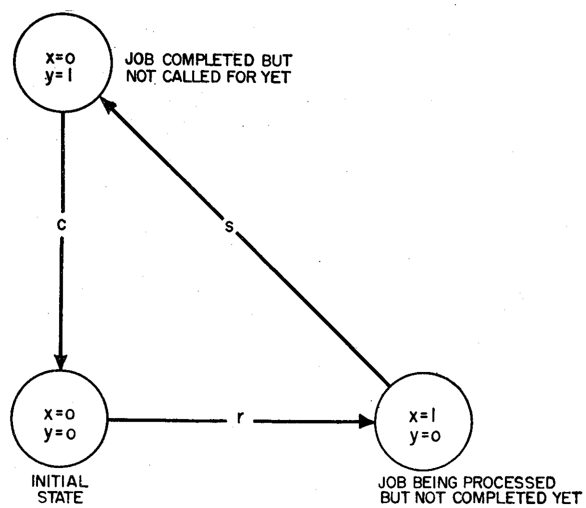


Fig. 5—Simple state diagram.

Analysis of a Simple System (Case 1)

The over-all structure of a very simple system (Case 1) in terms of these parameters is illustrated in Fig. 5. The system starts in the initial state $[x = 0, y = 0]$, primary computer running. Issuance of a request throws the system into state $[x = 1, y = 0]$. These exit transitions occur with average frequency r per unit time when the system is in state [0,0]. If the system is in state [0,0] a certain fraction of the time, denoted by $P(0,0)$, the over-all average frequency of occurrence of such transitions, averaged over-all time, is $rP(0,0)$. In the same way, transitions from state (1,0) to state (0,1) occur with over-all frequency $sP(1,0)$ and from (0,1) to (0,0) with frequency $cP(0,1)$.

Clearly, unless a system becomes deadlocked, the average number of transitions into any given state in the long run becomes equal to the number of transitions out of that same state. Hence, we have the relations:

$$\begin{aligned} rP(0, 0) &= sP(1, 0) \\ sP(1, 0) &= cP(0, 1) \\ cP(0, 1) &= rP(0, 0) \\ P(0, 0) + P(1, 0) + P(0, 1) &= 1. \end{aligned}$$

These may be written more systematically as follows:

$$\begin{aligned} -rP(0, 0) + cP(0, 1) \qquad\qquad &= 0 \\ - cP(0, 1) + sP(1, 0) &= 0 \\ +rP(0, 0) \qquad\qquad - sP(1, 0) &= 0 \\ P(0, 0) + P(0, 1) + P(1, 0) &= 1. \end{aligned}$$

One of the first three equations is evidently redundant. Solutions for the P's are readily found to be

$$P(0, 0) = \frac{\dfrac{s}{r}}{1 + \dfrac{s}{r} + \dfrac{s}{c}}$$

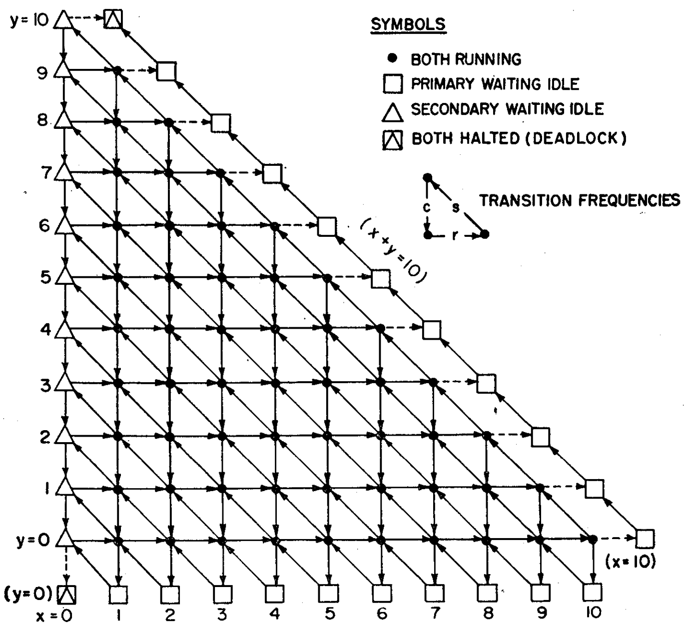


Fig. 6—Complete state diagram, Case 4, $n = 10$.

$$P(0, 1) = \frac{\frac{s}{c}}{1 + \frac{s}{r} + \frac{s}{c}}$$

$$P(1, 0) = \frac{1}{1 + \frac{s}{r} + \frac{s}{c}} \cdot$$

The corresponding parameters become

$$W_1 = P(0, 0) + P(1, 0) + P(0, 1) = 1$$

$$W_2 = P(1, 0) = \frac{1}{1 + \frac{s}{r} + \frac{s}{c}}$$

$$R = W_1 + W_2 = 1 + \frac{1}{1 + \frac{s}{r} + \frac{s}{c}} \leq 2$$

$$E = \tfrac{1}{2}R \leq 1.$$

The interpretation of these performance parameters will be postponed until later.

Analysis of General Systems (Cases 2, 3, and 4)

Using the principles of notation, just illustrated, a more general system (Case 4, $n = 10$) is formulated graphically in the state diagram in Fig. 6. In order to conserve space, the values of x and y are omitted from the actual state symbols, which are shaped according to a symbol code that indicates whether each computer is running or halted. Fig. 6 illustrates graphically the necessity for the programming convention that holds $x + y \leq n$, mentioned earlier. Dotted lines originating from states on the hypotenuse of the triangle represent the transitions forbidden by the convention, which inevitably lead to deadlock at $[1,n]$. Transition frequencies for these instances are arbitrarily set at zero. Another obviously illogical move on the part of the program would be the transition indicated immediately below $[0,0]$, which may also be assumed to occur with zero frequency.

It will be recalled that Case 3 differed from Case 4 primarily in respect to use of a "job-completed" signal from secondary to primary which served to avoid the possibility of premature call-back attempts by the primary ever taking place. Under Case 3, call-backs never take place when the number of jobs already completed by the secondary, y, is equal to zero. The state diagram for Case 3 differs from Case 4 only in not containing the idle primary states shown on the bottom row of Fig. 6 (square box symbols). Figs. 8 and 9 respectively (p. 120) show the state diagrams for Case 3 and Case 4 for the value of the storage capacity parameter $n = 2$. The still simpler Case 2 can be derived by comparable deletions of Case 4 states (see Fig. 10).

Following the previous method, the equations for all three cases can be written down immediately from inspection of the diagrams. Tables I-III give the coefficients of the unknowns $P(x,y)$ in matrix form. The right-hand side of each equation is equal to zero. An additional equation

$$\sum P(x, y) = 1$$

is understood to be implied in order to remove the underdetermination of the system. Tables IV-VI contain, for all cases, literal expressions proportional to the values of all of the unknowns.

Delivery Delay Parameters

In order to simplify the interpretation of the performance parameters, it is desirable to replace the rate parameters r, s, c, with another set of (reciprocal) parameters that specify time intervals rather than rates. It is convenient to normalize the parameters by measuring all time intervals in units of *the average time it takes the secondary to complete a job.* Hence, let $b = s/r =$ the average *request interval,* that is, the time interval between successive requests during the periods in which the primary is able, under the rules, to make requests.

The reciprocal of this quantity, $u = 1/b = r/s$ is a measure of the degree of utilization that the primary attemps to make of the secondary. If $u = 0$, the primary completely ignores the existence of the secondary; if $u = \infty$, the primary always transmits a request to the secondary the first possible instant it becomes permissible under the rules to do so.

Let $d =$ the average *desired delivery delay,* that is, the time interval between the issuance of a job request by the primary and the calling back (or attempted calling back) of the job result by the primary. This quantity is not necessarily equal to the actual delivery delay encountered, since the primary may in some instances be forced to wait

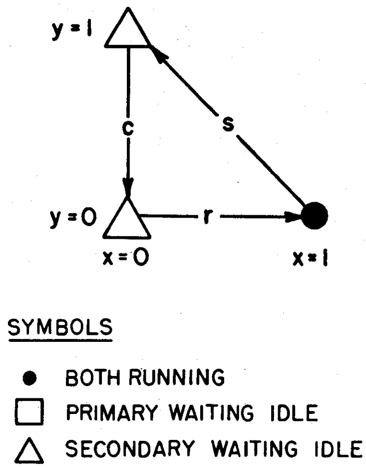


Fig. 7—State diagram for Case 1.

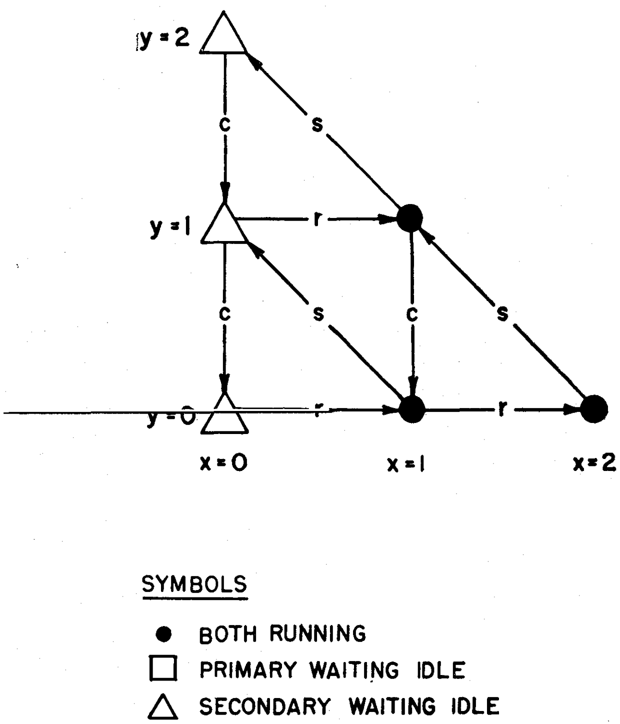


Fig. 8—State diagram for Case 3, $n = 2$.

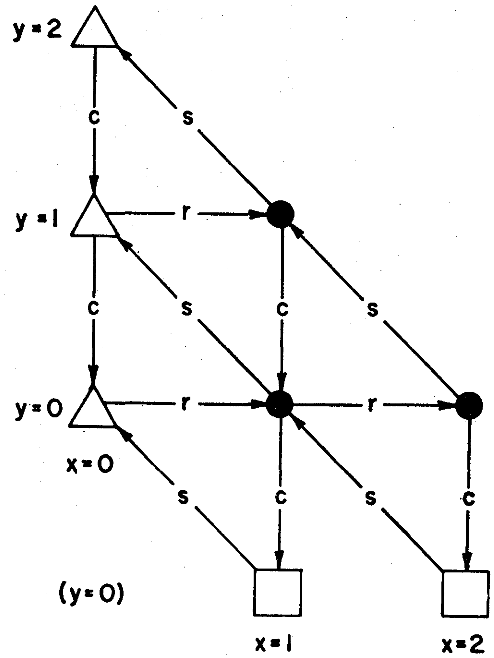


Fig. 9—State diagram for Case 4, $n = 2$.

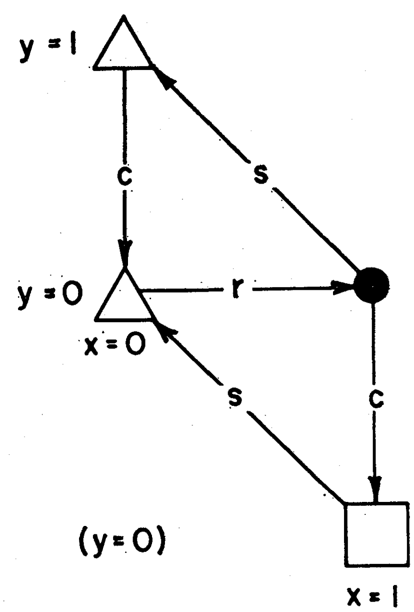


Fig. 10—State diagram for Case 2.

TABLE I*

Coefficients of $P(x, y)$, Case 2

$(x, y) =$	$(1, 0)$	$(0, 0)$	$(0, 1)$	$(1, -)$
	$-(s+c)$	$+r$		
		$-r$	$+c$	$+s$
	$+s$		$-c$	
	$+c$			$-s$

* The notation $(x, -)$ represents the states with $y=0$, primary waiting.

TABLE II

Coefficients of $P(x, y)$, Case 3, $n=2$

$(x, y) =$	$(0, 0)$	$(0, 1)$	$(0, 2)$	$(1, 0)$	$(1, 1)$	$(2, 0)$
	$-r$	$+c$				
		$-(r+c)$	$+c$	$+s$		
			$-c$		$+s$	
	$+r$			$-(r+s)$	$+c$	
		$+r$			$-(s+c)$	$+s$
				$+r$		$-s$

TABLE III*

COEFFICIENTS OF $P(x, y)$, CASE 4, $n=2$

$(x, y)=$	(1, 0)	(2, 0)	(1, 1)	(0, 0)	(0, 1)	(0, 2)	(1, −)	(2, −)
	$-(r+s+c)$		$+c$	$+r$				$+s$
	$+r$	$-(s+c)$						
		$+s$	$-(s+c)$		$+r$			
				$-r$	$+c$		$+s$	
	$+s$				$-(c+r)$	$+c$		
			$+s$			$-c$		
	$+c$						$-s$	
		$+c$						$-s$

* *Note:* The notation $(x, -)$ represents the states with $y=0$, primary waiting.

TABLE IV*

VALUES OF $P(x, y)$, CASE 2

(x, y)	$KP(x, y)$	Action
(1, 0)	1	Both run
(0, 0)	$\frac{s+c}{r}$	Secondary idle
(0, 1)	$\frac{s}{c}$	
(1, −)	$\frac{c}{s}$	Primary idle

* *Note:* $P(x, y)=KP(x, y)$ in Table $\div \sum KP(x, y)$.
K = normalizing factor.

TABLE V*

VALUES OF $P(x, y)$, CASE 3, $n=2$

(x, y)	$KP(x, y)$	Action
(0, 0)	1	Secondary idle
(0, 1)	$\frac{r}{c}$	
(0, 2)	$\left(\frac{r}{c}\right)^2$	
(1, 0)	$\frac{r}{s}$	Both run
(1, 1)	$\left(\frac{r}{s}\right)\left(\frac{r}{c}\right)$	
(2, 0)	$\left(\frac{r}{s}\right)^2$	

* *Note:* $P(x, y)=KP(x, y)$ in Table $\div \sum KP(x, y)$.
K = normalizing factor.

TABLE VI*

VALUES OF $P(x, y)$ FOR CASE 4, $n=2$

(x, y)	Values of $KP(x, y)$	Action
(1, 0)	1	Both run
(2, 0)	$\left(\frac{r}{s+c}\right)$	
(1, 1)	$\left(\frac{r}{s+c}\right)\left(\frac{s}{c}+\frac{s}{r+s+c}\right)$	
(0, 0)	$\left(\frac{s}{r}\right)\left(\frac{s}{s+c}\right)+\left(\frac{c}{r}\right)\left(1+\frac{s}{r+s+c}\right)$	Secondary idle
(0, 1)	$\left(\frac{s}{r+s+c}\right)+\left(\frac{s}{c}\right)\left(\frac{s}{s+c}\right)$	
(0, 2)	$\left(\frac{s}{r+s+c}+\frac{s}{c}\right)\left(\frac{s}{c}\right)\left(\frac{r}{s+c}\right)$	
(1, −)	$\left(\frac{c}{s}\right)$	Primary idle
(2, −)	$\left(\frac{c}{s}\right)\left(\frac{r}{s+c}\right)$	

* $P(x, y)=KP(x, y)$ in Table $\div \sum KP(x, y)$
K = normalizing factor.

an additional period for the secondary to complete the job. It does, however, serve as a lower bound for the actual average delivery delay, which is not known in advance. This *desired delivery delay, d,* represents an operational requirement laid upon the system by the nature of the program and, like the parameter n, describes the structure of the program rather than the dynamic performance of the dual-computer network. From this point of view, d is seen to be equal to the number of (secondary) job-completion periods that the primary can run after making a job request before it exhausts its supply of useful data to work on. Figs. 3 and 4 show that d is related to the previously used parameters in the following way:

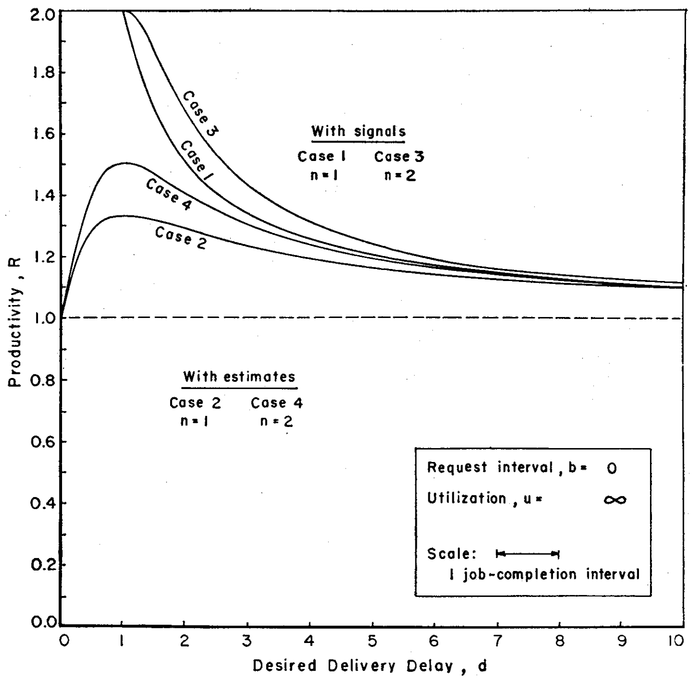


Fig. 11—Productivity, R, as a function of desired delivery delay, d, for $b = 0$.

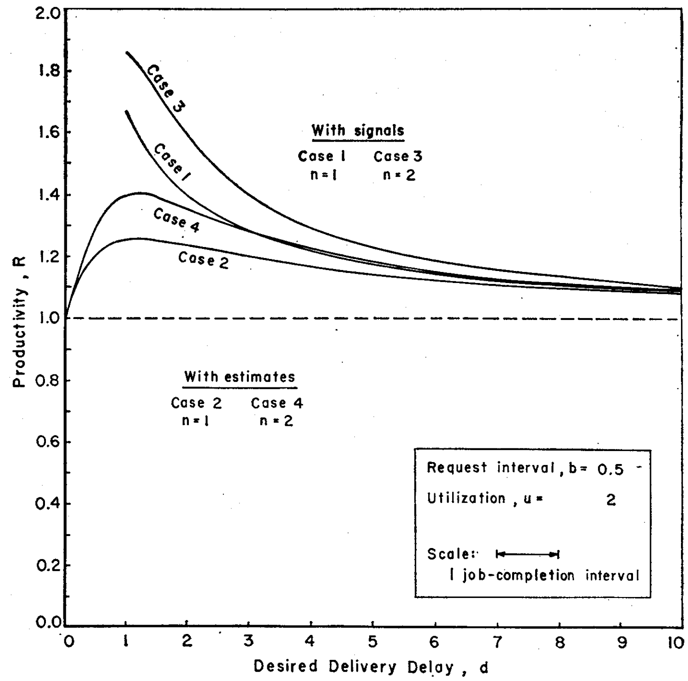


Fig. 12—Productivity, R, as a function of desired delivery delay, d for $b = 0.5$.

Cases 1 and 3 (explicit job-completion signal):

$$d = 1 + \frac{s}{c}$$

Cases 2 and 4 (estimated job-completion period):

$$d = \frac{s}{c}.$$

Replacing the r, s, c, parameters by the b, u, d parameters requires the following substitutions:

$$\frac{r}{s} = u, \text{ and}$$

$$\frac{s}{c} = d - 1, \text{ for Cases 1 and 3, or}$$

$$\frac{s}{c} = d, \text{ for Cases 2 and 4.}$$

The transformed formulas for R resulting from these substitutions in Tables IV-VI are illustrated by the curves in Figs. 11 through 16.

Another performance parameter of interest is the average *actual delivery delay* (measured in units of one average secondary job-completion time). Let D = the average actual delivery delay, that is, the time interval between issuance of a job request by the primary and the actual receipt of the completed job results by the primary.

Suppose that successive jobs are numbered consecutively with a serial number $i = 1, 2, 3$, etc. Let t_i = the length of time that the secondary works to complete the ith job; v_i = the length of time that elapses between the issuance of the ith request and the calling back of the ith request by the primary; D_i = the actual delivery delay for the ith job.

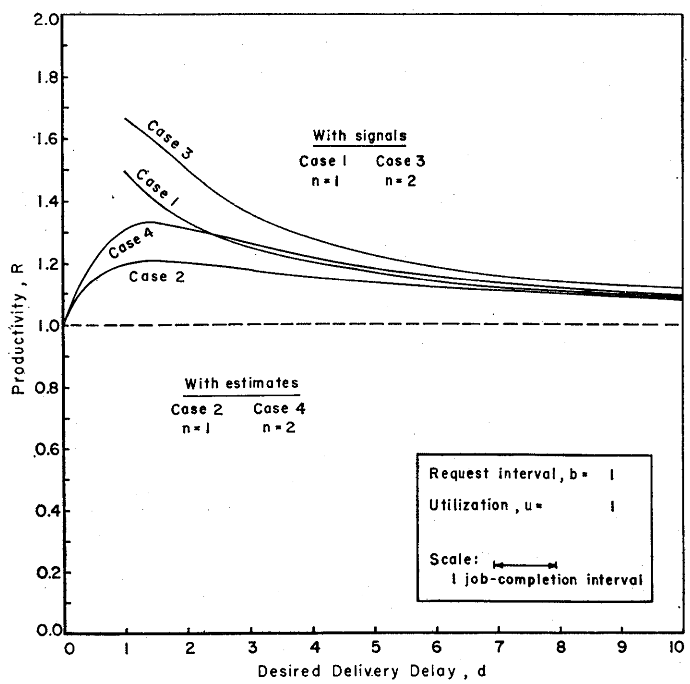


Fig. 13—Productivity, R, as a function of desired delivery delay, d, for $b = 1$.

The actual delivery delays in the various cases are as follows:

Case 1 and Case 3,

$$D_i = sv_i.$$

Case 2 and Case 4,

$$D_i = sv_i, \quad \text{if} \quad v_i \geq t_i, \text{ or}$$
$$= st_i, \quad \text{if} \quad v_i \leq t_i.$$

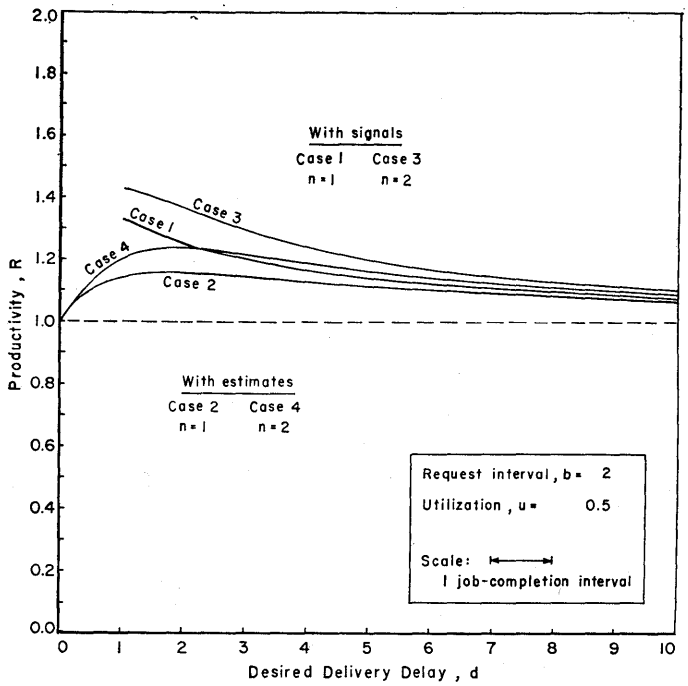


Fig. 14—Productivity, R, as a function of desired delivery delay, d, for $b=2$.

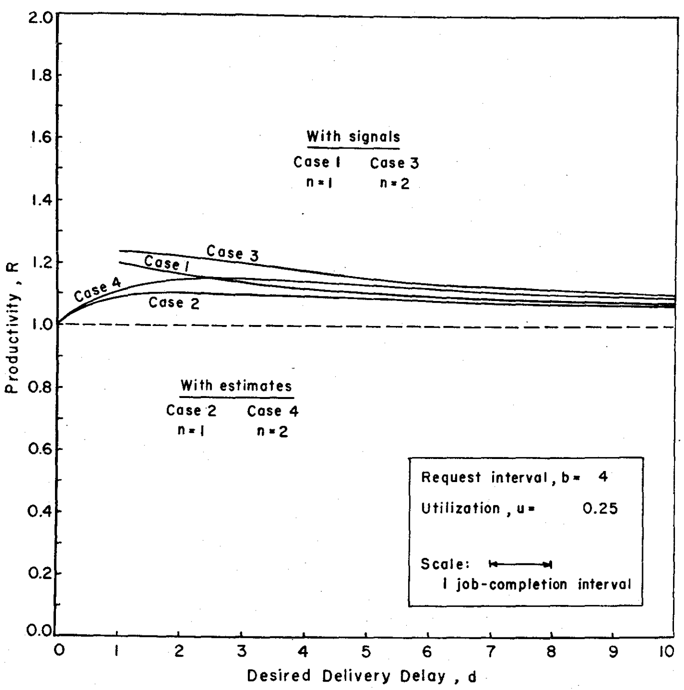


Fig. 15—Productivity, R, as a function of desired delivery delay, d, for $b=4$.

In Cases 1 and 3, $D=s\cdot\bar{v}$, where $\bar{v}$ is the average value of v_i for $i=1, 2, 3, \cdots \infty$.

In order to evaluate this performance parameter, it is necessary to make some assumptions about the frequency distribution of the various time durations between transitions.

Let $p(st,cv)dtdv=$ the frequency distribution function defining the proportion of job cycles for which

$$t \leq t_i \leq t+dt \text{ and } v \leq v_i \leq v+dv.$$

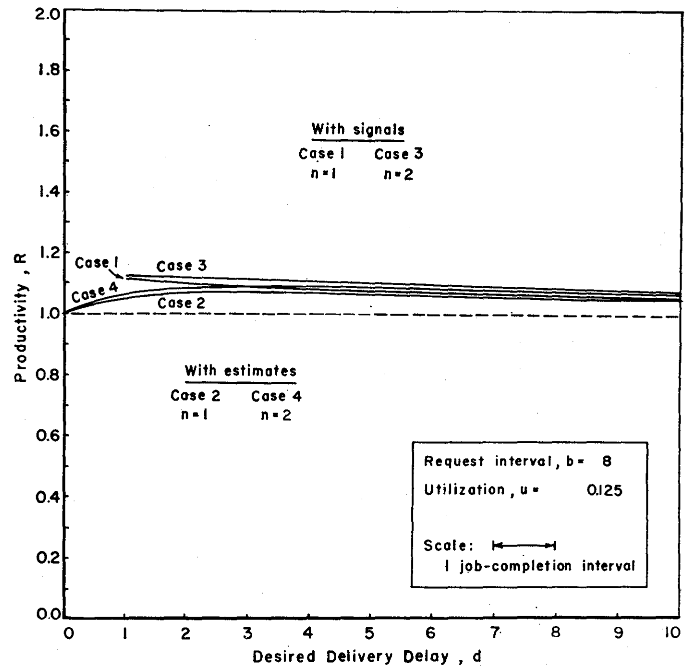


Fig. 16—Productivity, R, as a function of desired delivery delay, d, for $b=8$.

Then, for Cases 1 and 3

$$D = s\bar{v} = s\int_0^\infty \int_0^\infty vp(st,cv)dtdv.$$

In Cases 2 and 4

$$D = s\int_0^\infty \int_0^v vp(st,cv)dtdv + s\int_0^\infty \int_0^t tp(st,cv)dvdt. \quad (1)$$

Let us now consider two simple types of frequency distribution for the parameters v and t that may occur in practice. The first type is the simple case where $v_i=\bar{v}$ and $t_i=\bar{t}$, that is, constant for all i. We obtain immediately for Cases 1 and 3, (Fig. 3),

$$D = s\left(\frac{1}{s}+\frac{1}{c}\right)$$

$$= 1+\frac{s}{c}$$

$$= d \geq 1.$$

This constant-value type of distribution for v_i and t_i does not give rise to the Case 2 and Case 4 types of state diagrams.

The second type of distribution that is likely to be approximated in practice is the Poisson distribution. This distribution will apply if the decision that causes the transition out of a given state is the result of regularly repeated random trial drawings from populations containing fixed proportions of "request," "call-back," "complete job," or "continue in present state" commands peculiar to each of the principal types of states. For the Poisson distribution,

$$p(st,cv)dtdv = sc\, e^{-st-cv}\, dtdv. \quad (2)$$

Substituting (2) into (1), we find for Cases 2 and 4,

$$\frac{D}{s^2c} = \int_0^\infty \int_0^v e^{-st-cv} v\,dt\,dv + \int_0^\infty \int_0^t e^{-st-cv} t\,dv\,dt.$$

Putting $t=v'$ and $v=t'$ in the second integral, it becomes

$$\int_0^\infty \int_0^{v'} e^{-ct'-sv'} v'\,dt'\,dv',$$

which differs from the first only by the interchange of c and s.

Performing the integrations we find

$$\frac{D}{s^2c} = \frac{1}{sc^2} - \frac{1}{s(s+c)^2} + \frac{1}{cs^2} - \frac{1}{c(s+c)^2}$$

$$D = \frac{s}{c} + \frac{1}{1+\frac{s}{c}}$$

$$= d + \frac{1}{d+1} \text{ for Cases 2 and 4.} \qquad (3)$$

We see from (3) that $d \leq D \leq d+1$, that is, the average actual delay in practice is usually greater than the average desired delay but not by more than one job completion time. Results for the two kinds of parameter distributions are illustrated in Fig. 17.

Multiplicity of States and Transitions

It is evident from inspection of the complete state diagram (Fig. 6) that the actual path that a representative point follows through the phase space may be quite complex. Actual count on the diagram shows that within the stated rules $n(n-1)/2$ states have 3 possible exits, $2n-1$ states have 2 possible exits, and $n+2$ states have 1 possible exit. The total number of permissible states is $(n+1)(n+4)/2-1$; the total number of permissible transitions between states is $n(3n+7)/2$. For example, if $n=20$, these last two numbers are 251 and 670 respectively. Thus if the secondary is capable of storing up to 20 job requests, then at any moment up to 251 possible combinations of assignments may be in effect and any one of up to 670 possible events may happen next. It is clear, therefore, that even for moderate values of n, systematic methods are needed for insuring that events take place in proper logical sequence.

Keeping the Computer Programs in Step

Let us now consider some of the problems involved in keeping accurate control of the relative priority of events.

In preparing a program for a multicomputer network, certain priority requirements will usually be clearly apparent to the programmer from the basic logic of the

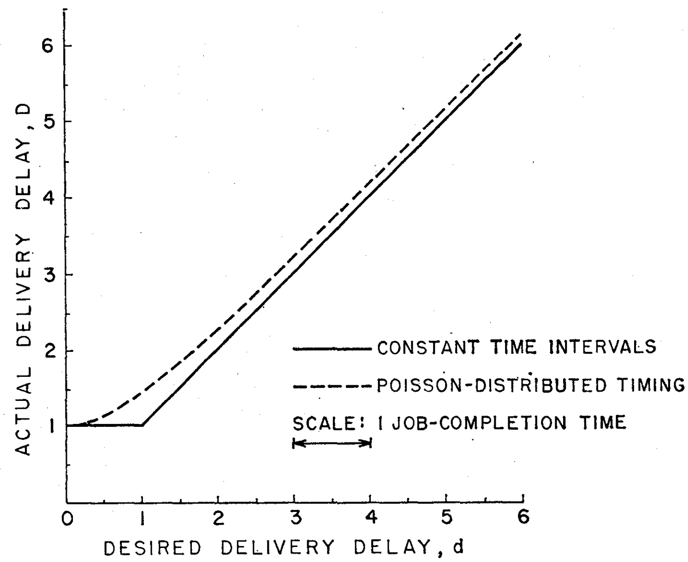


Fig. 17—Actual delivery delays for two kinds of parameter distributions.

problem he is trying to process. He will, for example, ordinarily be able to identify certain key steps (instructions or operations) in his program as having a definite priority relation to certain others. Thus he can label these various selected steps in the primary and secondary programs with *explicit* priority requirements. In imposing these requirements explicitly, however, he is simultaneously imposing certain other priority restrictions on the program *implicitly*.

To illustrate this point, let us consider the following list in which the programmer has imposed four explicit priority requirements. It is assumed that various steps in the primary and secondary programs have been arbitrarily designated with numbers 1, 2, 3 . . ., where the number sequence does *not* necessarily correspond to the desired program sequence.

(a) Step 1 precedes steps 2, 6, and 8,
(b) Step 4 precedes steps 2 and 9,
(c) Step 7 precedes steps 3, 4, and 6,
(d) Step 8 precedes step 9.

This list, however, is in a sense incomplete as written because the stated conditions, taken together, imply certain other conditions that are not explicitly stated. For example, the explicitly imposed conditions (a) and (d) together implicitly generate a fifth condition, namely:

(e) Step 1 precedes step 9.

Condition (e) was not realized beforehand by the programmer because steps 1 and 9 did not have any immediately obvious connection with each other.

The question naturally arises as to what systematic methods are available for detecting all of the implicit priority conditions and guarding against inadvertent contradictory conditions in a way that insures that no necessary conditions are overlooked and that no unnecessary conditions are imposed.

Matrix Representation of Priority Conditions

One method of attacking this problem is provided by elementary matrix theory. It is based on a correspondence between the rules for implied priority conditions and the rules for matrix multiplication.

The derivation of the implied priority conditions rests on the self-evident principle: if X precedes Y, and Y precedes Z, then X precedes Z.

The explicit priority conditions can conveniently be displayed in matrix form according to the following convention. Let each of the N steps under consideration be numbered arbitrarily with an index number $i = 1, 2, \ldots N$. Denote the fact that the step no. i precedes step no. j (an explicit statement) by setting up a matrix that contains a 1 in the element in the ith *row* and jth *column* (but not vice versa), that is, $M_{ij} = 1$ but $M_{ji} = 0$. Also, we never say "i precedes i"; hence $M_{ii} = 0$ always. Fig. 18 illustrates the matrix **M** for the four explicit conditions (a) through (d). (Zero elements are left blank in the figure.)

The process of generating the implied conditions consists of repeated exhaustive application of the priority principle mentioned previously, which may be restated in the following way. Let $\mathbf{M}^{(2)}$ be a matrix derived from the original matrix **M** by the following procedure: $M_{ij}^{(2)} = 1$ if, and only if, in **M** there exist at least one pair of elements M_{ik} and M_{kj} both of which $= 1$. This is merely a paraphrase of the statement: if i precedes k and k precedes j, then i precedes j. It may also be stated in the form

$$M_{ij}^{(2)} = \sum_{k=1}^{N} M_{ik}M_{kj}, \quad (4)$$

where the M_{ij} are variables in Boolean algebra which take on only the values 0 or 1, and the addition and multiplication operations are the Boolean operations of disjunction and conjunction, respectively. Comparison of (4) with the rule for ordinary matrix multiplication shows that

$$\mathbf{M}^{(2)} = \mathbf{M}^2.$$

Hence $\mathbf{M}^2$ gives additional implied conditions. The effect, if any, of adding these new statements to the original explicit statements can be found by adding $\mathbf{M}^2$ to **M**. By performing another multiplication, however, we can find

$$\mathbf{M}^3 = \mathbf{M} \cdot \mathbf{M}^2,$$

which may reveal still further implied conditions. We continue in this way until $\mathbf{M}^p = 0$, at which point we may stop. The complete set of conditions is then given by

$$\mathbf{H} = \mathbf{M} + \mathbf{M}^2 + \mathbf{M}^3 + \cdots + \mathbf{M}^p.$$

Depending on the facilities for computation available, several slightly different sequences of operations from the sequences described may be used for computing the same matrix **H**, but the essential point of procedure is that the matrix **H** contains the full set of priority conditions. Furthermore, **H** provides a check on the inner consistency of the explicit statements because of the requirement that the statement "i precedes i" must never appear; that is, the principal diagonals must remain equal to 0 in all matrices. Matrix theory shows that all powers of **M** of higher degree than some number $\mu \leq N$ can be expressed as a sum of lower powers. Hence, the process can be terminated after not more than $N - 2$ steps.

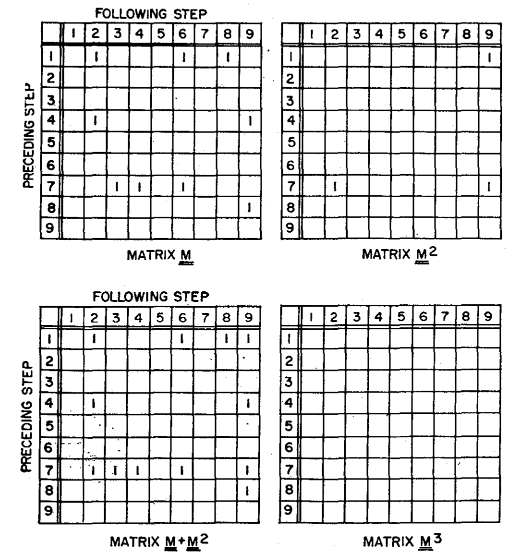


Fig. 18—Successive priority matrices.

Carrying out the indicated procedures for the example cited generates the successive matrices shown in Fig. 18. It is found that $\mathbf{M}^3 = 0$, so that in this case $\mathbf{H} = \mathbf{M} + \mathbf{M}^2$ simply. In addition to the implied condition (e) already noted, two other implied conditions are revealed by $\mathbf{M}^2$, namely: step 7 precedes step 2, and step 7 precedes step 9; but $\mathbf{M}^3$ shows that no further conditions are necessary.

Application of the Matrix Method

The method outlined is well adapted to the preliminary screening of the initial flow diagrams for the network system. For this purpose, use of a digital computer may well be required, if a great deal of priority data is involved. Special provisions for programming Boolean operations of this sort are actually available in the new large-scale computer and data-processor now being designed at NBS for investigation of large-scale network operations.

The possibility of performing these priority screening operations directly and automatically, however, (that is, while the computers are actually working on the problems) by using special conflict-monitoring units should not be overlooked. The rules for mechanizing such a process then become simply:

1) Record matrix **M**.
2) Compute $\mathbf{M} + \mathbf{M}^2 + \cdots + \mathbf{M}^\mu = \mathbf{H}$, the "halt" matrix.
3) Before performing step j, check column j. If any element in *column* $j = 1$, stop and wait. Otherwise, proceed to perform every permitted operation immediately.
4) After performing any permitted operation, erase its *row*.

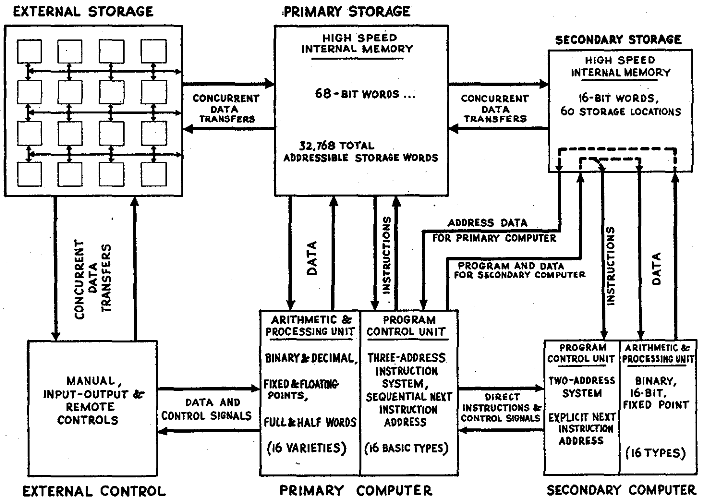


Fig. 19—Over-all block diagram of NBS dual-computer system.

Control System for a Multi-Computer Network

It is of interest to consider now what sort of system of instructions the programmer can use for transmitting requests from primary to secondary, for calling back (or attempting to use) the secondary's results, or for transmitting job-completion signals from secondary to primary. In order to provide a concrete illustration for some of these points, we will describe certain design features incorporated in the new NBS electronic computing and data-processing system now being built at NBS, which we expect to use in running practical tests on the feasibility of multiple-computer networks.

NBS Dual-Computer System

The system now being assembled at NBS actually is broken up logically into several independently operating blocks which run concurrently and from time to time, as the need arises, exchange data and commands (see Fig. 19). The three main processing units in this system are 1) the primary computer, 2) the secondary computer, and 3) the external-control unit. Of these, the first two are independently operable general-purpose computers with separate internally stored programs. Considered together, they serve as a logical model of an actual dual-computer network system. Although the primary computer in this system is a full-scale machine capable of carrying out a wide variety of arithmetical and other operations, the secondary computer is more limited in scope, being intended primarily as an adjunct to the main machine. While the main computer is performing the major arithmetical processing manipulations, the secondary computer concurrently carries out "red tape" procedures pertaining to the primary computer's program, such as 1) counting iterations, 2) systematically modifying the addresses of the operands and instructions referred to by the primary program, 3) monitoring the primary program, and 4) various special tasks. Through the use of special subroutines for the secondary computer, both computers acting in concert can be made to carry out a wide variety of complex operations without unduly complicating the writing of the primary computer programs.

Types of Interchange between the Two Computers

Associated with the secondary computer is the *secondary storage unit,* which contains 60 word-storage locations. Of these 60 secondary storage locations, 15 locations are used as *base registers* by the primary computer and are selected by the primary computer according to certain digit codes in the primary instruction word. These base registers contain the numbers that are used by the primary as automatic address modification constants or as alternative addresses for its next instructions.

Specifically, the primary computer operates with a three-address instruction system. Each instruction word contains the addresses of two operands, α and β, and usually the address of the result of the operation, γ, in the primary memory. The memory location of the next instruction word is specified by an address number contained in one of 15 possible base registers; a code in the instruction word specifies which one of the base registers contains the desired word. Choice instructions, used for program branching, from time to time may cause a new alternative address number to be inserted in the base register designated by the primary instruction as being the source of the address number of the next primary instruction.

The addresses α, β, and γ written in the instruction word are subject to automatic modification in the following way. Each primary instruction contains code groups associated with each of its three addresses in which any base register identification number, 1 through 15, may be written. When this is done, the address number actually referred to by the computer is equal to the sum of the address number written in the instruction word and the address number stored in the indicated base register.

Loading of the base registers and manipulation of the numbers stored in them are carried out by the separate concurrently operating *secondary computer.* The secondary computer follows an internally stored program of 2-address instructions and can carry out 16 types of processing, choice and control operations suitable for manipulation of address numbers, for keeping tallies, for monitoring the primary program, and for carrying out other similar subsidiary tasks. These tasks correspond to the "jobs" delegated by the primary to the secondary that were discussed earlier in this paper. The actual making of a "job request" can be carried out from time to time by having the primary instruction program order the insertion of a new instruction in the secondary instruction register (RS instruction). (See Fig. 20.) Alternatively, the primary program can also order the transfer of data between the primary storage units and the secondary storage unit (TS instruction). By these means, the primary computer is able to exercise unconditional control over the actions of the secondary program. On the other hand, the secondary computer can influence the program of the primary com-

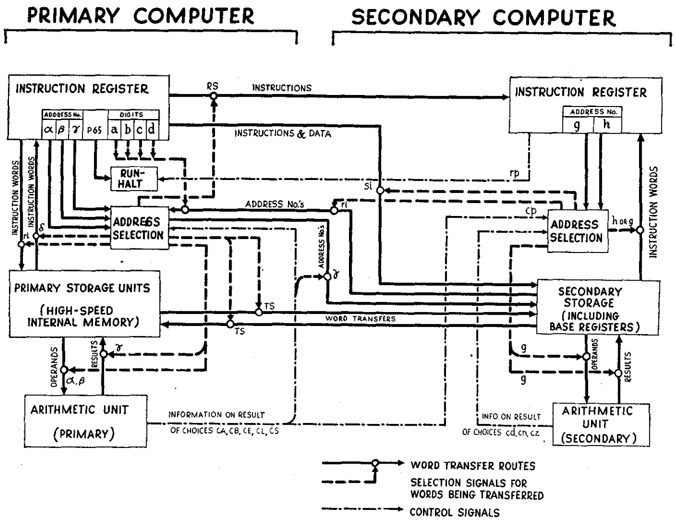


Fig. 20—Functional interrelationships between primary and secondary computer.

puter by manipulating the address numbers of the memory locations to which the primary program refers. The results of these manipulations correspond to the completed job-result data transmitted from secondary to primary in the dynamic models discussed earlier.

Synchronization of Interchanges Between the Two Computers

In order to synchronize the actions of the two computers properly, a programming system is employed in which certain selected instructions in the primary program are marked with a one-digit in a specified bit position. Instructions so marked ("marked primary instructions") are intended in most cases to be carried out *after* certain key instructions to be performed by the secondary computer. They correspond to the "call-back" situations mentioned earlier. When a "marked" instruction is reached in the primary program, control of the relative time sequencing of primary and secondary operations can be passed to the secondary program. The secondary computer program can then regulate the relative priority of execution of subsequent instructions in either program by means of its sequence-regulating *rp* (regulate primary program) instruction (see Fig. 20).

For example, in a typical application of the use of the *rp* instruction, the primary computer, upon reaching a "marked" instruction will check, before executing this instruction, to see whether the secondary computer is stopped and waiting with an *rp* instruction. If this is not the case, the primary program waits until the secondary program reaches an *rp* instruction. As soon as both programs have reached a mutual waiting status, that is, the primary program waiting with an unexecuted "marked" instruction and the secondary program waiting with an unexecuted *rp* instruction, the subsequent sequencing takes place according to the code written in the secondary (*rp*) instruction word (see Table VII). Various alternatives are available, under which priority may be accorded to either computer

TABLE VII*

CODE FOR SEQUENCE-REGULATING (*rp*) INSTRUCTION OF SECONDARY COMPUTER

Qn	Specification Questions	Digit Code for Answers	
		Digit #n = 1	Digit #n = 0
Q1	Should the secondary program now wait for a marked primary instruction?	Yes, wait.	No, proceed immediately to Q2.
Q2	If the answer to Q1 is *yes*, should the primary instruction be executed, before proceeding to Q3?	Yes, execute it and proceed to Q3.	No, postpone execution and proceed to Q3.
	If the answer to Q1 is *no*, should the barrier system now be *activated*?	Yes.	No.
Q3	Should the barrier system now be *deactivated* permanently, i.e, until ordered reactivated by a subsequent *rp* instruction?	Yes.	No.
Q4	Should the secondary program proceed immediately to the next secondary instruction (ending the operation at this point)?	Yes, end this operation here.	No, proceed to Q5.
Q5	Should the next encountered barrier (including the currently waiting marked primary instruction if any) now be removed? (Barrier system will be deactivated temporarily to permit one and only one barrier to be passed).	Yes, bring in the next primary instruction, then proceed to Q6.	No, proceed immediately to Q6.
Q6	Should the secondary program now proceed to its next instruction?	Yes.	No, remain waiting until further notice.

* The questions Q1 through Q6 are asked and answered in order. The answers (yes = 1, no = 0) are recorded as a six-bit code sequence. This code symbol is then written in the sequence-regulating (*rp*) instruction word in the secondary computer program. A *barrier* is a stoppage of primary program operations resulting from the presence of "marked" primary instruction word. These stoppages can take place only when the *barrier system* is *activated*. The barrier system can be activated and *deactivated* from time to time by the secondary computer's sequence-regulating (*rp*) instruction using the codes shown in the table. When the system is in the deactivated state, the marking digit does not cause any stoppages. The barrier system can be deactivated either 1) permanently, until specifically reactivated by a subsequent *rp* instruction or 2) temporarily, until a single marked instruction has been allowed to pass.

depending on whatever order of sequencing is most appropriate to the logic of the interchange. Possible orderings include repetition or skipping of the primary instructions.

Provision is also made at such mutual waiting points for the secondary program to branch to either of two next specified alternative instructions according to whether the primary computer was waiting in a *special-status* condition (*cp* instruction). Special status includes the various types of overflow and underflow (fixed point and floating point), etc.

When both computers are in a mutual waiting condition, the secondary computer can refer to the various base registers and control counters that the primary computer has just referred to or is about to refer to. These references by the secondary computer can be made either as a result of explicit register numbers, etc., written in the secondary instructions, or as a result of the fact that these locations are currently being specified in the currently-waiting primary instruction. In consequence, the secondary computer can be used to monitor or to interpret the program of the primary computer in a highly flexible fashion.

Conclusion

The foregoing discussion outlined some of the paths along which digital systems investigations are proceeding at the National Bureau of Standards. It is evident that the progress made thus far represents only a small start on a series of system problems of wide scope. These problems of network organization will become increasingly important as time goes on, because the large-scale deadline-meeting applications of the future (to the extent that they are limited by the basic component rates) will force designers more and more in the direction of widely extended network systems. In these extended systems, large numbers of independent machines will participate cooperatively in the solution of a common problem. Such systems will inevitably be more loosely organized with respect to centralized control than the relatively compact and logically self-contained machines of today. The dynamics of such loosely organized systems can thus be more effectively conceived of in terms of the group behavior of large numbers of randomly-interacting independent units rather than as a completely prescheduled, unified-machine process. Although the over-all performance of such systems can be best analyzed in probabilistic or statistical terms, the actual elementary building blocks in the system will still have to operate in a rigid, precisely defined way. The problem of system design then becomes one of contriving methods for combining the particular building blocks at our disposal so that fullest advantage is taken of the ways in which the laws of chance act to combine their individual, statistical-performance parameters.

A Program-Controlled Program Interruption System

F. P. BROOKS, JR†

Objectives

IN a computer complex now under development at IBM for the Los Alamos Scientific Laboratory,[1] a major objective is to improve performance by eliminating unnecessary waiting. A fundamental concept is that of multiple data processing units sharing a common memory and operating simultaneously and asynchronously. This complex must be capable of immediate and coordinated response to external signals. These two concepts demand special methods of switching any single unit from one program to another. The system by which a computer unit responds to arbitrarily timed signals with programs pertinent to each signal will be called a program interruption system.

There are two quite distinct purposes for which a program interruption system is necessary. The first of these is to provide a means by which a computer can make very rapid response to extra-program circumstances which occur at arbitrary times, performing useful work while waiting for such circumstances. These circumstances will most often be signals from an input-output exchange that some interrogation has been received or that an input-output operation is complete. For efficiency in real-time operation, the computer must respond to these forthwith. This demands a system by which such signals cause a transfer of control to a suitable special program.

The second purpose is to permit the computer to make rapid and facile selection of alternate instructions when program-activated indicators signal that special circumstances have occurred. For example, it is clearly desirable to have such a system for arithmetic overflow, since the alternatives are tedious and wasteful programmed testing or a costly machine stop when the condition arises. As another example, it is desirable to have a special routine seize control and to take corrective steps whenever the regular program attempts a division by zero.

These two purposes—response to asynchronously occurring external signals, and monitoring of exceptional conditions generated by the program itself—are quite distinct, and it would be conceivable to have systems for handling each independently. However, a single system serves both purposes equally well, and provision of a single uniform system permits more powerful operating techniques.

The program interruption system adopted must obey several constraints. The most important is that programming must be straightforward, efficient, and as simple as the inherent conceptual complexities allow. Secondly, the special circuitry must be reasonably modest, for mainte-

† IBM Corp., Poughkeepsie, N.Y.

[1] S. W. Dunwell, "Design objectives for the IBM stretch computer," *1956 Proc. EJCC;* p. 20.

nance economy as well as low first cost. Thirdly, the computer must not be retarded by the interruption system, except when interruptions do in fact occur. Finally, since there is little experience in the use of multiprogrammed systems, the system should be as flexible as possible. It is important to avoid inflexibility based upon assuming certain methods of use.

In the next section we shall consider the several parts of the system individually, examining the problem solved by each. A final section will show examples of the use of the system.

System Components

The first question to be answered in a program interruption system is: When is the time to interrupt? This requires a signal when there is a reason for interruption. It also requires a designation as to when interruptions are to be permitted. The solution to the first is straightforward. For each condition which may require attention, there is a flip-flop, called an *indicator,* which may be interrogated by the control mechanism. When the condition arises, the flip-flop is set on, and it may be turned off when the condition disappears or when the program has cared for it. In the system under development there are sixty-four such indicators, and they are grouped together into a single register. This indicator register has an address and can be treated as a data word for ordinary program operations.

The designation of times when interruption is permitted can be done in several ways. It is possible to organize a system so that any condition arising at any time can cause interruption. Alternatively one can provide a bit in each instruction which designates whether interruptions shall be permitted at the end of that instruction or not. These methods make no distinction among the interrupting conditions. It is highly desirable to permit selective control of interruptions, so that at any time one class of conditions might be permitted to cause interruptions, and another class might be prevented from causing interruptions. Therefore, each of the sixty-four conditions is provided with a program-set *mask* flip-flop, which allows that condition to cause interruption when it is on. When the mask bit is off, interruption cannot be caused by that condition. As with the indicators, the mask bits are assembled into a single register with an address, so that they can all be loaded and stored as a unit, as well as individually. The indicator register and mask register give the programmer full control as to which conditions are to be permitted to interrupt at any time.

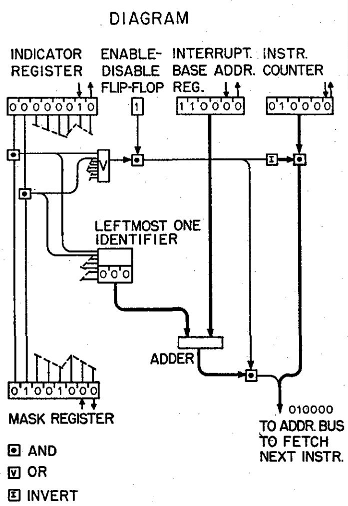
A second major question that a program interruption system must answer is: What is to be done when an interruption occurs? In the simplest systems, the program transfers to some fixed location, where a fix-up routine proceeds to determine which condition caused the interruption and what is to be done. This is rather slow. In order to save time, we provide a branch to a different location for each of the conditions which can cause interruption. The particular location is selected by a leftmost-one identifier. This device generates a number giving the position of the bit within the indicator register which signals the condition causing the interruption. This bit number is used to generate an instruction address for selecting the appropriate operation to be performed. Since it is anticipated that such a computer system will often be operated in a multiprogrammed manner, the bit address is not used directly as the instruction address, for this would require the whole table of fix-up instructions to be changed each time the computer switched to a different program. Instead, the bit address is added to a base address held in an interruption base address register. The sum is used as the next instruction address. One can easily select among several interruption instruction tables by setting the base address.

A third major question facing any interruption system is: How does control return to the main program when the fix-up routine is complete? One solution is to employ several instruction counters. A more economical solution is to have the instruction counter contents automatically stored in a fixed location upon interruption. We chose, however, to make no provision for automatic storage. Instead, the address of the instruction executed immediately after interruption is generated without disturbing the contents of the instruction counter. That instruction can store and alter the undisturbed instruction counter contents, if desired. A typical operation for such use would be Store Instruction Counter and Branch.

If the instruction counter is not altered by the interrupting instruction, the program automatically returns to the interrupted program and proceeds. This permits exceptionally rapid and simple treatment of the conditions that can be handled with a single instruction. More complex conditions are handled by a Store Counter and Branch instruction that enters a suitable subroutine just as any other subroutine would be entered. Program control of counter storage has two advantages: the storage location can be selected at will, and it saves time to perform counter storage and change only when needed.

A fourth question faced by any program interruption system is: How are the contents of the accumulator, index registers, etc., to be preserved upon interruption? Automatic storage of these is both time consuming and inflexible. It is better to use the standard subroutine philosophy—the fix-up routine is responsible for preserving and restoring any of the central registers it uses. Special operations simplify storage and restoration of central registers, but full flexibility is left with the subroutine programmer. He need not store and retrieve anything more than he intends to corrupt.

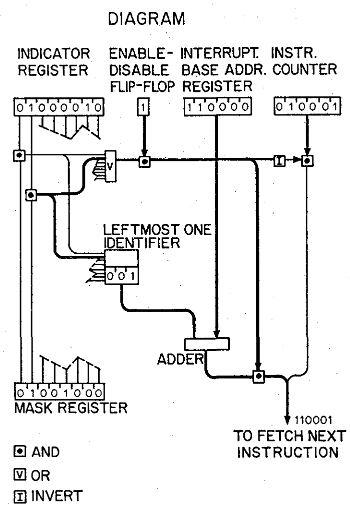
The fifth question that must be answered by a program interruption system is: How are priorities established among interrupting conditions and what provision is made for multiple interruptions? Provision of the full masking facility answers this problem, since any subset of the conditions may be permitted to cause interruption. Each fix-up



PROGRAM SEQUENCE

INSTR. LOCN. BIN.	DEC.	OPER.	ADDR.	ADDR.
001111	(15)	ADD	X	
→ 010000	(16)			

Fig. 1.



PROGRAM SEQUENCE

INSTR. LOC. BIN.	DEC.	OPER.	ADDR.	ADDR.
001111	(15)	ADD	X	
010000	(16)	MULT.	Y	
→ 110001	(49)	CLEAR ADD	L(0)	

Fig. 2.

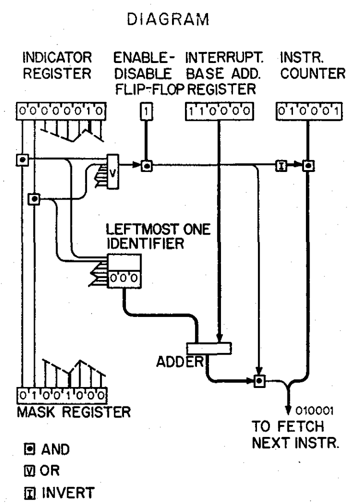
subroutine can use a mask of its own, thereby defining the conditions which are allowed to cause interruption during that routine. There is also provided a means to disable the whole interruption mechanism for those short intervals when an interruption would be awkward. One such interval is that between the time when a subroutine restores the interruption base address appropriate for the main program and the time when it effects the return to the main program. The mechanism is automatically disabled by certain operations and can be optionally enabled or disabled by others.

Simultaneous conditions are taken care of by the leftmost-one identifier, which selects that condition with the lowest bit address in the indicator register for first treatment. This is satisfactory because the fix-up routines for the several conditions are largely independent of one another. The positioning of conditions within the indicator register defines a built-in priority, but this priority can readily be overridden by suitable masking when the programmer desires. In fact, it might be said that the leftmost-one identifier solves the problem of simultaneity while the selectivity provided by the mask solves the problem of over-all and longer-term priorities.

Examples

Fig. 1 shows the system organization of an interruption system with only eight conditions. The indicator register has only one condition, number six, on. The mask register is set up to allow only conditions one and four to cause interruption. Instruction 15 has just been executed, and the instruction counter has been stepped up to 16. There is no interruption so the next instruction is taken from location 16 in the normal manner.

In Fig. 2 the execution of instruction 16 is accompanied by the occurrence of condition one. The leftmost-one identifier generates a binary one which is added to the 48 contained in the interruption base address register. The result, 49, is used for the address of the next instruction rather than the 17 contained in the instruction counter, which is unchanged.



PROGRAM SEQUENCE

INSTR. LOCN. BIN.	DEC.	OPER.	ADDR.	ADDR.
001111	(15)	ADD.	X	
010000	(16)	MULT.	Y	
110001	(49)	CLEAR ADD	L(0)	
→ 010001	(17)	STORE	Z	

Fig. 3.

In Fig. 3 is shown the case when the instruction at location 49 does not change the instruction counter. The interruption mechanism has turned off condition one which caused the interruption. No other condition and mask bits coincide. After the instruction at location 49 is complete, the next instruction is taken from the location specified by the instruction counter, which still contains 17. This one-instruction fix-up routine might be used to clear the accumulator after a floating point underflow.

Fig. 4 shows a different sequence that might have followed Fig. 2. Suppose indicator 1 represented an end-of-file condition on a tape and several instructions are needed to take care of the condition. In this case, the instruction at location 49 disables the interruption mechanism, stores the instruction counter contents (17) in location 24, and causes an unconditional branch to location 39. The fix-up routine proper consists of the three instructions from 39 to 41. It might be any length, and might include testing and scheduling of further input-output operations. During the routine no further interruptions can occur. Instruction 42 is a Branch and Enable instruction, which causes a

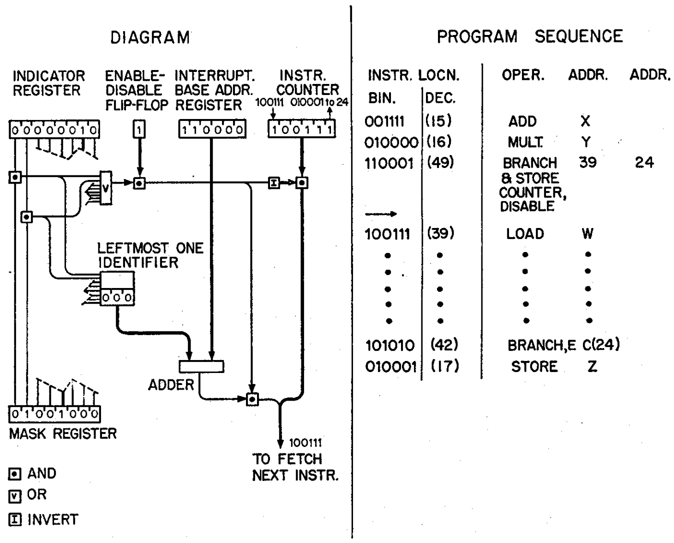


Fig. 4.

branch to the location specified in 24. This returns control to the interrupted program at location 17 and re-enables the mechanism so that further interruptions are possible.

The program in Fig. 4 assumes that it is desired to prevent further interruptions during the fix-up routine. If further interruptions were to be allowed during the routine, and the same mask still applies, one would use a simple Store Counter and Branch instruction at location 49, and a simple branch instruction at location 42. This procedure is appropriate when, and only when, the programmer is certain the condition one cannot arise during the fix-up routine or during any that might interrupt it.

In the most sophisticated use, where it is desired to use a long fix-up routine which is to be interrupted under a different set of conditions, the program shown in Table I is suitable. The mechanism is disabled at the time of the first instruction after interruption. The new mask is loaded and the old preserved. The mechanism is then enabled. At the end of the routine the mechanism is disabled, the old mask restored, and the mechanism is re-enabled as control is transferred to the originally interrupted routine at location 17.

This procedure is clearly suitable for any number of levels of interruptions upon interruptions, each of which may have a different set of causing conditions. Each level of routine is under only the usual subroutine constraint of preserving the contents of the registers it uses. Full program control simplifies programming and multiprogramming, as does the refusal to assign special functions to fixed memory locations. The task of the programmer of fix-up routines is simplified by the provision of special operations and by the adoption of the same conventions and requirements for interruption routines as for ordinary subroutines.

An especially important feature of the program interruption system just described is that it makes almost no demands upon the writer of the lowest level program. He need only set up the interruption base address register and the mask register. He need not even understand what

TABLE I

Inst. Location Binary	Inst. Location Decimal	Operation	Address	Address
00111	(15)	ADD	X	
010000	(16)	MULT	Y	
110001	(49)	BRANCH, STORE COUNTER, DISABLE	39	24
100111	(39)	SWAP	Mask	23
101000	(40)	ENABLE		
101001	(41)	LOAD	W	
.	.	.	.	
.	.	.	.	
.	.	.	.	
101101	(45)	DISABLE		
101110	(46)	SWAP	Mask	23
101111	(47)	BRANCH, ENABLE	c(24)	
010001	(17)	STORE	Z	

he puts there or why, but may follow the local ground rules of his installation. Priorities, preservation of data, and other programming considerations that are inherent in program interruption concern only the author of the fix-up routines. In open-shop installations it is important that any programming burden inherent in such sophisticated operation fall upon the full-time utility programmer rather than upon the general user.

In summary, the program interruption system includes an indicator register in which are assembled flip-flops for the conditions that may cause interruption. These conditions may be program-generated, as by invalid instructions, or external, as by remote interrogations. Corresponding to each condition is a mask bit, and these are assembled into a mask register. When any condition occurs whose mask bit is on, the bit address of that indicator is added to a base address to determine the next instruction to be executed. This instruction is essentially inserted into the normal instruction sequence. If the inserted instruction is not a branch, the program executes it and proceeds exactly as if the extra operation were a part of the main program itself. If the inserted instruction is a branch, it provides for entry to a fix-up subroutine in the same manner any other subroutine would be entered. When the fix-up subroutine is complete, return to the main program is effected just as it would be from any other subroutine. Programming is facilitated by uniformity with ordinary subroutine control, by special operations, and by full programmable selection of interrupting conditions, storage locations, and fix-up routine locations. The governing philosophy of design has been that flexibility and power of program control is of more value to the sophisticated user than would be pseudoconvenience of an inflexible automatic interruption system, but that the power and flexibility made available to the sophisticated user must not be gained at the expense of convenience and simplicity for the casual user.

Acknowledgment

The author wishes to acknowledge the contributions of several colleagues: D. W. Sweeney, who proposed many of the general concepts; Dr. G. A. Blaauw, who offered many suggestions; and Dr. W. Buchholz, under whose guidance this system was designed.

Discussion

B. J. Carr (Applied Physics Lab., JHU): Is this interrupt scheme now incorporated in a computer in existence?

Mr. Brooks: This system is being designed for a very powerful new computer and is not incorporated in a computer in existence. The computer is in part described in "Design Objectives for the IBM Stretch Computer," by S. W. Dunwell, in the 1956 *Proceedings of the EJCC.*

F. M. Verzuh (M.I.T., Cambridge, Mass.): Does IBM plan to make this program interruption device available on 704-709 on RPQ basis? If so, what is the rental price?

Mr. Brooks: I know of no such plans.

H. Siegal (Remington Rand Univac): If one bit in the indicator and mask registers is indicative of an overflow interruption, for example, how would you distinguish between a programmed overflow and a nonprogrammed one, within a sequence of operations?

Mr. Brooks: You can distinguish by masking out interruptions from overflow when intentional overflows occur. For several reasons, intentional overflows will be quite rare in this system. If you did permit interruptions to occur anywhere, you can, in the fix-up routine, decide whether the overflow was intentional by the instruction location at which the overflow occurred.

Mr. Hirmes (Curtiss Wright): Is the particular type of error indicated by each digit of the indicator register built into the machine or can it be varied? Second, wouldn't the second instructions of a two or more instruction fix-up routine be the first instruction of the fix-up routine of the indicator digit following the one under consideration?

Mr. Brooks: First, the particular condition for each indicator is built into the machine. Second, because the system only inserts one instruction into the program, the condition you describe would not occur. Since the counter is not changed from 49, for example, it can't step on to 50. Therefore, if you want a fix-up routine with several instructions, you must write it as a subroutine and enter it as a subroutine. The single instruction inserted into the program is the store instruction counter and branch.

F. B. Banan (General Electric Co., Evansdale, Ohio): Could you please say a few words about the application of interrupt in cases where it is desired to save the contents of the console and memory, read in a completely different program to process data for which immediate answers are desired, and then restore the original programs and proceed from point of interruption.

When will this interrupt system be available? What will be its cost?

Mr. Brooks: First, the interruption system permits operation in the mode you describe. This demands a different approach to console philosophy, but otherwise the procedure is straightforward. One must have the dumping routine in memory, at least enough of it to dump part of memory and load a more powerful routine. The console and memory can be dumped on tape or into slow memory, and the pressing problem handled. At its conclusion, the console and memory contents are restored with fairly simple programming, and one proceeds as usual. The interruption system permits the high-priority program itself to be interrupted by anything of higher priority (errors, at least) as defined by the programmer.

Second, the first machine with such an interruption system is scheduled for delivery in early 1960. The interruption system is integrated into the machine, and it is not possible to separately identify its cost.

A Transistor-Circuit Chassis for High Reliability in Missile-Guidance Systems

G. A. RAYMOND†

Organization

IN recognition of the special needs of the situation, a unique team was organized to develop this computer. Early in the development program, the usual project organization was augmented with engineers trained in quality control, specifications, production engineering and mechanical design, plus mathematicians and statisticians skilled in the use of computers. Organizational lines were bypassed, and a closely-knit working group was established. As development progressed, a special manufacturing department was organized to work directly with the design group so that manufacturing processes would be consistent with the reliability requirements. Much of the success of the program resulted from the close working relationships between the design and manufacturing groups.

† Remington Rand Univac, Div. of Sperry Rand Corp., St. Paul, Minn.

Circuit Reliability

The desire to use mechanized-design techniques[1] as an aid to the logical design, plus a requirement for simplified maintenance, dictated the use of a small, relatively simple building block. The standard circuit (Fig. 1) uses diode logic with a transistor amplifier. The input diodes function as an OR circuit, the surface-barrier transistor as an inverter, or NOT circuit, and the output diodes as AND circuits. Direct coupling is used with -2 vdc representing binary zero and ground potential representing binary one. Up to six inputs may be provided. A *p-n-p* alloy-junction transistor connected as an emitter follower provides the power amplification necessary to drive the eight possible outputs. Over three thousand of these basic circuits are used in the computer, some as the basic inverter circuit,

[1] S. R. Cray and R. N. Kisch, "A progress report on computer application design," *Proc. Western Joint Comp. Conf.;* 1956.

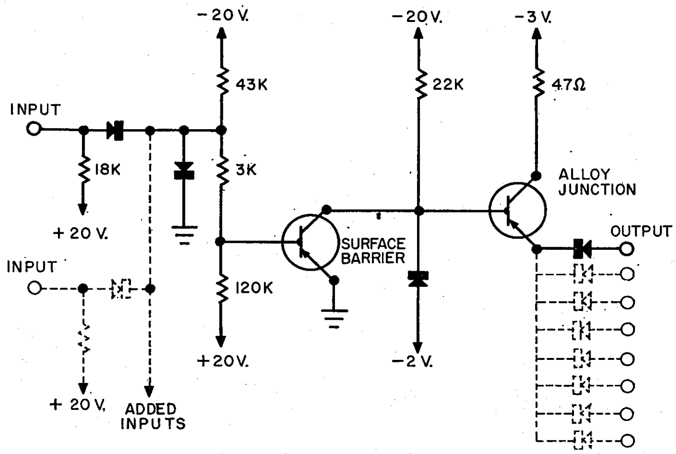


Fig. 1—Standard circuit for computer building block.

and others interconnected to form flip-flops. Altogether, 75 per cent of the computer is composed of this one basic circuit.

Since so many identical basic circuits are used, it became obvious that added circuit reliability could pay large dividends in computer reliability, and thus a major reliability effort was applied. Having established the circuit configuration and made the usual laboratory experiments, a program for the Univac Scientific was prepared using the circuit-design equations to compute optimum-circuit parameters.[2] After establishing criteria for expected parameter variation with life, the values of all parameters were calculated to give the maximum circuit stability. The results are impressive. Even after the beta gain of both transistors has dropped to two thirds of the purchase value, the diode reverse currents have increased from 50 to 400 μa, and all of the resistors have drifted 10 per cent in the worst direction, the circuit will still suffer a 10 per cent voltage variation without failure. Thus, it has been demonstrated that careful, detailed engineering will produce reliable circuits that remain reliable as the components age.

Component Reliability

The selection of components likely to contribute most to the reliability of the computer also required a comprehensive program. The decision to use transistors rather than vacuum tubes or magnetic-core switches resulted from an extensive investigation during which several small self-checking computers were built and operated. Substantial quantities of all types of components were subjected to heat, humidity, shock, vibration, low temperatures, and other destructive environments that might contribute information on comparative component reliability.

Particular care to detect a tendency toward catastrophic types of failure was necessary in this investigation. Components that deteriorate gradually with time would be detected and removed before failure occurred, while catastrophic failures would mean circuit failure every time. The

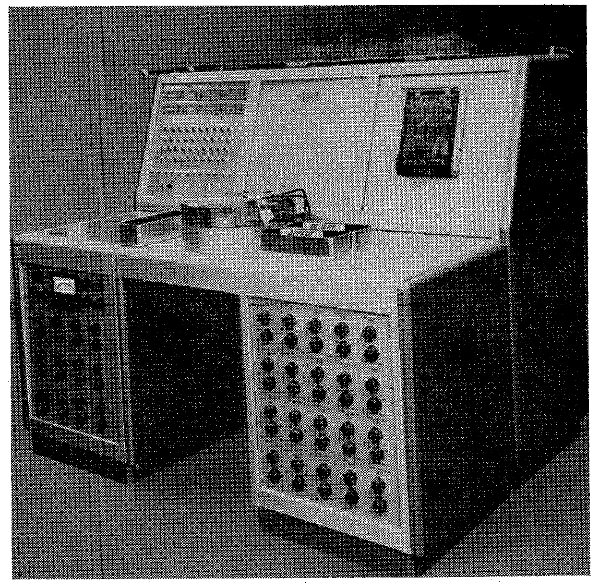


Fig. 2—Automatic transistor tester.

final decisions on component choice had to be based on reliability and not electrical characteristics. Every engineer had to put reliability ahead of all other design requirements, and circuits had to be redesigned to use less efficient components where these proved to be more reliable.

Having established those components that were to be used, it became necessary to establish controls to assure that only these components would be used in manufacture of the equipment. Specifications were written covering every critical component with quality-level requirements exceeding the most rigid military specifications. Large samples drawn from every lot of components had to be subjected to rigid acceptance tests at the manufacturer's plant and again at Remington Rand Univac. To insure compliance with the specifications, Univac quality control representatives are stationed at each manufacturer's plant during the production and testing of the components.

The final assurance that only reliable components are used is the complete test of every component prior to introduction into the computer. In most cases this test is performed on specially designed automatic machines such as the transistor tester shown in Fig. 2. This unit, with a turntable arrangement, moves the transistor through a number of test stations. The test circuitry and parameters to be measured at each station are programmed on the plugboard at the upper right. Counters at the upper left record the rejects on each separate test while the components are being sorted into "accept" and "reject" categories. Extreme precaution had to be taken in the design of the test machines so that transients or equipment failures would not cause damage to the components being tested. This is extremely important in the case of surface-barrier transistors where even small transients may cause complete failure.

Continuous improvement of final screening tests is an important area where much can be done toward eliminating

[2] J. Alman, P. Phipps and D. Wilson, "Design of a basic computer building block," *Proc. Western Joint Comp. Conf.;* 1957.

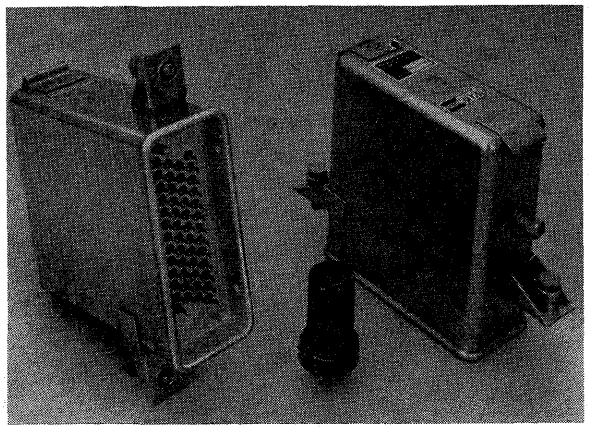


Fig. 3—Standard circuit packaging with vacuum tube for size comparison.

the "weak sisters" from component lots. Improvement of these tests is continuing as results from a large-scale component-testing program are received.[3] This program is being conducted by Inland Testing Laboratories in Chicago, Ill., and Battelle Memorial Institute in Columbus, Ohio, and will give life characteristics and data for screening tests on transistors, diodes, and resistors used in the computer. Approximately 60,000 components are on test in this program.

Reliable Packaging

Many of the components tested during the development program showed tendencies toward deterioration under certain environmental conditions. High humidity proved to be the worst offender, and complete protection of the components from humid atmosphere showed prospects of improving reliability. The results of this phase of the design effort are perhaps the most unique of all the work done to achieve greater reliability.

A specially designed connector, mounting two etched-circuit boards, is the standard package for all electronic circuitry in the computer, and all circuits are mounted in this fashion. The package design, shown in Fig. 3, provides positive hermetic sealing. Connector pins are brought out through glass to metal seals in the base, and a pressure valve is provided for pressurizing the chassis with dry gas. The seal around the base is made by induction soldering and may be unsoldered for repair of the circuitry. The manufacturing process includes a complete bake out of the chassis under high vacuum (see Fig. 4) to reduce the relative humidity below 1 per cent after sealing. The relative humidity may be checked by means of a humidity-sensing element inside each chassis.

The highly reliable contact arrangement is shown in Fig. 5. The arrangement is reminiscent of the knife switch used in power circuits with a flattened male pin and tuning fork shaped female contacts. Two completely independent pairs of contacts in the female connector give redundancy for added reliability. Of the 100,000 contacts in connectors used to date, not one single case of poor contact has occurred.

[3] D. R. Bair and P. Gottfried, "Reliability results from large-scale testing," *Elec. Equip.*; January, 1957.

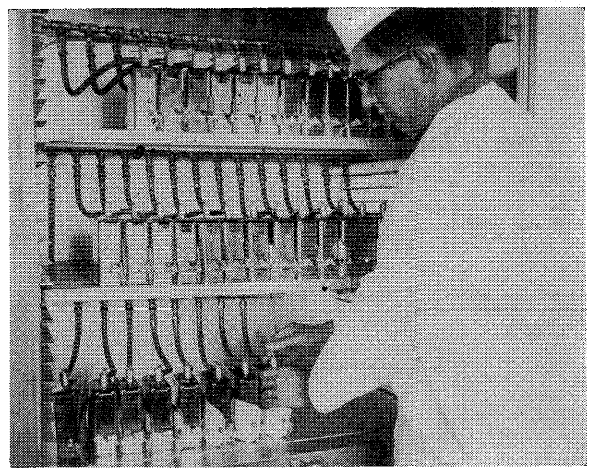


Fig. 4—Chassis are baked under high vacuum prior to sealing.

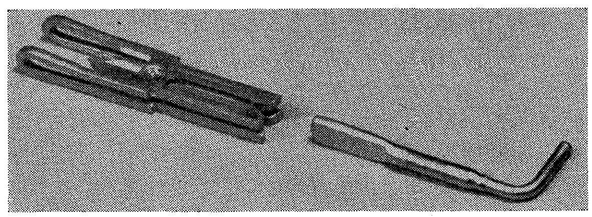


Fig. 5—Male- and female-contact assemblies used in special reliable connectors.

The etched-wiring boards mounted on the connector are fiberglass-epoxy laminate with rolled copper foil on one side. Extreme care is observed in selecting and processing this material so that the finished boards are completely free of scratches, pinholes, or other defects such as warpage or contamination. Assembly of the entire chassis is a "clean-room" operation with temperature and humidity control, white smocks and gloves for all operators, and strict process control. Policy prohibits touching any component with the bare hands. Any component that is dropped, even in a container, is rejected. Rework is carefully controlled and strictly limited. Complete records are kept of each operation on each unit including the time, date, and operator number so that assembly reference can be made during the routine failure analysis which follows every failure.

Dip soldering is used to attach the component leads to the etched wiring and also to connect the etched wiring to the connector pins. These two separate operations are performed on a selective soldering machine which permits masking the entire circuit and exposing only the areas where solder is desired. This procedure reduces the heat transfer to components and permits attaching the components to the board in one operation and the board to the connector in a following operation. Results of selective soldering are shown in Fig. 6. Following the dip soldering and prior to an electrical test, the completed board is given a temperature shock from room temperature to +155°F and then to −50°F. The temperature-shock treatment is further insurance against marginal components or connections that might later show intermittent failure.

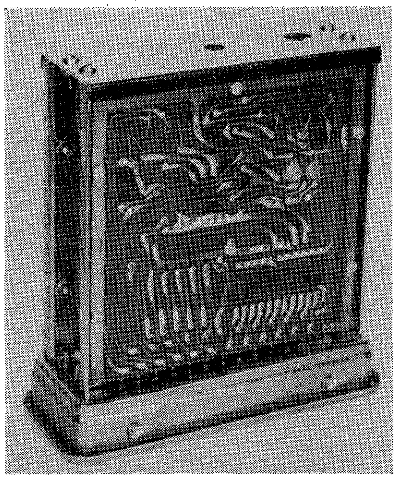


Fig. 6—Standard package without cover—showing results of selective soldering.

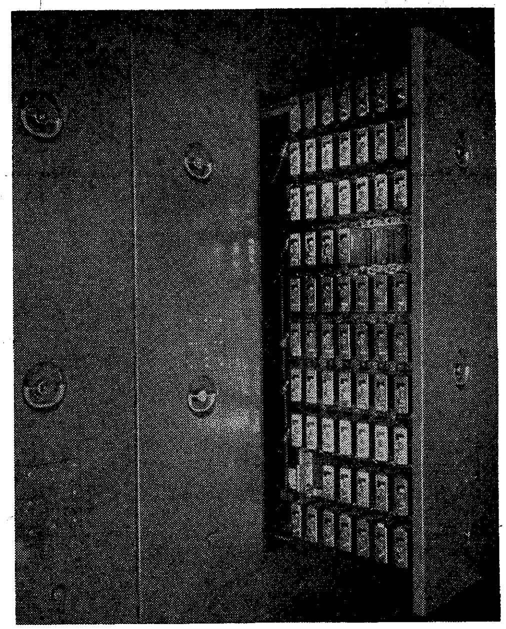


Fig. 7—Completed packages installed in computer rack.

Final assembly of the sealed chassis includes pressurization with a mixture of nitrogen and helium thus allowing a standard helium-leak detector to be used for checking the final seal.

With the final seal complete and final electrical checkout satisfactorily performed, the chassis is installed in the computer panel as shown in Fig. 7. Insertion is performed with a special tool to prevent damage to the connector pins. The chassis is fastened in place with hold-down screws and is ready for operation.

Conclusion

The purpose of this paper is to present a broad picture of a reliability-design effort that achieved true reliability. The salient points may all be summarized in the one word—*attitude*. The desire for reliability must be present in the mind of each person from the chief engineer to the girl on the assembly line. The finest quality control cannot make a poor design reliable and the finest design will not be reliable if the person who builds it is careless. Careful attention to minor details in the selection of components, the design of the circuitry, the packaging of components, and the manufacturing process can pay off in a big way where reliability is the most important requirement.

Discussion

Since the discussion at the Conference dealt mostly with types and causes of failures, an updated summary from December, 1957, through February, 1958, follows.

From completion of the final checkout May 17, 1957, until March 1, 1958, the computer has operated 1613 hours. Failures considered in determining computer reliability were limited to those which, had they occurred during guidance missions, would have caused the mission to fail. They have been seven failures in this category. They are summarized below.

1) Intermittent chassis—the defect has not yet been located.
2) Intermittent chassis—defective solder connection inside pulse transformer.
3) Defective chassis—two shorted diodes, apparently damaged by externally applied voltage.
4) Defective chassis—collector-emitter short in transistor.
5) Defective chassis—collector-emitter short in transistor.
6) Possible intermittent chassis—has not been established as a definite failure, but is suspected.
7) Two rectifier stacks in the power supply—resulted from improper design in the switching circuitry.

Transients from the power supply are suspected as the cause in 4) and 5). No transistor failures have occurred since October when this defect was eliminated.

Two failures previously reported have been removed from this list after detailed study indicated that in one case there was no defect, and in the other case, that the defect would not have caused a computer failure.

There have been twelve chassis removed for reasons which would not have caused computer failure. Of these, four chassis were removed because of low gas pressure, six because of high humidity indication, and two because of defective indicator transistors. No other chassis have been removed for any reason.

A Method of Coupling a Small Computer to Input-Output Devices without Extensive Buffers

JAMES H. RANDALL†

General Description of Computer

THE computer referred to in this paper is contained in a package about desk size. It is intended to sell at a relatively low cost as compared to other general purpose type computers and accordingly would be considered in the "small" class. The basic components used are transistors and diodes. Although the design was aimed specifically at business applications, the computer is nevertheless a general purpose machine with internally stored program. Typical of the applications intended are the problems of small businesses such as payroll, stock inventory, production control, interest calculations, etc. The computer can be considered as an extension of the accounting machine system rather than an integrated data processor.

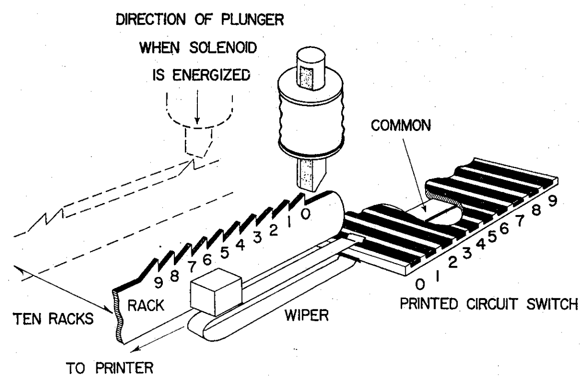


Fig. 1—Illustration of a rack with accompanying solenoid and position detector switch.

Mechanical Accounting Machine

In business applications, one must use all shapes and sizes of business forms. Therefore, a standard mechanical accounting machine with a large carriage suited to these forms was utilized for printing data from the computer and as a source of keyboard entry into the computer. The use of the accounting machine provides the additional advantage that format control of printing on the forms is taken care of on the accounting machine itself and does not have to be stored in internal memory.

This accounting machine is essentially a parallel digit device. The maximum word size has been chosen as ten digits (numeric only). As is conventional in accounting machines the keyboard has ten columns of nine keys each, one column for each digit position in the word. Depressing a key in any particular column determines the value, zero through nine, of the digit printed in that digit position. (Depressing no key causes a zero to be printed.) After all the desired keys have been depressed, the machine cycle is initiated and all the digits of the word are printed simultaneously.

Associated with each column of the keyboard is the conventional rack as shown in Fig. 1. During the machine cycle all ten of the racks are simultaneously driven in a setting direction parallel to their long axes, successively passing through positions representing digital values, zero through nine. Each rack may be stopped at any one of the ten positions, depending upon which key is depressed in that corresponding column of the keyboard. The racks are connected to the printing mechanism so that the value of each digit printed is determined exactly by the position of its corresponding rack at the time of printing. After all of the racks have assumed their proper positions they are held stationary for a period of time sufficient to complete the printing operation and then restored to their original positions.

For keyboard entry of data, the computer must detect the positions of all racks after they have been stopped by the keys, and transfer this parallel information into the memory. In order to print words from the memory, the computer must stop each rack at the proper digital position. This is actually done by a process that pulls in a solenoid for each rack, stopping it at the proper position as it is moving in the setting direction.

Magnetic Ledger Card

In a large percentage of businesses, the data continually being used in calculations are stored in printed form on ledger cards. In general, the problem is to take data from a ledger card, combine it with new data to get the desired result, and bring the information on the ledger card up to date. If the ledger card system is to be retained, it is advantageous if the data on the card is also in machine-readable language. To accomplish this, a strip of magnetic material was added along one edge of the ledger card. On this strip are recorded magnetically certain controls and all of the current information printed on the card. The computer must both record data and read data from this magnetic strip.

To make the magnetic ledger card really practical for the applications intended, it is necessary to store 50 to 100 digits of information on the magnetic strip. The choice of scanning speed and recording density resulted in a read and record rate of about 400 pulses per second.

† The National Cash Register Co., Dayton, Ohio.

There is a data track and a clock track on the strip and the data are recorded serial digit and serial code (four bits per digit). To get the maximum amount of data on the card, a variable word length system is used which requires that an end-of-word symbol follow each word.

Recording on the card is under control of the computer and must be done at the rate of 400 pulses per second, which is also the reading rate.

Reading Punched Cards

In certain applications it is necessary to do some types of distribution for writing reports, etc. The computer was accordingly designed to read punched cards which are sorted on available commercial equipment.

A modified IBM 026 card punch was used for reading the cards. This punch has a duplicate feature which reads one card and simultaneously punches the information read into the following card. By temporarily disabling the punches and operating in the duplicate mode, the cards can be read one after the other at the normal reading station. This data appears as parallel-code serial-digit at a rate of about eighteen digits per second. The computer must take the data at this rate and store it in internal memory.

Internal Construction

Both a core and drum memory were considered for this computer. Since the capacity of the memory is relatively small (100 words), it seemed more economical to use a drum. However, with a drum, communication with each of the aforementioned input-output devices would require extensive buffering and add substantially to the cost of the computer. Therefore, a core memory was considered. The core memory has three major advantages. First, any address can be selected in a very short time. Second, the read-write operation can be stopped and started almost instantaneously and at any point in a word. Third, it is possible to read out a word starting from either the high or low-order end. By exploiting these characteristics it was possible to synchronize, rather than buffer, the memory to the communications devices.

The basic clock frequency of the computer is 25 kc, making a bit time and the read-write cycle for a core 40 μsec. The word length is ten digits of four bits each, or 40 bits. Reading out of the memory is always done in a completely serial fashion. There are two basic modes for this process, "word cycles" and "digit cycles." Initiating a word cycle causes an entire word of the memory to be read (or written into) without interruption. With two extra bit times for control, this cycle takes a total of 1680 μsec. A digit cycle reads only one decimal digit, or four bits, from the memory. The length of this cycle is 200 μsec including an extra bit time for control. These digit cycles may follow one after the other in sequence to read out an entire word. However, any amount of time may elapse between the completion of one digit cycle and the beginning of the next.

In the computer are two single-digit registers of four bits storage capacity each. They are used for certain arithmetic and control operations and also are used as buffers, as will be shown.

Reading Data from Accounting Machine

Entering data into the computer from the accounting machine is comparatively simple. As previously discussed, the value of each digit of the word entered into the keyboard is represented by the differentially set position of a corresponding rack. As shown in Fig. 1, a switch was added to the machine which has ten parallel conductors on one surface, extending in a direction perpendicular to the racks and spaced the same distance apart as the digital positions of the racks. A wiper on each rack then makes contact with one of these ten conductors, dependent upon the position at which the rack is stopped. The wiper also makes continuous contact with a single common conductor for each rack, extending in the direction of movement of the rack. Consequently, if voltage is applied to any one of these common conductors, the same voltage appears, via the wiper, on the conductor corresponding to the position of the rack.

When the computer receives a signal that the racks are in position, it initiates a word cycle that scans the switch and copies the information into memory at the same time. That is, the digit selector selects the digit position in memory in which to write and at the same time selects the corresponding rack to be examined for its digit position. Effectively, the computer is taking the parallel digit one-of-ten coded data from the switch, properly encoding and serializing it and copying it into memory.

Since the computer is operating at the 25-kc rate during scanning of the switch and the racks remain stationary for a sufficient length of time to allow the scanning to be completed, no buffering is necessary.

Printing with the Accounting Machine

Printing a word from internal memory presents a more complex problem. Each rack must be stopped at the proper digital position by a solenoid, shown in Fig. 1.

A means was provided for generating a pulse each time the racks are moved from one digital position to the next. These pulses are fed into a counter which operates to produce in coded form a number representing each of the digital positions of the racks as they move from positions zero through nine.

When a signal is received to start the print operation and before the racks move, a word cycle occurs which successively loads the digits of the word to be printed into one of the digit registers. Each digit is compared to the number in the counter, and if any digit in the word is a zero, a solenoid is energized which prevents movement of the rack corresponding to that particular digit. For example, if the five high-order digits of the word were each zero, the five corresponding racks that cause the printing of those digits would be prevented from moving.

The remaining racks then begin to move in the setting direction. When they reach the "one" position, the counter would then be storing the number representing that position. At this time the same word cycle is repeated, but now in all positions of the word where the stored digits are equal to "one," solenoids will be energized to stop all the corresponding racks at the "one" position. This entire sequence is repeated each time the racks reach a new digital position.

The racks move at a constant velocity with 8 msec time elapsing between digit positions. However, the word cycle at each position requires only 1.68 msec, which, with the aid of high-speed solenoids, is adequate time to catch the racks. Thus, a timing system and a series of word cycles operating at the internal 25-kc rate eliminate the need of a word-length buffer.

Recording on Ledger Cards

The synchronizing technique is applied in the following manner to recording on the ledger card. A card clock is generated internally at the 400 pulse-per-second rate. This clock records the clock track on the magnetic strip and also synchronizes the computer. The card is scanned in opposite directions for read and record; accordingly, the data are recorded on a card from high-order to low-order digit so that it may be read from low-order to high-order digit.

Upon receiving a signal to record, the computer initiates a digit cycle which loads the high-order digit of the first word to be recorded into one of the single-digit registers. In the following bit time the content of the register is examined to determine if the digit is a significant one—that is, a number other than zero. If not, another digit cycle follows immediately, loading the next lower order digit into the same register and the check of the content is repeated. This process continues until the first significant digit is detected. At this time the word scanning operation halts and the next card clock pulse that occurs starts the recording of an end-of-word symbol. When this is completed, the bits making up the digit stored in the register are transferred to a single bit storage one at a time in sequence, each time a card clock pulse occurs. The output of this single bit storage as the bits are stored one after another determines precisely what is recorded on the data track of the magnetic strip. When the last bit of the digit has been transferred to the single bit storage, another digit cycle occurs which loads the next lower order digit into the digit register. The following card clock pulse then begins the serial recording of that digit. This sequence of events continues until all of the digits have been recorded. If another word is to be recorded, the new address is selected in one bit time and the scanning process begins again.

If the word is found to be all zeros only an end-of-word symbol is recorded. In this case the ten-digit cycles required to examine the word can easily occur while the next end-of-word symbol is being recorded. It is not necessary, therefore, to have any unused space between words on the card.

By synchronizing the memory to a slow card clock only five bits of buffer storage are needed for recording any number of digits.

Reading Ledger Cards

Reading ledger cards is essentially the reverse of recording, but with the 400 pulse-per-second clock output from the card being in control. Upon receiving a signal to start reading a card, the computer first initiates a word cycle that clears out (writes all zeros into) the address in which the first word is to be stored. As the data come from the cards serially, each bit is loaded into the digit register having the four bit capacity. A check is made to see if this digit is an end-of-word symbol. If it is not, a digit cycle is initiated and the contents of the digit register are loaded into the low-order digit position of the word. In similar fashion the succeeding digits are first loaded into the register and then into the next higher order digit position in memory. When an end-of-word symbol is detected, no digit cycle occurs and the digit selector of the memory is reset to the low order digit position. The next address in the memory to be loaded is then cleared. This takes place before the next bit of data is received from the card. The process then is repeated for each word to be entered from the card.

By synchronizing the memory to clock pulses received from the card, only one digit of buffer storage is needed in the reading of magnetic ledger cards.

Reading Punched Cards

The punched cards are read in a serial digit fashion from high to low-order digit. A clock pulse is received from the reader which signals the computer that the card is in a position for reading a digit. At this time a word cycle clears out the address to be loaded, and the output of the card reader is loaded by a digit cycle into the low order digit position in memory. When the next clock pulse is received, the previous digit is shifted into the next higher order position of the word in memory and the new digit is loaded into the low order position. All of this can easily take place between card reader clock pulses which occur about every 55 msec. When the last digit of a word is loaded, the word is stored in the proper position. Again, synchronizing the memory to the card reader clock eliminates the need for any buffering at all.

Conclusion

By synchronizing the core memory in a small computer to input-output devices, the buffering required is greatly reduced. Additional savings are realized because small existing registers which are already a necessary part of the computer can be used as the buffers.

Discussion

M. W. Marcovitz (Burroughs Corp., Paoli, Pa.): Is the program for this machine stored in the core memory? If not, where?

Mr. Randall: Yes, the program is stored in the core memory.

T. A. Dowds (Burroughs Corp., Paoli, Pa.): Is this machine available? How are nonsignificant zeros stored when reading from punched cards and from ledger cards? Is this a single-address program?

Mr. Randall: No, this computer is not yet commercially available. The method of storing nonsignificant zeros is to first clear (write all zeros into) the memory cell to be loaded. The data are then loaded into the cell starting at the low-order end of the word until a signal is received that there are no more significant digits for that word. When reading punched cards this signal is an "end-of-field" signal from the card punch. When reading magnetic ledger cards the signal is an "end-of-word" symbol read from the card. The instruction format is not single address, but rather "three-plus-one" address.

R. A. Wallace (Burroughs Corp.): What is the method of feeding the ledgers? Is part of the information used to store line information? What provisions are there for checking?

Mr. Randall: The ledger cards are driven into the carriage by a mechanism added to the mechanical accounting machine. Line information is stored on the card so that it is stopped on the proper line for posting. There is practically no internal checking. However, checking can be accomplished by normal programming methods.

The Synthesis of Computer-Limited Sampled-Data Simulation and Filtering Systems*

ARTHUR S. ROBINSON†

THIS PAPER concerns the synthesis of systems in which a single digital computer is to be used in conjunction with an array of output "holds" or filters either to simulate the dynamic transfer characteristics of a number of linear continuous systems or to filter random messages from a number of continuous inputs, each of which consists of a mixture of random message plus random noise.

A block diagram of such a system is shown in Fig. 1. The system inputs are the continuous time functions

$$r_1(t),\ r_2(t),\ \cdots\ r_{N-1}(t),\ r_N(t).$$

The system outputs are the continuous time functions

$$c_1(t),\ c_2(t),\ \cdots\ c_{N-1}(t),\ c_N(t).$$

Each system output is obtained from an individual continuous output filter. This filter is in turn actuated by sampled signals periodically derived by the computer as it moves sequentially from channel to channel.

In systems of this type, in which a digital computer has available and is to supply continuous data, *the computer operates on sampled data only because a series of sequential operations are to be performed, each requiring a finite amount of time.* A typical simulator channel requires time for switching between channels and for analog-to-digital and digital-to-analog conversions, defined as s_j, and

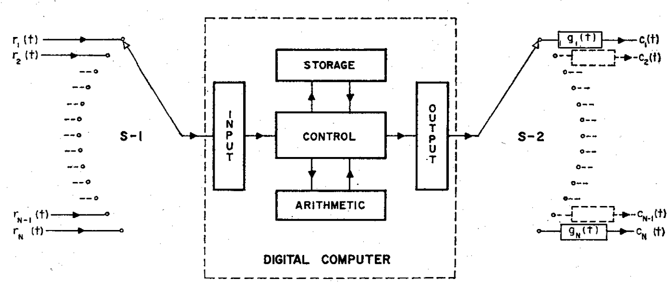


Fig. 1—Simulation system functional block diagram.

time for internal computer data transfers, multiplications, additions and subtractions, defined as k_j. The time required for switching and conversion is generally constant, whereas the time required for internal data processing and computation is governed by the complexity of the computer program. That is, in general, k_j will be a function of the number of terms in the computer program for that channel, so that T, the system sampling period, will be a function of the total number of program terms the computer is required to process for all channels.

The term Computer Limited has been coined to designate sampled data systems of this type, in which the system sampling period T is a function of limitations imposed by the computer implementation. Computer Limited sampled data systems can be contrasted to Data Limited systems, in which the system sampling rate is limited by some fixed external constraint in the data measuring equipment, such as the speed of rotation of a radar antenna. It is important to understand this basic difference between sys-

*Details pertinent to both this paper and footnote 1 are contained in footnote 2.

† Eclipse-Pioneer, Div. Bendix Aviation Corp., Teterboro, N.J.

tems, since synthesis techniques that have been derived heretofore for the optimization of Data Limited sampled data systems are not applicable to the Computer Limited problem.

It is interesting to note that Computer Limited sampled data systems are not restricted to the simulation systems described in this paper. *In general, any system that utilizes a digital computer in real time is a Computer Limited sampled data system, unless the available input data rate is slower than the speed with which computer solutions can be obtained.* Many analog computing systems that utilize multiplexed elements are also subject to Computer Limited constraints.

An application of Computer Limited theory to control system synthesis has already been presented.[1] It is suggested that the basic techniques to be presented in this and in supplementary papers[2] can be used as tools in the general synthesis of Computer Limited sampled data systems.

Returning to the stimulation and filtering application. When the system illustrated in Fig. 1 is to simulate the dynamic transfer characteristics of a number of linear continuous systems actuated by deterministic input signals, the ideal system outputs are as defined by the block diagram of Fig. 2, where

$$h_1(t),\ h_2(t),\ \cdots\ h_{N-1}(t),\ h_N(t)$$

are the impulsive responses corresponding to the ideal input-output relationships. When the system inputs are random functions of time of known statistical characteristics, or when the function of the system is to filter random messages from the continuous inputs, each of which consists of a mixture of random message plus random noise, the ideal system outputs are as defined in Fig. 3. As indicated, the ideal filtering response for each channel would result if a signal equal to the noise input could be effectively subtracted from the total input, so that only the random messages

$$r_{m_1}(t),\ r_{m_2}(t),\ \cdots\ r_{m_{N-1}}(t),\ r_{m_N}(t)$$

actuated the ideal filters. Since the block diagrams of the filtering and simulation problems are almost identical, it is convenient to visualize the filtering problem as the simulation of ideal filters, and so to combine the discussion of system synthesis techniques under the single heading of "simulation."

The ideal system responses shown in Figs. 2 and 3 define reference outputs against which the performance of the actual system shown in Fig. 1 must be compared to evaluate the effectiveness of the simulation. When system inputs are deterministic, the synthesis techniques to be described permit the system designer to determine the linear digital computer program that will minimize the integrated error squared between ideal and actual continuous system out-

[1] A. S. Robinson, "The Synthesis of Computer Limited Sampled Data Control Systems," presented at AIEE Computer Conf., Atlantic City, N.J., October, 1957.

[2] A. S. Robinson, "The Optimum Synthesis of Computer Limited Sampled Data Systems," D. Eng. Sc. Dissertation, Columbia University, New York, N.Y.; May, 1957.

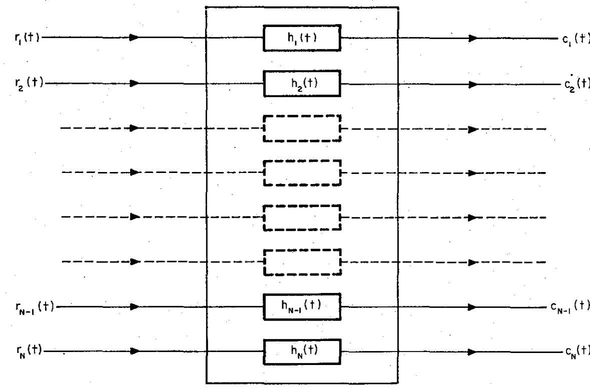


Fig. 2—Ideal simulator input-output relationships —deterministic inputs.

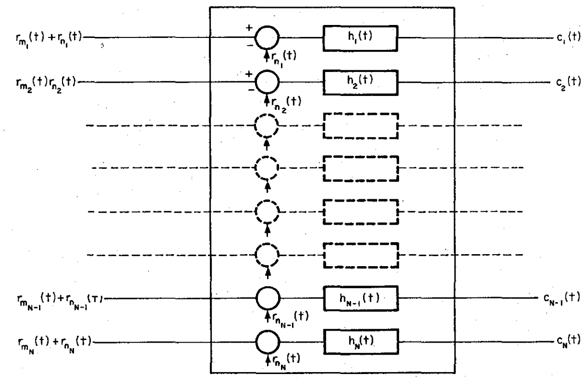


Fig. 3—Ideal simulator input-output relationships—random inputs.

puts. When system inputs are random, the synthesis techniques lead to minimization of the mean squared error. When only the form of the channel output filters are specified, the synthesis procedure permits the filter parameters to also be optimized.

Returning to Fig. 1, a mathematical model for a given channel can be obtained by tracing the channel from its input to its output. At the input the continuous signal $r_j(t)$ is periodically connected to the digital computer input, where the sampled analog voltage is converted to a digital number. Each such digital number is effectively an instantaneous sample of the value of the continuous function. If the period between sampling instants is denoted by T, the conversion of the continuous deterministic input signal $r_j(t)$ to the series of digital numbers

$$r_j(0),\ r_j(\mathrm{T}),\ r_j(2\mathrm{T})\ \ldots$$

can be visualized as a process of impulse modulation, with the area of each impulse equal to the value of the corresponding digital number.

Each time the computer receives new input data it proceeds through a new series of computations and delivers a new solution at its output after the time delay required to perform these computations. The computed problem solution is then used to actuate the output filter, characterized by its impulsive response, $g_j(t)$, so that as the computer moves on to other simulator channels, the output filter provides a continuous output response for the jth channel.

The mathematical model for a single channel with a deterministic input is shown in Fig. 4 in terms of Laplace and Z transforms. It is convenient to describe the digital computer by an instantaneous response, characterized by $C^*(Z)$, followed by a time delay, described by $Z^{-k/T}$. The transfer function of the continuous output filter is defined by $G(s)$. The meaning of each of these symbols is shown more precisely in Table I. $r^*(t)$ describes the train of impulses at the computer input. Z^{-1} is, of course, simply the delay operator e^{-sT}. The Z transform of the computer input is $R^*(Z)$. $G(s)$ is the Laplace transform of the output filter impulsive response.

Dr. Salzer[3] has shown that $C^*(Z)$, the linear program of a digital computer operating in real time, is physically realizable only when it can be described by a ratio of polynomials in Z^{-1}, as indicated in Table I. Referring to Table II, this constraint states that the impulse sequence corresponding to the effective computer instantaneous output, characterized by $P_i^*(Z)$, can only be formed as the sum of appropriately weighted past and present values of the computer input, and past values of the computer output. That is, $R^*(Z)a_0$ corresponds in the time domain to the input impulse sequence multiplied by a_0. $R^*(Z)Z^{-1}a_1$ corresponds to the input impulse sequence delayed by T and multiplied by a_1. $R^*(Z)Z^{-2}a_2$ corresponds to a delay of $2T$ and multiplication by a_2, etc. $P_i^*(Z)Z^{-1}b_1$ corresponds to the computed output sequence delayed by T and multiplied by b_1. $P_i^*(Z)Z^{-2}b_2$ corresponds to a delay of $2T$ and multiplication by b_2, etc. Only past and present terms can be utilized in a real time digital computer because future terms (positive powers of Z) will not, of course, be available. Table II also indicates the existence of the constraint that exists in an actual system implementation between the number of computer program terms, the channel computing time k_j, and the system sampling period T. This relationship between computer program complexity and required computing time must always form part of the basic problem statement. The relationship need not be linear. Any nondecreasing function can be accepted, defined in analytical, graphical, or tabular form.

Fig. 5 summarizes the relationships between true and desired system outputs for a single simulation channel with a deterministic input. The object of the synthesis procedure is to determine the linear digital computer program $C^*(Z)$ that will result in minimization of the integral of the continuous error squared. That is, the quantity

$$\int_0^\infty [\epsilon(t)]^2 dt = \int_0^\infty [c_t(t) - c_d(t)]^2 dt$$

is to be minimized. Note particularly that it is the continuous error, squared and integrated, that is being used to evaluate system performance, not the error at sampling instants only. Thus inter-sample ripple is automatically accounted for. When only the form of $G(s)$ is specified, it is also required that the optimum parameters for $G(s)$ be determined.

[3] J. M. Salzer, "Treatment of Digital Control Systems and Numerical Processes in the Frequency Domain," D. Eng. Sc. Dissertation, Dept. of Elec. Eng., M.I.T., Cambridge, Mass.; 1951.

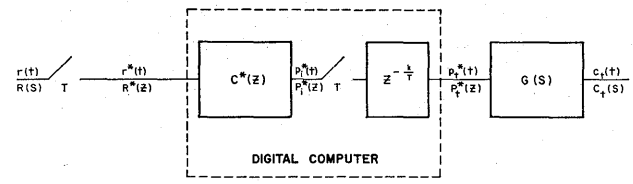


Fig. 4—Simulation system single channel—block diagram.

TABLE I

BLOCK DIAGRAM DEFINITIONS

$$r^*(t) = \sum_{n=0}^{\infty} r(t)\delta(t - nT) \quad (1.1)$$

$$Z^{-1} = e^{-ST} \quad (1.2)$$

$$R^*(Z) = \sum_{l=0}^{\infty} r(lT)Z^{-} \quad (1.3)$$

$$C^*(Z) = \frac{\sum_{p=0}^{p_m} a_p z^{-p}}{1 + \sum_{q=1}^{q_m} b_q z^{-q}} \triangleq \frac{a^*(z)}{b^*(z)} = \frac{P_i^*(Z)}{R^*(Z)} \quad (1.4)$$

$$G(S) = \int_0^\infty g(t)e^{-st}dt \quad (1.5)$$

TABLE II

COMPUTER PROGRAM RELATIONSHIPS

$$C^*(Z) = \frac{\sum_{p=0}^{p_m} a_p z^{-p}}{1 + \sum_{q=1}^{q_m} b_q z^{-q}} = \frac{P_i^*(Z)}{R^*(Z)} \quad (2.1)$$

$$P_i^*(Z) = R^*(Z)[a_0 + a_1 z^{-1} + a_2 z^{-2} + \cdots a_{p_m} z^{-p_m}] - P_i^*(Z)[b_1 z^{-1} + b_2 z^{-2} + \cdots b_{q_m} z^{-q_m}] \quad (2.2)$$

$$Z^{-\frac{k}{i}}, \quad k = f(p_m + q_m) \quad (2.3)$$

$$T_j = k_j + s_j \quad (2.4)$$

$$T = \sum_{j=0}^{N} T_j \quad (2.5)$$

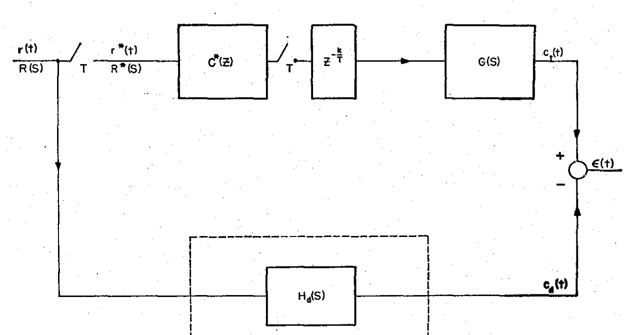


Fig. 5—Simulation channel error—deterministic input.

TABLE III

OPTIMIZATION RELATIONSHIPS FOR COMPUTER LIMITED SIMULATION SYSTEMS—DETERMINISTIC INPUTS

(a)

$$\int^{a_p} E^{2*}(Z) = 2\gamma^2 R^*(Z) R^*\left(\frac{1}{Z}\right) \Phi_{g_k g_k}{}^*(Z) \frac{a^*(z) z^{+p}}{b^*(z) b^*\left(\frac{1}{z}\right)} - 2\gamma \frac{R^*\left(\frac{1}{Z}\right) \mathbb{F}^*(Z) Z^{+p}}{b^*\left(\frac{1}{z}\right)} \tag{3.1}$$

$$\int^{b_q} E^{2*}(Z) = -2\gamma^2 R^*(Z) R^*\left(\frac{1}{Z}\right) \Phi_{g_k g_k}{}^*(Z) \frac{a^*(z) a^*\left(\frac{1}{z}\right) z^{+q}}{b^*(z) b^*\left(\frac{1}{z}\right)^2} + 2\gamma R^*\left(\frac{1}{Z}\right) \frac{\mathbb{F}^*(Z) a^*\left(\frac{1}{z}\right) z^{+q}}{b^*\left(\frac{1}{z}\right)^2} \tag{3.2}$$

$$\int E^{2*}(Z) = R^*(Z) R^*\left(\frac{1}{Z}\right) C^*(Z) C^*\left(\frac{1}{Z}\right) \gamma^2 \Phi_{g_k g_k}{}^*(Z) - 2\gamma R^*\left(\frac{1}{Z}\right) C^*\left(\frac{1}{Z}\right) \mathbb{F}^*(Z) + \Phi_{c_d c_d}{}^*(Z) \tag{3.3}$$

(b)

where

$$C^*(Z) = \frac{a^*(z)}{b^*(z)} \tag{3.4}$$

$$\Phi_{g_k g_k}(Z) = Z[G(S)G(-S)] \tag{3.5}$$

$$\mathbb{F}^*(Z) = Z[G(-S)e^{-sk}R(S)H_d(S)] \tag{3.6}$$

$$\Phi_{c_d c_d}(Z) = Z[R(S)R(-S)H_d(S)H_d(-S)] \tag{3.7}$$

γ = impulse duration

TABLE IV

SUMMARY OF BASIC SYNTHESIS PROCEDURE—SIMULATION SYSTEMS—DETERMINISTIC INPUTS

Step Number	Evaluate	From
	(a)	
1	$\Phi_{g_k g_k}{}^*(Z)$	$\Phi_{g_k g_k}(S) = G(S)e^{-sk}G(-S)e^{-sk} = G(S)G(-S)$
2	$\mathbb{F}^*(Z)$	$\mathbb{F}(S) = G(-S)e^{-sk}R(S)H_d(S)$
3	$R^*(Z)$	$R(S)$
4	$\frac{\partial}{\partial a_p}\int \epsilon^2 = 0;\ p = 0, 1 \cdots p_{ms}$	The values of the $(p_{ms}+1)$ time functions corresponding to $\int^{a_p} E^*(Z)$ (3.1) evaluated at $t=0$.
	(b)	
5	$\frac{\partial}{\partial b_q}\int \epsilon^2 = 0;\ q = 1, \cdots q_{ms}$	The values of the q_{ms} time functions corresponding to $\int^{b_q} E^{2*}(Z)$ (3.2) evaluated at $t=0$.
6	$C^*(Z)$	Simultaneous solution of the $(p_{ms}+q_{ms}+1)$ equations defined by steps 4 and 5.
7	$\int \epsilon^2$	The value at $t=0$ of the time function corresponding to $\int E^{2*}(Z)$ (3.3), with T determined from the stated relationship between sampling period and computer program complexity.

Detailed derivations of the synthesis procedures to be described are available.[2] It is possible, without proceeding through these derivations, to acquire an understanding of the capabilities and limitations of the technique, and a physical insight into the fundamental processes that are automatically brought into play by the synthesis procedures.

The three key synthesis equations for systems with deterministic inputs are listed in Table III. In essence, these equations summarize in transform form all of the time domain convolutions, integrations and differentiations required to establish the conditions for optimum operation of a given channel, given the total number of terms in the computer program numerator (p_m+1) and in the program denominator (q_m). These equations contain both negative and positive powers of Z, so that their corresponding time functions have values for both positive and negative time. The synthesis procedure requires that the value at $t=0$ of the time functions corresponding to (3.1) $(p=0, 1 \cdots p_m)$, (3.2) $(q=1, \cdots q_m)$, and (3.3) be evaluated. The results of each of the (p_m+1) evaluations of (3.1) and of the q_m evaluations of (3.2) are then set equal to zero. This effectively constrains the derivatives of the integrated error squared with respect to each of the computer program parameters to be zero. The resultant (p_m+q_m+1) simultaneous equations are then solved for the (p_m+q_m+1) unknowns $(a_0, a_1 \cdots a_{p_m}, b_1, b_{q_m})$. This establishes the optimum values for the computer program parameters. When these values are substituted in the result of the evaluation of (3.3), the resultant number, (still a function of k and T), will correspond to the system integrated error squared. Finally, k, which is a function of single channel computing time, and T, which is a function of total system computing time, can be evaluated, and the integrated error squared will then correspond to a known number. Table IV summarizes the steps involved in the determination of the optimum program parameters and integrated error squared for a computer program of stated complexity (p_m, q_m). In the event that only the output filter form is specified, the filter parameters can also be optimized by differentiating the integrated error squared with respect to each parameter, setting each equation equal to zero, and solving for the optimum parameters.

Note that the factor γ that appears in the basic synthesis equations of Table III corresponds to the duration of the computer output pulse that actuates the channel output filter. Normally, the duration of this pulse is important, since the output filter is specified in terms of its impulsive response, and both the amplitude and duration of the computer output affect the filter output. When the output filter consists of a hold circuit that responds only to the amplitude of the computed output, and is unaffected by the time it takes to actuate the hold, a factor $1/\gamma$ must appear in the hold transform to compensate for this effect. That is, the transform of a first-order hold is simply

$$\frac{(1 - e^{-sT})}{\gamma s}.$$

Data Limited theory provides an upper limit to the number of potential computer programs that must be considered before the optimum Computer Limited program can be established. Dr. Franklin[4] has shown that the optimum Data Limited $C^*(Z)$ is given by the equations listed in Table V. The derivation of the optimum Computer Limited program requires consideration of both this Data Limited solution and all less complex programs, and the evaluation in each case of the corresponding integrated error squared, subject to the computer complexity-sampling period constraint. Based on these evaluations, the optimum Computer Limited program will be evident as the program resulting in least integrated error squared.

For example, if a given Data Limited solution results in the program form shown in Table VI, (6.1) Computer Limited theory would require, in effect, that each of the potential programs shown in (6.2) through (6.4) be considered. Programs of greater length would not have to be considered, since they would always lead to greater integrated error squared. This is the case because Data Limited theory imposes no penalty for computer program complexity, so that the Data Limited program cannot be improved by increasing the number of program terms. However, when the sampling period—computer complexity constraint is taken into account, a longer program would result in a longer sampling period and therefore in greater integrated error squared. In general, the advantage of the Computer Limited approach lies in its ability to indicate shorter programs, and therefore shorter sampling periods, than those nominally dictated by Data Limited theory.

In implementing the procedure described above the system designer can, if he wishes, use only the five term program shown in (6.5), solve the required five simultaneous equations to obtain a general solution to the problem, and simply reduce the unused parameters to zero when considering each potential program in turn. This approach is to be compared with the solution of three sets of three simultaneous equations

$$(p_m = 2,\ q_m = 0), \quad (p_m = 1,\ q_m = 1), \quad (p_m = 0,\ q_m = 2).$$

The selection of a particular procedure can be made to suit the convenience of the system designer, since the results are independent of the procedure.

[4] G. Franklin, "The Optimum Synthesis of Sampled-Data Systems," D. Eng. Sc. Dissertation, Columbia University, New York, N.Y.; May, 1955.

TABLE V

OPTIMIZATION RELATIONSHIPS FOR DATA LIMITED SIMULATION SYSTEMS DETERMINISTIC INPUTS

$$C^*(Z) = \frac{W_1^*(Z)}{\gamma \left\{R^*(Z)R^*\left(\frac{1}{Z}\right)\right\}^+ \{\Phi_{g_k g_k}^*(Z)\}^+} \qquad (5.1)$$

$$W^*(Z) = \frac{R^*\left(\frac{1}{Z}\right) IF^*(Z)}{\left\{R^*(Z)R^*\left(\frac{1}{Z}\right)\right\}^- \{\Phi_{g_k g_k}^*(Z)\}^-} \qquad (5.2)$$

$$= W_1^*(Z) + W_2^*(Z)$$

(W_1^*: Poles Inside; W_2^*: Poles Outside — The Unit Circle)

$$\Phi_{g_k g_k}^*(Z) \triangleq \{\Phi_{g_k g_k}^*(Z)\}^- \{\Phi_{g_k g_k}^*(Z)\}^+ \qquad (5.3)$$

($\{\}^-$: Poles Outside; $\{\}^+$: Poles Inside — The Unit Circle)

TABLE VI

EXAMPLE OF POTENTIAL COMPUTER LIMITED PROGRAMS

Data Limited Program $C^*(Z) = \dfrac{a_0 + a_1 z^{-1}}{1 + b_1 z^{-1} + b_2 z^{-2}}$ (6.1)

Alternate Programs to be Considered

Program	Period	
$C^*(Z)_{a-1} = a_0$	T_a	(6.2)
$C^*(Z)_{b-1} = \dfrac{a_0}{1 + b_1 z^{-1}}$; $C^*(Z)_{b-2} = a_0 + a_1 z^{-1}$	T_b	(6.3)
$C^*(Z)_{c-1} = \dfrac{a_0}{1 + b_1 z^{-1} + b_2 z^{-2}}$; $C^*(Z)_{c-2} = \dfrac{a_0 + a_1 z^{-1}}{1 + b_1 z^{-1}}$;		
$C^*(Z)_{c-3} = a_0 + a_1 z^{-1} + a_2 z^{-2}$	T_c	(6.4)

General Program Containing All Applicable Terms

$$C^*(Z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}} \qquad (6.5)$$

When N similar problems are being simulated, the system sampling period T is N times the channel sampling period, so that a channel sampling period T_j will result in a system sampling period $T = NT_j$. The synthesis of the entire system can then be accomplished by optimizing the program for a single channel. When the N problems to be simulated are not similar it is necessary to consider each combination of potential single channel programs that could produce a given system sampling period, and each potential system sampling period, evaluating in each case the corresponding integrated error squared for each channel. The sum of all channel integrated error squared terms can then be observed and the least value selected as the optimum system operating point. Further details on such a procedure are available.[2]

When the system inputs are mixtures of random message plus random noise, a slightly modified mathematical

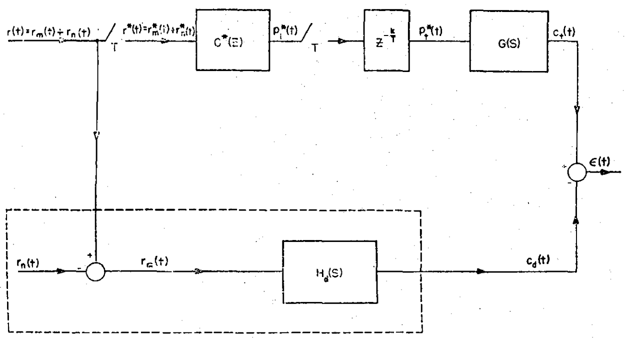


Fig. 6—Simulation channel error—random inputs.

TABLE VII

OPTIMIZATION RELATIONSHIPS FOR COMPUTER-LIMITED SIMULATION SYSTEMS RANDOM INPUTS

(a)

$$\overline{E^{2*}}^{a_p}(Z) = \frac{2\gamma^2}{T}\Phi_{rr}^*(Z)\Phi_{g_kg_k}(^*Z)\frac{a^*(z)z^{+p}}{b^*(z)b^*\left(\frac{1}{z}\right)} - \frac{2\gamma}{T}\frac{F^*(Z)z^{+p}}{b^*\left(\frac{1}{z}\right)} \quad (7.1)$$

$$p = 0, 1, \cdots p_m$$

$$\overline{E^{2*}}^{b_q}(Z) = -\frac{2\gamma^2}{T}\Phi_{rr}^*(Z)\Phi_{g_kg_k}^*(Z)\frac{a^*\left(\frac{1}{z}\right)a^*(z)z^{+q}}{b^*\left(\frac{1}{z}\right)^2 b^*(z)}$$

$$+\frac{2\gamma}{T}\frac{a^*\left(\frac{1}{z}\right)}{b^*\left(\frac{1}{z}\right)^2}F^*(Z)z^{+q} \qquad q = 1, \cdots, q_m \quad (7.2)$$

$$\overline{E^{2*}}(Z) = \frac{\gamma^2}{T}\frac{a^*(z)a^*\left(\frac{1}{z}\right)}{b^*(z)b^*\left(\frac{1}{z}\right)}\Phi_{rr}^*(Z)\Phi_{g_kg_k}^*Z$$

$$-\frac{2\gamma}{T}\frac{a^*\left(\frac{1}{z}\right)}{b^*\left(\frac{1}{z}\right)}F^*(Z) + \Phi_{c_dc_d}^*(Z) \quad (7.3)$$

(b)

$$\Phi_{g_kg_k}^*(Z) = Z[G(S)G(-S)] \quad (7.4)$$

$$\Phi_{rr}^*(Z) = Z[\Phi_{rr}(S)] \quad (7.5)$$

$$F^*(Z) = Z[G(-S)e^{sk}H_d(S)\Phi_{rr_m}(S)] \quad (7.6)$$

$$\Phi_{c_dc_d}(Z) = Z[\Phi_{r_mr_m}(S)H_d(S)H_d(-S)] \quad (7.7)$$

model must be used to define the error between true and ideal system outputs for a given channel. As illustrated in Fig. 6, the main difference lies in the fact that the ideal output $c_d(t)$ is assumed to be derived from an ideal transfer characteristic $h_d(t)$ that is actuated by the random message alone. Since the stationary random signals are assumed present over all time, the continuous mean squared system error, rather than the integrated error squared, is to be minimized. The techniques required to synthesize a system of this type are essentially the same as those required to synthesize a system with deterministic inputs, although a different set of optimization equations now apply. The optimization equations for the synthesis of simulation systems with random inputs are presented in Table VII, the corresponding step by step synthesis procedure is tabulated in Table VIII, and Prof. Franklin's Data Limited solution to this problem is shown in Table IX.

Note that the channel inputs are now defined by the

TABLE VIII

SUMMARY OF BASIC SYNTHESIS PROCEDURE—SIMULATION SYSTEMS—RANDOM INPUTS

Step Number	Evaluate	From
	(a)	
1	$\Phi_{g_kg_k}^*(Z)$	$\Phi_{g_kg_k}(S)=G(S)e^{-sk}G(-S)e^{sk}=G(S)G(-S)$
2	$\Phi_{rr}^*(Z)$	$\Phi_{rr}(S)$
3	$F^*(Z)$	$F(S)=G(-S)e^{sk}H_d(S)\Phi_{rr_m}(S)$
4	$\frac{\partial\overline{\epsilon^2}}{\partial a_p}=0;\ p=1,\cdots p_{ms}$	The values of the $(p_{ms}+1)$ time functions corresponding to $\overline{E^2}^{a_p}*(Z)$ (7.1) evaluated at $t=0$.
	(b)	
5	$\frac{\partial\overline{\epsilon^2}}{\partial b_q}=0;\ q=1,\cdots q_{ms}$	The values of the q_{ms} time functions corresponding to $\overline{E^2}^{b_q}*(Z)$ (7.2) evaluated at $t=0$.
6	$C^*(Z)$	Simultaneous solution of the $(p_{ms}+q_{ms}+1)$ equations defined by steps 4 and 5.
7	$\overline{\epsilon^2}$	The value at $t=0$ of the time function corresponding to $\overline{E^{2*}}(Z)$ (7.3) with T determined from the stated relationship between sampling period and computer program complexity.

TABLE IX

OPTIMIZATION RELATIONSHIPS FOR DATA—LIMITED SIMULATION SYSTEMS—RANDOM INPUTS

$$C^*(Z) = \frac{1}{\gamma}\frac{W_1^*(Z)}{\{\Phi_{g_kg_k}^*(Z)\}^+\{\Phi_{rr}^*(Z)\}^+} \quad (9.1)$$

$$W^*(Z) = \frac{F^*(Z)}{\{\Phi_{g_kg_k}^*(Z)\}^-\{\Phi_{rr}^*(Z)\}^-} \quad (9.2)$$

$$= W_1^*(Z) + W_2^*(Z) \quad (9.3)$$

Poles Inside, Poles Outside — The Unit Circle

$$\Phi_{rr}^*(Z) = \{\Phi_{rr}^*(Z)\}^-\{\Phi_{rr}^*(Z)\}^+$$

Poles Outside, Poles Inside — The Unit Circle

TABLE X

DETERMINING THE VALUE AT $t=0$ OF THE TIME FUNCTION CORRESPONDING TO $\Phi_{rc}^*(Z)$—METHOD 1

$$\Phi_{rc}^*(Z) = \frac{-1.2Z^{+1} + 4.24 - 0.8Z^{-1}}{(1 - 0.6Z^{+1} + 0.08Z^{+2})(1 - 0.4Z^{-1} + 0.03Z^{-2})} \tag{10.1}$$

$$\phi_{rc}(0) = \frac{1}{2\pi j}\oint_{\Gamma} \Phi_{rc}^*(Z)Z^{-1}dz \tag{10.2}$$

$$\Phi_{rc}^*(Z) = \frac{-1.2Z^{+1} + 4.24 - 0.8Z^{-1}}{(1 - 0.2Z^{+1})(1 - 0.4Z^{+1})(1 - 0.1Z^{-1})(1 - 0.3Z^{-1})} \tag{10.3}$$

$$\phi_{rc}(0) = \left.\frac{-1.2Z^{+2} + 4.24Z^{+1} - 0.8}{(1 - 0.2Z^{+1})(1 - 0.4Z^{+1})(Z^{+1} - 0.3)}\right|_{z=0.1} + \left.\frac{-1.2Z^{+2} + 4.24Z^{+1} - 0.8}{(1 - 0.2Z^{+1})(1 - 0.4Z^{+1})(Z^{+1} - 0.1)}\right|_{z=0.3} \tag{10.4}$$

$$\phi_{rc}(0) = 2.062 + 2.200 = 4.262 \tag{10.5}$$

to the form shown in (10.3), so that the integral can be evaluated using its residues inside the unit circle as shown in (10.4) and (10.5). The disadvantage of this approach lies in the requirement for factoring the denominator of $\Phi_{rc}^*(Z)$ in order to determine the pole locations.

A second approach to the problem is to factor $\Phi_{rc}^*(Z)$ into the sum of two parts, one inside and the other outside the unit circle. This approach is presented in Table XI. When the numerator of the assumed and actual polynomials are equated, as shown in (11.1), the set of linear equations listed in (11.2) result. These equations can be solved for α_{20}, the value at $t = 0$ of the time function corresponding to the original two-sided Z transform, as shown in (11.3).

TABLE XI

DETERMINING THE VALUE AT $t=0$ OF THE TIME FUNCTION CORRESPONDING TO $\Phi_{rc}^*(Z)$—METHOD 2

$$\Phi_{rc}^*(Z) = \frac{-1.2Z^{+1} + 4.24 - 0.8Z^{-1}}{(1 - 0.6Z^{+1} + 0.08Z^{+2})(1 - 0.4Z^{-1} + 0.03Z^{-2})} = \frac{\alpha_{20} + \alpha_{21}Z^{+1}}{(1 - 0.6Z^{+1} + 0.08Z^{+2})} + \frac{\alpha_{11}Z^{-1} + \alpha_{12}Z^{-2}}{(1 - 0.4Z^{-1} + 0.03Z^{-2})}$$
$$= \frac{(\alpha_{20} + \alpha_{21}Z^{+1})(1 - 0.4Z^{-1} + 0.03Z^{-2}) + (\alpha_{11}Z^{-1} + \alpha_{12}Z^{-2})(1 - 0.6Z^{+1} + 0.08Z^{+2})}{(1 - 0.6Z^{+1} + 0.08Z^{+2})(1 - 0.4Z^{-1} + 0.03Z^{-2})} \tag{11.1}$$

$$\begin{aligned} 0\alpha_{11} + \alpha_{12} + 0.03\alpha_{20} + 0\alpha_{21} &= 0 \\ \alpha_{11} - 0.6\alpha_{12} - 0.4\alpha_{20} + 0.03\alpha_{21} &= -0.8 \\ -0.6\alpha_{11} + 0.08\alpha_{12} + \alpha_{20} - 0.4\alpha_{21} &= +4.24 \\ 0.08\alpha_{11} + 0\alpha_{12} + 0\alpha_{20} + \alpha_{21} &= -1.2 \end{aligned} \tag{11.2}$$

$$\alpha_{20} = \frac{\begin{vmatrix} 0 & 1 & 0 & 0 \\ 1 & -0.6 & -0.8 & 0.03 \\ -0.6 & 0.08 & 4.24 & -0.4 \\ 0.08 & 0 & -1.2 & 1 \end{vmatrix}}{\begin{vmatrix} 0 & 1 & 0.03 & 0 \\ 1 & -0.6 & -0.4 & 0.03 \\ -0.6 & 0.08 & 1 & -0.4 \\ 0.08 & 0 & 0 & 1 \end{vmatrix}} = \frac{-3.317}{-0.778} = 4.262 \tag{11.3}$$

power density spectra of the input signals (random message plus random noise). Also note that $F(s)$, one of the transforms required in the synthesis procedure, is derived from $\Phi_{rr_m}(s)$, the power density spectrum corresponding to the cross-correlation between message plus noise and message alone.

In order to implement the synthesis procedures that have been described, it is necessary to determine the value at $t = 0$ of various two-sided Z transforms. Before proceeding to a problem example, three techniques for evaluating two-sided Z transforms will be described. The simple two-sided Z transform listed in Table X will be used as an example.

One direct and straightforward method for determining the value at $t = 0$ of the time function corresponding to this transform is to use the relationship presented in (10.2). This approach requires that $\Phi_{rc}^*(Z)$ be factored

The third approach to the problem is illustrated in Fig. 7 and tabulated in Table XII. Referring to the figure, it can be noted that the original two-sided Z transform is the product of a numerator and of two terms in the denominator, one with poles inside and the other with poles outside the unit circle. Both denominator expressions can be expanded to a finite number of terms rational in positive and negative powers of Z respectively. These expansions can be cross-multiplied to obtain a new two-sided Z transform. This transform can then be multiplied with the numerator, considering only terms that contribute to the value at $t = 0$ of the corresponding time function. The example in Fig. 7 shows only three numerator terms and therefore only three terms of the new two-sided transform need be considered. Table XII further demonstrates the process.

As a simple example of the theory that has been outlined, consider the problem of simulating a pure prediction

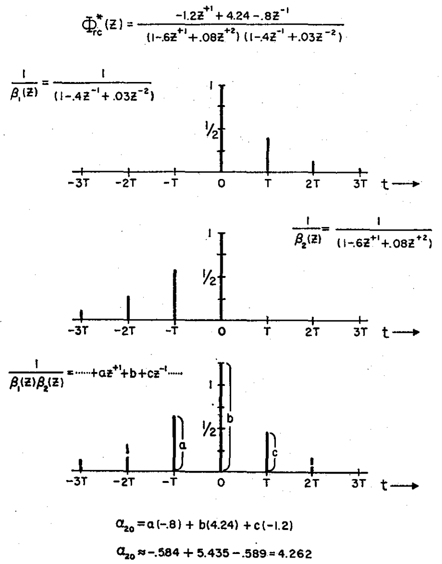


Fig. 7—Approximating the value at $t=0$ of the time function corresponding to $\Phi_{rc}^*(Z)$.

TABLE XII

DETERMINING THE (APPROXIMATE) VALUE AT $t=0$ OF THE TIME FUNCTION CORRESPONDING TO $\Phi_{rc}^*(Z)$—METHOD 3

$$\frac{1}{\beta_1^*(Z)} \approx 1 + 0.4Z^{-1} + 0.13Z^{-2} + 0.04Z^{-3} + 0.121Z^{-4} + \cdots \tag{12.1}$$

$$\frac{1}{\beta_2^*(Z)} \approx 1 + 0.6Z^{+1} + 0.28Z^{+2} + 0.12Z^{+3} + 0.0496Z^{+4} + \cdots \tag{12.2}$$

$$\begin{aligned}\Phi_{rc}^*(Z) &= \frac{(-1.2Z^{+1} + 4.24 - 0.8Z^{-1})}{\beta_1^*(Z)\beta_2^*(Z)} \approx [0.0496Z^{+4} + 0.13984Z^{+3} \\ &\quad + 0.33445Z^{+2} + 0.72958Z^{+1} + 1.28180 + 0.49065Z^{-1} \\ &\quad + 0.15739Z^{-2} + 0.04726Z^{-3} + 0.0121Z^{-4}] \\ &\quad [-1.2Z^{+1} + 4.24 - 0.8Z^{-1}]\end{aligned} \tag{12.3}$$

$$\phi_{rc}(0) \approx 4.2624 \tag{12.4}$$

in each of N simulation channels when the system inputs are random messages with power density spectra

$$\Phi_{rr}(s) = \frac{2b}{b^2 - s^2},$$

each output filter is to be a simple exponential hold of the form $G(s) = 1/(s+a)$, and the constraint between computer program complexity and system sampling period can be approximated by a delay of T_1 seconds for each program term. Since N similar problems are to be solved, the system sampling period is given by

$$T = (p_m + q_m + 1)NT_1,$$

and the delay in a given channel is

$$k = (p_m + q_m + 1)T_1.$$

If the desired prediction for a given channel is α, then

$$H_d(s) = e^{\alpha s}.$$

A Data Limited solution to this problem specifies the program and mean squared error listed in Table XIII, (13.1) and (13.2).[4] When the computer com-

TABLE XIII

DATA LIMITED SOLUTION TO PROBLEM EXAMPLE RANDOM INPUT

$$C_{DL}^*(Z) = \frac{2a}{\gamma} \frac{e^{-\alpha b}}{(a+b)} \frac{(1 - e^{-aT}e^{-bT})}{(1 - e^{-2aT})} (1 - e^{-aT}z^{-1}) \tag{13.1}$$

$$\overline{\epsilon^2} = 1 - \frac{1}{T} \frac{2a}{(a+b)^2} e^{-2\alpha b} \frac{(1 - e^{-aT}e^{-bT})^2}{(1 - e^{-2aT})} \tag{13.2}$$

$$C_{DL}^*(Z) = \frac{2a}{\gamma} \frac{e^{-\alpha b}}{(a+b)} \frac{(1 - e^{-2aNT_1}e^{-2bNT_1})}{(1 - e^{-4aNT_1})} (1 - e^{-2aNT_1}z^{-1}) \tag{13.3}$$

$$\overline{\epsilon^2} = 1 - \frac{1}{NT_1} \frac{a}{(a+b)^2} e^{-2\alpha,} \frac{(1 - e^{-2aNT_1}e^{-2bNT_1})^2}{(1 - e^{-4aNT_1})} \tag{13.4}$$

TABLE XIV

COMPUTER LIMITED SOLUTION TO PROBLEM EXAMPLE RANDOM INPUT

$$C^*(Z) = \frac{2a}{\gamma} \frac{e^{-b(k+\alpha)}(1 - e^{-aT}e^{-bT})}{(a+b)(1 + e^{-aT}e^{-bT})} \tag{14.1}$$

$$\overline{\epsilon^2} = 1 - \frac{2a}{T} \frac{e^{-2b(\alpha+k)}(1 - e^{-aT}e^{-bT})}{(a+b)^2(1 + e^{-aT}e^{-bT})} \tag{14.2}$$

$$C^*(Z) = \frac{2a}{\gamma} \frac{e^{-b(k+\alpha)}(1 - e^{-aNT_1}e^{-bNT_1})}{(a+b)(1 + e^{-aNT_1}e^{-bNT_1})} \tag{14.3}$$

$$\overline{\epsilon^2} = 1 - \frac{2a}{NT_1} \frac{e^{-2b(\alpha+k)}(1 - e^{-aNT_1}e^{-bNT_1})}{(a+b)^2(1 + e^{-aNT_1}e^{-bNT_1})} \tag{14.4}$$

plexity—sampling period constraint is applied to these results, $T = 2NT_1$ and the solutions are as shown in (13.3) and (13.4).[2]

If the Computer Limited synthesis procedure tabulated in Table VIII were carried out for the program $C^*(Z) = a_0 + a_1Z^{-1}$, a solution identical to the Data Limited result would be obtained. When the program $C^*(Z) = a_0$ is considered, however, the optimum program and mean squared error expressions are found to be as shown in Table XIV, (14.1) and (14.2), respectively. When the computer complexity—sampling period constraint is applied to these results, $T = NT_1$, and the solutions are as shown in (14.3) and (14.4).

A comparison (13.4) and (14.4) indicates that when $a = 1/NT_1$, the one term program always results in less mean squared error. When $b = 1/NT_1$, the one term program is superior whenever $a > 0.03/T_1$. The one term program is therefore superior under the majority of potential operating conditions. The reason for the success of the Computer Limited program is illustrated in Fig. 8, where the mean squared error corresponding to (13.2) and (14.2), $a = b = 1/NT_1$, $N = 20$, are plotted as functions of T. Note that if T were not related to computer program complexity the two term program would always be superior to the one term program. This is essentially the assumption of Data Limited theory. The one term Computer Limited program superiority arises because the two term program actually imposes the requirement for a system sampling period of $2NT_1$, while the one term program requires a system sampling period of only NT_1.

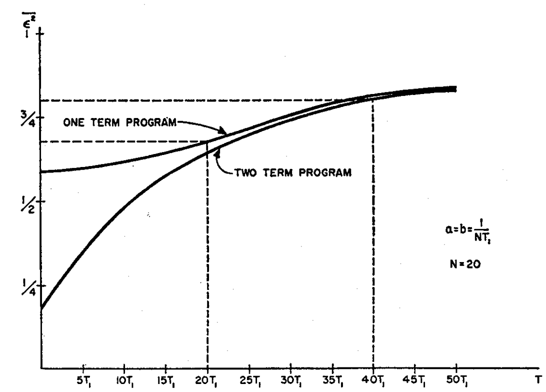


Fig. 8—Mean squared errors—one and two term programs —random inputs.

Further physical insight into the reason for Computer Limited program superiority can be gained by considering the same problem when the system input is an exponential $r(t) = e^{-bt}$ and $\alpha = 0$, $N = 1$. The Data Limited and Computer Limited solutions to this problem are tabulated in Tables XV and XVI respectively.

When $b = 0$, the Computer Limited solution is always superior to the Data Limited solution. The reason for this superiority can be seen from the time responses from the two systems plotted in Fig. 9. Note that after a delay of $2T_1$ the two term program immediately achieves the best response of which it is capable, and that while the one term program response is delayed by only T_1, it requires a transient period to reach its optimum condition. The one term program is superior because its shorter length results in a higher permissible data rate, with a corresponding reduction in achievable integrated error squared. In this particular example there is no penalty associated with the poorer transient response of the one term program, since the time response to a step input extends to infinity, and it is therefore the steady-state response that is being optimized. For this reason the one term program is superior for any finite a.

When b is not zero, so that the system output is exponentially damped, it is possible for the poorer transient response of the one term program to overshadow its higher data rate advantage, in which case the two term program will be superior.

In general, the shorter program has the advantages of a shorter delay before a change in the input is reflected in the output and a higher data rate, and the disadvantage of a poorer transient build-up to the optimum response condition.

The detailed procedures that have been described in this report pertain to the synthesis of linear systems employing linear digital computer programs, in which the minimization of mean squared error or integrated error squared is an effective optimization criterion. It is important to realize that many other classes of Computer Limited sampled data systems exist, and that a great deal of further work is required to extend the basic approach to the Computer Limited problem presented in this report to these problem areas. For example, the problems of simulating nonlinear systems, of utilizing nonlinear computer programs and of employing optimization criteria other than mean squared error minimization present considerable challenge. Many of the actual problems to be simulated by multiplexed real time digital computers are actually nonlinear or can, in specific instances, benefit from the utilization of nonlinear programs. Unfortunately, the sharpness of presently available mathematical tools seems to limit the generality of synthesis techniques applicable to these nonlinear problems.

The solutions presented in this report benefit from the analytic flexibility inherent in the analysis of linear systems. It is submitted that, pending the development of comprehensive nonlinear synthesis techniques, linear theory can provide basic insights into the fundamental problems associated with the synthesis of Computer Limited sampled data systems, and that this fundamental understanding can in turn serve as a guide in the synthesis of systems considerably more complex than those covered by the basic theory.

TABLE XV

DATA LIMITED SOLUTION TO PROBLEM EXAMPLE DETERMINISTIC INPUT

$$C^*(Z) = \frac{2ae^{-bk}(1 - e^{-bT}e^{-aT})}{\gamma(a+b)(1 - e^{-2aT})}(1 - e^{-aT}z^{-1}) \quad (15.1)$$

$$\int \epsilon^2 = \frac{1}{2b} - \frac{2ae^{-2bk}(1 - e^{-bT}e^{-aT})^2}{(a+b)^2(1 - e^{-2bT})(1 - e^{-2aT})} \quad (15.2)$$

$$C^*(Z) = \frac{2ae^{-bk}(1 - e^{-2bNT_1}e^{-2aNT_1})}{\gamma(a+b)(1 - e^{-4aNT_1})}(1 - e^{-2aNT_1}) \quad (15.3)$$

$$\int \epsilon^2 = \frac{1}{2b} - \frac{2ae^{-2bk}(1 - e^{-2bNT_1}e^{-2aNT_1})^2}{(a+b)^2(1 - e^{-4bNT_1})(1 - e^{-4aNT_1})} \quad (15.4)$$

TABLE XVI

COMPUTER LIMITED SOLUTION TO PROBLEM EXAMPLE DETERMINISTIC INPUT

$$C^*(Z) = \frac{2ae^{-bk}(1 - e^{-bT}e^{-aT})}{\gamma(a+b)(1 + e^{-bT}e^{-aT})} \quad (16.1)$$

$$\int \epsilon^2 = \frac{1}{2b} - \frac{2ae^{-2bk}(1 - e^{-bT}e^{-aT})}{(a+b)^2(1 - e^{-2bT})(1 + e^{-bT}e^{-aT})} \quad (16.2)$$

$$C^*(Z) = \frac{2ae^{-bk}(1 - e^{-bNT_1}e^{-aNT_1})}{\gamma(a+b)(1 + e^{-bNT_1}e^{-aNT_1})} \quad (16.3)$$

$$\int \epsilon^2 = \frac{1}{2b} - \frac{2ae^{-2bk}(1 - e^{-bNT_1}e^{-aNT_1})}{(a+b)^2(1 - e^{-2bNT_1})(1 + e^{-bNT_1}e^{-aNT_1})} \quad (16.4)$$

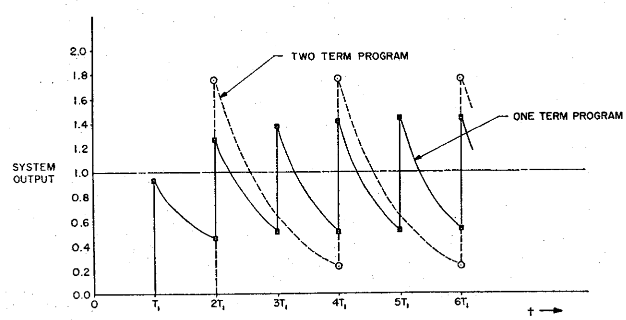


Fig. 9—Data Limited and Computer Limited response for single time constant-output filter—deterministic input.

Discussion

A. G. Favret (American Machine and Foundry Co., Alexandria, Va.): Explain the difference between a one and a two term program to perform the same operation.

Dr. Robinson: Let us say that you are attempting to use a digital computer to simulate a certain dynamic response, such as the response of an aircraft. It is possible to attempt the simulation for a given channel using any one of a host of different programs. A one term program would generate an output equal to the present value of the input, multiplied by a constant. Two different two term programs are possible. One would generate an output equal to the present value of the input multiplied by a constant, plus the value of the input at the prior sampling period, multiplied by a different constant. The other would generate an output equal to the present value of the input multiplied by a constant, plus the value of the output at the prior sampling period multiplied by a different constant. In general, a $(p_m + q_m + 1)$ term program operates on $(p_m + 1)$ present and past values of the input and q_m past values of the output.

A one term program does not require storage of a computed variable and takes a computing time T_1. A two term program requires the storage of one computed variable (corresponding either to the past input or the past output) and takes a greater computing time, for example, $2T_1$. A three term program requires the storage of two computed variables and takes a still greater computing time, and so forth.

S. H. Cameron (Armour Research): In what sense is a program "optimum" in a Data Limited system?

Dr. Robinson: In exactly the same sense as in a Computer Limited system. That is, an optimum Data Limited system is derived by minimizing the mean squared error between actual and ideal system outputs. The difference between systems lies in the factor that limits the system sampling period.

In a Data Limited system, the system sampling period is limited by the data source, for example, the limited speed of rotation of a radar antenna. If the derived data rate is slower than the effective computer speed, long computer programs carry no penalty.

In a Computer Limited system, the system sampling period is limited by the computer speed. A Computer Limited system is characterized by the fact that the longer you make your program, the longer your system sampling period is going to be.

Any system in which a digital computer operates on continuous data is a Computer Limited sampled data system. A system in which a digital computer operates on data that is already sampled could be either Data Limited or Computer Limited, depending on the data rate and the computer speed.

SAGE—A Data-Processing System for Air Defense*

R. R. EVERETT†, C. A. ZRAKET†, AND H. D. BENINGTON‡

The Requirement for SAGE

DURING the past decade, the continental United States has faced the continually increasing threat of enemy air attack. High-speed, high-altitude intercontinental bombers can deliver thermonuclear weapons to any part of our country. Even though ICBM capabilities are rapidly approaching operational status, it is firmly expected that the manned bomber threat will continue and grow well into the 1960 time period. Until very recently, we have relied on an air-defense processing system whose traffic-handling techniques were almost identical with those used during World War II. Fortunately, there has been substantial improvement in our inventory of automated air-defense *components*. These include: improved radar systems, automatic fire-control devices, automatic communication links for ground-to-ground or ground-to-air communication, navigational systems, and both missiles and manned aircraft whose performance equals the threat of the newest manned bombers. But, successful air defense requires both good components and intelligent utilization of these components. A long-range supersonic interceptor is of little value unless enemy targets can be detected and tracked at long ranges. More important, intelligent commitment of many such interceptors requires up-to-date knowledge of the complete enemy threat and of the success of weapons already committed.

In early 1950, the military concluded that the manual air-defense system in use at that time could not adequately coordinate use of our improved hardware against the growing enemy threat. The capacity of the system was too low; the speed with which enemy aircraft could be detected, tracked, and intercepted was too slow; and the area over which an air battle could be closely coordinated was too small. The problem was one of inadequate, nationwide data-handling capability: facilities for communication, filtering, storage, control, and display were inadequate. A system was required which would 1) maintain a complete, up-to-date picture of the air and ground situations over wide areas of the country, 2) control modern weapons rapidly and accurately, and 3) present filtered pictures of the air and weapons situations to the Air Force personnel who conduct the air battle.

The Semiautomatic Ground Environment system—SAGE—was developed to satisfy these requirements. SAGE uses very large digital computing systems to process nation wide air-defense data. SAGE is a real-time control system, a real-time communication system, and a real-time management information system. The basic ideas

* The research in this paper was supported jointly by the U. S. Army, Navy, and Air Force under contract with the Massachusetts Institute of Technology.

† Lincoln Lab., M.I.T., Lexington, Mass.
‡ System Development Corp., Santa Monica, Calif. Formerly at Lincoln Lab., M.I.T., Lexington, Mass.

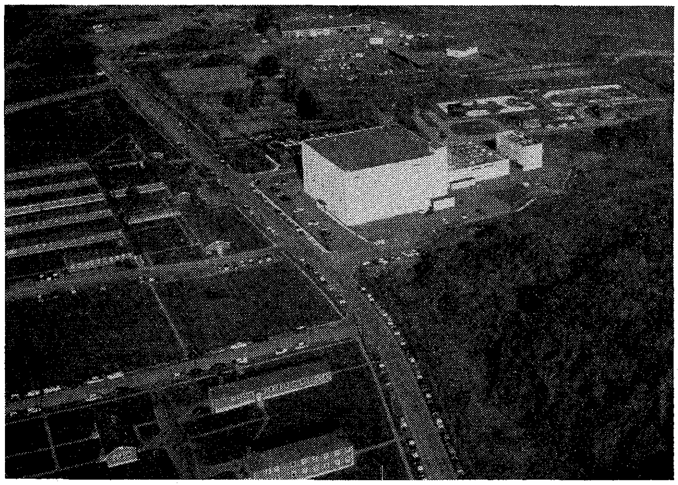


Fig. 1—A SAGE direction center building contains power generation and computing equipment, operational areas for directing sector operation, and office and maintenance facilities. Data are transmitted to this center both automatically and by voice phone. The center communicates with adjacent SAGE centers and transmits guidance data to weapons under its control.

of this system resulted from the efforts of Drs. George E. Valley and Jay W. Forrester of M.I.T.

A large number of organizations have contributed to the development of SAGE since its conception in the Air Force and at M.I.T.'s Lincoln Laboratory. The International Business Machine Corporation (IBM) designs, manufactures, and installs the AN/FSQ-7 combat direction central and the AN/FSQ-8 combat control central including the necessary special tools and test equipment. The Western Electric Company, Inc., provides management services and the design and construction of the direction center and combat center buildings. These services are performed with the assistance of the subcontractor, the Bell Telephone Laboratories. The Burroughs Corporation manufacturers, installs, and provides logistic support for AN/FST-2 coordinate-data transmitting sets. The System Development Corporation (until recently a division of the RAND Corporation) assists Lincoln Laboratory in the preparation of the master computer program and the adaptation of this program to production combat and direction centers. At the present time, SAGE is in production; a prototype unit has been successfully operated for some time.

Sectors and Direction Centers

With SAGE, air defense is conducted from about thirty direction centers located throughout the United States (Fig. 1). A center is responsible for air surveillance and weapons employment over an area called a *sector.* Each center contains a digital computing system—the AN/FSQ-7—containing almost 60,000 vacuum tubes. Over one hundred Air Force officers and airmen within the center control air defense of the sector. Most of these men sit at consoles directly connected to the computer where they receive filtered displays of the computer's storage of system status data; they direct the computer through manual keyboards at each console. The Boston Sector is typical; its direction center is located at Stewart Air Force Base in New York. Its area of responsibility extends from Maine on the north to Connecticut on the south; from New York on the west to a point hundreds of miles off the sea coast on the east.

The computer in the direction center can store over one million bits of information representing weapons and surveillance status of the sector at one time (Fig. 2). These bits represent thousands of different types of information. For example, the computer generates and stores positions and velocities of all aircraft, or it stores wind velocity at various locations and altitudes. Within the computer, a program of 75,000 instructions controls all automatic operations; input data are processed, aircraft are tracked, weapons are guided, outputs are generated. Each second, the computer can generate over 100,000 bits of digital information for display to Air Force operator consoles. Each operator receives cathode-ray tube displays which are tailored to his needs, and he may request additional information or send instructions to the computer by means of keyboard inputs on his console. Each second, the computer can generate thousands of bits of information for automatic digital transmission via telephone or teletype to weapons and missiles, to adjacent centers or higher headquarters, and to other installations within the sector.

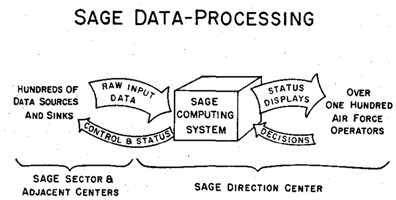


Fig. 2—The direction center continuously receives input data from hundreds of locations within and without the sector. Some of these data are transmitted digitally over telephone lines and read directly into the computer; some are transmitted by teletype or voice phone and transcribed onto punch cards before input to the computer. During one second, over 10,000 bits of data representing hundreds of different types of information can be received at the direction center.

How fast is this system? Obviously, response times from input-to-output vary with the task performed. Fastest response is required by automatic control functions (such as weapons guidance) and for man-machine communication (such as displays of requested information). For many of these functions, only several seconds are required from stimulus to response. For others, several minutes may elapse before the effects of new data are reflected throughout the system. We shall consider now, in somewhat more detail, the three major areas which comprise SAGE data processing. First, the *sector* or *environment* which contains the data sources or sinks coordinated by the direction center. Next, the *man-machine component:* how the operators within the direction center are informed of the air situation and how they affect its progress. Finally, we shall describe the *computing system* which performs the automatic component of the direction center function.

The SAGE Sector

The direction center communicates with over one hundred adjacent installations (Fig. 3). Air surveillance data are received from several types of radars: long-range

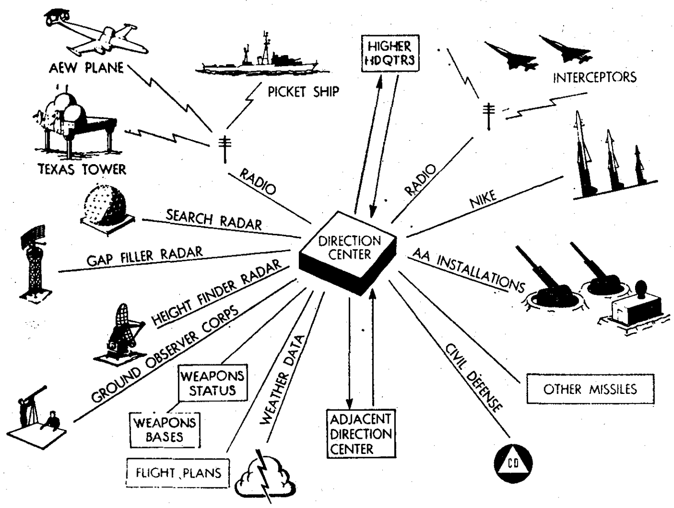


Fig. 3—A direction center receives digitally-coded data automatically and continuously from search radars and height finders over voice-bandwidth communications circuits. Data on flight plans, weapons status, weather, and aircraft tracks are received, respectively, from the Air Movements Identification Service (AMIS), weapons bases, USAF weather service, Ground Observer corps, and Airborne Early Warning and Picket Ships over teletype and voice-telephone circuits. Similarly, data from the direction center are transmitted in digitally-coded form over voice-bandwidth communications circuits to ground-air data-link systems, to weapons bases, to adjacent direction centers and to command levels. Data to other users are transmitted over automatic teletype circuits.

search and gap filler radars located throughout the sector provide multiple coverage of the air volume within the sector; picket ships, early warning (AEW) aircraft and Texas Towers extend this coverage well beyond the coastline; height finders supply altitude data. Within the direction center, these data are converted by the computer to a single positional frame of reference and are used to generate an up-to-date picture of the air situation. Other inputs to the direction center include missile, weapons, and airbase status; weather data; and flight plans of expected friendly air activity. Such data, which are received from many installations within and without the sector, are automatically processed by the computer and used by direction center operational personnel to assist identification of aircraft, employment of weapons, or selection of tactics.

The direction center computer communicates automatically and continuously with adjacent direction centers and command level headquarters in order to insure that air defense is coordinated smoothly between sectors and conducted intelligently over larger areas than a single sector. For example, an aircraft flies out of a sector: surveillance data from the center are automatically transmitted to the proper adjacent center in order to guarantee continuous tracking and interception. In this way, adjacent centers are continuously warning, informing, and acknowledging.

Finally, the direction center continuously transmits status, command, or guidance data to airborne interceptors and missiles or to related ground installations.

Three types of data transmission are used for both inputs and outputs. First, data sources or sinks which require high transmission rates communicate directly with the SAGE computer by means of digitally coded data transmitted at 1300 pulses per second over voice-bandwidth telephone lines and radio channels. Typical applications of this type of channel are inputs from search radars and intercommunication between adjacent centers. Teletype provides a second channel which is slower but equally automatic. In this way, input flight plans are transmitted from Air Movement Identification Services. Finally, voice telephone communications are used in cases where high automaticity is either unnecessary, too expensive, or not feasible. If such information must be entered into the computer, either punched cards or operator keyboard inputs are used.

All data sources and sinks in the sector operate asynchronously. Inputs from each source arrive at the direction center with very different average and peak rates. Each source is processed by the computer with a priority and sampling rate consistent with the role of the particular data in the over-all air defense function. Likewise, the computer generates output messages with a frequency and timing which will insure adequate transmission of guidance and status data and yet utilize finite phone-line and teletype capacity to optimum advantage. *One of the major functions of the SAGE computer is coordination and scheduling in real time of sector inputs and outputs with the manual and automatic functions performed in the direction center.*

The Man in the System

Although SAGE has made many of the data-processing functions in a direction center automatic, many tasks remain which are better performed by the man. Operators can relay computer outputs by phone or radio to adjacent installations and weapons, and they can recognize certain patterns of radar data or tracks more rapidly and meaningfully than any of our present computers and take appropriate action. Most important, operators are required for tactical judgments such as aircraft identification or weapons deployment and commitment. If a major advantage of the FSQ-7 computer is its ability to maintain and store a complete picture of the sector situation, then an equally important advantage is that the *same* computer can rapidly summarize and filter these data for individual presentation to the more than one hundred Air Force personnel who both assist and direct air-defense operations.

The fourth floor of the center contains operational areas from which Air Force personnel supervise the computer and the sector.

Each of the major air-defense functions is supervised from a separate room: radar inputs, status inputs, air surveillance, identification, weapons control, operations analysis, training simulation, and sector command. (Fig. 4.)

Each operator sits at a console which contains display and input facilities tailored to his responsibilities (Fig. 5). The operators insert data into the computer by setting keyboard buttons (Fig. 6). Each console is provided with an input capacity to the computer of 25 to 100 bits of information at one time: the total keyboard input capacity for all consoles is over four thousand bits which are sampled by the computer every several seconds.

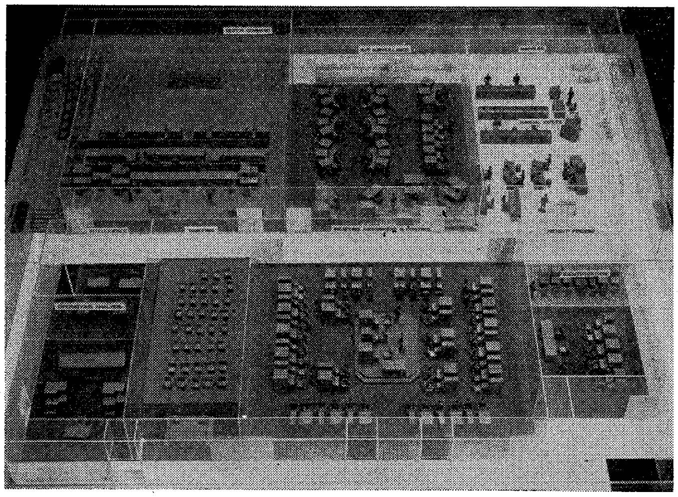


Fig. 4—The fourth floor of the direction center contains separate operational rooms for surveillance, identification, status input, weapons assignment and control, and command functions. Up to 50 operators are required in one room to man the consoles which are directly connected to the computer.

Fig. 5—Each operator sits at a console which contains display and input facilities tailored to his responsibilities.

A 19-inch Charactron[1] cathode-ray tube displays geographically oriented data covering the whole or part of the sector (Fig. 7). On this *air situation display scope,* the operator can view different categories of tracks or radar data, geographical boundaries, predicted interception points, or special displays generated by the computer to assist his decision.

Every two and one-half seconds, the computer displays about two hundred different types of displays requiring up to 20,000 characters, 18,000 points, and 5000 lines. Some of these displays are always present on an operator's situation display. Others he may select. Some he may request the computer to prepare especially for his viewing. Finally, the computer can force very high priority displays for his attention.

The operators' console can also contain a five-inch Typotron[2] digital-display tube which is used to present status or attention data such as weather conditions at several air-

[1] Developed by Hughes Products Co., Los Angeles, Calif.
[2] Developed by San Diego Div., Stromberg-Carlson (formerly Convair Div., General Dynamics Corp.).

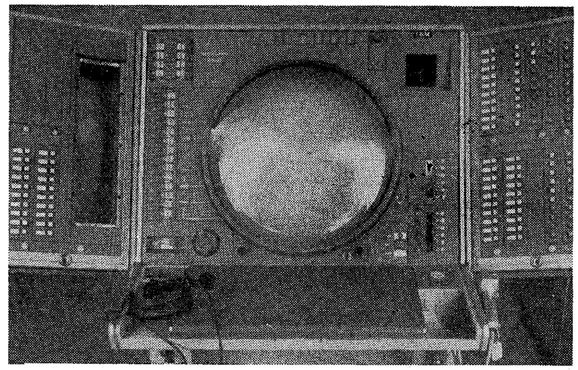


Fig. 6—The operators insert data into the computer through keyboard actions.

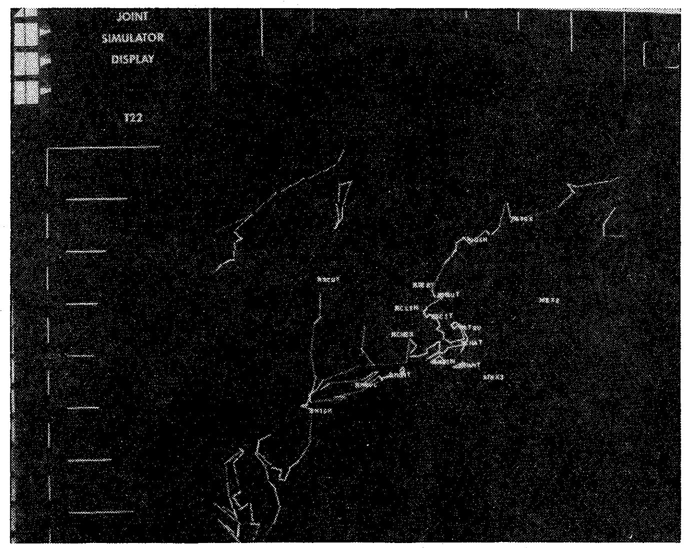


Fig. 7—Situation display. New England coastline, adjacent installations.

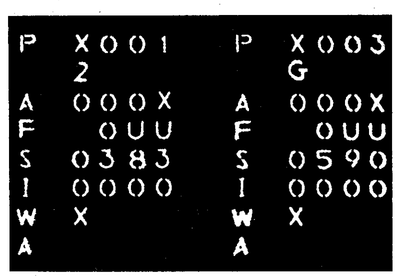


Fig. 8—Typotron digital display.

bases or the "reason" why the computer rejected an operator's actions (Fig. 8). Sixty-three different characters are available in the Typotron. The FSQ-7 display system can display these characters at the rate of 10,000 characters every few seconds to all the digital display scopes.

SAGE Computing System

The SAGE FSQ-7 computer occupies the entire second floor of the direction center. About seventy frames containing almost 60,000 vacuum tubes are required to handle all input-output data, to perform air-defense calculations, and to store system status data. To insure round-the-clock operation two identical computers are required. These are located on opposite sides of the floor with un-

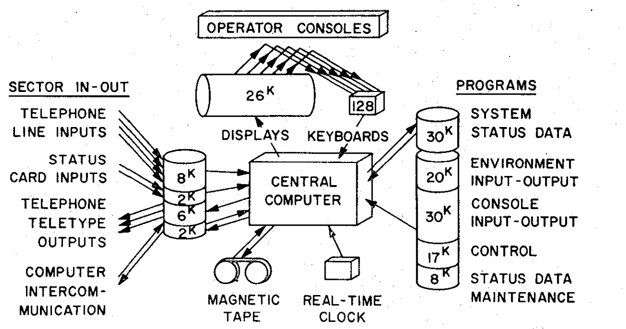


Fig. 9—Each of two identical computers consists of the following major components: a central computer which performs all calculations, the 75,000-instruction air-defense program, and the million bits of system status data. Both of the latter are stored on auxiliary magnetic drums.

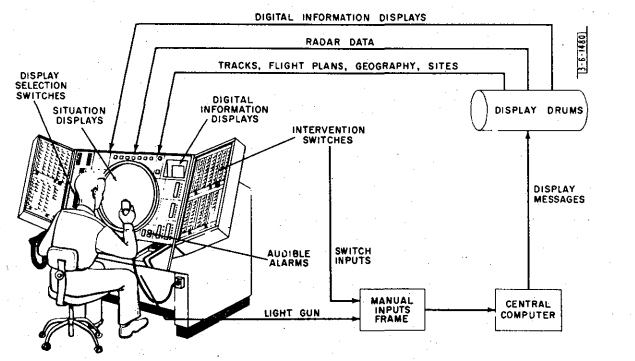


Fig. 10—The processing ability of the buffer devices is fully utilized in the display system. In this case, the central computer maintains a coded table on the buffer display drums. This table is interpreted and displayed by special-purpose equipment each two and one-half seconds at appropriate consoles. The central computer can change any part of the display at any time by rewriting only appropriate words on the drum.

duplicated input-output equipment and maintenance consoles situated in between.

Fig. 9 shows the logical organization of one of the two identical computers. Since only one of these computers performs the real-time air-defense function at any one time, we can discuss simplex processing before considering the problems of duplex operation.

The computer system consists of the following major components: a central computer, the air-defense computer programs, and the system status data stored on auxiliary magnetic drums. The central computer is buffered from all sector and console in-out equipment by magnetic drums (except for the console keyboard inputs which use a 4096-bit core memory buffer). Finally, a real-time clock and four magnetic tape units (used for simulated inputs and summary recorded outputs) complete the FSQ-7 computing system.

The *central computer* is a general-purpose, binary, parallel, single-address machine with 32-bit word length and a magnetic core memory of 8192 words. The memory-cycle time is 6 μsec. Each instruction uses one 32-bit word and the effective operating rate is about 75,000 instructions per second. Four index registers are available for address modification. One unique feature of the central computer is the storage and manipulation of numerical quantities as

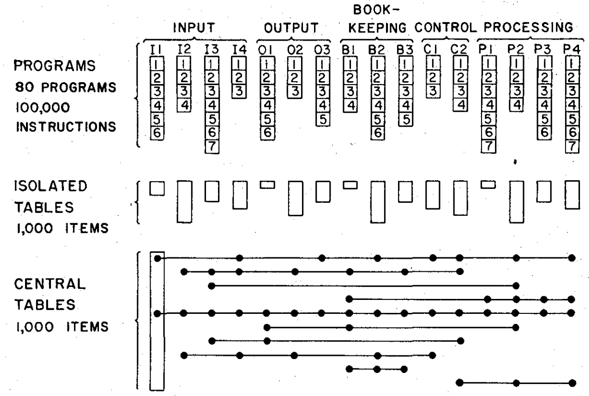


Fig. 11—Static program organization.

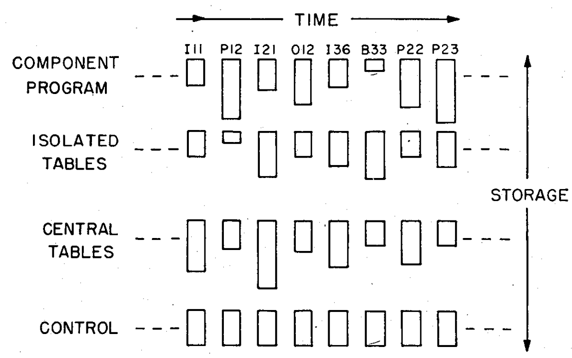


Fig. 12—Dynamic program operation.

two-dimensional vectors with two 16-bit components. Thus, a single sequence of instructions can simultaneously process both components of positional data, effectively doubling computing speed for this type of processing.

Twelve magnetic drums, each with a capacity of 12,288 32-bit words, are used for storage of system status data, system control programs, and buffer in-out data. Under control of the central computer, data can be transferred in variable length blocks between these drums and core memory. The total drum storage capacity is about 150,000 32-bit words.

During an average one-second period the central computer transfers from twenty to fifty blocks of data, each containing 50 to 5000 words, between the central computer's core memory and the terminal devices. In order to insure maximum utilization of the central computer for air-defense processing and control, an *in-out break feature* is used. With this feature, calculations in the central computer continue during input-output operations; they are interrupted only for the one core-memory cycle required to transfer a word betwen the core memory and the terminal device. This feature has proven very valuable since considerably more than 50 per cent of real time is required for input-output searching, waiting, and transferring.

The input-output buffering devices process in-out data independently of the central computer and so free the computer to do more complex air-defense processing. (Separate read-write heads are provided for the buffering equipment and for the central computer.) In their buffer-

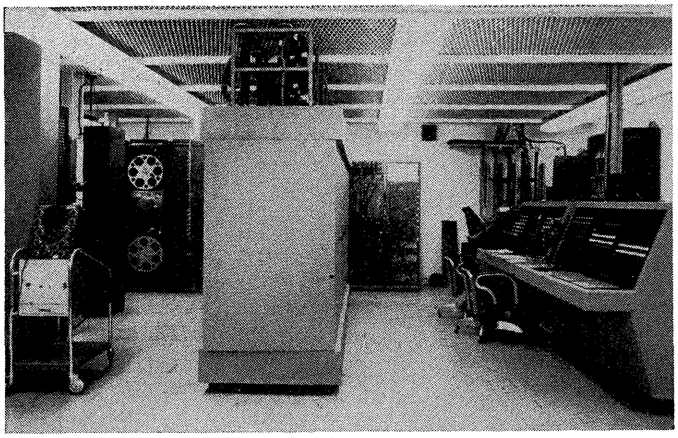


Fig. 13—Inputs room. Digital data transmitted automatically to the direction center via telephone lines can be selected for insertion into the computer at an input patch panel.

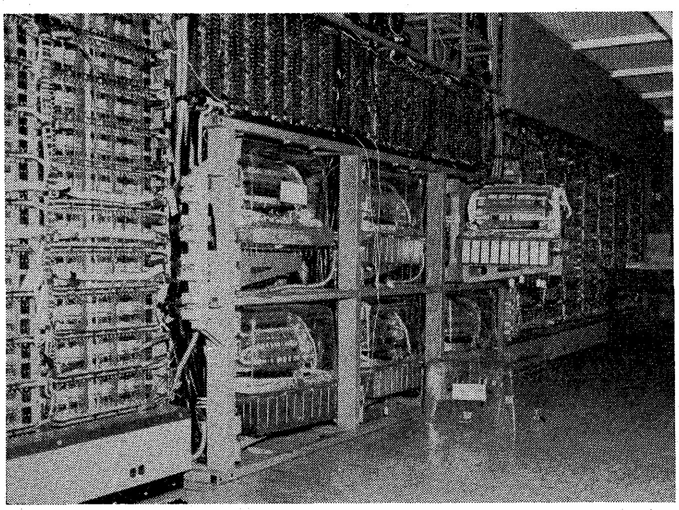


Fig. 14—Drum frames. Magnetic drums are used for buffer storage of input-output data, and storage of system status data and computer programs. Twelve physical drums (six shown) have a capacity for almost 150,000 32-bit words. Half of this capacity is required for storage of the real-time computer program.

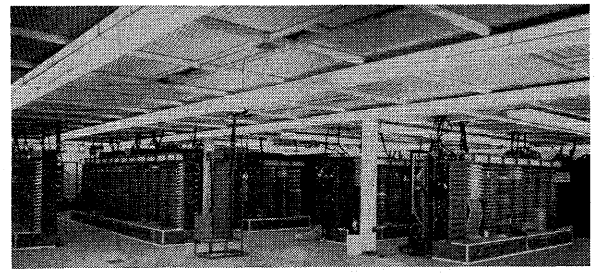


Fig. 15—Central computer frames. There are about 70 frames in the system containing nearly 60,000 vacuum tubes.

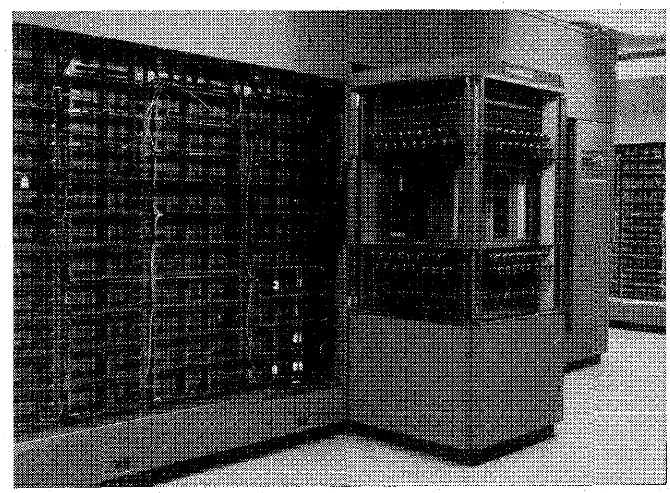


Fig. 16—Magnetic core memory. The central computer is a binary, parallel machine with an 8,192-word core memory and a speed of roughly 75,000 single-address instructions per second. Numbers are stored and processed as vectors with two 16-bit components in order to facilitate processing of positional data.

ing role, these devices can receive or transmit data while the computer is performing some unrelated function.

Consider, for example, the general manner in which input data from voice bandwidth phone lines is received. The serial 1300-pulse-per-second message is demodulated and stored in a shift register of appropriate length. When the complete message has been received, the message is shifted at a higher rate into a second shift register (whose length is a multiple of 32 bits). In this way, the first register is free to receive another message. When the first empty register is located on the input buffer drum, parallel writing stores the word in 10 μsec. A relative timing indicator is also stored on the drum with the message since the computer may not process the message for several seconds and since time of receipt at the direction center is often critical. The central computer can read this randomly stored data by requesting a block transfer of occupied slots only. Output messages are processed conversely. In a few milliseconds, the central computer can deposit a series of messages on the output buffer drum which will keep several phone lines busy for ten seconds.

The processing ability of the buffer devices is fully utilized in the display system (Fig. 10, p. 152). In this case, the central computer maintains a coded table on the buffer display drum. This table is interpreted and displayed by special purpose equipment each two and one-half seconds at the appropriate console. The central computer can change any part part of the display at any time by rewriting only appropriate words on the drum.

The central computer performs air-defense processing in the following manner (see Figs. 11 and 12). The buffer storage tables, the system status data, and the system computer program are organized in hundreds of blocks, each block consisting of from 25 to 4000 computer words. A short *sequence control* program in the central computer's core memory transfers appropriate program or data blocks into core memory, initiates processing, and then returns appropriate table blocks (but never programs) back to the drum. In order to take advantage of the in-out break feature, the operation of each air-defense routine is closely coordinated with the operation of sequence control program so that programs and data are transferred during data processing.

By time-sharing the central computer in this way, each of the air-defense routines is operated at least once every minute; many of the routines are operated every several

Fig. 17—Control console. Separate-control consoles (including standard IBM punched-card equipment) and magnetic-tape units are provided for each of the duplexed computers.

Fig. 18—Manual inputs. The direction center receives status data by voice phone and teletype from weapons installations, air-traffic control centers, weather stations, etc. These data are transcribed onto IBM punched cards and read directly onto a buffer-input drum.

seconds. One interesting feature is that the frequency of program operation is locked with real time rather than allowed to vary as a function of load. During light load conditions the sequence control program will often "mark time" until the real-time clock indicates that the next operation should be repeated. Such synchronization with real-time simplifies many of the control and input-output functions without causing any degradation in system performance.

Reliability

One last aspect of the computing system remains to be discussed—reliability. As mentioned earlier, 24-hour-per-day uninterrupted operation of the computing system was a requirement which could not be compromised. The FSQ-7 is a crucial link in the air-defense chain: if the computing system stops, the surveillance and control functions are interrupted, men and machines throughout the sector lose vital communications, and the sector is without air defense.

In order to insure continuous system operation, any component whose failure would cripple the system has been duplexed whenever possible. As a result, two complete, independent computers are provided; each has

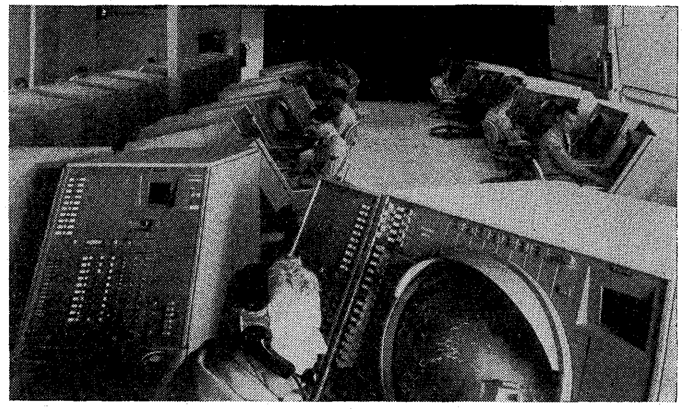


Fig. 19—The air surveillance room contains the Air Force operators who direct aircraft detection and tracking, and communicate with adjacent direction centers. The operator in the foreground is informing the computer to assign one of the tracks shown on his 19-inch cathode-ray display (Charactron) to another operator for special monitoring. Situation displays on this tube can be forced by the computer or requested by the operator. The small five-inch tube (Typotron) is used for display of tabular status data. In addition, the console contains keyboard facilities for inserting data into the computer and telephone facilities providing appropriate priority communications with other stations within and without the direction center.

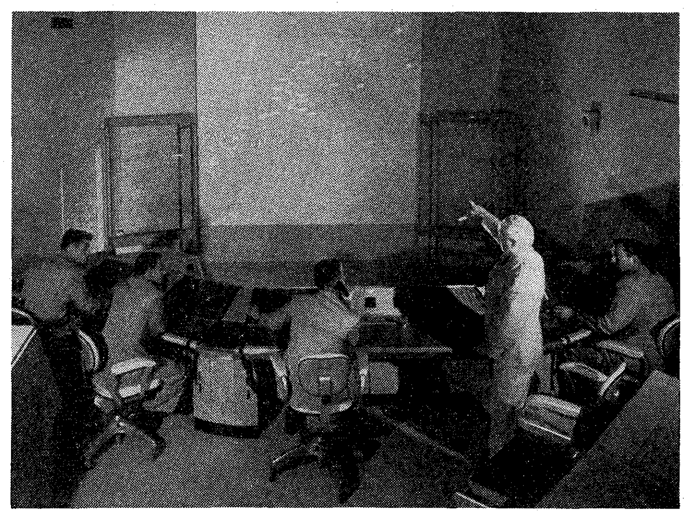


Fig. 20—Command post (Experimental SAGE Sector). Operation of the direction center and the sector are supervised in the command post by the sector commander and his staff. In addition to normal console facilities, a large board display shows the over-all air situation within the sector and adjoining areas. This display is projected from periodic photographs of the computer-generated console displays.

separate drums, central computers, input-output buffering devices, and magnetic tapes. Equipment associated with individual input-output channels is generally not duplicated: consoles, phone-line demodulators, shift registers, etc. Loss of one of these equipments would merely cause loss of some data and minor system degradation, rather than complete shutdown of the direction center.

At any one time, one computer performs the air defense job—this is the "active" computer. The "stand-by" machine may be operating in one of several modes: it may be down for repair (unscheduled maintenance time), it may be undergoing routine preventive maintenance (marginal checking), or even assisting in the maintenance of other equipments within the sector.

Switchover consists of interchanging the roles of each computer: the stand-by machine goes active, the active machine goes to stand-by. Simplexed devices which had been connected to one machine are automatically transferred to the other and the air-defense program begins operation in the newly active machine. From an equipment point of view, switchover requires only a few seconds. However, all of the system status data which had been available before switchover must be available to the newly active computer. Otherwise, the entire air-situation picture would need to be regenerated and this would cripple sector operations as effectively as if both computers stopped. Accordingly, the active machine transmits changes in the air-situation data to the stand-by machine several times per minute via an intercommunication drum. Computer switchover is hardly noticeable to operating personnel.

Although the requirement for continuous operation is a stringent one, SAGE is less vulnerable than many other digital computer applications to transient errors in the FSQ-7. For most operations, the computer operates iteratively in a feed back loop. In these applications, the system is self-correcting for all but a few very improbable errors. Parity checking circuits in the input and output buffer equipment and in the computer memory system eliminate some data subject to transient errors.

Bibliography

[1] Ogletree, W. A., Taylor, H. W., Veitch, E. W., and Wylen, J. "AN/FST-2 Radar-Processing Equipment for SAGE," this issue, p. 156.

[2] Vance, P. R., Dooley, L. G., and Diss, C. E. "Operation of the SAGE Duplex Computers," this issue, p. 160.

[3] Astrahan, M. M., Housman, B., Jacobs, J. F., Mayer, R. P., and Thomas, W. H., "The Logical Design of the Digital Computer for the SAGE System," *IBM Journal of Research and Development*, Vol. 1 (January, 1957).

[4] Benington, H. D. "Production of Large Computer Programs," *Proceedings of the Symposium on Advanced Programming Methods for Digital Computers*, sponsored by the Navy Mathematical Computing Advisory Panel, and the Office of Naval Research (1956).

[5] Israel, D. R. "Simulation in Large Digital-Control Systems," presented at the National Simulation Conference, Houston, Texas, April, 1956.

Discussion

W. G. Cumber (American Airlines, Inc., New York, N.Y.): When introducing data to SAGE by means of IBM punched cards, has it been found necessary to verify this data in the normal manner?

Mr. Benington: For some punch card inputs, IBM verifiers are used before the data are inserted into the computer. In addition, the real-time SAGE computer program extensively tests input data for reasonableness. Several studies now in progress are aimed at improving both manual and automatic capabilities of this input channel.

H. Siegal (Remington Rand Univac): How does the over-all system handle overlapping of information between sectors?

Mr. Benington: Adjacent direction centers can communicate both automatically (from the computer in one center to the computer in the next) and manually (from an operator sitting at a console in one center to an operator in the next). A wide variety of automatic and manual operating procedures constantly attempt to correlate overlap data.

Mr. Siegal: Is the computer capable of recognizing duplication of input data?

Mr. Benington: I understand you to mean, can the computer correlate different pieces of input data which actually were caused by the same stimulus? For example, two adjacent centers can report two incoming tracks which are, in fact, the same track. In many, many cases and in many, many ways, the SAGE computer attempts to correlate each piece of input data with other data and with the computer's history of system status. As a result of these correlations, many input data can be combined or rejected. Aircraft tracking performed by the computer is exactly this type of process. Several radars will transmit multiple returns at different times on one track. These returns must be correlated with a stored history of aircraft tracks (not radar returns) and the appropriate returns must be used to update the track data.

S. C. Redd (Bell Telephone Labs., Whippany, N.J.): Does semiautomatic imply that SAGE is ultimately controlled by a human?

Mr. Benington: In one sense, yes. SAGE is a *management information processing* system (in addition to being a control or communication system) insofar as raw air-defense data is processed, combined, abstracted, and displayed to human operators who make better tactical decisions as a result of receiving an up-to-date, filtered, complete picture of system performance and status.

Mr. Redd: If so, is there any provision for the SAGE computer to assume these manual functions? For example, in the case of more rapid air traffic, IRBM, etc.?

Mr. Benington: In our initial design of SAGE, we attempted, for the most part, to automate existing air-defense procedures rather than to overthrow completely these procedures and attempt both automation and revolution of the air-defense operations. For example, theoretically, SAGE could automatically commit weapons; we felt that this should primarily remain a human decision in the initial design of SAGE. (This type of conservatism has been shared by many other new users of electronic data processing equipment.) At present, several groups are profitably studying which parts of SAGE should be further automated; equally important, which parts of SAGE should be more "manual." Military security precludes a more detailed discussion of specific examples.

P. L. Phipps (Remington Rand Univac): How many man-years has it taken to write the 75,000 instructions?

Mr. Benington: It depends on your definition of "write." If you include programmer training and necessary research and development, the cost is astronomical. Even if you omit these charges, the cost is probably in the range of $25 to $75 per instruction, which I believe is an order of magnitude higher than that quoted for much smaller programs. See my article "Production of Large Computer Programs," *Proceedings of the Symposium on Advanced Programming Methods for Digital Computers*, Office of Naval Research, 1956.

J. A. Cheeseman (Telecommunications Div., General Services Administration, U. S. Government): What type of circuits are used between locations for the transmission of data in and out of the computer?

Mr. Benington: Standard, high-grade, telephone lines are used for automatic transmission of digital data. Each line can transmit 1300 pulses per second. If higher capacity is required for one communication channel, several lines are used in parallel. In order to assure high system reliability, many of these channels are duplexed; that is, between two sites, separate lines are provided over separate routes.

C. L. Kettler (Texas Institute, Inc.): Your direction center seems rather unprotected from the one bomber in a thousand that gets through. What happens to the sector if this headquarter is destroyed? Is there an alternate control center?

Mr. Benington: A sector can fail if the direction center processing equipment fails, if the communications lines fail to the extent that grossly inadequate data is available, or if the center itself is severely damaged. Duplexed central computers protect us against us against the first contingency. Duplexed telephone lines provide some defense against the second. If the sector should fail or if operations should be seriously degraded, then adjacent sectors and direction centers can expand their areas of responsibility to cover the lost sector.

AN/FST-2 Radar-Processing Equipment for SAGE

W. A. OGLETREE†, H. W. TAYLOR†, E. W. VEITCH†, AND J. WYLEN†

Introduction

THE Coordinate Data-Transmitting Set AN/FST-2 is a real-time special-purpose digital-data processing machine which is operated at heavy radar sites as part of the SAGE air-defense system. The AN/FST-2 has a twofold mission in SAGE, conveniently identified as the Fine-Grain Data (FGD) function, and the Semiautomatic Height Finder (SAH/F) function.

The FGD functions are essentially 1) the acceptance of raw data from surveillance radars and Mark X (IFF) equipment, 2) the detection of targets, 3) the determination of the range and azimuth of the targets, 4) the labelling of Mark X targets, 5) the temporary storage of target information, and 6) the processing of the data for the digital-data telephone-service equipment which transmits it to a SAGE Direction Center (DC).

The SAH/F functions are 1) the acceptance of request messages from the direction center for height or other information about specific targets, 2) the processing of the message to provide bearing information to the height-finder radar antennas and visual information to the operators concerning specific targets, and 3) the preparation of the operator's decisions for transmittal to the direction center.

Physical Description

Because the SAGE system must be operational 24 hours a day every day, the AN/FST-2 is a duplex system, consisting of two identical simplex systems. Since no data are stored in AN/FST-2 for more than a few seconds, no necessity for cross-telling of data exists between halves of a duplex, such as is done in the AN/FSQ-7 at the direction center.

The AN/FST-2 is intended for fixed installation in a building at a heavy radar site. The electronic equipment and power supplies for a duplex are contained in 21 air-conditioned cabinets. The cabinets are arranged in three rows of 8, 5, and 8 cabinets each. For reliability, either of the two power supplies, which are located in four of the five center-row cabinets, can be switched to either of the simplex systems by means of switching circuitry located in the fifth center cabinet. A clear view of the data processing cabinets is shown in the photograph of a simplex equipment (see Fig. 1). Several cathode-ray-tube display consoles and operators' consoles also are part of the system. Approximately 43.5 kva of prime power is required to operate the duplex system, and an air-conditioning capacity of about 40 tons is needed for cooling.

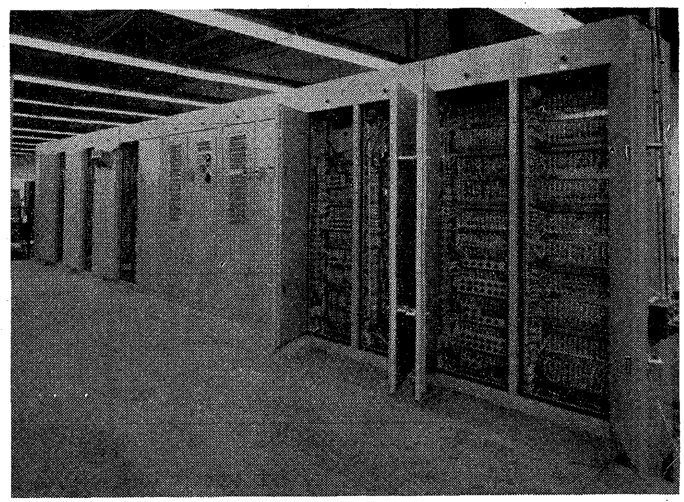


Fig. 1—Data processing cabinets for simplex equipment.

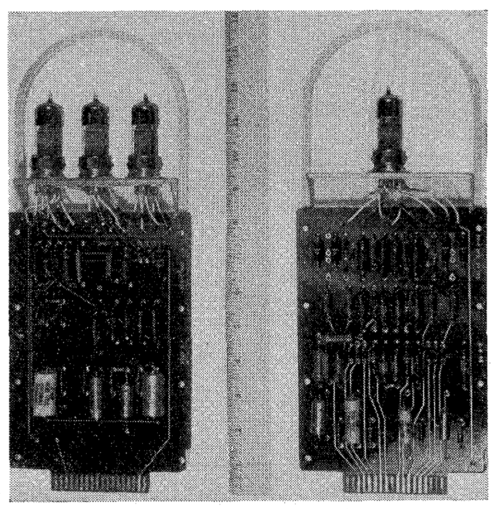


Fig. 2—AN/FST-2 plug-in package—component side.

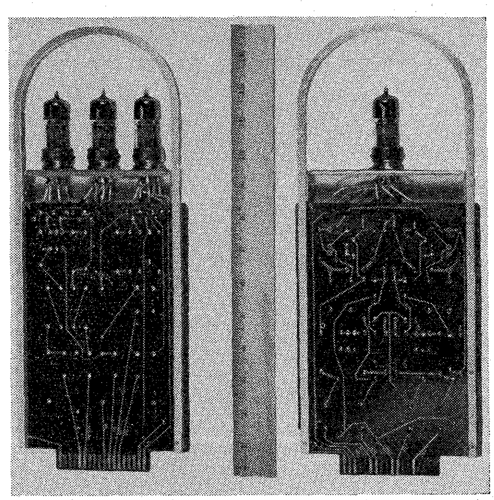


Fig. 3—AN/FST-2 plug-in package—wiring side.

† Burroughs Corp., Paoli, Pa.

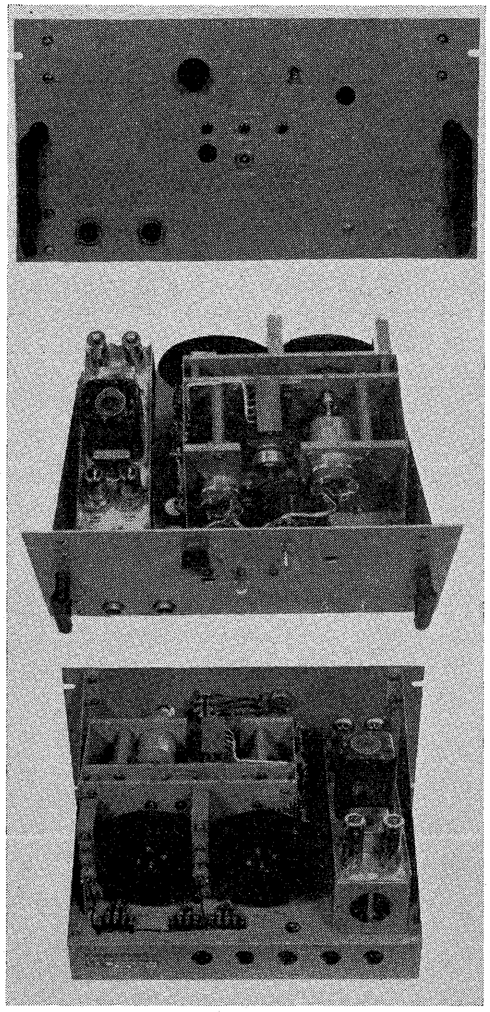


Fig. 4—AN/FST-2 chassis—azimuth mark generator.

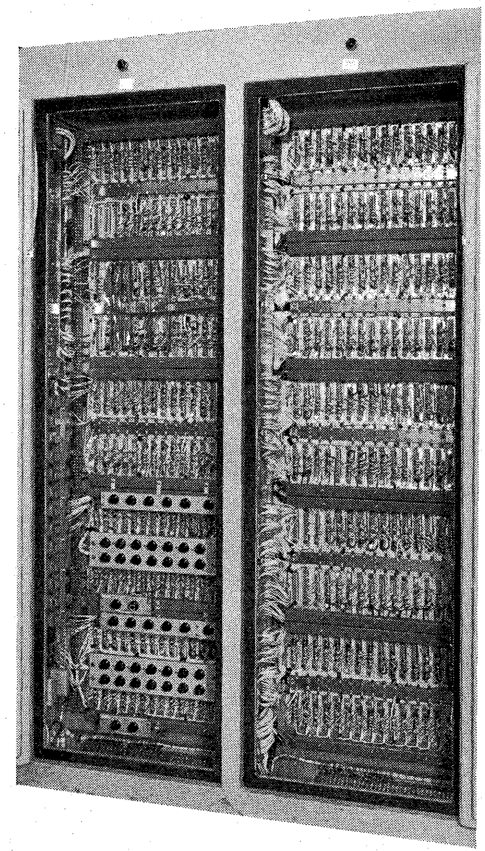


Fig. 5—AN/FST-2 cabinet—wiring side.

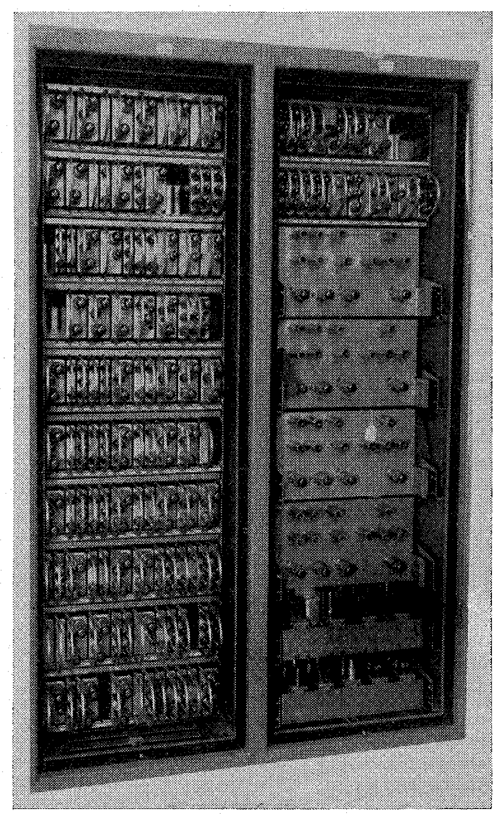


Fig. 6—AN/FST-2 cabinet—tube side.

The machine is designed with diode logic and vacuum-tube circuits. Approximately 6900 vacuum tubes and 24,000 diodes are used per duplex. The basic building module is a printed circuit plug-in package such as that shown in Figs. 2 and 3. Computer circuits, such as flip flops, buffer and inverter amplifiers, gates, pulse amplifiers, and diode matrix groups, are assembled as complete units on these packages. This approach facilitates machine assembly, simplifies trouble isolation, and permits a "replace and repair" maintenance procedure. Circuits with large components or special requirements in wiring or fabrication are assembled on standard chassis, such as that shown in Fig. 4. These chassis are mounted in the cabinets on sliding racks to permit easy access. Figs. 5 and 6 show two views of AN/FST-2 cabinets (note chassis arrangement in Fig. 6).

Functional Description

Fine Grain Data Section (FGD)

The input circuits of the FGD accept the output signals from the search radar and Mark X (IFF) equipment. These signals are processed so as to eliminate or minimize the effects of radar-receiver noise and residual ground, sea, and cloud clutter. Digital circuits within the FGD provide a feedback signal to the input circuits to assist in the input filtering process. Radar signals which are accepted by the FGD input circuits are quantized in range, and initiate a standard pulse for further processing in the magnetic-drum portion of the machine.

The quantized video signals which pass the input section of the FGD are stored on a high-speed magnetic drum. The drum serves as a set of delay lines to store data associated with each range while the computer circuitry is processing other ranges. Each track on the drum utilizes a write head and a read head. The rotation speed of the drum is accurately synchronized with the PRF of the radar transmitter by a sensitive servosystem, so that the delay time on the drum between the write and read heads corresponds to the radar PRF. The target range stored on the drum is thus represented by the amount of time its appearance lags behind the radar trigger pulse. Digital range information is provided by a radar-synchronized precision range mark generator and an appropriate flip-flop counter.

The detection of targets is achieved by a technique which is conveniently referred to as "sliding window" detection. The quantized video returns from a number, N, of consecutive radar pulse transmissions are stored on recirculating "detector" tracks on the magnetic drum. As the returns from the latest "main bang" are written onto the drum, the signals from the oldest are eliminated. At any given instant, the drum read heads on the N detector tracks of the drum are examining the quantized video returns at a given range R_1. An accumulator and comparator circuit determines whether the number of stored video returns, n, $(0 \leq n \leq N)$ is equal to a preset threshold value, n_1. The number n_1 can be selected by the choice of a plug-in package to establish the statistical criterion for the leading edge of a target.

As long as the target remains within the beam of the transmitted radar pulse, it is expected that the number of video returns in the R_1 range block on the drum will remain greater than n_1. As the beam rotates, however, and the target gets into the trailing edge of the beam, the number of returns will decrease. A second comparator, switched into operation when the leading azimuthal edge of a target is observed, is used to determine when n becomes equal to n_2, the statistical criterion for the trailing edge. The number n_2 is normally lower than the number n_1. Thus the leading azimuthal edge of a target is determined when the number of video returns at range R_1 is n_1 out of a possible N, and the trailing edge is determined when the number of video returns decreases to n_2 out of a possible N.

Upon determination of the leading edge of a target, the FGD initiates the "beam-splitting" process which calculates the azimuth (bearing) of the target to great accuracy (see Fig. 7). The rotation of the search radar antenna is converted into azimuth pulses which are counted in the FGD azimuth counter. The azimuth counter stores, at any given instant, the azimuth of the antenna in digital form. A "north-marking pulse" synchronizes the counter once each revolution. It should be noted, however, that rotation is asynchronous with the radar trigger PRF, and also that the rotation velocity is not necessarily uniform.

The azimuth of the target is considered to be the average of the leading-edge azimuth and the trailing-edge azimuth. A set of channels on the magnetic drum is used to aid in this calculation. At the leading edge of the target a zero is written in these channels in the appropriate range block (determined by drum position). As long as the radar beam is on target, as determined by the detector, every azimuth change pulse is added into the number stored in these drum channels at half rate. This is achieved by providing a digital counter which counts the azimuth change pulses between radar main bangs and then adding one half of this number into the stored accumulation each time the R_1 range block passes under the drum read heads. If at the trailing edge of the target this number were subtracted from the number in the azimuth counter, the result would be the azimuthal center of the target. However, since the output telephone equipment may not be immediately available, the accumulated count remains on the drum and continues to accumulate azimuth change pulses, but at full rate. When the target is transferred to the output registers, this final accumulation is subtracted from the number in the azimuth counter at that time to give the true azimuth of the target (see Fig. 7).

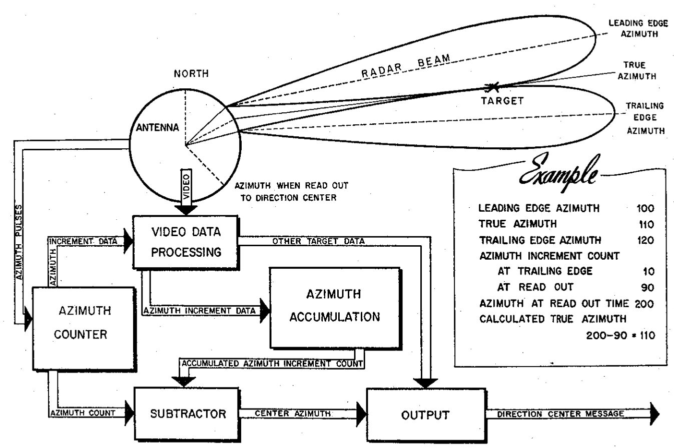


Fig. 7—AN/FST-2 beam splitting.

The target information is transmitted from the AN/FST-2 to the SAGE Direction Center. The target message, which is made up in the output section, includes range data, azimuth data, "run-length" (representing the number of azimuth change pulses between the leading and trailing edge of the target), "time-in-storage" (see below), the message label, and a sync pulse; azimuth information is taken from the drum, and digital range information from the range register. The data are transferred at high speed into a magnetic shift register whenever the telephone equipment is ready to transmit a new message. The data are then transferred (at a 1300-bit-per-second rate) from the shift register into the converter, which changes the pulse train into a sinusoidal signal. The present communication link to the SAGE Direction Center can transmit up to 50 target messages per second.

When the air traffic is heavy, it may be necessary for the AN/FST-2 to store target information for up to several seconds before the digital-data transmitter can accept it for transmission to the direction center. A "clock" circuit in the AN/FST-2 is used to measure the time between the trailing edge of the target and its transfer to the output section. This storage time is significant information, and is transmitted to the direction center. If the target coordinates are stored in the AN/FST-2 for longer than a predetermined time, the information is erased. The assumption is made that the target will be detected again on the next antenna scan, and the "old" information is useless at the direction center. Erasure of valid target data would occur only in an unusually heavy air-traffic situation.

The AN/FST-2 processes Mark X (IFF) targets in a separate channel of the equipment. Because the airborne transponder returns are usually stronger than the "skin" returns received by a search radar, the detection criterion for Mark X targets is different. A Mark X target is denoted by a special bit in the message-label portion of the AN/FST-2 output word.

Other messages which are transmitted to the direction center and marked by special codes in the message label are 1) the "test target," which is sent once per scan and provides a limited means for checking the AN/FST-2 operation, and 2) the SAH/F reply word, which is multiplexed into the same telephone equipment and takes precedence over other messages for transmission.

Semiautomatic Height Finder (SAH/F)

The SAH/F section of the AN/FST-2 identifies and automatically accepts digital request messages received via telephone lines from the direction center. These messages normally contain the x and y coordinates of a target for which the direction center needs up-to-date height data; the request includes the estimated or last known height of that target and such essentials as the address of the radar site and the identification number assigned to the target.

By means of digital-to-analog conversion and a servosystem which includes a resolver, the digital x-y coordinates of the target are simultaneously transformed into polar coordinates and converted to analog form. The angle becomes a shaft position which, in turn, is transmitted to the height-finder radar antenna. The range becomes a voltage which is utilized to generate a range-strobe pulse. Thus the SAH/F causes the height-finder antenna to be aimed at the target, and to generate a range pulse which marks the known range of the target on a Range-Height Indicator (RHI). The SAH/F also converts the digital data contained in its height register into a dc voltage.

Upon receipt of a request message by the SAH/F, and while the antenna is being oriented, the operator at the RHI associated with the height-finder radar is alerted by a visual signal on his console. Range and height lines appear on the CRT face of the RHI along with the video (see Fig. 8).

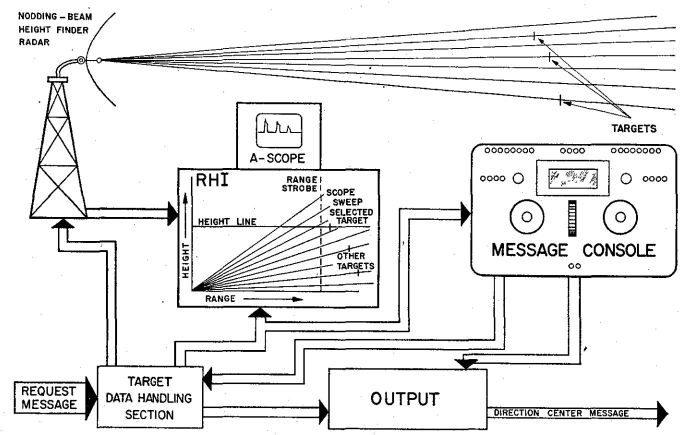


Fig. 8—Semiautomatic height-finder operation.

The elevation of the antenna and the video target returns are presented by the radar. The range-strobe from the AN/FST-2 brightens the scope sweep each time it passes the proper range point, and the height voltage from the SAH/F is interlaced with the sweeps to present a height line. The selected target will appear near the "intersection" of the horizontal height line and the vertical line made by the range-strobe points.

The operator's console contains a height wheel whose motion is converted into pulses which are counted into the height register in the SAH/F. The direction of rotation of the wheel determines whether the pulses are added or subtracted in the height register. The operator's assignment is to observe the relative positions of the target and the height line, and to rotate the height wheel until the height line is centered on the target. At that time he presses a release button which freezes the information in the height register until it is automatically transmitted back to the direction center.

Facility is provided in the SAH/F to permit the console operator to perform additional data-gathering functions. Special requests (for example, formation, number of aircraft) may be received from the direction center. These are communicated to the operator by means of visual displays on the console. An A Scope with a 5-mile range expansion provides the operator with a close-up view of the area around the selected target. Switches on the console permit the operator to select predetermined digitally coded messages for inclusion in the reply word to the direction center. Each simplex section of the AN/FST-2 is designed to operate with two height-finder radars and two console operators.

Maintenance and Reliability

The function of the AN/FST-2 has been described in simplified "block diagram" form. The actual design and operation of the machine is quite complex, as might be indicated by the size of the equipment and by noting that the radar PRF, the antenna rotation, and the telephone bit rate are asynchronous. This tends to make troubleshooting and signal tracing somewhat difficult.

The AN/FST-2 was designed using the "worst-case" philosophy to achieve reliability. In this approach, all circuit components are assumed to be at their "end-of-life" values simultaneously, in the worst direction for design purposes. For example, a 1 per cent resistor is assigned a design value 5 per cent from nominal in the direction which will make performance most marginal. This approach, combined with the use of the best available tubes, diodes, and other components, is expected to provide the optimum in equipment reliability attainable in the art today.

To facilitate machine checking and preventive maintenance, the AN/FST-2 incorporates a marginal-checking system and an extensive variety of internal check and test circuits. Many of the checks are continuous during normal on-line operation. In addition, a test simulator is provided. The simulator generates complex repetitive target patterns which completely exercise all parts of the AN/FST-2. At the maintenance man's discretion, it can also force the synchronization of the simulated radar trigger, azimuth, and telephone-pulse repetition rates.

Two displays are provided as part of the AN/FST-2 to permit on-line monitoring and to aid in maintenance. A PPI monitor can display various quantized video and detected target signals near the front end of the machine. A digital B Scope permits detailed examination of the range and azimuth coordinates of each target that is transmitted to the direction center. A random-access PPI (RAPPI) and associated word printer are also located at the radar site for output monitoring and recording. It is recognized that no matter how careful the design, good machine performance requires good maintenance, and considerable thought was, and is, being devoted to improving the maintainability and maintenance procedures for the AN/FST-2.

Conclusion

The AN/FST-2 Coordinate Data Transmitting Set is a vital link in the SAGE system for the air defense of the continental United States. Its function of "filtering" raw radar data and providing accurate target information to the SAGE Direction Center represents an important application of digital data processing equipment to the solution of real-time problems. Many units of the AN/FST-2 equipment have been produced and installed in the field. Their performance to date has been up to expectations.

Acknowledgment

The FGD and SAH/F functions for SAGE were conceived and initially developed at Lincoln Laboratory, Massachusetts Institute of Technology. Particular credit is due Group 24 of Lincoln Laboratory which was instrumental in the early program and provided the essential transfer of know-how which resulted in the production equipment.

The authors are also indebted to the many people of the Burroughs Corporation who made significant contributions to the successful development of the AN/FST-2.

Operation of the SAGE Duplex Computers*

P. R. VANCE†, L. G. DOOLEY‡, and C. E. DISS§

Introduction

Large scale digital computers perform the routine control and data-processing functions of SAGE direction centers. For practical air defense, the SAGE direction centers must operate around the clock, a goal not yet achieved by present-day computing equipment. In order to achieve 24-hour-per-day operation, two identical computers are provided at each center. One computer (the active computer) operates the direction-center program. The other computer (the stand-by computer) is available for preventive maintenance and a limited amount of routine data processing. The functions of the two machines are interchanged by switching the direction-center inputs and outputs from one machine to the other. Thus, if the active computer fails, the task of operating the direction-center program can be transferred to the stand-by

* The research reported in this paper was supported jointly by the U. S. Army, Navy, and Air Force under contract with Mass. Inst. Tech. and the IBM Corp., Kingston, N.Y.
† M.I.T. Lincoln Lab., Lexington, Mass.
‡ System Development Corp., Santa Monica, Calif.
§ Military Products Div., IBM Corp., Kingston, N.Y.

machine, and as far as the direction center is concerned, computer "down time" is reduced to those periods when both machines are simultaneously inoperative.

Unique to the duplex installation are the equipment design features that accomplish switching direction-center inputs and outputs, and those that provide communication between the two computers. The duplex switch (which is manually operated) transfers external input and output communication lines, inputs from direction center operators, and control of the display equipment from one computer to the other. The switching facilities are shown in Fig. 1. Each computer can read or write upon its own intercommunication drum, and can read from the other computer's intercommunication drum. These drums permit the transfer of large amounts of data from one computer to the other. The intercommunication lines terminate in sense units (flip-flops) whose conditions (0 or 1) are controlled by one computer and sensed by the other. Thus the intercommunication lines enable the program operating in one computer to determine that a given event has occurred in the other. These lines are primarily used to synchronize the programs operating in the individual computers. The alarm lines, like the intercommunication lines, terminate in sense units. Thus an overflow, memory parity, or drum-parity alarm occurring in one computer can be detected by the program operating in the other. The alarm signals from one computer can also be utilized to interrupt the operation of the other. The settings of certain switches, located on the operating console, determine whether an alarm signal originating in one computer will be ignored by the other, or will cause the other to stop or branch to test memory. The intercommunication facilities are shown in Fig. 2.

The primary task assigned to the active computer is the operation of the direction-center program, and that assigned to the stand-by computer is the operation of maintenance programs. These are the so-called simplex functions of the active and standby machines. In addition, each machine performs certain functions unique to a duplex installation. These functions are directed toward insuring minimum interruption of direction-center operation, should an interchange of computers (switchover) be required. The remainder of this paper is devoted to a discussion of the duplex functions performed by each computer, and to a discussion of the switchover process itself.

Duplex Functions of the Active Computer

Continuity of direction-center operation can be retained after switchover if data representative of the current air situation can be made available to the newly active computer. Part of these data, generated by the previously active computer and stored upon the magnetic-drum fields of that computer, represents a myriad of decisions and manual actions made by the Air Force personnel who operate the direction center. Loss of these data during switchover would necessitate repetition of these decisions and actions, resulting in a transient degradation of direction-center

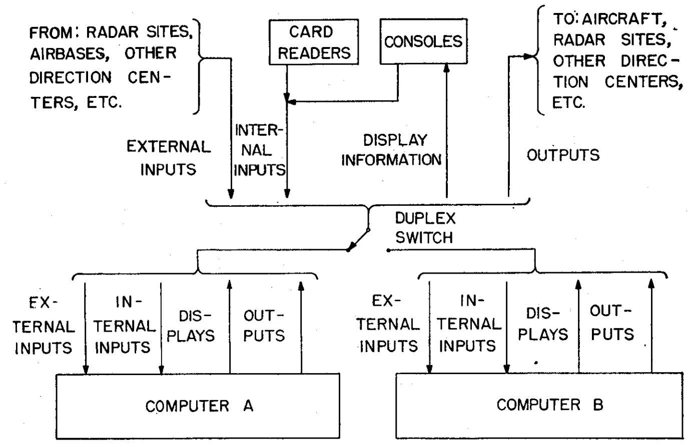


Fig. 1—Switching facilities.

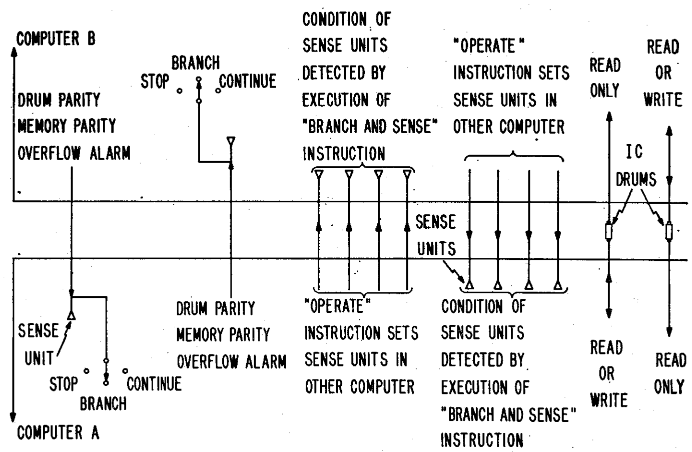


Fig. 2—Intercommunication facilities.

operation. The duration of the transient would depend upon the complexity of the air situation at the time of switchover. This transient period is minimized by maintaining certain key data upon the drum fields of both the active and stand-by computers. Thus, an up-to-date summary of the air situation is available to the direction-center program immediately after switchover.

The summary air-defense data stored in the stand-by computer are periodically assembled by the direction-center program operating in the active computer, and transmitted to storage in the stand-by computer via the intercommunication-drum system. The amount of data transferred is limited by the program operating time available in the active computer, and the drum storage available in the stand-by computer. Operating time is a critical factor because the direction-center program is part of a real-time control system, and any increase in operating time degrades over-all system response. The drum storage available in the stand-by computer is limited by the storage requirements of the stand-by programs.

The nature of the summary air-defense data can best be discussed in terms of the types of data tables used and generated by the direction-center program. There are four broad categories of data tables: input, output, display, and central.

Input tables contain data awaiting processing by the direction-center program. These data are generated at external sources (*e.g.*, radar data), and by the Air Force personnel within the direction center. Output tables and display tables contain data awaiting sequential transmission to locations outside the direction center (*e.g.*, air bases), or display to direction-center operators. No portions of the input, output, or display tables are transmitted to the standby computer for storage as summary air-defense data. The net result is loss of direction-center operation during the switchover period. If this period is short, the effect is not serious, because the input data are accumulated again after switchover, and the display and output tables are regenerated by the direction-center program itself. The central tables, on the other hand, are the heart of the air-defense program. In a broad sense they represent a mathematical model of the air situation on which the operation of the direction center is based. It is the central tables, or more specifically, the key portions of the central tables that are transferred to the stand-by computer as summary data.

The only other duplex function of the active computer is that of monitoring the intercommunication lines to determine if a scheduled switchover is to take place. The program operating time required to accomplish this monitoring function is negligible. The switchover process itself will be discussed later.

Duplex Functions of the Standby Computer

The stand-by computer must operate maintenance programs, and at the same time be readily available for operation of the direction-center program. Certain duplex functions, then, must be performed by the stand-by computer:

1) Monitor active-computer alarms,
2) Maintain the direction-center program on the stand-by computer drum fields,
3) Transfer and store summary air-defense data assembled by the active computer,
4) Monitor operator-inserted switch requests controlling standby operations,
5) Prepare digital displays indicative of the status of standby computer operation.

Only the first of these functions (alarm monitoring) is an equipment function; the others are programming functions.

Memory parity, drum parity, or arithmetic overflow alarms that occur in the active computer cause an automatic branch of program control to test memory in the stand-by computer. The sequence of instructions in test memory initiate preparation of the standby computer for switchover. Preparation for switchover includes erasure of all maintenance programs and tables from core and drum storage, restoration of the direction-center program upon the stand-by drum fields (if necessary), and a final transfer of the summary data from the active machine. Having completed preparations for switchover, the stand-by computer simply waits for switchover to occur, or for a manual intervention to restore normal stand-by status.

Maintenance of the direction-center program on the standby computer's drum fields permits rapid recovery of direction-center operation after switchover. The fact that the direction-center program is properly stored is verified by reading each program drum field into core memory, computing the sum of the binary numbers stored on the drum field, and comparing the result with the correct sum (also stored on the drum field). If the computed sum is incorrect, the offending drum field is reloaded from magnetic tape. The process of checking and loading the program drum fields occurs automatically whenever a maintenance program has destroyed the contents of a program field, or whenever preparation for switchover is initiated by an active computer alarm. It can also be requested by a manual switch action.

The duplex functions of transferring summary air-defense data, monitoring operator switch actions, and the preparation of digital displays are executed periodically. The frequency with which these functions are performed depends upon the mode of operation of the stand-by computer. One of three modes may be selected. Each provides a different frequency of execution of the periodic duplex functions, in the range of once every few seconds to once every few minutes. This requirement for interleaving simplex and duplex operations imposes stringent requirements for manual and automatic control of the maintenance programs. Control of the sequence of operation of maintenance programs, and selection of the mode of stand-by operation is accomplished by manually-inserted operator-switch actions. The running time of each maintenance program is known to the stand-by control program, and either manual or automatic selection of long running maintenance programs is automatically prevented if the selected mode of stand-by operation requires frequent execution of the periodic duplex functions. In order to relieve this running-time restriction on the selection of maintenance programs, the programs are designed as a collection of program units to permit operation of long running maintenance programs by operating them one program unit at a time.

Switchover

Switchover requires transfer of direction-center inputs and outputs from the active to the stand-by computer, and activation of the direction-center program in the stand-by computer. Preparation of the stand-by machine for switchover is initiated automatically by an active computer alarm, or manually by an operator switch action. After the stand-by computer has completed its preparations for switchover, and after the duplex switch has been operated, control of the stand-by computer is transferred from the stand-by control program to the startover program. The startover program performs the function of activating the direction-center program in the standby computer, and thereby completes the transition of that machine from standby to active operation.

Two modes of switchover have been provided. The emergency switchover mode is used when switchover occurs after the active computer has become inoperative. The scheduled switchover mode is used when both machines are in operating condition at the time of switchover. The major difference between these two modes is in the amount of air-defense data that is made available to the stand-by computer. Normally, only summary air-defense data are available upon the drum fields of the stand-by computer. As was mentioned before, these data are transferred to the stand-by computer periodically, and the amount of data that can be transferred is limited by the computing time and storage-space restrictions imposed upon a periodic operation. In short, program operating time is not available to transfer a voluminous amount of data during each cycle of the direction-center program. If, however, switchover is scheduled when both computers are operating, more complete data can be transferred as a "one-shot" process during switchover. This wholesale transfer of data is accomplished by interrupting operation of the direction-center program just prior to switchover, and transferring the contents of the central tables and display tables from the drum fields of the active computer to the corresponding drum fields of the stand-by computer. Proper timing of successive drum transfers is achieved by signals transmitted between computers via the intercommunication lines.

The following conditions describe the status of the stand-by computer at the time that control is transferred to the startover program.

1) The direction-center program is properly stored upon the drum fields of the stand-by computer.
2) The summary air-defense data are stored upon one stand-by drum field.
3) All traces of stand-by program operation have been erased from core and drum storage.
4) In the case of emergency switchover, all program tables are cleared.
5) In the case of scheduled switchover, the air-defense information transferred from the active computer is stored on the proper table drum fields.

Completion of the switchover process requires that the startover program process the air-defense data stored in the stand-by computer to make it usable by the direction-center program. The startover program then transfers the control portion of the direction-center program into the core memory of the stand-by computer, and transfers computer control to the direction-center program.

Sorting and extrapolation of air-defense data are performed by the startover program. In the case of emergency switchover, only summary data are available to the stand-by computer. These data, which were gathered from several central tables, occupying different drum fields in the active computer, are packed together upon one drum field in the stand-by computer. The startover program sorts these data and distributes them among the appropriate stand-by table drum fields. In the case of scheduled switchover, the air-defense data are already stored upon the proper standby drum fields, and the sorting process is unnecessary. The extrapolation process performed by the startover program adjusts the air-defense data to compensate for the program operating time lost during switchover. The process is primarily a matter of extrapolating the position of aircraft tracks along their last known velocity vectors.

Future Developments

The duplex problem is that of determining how best to utilize two computers to enhance direction-center reliability. A secondary consideration is that of determining how to make efficient use of the stand-by computer without jeopardizing the primary requirement that it be readily available to perform air defense. When more information has been gathered regarding maintenance requirements, and when maintenance techniques have been perfected, it may be possible to utilize the stand-by computer for a limited amount of data processing, or for simulation of battle conditions during training exercises. Such applications must, of course, be designed within the ground rules established by the primary requirements of adequate stand-by computer maintenance, and availability for rapid switchover.

A Digital System for Position Determination

DAN C. ROSS†

ONE of the most important functions in the air traffic control (ATC) system is that of aircraft position reporting. A large fraction of the equipment and effort required in CAA operations is involved in the initiation and handling of position reports. A pilot must divert attention from the actual control of the aircraft to talk with ground controllers for the purpose of reporting position, or for the purpose of procedural communications associated with position reports. Many of the adjustments and readings of navigation instruments are performed not because the pilot wishes such frequent information for his own uses but because of the necessity of reporting position. A large amount of electromagnetic spectrum is consumed at present in the position reporting function, and the need for spectrum will increase still further as air traffic control is expanded, unless some new reporting technique is developed.

Future systems of air traffic control will utilize automatic data processing machinery to handle many of the routine functions and will permit the human controller to handle with safety more traffic than he can today. The effectiveness of the combination of human controllers and automatic computers will be greatly improved with the advent of an automatic means of providing frequent and accurate position reports on all aircraft in the system. Unfortunately, there is no existing system of position reporting which meets the requirements of air traffic control.

A great deal of effort has gone into the development of several methods of providing aircraft position information to the traffic control system. The three techniques which have received the most attention are radar, beacon transponder, and data link. The principal advantages of radar is that no equipment is required in the aircraft, but this advantage is offset by the lack of identification, the lack of correlated altitude data, and excessive noise and interference of various sorts. The conversion of the raw signal from the radar into a form suitable for air traffic control requires the continuous solution of a difficult correlation problem involving either a great deal of computing capacity or the full attention of many human operators. Both the beacon and the data link overcome the fundamental problems of radar at the expense of adding equipment to the aircraft and of introducing a number of difficult technical problems which are yet to be solved. At least in the case of the data link, it appears that the various technical difficulties will eventually be surmounted, but the expense in terms of airborne equipment costs and total usage of the electromagnetic spectrum may be quite high.

† IBM Corp., Kingston, N.Y.

The purpose of this paper is to present the basic principles of operation of a position reporting technique which satisfies the present and future requirements of air traffic control and overcomes the known technical difficulties in the radar, beacon, and data link systems, and yet promises to accomplish these goals with a minimum of expense. The technique to be discussed is known as Automatic Position Telemetering (APT). The APT system has progressed to date through preliminary design and testing phases which have concentrated on the radio communication portions of the system. The results of this work indicate the feasibility of the proposed system and point the direction for the design of a complete prototype.

The basic requirements for an automatic position reporting system are obtained from a consideration of the present manual techniques, the characteristics of semiautomatic data processing systems, and the operational and technical shortcomings of radar, beacon, and data link. First, the position reporting system must be capable of integration with automatic data processing machines and must eliminate or minimize the manual operations required of pilots and controllers. Second, the total electromagnetic spectrum assigned to the system must be held to a minimum; system planners ought to regard bandwith as one of our most precious national assets. Third, the airborne element of the system must be kept as small and inexpensive as possible because of compounding effects on the weight, reliability, maintainability, and cost of the total airframe system. Fourth, the system design must be based on fundamental logical and physical principles selected to minimize the technical difficulties which have been experienced in recent beacon and data link development programs.

A major design objective is the provision of service to all aircraft in the system on a single radio channel. Accomplishment of this objective completely eliminates tuning operations as far as either the controller or pilot is concerned. The complexity of both the airborne and ground-based portions of the system is greatly reduced if single-channel operation can be realized. A related secondary objective is the minimization of the bandwidth required for the single channel.

If a single channel is to suffice, it is mandatory that some sort of time-division technique be utilized. This leads naturally to completely synchronous operation for allocating use of the channel and to discrete time-slot addresses uniquely assigned to each aircraft. Use of the time-division addressing principle eliminates the need for narrow-beam antennas and complex antenna-guidance equipment.

Another design objective of major importance is that the amount of information transmitted from the aircraft to the ground environment must be minimized. If possible, the transmission ought to be limited to a single impulse for the sake of simplicity. Any information which could just as well be determined at the receiver, even if this amounts to a redetermination of data previously known at the transmitter, ought to be so determined rather than wasting channel capacity in its transmission. If the ideal of position reporting by means of a single impulse can be realized, then many of the technical problems involved in pulse transmission systems are greatly simplified, in particular, the problem of interference from multipath echo phenomena.

Noise phenomena such as atmospherics, receiver noise, and ignition noise will plague any sort of radio communication system. In order to minimize noise effects, the transmitter must produce high pulse power and the receiver must be designed for low-noise performance and located in a relatively quiet environment. The carrier frequency ought to be selected in the 1000-mc region to obtain an optimum balance between atmospheric noise at lower frequencies and receiver noise at higher frequencies. For the position reporting application, the line-of-sight limitation of UHF communication is actually an advantage because several aircraft can be given the same address provided only that the aircraft are separated by several hundred miles.

All of the requirements and design objectives introduced in the foregoing discussion can be realized in the proposed APT system. In addition, the system is capable of handling traffic densities considerably greater than the densities predicted for 1975. There would be no difficulty in providing APT service to several thousand aircraft simultaneously within the area covered by an air route traffic control center.

The principle of operation of the APT system along with the interrelationship between the major equipments involved is shown in Fig. 1. In each major terminal area, four ground-based receiving equipments are arbitrarily located, provided only that the area of maximum traffic density is central with respect to the four receiver sites. Maximum separation of any two receivers in the group would be about 20 miles. Each aircraft is provided with a transmitter which emits an intense pulse of UHF energy every few seconds. Interference between aircraft in the system is prevented by means of time-division multiplexing techniques.

As shown in the plan view of Fig. 1, the pulse of UHF energy leaves the airborne transmitter at instant t_P and travels outward at the speed of light. At some later instant of time t_A the wave front reaches receiver A. The detected pulse at A is used to sample the contents of a free-running clock, recording t_A in digital form. Immediately thereafter, the binary representation of t_A is serially transmitted on a digital data line to a centrally located Coordinate

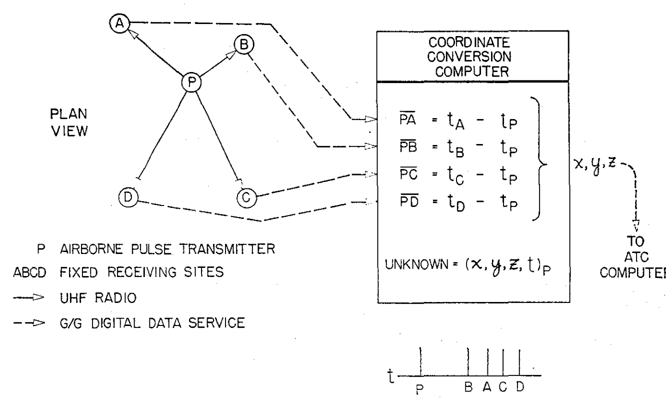


Fig. 1—System diagram.

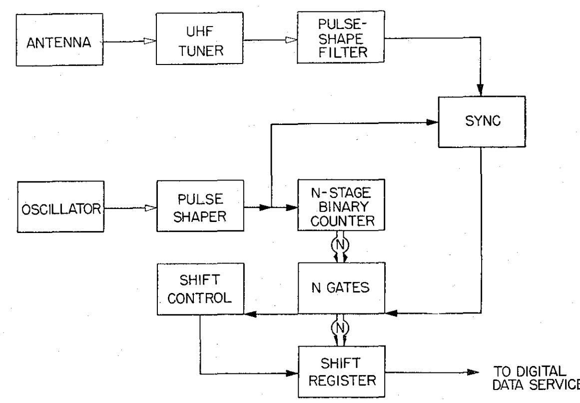


Fig. 2—TOA equipment.

Conversion Computer (CCC). In similar fashion, the times of arrival of the signal at receivers B, C, and D are obtained and transmitted to the CCC. The equipment at each of the four receiving sites will be referred to as Time of Arrival (TOA) equipments.

Each of the four unknowns can be determined in theory from the four TOA measurements. One way of stating the functional relationships between these quantities is shown in Fig. 1. However, there is no interest in t_P and precise values of aircraft altitude cannot be obtained from the TOA data. Thus a practical design for the CCC would provide for the calculation of x and y only, and the altitude z must be found by another method. The output data from the CCC is transmitted in serial form over a digital data line to locally or remotely located ATC data processing equipment.

A block diagram of the major sections of each TOA equipment is shown in Fig. 2. The oscillator, shaper, and counter constitute a high-speed digital clock capable of resolving intervals of a fraction of a microsecond. The precision of this counter is directly related to the precision of the final position measurement (1 μsec = 0.186 mile). The received pulse is used to sample the high-speed counter and transfer its contents to a shift register. The received pulse also initiates a series of shift pulses to transmit the

TOA information to the CCC. The number of stages, N, in the high-speed counter must be sufficiently large to insure that no more than one end carry occurs during the passage of a wave front across the net of four TOA stations.

To make the UHF transmission highly reliable in spite of various noise phenomena, it may be necessary to send doublets or triplets rather than single pulses. The purpose of the "pulse-shape filter" shown in Fig. 2 is to produce an output pulse when and only when the input signal is within appropriate tolerance limits of the coded waveform transmitted by the airborne unit.

The airborne pulse transmitter for the APT system is shown in Fig. 3. The transmitter proper consists of a high-power modulator which excites a cavity-tuned power oscillator. The approach which appears most practical employs a hydrogen thratron discharging a delay line to modulate a "lighthouse" triode. Pulse power levels of several kilowatts are desired. Satisfactory results were obtained during the tests with a pulse power of 1.5 kw and there appears to be no difficulty in designing economical modulators and oscillators that will produce up to 10 kw.

Positive identification is provided and intrasystem interference prevented by means of a synchronized time-slot counter in each aircraft. The counter in any given aircraft is set to the aircraft address each time that the "framing pulse" is received from the ground. The framing pulse is sent to all aircraft simultaneously at the rate of about once a minute. The address counter in each aircraft counts down toward zero under the control of a medium-precision oscillator in the airborne equipment. When the address counter reaches zero and produces an end-carry pulse, the modulator is triggered. Since each aircraft would be given a different address, the UHF impulses all occur at different times.

The operation of the proposed address timing technique can be clarified by considering a typical example. In Fig. 3 and Fig. 4, the following numerical parameters are assumed: framing-pulse period = 1 minute, number of addresses = 2^{12} = 4096. It is further assumed that the aircraft address is established by means of four octal switches. The aircraft chosen for the purposes of this example has address 13 (octal). The framing pulse transfers the number 13 (octal) from the address switches to the counter, and once each 15^{-} msec the counter contents are reduced by one. As the address counter changes from 0000 to 7777, the end-carry pulse is produced and the UHF transmission occurs.

The framing pulse is transmitted from ground to air by multiplexing it on the VHF and UHF voice channels to economize on both equipment and bandwidth. Since the addressing technique proposed does not require precise timing, the framing pulse can be handled on an audio channel. Assuming that the voice signal could be cut off at about 4 kc without serious loss of fidelity, it seems practical to use a subcarrier of about 6 to 8 kc for the framing pulse. In order to reject extraneous noise pulses, it is desirable to employ two or three short bursts of the subcarrier for each framing signal. The purpose of the pulse-shape filter shown in Fig. 3 is to decode the waveform used for the framing signal in order to distinguish it from various interfering signals. The framing-pulse transmitters are synchronized on a national basis via transmissions over lf channels from a central timing standard. Frequency-control servomechanisms are provided to keep the framing pulses together during outages or noisy periods on the LF channel.

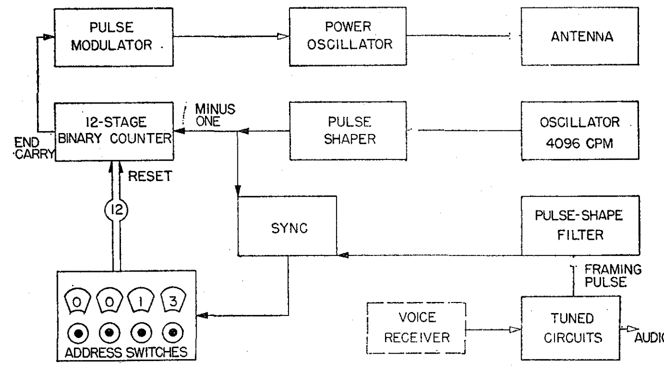


Fig. 3—Airborne equipment.

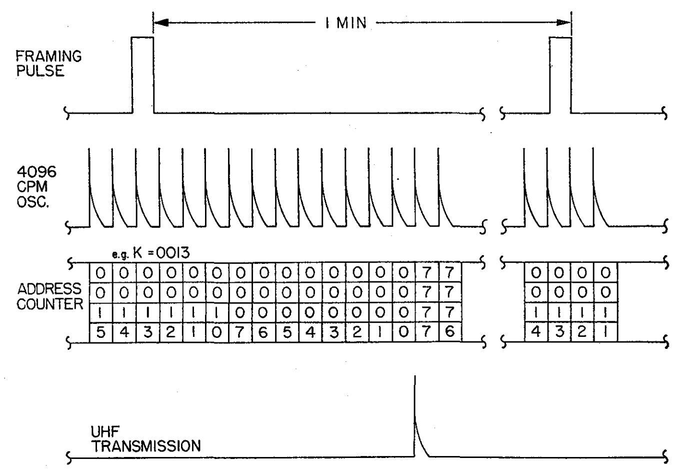


Fig. 4—Timing.

It should be noted that the APT system provides both position and identification data to the ground environment without actually transmitting either one of these quantities. Both the position and the identification are determined at the final receiver on the ground the position by a precision time-difference technique and the identification by medium-precision time division. To add automatic reporting of altitude, the altitude signal is multiplexed on the air-to-ground transmission. Altitude data can be readily included in the APT system by providing a second UHF pulse such that the spacing between the two pulses represents the altitude code. One way to produce the second pulse is to use the end-carry pulse from the address counter to drive a delay multivibrator which produces two pulses to trigger the modulator. The delay between the two pulses would be controlled by an electromechanical connection from a sealed aneroid altimeter in the aircraft. In the TOA equipment, the first received pulse would be used to start an altitude counter and the second pulse would transfer the contents of this counter to an extension of the shift register shown in Fig. 2. The second pulse would also be used to reset the altitude counter. Thus, the altitude information turns out to be the only data actually transmitted from the aircraft to the ground in the usual sense of the term "transmission," while the position and identification are both ground-derived.

The high-speed counter used in the TOA equipment is not synchronized with any of the other counters in the TOA net. The effect of synchronization is accomplished much less expensively by the provision of a set of equipment identical to that used in the aircraft but located on the ground near the center of the net. TOA signals arriving at the CCC during the time slot assigned to the ground-based pulse transmitter are then compared with the values which would have been obtained if the four counters had actually been synchronized. The differences so obtained are then held in temporary storage in the CCC and are used to correct the TOA values received during each of the other time slots in the frame. The additional calculations required in the CCC amount to a few subtractions and represent no important increase in the amount of electronic equipment required.

TABLE I

A Possible Set of Parameters

Airborne Pulse Transmitter			
Reporting rate:	Enroute phase	1	msg/min
	Terminal phase	8	msg/min
Number of addresses:	Enroute phase	2048	a/c
	Terminal phase	256	a/c
Number of time slots:	Enroute phase	1×2048=2048	slots
	Terminal phase	8× 256=2048	slots
	Total	4096	slots
Time-slot duration		60/4096= 14.65	msec
Carrier frequency		1400	mc
Pulse power		10	kw
Pulse duration		0.2	μsec
Timing for altitude pulse:	Minimum delay	1000	μsec
	Maximum delay	2600	μsec
TOA Equipment			
Receiver bandwidth		5	mc
Antenna height		100	feet
Clock-oscillator frequency		5	mc
Number of stages in clock		9	bits
Clock cycle duration		$2^9 \times 0.2=$ 102.4	μsec
Maximum span of TOA net		102.4×0.186= 23.1	miles
Message content:	Time of arrival	9	bits
	Altitude	6	bits
	Parity check	1	bit
	Spares	3	bits
	Total	19	bits
Message rate		4096	msg/min
Shift rate		19×4096/60=1297	bit/sec
Altitude-oscillator frequency		0.04	mc
Number of altitudes		1600×0.04= 64	levels
APT System			
Minimum altitude coverage:	Terminal phase	0	feet
	Enroute phase	4000	feet
Maximum range		90	miles
Precision:	Inside TOA net	0.05	mile
	At maximum range		
	azimuthal	0.5	mile
	radial	1.5	mile
	At 50-mile range		
	azimuthal	0.3	mile
	radial	0.5	mile

The principal computing problem which the CCC must solve is the conversion of the hyperbolic coordinates represented by the TOA values to a more convenient set of coordinates for use by the controllers and data processing machines involved in ATC operations. The output coordinates from the CCC may be chosen to be rectangular or geodetic with very little difference as far as the cost of the CCC is concerned. Several possible designs of the CCC have been considered; one based on table look up and interpolation, a second method based on an iterative technique, and a third and most promising method based on direct calculation. The time provided for each computation cycle is sufficiently long that the CCC requirement can be met with a simple design based on a serial arithmetic unit controlled by a fixed program. The special geometry associated with each TOA net can be accommodated by well-known storage techniques. Since no human intervention is required in the normal operation of the CCC, the input-output requirements are easily satisfied.

The numerical values of the APT parameters may be chosen from a fairly wide range of values; in many cases the range of practical choice extends over several orders of magnitude. The numerical values of the parameters used in Table I are presented in the interest of clarity and do not necessarily represent recommended values. Some of the more flexible parameters are: the number of time slots, the time-slot duration, and the reporting rate. These parameters are limited only by the requirement that the interval between reports from the same aircraft must equal the product of the number and duration of the time slots. This requirement is modified if a higher reporting rate is desired in the terminal phase of aircraft flight than in the enroute phase. For example, the 4096 time slots discussed earlier could be divided into two groups of 2048 slots each. One of these groups could be used to handle 2048 aircraft in the enroute phase of flight, and the remainder could be divided into 256 groups of 8 slots each to accommodate an additional 256 aircraft in the terminal phase with a reporting rate 8 times that used in the enroute phase. By properly choosing the numerical values involved, the pilot's attention required in the setting of the APT address can be limited to a single setting of the address switches just before take-off, plus the operation of a two-position switch at the beginning of the enroute phase of flight and once again at the beginning of the terminal phase.

The precision of the APT system in the near zone depends only on the TOA clock resolution. At maximum range, the precision depends on the ratio of the TOA station spacing to the clock resolution. For a value of clock resolution of 0.2 μsec and a TOA station spacing of 20 miles, the precision will vary from about 0.05 mile inside the net to about 1.5 miles at a range of 90 miles. At long range, the azimuthal precision is considerably better than the radial precision.

The automation of the position reporting function will eliminate a large fraction of the present communications load. Greater precision of position data will reduce the frequency of conflicts with a corresponding reduction in the number of transmissions required to each aircraft. A standard clearance signal requiring no human intervention can be employed except in the small number of cases requiring special transmissions. It therefore appears doubtful that an automatic ground-to-air data service can be justified for more than a small fraction of the aircraft in the future ATC system. These reasons explain the emphasis of this paper on the air-to-ground reporting service and the neglect of the reverse direction of transmission.

The proposed APT system meets the basic requirements of position reporting for air traffic control. In the interest of economy, the special needs of the more advanced aircraft ought to be met by providing additional equipment to supplement that used for the basic functions. All aircraft in the system need not carry high-precision navigation equipment plus automatic two-way digital communication merely because a minority of the aircraft requires these devices. One of the advantages of APT is its compatibility with all sorts of navigation techniques including contact flying, VOR and other air-derived navigation systems, deadreckoners corrected manually or automatically from ground-derived data, and advanced inertial systems.

There are many design problems in the APT system which remain to be attacked, so it is too early to predict success. However, the investigations and tests to date indicate that the system is feasible and has several strong points. Some of the more important advantages are: simplicity of airborne equipment, flexibility of parameter choice, minimization of pilot attention, and perhaps most important of all is the independence of position-measuring accuracy with respect to malfunctions or misadjustments in the airborne unit.

Discussion

Mr. Bhippel (U. S. Signal Corps): Wouldn't three ground stations be sufficient for position determination?

Mr. Ross: Assuming that accuracy of the order given in the paper is required, then one must acknowledge that the final results depend on variation in three dimensions and, therefore, three independent time differences are required. The number of independent time differences is one less than the number of TOA receivers.

D. C. Friedman (National Bureau of Standards, Washington, D.C.): What provision would be made for control when there is a plane transmitter outage, possibly unknown to the pilot? Or a plane with no transmitter?

Mr. Ross: In any future air traffic control system, two-way radio would continue to be employed; thus any outage of the APT transmitter could be overridden by reverting to voice transmission of estimated times of arrival over various fixes as determined by airborne navigation equipment. Aircraft not outfitted with APT would, of course, be required to file position reports by voice radio at all times while flying under instrument conditions.

Mr. Friedman: How many computers would be required for current airways? How many ground stations? What would be the cost for the ground and aircraft installations?

Mr. Ross: One coordinate conversion computer and four TOA receivers are required for each major airport. Throughout most of the United States, the enroute area would be adequately covered by the installations at the terminals. In the areas where the terminal installations do not provide sufficient enroute coverage, one has the alternative of installing supplementary APT nets or requiring the use of voice radio for position reports while flying through these areas. It is too early to state cost estimates for either ground or the aircraft installations.

T. Kampe (Librascope, Glendale, Calif.): How many ground stations are envisioned across the country?

Mr. Ross: The number of APT installations depends entirely on the number of terminal areas requiring automation of the position-reporting function. The determination of the traffic level needed to justify such service would have to be made by the Civil Aeronautics Administration in the case of civil airports and the cognizant military service in the case of military airports. Further technical development and product engineering work should be carried out before these questions of system economics can be answered intelligently.

Mr. Kampe: How are aircraft with malfunctioning sets to be detected, and how handled?

Mr. Ross: It is important to note that malfunction of the APT transmitter does not produce erroneous position measurements. However, there are two important malfunctions to consider. First, a complete absence of the UHF pulse would be detected at the ATC data processing center in the form of a missing position report in some particular time slot. If this situation continued, the pilot of the offending aircraft would be notified by voice radio to revert to voice position reporting over certain fixes. An outage of the framing pulse would result in a very slow drift in the aircraft identification number—a difficulty which is easily resolved either by automatic or manual "identification tracking" at the air traffic control center.

Mr. Kampe: How are transmissions between different sets of ground stations, for a specific aircraft, to be integrated into a coordinated picture of aircraft?

Mr. Ross: The output of the coordinate conversion computer associated with each APT ground installation would be transmitted automatically over ground-to-ground communication facilities to the terminal air traffic control facility and also to the air route traffic control center covering enroute operations in the area. Thus, each air traffic control facility receives position data on all aircraft within its area of responsibility. Digital data transmission techniques presently available are entirely satisfactory for the APT application. The necessary data processing and display equipment at the air traffic control centers would be designed to include data from APT along with data from other sources.

Real-Time Data Processing for CAA Air-Traffic Control

G. E. FENIMORE†

FOUR years ago, the Eastern Joint Computer Conference was held in this same city. At the first session of that Conference, Vernon Weihe, representing the Air Transport Association of America, stated that "the need for automatic computation and automatic data handling (in air-traffic control) is immediate and urgent." He challenged the computer industry and the aviation industry to meet this need with sound system design incorporating human engineering and the rapidly advancing technical developments of the day. The paper took note of the fact that a start had already been made with the installation of a magnetic drum-message storage and processing system at the CAA Technical Development Center in Indianapolis, Ind. This present paper is somewhat in the nature of a status report, describing how the Civil Aeronautics Administration is beginning to use electronic computers for air-traffic control operations. In order to understand this application, it will be necessary to consider briefly the manual operations which are to be replaced.

Air-traffic control is exercised in two types of areas. The first type, called the terminal area, is that airspace in the proximity of an airport where aircraft are under the jurisdiction of an approach controller or tower controller, located at the airport itself. The second type, which is called the enroute area, is that airspace designated as Federal Airways, which are the well-traveled highways of the sky. In the enroute area, control is exercised from an air-route traffic-control center, of which there are 27 within the continental limits of the United States. A typical center has jurisdiction over an area approximately 300 miles across. Plans are well along to expand enroute control area to include all airspace above a certain designated altitude, such as 24,000 feet. This paper will concern itself mainly with the operations of the enroute area.

Fig. 1 shows an air-route traffic-control center. The individual controllers are responsible for a portion of the area called a sector. Each sector has a tabular display in the form of a board in which are inserted flight-progress strips. A close-up view of one of these boards is shown in Fig. 2. Within the geographical area of the sector, there are several key traffic-control points generally located at the intersection of airways which are called fixes. Aircraft are required to report to the ground by radio whenever they pass over one of these fixes, in order that the controller may ascertain their position and maintain proper separation from other aircraft both in altitude and in time.

† Chief, Air Traffic Control Equipment Branch, CAA Tech Dev. Center, Indianapolis, Ind.

Fig. 1—Indianapolis air-route traffic-control center.

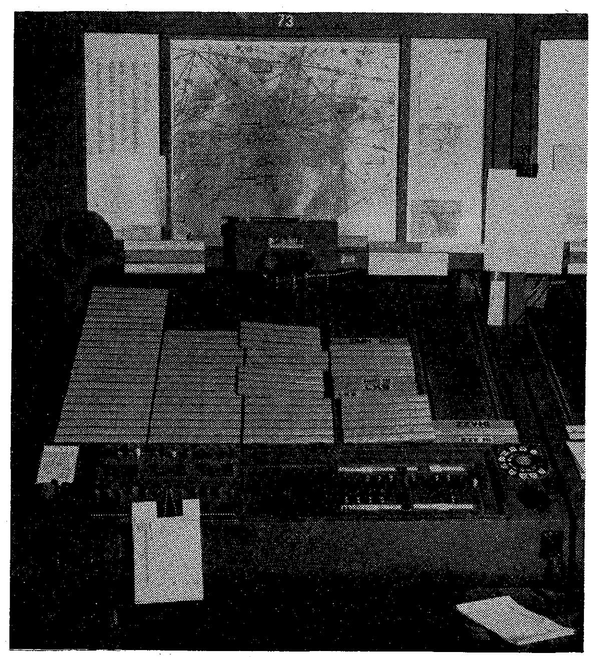


Fig. 2—Flight-progress board.

Aircraft pilots who intend to make a flight under instrument conditions or under the supervision of CAA air-traffic control must file a flight plan with the control agency. This flight plan will include the aircraft identification, aircraft type, speed, take-off point, altitude, route, and destination. From the information contained in a flight plan, individual flight-progress strips are prepared and posted at each fix over which the aircraft will pass. Fig. 3 shows a typical flight-progress strip prepared by a traffic controller for American Airlines Flight No. 34. The strip shows the following data: the aircraft is a DC-7 with a speed of 370 knots, its route of flight is from Chicago Midway (MDW) over airways designated as V 6, V 168, and R 15, to Idlewild (IDL). The flight was first estimated over Youngstown, Ohio, (Y) at 0958 which was later re-

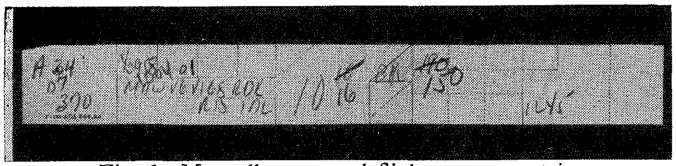


Fig. 3—Manually prepared flight-progress strip.

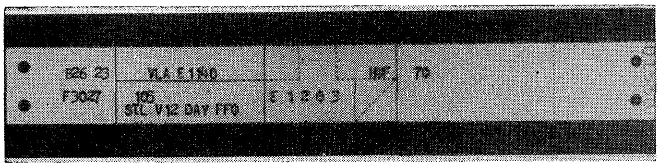


Fig. 4—Automatically prepared flight-progress strip.

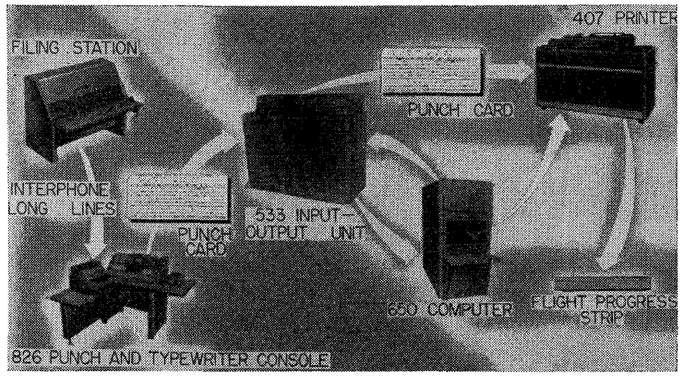


Fig. 5—IBM 650 air-traffic control operation.

vised to 1004 and actually reported at 1001. It was originally estimated to pass over Brookville, Pa., (BKL) at 1010 and this estimate was revised at 1016 on the basis of the Youngstown revision. The altitude was originally 19,000 feet and this was later changed to 15,000 feet.

During a busy hour in the New York center, as many as 1200 to 1500 flight strips will be prepared. The system has a deadline to meet in that the strips must be prepared and distributed to the proper controller approximately 30 minutes in advance of the actual arrival of the aircraft over the fix, in order that he may compare the time and altitude with that of other flights in the area and make sure that no conflicts exist. On peak days and at peak hours of traffic, the time involved in getting flight plans to the center, processing strips, and in getting clearances back to the pilot sometimes causes the delays in take-off that all of us have experienced. A great deal of manual effort is involved in gathering the data, preparing and distributing the flight-progress strips. It is in this area that the first application of computers is being made.

In 1955, under an Air Navigation Development Board project, system experimentation at the Technical Development Center began to determine the data-processing requirements for automatic printing of flight-progress strips. Although the magnetic drum-storage equipment lacked computing capabilities, a group of specially trained operators were used to simulate the computing functions. These operators would receive a flight plan and process it by breaking down into the various fixes over which the aircraft would report. They would then prepare messages in the form of fix postings and send these messages over a teletypewriter circuit to the magnetic drum equipment. The magnetic drum equipment would subsequently read out each message to one of several printers when it came time for the flight strip to be displayed in front of the air-traffic controller.

Fig. 4 shows a flight-progress strip printed during the evaluation of this system.

During the past year, an IBM 650 computer has been installed in the Indianapolis Air Route Traffic Control Center in order to carry out an operational test of printed flight-progress strips. See Fig. 5. Traffic controllers, receiving flight plans by interphone, have been preparing a punched card for each flight. These cards are fed into the computer, which determines the route of flight, calculates estimates, and prints all the strips required automatically. A complete report of this operation has been prepared and will soon be published by the CAA as a Technical Development Report. In addition, this subject was also presented in a paper given by G. B. Harwell of the Technical Development Center during an IBM 650 Scientific Symposium in Endicott, N.Y., the first week of October, 1957. Plans are now being made to expand the capability of this installation by adding RAMAC and on-line input-output facilities.

Plans are also being made to install the Model I Univac File Computer in the traffic-control centers of New York and Washington, D.C., by next summer for automatic preparation of flight-progress strips. A prototype of this system is now being assembled at the Technical Development Center in Indianapolis for testing and evaluation. Fig. 6 is a block diagram of the prototype system. In the Washington and New York installations, space limitations will require that the computer be located in a distant room or even on another floor of the building which houses the center-operation area. Provisions must be made, however, for input and output at several locations in the center itself. In the initial phases of the operation, the principal input to the system will be in the form of flight plans. Flight-plan data will be fed to the computer in two ways. Those which are received by interphone or off-line teleprinter will be encoded by operators in the center itself and transmitted to the computer. Those received from another computer-equipped center or remote station in the local area properly equipped for on-line communication will be fed directly into the computer.

One feature of the system will be the adaptation of a scanner and speed-conversion device which Remington Rand is producing for their airline reservations systems. By means of this equipment, a number of communication channels, operating at teletypewriter speeds, can work into a single high-speed input to the computer.

At the manual input position, flight plans which are received by interphone will be encoded by operators and transmitted to the computer. The computer will analyze

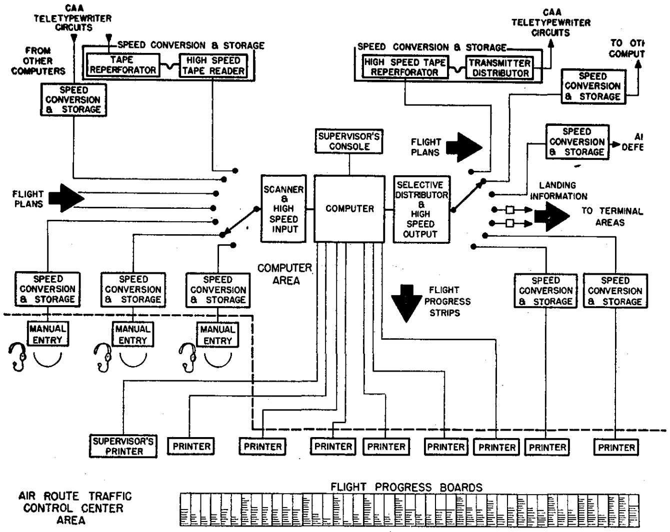


Fig. 6—Block diagram of prototype data-processing system.

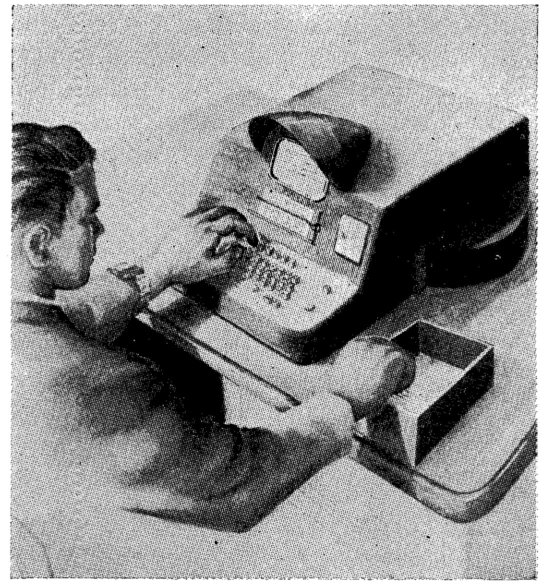


Fig. 7—Flight data-entry equipment.

the route of flight, determine over which fixes the aircraft will pass, estimate the times of arrival over each fix, assemble data in the proper form for printing flight-progress strips, and transmit the data to one or more of the flight-progress strip printers in the center area. At the same time it will keep a record of the flight plan itself and all fix data within its memory. Flight plans which enter the system from distant points over the communications lines will go directly into the computer for processing.

If a flight is to continue into an area served by another air-traffic-control center, flight-plan data must be forwarded to that center in order that strips may be prepared. At the appropriate time, the computer will locate the flight-plan information in its memory and transmit automatically over the communications circuit to the proper destination. A configuration of selective distributor and speed-conversion units will permit one high-speed output from the computer to serve a number of low-speed communications channels. During the evaluation of the prototype system, the procedures of transmitting flight plans to Air Defense and messages to terminal areas regarding landing aircraft will be investigated.

A supervisory console will be provided in the computer area for maintenance operations, and an additional printer will be located in the center area near the manual input position for flight plans which the computer may reject as erroneous.

A problem area in the preparation of data for automatic handling by computers is the rigid format required to insure that the computer treats each part of the data received in its proper category. This is particularly true in the case of a relatively long and involved series of data such as flight plans which will have an average length of 110 alpha-numeric characters. Two major types of errors may occur in the preparation of such messages. Operator errors may range from the addition or deletion of one or more characters, to failure to follow the proper ground rules. Communication errors may also occur, and with the five-unit code of standard teletypewriter systems, these may pass through the system undetected.

In order to permit rapid and accurate composition of flight-plan messages for transmission to the computer over communications circuits, the Technical Development Center has contracted with Aeronutronic Systems, Inc., of Glendale, Calif., for the development of FLIDEN (Flight Data Entry) equipment. Fig. 7 is their artist's conception of the device whereby an operator may compose a message which is displayed on a cathode-ray tube and stored electronically on a magnetic drum during composition. If the operator makes an error, he may backspace rapidly or a character at a time to the position where the error occurred. Having corrected the error, he may shift rapidly back to where the message composition was interrupted and continue. A form on the face of the display designates categories of information which should be entered. All fixed characters and functional characters are entered automatically for ease of composition and accuracy of format. The completed message is checked for accuracy by the equipment before transmission.

For detection of errors during transmission, an error check character is automatically inserted at the end of each line of data. This is accomplished by making a longitudinal count of marks at each code level and adding a check bit to make the total count odd. The check bits are combined to produce "nonsense" or check characters which are transmitted with the message. This basic technique is described by Vincent.[1] At the receiving end, a similar count will be made and the resulting check character at the end of each line of text will be compared with the check character transmitted by the FLIDEN equipment. Messages which contain

[1] G. O. Vincent, "Self-checking codes for data transmission," *Automatic Control*, vol. 5, pp. 46-49; December, 1956.

errors detected by this means will be directed to the computer supervisor's console printer for manual handling.

The specifications under which the FLIDEN equipment is being developed require that transmission of flight-plan messages be made over standard teletypewriter code. However, the logical design of its internal circuitry has been planned that transmission at higher speeds using six or seven-level code will be possible with minor modifications in the final equipment.

With the advent of this type of equipment in the field, on-line filing of flight plans directly into the computer system for flights originating in the center-control area will be possible. This will ultimately reduce the quantity of flight plans which are received by interphone in the center area and manually prepared for insertion into the computer.

Reducing the clerical workload of controllers in the CAA air-traffic control system is only the first step. Anyone who is familiar with the background of thinking on air-traffic-control systems and the great volumes of studies, papers, programs, and system configurations produced during the past ten years may well ask, "What about pictorial displays, radar inputs, data links, automatic position reports, etc.?" This introduction of computers into what is presently a completely manual operation should speed the day when these very desirable features will be included. With the formation of the Airways Modernization Board and the rapid progress that is being made in their data-processing and display program, it is anticipated that air-traffic control operations will soon be provided with powerful tools commensurate with the jet-age traffic-control problem.

Design Techniques for Multiple Interconnected On-Line Data Processors

F. J. GAFFNEY† AND S. LEVINE†

INTRODUCTION

THE application of data-processing techniques to large-scale nation wide industries, such as the transportation industry, has resulted in the need for remoting of subscribers and for multiple data processors regionally located. These equipments form elements of a complete integrated data-processing system or network of systems to solve such problems as reservations control on a complete system basis for railroads and airlines. Early applications of digital data-processing equipment to solve the reservation problem were concerned with limited area problems.[1] The improvement and extension of these systems to integrate the entire reservation-space problem has been a logical evolution.

SYSTEM CONSIDERATIONS

In a reservations system, large numbers of inquiries as to availability of space as well as actual transactions involving sales, cancellations, and waitlisting must be processed each minute. Responses to these inquiries from agents all over the country must be in the order of seconds, so that minimum delay is introduced in handling a customer inquiry over the telephone; further, these inquiries may occur at almost any time of day, particularly in systems for nationwide carriers.

System analysis of these traffic, speed, and operational requirements has led to the design of integrated systems for airlines and railroads which include a number of data processors, each of which provides access to agents using specially designed input-output devices, either in the large reservation offices adjacent to the data processor, or in remotely located offices using specially designed communications equipment. The individual data processors are interconnected by means of high-speed data links so that updating of regional processors by a central data processor can be accomplished rapidly. These systems must be capable of continuous operation for at least 22 hours a day so that reliability considerations are important factors in the design of the system.

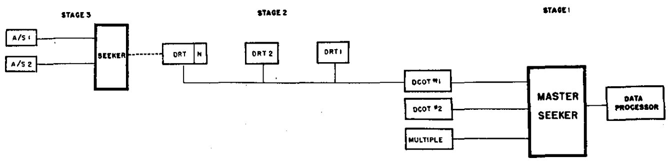
The shutdown of a data processor for maintenance failure for more than a few minutes during peak traffic periods can result in large queues, with resulting loss in business. Requirements for uptime (*i.e.*, the ratio of time the system is actually available to the scheduled time) for systems of this type are in the order of 99.5 per cent.

REMOTING OF INPUT-OUTPUT DEVICES

One of the first extensions of the Reservisor concept was that of providing facilities for the connection of remote agents to the central information store. One method

† Teleregister Corp., Stamford, Conn.

[1] M. L. Haselton and E. L. Schmidt, "Automatic inventory system for air travel reservations," *Elec. Eng.*, vol. 73, pp. 641-646; July, 1954.



T = mean processing time by DCOT.

V = transmission time per message.

C = call rate.

S_1 = mean roll call time of calls delayed.

S_2 = mean roll call time of calls not delayed.

P = proportion of calls delayed.

H = mean service time (DCOT processing plus roll call and transmission times).

E = effective load factor of server (DCOT together with communication links).

(1) $H = T + V + PS_1 + (I - P)S_2$

(2) $E = CH$

(3) Mean Response Time $R = H + D$

(4) $P = E$

(5) Mean Delay Time $D = \frac{1}{2}\,\frac{EH}{1\text{-}E}$

(6) Mean Response Time $R = \frac{1}{2} H \left[\frac{2\text{-}E}{1\text{-}E}\right]$

Fig. 1—Queueing analysis.

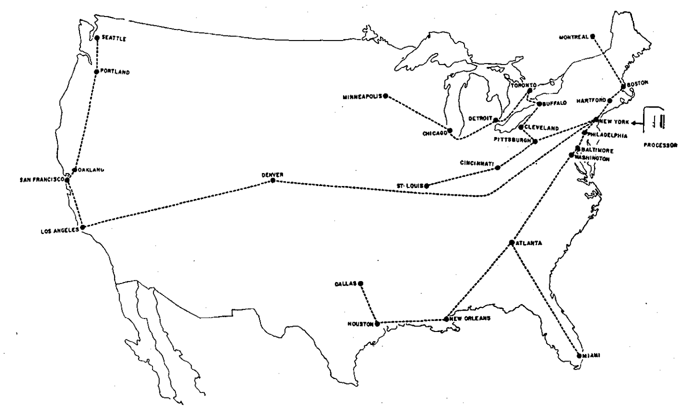


Fig. 2—Pan American Airways reservisor network.

for accomplishing this utilized an editor for ordinary teletype messages arranged in a standard format. Space sales or cancellations can then be routed to the data processor by existing communications facilities such as the 81-D-1 system of A. T. & T. Automatic broadcasting of stop and resume-sale messages can be accomplished by the data processor when indicated by the inventory level of a given flight. This type of system has been installed recently for Braniff Airways and is currently in use.

There exists, however, the requirement for direct connection of agent sets and the data processor to serve locations which generate appreciable traffic in order to minimize time delays between sales of space and their effect on the central inventory.

One obvious solution to this problem is to connect each remotely located group of agent sets to the data processor by means of individual teletype lines. This procedure, while economically applicable to centers of high-traffic generation, is uneconomical of line costs for many remote locations. It is necessary, therefore, to arrange for common utilization of a transmission line by a number of remote locations along its length. This can be done on a time-sharing basis.

In order to minimize transmission delays it is necessary to provide way-station selection and roll-calling equipment which operates in minimum time consistent with the bandwidth of the communication facility utilized. Such equipment has been developed in a joint Teleregister-Western Union development program.

Fig. 1 shows a typical arrangement of remote locations and central processor. The central processor is sequentially connected, by means of a master seeker, to a number of remote line terminations called Distant Central Office Transceivers, and to multiples which serve local groups of keysets. Each of these inputs is served in turn and remains connected to the processor until a reply has been generated and transmitted .

Each Distant Central Office Transceiver (DCOT) is the termination of a teletype line along which are located remote stations, each equipped with a Distant Remote Transceiver (DRT). Each DRT, in turn, connects sequentially through a seeker to a number of agent sets at its location.

In the sequence of operations, a remote agent set bids for access to its DRT. The DRT then bids, by means of a roll-call procedure to be described presently, for access to its DCOT. The DCOT, in turn, bids for access to the data processor. When this chain of events has been completed, the agent set sends its message over the line and receives its reply from the processor. It is then automatically disconnected from the line and another transaction from a different agent set proceeds in the same way. Fig. 1 also shows representative equations for queueing time for a DRT under such a set of conditions. Eq. (6) shows mean response time at a DRT for an idealized system. Actual response time for any given fraction of the calls for consant holding time at the processor is obtained from charts. It is possible to predict, for example, the fraction of calls which will be delayed one holding time or five holding times. In actual practice with current equipment utilizing a 75-word-per-minute teletype line, the mean response time is of the order of several seconds.

The roll-call operation which connects the DRT's successively to the DCOT is of the Round Robin type, that is, each DRT not having traffic when queried generates the call letter of the following DRT. Single letter addresses are utilized for economy of line time. Parity check of all messages is provided.

A remoting system utilizing these techniques was installed early this year for Pan American Airways and is shown in Fig. 2. The central data storage is located in Long Island City, N.Y. The communications system serves 26 cities arranged along four transmission lines.

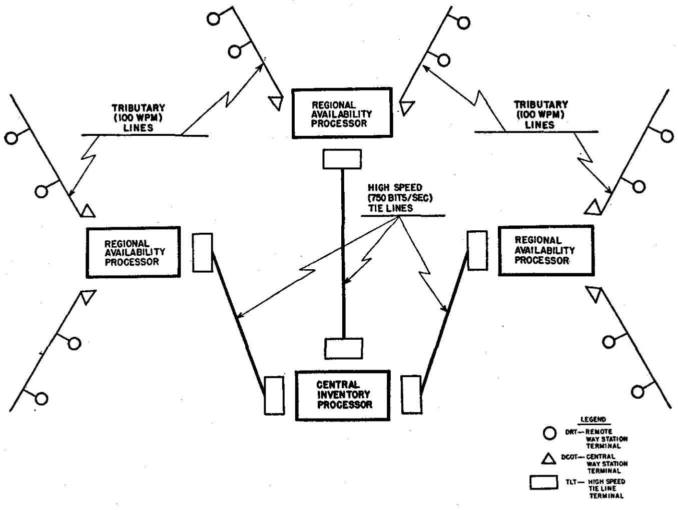


Fig. 3—Integrated airline reservation system.

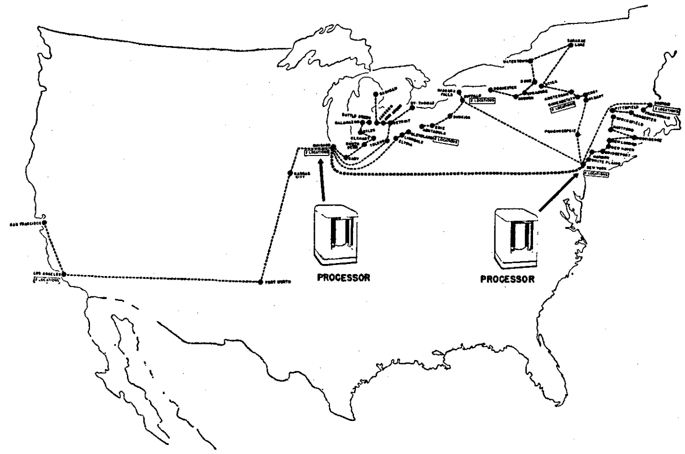


Fig. 4—Railroad Reservisor transmission network.

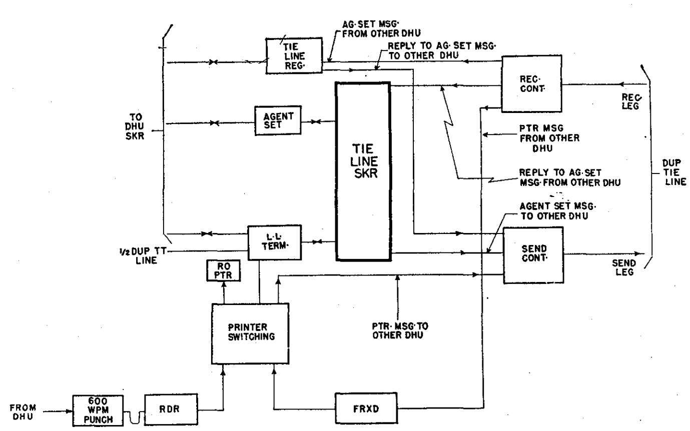


Fig. 5—Railroad Reservisor tie-line terminal.

Integrated Airlines Reservation System

Where traffic volumes in regional areas are high, greater economy can be served by the use of regional data processors connected together to form an integrated system. Such a system is shown in Fig. 3. The regional availability processors, in this system, do not store an actual seat-count type of inventory, but do store several levels of availability status (such as available, not available, in cushion, etc.) for each flight or flight segment. Automatic provision is made for routing sell-cancel calls over the tie line to the central processor which does store the actual inventory on all flights and flight segments. Since there are several availability requests for each sell-cancel transaction, the availability processors act as filters to reduce the tie-line traffic. The availability processors are updated automatically by the central inventory processor whenever the inventory level of a given flight segment reaches a value which dictates a different availability status.

The tie lines utilize voice channels which are capable of transmission at the rate of 750 bits per second. Tie-line terminal equipment incorporates buffer stores of the magnetic-core type to permit optimum-line loading.

Each of the availability processors serves local keysets as well as a number of remote lines which feed traffic from cities in its regional area.

Integrated Railroad Reservation System

The requirements of a railroad reservation system differ considerably from those of an airline system, due principally to the requirements for storage of information on a large number of types of reserved space. The economies of regional storage of data can still be effected, however, since sales of space on certain trains will predominate in the region from which the trains depart.

In the Teleregister system designed to serve the New Haven, the New York Central, and the Atchison, Topeka, and Santa Fe Railroads, two identical data processors are located in New York City and Chicago, respectively. These processors are connected by a 75-word-per-minute tie line. This tie line enables either center to direct an agent set message to the other center and receive a reply, to receive a message and send a reply, to receive a printer message, and to send a printer message. The total inventory is split between the two processors in such a way as to minimize tie-line traffic. Agent set calls are automatically addressed to the appropriate data processor as determined by the particular space involved in the transaction.

A map of this system is shown in Fig. 4. As in the case of the airline system, each processor serves a number of remote lines along each of which are located cities having a group of agent sets. Each processor also serves agent sets in its local reservation rooms and ticket offices.

A block diagram of the tie-line equipment is shown in Fig. 5. A data-processing center has local agent sets and long-line terminals serving agent sets at remote cities. Any of the local agent sets or long-line terminals has access to either the local Data-Handling Unit or the remote Data-Handling Unit. The determination of which Data-Handling Unit is to be used is automatic, and is determined by the addressing code generated by the request for specific train routes when a particular plate is inserted into an agent set. When the remote Data-Handling Unit is required, the agent set or long-line terminal bids through the tie-line seeker for the send selector. When an input is connected, this seeker bids against the printer-switching equipment and the tie-line register for the use of the send control. When this unit is available, it controls the pulsing out of the message in 5-element, 7.42 unit code to the remote tie-line terminal. A distinctive character precedes the message to identify it as an agent set query. The receiving control at the remote central station recognizes the character and causes the receive selector to steer the incoming message to the tie-line register where it is stored in relays. Upon completion of the outpulsing, the send control in the originating station becomes available for inputs from the printer-switching equipment or from its own tie-line register. The receive control at the terminal station is now available for printer messages or a reply to one of its own agent-set queries.

Upon completion of the message the remote central station tie-line register bids against the local agent sets and long-line terminals for the Data-Handling Unit. When it has received and stored its reply, it bids against the tie-line seeker and printer-switching equipment for the use of the send control. When connected, the send control pulses out the reply preceded by a distinctive character to identify it as such. The receiving control at the original station recognizes this switching character and thereupon switches the reply through the tie-line seeker to the original agent set or long-line terminal.

Certain agent set calls and some Data-Handling Unit operations cause printer messages to be perforated by the Data-Handling Unit 600-wpm, 5-unit punches. These printer messages can be directed to local Receive-Only printers, printers on the long lines connected to the center, or to local or remote printers associated with the remote Data-Handling Unit. The perforated messages have sufficient addressing characters to select system, line, station and printer. When the printer-switching equipment sees an address associated with the remote Data-Handling Unit, it bids against the tie-line seeker and tie-line register for the send control. When given access, it pulses out its message preceded by an appropriate switching character. This switching character directs the receiving control to feed the incoming message into the FRXD. This FRXD or RT is a receiving reperforator and a transmitter distributor. The machine is built so that a loop of tape can be stored between the punch and distributor. This loop forms as the message comes in. When the message is complete, the FRXD is connected to the printer-switching equipment which selects the designated local printer or long-line terminal associated with a remote printer and completes the routing of the message. The tie-line equipment is designed such that two agent set messages, one from each end, can be simultaneously in progress. By the same token a printer message from one end can be in progress at the same time as an agent set query from the other end.

The tie-line equipment is equipped with odd parity checking and various time outs to prevent service delays.

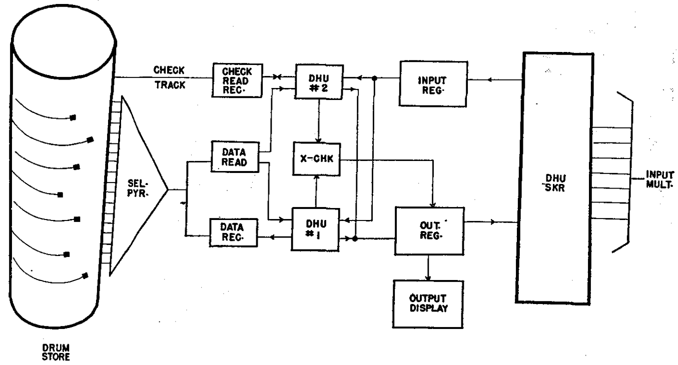


Fig. 6—Dual data processor operation.

Reliability Considerations

In an on-line system of interconnected Data-Handling Units, reliability of the various components of the over-all system becomes a prime consideration. The system must be designed and maintained so as to secure long periods of trouble-free operation and rapid correction of failures which do occur. Means must be provided wherever possible to permit independent operation of the various units, so that the system may be operated with somewhat reduced performance characteristics, even when one of the component subsystems has failed.

Of great importance, of course, in maintaining this reliability is the use of carefully chosen electronic and electromechanical components of high quality. To effect this requires an active standards program and careful life testing of components intended to be added to the standard list. In the equipment described, for example, vacuum tubes expressly designed for long life have been chosen. These tubes, when operated under derated conditions in the equipment, have demonstrated average lifetimes of well over 50,000 hours. Bifurcated contact relays of the telephone type are used for electromechanical switching and are capable of several million operations before any need for adjustment.

To further assure reliability of the central Data-Handling Units, portions of the equipment such as the electronic data processor are duplicated. Reliability can be increased by a large factor through the use of this technique. For example, if the probability of failure of a given unit in a specific time interval is one part in a thousand, the probability of failure of both of two identical such units during the same specific time interval is of the order of one part in a million.

A block diagram of such a duplicate processor is shown in Fig. 6. Calls coming in to the Data-Handling Unit seeker shown at the right-hand side of the diagram are stored in an input register and fed simultaneously to two Data-Handling Units. The process of a given call as it progresses through each of the units is checked by means of a cross-checking circuit, and the call rejected, should a check not be obtained. Where the call is one which requires writing on the drum to change the stored information, this is done simultaneously on a check track fed by one of the

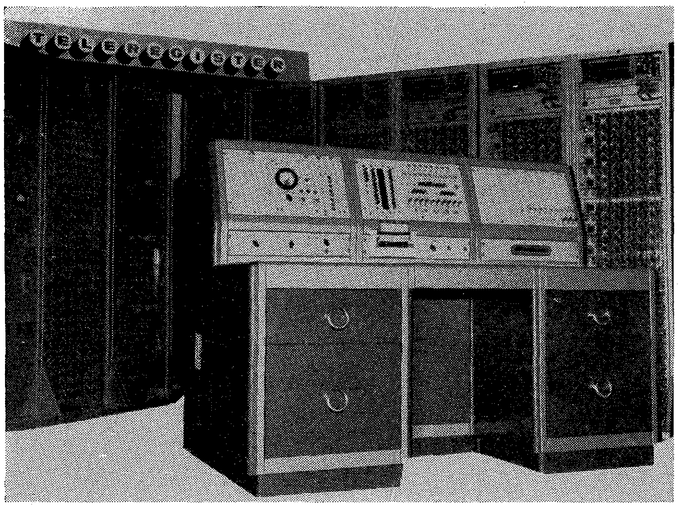


Fig. 7.

Fig. 8.

Data-Handling Units and on the appropriate inventory track fed by the other Data-Handling Unit and selected by a track-selection pyramid. The changed information is then read back from both tracks through the Data-Handling Units and compared by the checking circuit for certification. Should verification not be obtained, the call is printed out so that corrective action may be taken.

In the design of some systems of this type, facilities are provided for automatic switchover in case of failure of one of the systems to the remaining system, which can operate alone. Indication of this type of operation is provided to maintenance personnel.

To provide further reliability, the Data-Handling Units are operated in a temperature-controlled environment and with fully attended maintenance. Maintenance and adjustment periods of approximately two hours per day provide opportunity for the employment of marginal-checking techniques for weeding out components for which failure is incipient.

Open rack type of construction is used for the equipment to enable access to all of the components, in order that repair time may be minimized. The equipment is provided with a maintenance console which indicates to maintenance personnel the progress of calls through the Data-Handling Unit and which provides a facility for the introduction of test calls in order that failures may rapidly be located. A typical system arrangement is shown in Fig. 7.

Of equal importance in the over-all problem of system reliability is the minimization of human error by operators who have access to the system and the facility to change the stored information. It is of paramount importance that agent sets be well designed from the human engineering standpoint, with a view to simplicity, ease of operation and protection against human error. Fig. 8 shows an agent set designed for airlines use in accordance with these principles. The use of a plate inserted into the agent set provides an almost fool-proof method of addressing to a particular location on the magnetic drum. It provides the

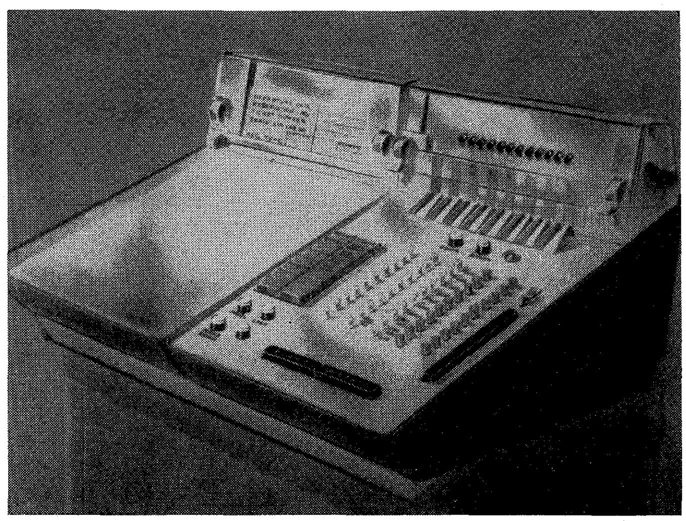


Fig. 9.

facility for automatically coding information which would otherwise require a number of button depressions. The addresses to which the plate directs the inquiry are clearly shown in printed English on the plate itself, which can also be used to display other information, such as fares. The insertion of the plate in one position together with the use of eight matching keys located immediately below the lower edge of the plate enables simultaneous access to the store for availability information on each of eight flights, legs, or flight segments. Depression of the appropriate button then restricts sell or cancel action to the particular flight or leg involved.

Facilities for inserting, by means of clearly marked buttons, information on the number of seats required, the data and the action required (such as sell, cancel, waitlist, etc.,) are provided. This design has been shown by experience to require very little operator training and to result in few operator errors.

The same general approach was employed in the design of a similar set for use with the Railroad Reservisor system. This is shown in Fig. 9. Here it is necessary to provide additional buttons due to the requirement for listing a number of types of space, such as seats, bedrooms, etc., but the design of the equipment is basically similar to that of the airlines agent set. In the case of the Railroad Reservisor, the reply-back information from the central data processor is also somewhat more complex, and it has been found best to display this information in printed form rather than with lights, as in the case of the airline set. In the illustration shown, the printer is shown mounted adjacent to the agent set. The reply-back information is ejected from the printer on prepared forms, one copy of which can be given to the customer.

It is believed that the systems described above represent tangible forward steps in the development of on-line data-processing systems, which enable these systems to be greatly extended geographically, with consequent increased utility.

Discussion

W. A. Morgan (International Business Machines, Owego, N.Y.): Do you buffer your inputs and outputs or allow each agent set to tie up the data-handling unit during processing time?

Mr. Gaffney: Each agent set contains its own buffer register. The call is set up in the agent set by the operator and a master seeker permits each agent set to have access to the data-handling unit in turn. The data-handling unit has its own input-output buffer and processes agent calls at its own high-speed rate.

E. Ziolkowski (Datamatic): In interrogating the system, how is the problem handled when two requests are made simultaneously for only one available reservation? Can conflict occur?

Mr. Gaffney: There is no conflict. The data-handling unit processes the calls sequentially so that only the first one having access to the equipment will obtain the reservation; the second call will be rejected.

Claude Kagan (Western Electric Co.): What error detection and correction facilities are provided in the Railroad Reservisor tie-line system?

Mr. Levine: In this system, the primary error detection facility is an odd-even parity check on each character. In addition, there is an over-all message character count. Error detection facilities are available on the data-handling unit so that nonallowable codes will be rejected. Error correction facilities are not provided; however, in the event of a detected error, the agent set receives an error signal and no change in the inventory is made. The call is then repeated.

Mr. Harris (Stromberg Carlson): Please describe the physical facilities over which your high-speed—750 bits a second—transmission takes place. For example, carrier, long-distance telephone, leased wire.

Mr. Levine: The physical plant facility to be used for transmission of 750 bits per second data will be a private wire, leased, telephone quality voice channel. The terminating device will be a data modulator-demodulator unit to be furnished by the communications common carriers. Our data terminal equipment will include necessary buffer registers and will serialize the data in the form of a series of dc pulses fed to the data modulator-demodulator.

Mr. Harris: How many stations does the Pan-Am system serve? By station, I mean one group of individual agents' equipment.

Mr. Levine: In the Pan-Am system at present we have 26 cities served by the system, and a total of 120 agent sets.

P. F. Radue (Automatic Electric Co.): What technique is used to assure correct transmission of information where parity check leaves gaps?

Mr. Levine: The technique for error detection that we expect to use will include both vertical and horizontal parity checks. The vertical parity check is a single parity check on each character while the horizontal check is performed on the entire message. This technique is expected to reduce the number of undetected errors to a negligible value. In addition, as discussed above with reference to the railroad system tie line, the data processor will reject any nonallowed codes. The message will be retained in a buffer register at the transmitting end until an acknowledgment signal is received. The message will be repeated if errors are detected. If the acknowledgment is not received, a print-out will occur after a suitable time out.

F. A. Reynolds: What is the effect of errors on increasing the traffic on the lines? How do errors break down between operator errors and facility errors?

Mr. Levine: Both of these questions are difficult to answer precisely in that systems of the type discussed have not been in operation long enough to provide a good statistical sample. The first part, on the effect of errors on increasing traffic, may be answered as follows. With proper operation and maintenance of both lines and terminal equipment, the number of retransmissions due to error should not exceed approximately 1 per cent. This type of performance has been achieved. However, during initial breaking periods this rate has been somewhat higher. The retransmissions are not expected to increase traffic sufficiently to cause difficulty. The second part of the question concerning distribution of operator and facility errors may be answered as follows. Error checking facilities are included in the operator's equipment to reject improper input data; however, certain operator errors will be undetected except through audit procedures. The number of undetected operator errors should be low.

B. Hasbrouck (Atlantic Refining Co.): If space is sold out, can all interested regional processors reject such requests locally? If space is still available, can any requests be answered on the regional level or must each such request go individually to the proper central for servicing?

Mr. Levine: Yes, the regional processor rejects requests locally if space is sold out or permits acceptance of sales if space is available. Agent sets receive their answers directly from the regional processor which forwards the transaction to the central processor.

Reservations Communications Utilizing a General Purpose Digital Computer

R. A. McAVOY†

THIS paper describes a communications system developed for the purpose of establishing communications between a general purpose digital computer and a large number of employees whose work requires frequent and immediate access to the services of the computer. The communications system is, to a degree, "general purpose" in nature; however, in the specific application to be described, the system will be used by airline employees in communicating with a computer which is engaged primarily in processing airline reservations data.

An over-all view of the system may be obtained by reference to Fig. 1. The rectangular area at the top of the drawing represents the common location of those portions of the system symbolized there, including the central computer. In the lower right-hand portion of the drawing, a smaller area represents a typical location remote from the computer. The remaining portion of the drawing contains a brief legend for later reference; it may be disregarded at this time.

Referring again to the upper portion of the drawing, the group of small circles to the left represents a large quantity of units called "agent sets." The agent set is the primary communications device through which the individual employee transmits and receives information. This device will be described quite completely later. At this point, it is only important to note that messages to be transmitted are formed by a combination of actions including the pressing of buttons and the operation of other controls. Messages received by the agent set are displayed in the form of illuminated signs or signals.

To the right of the agent sets, the block labeled HSPS represents a group of units called high-speed programmer scanners, which perform the following functions.

1) They provide service, sequentially, to agent sets which have complete messages ready for transmission.
2) They establish the sequence in which the characters of the message are transmitted and cooperate in the transmission of messages at the rate of 200 characters per second.
3) They receive reply messages from the computer and translate the replies into selective illumination of signs or signals on the agent set, causing the illumination to persist until retired by a specific manual operation at the agent set. This latter activity does not totally occupy the programmer scanner which proceeds immediately to serve the next agent set after translating the reply for the first.
4) They perform parity check and character count check on received messages. Upon recognition of error, the programmer scanner causes repetition of the reply. If the second reply is error-free, normal operation is resumed. If the error persists, the programmer scanner will cause an appropriate sign to be illuminated on the agent set. Depending upon the nature of the input message, the agent set will display a sign reading ERROR or a sign reading RESET.

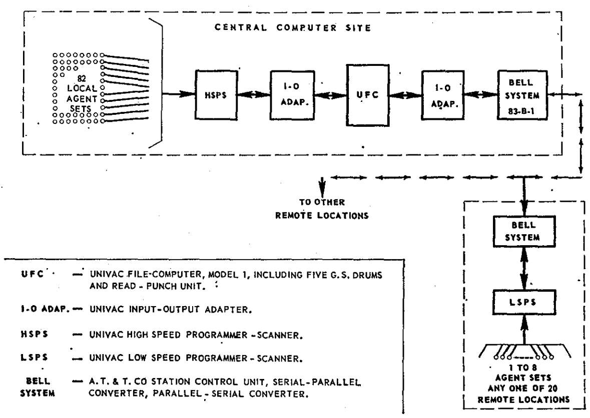


Fig. 1.

Continuing to the right on the diagram, the next box represents a unit called "input-output adapter." This unit relays messages between the high-speed programmer scanners and a section of the computer which we shall call the "message box." The loading and unloading of the "message box" by the input-output adapter is independent of the processing functions of the computer.

The input-output adapter loads the message box and then notifies the computer to that effect. The computer subsequently processes the message, deposits a reply in the message box, and notifies the input-output adapter. The adapter immediately relays the reply back down the line to the agent set.

The input-output adapter also checks parity and character count, notifying the computer when an error is observed. Error notification to the computer is used to make appropriate modification to the program.

The entire communications process just described occupies a period of time ranging from 90 to 125 msec depending upon the nature of the input message. This figure does not include processing time in the computer which will range from about 100 msec to slightly more than one

† Supt., Communications-Data Processing, Eastern Air Lines Inc., Miami, Fla.

second. Most input messages will be processed by the file computer in less than 250 msec.

Referring again to Fig. 1, the large box labeled UFC is, of course, a representation of the Univac file computer. Although only one box is shown on the diagram, the computer system will be composed of approximately a dozen large equipment cabinets. The system will include five magnetic drums having a storage capacity of 900,000 alpha-numeric characters. The storage system is capable of expansion to thirty-three drums having a storage capacity of nearly 6,000,000 characters.

At this point it is appropriate to call attention to the fact that so far we have been referring to a communications system which is wholly contained on our premises. The greatest distance intervening between an agent set and the file computer is not likely to exceed 300 feet. Let us now direct our attention to the communications system which serves agent sets at remote locations.

Referring now to the lower right-hand portion of Fig. 1 we note that the area represents "any one of 20 remote locations." These locations may be as far distant from the computer location as one chooses to think, whether it be one mile, a hundred miles, a thousand miles, or more. The only requirement is that the remote location be capable of being connected to the computer location by a dependable telegraph circuit.

The limit of twenty remote locations is established by choice. Up to forty remote locations may be served by one telegraph network of the type referred to herein.

If we follow the path between the remote agent sets and the Univac File Computer we note that the labels in two of the boxes are somewhat familiar. At the top end of the path next to the computer we find an input-output adapter. This unit performs essentially the same functions as the other unit of the same name. The essential difference is that it contains a few relays which are required to compensate for the lower speed of communication.

At the other end of the path, we see a box labeled LSPS which represents one unit named "low-speed programmer scanner." This unit performs essentially the same function as the high-speed programmer scanner including parity checks and character count. In the case of this unit there is not only a difference in speed of operation but, as the diagram indicates, one unit serves a maximum of eight agent sets. Although the unit is capable of operating speeds up to 20 characters per second, it is limited by the speed of the telegraphic equipment which is, in this case, 10 characters per second.

A feature of the low-speed programmer scanner is the ability to operate two such units as a single unit with a capacity to serve as many as sixteen agent sets. Doubling of units in this manner does not change the appearance to the telegraph network.

As indicated on the diagram, the telegraph network is leased from the Bell System. The Bell System in this case is represented by the A.T.&T. Company which operates most of the intercity circuits of the Bell System. The basic system is called "83-B-1 Selective Calling System;" in this application it is modified for use with the Univac File Computer.

A complete description of this system is beyond the scope of this paper. However, the principal characteristics of its operation will be evident in the following description of a cycle of operation.

The elements of the 83-B-1 system in this application are a "control station" located at the central computer site and a number of "station control units"—one at each remote location.

The "control station" initiates a continuous series of events by transmitting a particular set of two characters which are received by all of the station control units. This set of two characters is referred to as a "start pattern." One, and only one, of the station control units recognizes this particular set of two characters and responds in one of two ways. If the associated low-speed programmer scanner has *not* indicated a need for service, the station control unit transmits one particular character which when received at the "control station" causes that station to transmit a different set of two characters; *i.e.*, a different start pattern, thus directing a different remote station to respond.

If the low-speed programmer scanner *has* indicated a need for service, the station control unit transmits characters submitted to it by the low-speed programmer scanner. The characters submitted by the LSPS are, of course, those which are represented by the position of the controls on the particular agent set which is then being served. Should the "control station" fail to receive a reply of either type, audible and visual alarms will be set off at the control station location.

At the completion of this transmission the telegraph network becomes idle until the control section of the File Computer tells the input-output adapter that a reply message is ready in the "message box." The control station transmits the characters submitted to it by the input-output adapter, thus completing the two-day communication.

If a second agent set at the remote location is awaiting service at this time, its message will be transmitted after a delay ranging from roughly 50 to 150 msec. The amount of the delay is determined primarily by the relative positions of the two agent sets in the fixed service sequence. As the number of sets requiring service increases, this delay decreases.

When each of the active agent sets at a particular location has been served once, the low-speed programmer scanner submits for transmission a particular set of two characters, which are then transmitted by the station control unit. The control station at the central computer location responds by transmitting a start pattern for a different remote station.

When all of the remote stations have been served, the control station begins a new cycle without delay and the process continues without interruption until the hour arrives for the start of daily maintenance procedures at the computer center.

The telegraph system in this application will be operated at the nominal speed of 100 words per minute or approximately 10 characters per second. The system includes protective features which supply visual and/or audible alarms in the event of various types of failures which may occur.

Having completed the general description of the communications system, attention may now be directed to a more detailed examination of a most remarkable piece of communications equipment—the agent set.

Fig. 2 is a photograph of an agent set. The predominant feature of this unit is a photographic projection system in which a light beam is passed through a single frame of a conventional 35-mm film projecting the image of the film on a viewing screen on the upper panel of the agent set. Each frame of film is precisely mounted in a holder two inches square. A cartridge containing thirty holders, *i.e.*, thirty pictures, fits loosely in a tunnel, the opening of which is visible at the lower right-hand corner of the agent set.

A rack of teeth along the edge of the film cartridge engages a pinion gear controlled by a knurled wheel visible along the right edge of the lower panel. The top of the film cartridge contains an index which is visible to the operator of the set.

Therefore, a selection of thirty images is immediately available to the operator. Additional groups of thirty images may be referred to quickly by removing one film cartridge and inserting another.

The cartridges incorporate a simple locking mechanism which prevents accidental removal of individual film holders. The potential difficulty associated with spilling and subsequent faulty rearrangement of the file is thus avoided.

On each individual film, a narrow strip along the right edge and a narrow strip along the top are reserved for digital coding in a five-channel code. These strip areas are appropriately divided into three discrete areas each of which in turn is divided into five areas. Each of the five areas is made black or clear white to represent binary "1" or binary "0." Thus, there is created the means for specifically identifying any one of a thousand images, assuming the identification system is based on the decimal system.

After a film slide has been selected by reference to the index and by operation of the knurled wheel, a slide lever is moved to the left, causing the projection lamp to be lighted and the selected film to intersect the resultant light beam. In addition to the image which appears on the screen, a pattern of light described by the coding along two edges of the film is impressed on fifteen light-sensitive elements located behind the upper panel bordering the viewing screen. The condition of these elements is subsequently sensed by the programmer scanner to transmit a specific set of three digits to the computer.

All other information transmitted from the agent set is obtained by sensing the five contacts of the buttons which have been depressed. Each button is one of a particular set; the button supplies the decimal information in five-level code while the identity of the set of buttons is retained by virtue of the position of the digit in the total message. Mechanical interlocks prevent the selection of more than one button of a set.

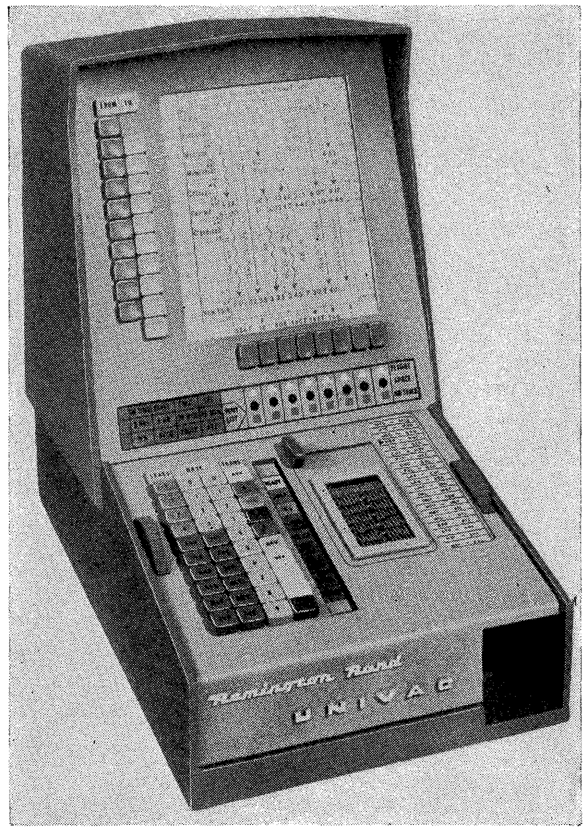


Fig. 2.

A system of electrical interlocks prevents the agent set from attempting to establish communication if any of the required elements of a message have not been entered.

Before referring to the next illustration, note that there are two sets of buttons along the left edge of the viewing screen and one set of buttons along the lower edge.

Refer now to Fig. 3, which is an example of an image which might appear on the screen of the agent set in our application of airline reservations.

An employee wishing to inquire about the availability of two reservations from New York to Houston would have selected this slide. The employee would then enter the set of buttons meaning FROM and push the particular button to the left of the words NEW YORK, and next, the TO button opposite the word HOUSTON and other buttons specifying month, day, number of seats, and type of transaction.

No further manual operation is required to code this message, gain access to the communications circuit, transmit the message, consult the records, prepare a reply, transmit the reply, receive it, decode it, and display the answer in front of the agent. Yet the entire operation takes place in less time than it takes to describe it.

The answer to an inquiry of the type illustrated covers the availability of all flights represented in the horizontal

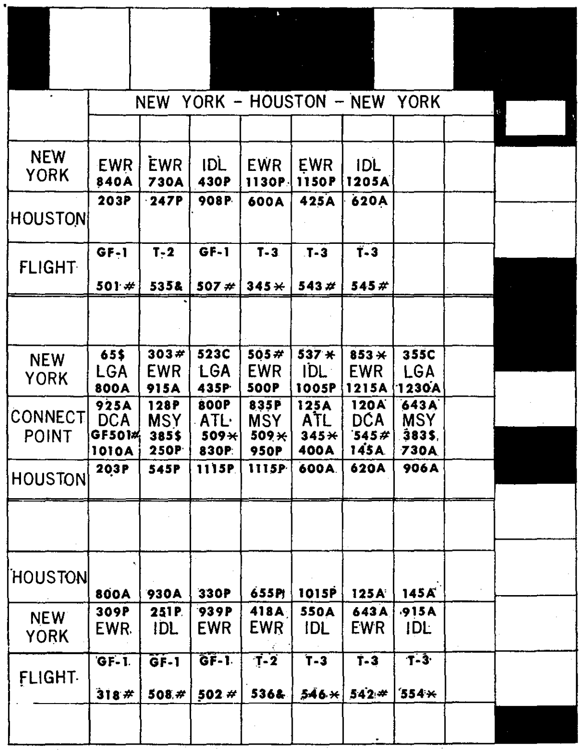


Fig. 3.

columns corresponding to the FROM, TO buttons selected.

Referring back now to our picture of the agent set, Fig. 2, attention should be directed to a long rectangular area near the bottom edge of the upper panel. In this area are located the twenty-eight individual signs or signals which may be illuminated to provide a direct answer or to signify an answer to an inquiry.

The twelve lamps to the left, arranged in three rows and four columns, are individually controlled and provide back illumination to words or phrases which, in the airline application, constitute the range of replies to inquiries concerning estimated time of arrival or departure of the flight specified in the input message.

The sixteen lamps to the right are arranged in eight pairs, each pair corresponding with one of the buttons immediately above it and thus corresponding with the flight which appears in that column on the viewing panel. Each pair of lights may have one of three acceptable conditions of illumination which represent one of three conditions of availability; *i.e.*, available, not available but expected to become available, not available and not expected to become available. The fourth possible condition, which is both lights not lighted, is not an acceptable condition and is interpreted as evidence of malfunction.

Near the center of the bottom panel, just to the right of the four columns of buttons, nine lamps provide back illumination for an equal number of words or phrases which supply verification or instructions to the operator.

In addition to the words ERROR and RESET previously referred to, there is one other instruction word, WAIT. Illumination of the WAIT sign informs the operator that transmission has started. This sign remains on until a reply is received.

The set is used by the airline employee to adjust the central inventory when reservations are made or cancelled or an auxiliary inventory may be adjusted to reflect unsatisfied demand. Each such transaction is verified by illumination of an appropriate sign in this bank.

Unfortunately, time does not permit description of the intended expansion of the system. Much of it can be deduced from the communications capabilities described. Widely separated File Computers can communicate with one another through their "message boxes" and much simpler teletype networks than the 83-B-1.

An accessory now under development will establish an interlocking relationship between a specific punched card record and a specific agent set transaction.

An accessory available immediately will provide discrete identification of the agent set, thus opening the door to measurement of employee effectiveness and other useful benefits.

Credit for the development of this system is due to the many individuals in Remington Rand Univac, A.T.&T. Company, and Eastern Air Lines, Inc. who cooperated with one another so magnificently to convert ideas to reality.

Discussion

Mr. Johnson (Adalia Limited): What is your specification on error rate, and how much redundancy have you found it necessary to add to achieve this specification over leased communications lines? What error correction and detection facilities are provided in the remote agent set system?

Answer: The agent set and its associated programmer scanner cooperate to reject attempted transactions which omit needed data. For example, if an agent should attempt to book space on a flight without specifying the number of seats, an indicator on the agent set would be lighted reading RE-ENTER. The message would not proceed further than the associated programmer scanner, thus avoiding waste of valuable communications and data-processing time.

We use a five-channel code in transmitting input and output messages between the agent sets and the input-output adapters. The four low-order channels are used to form an "excess-three" binary code which is identical with the four low-order channels of the Univac seven-level code. The fifth level is a checking level in which a binary "1" is stored or not stored so as to form a character having an odd number of bits. A parity check is performed on the receiving end of each interunit communication.

Upon recognition of a parity error, appropriate action is taken depending upon the stage at which the error becomes apparent. Generally, the agent set signals RE-ENTER when no inventory adjustment was likely, or ERROR when there was a possibility of inventory adjustment.

Question: If error is indicated on the agent set in a remote area, is the agent able to determine from the nature of the error indication whether the error occurred in the equipment in his remote area or in the central computer?

Answer: The agent set has two back-illuminated signs reading RE-ENTER and ERROR, respectively. In normal operation, the agent's entry of a transaction is followed by illumination of a sign reading WAIT. This informs the agent that his message has been transmitted from the programmer scanner on to the next stage of communication, whatever that might be. However, if the agent omits some necessary portion of the message, the WAIT sign does not light; instead, the RE-ENTER light comes on immediately and thus the agent knows that he has erred. A RE-ENTER light following a WAIT light is indication to the agent that one of several conditions has occurred; the system has detected a parity error at some stage of the process which cannot cause erroneous adjustment of inventory, or the agent has attempted some nonvalid operation such as entering the date JUNE 31, FEBRUARY 30, entering a date November 21 while using a film slide for a flight schedule which expires November 15, etc. An ERROR sign will be lighted only when the following conditions coincide: the transaction involved is one which is intended to adjust inventory and the message has reached the computer; the reply reaching the programmer scanner contains either a parity error or an insufficient number of characters, and this is the second erroneous reply received at the programmer scanner (it having requested a repeat when it received the first such reply).

Communications circuits provided by the communications companies will be equipped with alarms and signals which will provide information regarding cetrain types of circuit failure. These alarms and signals will be available to the agent or, in the case of large installations, to the supervisory personnel.

Question: In the midst of all this wonderful development, is there some way to get Eastern and other airlines to answer their telephone when one calls for a reservation?

Answer: We believe that he development and application of equipment here under discussion is one way to come more near the fulfillment of the objective of the management and employees of Eastern; that is, to serve our actual and potential customers to their complete satisfaction. It may be of interest to you to know that this objective is continually presented to all employees because every paycheck has clearly printed on it, in very large red letters, "The customer pays our wages."

Question: When a cancellation is received on a flight which had been filled, what is done to insure that the seat is first made available to those on the waiting list rather than to the next agent who happens to request the fight?

Answer: This question, and the answer, exemplify one of the reasons we chose to employ a general purpose computer. The computer will store not only an inventory of seats, it will also store the number of unfilled requests for seats, *i.e.,* the waiting list. As each cancellation is received, the computer determines whether there are any unfilled requests; if there are unfilled requests, the cancelled space is reported out immediately on a punched card which is then routed to the position where the waiting list is held.

Question: Is there some provision for using the file computer on other problems during the night or at times when the computer is not ordinarily busy?

Answer: At the present time we are not planning such use during the busier hours of the day because we have yet to learn how much spare time will be available. However, assuming available computer time, there is no reason why such use should not be made. The equipment is capable of such operation. We plan to use it at night in slack hours to develop statistical data from the inventory during the same hours when it is still serving agent sets.

Question: If a flight leg can occur on several different routings, does the agent have to check several plates?

Is not the agent set itself able to store information received? Also information requested?

Answer: Some definition of terms is necessary to avoid misunderstanding. A "leg" is defined as that portion of a flight from take-off to the next point of landing. A "segment" is defined as that portion of a flight which lies between the boarding point and the deplanting point of a particular passenger. A "flight" is the operation of an aircraft from the point where it is identified by a particular flight number to the point where it is no longer identified by that same number. Within these definitions, the question is best answered by stating that the agent never has to check more than one film slide to obtain the desired information on any flight leg or on any flight segment. If a passenger's itinerary involves more than one flight, it may be necessary to refer to a maximum of one slide per flight.

So long as the keys on an agent set are depressed, the information so represented is available to the programmer scanner. And, of course, the film slides store a great quantity of information. When an answer has been displayed on an agent set, that answer is continuously available until the agent "clears" the transaction key or operates the lever to restore the film slide in the magazine.

Question: How long does it take for the two-character start pattern to reach a remote station?

And, can a busy remote station's agent sets be served more than once in a given recognition cycle?

Answer: The time will be dependent upon the traffic situation at any given time. Under a "no traffic" condition each remote station will be polled once every five seconds, making the average access time two and one-half seconds. This answer refers to the 83-B-1 ten-character per second system, with twelve remote stations.

The agent sets at a given remote station can be served only once during a given "recognition cycle." When more than one set is served on the same poll, the average access time is correspondingly reduced.

Question: Is the information from the local agent sets transferred to UFC through the HSPS and IO adapt in a serial or parallel mode?

And, where are the messages from the remote agent sets buffered? What is the capacity of the buffers and how many buffers are there?

Answer: In serial.

For the purpose of this question, the remote agent sets should be considered to be grouped so that all agent sets served by particular input-output adapter are in a common group. Within such a group, agent sets are served sequentially; thus, each agent set stores its own message until it is served. Each input-output adapter stores the message it is then handling on to a portion of one track of the high-speed drum in the UFC. When the UFC tests the input-output adapter and receives a reply that there is a message waiting, the UFC processes the message, stores the reply in an unused portion of the same track, instructs the input-output adapter to take the reply and begins testing other units to see if there is work waiting to be processed. At the outset, we will be using two input-output adapters (therefore, two buffers) for agent set work. Other tracks on the high-speed drum will act as input-output buffers for the electric typewriters and the read-punch unit.

Question: Is there any provision for expanding the system to handle reservation names as input—printed output—ticketing, et cetera?

And, how many agent sets, total, can one system handle and still provide satisfactory service?

Answer: Initially, we plan to have the read-punch unit produce punched cards containing the reservation details. These cards will be matched with similar cards first written by the agent and then keypunched. This process will detect errors of many kinds. Although there has been considerable reflection and discussion regarding extending into the ticketing area, it would be an exaggeration to say that such provision has been made.

The answer is entirely dependent upon the average handling time of the transactions at each of their various stages, and the speed of service which is considered "satisfactory." At the present time there is not sufficient information available regarding the "mix" (relative quantities of various types of transactions), nor is there available information of sufficient accuracy regarding the handling times. Because of this uncertainty, we have (we believe) deliberately underloaded the system for our initial application in order that we may safely collect the information necessary to plan intelligent expansion.

Question: What developments are there for storing passenger identification for preparing lists to be checked against later ticket purchase?

Answer: The answer to a previous question partially answers this one; that is, the comparison of computer produced punched cards with the manually written and keypunched cards. Ticket purchases will be related to the keypunched cards (now audited for comparison with inventory) for verification. When reservations are made at the time of ticket purchase, the computer produced card (produced when the ticketing agent operated the agent set) will be held for comparison with a keypunched card produced following a telephone call from the ticket agent to a voice recorder at the reservations office.

Question: What computer do you use? How long has it been used in this operation?

Answer: We plan to use the Remington Rand Univac, Model 1 File Computer, and at this date (March 10) the program debugging has progressed to the point where it is about 95 per cent complete. Agent set sales have been entered through prototype equipment into the particular Model 1 File Computer which will be installed in our New York Office this summer. All the transactions mentioned have been tested and have produced the results anticipated and related here.

Question: Is this system operational? If so, what is the operating reliability experience?

Answer: See answer immediately preceding. We are asking for a reliability exceeding 99.7 per cent of the scheduled operating time.

Question: What is the present state of the development of the system? What, if any, portions are in operation?

Answer: See answers immediately preceding.

Question: What steps do you take if there is an interruption between the agent's set and remote location?

Also, is the information contained by the central computer retained elsewhere for reference in the event of system failure?

Answer: The action to be taken will depend to some degree upon the particular circumstances, such as the duration of the interruption, the proximity of other agent sets and their operating condition, the nature of the interruption, etc. In general, a location dependent upon agent set service will use other communications facilities such as telephone and/or teletype when the agent set service is not available.

Each time that an agent set operation results in adjustment of the inventory, the computer produces a punched card record of the transaction (as stated in answer to previous question). This punched card record also contains a record of the entire inventory for the flight as it exists after the adjustment which produced the record. It is planned to process these records at fifteen-minute intervals on conventional punched card processing equipment so that it will be possible to revert to a completely manual system within a few minutes if necessary. This protection will be removed when experience proves that it is no longer necessary.

Question: What were some of the considerations that led you to choose a general purpose computer rather than a special purpose computer?

Answer: The answers to previous questions have partially answered this one. Perhaps the most important single consideration was recognition of the fact that we didn't really know *in sufficient detail* just what we wanted the computer to do. The electronic data-processing science and our need to apply it were developing at the same time. We had to get a program under way without being able to supply detailed specifications. It was apparent that it would take us several years to reach the point where we would know what to specify; it also was apparent that we would need the equipment by that time. We, therefore, decided to select as the central data processor, a general purpose computer which would meet the broad requirements of large capacity random-access storage and something faster than slow-speed processing. (We would like to have had a much faster data processor.) With this much behind us we could concentrate on the detailed specifications for the equipment which at that time appeared to be more special purpose; *i.e.*, the agent set and associated communications equipment. As it has turned out, the agent set and related equipment are a great deal more general purpose than any of us anticipated. Within the next decade there will be a tremendous increase in the use of these devices.

Question: What is the required up time per day for this system and what provisions are made for handling the load in the event of a major electronic or mechanical failure?

Answer: The required up time is twenty-two hours per day, seven days per week. As previously indicated, punched card records will be available to provide the basis for reverting to a manual system of control. The manual system would be comparable to our present system, modified only to fit the new type of record and to take advantage of then existing facilities.

Stock Transaction Records on the Datatron 205

A. H. PAYNE†

SEVERAL users of stock market transaction records maintain teams of ticker watchers to compile price files. Six to eight such teams are kept by wire services and some newspapers. Each of the two major odd-lot brokers and Teleregister Corporation, with its quotation board service, are among others who monitor the ticker visually. Not all of these cover more than one exchange but some do.

The use of a computer to monitor the ticker automatically was accomplished by Melpar for Standard and Poor's Corporation in order to compute their "Standard 500" indexes. Fig. 1 shows schematically the flow of information from the standard Western Union 6-channel ticker code through a converter to a punched paper tape which is suitable for input to the Datatron 205 computer. Examples of the paper tape involved are shown in Fig. 2. In Fig. 3 is shown the correspondence between ticker characters and digit-pairs within the computer. The 4000 words of main storage within the computer are used as diagrammed in Fig. 4. Outputs from the computer are the hourly indexes, daily small-group indexes, and price files for starting next day.

† Melpar, Inc., Boston, Mass.

Operating Speed

At maximum ticker speed the computer is occupied about 55 minutes of each hour, processing, computing indexes, printing rejected prices, and punching the price file periodically for safety. The level of market activity varies considerably and on less active days, the computer may be free over half the time. Eventually this extra time may be used for other jobs, but during our development phase we have preferred to use it for increased reliability.

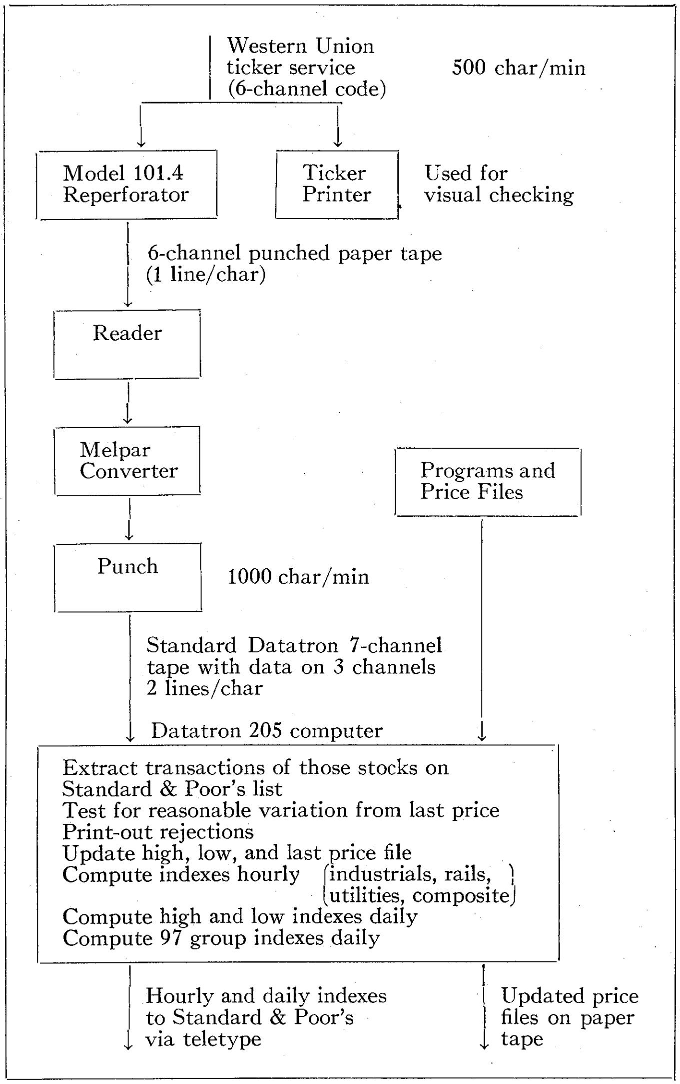
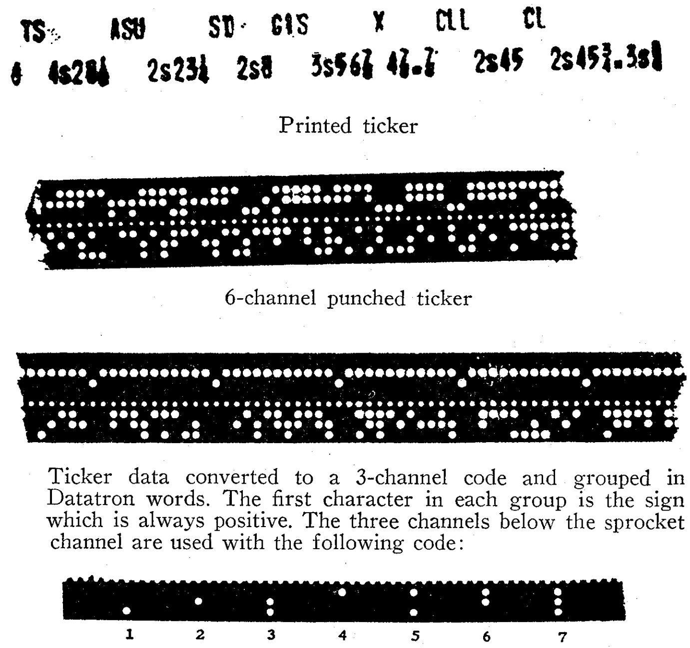


Fig. 1.



Ticker data converted to a 3-channel code and grouped in Datatron words. The first character in each group is the sign which is always positive. The three channels below the sprocket channel are used with the following code:

Fig. 2—Punched tape coding.

Upper Case				Lower Case			
A	30	T	01	0	42	$\frac{5}{8}$	67
B	23	U	34	1	70	$\frac{3}{4}$	65
C	16	V	17	2	63	$\frac{7}{8}$	61
D	22	W	31	3	56	b	73
E	20	X	27	4	62	⊙	44
F	26	Y	25	5	60	c	50
G	13	Z	21	6	66		
H	05	&	33	7	53		
I	14	Pr	04	8	45		
J	32	⊙	10	9	54		
K	36	¢	02	£	76		
L	11			$	47		
M	07			—	43		
N	06			*	75		
O	03			s	64		
P	15			$\frac{1}{8}$	41		
Q	35			$\frac{1}{4}$	74		
R	12			$\frac{3}{8}$	57		
S	24			$\frac{1}{2}$	71		

Fig. 3—Correspondence between printed characters and computer digit pairs.

Reliability

Reliability in the system is attained by hardware duplication, to be discussed later, and by programmed checks. Of the dozens of checking features incorporated into the program, the most important is the check for price reasonableness. Each new price is compared with the last previous price and, if the difference is more than approximately 2 per cent, the price is not immediately accepted but is printed out along with the old price for visual inspection. Such a "rejection" may be caused by:

1) An error on the ticker. In this case the price is properly rejected and a correction will appear later.
2) An unusual configuration on the ticker which looks like a transaction, for example, the report of a dividend in cents.
3) A legitimate price change greater than 2 per cent. The new price is inserted as a special operation.
4) A price-file error. Same as 3).
5) Malfunction of the tape-converter system. The converter is replaced by the spare.

This check therefore serves not only to minimize degradation of accuracy of the answers, but as a diagnostic feature. In this connection, it is desirable to distinguish between a system error which results in degradation of the answers and one which is incapacitating.

Degradation

The computer almost never makes errors which have the effect of degradation; if it is not functioning properly, the program will quickly hang up. Therefore, the primary source of degradation is faulty data input caused by converter malfunction. However, this effect is not serious for several reasons: 1) the aforementioned rejection of gross errors, 2) errors in the input data show up as rejections and alert the operator to switch to the second converter, 3) with 500 stocks in the index an error of no more than 2 per cent in a few of them is not significant, and 4) there is

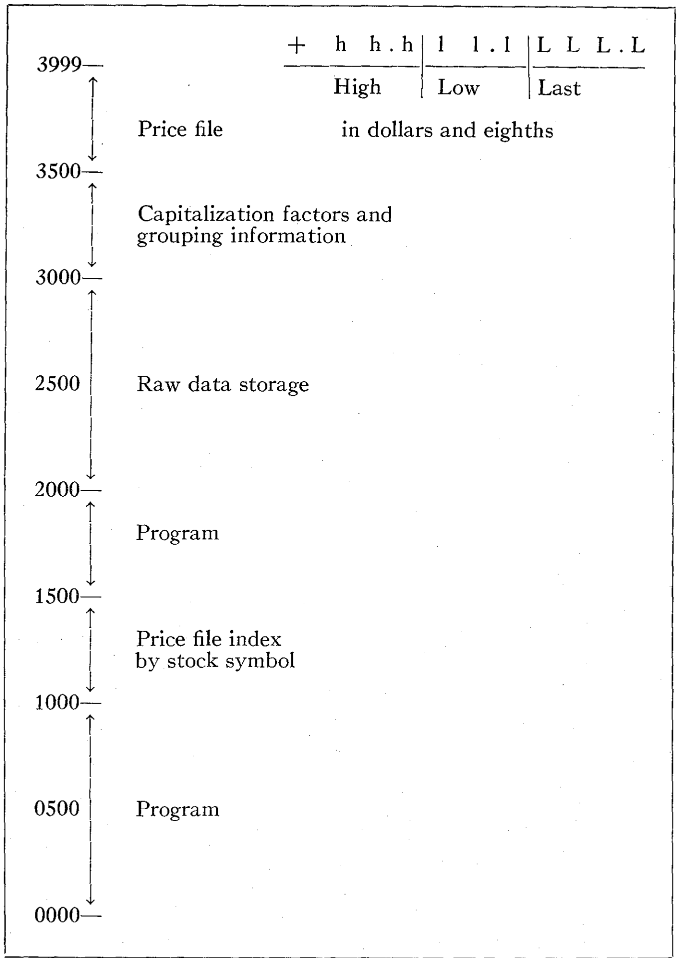


Fig. 4—Computer storage chart.

a chance that an error introduced during the hour will be replaced by a legitimate transaction before the end of the hour when the indexes are computed. (While true of last price, this is not the case for high and low.)

Thus, an acceptable level of accuracy is maintained without operating the converters in parallel, but by merely having a spare ready to insert when one shows signs of failing. The possibility of an incapacitating failure has led us to seek a measure of the reliability of a system in terms of on-line or time-sensitive work.

Time Sensitivity

There is a difference in the quality of urgency one feels about the computation of a table of elliptic functions as opposed to the computation of the trajectory of an enemy missile, for example. But this difference is not solely one of the existence or nonexistence of deadlines—the table of elliptic functions may have a publication deadline. There is a question of cost involved along with some subjective factors which can probably be converted to dollars and cents. We would like to have a formula, a technique of obtaining a quantitative measure of the fitness of a particular system to perform a given job. Such a formula to be comprehensive would be very complicated. The simple approximation of Fig. 5 illustrates what might be done.

To say that S must be less than or equal to unity is to say that the processing time cannot be greater than the interval time.

I = Interval: maximum permissible time between the availability of raw data and the completion of its processing.

P = Processing time: time required for the computer to process the data collected in interval I for which answers are required at the end of interval I.

C = Change-over time: time required to switch from the given job to another one, perform some useful work, and return to the given job.

S = Sensitivity index: a measure of the time sensitivity of a given job on a given computer:

$$S = \frac{P + C}{I + C}, \qquad 0 \leq S \leq 1$$

M = Mean error-free running time: mean time interval which a system will operate without an error.

R = Reliability index: a measure of the reliability of a particular system for a particular job.

$$R = \frac{M}{M + PS}.$$

Fig. 5—Formulas.

The closer S approaches to unity, the more time-sensitive the job is.

For special purpose systems, the change-over time may be said to be prohibitively large, therefore S is close to unity regardless of P and I.

The relation

$$R = \frac{M}{M + PS}$$

implies that, even if M is small, if the term PS can be made small in comparison to M, then a high reliability can be attained. This assumes an efficient error detection and recovery procedure so that available system time can be used to reprocess the data.

R may be regarded as the probability that the job will get done.

If an R is computed for the separate components of a system, then the over-all R is given by the product of R's for the components.

If R_1 and R_2 are the reliability indexes of two parallel systems performing the same job, then

$$R_{\text{total}} = R_1 + R_2 - R_1 R_2.$$

Operating Experience

This operation was entered into by Melpar primarily as a research project and for this reason we have had very little "typical" operation as our control programs and procedures have evolved. One result of this is that the human operators reach the point of boredom without having attained enough automatic proficiency to eliminate errors. In fact, most errors in the system have been traceable either to a human error or to faulty recovery from an equipment error not very serious in itself, thereby reinforcing our belief that human participation in such a system should be kept to a minimum. However, we have devoted considerable effort toward making it difficult for an operator to make an incapacitating error. Most such safety features take the form of programming tricks peculiar to the Datatron computer.

A second result of the lack of "typical" operation is that the figures tabulated in Fig. 6, continuing the reliability measurement example, are based partly on extrapolations and estimates. We have done worse in the past, especially during the first few weeks of operation, but are doing better now, and expect to do much better in the future than the figures indicate.

Comments on Fig. 6

In estimating M, only those errors which resulted in delivery of late or incorrect indexes as a computer output have been counted. Accurate indexes can be extrapolated from previous ones using a few market leaders as weighting factors. This provides a level of backup to our system.

The reperforator is a well-engineered heavy-duty device. However, its large M is due in part to the fact that it is buffered from the rest of the system by the converter, an editing device.

In most cases of human error, the error was precipitated by an equipment malfunction which might have been overcome with little interference by perfect human response.

The figures used for human-linkage P and S contain a larger portion of pure estimation than the others.

Most incapacitating converter errors arise from the fact that an improperly edited data word may take the form of a control word for the computer. Tape-reading control on the Datatron 205 is exercised via special characters on the tape. An additional checking feature is being built into the converter to combat this.

				R per hour		Cumulative R per month	
	P	S	M	Single Unit	Duplicated	Single Unit	Duplicated
Reperforator	1	1	1000	0.999	—	0.862	—
Converter	0.9	0.9	50	0.984	0.9997	0.089	0.956
Human Linkage	0.4	0.4	40	0.996	—	0.548	—
Computer	0.9	0.9	60	0.987	0.9998	0.141	0.969
Composite				0.9664	0.9945	0.006	0.44

$$R = \frac{M}{M + PS}$$

Fig. 6—Reliability of components. The figures shown for composite R may be interpreted as probabilities that there will be no delay of answers or production of wrong answers during the indicated period by the system described here. Another level of backup for the "Standard 500" is provided by extrapolation and manual techniques.

Conclusion

This application has demonstrated the feasibility of processing stock market data as reported on the ticker tape. It has also afforded another example of the use of general-purpose digital computers in on-line jobs. The sensitivity and reliability indexes, while not precise measures nor always applicable, are valuable devices for use in the analysis of system performance.

It is possible to translate computer operating speed into reliability through its effect on sensitivity.

The best applications are those which have associated with them enough off-line processing to support an additional computer which also provides backup to the on-line system.

Discussion

Question: Can you indicate specifically the human links within the described system?

Answer: The converter output is a punched tape which is wound and transported to the photoelectric reader of the computer by hand. The indexes are transmitted from our Boston Laboratory to Standard and Poor's offices in New York via ordinary teletype which involves a manual keyboard entry.

Question: Have you made any cost evaluation between the described system and all-manual operations? What is the relative occurrence of error in the indexes delivered to Standard and Poor's with the computer system as with the previous manual methods?

Answer: This operation has not been done manually. Standard and Poor's did a 90-stock index before going to the present operation. This has been a research project for us and our conclusion is that the job can be done far more economically and accurately on a computer than by manual methods.

Question: How do you check the accuracy of your price files?

Answer: There are several programmed checks, such as the test for reasonableness already mentioned, designed to preserve the integrity of the price files. If the file is read out of memory and then back in again, the file is protected by the formation of a check sum. If the check sum fails to check, a price reasonableness test is run against an earlier file known to be accurate in order to pick out which price is wrong.

A Small, Low-Cost Business Computer

ALEX B. CHURCHILL†

THE Monrobot IX is a desk-size electronic digital computer expressly designed for on-line business applications of those types which are basically repetitive. We include in this category such applications as invoicing, prepayroll computation, and production planning. In all of these applications, an operator must be able to receive problem solutions promptly after insertion of data into the machine. The Monrobot IX produces printed solutions to problems in a fraction of a second to a few seconds, depending upon the particular application.

Fig. 1 is an over-all picture of the computer in its desk. Input and output is by the electric typewriter, and the computer itself is entirely contained within the single pedestal of the desk. The power required is less than 750 watts at 115 volts ac.

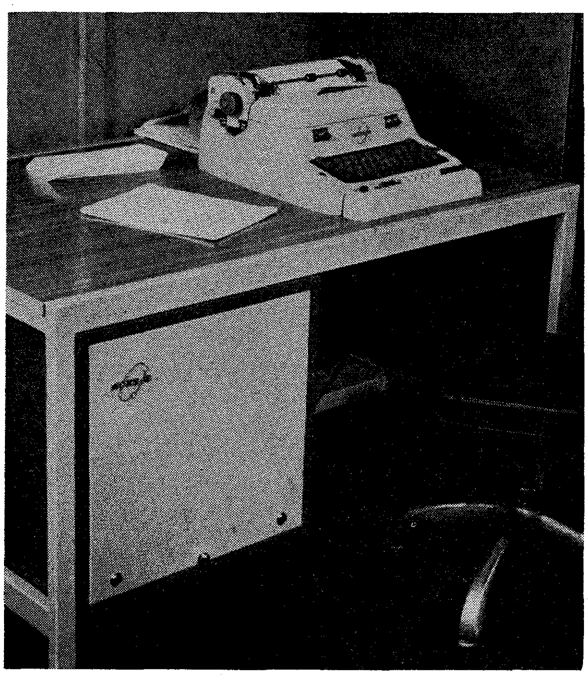
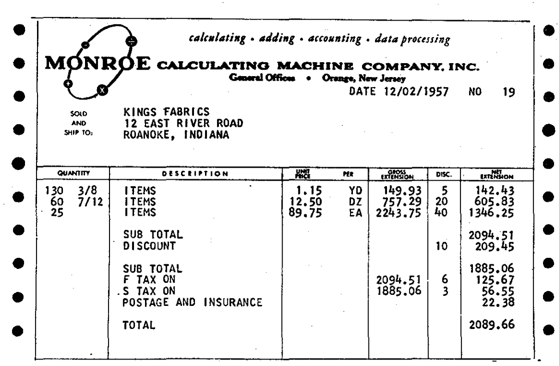


Fig. 1.

All operator controls are at the typewriter keyboard. Fig. 2 is a close-up of that keyboard and helps to illustrate operator requirements as to training and ability. The typewriter itself is a completely standard machine. The nonstandard assembly located at the front of the typewriter is the program-selection keyboard. In this particular illustration the computer is programmed for invoicing. Each program key is labeled to indicate the type of invoice line for which the machine has been programmed. For example, whenever the operator totals out an invoice, she depresses the proper key and the computer, causing the word "Total" and its dollar amount to be automatically printed out by the typewriter in the proper columns. Simultaneously, the accumulation to total accounts receivable is made internally, and the register being used for subtotals is cleared in preparation for the next invoice.

Fig. 2.



calculating · adding · accounting · data processing

MONROE CALCULATING MACHINE COMPANY, INC.

General Offices · Orange, New Jersey

DATE 12/02/1957 NO 19

SOLD AND SHIP TO: KINGS FABRICS
12 EAST RIVER ROAD
ROANOKE, INDIANA

QUANTITY	DESCRIPTION	UNIT PRICE	PER	GROSS EXTENSION	DISC.	NET EXTENSION
130 3/8	ITEMS	1.15	YD	149.93	5	142.43
60 7/12	ITEMS	12.50	DZ	757.29	20	605.83
25	ITEMS	89.75	EA	2243.75	40	1346.25
	SUB TOTAL					2094.51
	DISCOUNT				10	209.45
	SUB TOTAL					1885.06
	F TAX ON			2094.51	6	125.67
	S TAX ON			1885.06	3	56.55
	POSTAGE AND INSURANCE					22.38
	TOTAL					2089.66

Fig. 3.

The capabilities of the machine can best be seen by looking at Fig. 3, which is a complete sample application. This figure illustrates a completed invoice in which multiplication by two different fractions are involved, and taxes are applied to two different subtotals. In this particular example, the date and invoice number, including the alphabetic characters, are automatically typed as the result of operator depression of the date and number-program key. Name and address are normally typed. The operator then selects the proper program to extend quantity times unit price less discount in which the fraction "eighths" occurs

† Monroe Calculating Machine Co., Orange, N.J.

in the quantity column. She enters the value 130, manually tabs, enters the numerator, 3, and the oblique dash, whereupon the computer causes the denominator "8" to be typed and the carriage to be tabbed into the description column. The start signal for the computer is obtained by operator depression of the manual tab or oblique dash key of the typewriter.

As soon as the operator completes the item description, she again manually tabs and the computer is ready to accept entry of the unit price. It is not necessary for the operator to align decimal points. As she makes her entry of unit price, the accumulation of the partial product within the computer occurs as each digit key is depressed. The computer may be programmed equally well to handle decimal or fractional parts of a cent in the price column. After completing the entry of unit price, the operator again tabs, the identifying letters *YD* are automatically typed, indicating that pricing was on a per yard basis, and the product of quantity times unit price is rounded off to the nearest whole cent, stored, and printed out in the gross column. The decimal point is automatically aligned. The typewriter tabs automatically and the machine is ready to accept entry of the discount percentage. The operator enters the discount value and tabs; the discount is applied, the answer is rounded off, accumulated to the subtotal and printed out. An experienced operator can complete that line, including manual typing of the word "Item," in nine seconds. The time required for computation of the net extension after the discount entry has been made and accumulate it to the subtotal is less than six-tenths of a second.

The next two lines of this invoice illustration use the same basic program with a few modifications. All that the operator has to do for the remainder of this invoice is to make the proper selections of programs in sequence, and at the appropriate times enter the number ten to effect a discount of 10 per cent on the subtotal, the 6 and 3 per cent tax rates, and the dollar amount for postage and insurance. All other information, both numeric and alphabetic, is automatically printed out by the computer. If the state or federal tax is a constant percentage, then it too could be automatically typed and computed.

Not indicated in the illustration is the fact that accumulations are being made of total sales, discounts, federal taxes, state taxes, postage and accounts receivable, all of which may be printed out whenever desired by selecting the appropriate program. Any other desired accumulation can be programmed.

A good operator can complete this entire invoice as shown, excluding the date and number line and typing of the name and address, in less than 78 seconds. In a competetive run between the Monrobot IX and an experienced desk-calculator operator, the computer cut almost 70 per cent off the time required by the desk-calculator-typewriter combination to perform the identical job.

It has been found that very little time is required to train an operator. Within one hour a Monrobot IX operator can outproduce a skilled typist and desk-calculator operator team. Her speed and accuracy will continue to rise and reach peak performance in less than a week.

This machine may readily be programmed for virtually any invoicing application, including step-rate utility billing and tax billing.

In the case of utility billing, the only operator entries required are the two meter readings. The quantity being billed and the dollar amount of the billing are both computed and printed within five seconds in the case of a rate structure having three steps.

We have said something about the field of application of the Monrobot IX. We would like to point out some of the features of the computer system design. Fig. 4 shows a block diagram of the computer and indicates the control and information paths. Program control is achieved by means of stepping switches in conjunction with a plug board. The computer is capable of the four common arithmetic operations, decimal shift right and decimal shift left. Other commands exist for automatic typewriter control and alphabetic printout.

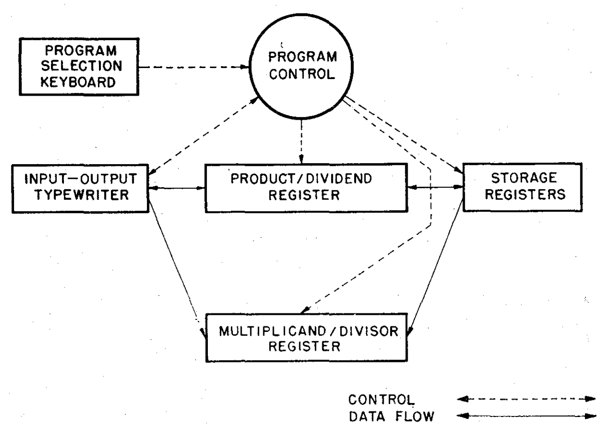


Fig. 4—Monrobot IX system.

Word size is equivalent to 18 decimal digits. Information is coded in straight binary form. Storage registers can be split in any desired manner by proper programming; thus, for example, for some applications the machine can be considered as having 42 six-digit registers, or 28 nine-digit registers.

Fig. 5 is an over-all view of the completed computer less typewriter and program-selection keyboard. The magnetic drum, which can be seen at the front of the assembly, rotates at a modest 2500 rpm. The one information track and the three clock tracks occupy less than one third of the drum surface. The extra width is unused.

The electronic unit is shown expanded as though for servicing. The main circuit section, which is visible at the center of the illustration, and the programming section,

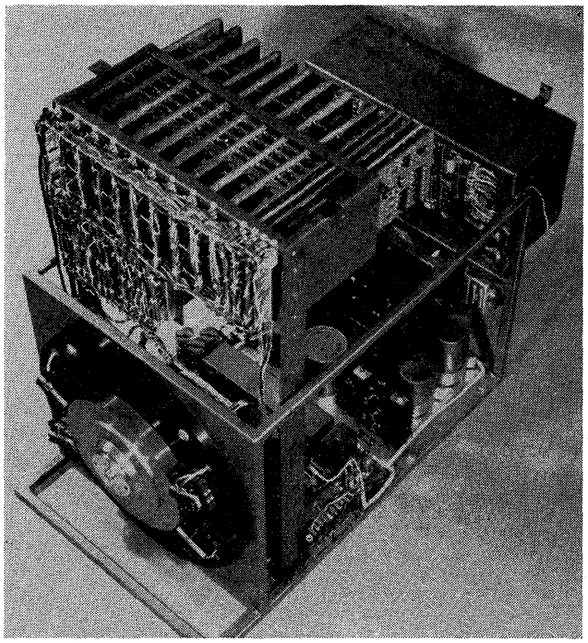


Fig. 5.

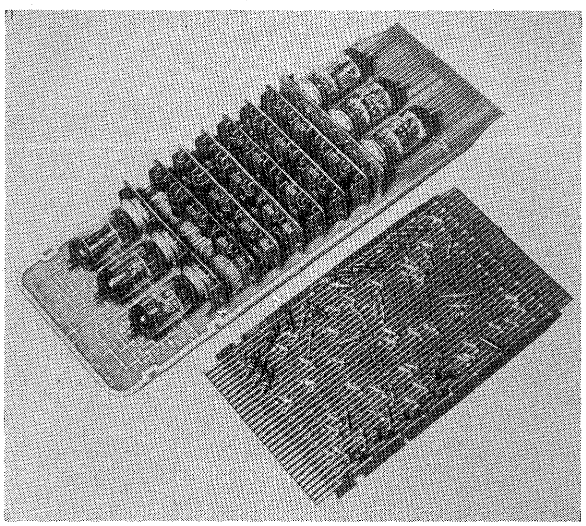


Fig. 6.

visible at the rear, folds into the frame to make a compact unit. The machine is divided into four basic subassemblies to simplify construction and repair.

Our application of printed-wiring techniques is shown in Fig. 6. Each tube board contains three flip-flops, three inverters, and their associated circuits, although other combinations are possible without modification of the basic printed wiring. Each diode card can readily accommodate 60 diodes. There are 9 printed tube circuit boards and 15 diode boards in this machine.

Monrobot IX uses approximately 1000 diodes and 71 tubes, of which 23 are flip-flops and 24 are inverters. Minimization was of tubes rather than diodes since we are able to use a type of diode costing 23 cents each. Logical levels are plus and minus 3 volts.

A four-stage counter, not shown in the block diagram, serves as buffer storage between typewriter and computer and as storage location-selection control. Multiplication and division are by repetitive addition or subtraction. Conversion of a number from pure binary to decimal form for read-out is achieved by dividing that number by the appropriate power of ten. A count of the successful subtractions before the remainder goes negative yields the desired decimal digit. The next lower order decimal digit is obtained by decimally shifting the remainder and repeating the iterative subtraction.

The computer is fast enough to be able to read a number out to the typewriter at the rate of twelve characters per second, which corresponds to the maximum rated speed of the typewriter.

We mentioned that there was only one information track on the magnetic drum. The two fast access loops, the product/dividend register and the multiplicand/divisor register, are interlaced together with the storage registers in such a manner that only one record circuit and one playback circuit are required for the handling of all information. Fast access loops are regenerated continually, whereas storage registers remain untouched except on the occurrence of a store command. Pulse density on the information track is approximately 75 bits per inch, and pulse-repetition rate is about 80 kc.

Negative numbers are not encountered in this machine because the subtract operation has been modified to what has been called the diminish operation. The result of this operation is zero whenever the subtrahend is greater than the minuend. Under any other conditions the operation is a normal subtraction. This feature is particularly useful in handling such problems as step-rate billing and payroll computation in that it eliminates the need for branch programs. The diminish operation and its field of application has recently appeared in the literature.[1]

To summarize, the Monrobot IX is an on-line business machine that is well suited to several basic business functions in which format and computation are repetitive; for example, invoicing. The machine is sufficiently versatile to be able to compute answers involving fractions such as are encountered in lumber billing. The machine can be applied to any currency in the world, including that of the British Sterling. Problem solutions are printed within a fraction of a second to a few seconds after entry of input data, depending upon the particular application.

Training time for an operator is virtually negligible provided the operator commences training with the ability to type.

One of the chief advantages of this machine is to be found in the form of a by-product, that is, accumulations of group totals, such as total accounts receivable, total federal taxes, total quantities, and so forth, which are readily available simply by the push of a button.

[1] R. W. Murphy, "A positive integer arithmetic for data processing." *IBM J. Res. and Dev.;* April, 1957.

Discussion

The answers were provided by W. Burkhart, Chief of Electronics Division, Monroe Calculating Machine Co., Inc., Orange, N.J.

Question: While entering unit price, what does the operator do if she makes an error?

Mr. Burkhart: The method of correcting errors depends on the particular program used. At worst, the error will have affected daily balances, in which an error program would correct it.

Question: What is the memory capacity of your computer?

Mr. Burkhart: The memory capacity is fourteen registers of eighteen decimal digits each. Through programming, these may be split. The average arithmetic speed for multiplication and division is 1.6 seconds for a five-digit multiplier or quotient. Actual addition time is 2.5 milliseconds, while average access time is 10 milliseconds. The actual speed generally is governed by the speed of input and output on the electric typewriter.

Question: How many program steps are available? What arithmetic operations can be performed? Can the machine be used for scientific use?

Mr. Burkhart: Fifty-two program steps are available on each program. Eight basic programs are available, and through the use of extra program selection keys, as many as thirty-two programs are possible. The machine can be used for scientific use if the programs involved are within the capacity of the machine.

Question: With so many VT in a confined area, what means is used to control temperature?

Mr. Burkhart: There are only seventy-eight tubes in the machine, and the total power dissipation is about 600 watts, so that a small fan suffices to keep the interior cool. Since the vacuum tubes themselves are near unwired portions of the printed circuits, there has been no component deterioration due to heat.

Question: What provisions are made by circuit design or computer logic to prevent errors in computation to be printed and invalidate correct work already processed?

Mr. Burkhart: There is no means for preventing the printing of errors due to operator mistake or computation error.

A Self-Checking System for High-Speed Transmission of Magnetic-Tape Digital Data

E. J. CASEY†

SEVERAL years ago it became evident that high-speed communications facilities were required as part of many future data-processing systems. The need had appeared in the planning phases of several systems and certainly, as the speeds of computer operation increased, it would be desirable to centralize computing systems to take maximum advantage of them. Situations indicating its need are:

1) Real time surveillance and control systems of military significance.
2) Faster computers which are able to do the total data processing for a large business so the data from the many sources must be brought to the computer site.
3) The necessity for prompt sending of data to a central location to permit over-all control, even if the development of small internally programmed computers permits many geographically separated computer installations.
4) There is a gross difference between the new and the old data-processing speeds.

Important practical considerations in the selection of a data-transmission facility as part of a data-processing system are that the media to hold the transmitted and received data, and its code and format must be compatible with the rest of the data-processing system. Although an "on-line" data code and format significantly different than that on the input and output media are possible, the present state of the art indicates that those of the data-processing media be retained with minimum alteration for the data transmission.

The quantity of data to be transmitted in future systems, when the facility for rapid, accurate data transmission is widely available, naturally is unknown now. The situation may be compared to that in earth moving. The number of millions of yards of dirt that needed to be moved when shovels and wheelbarrows were the only tools was much different than the number that "needs to be moved" today, now that huge power shovels, long conveyor belts, and large tractor-scrapers are available. Even today, data transfer by some concerns involves 5 to 30 million characters per day; while this accomplishment with techniques used may be likened to the *tour de force* of the Egyptian in pyramid building, it does indicate that ten to one hundred times as much is not out of line for future needs.

The "state of the art" digital transmission speeds may be compared with theoretically possible data rates by noting that a fairly low quality phone channel of 1700 cps bandwidth and 22-db signal-to-noise ratio should, by the $B = W \log_2 (1 + S/N)$ formula yield an error-free transmission rate of approximately 12,000 bits per second, but present practicabilities offer more like 750 bits per second. Compared to teletype and telegraph service of 30 informa-

† Remington Rand Univac, St. Paul, Minn.

tion bits per second in 60-wpm service, this is a significant increase (25x). With a 7-bit code, 750 bits per second could yield 80 to 100 characters per second. A 30 million character-per-day transmission load, at 100 characters per second, would need this kind of line for 300,000 seconds per day, or three of these lines would be needed to transmit this much information per day. A large industrial complex may have only 3000 to 300,000 characters per day to send from each location, so many lines could be involved but for only a few hours per day.

One of the most important aspects of digital-data transmission is the accuracy—the accuracy needed by the user and the accuracy given by the digital-data-transmission service. The present indications concerning the phone-line error characteristic is that on the average, between one in 10,000 bits to one in 100,000 bits will be in error. These data are derived mostly from teletype and telegraph experiences. The redundancy planned for digital-data communications and the extensive "intelligence" built into the terminal equipment will detect these errors and can ask for a repeat of the message; so even if the error rate on the line should remain as indicated, the error rate in the records submitted to the user will be dramatically less. The checking features of the digital-data-transmission systems can be evaluated thoroughly only after detailed analysis involving as yet unavailable detail error characteristics of the line and modem equipment, or by months of actual "on-line" testing. Then, with good records kept of those errors found by other checking, as with a large computer, and traced to the records submitted by the digital-data-transmission equipment to the data-processing center, the accuracy can be evaluated. To indicate in terms of present-day media, punched cards and magnetic tape, the accuracies expected in checked digital data transmission, one hole punched wrong in one of 80,000 eighty or ninety-column cards and not caught by the checking facilities, would yield a probability of error of one in 40,000,000. This quantity corresponds to sending eight cards per minute, twenty-four hours per day for one week. This is about 1/1000 of the rate indicated as the error rate on the line. Similarly, for a magnetic tape-transmission system at ninety characters per second for 168 hours (one week), one bit in error in the submitted record would indicate a probability of undetected error of one in 400,000,000 or 1/10,000 the rate indicated above as the "on-line" error rate.

The MTM Transrecorder, which transmits data from one magnetic tape via voice-band facilities to another, is an example of a facility for providing high-speed, accurate, automatic, digital-data transmission. The source media can be any Univac tape from 200-foot perfect tape reels to 1500 or 2400-foot reels with random bad areas or splices. Data are recorded in C10 code, in blockette or high-speed printer format on both source and receiving tapes, and so is compatible with other system equipment.

The philosophy of error correction in the Transrecorder is to introduce into the transmitted message sufficient redundancy to detect errors and have the receiver check the received message, and if it is erroneous, request a retransmission. This is accomplished by:

1) Dividing the total message of 100 to 2000 blocks of information into submessages each consisting of an integral number—1 to 16—of blocks of information. A block consists of 720 characters which in the high-speed printer format, used with the Transrecorder, is divided into six blockettes of 120 characters each.
2) Sending one submessage from transmitter to receiver and awaiting reply.
3) Checking the incoming data at the receiver during submessage reception and checking the recorded data corresponding to this submessage to insure that they satisfy the proper checks, which are:
   - Character parity.
   - Correct character count/blockette.
   - Long parity check during reception.
   - Proper number of blockettes/submessage.
4) Receiver sending a reply to the transmitter indicating that the last submessage was received and recorded correctly, or that it is to be retransmitted.
5) Continue sending submessages until the end of the total message.

This is error correction by error detection at the receiver and then use of feedback from receiver to transmitter to advise the transmitter that the data have either been received satisfactorily, or was received with errors and is to be retransmitted. It is to be differentiated from error correction by error correcting codes as discussed by Shannon, Fano, and others, which would permit reconstruction of the information from mutilated incoming signals without the necessity for reverse transmission.

From the time the operators initiate transmission until end of rewind, at both transmitting and receiving locations, the operation is automatic.

Some of the automatic features of the Transrecorder are as follows:

1) After mounting the reels of tape on the tape handlers and initiating operation, the control automatically advances both source and receiving reels far enough to insure that good magnetic tape will be under the heads. The receiving station simultaneously erases the leader and tape which feature permits reuse of tapes without requiring pre-erasure. The transmitting tape advances ten feet prior to initiating "read" to avoid noise due to clips and leader-tape junctions, and the receiving advances fifteen feet before recording to permit adequate tolerances between the various tape handler-control combinations of a data-processing system.
2) The data read from the tape at the transmitting station are checked for character parity and for 120 characters per blockette. This data is stored in a 120-character magnetic core buffer.

3) In transmitting the data from the buffer to the modem and line, the character parity is again checked and a "horizontal parity" character, the 121st character sent for each blockette of information, is generated to indicate the sum modulo 2 of the "ones" for each information level.
4) The control unit sends before each submessage, a submessage preamble consisting of a train of alternate one-zero pulses. This alerts the receiver that a submessage is to be sent, prepares the phone line for propagation in the transmitter to receiver direction, *i.e.,* reverses possible echo suppressors and stabilizes compandors enroute, and establishes proper receiving-clock phasing. This receiving clock establishes what we might call "bit synchronization."
5) The preamble is followed by one or more special characters which effect a "character synchronization," which lets the receiving control unit know the first bit of each of the incoming serialized characters. This special character both provides this "character synchronization" and a time-buffering action which permits proper fitting of the tape-unit advance and bad-area traversal times to the required bit and character synchronization during the submessage. Each "on-line" blockette of 121 characters is preceded and succeeded by a special character to permit the receiving station to effect a character count check on the incoming data. It also permits a long parity character to be generated from the first 120 characters of the incoming blockette. This is then compared with the 121st incoming character.
6) At the end of each submessage or block group, which may be chosen by a switch to be from about 5000 information bits to about 80,000 information bits long, the transmitting station stops transmitting and awaits a reply from the receiving station. If all the information sent since the last "answer back" was received *and recorded* correctly, the receiving station sends back a train of alternate 1-0 signals long enough to reverse the direction of propagation on the line and to register the "resume" order in the transmitting site's control circuits. If errors were detected in the submessage as it was received, or if after three read tries the recorded submessage at the receiving site fails to check properly, the receiving MTU repositions the receiving tape to the beginning of the submessage improperly received or recorded, and sends back a train of alternate 2 "1" 's—2 "0" 's. This is sent long enough to stabilize the line transmission in the reverse direction and register the "retransmit" signal in the control circuitry. If the answer back was "resume," the transmitter proceeds to send the next submessage, if "retransmit," it repositions to the beginning of the last submessage and proceeds to retransmit it.
7) The "end of message" indication on the transmitting tape is five feet of erased tape beyond the last blockette of the message. The transmitting control unit senses this and after insuring that the last information sent was received and recorded correctly at the receiving site, it sends an "end of message" signal. Then the receiving and transmitting MTU's rewind the tapes onto the original reels and indicate the completed message condition to the attendant.

The Transrecorder consists of basically three units, a Control Unit (CU), a Magnetic Tape Unit (MTU), and a modulator-demodulator (Modem) to take the "square-wave" voltages from the CU and apply signals to the line at the transmitting site, and to receive the signals from the line and change to "digital signal" form at the receiving end. The Univac Modem was designed for use on customer owned lines and to permit gaining experience and insuring compatibility between the facility "Modem" service and the CU of the Transrecorder. When the Transrecorder is used on public communications facilities, the facility's modem will be in a separate cabinet to the left of the CU. Any Transrecorder installation can be used to transmit data or to receive by switch selection on the control panel at the top of the CU so an installation at a regional office or remote factory can first submit a reel of tape data to the Data Center and later switch to the receive mode and receive data, instructions, shipping schedules, etc., from the center.

With the exception of the Modem, the total CU is transistorized. The MTU involves both tube circuitry for tape-transport control and reading and writing and transistor circuitry for logical control functions.

The maintenance requirements due to plug-ins and components have been very low in installations using this type of construction; these plug-in circuit designs are used also in the perforated paper tape to magnetic tape converter, in the magnetic tape to perforated paper tape converter, and in an a/d converter and recorder. Provision is made for convenient preventative maintenance by altering supply voltages to selected racks of machine in the test mode and observing limits. Supply variations of ± 25 per cent and more on entire racks are permissible when all components are within limits, and the maintenance-panel indicators, the selective alteration of supply voltages, and the plug-in construction allows detailed analysis and prompt correction of a fault if it occurs during scheduled operation.

The Transrecorder operates on 115-volt 60-cps phase power, the total average power requirements being less than 3 kw, and it is capable of operating in 90°F ambient temperatures.

The author wishes to credit C. W. Fritze, B. L. Meyer, and R. Goossens with the majority of the control and logical features of the Transrecorder and the continuing effort to bring this development to its present stage.

Discussion

Question: What percentage of the total number of bits transmitted are redundancy bits for error detection?

Answer: In the Univac data automation system, each character is composed of six information bits and one odd parity bit. This error checking feature is retained in the high-speed digital data transmission system where it is often referred to as the "vertical parity" check. Hence in each block of 720 characters, 720 of the 720 x 7 bits are "vertical parity" redundancy bits. To each blockette of information, an additional "horizontal parity" character is added, giving 42 additional bits per block. Hence, about 15 per cent of the transmission consists of error checking bits. In addition, timing bits and special spacing characters introduced for purposes other than error checking are checked against the *a priori* knowledge that they should be present at particular intervals, and so also serve as "error checking" bits. But, this latter feature is a sort of bonus, since their primary purposes are for timing and inter-blockette and interblock spacing.

Question: Will the equipment handle both metal and plastic tape?

Answer: Yes.

Question: Is the equipment now available?

Answer: The equipment is undergoing laboratory and system testing and is not available for immediate delivery. For delivery and similar information, please contact the Communication Department, Remington Rand, 315 Fourth Ave., New York, N. Y.

Question: Will the equipment transmit data over standard telephone lines?

Answer: Yes, and this was an important consideration in the design. The goal was that any phone line over which satisfactory voice communication could be obtained should be suitable for digital-data transmission.

Question: Can the blockette generally be arranged by computers of other manufacturers?

Answer: This question has several ramifications. The logical structure of a blockette, *i.e.,* 120 characters per group, odd parity characters, particular bits significance in each character, etc., could be prepared by any computer. The problem would arise in the magnetic head structure, writing densities, writing mmf, read-back signals, track orientations with respect to tape edge, head gap staggering for various tracks, etc. Hence for most practical purposes, the Transrecorder could be expected to accept only tapes prepared on Univac equipment.

Question: What is the form of the transmission on the line, the signal representations of the "0"'s and "1"'s?

Answer: This probably will vary with each different manufacturer's Modem, or each different communication company. The Univac Modem uses 100 per cent amplitude modulation of a tone carrier of about 1500 cps, "1"'s represented as full amplitude and "0"'s as zero amplitude signal. The signal levels on the line can be varied but are generally considered as about 0 dbm into a 600-ohm balanced phone line. A binary or two-state FM system is being developed by others, with one tone near the lower edge of the phone channel pass band for one digital state and a tone near the upper edge of the band for the other; the tones are put on the line one at a time. The binary FM appears to have some advantages in increased S/N, for ease of implementation of gain control, and for bit detection implementation, since a comparison should be more reliably made between the power in the two tones on a variable loss link than can the determination of whether the incoming power level on such a line exceeds a preset level. Other systems use mutiple tones, or different phases of a "continuous" tone, all presently known ones having the tones in the audio phone band at the modulator output and at the demodulator input. In the normal trunking facilities where a given channel may be subjected to frequency or time multiplexing techniques, the power spectrum and type of modulation may be very different at enroute points than at the Modem terminals. It should be mentioned that the Transrecorder (less Univac Modem which is furnished for use only where a communications company facility, with its own Modems, is not available) is not interested in the "on-line" signal representation, if the Transrecorder-Modem interconnection signals are appropriate.

Question: What reasons led to the selection of a serial rather than parallel mode of line transmission of the bits comprising the characters?

Answer: Primarily, this decision was based on ease and simplicity of instrumentation and the consequent economy. Also, with the normal type frequency separation techniques, the percentage loss of total useful bandwidth due to "guard bands" makes for less efficient bandwidth utilization in multiple tone systems, and the ratio of peak power to average or rms power on the channel increases when more than one tone at a time is impressed on the channel. Since generally, actual channels have a peak power limitation as well as an rms power limitation, the rms signal that can be impressed is higher for single tone modulation than for multiple tone. The development of new frequency separation techniques and the increase in duration of each signal interval as the number of simultaneous bits per signal interval is increased, which increase minimizes the effects of certain kinds of noise, suggests a multiple tone system. But, the economy of Modem implementation seems still to be in favor of serial transmission.

Question: On a low-grade phone circuit of about 1200-cps bandwidth, did you mean that theoretically this should handle 12,000 bits/second?

Answer: The example given was for a 1700-cps channel bandwidth having a 22-db S/N ratio ($S = 159.N$ and the transmission channel degraded only by additive noise) and this facility then should, per Shannon's formula $B = W \log_2 (1 + S/N)$, give a "long time average" bit rate of in excess of 12,000 bits/second. The actual accomplishment of this rate of transmission on the above phone channel awaits the development of considerably more sophisticated methods than we have at present.

Question: What is the transmission medium employed?

Answer: When the Modem is furnished by the communication facility, the medium is of no interest to the Transrecorder proper; but in normal installations it is expected that it will consist of a 2 or 4 wire, one-half or full duplex phone channel facility. When the Univac Modem is employed, a 2 or 4 wire, 600-ohm nominal characteristic impedance, preferably balanced to ground, with inputs to the phone line from the Modem of between +3 and −6 dbm, and outputs from the line to the Modem of between +3 dbm and approximately −25 dbm, is required. After the 2 or 4 wire lines leave the vicinity of the Modem, especially if long-distance transmission is involved, it is expected that frequency or time multiplexing techniques will be used, and the channel may go on open wire lines, coaxial cables, microwave links, or similar trunking facilities, but will reappear on 2 or 4 wire lines in the vicinity of the Univac Modem at the remote location.

Question: Are blocks and blockettes so recorded on the sending tape or does the control unit do the subdivision and control the tape feed?

Answer: In the system implemented, the information on the magnetic tape at the sending end is divided into blocks and blockettes (so-called high-speed printer format) and is reproduced in the same fashion on the receiving end tape.

Question: What is the bit rate over the line in the existing system?

Answer: The existing equipment is working at 750 bits/second and at 800 bits/second. The change from one to the other involves a change in transmitting clock generator, and receiving clock recovery circuit plug-ins. Since the maximum transmission speeds are so intimately associated with the transmission channels, it is planned that the transmitting Modem will establish the bit rate by furnishing a "transmitting clock" signal to the Transrecorder and the receiving Modem will recover a "receiving clock" from the incoming signal and supply it, with the received data, to the receiving Transrecorder. As accurate transmission at higher speeds is accomplished, due either to more sophisticated methods on given channels or the installation of wider band facilities, the Transrecorder then can very conveniently utilize the higher speeds.

Question: Is it necessary to have two sets of Modems on long-distance transmission, one furnished by Univac and the other by the telephone company?

Answer: No. The Univac Modem is a self-contained, panel-mounted unit easily removed from the control unit in situations where the communications facility furnishes the Modem and associated "clocks." In the continental U.S. it is anticipated that the great majority of the installations will not have the Univac Modem.

Question: What is the information transfer rate? How much time is required to transmit, check and rewind, say, 100 blocks of 720 characters each?

Answer: This information transfer rate depends on many factors among which are "on-line" bit rate, tape leader length, tape handler speeds, writing density, on line error rate, number of blocks per block group, number and length of bad spots, "end of message" designation, line reversal time, etc. An illustrative example representative within 20 per cent for all except the most unusual cases would be that with an 800 bits/second "on line" bit rate and a fair quality line, hence few requirements for retransmission. It would require about 7.8 seconds per block or 780 seconds plus approximately 20 seconds rewind time $\cong$ 800 seconds for 100 blocks or for about 500,000 bits of information.

Question: How long does the receiving unit require to reread the blockette and compare to stored information, and is the data transmission stopped during this time?

Answer: The information is read and checked at the receiving end in block groups rather than per blockette, so all the information sent during a block group is checked in one continuous operation. The information recorded on the tape is checked against *a priori* known criteria rather than against stored information. The check consists of insuring that each character of the block group has satisfactory character parity, that there are 120 characters per blockette and 6 blockettes per block. The time required to check a block group then depends on the number of blocks per block group. As an example, if 4 blocks per block group are chosen and a bit rate of 800 bits/second is used, the time to transmit and record the block group is about 32 seconds; whereas the time required to check the block group recorded at the receiving end and reposition the tape would be about 1.75 seconds, or about 5 per cent of the time is used for checking. During this checking time, transmission from the transmitting end is stopped pending receipt of a resume or retransmit signal from the receiving end.

Question: Why do you use five feet of blank tape to detect the end of message rather than use a specific code?

Answer: With the exception of the odd parity bit redundancy deliberately introduced for error detection, the Univac code is a very low redundancy, or highly efficient, coding scheme, so it is not possible to use a specific single character code to detect reliably a mark or signal of such important logical consequence as an "end of message" signal. Even to limit detection of, and action on, such a single character code to the intervals such as end of a block of information known *a priori* possibly to contain it is not sufficiently reliable for so important a logical operation. Hence, it would require instead an entire blockette or block of a very unusual code pattern to reliably establish an "end of message" signal at the transmiting tape, and then either its accurate transmission to the receiving end, or the sending of a less redundant signal to the receiver and the regeneration and recording there of a similar coded blockette or block. Further, such implementation would require considerably more instrumentation at both Transrecorders, as well as imposing such an "end of message" coding on all source data devices. The relative case of implementation and reliability of generation and detection of an "end of message" indication with a short erased section after the last useful data resulted in this choice.

Question: To what extent does the time required to reverse line echo suppressors affect transmission time; for example, percentage increase per packet of data?

Answer: This relationship also is a rather complex one in the general case. The higher the bit rate and the fewer the number of blocks per block group, the larger is the percentage of the total time assigned to echo suppressor stabilization (line reversal). Also in the "answer back" mode, two line reversal times per block group are involved. If a one block per block group mode is selected, and 800 bits/second is the "on-line" data rate, approximately 8 seconds are required to transmit the data. The maximum echo suppressor stabilization may approach 0.3 second or 0.6 second for the block group cycle, so a 6 to 8 per cent time increase may be involved. At significantly higher bit rates this effect would be more important.

Communication between Remotely Located Digital Computers

G. F. GRONDIN† AND F. P. FORBATH†

Introduction

THE usefulness of complex data-processing centers can be increased by rapid and accurate communication between remote locations. The problems encountered in the data transfer are not new to the communicator; however, the familiar characteristics of the communications link assume increased significance when the digital nature of the data, the high information rate and the required degree of accuracy are considered. The stringent requirements demand that the communication system place special emphasis on providing maximum utilization of channel capacity, on minimizing the raw error rate, and on using special coding techniques to achieve unprecedented error detection.

The reliability achieved even by near-optimum communications systems falls short of the accuracy demanded. In spite of the communication-link limitations, the desired degree of accuracy is attainable by error-detection techniques and data repetition. The burden of error control as well as the task of providing compatibility between the various data sources and the transmission equipment falls on special converters (input-output devices). Their design is dominated as much by the inherent limitations and peculiarities of the communication system as by the characteristics of the data source. One such special converter

† Collins Radio Co., Burbank, Calif.

intended for high-speed punched card transmission over voice-quality circuits is described, and it illustrates how a particular combination of parameters meets this specific requirement.

General

Since common wire-line facilities represent a vast available network, economical data transmission depends on efficient utilization of the voice channel. Unlike speech, the inherent redundancy of digital data is extremely low and a single error may cause misinterpretation. Therefore, three important properties the transmission equipment must have are 1) efficient utilization of bandwidth, 2) minimum binary error rate in presence of noise, and 3) low undetected error probability. The first two are related to the binary communication system while the third is achieved by redundancy and coding techniques.

The system's basic error rate or susceptibility affects the information rate. It determines the percentage of data which needs to be retransmitted or corrected and the amount of redundancy that must be added to detect erroneous data. Although theoretically any desired accuracy can be attained, the complexity and cost of doing so are directly related to this factor, and may be prohibitive.

Any error-detection method should meet system requirements with minimum redundancy, simplicity of coding, and freedom from systematic errors.

Kineplex

A communication system known commercially as Kineplex, which uses "predicted wave" techniques, is particularly well suited to digital data transmission. Its theory of operation and performance characteristics over radio circuits have been described in several papers.[1–3]

Kineplex lends itself to frequency, time and phase multiplexing for spectrum conservation; near zero crosstalk between adjacent channels is effected by synchronous keying and sampling of infinite-Q detection filters. The detection method provides perfect integration of the signal over the pulse duration while noise which lacks phase coherence is increased only on a rms basis. Phase-shift coding permits two independent bits of information to be encoded on each pulse by resolution of phase into quadrature components. Thus, predicted wave detection yields a gain in signal to noise ratio accompanied by a lowering of usable signal threshold and a narrowing of the required bandwidth

Wire-Line Application

The above techniques have been applied in the design of the TE-206 Kineplex Data System (Fig. 1), a general-purpose, high-speed binary data transmission system for voice quality circuits. It features efficient bandwidth utilization, low susceptibility to noise, adaptability to use with a wide variety of inputs, and parallel data transmission. Its proven superior performance is derived from the phase-shift keying and the ideal detection techniques summarized and referenced above.

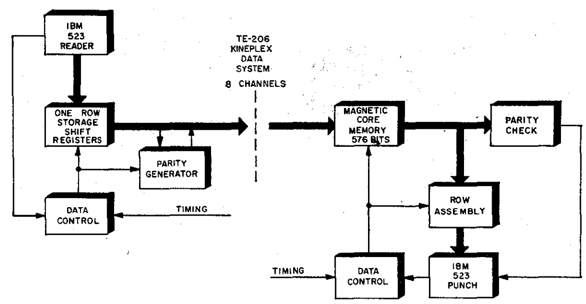


Fig. 1.

Specifically, it accommodates 2400 bits per second within a 2200-cycle minimum bandwidth. It provides eight parallel input channels and can therefore accept 8-bit characters at a rate of 300 per second. Each of the four tones, spaced 440 cycles apart, carries information from two input channels; the actual tone frequencies are determined by the line characteristic. To accommodate a majority of known facilities, tone frequencies of 935 cps, 1375 cps, 1815 cps and 2255 cps were selected for the TE-206. The 3.3-msec pulse length was selected to be several times longer than the expected duration of impulse noise, longer than the incremental delay distortion across the band of unequalized voice circuits, and yet short enough to provide frequency-error tolerance for carrier systems.

Since data can be handled in parallel by the transmission channels, the necessity of parallel to series conversion is avoided, and the cost and complexity of associated converters are reduced.

Kinecard (Fig. 2)

The wide use of the punched card as a versatile and reliable source document has produced the need to duplicate its information content at remote locations. The Kinecard converter system permits continuous and accurate transmission of scientific and business data from punched cards over common voice facilities. It illustrates how the various design parameters can be combined to maximize performance within the bounds of economic feasibility.

Punched cards are processed at a nominal rate of 100 cards per minute. This makes possible on-line use of IBM 523 Gang Summary Punches for local reading and remote punching of cards. Data are accepted from the card reader, indexing markers and check characters are added, and the information is presented as synchronous 8-bit characters suitable for Kineplex transmission equipment; at the remote end, the data are stored until required by the punch, its validity is checked, cards are punched and erroneous

[1] M. L. Doelz, E. T. Heald, and D. L. Martin, "Binary data transmission technique for linear systems," Proc. IRE, vol. 45, pp. 656-661; May, 1957.
[2] A. A. Collins and M. L. Doelz, "Predicted Wave Signalling," Collins Radio Co., Burbank, Calif.; June 22, 1955.
[3] R. R. Mosier and R. G. Claybaugh, "Kineplex, a bandwidth efficient binary transmission system," *AIEE Trans.*, to be published.

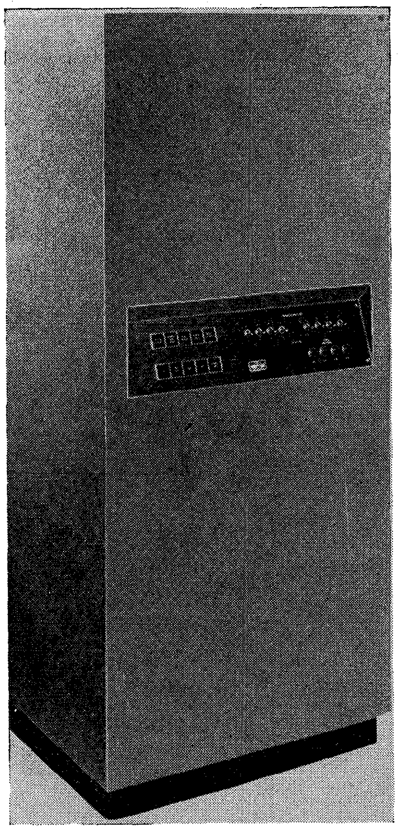


Fig. 2.

ones are offset. Operation is continuous without provision for answer-back or automatic fills.

Other than establishing a communication link prior to a card run, no special operational procedures are needed. Interlocks prevent initiation of a run unless Kineplex, Kinecard and IBM punch are ready. The converter controls parallel the punch controls and the system can be operated from either.

Design Considerations

It may be fairly stated that if cost and complexity are not considered, just about any combination of operating features may be provided. Features which were considered in Kinecard were code translation, format control, card verification, automatic error correction, and interchangeability of terminal devices. In its present form, Kinecard is a special-purpose device having reasonable efficiency and adequate error detection for wire-line applications.

The punched-card code contains twelve elements per card column to accommodate about 50 alphabetical, numerical and special characters. The 12-bit coding could be translated to a 6-bit code thus doubling the information rate of the transmission system. However, most card readers present the data row by row, 80 bits at a time, such that characters represented by each column can not be fully interpreted until a whole card has been read and stored. Transmitting the card as on a row-by-row basis eliminates extensive storage and code-translating circuitry.

The card reading and punching operations are not verified even though 2 or 3 machine errors per 10,000 cards are possible. Since these errors are not introduced nor aggravated by the communication equipment, their detection should be by routine accounting-type cross checks.

Transmission system detected bit error rate	10^{-3}	10^{-5}
Per cent cards in error (off-set)	60%	1%
Number of erroneous cards undetected	1 in 20	1 in 200,000

Fig. 3.

The error-detection scheme takes into account the nature of the noise over wire lines and the related error probabilities introduced by the Kineplex equipment in deriving its phase reference. The impulse noise which may affect all channels and the possibility of occurrence of adjacent bit errors are countered by deriving two separate lateral parity-check bits on each channel.

Assuming random-error distribution, the number of erroneous cards, detected and undetected, is tabulated as a function of system error probability (Fig. 3).

Operational tests are planned to determine the effectiveness of the error detection. If additional protection is required there is ample time between each card transmission to add more check bits.

Operation

Reference to the transmitted card format (Fig. 4) will help clarify operation of the converter.

Several control signals are derived from the card reader to indicate the start of the card-reading cycle and to identify each row of information.

A reader-card start impulse initiates the emission of several "start of card" characters which serve to index the remote punch-control equipment.

As each row of information becomes available from the reader it is transferred into eight 10-bit shift registers. A row-start character precedes each row-transmission cycle which consists of reading out all eight registers in parallel with synchronous pulses derived from Kineplex. The register is emptied before the next row is presented by the card reader.

At the end of the twelfth row the parity checks are inserted. Two parity-check characters are obtained from alternate data characters; two bits per row are derived. Each bit is formed by adding the number of punches and complementing to an even multiple of two. Fig. 5 is a simplified block diagram of the converter.

Since the reader undergoes speed variations, synchronization is achieved by inserting no-information characters between rows and between cards as required.

At the receiving terminal, card- and row-start markers are identified and they control the assembly of the incoming data into a magnetic core memory. The memory ca-

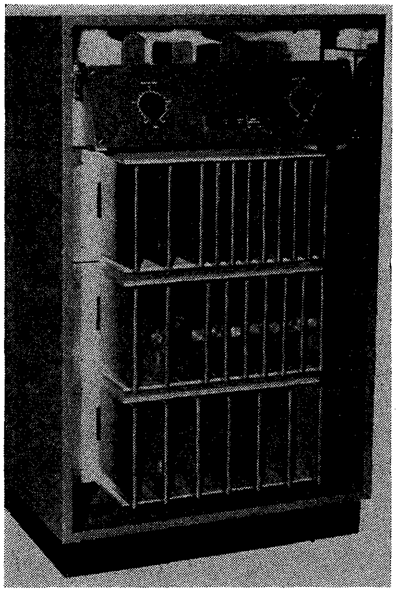


Fig. 4.

pacity is 72 characters. The punch operation is started when sufficient data has been stored in the memory to assure that the punch will not overtake the incoming information even if it is running at its fastest tolerance of 107 cards per minute.

An 80-bit storage register assembles a full row from the memory. The punched card is reproduced row by row. If the parity checks indicate an error, the punched card is offset in the stacker.

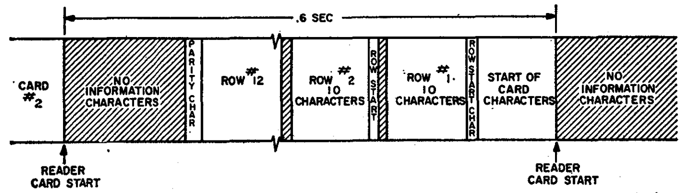


Fig. 5.

The converter is completely transistorized including the punch-magnet drivers. Construction is modular and consists of printed-circuit cards many of which are identical. A complete transmit and receive terminal is contained in a 5-½ foot cabinet.

Conclusions

Reliable communication between digital computers and on-line use of business machines utilizing wire-line and radio facilities can be accomplished with efficient data-transmission systems and special-purpose converters.

Economic considerations, clarifications of use requirements, lack of common language, format and data-rate standardization, incompatability of equipments and operational inexperience are some of the limiting factors.

The Kineplex data-transmission system provides a common signaling method for use with a variety of existing and future input devices which want to take advantage of available voice channels.

The Kinecard converter increases the speed of transmission of punched-card data to the usual operating speed of punching machines thereby permitting on-line use.

Communication Switching Systems as Real-Time Computers

A. E. JOEL†

Introduction—The Nature of Communication Service

Automatic communication switching systems were the first practical and mass-produced data processing systems. Initially, they were designed with electromechanical elements, such as selector switches and relays. However, as will be covered in more detail in another paper,[1] electronic data processing techniques are rapidly being applied to these systems.

† Bell Telephone Labs., Inc., Whippany, N.J.

[1] R. W. Ketchledge, "An introduction to the Bell System's first electronic switching office," this issue, p. 204.

Not only were automatic communication systems the first mass-produced data processing systems, they were also the first "real-time" computers. What makes them "real-time" computers? They must serve on demand of the customers "quickly when wanted" and "all of the time." A service request by a communications customer is a perishable commodity and the longer the delay in serving, the more likely the chance that the request will be withdrawn. Therefore, in communication switching the time between a service request and its fulfillment must be kept small, if we are to provide the service for which we are granted public franchise and to capitalize to the maximum extent by making the supply of service approach the demand as closely as possible. Public reaction to inability to serve is an important factor in providing service when requested.

If service cannot be offered when requested, there is either abandonment or delay. The amount of delay, in this case, may determine whether the system is truly real time or not. Real time is relative. A one-second delay on a call which lasts for 180 seconds is not excessive. Perhaps a 10-second delay is excessive on a short distance call but not on a longer distance call.

Equipment failures are inevitable in a system which must serve "all the time." Even with only one source demanding service on request, duplicate or multiple facilities are provided to greatly reduce the chance of expected failure to serve on demand. Improved maintenance facilities and device reliability also reduce outage time of equipment which has failed and thereby reduce the amount of multiple facilities required to maintain service.

In the telephone business we also speak of service orders when someone requests telephone service, that is, installation of an instrument. In so-called "common carrier" type of communication system, the service must be offered to all who desire it. The ability to add new telephones on demand imposes many problems on this real-time system, since in "real time" there is a limit to the number of telephones and calls therefrom which a system may serve with a given amount of equipment. Each new telephone served will require additional equipment capacity in this real-time system.

Therefore, we have many sources of demand. Each source originates requests more or less at random and independent of others, but over a long period of time patterns do exist which influence the engineering of these systems. Another paper[2] covers these traffic aspects in considerably more detail. Each new source also becomes a "sink" since communication traffic is usually two-way and, therefore, facilities to interconnect all sources must be provided. One of the differences between a communications system and a computer using the same data processing techniques is this necessity for interconnecting sources as well as processing data for each. Many of the real-time computers have a plurality of sources, but these sources seldom need to be interconnected or interrelated directly through the computer.

Signaling

In recent years we have tended to separate the switching problem into two parts and deal with each separately. These two parts are: 1) the receipt and recording of service requests and the control of the establishment of the desired connections, and 2) the arrangement of switching devices to permit any desired pattern of interconnection. The first part is a digital data processing problem. Thinking of a communication switching system as a real-time system brings traffic and other aspects which have long been with us in this industry to the attention of those designing complex digital computers. The second part of the system is an inherent characteristic of most communication switching systems and will not be dealt with here.[3] The analogy between telephone switching systems and computers has been covered previously[4,5] and will be assumed as background for this paper.

The devices with which these problems were first solved were several orders of magnitude slower than those presently being employed in digital computers or contemplated for electronic switching. The earlier and most familiar dial telephone systems, such as the step-by-step system, combine these two parts inexorably. Nevertheless, their study in the light of the new electronic techniques teaches us many things about the nature of the real time problem. For one thing, the speed of the devices in the central office was fast enough to keep up with the maximum rate at which human beings could spontaneously actuate call devices, such as dials and keysets, to place information directly into the system.

Assuming each channel receives information from only one person, the rate at which information enters the system is thus limited by his sending rate. Common carrier systems usually involve very large numbers of sources; there are at present some 62,000,000 telephones in the United States. To select one of a large number in a single operation would require a very large calling device, such as a dial with 62,000,000 holes, and tremendous dexterity on the part of the user. It would also require complex or time-consuming signal generating and receiving equipment. Considerable time might also be required to select the desired telephone. A practical way to accomplish a selection among a large number is to use a sequence of digits to sift through all the possibilities. This means that several digits must be sent and a compromise made as to the number base of the system, the complexity of the calling device for encoding the signal, and the number of digits to be transmitted. It is well known that the base 10 has become universal because it is most readily used by the public and the number of digits, up to recently, have not been excessive. The calling device is simple—a 10-hole dial or 10-button keyset.

Here human engineering comes into the picture. Obviously, the base 10 is used because it is best known. However, one of the early methods was to use the first letters of central office names arranged in base 10 as a mnemonic aid. Recent psychological studies indicate that such an aid may not be as useful and helpful as we once believed. In any event, such aids should not be ignored since they may provide better service by reducing dialing errors. Reduction of dialing errors allows the equipment to operate

[2] J. A. Bader, "Traffic aspects of communications switching systems," this issue, p. 208.

[3] C. Y. Lee, "Analysis of switching networks," *Bell Sys. Tech. J.*, vol. 34, pp. 1287-1315; November, 1955.

[4] W. D. Lewis, "Electronic computers and telephone switching," *Bell Labs. Rec.*, vol. 32, pp. 321-325; September, 1954.

[5] W. D. Lewis, "Electronic computers and telephone switching," PROC. IRE, vol. 41, pp. 1242-1244; October, 1953.

more efficiently by eliminating waste usage. Also, customers dial more rapidly, thereby reducing holding time of the call receiving circuit.

With the call information broken up into a number of digits it is not always possible for the signal receiving equipment in the switching machine to act on one digit at a time. Most systems require that a number of digits, such as the first 3 which represent the central office name, be received, before any action may be taken. This introduces the need for storage of digits in the central office until sufficient digits of the number are available for processing. These receiving and storage circuits must be provided in sufficient quantity to care for the maximum number of sources sending simultaneously. Interconnecting means must be provided to associate these circuits with the calling lines. Circuits which are called "registers," "senders," or "directors" are used only during the period when the customer is sending selection information into the machine. These circuits may be dropped from the line after this phase of each call to be reused by other customers. In this way they are provided and used more efficiently than if one stayed associated with each call until its completion. But still one of these call receiving and storage circuits is utilized for each call being dialed. Circuits of this type may serve both operators and customers, or separate groups may be provided for each class of input. This is a designer's choice which is determined by the degree of difference in the logic between a register used for operator calls and one used by customers. For example, if operators use 10-button keysets instead of dials the difference is sufficient to warrant a separate group of registers to work with keysets rather than providing a single group in which all registers are capable of recording either dial pulses or keyset pulses.

The holding time varies with the customer and the type of call. Some customers take longer to start dialing. Dialing time varies. The number of expected digits may vary; for example, local calls require 7 digits and long distance calls 10 digits. (This is equivalent to variable word length in computers.) The expected number of digits may be made known to the receiving circuits in several ways: 1) by the coding of the number dialed assignments, 2) by allowing time after each digit is dialed for another digit to be started; if no new digit is started after 2 or 4 seconds, it may be assumed that sufficient digits have been received, or 3) by an end of dialing signal which eliminates the need for timing for further digits and is most useful where coding conflicts or large differences occur in the number of expected digits.

To reduce the call receiving circuit holding time still further and to improve service, overlap features are sometimes provided so that when the circuit has received sufficient information to interpret the general destination of the call, this information is processed before awaiting the receipt of the complete called number. For example, while the fourth and fifth digits are being received the first three are interpreted and a connection is established to the desired office. While the sixth and seventh digits are being received the information interpreted from the fourth and fifth digits is being transmitted to the office of the called number. By introducing overlap operation, a system is better able to serve in real time with a minimum of delays. It means that the receiving circuits must be more complex and generally capable of sending as well as receiving. Overlap may also be employed within the control so that a single control unit may act simultaneously on different phases of the different calls. Again this improvement in real-time service requires more complex circuitry. In telephone system design this complexity must be matched against the additional delays that might be encountered without the feature or the cost of additional control circuits to provide equivalent capacity.

The logic of the call receiving circuits or program devised for interpreting call data must be capable of dealing with much more than the normally expected call inputs of 7 or 10 digits. There are the types of service calls with "0" or 3 digit codes. All possible codes are not used or usable. If one of these were dialed, the system must be capable of so informing the customer, and more important, of disposing of the call to free the common circuits for use in more productive work. The system must also handle such nonproductive situations as when the handset is knocked off the "hook" momentarily or even for extended periods, or incomplete dialing, or not hanging up at the end of a call. Finally, it must be prepared to receive a hang-up, disconnect, or abandonment of the call at any time and be ready to serve that input again as quickly as possible thereafter. This is usually the customer's way of indicating an error as well as a change of mind. In short, the logic or program for the system must be prepared for any eventuality and always have a plan of action. There can be no dead ends or stops in logic circuits or program operation if the next expected piece of information is not forthcoming.

To accommodate these factors, a series of tones and announcements are used to inform the calling customer of the status of his call. A dial tone, busy tone, and ringing tone all let the customer know of the progress being made to serve his request. They may also serve to notify the customer that the machine is ready to serve him, and thus discourage him from sending in information when the call receiving equipment is not ready. Recorded announcements are used to inform of expected duration of delays and for intercepting on incorrectly dialed or unassigned central office codes and telephone number series.

Most communication switching systems are designed with call storage means provided on a traffic or when-needed basis at the central office. Call storage can also be provided at each source.

Certain tape teletype systems are of the type which include source storage.[6] But here traffic is delayed to insure

[6] W. M. Bacon and G. A. Locke, "A full automatic private line teletypewriter switching system," *AIEE Trans.*, vol. 70, pt. 1, pp. 473-480; 1951.

a high occupancy of the communication channels. Telephone systems where the complete number is preset before sending it are also of this type.[7]

Multiple Inputs

The fact that communication switching systems serve a plurality of inputs has been referred to several times in the foregoing discussion. The traffic problems which this raises are probably the most important real-time aspect of communication switching systems. There are other factors that influence design, which might be of interest in other multiple input computers. Needless to say, when there is a plurality of inputs to be served in real time, there may, by necessity, be a plurality of control circuits provided to keep up with the input information processing. (It is possible where traffic is light or by introducing higher speed control devices a single control circuit will suffice.) Each input is two-way. It is designated by a number. Since there are few, if any, restrictions on the requests from any input with respect to what numbers may be called, the control circuits must be capable of reaching any number. The more inputs there are, the more digits must be received, the more complex the selecting functions each control circuit must perform, and the longer the time it may take to do them. Furthermore, with more inputs, there will be more control circuits and a greater interaction among them. Naturally, this greater complexity increases the unit cost.

In a single input system the call receiving portion is presumably always ready to serve. When the receiving circuits are provided on a traffic basis in multiple input systems, it is possible that one will not be available without some delay. Premature dialing due to these short delays in assigning call receiving equipment is one of the most difficult traffic problems in real-time telephone systems. This is the doorstep of the system. Once inside, delays can be controlled by providing adequate storage if the delays are tolerable and the control is alert to abandonment of calls by the customer. It is possible to develop conditions known as "snowballing" of troubles. Under these conditions, serious overloads, or the "nervous breakdowns" of communication switching systems that you may have heard about, can be produced. When a call receiving circuit is available it may be connected to an input which has already started to dial. A wrong number or partially dialed call will result since the system does not have the correct call information. This means wasted usage of the limited call receiving circuits when they are most needed and consequently system capacity is reduced during overload.

Other overload conditions peculiar to multimachine operation will be described in the next section.

There are other factors which must be considered with multiple input systems. We have already mentioned some differences, for example, between operator and customer inputs. There are many different types of customer inputs, *e.g.,* coin sources and business or residence sources with flat or message rate. Furthermore, the customers may be on single or multiparty lines, or have a subswitching office such as a PBX. Each of these classes of input may require some different treatment on calls originating from and even terminating to these lines. These differences as well as differences in signaling languages and the necessity of anticipating abnormal actions make complex the logic of the control circuits or the programs for these systems. Each request must be interpreted in accordance with the class of the input and the desired output. Some noncoin customers, for example, may be able to reach numbers not dialable by coin customers.

Despite all these differences in class of inputs, certain standards, particularly electrical, are set so that the same digital receiving equipment will operate with most classes of inputs. Standards are set on signaling limits and also for transmission so that the established communication path, regardless of length, will be satisfactory. Furthermore, where there is a large number of sources it is important that these be standardized as much as possible to insure low initial and repair costs. For this reason, it is important that the many needs of the customer be satisfied by combinations of standard instrumentation.

Multimachine Operation

It is well known that the automatic communication central office was the first successful large scale digital information processing machine, and that such machines are interconnected over great distances and communicate with one another. The network of these machines is the largest and most extensive digital processing equipment ever to be assembled or likely to be assembled. Machines therefore have inputs not only from customers but also from other machines. There are a number of interesting engineering problems which arise in networks of such machines.

As was the case for inputs from customers, these inputs also transmit digital information, and call receivers are connected to them when digital information is sent for processing. Since the interconnecting links are sometimes quite long and costly, it is usually economical to use them in two directions which means that a request for service may originate at either end. Once the channel is seized at one end, precautions must be taken to prevent seizure at the other end and to connect a call receiver. Guard means are also provided to prevent immediate reseizure of the channel or to delay digital transmission when an interoffice channel is reused after release from a previous connection.

When signaling between machines in this real-time system, the information to be transmitted is usually available by the time the path between offices is seized. It is desirable to devise signaling methods for this application which

[7] W. A. Malthaner and H. E. Vaughan, "An experimental electronically controlled automatic switching system," *Bell Sys. Tech. J.,* vol. 31, pp. 443-468; May, 1952.

are faster than those serving customers, since only a small part of the total time required by the customer to send digital information to the machine is signaling time. As the art has progressed, new and improved signaling techniques have been devised. Rather than thwart progress, the new signaling methods have been adopted with new systems. Therefore, when switching machines of different vintages are placed together in a network they do not necessarily have a common language. When it is determined to which machine a call is to be routed, the method of signaling is also determined. Means must be provided in the originating office to send at least one of the languages which the machine to which it connects is capable of receiving. The smaller the number of languages which an office must send or receive, the simpler its circuitry and the lower its cost.

Signaling distances between machines are generally greater than from the customer to the machine. Therefore, it is sometimes necessary to place intermediate signaling equipment in the intermachine paths to regenerate or amplify the signals. Regeneration usually involves storage which delays the retransmission of the information.

Also, when paths or connections are established between machines, a new signaling problem develops. It is necessary to send signals over the connection in the direction opposite to that of the original digital transmission to indicate when the called party answers so that call changing may start, and also when the called party disconnects to indicate that the connection should be taken down. In general, these interoffice or intermachine signals are known as "supervisory signals" and the planning for these signals is as important as the planning for the digital transmission. As a real-time problem it is even more critical, for the longer the time intervals in the various unguarded periods for seizure, release, and reseizure, the greater is the chance of malfunctioning of the system.

In a complex network of offices or machines it may not be economical to provide paths from every office to every other office. For this reason it was realized early that intermediate switching machines should be established to provide more efficient trunking. They could also be used to regenerate the signals and translate the languages in both directions.

Normal traffic between machines is sometimes handled by means of alternate routing so that when all direct paths are busy the originating machine reroutes the call to an intermediate machine which also has paths to the desired terminating machine. This means that both the numericals of the called number and the desired central office code must be sent ot the intermediate office. The signaling language between the originating and intermediate office may be different from that which would have been used on a direct connection to the called office. The control circuits must take all this into account in processing the call without greatly increasing the holding time. Calls routed through intermediate offices take longer to complete, but this is minimized by employing overlap in the signaling.

Once intermediate offices are set up for traffic reasons they can be used to concentrate other complex switching functions which are required for only a small percentage of the calls. Thus a hierarchy of information processing machines has been established. More recently, automatic long distance switching equipment has been placed directly at the disposal of many customers and in the near future a large percentage of the nation's telephones will be able to call one another completely automatically by "direct distance dialing" through combinations of local and long distance switching systems.

There are overload situations which occur where the machines talk with one another. An example of this is on calls coming into an office from other offices. The incoming call receiving circuits here seem to act fast since there are usually no delays waiting for the calling customer. He has already given all the required information by this time. The holding time of these circuits is short; therefore, not many are required. But at the called customer's number, particularly a private branch exchange (PBX), a bottleneck may develop even though the PBX has sufficient lines to handle most of the traffic. The more traffic to this number, the more hunting required to find an idle path. Hence, delays are encountered by the incoming call receiving equipment which causes delays in call sending equipment in offices all over town that are trying to reach this office. With the call sending circuits so tied up, they cannot be used to complete calls to other offices. Again, an overload reduces call carrying capacity when it is most needed.

There are several ways of dealing with these overload situations. In electronic switching systems now being devised[1] where all lines are supervised on a time division basis, it may be accomplished by abundant call storage since such devices are relatively inexpensive. In electromechanical systems the overloads may be partially alleviated by introducing a speed-up in the time allowed for certain real-time functions when the office or some part thereof is working near maximum capacity. For example, the time allowed to determine whether the call is partially dialed or incomplete may be cut from 10 seconds to 2.5 seconds. Another design feature is to eliminate certain safeguards or trouble detection features which utilize control circuit time on normal calls during these overload periods.

The transmission of digital information must be as accurate as possible. This is particularly true between machines as contrasted to human sources since they usually have higher occupancy and transmit over greater distances. Here it is not uncommon to insert in the digital signaling paths compensatory electrical networks to insure that the signaling paths are more nearly identical in electrical characteristics. Such networks are automatically inserted on each call based on the call information. In a similar manner, transmission compensation including amplifiers may be automatically added.

System Changes

Another important characteristic of communication switching is that the system is dynamic, ever changing. As the system is changed it must continue to function, in real-time. A communication switching system cannot be taken out of service for maintenance, to add new service features, to add facilities to handle an increase in the number of inputs served, or to obtain information for the benefit of those administering the system.

When a general purpose computer is designed, compromises are usually made to produce a design which will satisfy to a maximum degree the greatest number of potential customers. Thereafter, modifications may be made for specific applications where the basic machine is not satisfactory. In the application of communication switching systems the needs for each community to be served are different. One machine design must fit the needs of a heavy traffic area in a large metropolis where there are many calls but with shorter holding times. This means more control or digital processing equipment in proportion to the switching network facilities. Another office might be in a small isolated area with fewer long distance facilities and little call handling capacity. The machine design may also be influenced by the method of charging for service, particularly the extent to which coin, message rate or flat rate noncoin services are offered.

The central office switching machine must be designed so that it can be manufactured, installed, maintained, and expanded over its entire range of applications. There can be little compromise between the machine's requirements and those of the customers. We cannot, for example, reduce the amount of equipment by taking more time—adding another eight-hour shift. Furthermore, when we determine that additional equipment is required, we cannot shut down the machine while it is being added. "Real time" here also means "all of the time." The number of combinations of the services available to customers and administrative and service features available to the communications companies mapped into the characteristic needs of each community gives a very large number of design variables for which each installation must be manufactured and engineered.

Imagine adding an additional arithmetic unit to a computer while it is working. In most communication switching systems there are usually new additions periodically. Engineering of each machine installation is continuous. We install enough capacity to deal with the real-time needs for a limited period, say one or two years. It is necessary to obtain a compromise between the over-all investment in idle equipment and the cost of engineering and installing to suit each change in demand. This means that from the start the equipment must be designed and the installations planned for growth.

Traffic Measurement

To carry out an intelligent and efficient engineering program, the system must be fitted with built-in measuring devices so that the traffic characteristics and the degree to which the equipment is handling the load offered may be measured and compared, and engineering steps may be taken to insure a satisfactory grade of service. Since the system operates in real time these data cannot be recorded at any other time except when the calls are taking place. The measurements must be made without impairing the service.

The results of traffic measurements may show the need to rearrange the available plant on a seasonal basis in addition to providing new equipment. Digital data processing of these traffic measurement records is helping the traffic engineer to keep up with the real-time central office machines by shortening the interval between measurement and valuation.

In designing a complex digital data processing and switching system it is difficult to predict by theory or analysis the way it will respond to the real-time input conditions. Therefore, it is not uncommon to set up existing digital machines to simulate the functioning of the real time machine under typical and overload conditions. In this way, "real time" can be slowed up and studied microscopically.[8] With the advent of general purpose digital computers some communication switching system simulation problems have been solved with these tools. The difficulty of programming these nonmathematical problems and the frequent lack of sufficient memory capacity have impeded the use of digital computers for this application.

System Maintenance and Reliability

Earlier it was said that in communication switching systems real time also means all of the time. This means that the detection, location, and repair of system faults must go on with the system still giving service, preferably without impairment. This means that the system must contain redundancy so that it is not dependent upon one and only one of a certain element to handle the traffic. When more than one element is required for traffic reasons, then a failure of one merely reduces the capacity. By providing more elements than required for normal traffic two objectives may be fulfilled. First, overload peaks are better accommodated. Second, taking an element out of service for maintenance will not upset the normal traffic capacity.

Of course, once a system element fails and is removed from service it is important that the trouble location and repair time be kept small enough relative to the failure rate so that the multiple failures occur infrequently and are of short duration. To achieve this, the circuits are frequently designed with self-checking and even error-correcting features. Trouble recorders are provided to indicate the state of the circuits when error checks detect troubles. In the more complex systems the trouble records may be automatically analyzed to diagnose the trouble and to indicate its location.

[8] G. R. Frost, W. Keister, and A. E. Ritchie, "A throwdown machine for telephone traffic studies," *Bell Sys. Tech. J.*, vol. 32, pp. 292-359; March, 1953.

Some types of circuits, particularly those associated with signaling and transmission, cannot readily detect their own troubles so that auxiliary routine test circuits are provided which, once started, automatically connect to these circuits one at a time and put them through their paces, usually with marginally acceptable signals. Automatic recording is also provided with such test circuits.

To avoid calls from being lost due to malfunctioning equipment, second trial features are sometimes provided which allow a different combination of system elements to be employed on a call, if it does not progress satisfactorily on a first attempt. Failure of a second trial usually results in a tone connection which indicates to the customer to try again.

The above is an example in a real-time system where trouble not only cuts down the capacity of the system but, to insure some service to each call, additional work load is taken on when it can be least afforded. Features such as this may be automatically circumvented in periods of overload, but are provided to avoid failure of calls to be completed in nonbusy periods. In this way the modern offices with low failure rates may be left unattended by maintenance forces during nonbusy periods with reasonable assurance of providing good service to all customers. Even when the office is left unattended, remote alarm indications are given at a manned control center so that steps may be taken to repair serious troubles before they do affect traffic.

System Power

What has been said about service continuity in case of equipment failure applies equally well to power equipment. Communication switching systems in central locations normally use power that is available commercially. This power source is backed up by local diesel generators which usually start automatically if commercial power fails. Being a real-time system, a communication switching system cannot have power off even during the short period required to start a diesel engine. If this happened, all the communication paths established through the office would be temporarily or permanently lost, depending upon whether or not the memory associated with that part of the office maintaining the switched paths was "volatile." Such service would not be considered satisfactory, especially where connections would not be reestablished after power was available from the diesel generator. Therefore, power storage in the form of wet batteries has been an essential part of a communication switching office. The capacity provided in these batteries is sufficient to cover the maximum expected period between the loss of commercial power and the full operation of the reserve power source.

Other power considerations in these systems call for dispersion of power so that trouble on any one feeder will not put the entire machine out of service. Such considerations apply also to other intramachine cabling.

Other Forms of Systems

Another form of communication switching system which functions in real time is that used for network broadcasting of radio and television.[9–11] Here the real-time switching is cued to the actual time specified by the customer. A "program" of such times and desired connections are the inputs to the system. There can be no delay or information is lost. Perhaps with the increase in data transmission this type of "real" real-time system will find greater application.

Conclusion

It has been shown that communication switching systems are a form of real-time digital computers. Some of the engineering considerations in designing such systems have been presented. These requirements are difficult to meet because of their complexity, the dynamics of the service demands, numbers of inputs and their characteristics, and the need to provide absolute continuity of service. Despite this, communication switching systems have been designed, built, and installed, and are operating as the world's largest aggregation of real-time digital data processing machines.

[9] C. A. Collins and L. H. Hofman, "Switching control at television operating centers," *Bell Labs. Rec.*, vol. 35, pp. 10-14; January, 1957.
[10] A. L. Stillwell and A. D. Fowler, "Switching at tv operating centers," *Bell Labs. Rec.*, vol. 34, pp. 366-369; October, 1956.
[11] P. B. Murphy, "Program switching and pre-selection," *Bell Labs. Rec.*, vol. 20, pp. 142-148; February, 1942.

An Introduction to the Bell System's First Electronic Switching Office

R. W. KETCHLEDGE†

A FULLY electronic telephone central office is being developed for experimental Bell System service. Both the electronic devices themselves and their system organization represent major changes in the art of telephone switching. The electronic switching system consists of electronic voice frequency switches controlled by electronic memory and logic. However, the system is not designed on the basis of direct substitution of electronic circuits for corresponding relay circuits of an existing switching system. Rather, the electronic circuits are organized in ways that exploit the advantages of the electronic technology.[1]

One of the obvious differences between the electronic and the electromechanical switching technologies is simply speed of operation. The electromechanical devices such as relays operate in times measured in milliseconds, often many milliseconds. The operate times of electronic devices such as transistors are measured in microseconds, often small fractions of a microsecond. Thus, the relative speeds differ by a ratio of well over 1000. This dramatic difference in speed permits a given amount of electronic equipment to handle much more telephone traffic than a corresponding amount of relay equipment. The high-speed operation also permits the system designer to organize the electronic system quite differently and often much more efficiently.

Comparison of existing types of electromechanical systems with the electronic system now under development shows that in the electronic system the various equipment units are much more specialized. This functional concentration or specialization of function is well illustrated by the case of memory. In relay systems the memory function is widely dispersed throughout the system. For example, in a modern relay switching system, memory functions are performed in all of the various kinds of circuits. In the electronic system much of the memory function is concentrated in two memories, one temporary and one permanent, which perform most of the memory functions for the system.

In a similar manner most of the logical operations of the electronic system are performed in a single functional unit. Again this contrasts sharply with electromechanical systems whose relay contacts are used for logical operations in all parts of the system.

Functional concentration permits the electronic system to be organized into a group of major components, each of which performs some single major system function. These equipment units can thus be designed to perform their function very efficiently and, because of the high speeds, perform it for the entire system. A further result is the simplification of the interconnections between the various equipment units. The number of wires connecting these units together is measured in tens rather than the hundreds or even thousands of wires often used in relay systems. The electronic system has a further advantage of understandability by virtue of the simplified relationships among the functional units.

Most of the signals that flow between units can be described as a combination of an action and an address. The address gives the location at which the action is to be performed. For example, the temporary memory might receive an order to write or read at a particular memory site. Alternatively, the switching network might receive a set of signals representing the action of "connect a voice path" and the addresses or identities of the terminals to be joined.

System Operation

A block diagram of this electronic switching system is shown in Fig. 1. The memory and logic units are separated from the voice switches and gain access to lines and trunks only through the scanner and selector. The scanner and selector are multiposition diode switches which may be directed to particular terminals for collecting or transmitting information. The function of the scanner and selector is to permit the fast control circuits to be time shared among the telephone customers. Information gathered by the scanner is processed by the controls and results in orders to the switching networks or to trunks via the selector. At any instant the system is usually engaged in processing only a part of a single call. Simultaneous actions involving more than a single call rarely occur.

In order to meet the real-time demands of the telephone customers, some system actions are given higher priorities than others. For example, it is more important to count a dial pulse than to detect a call origination because of the transient nature of the dial pulse. Thus, in each 5-msec interval, the system goes through its more urgent tasks first and then, if it has time, completes its less pressing commitments. The cycle time for any single logic memory, or scanner function is 2.5 μsec. While this provides 2000 operations in each 5-msec interval, many of the tasks take

† Bell Telephone Labs., Inc., Whippany, N.J.

[1] A. E. Joel, "Electronics in telephone switching systems," *Bell Sys. Tech. J.*, vol. 35, pp. 991-1018; September, 1956.

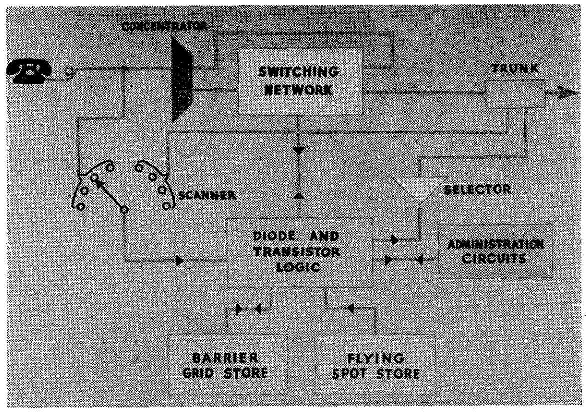


Fig. 1—Electronic switching system block diagram.

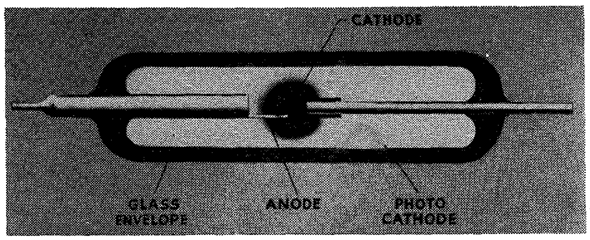


Fig. 2—Gas diode electronic crosspoint.

a number of operations to complete. Further, the number of tasks varies sharply with the telephone traffic. In a typical interval the system would first scan about half of the lines which are dialing in order to gather and record any new dial pulses. Next would come selector actions involving, perhaps, pulses being sent out on trunks to other offices. Then, any traffic awaiting network action might be completed. If this exhausts the interval because, for example, a large number of dial pulses occurred, then the system immediately goes back and repeats these high priority tasks in the next interval. This defers the lower priority tasks, but not for long since the probability that two successive intervals will both be overloaded is low. In the second interval the other dialing lines are scanned, and so forth. Then, lower priority tasks such as regeneration of the barrier grid stores, scanning of all lines for call origination, etc., are completed.

These actions result in all lines being scanned ten times a second for call origination. If a line is found which is drawing current but which the memory reports was idle on the last look, the controls recognize a service request and record the situation in the memory. The presence of this line number in the memory results in the line being scanned 100 times a second during dialing. The higher scan rate is required to insure detection of all dial pulses.

Switching Network

The switching network and the associated concentrator network provide the voice frequency paths for interconnecting telephone lines with each other and with trunks and various signals (ringing, dial tone, etc.). The switching element is a cold-cathode gas tube as shown in Fig. 2.

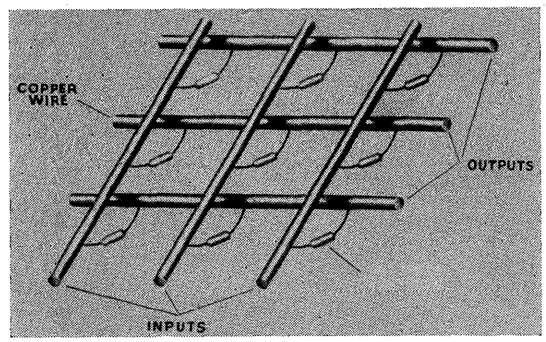


Fig. 3—Gas diode switch.

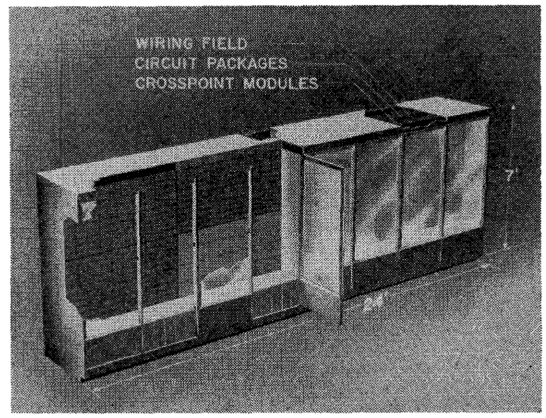


Fig. 4—Cabinet arrangement for the switching network.

It is a neon-filled diode utilizing a hollow cathode to obtain negative resistance in the conducting condition.[2] This tends to compensate for transmission losses of transformers and other elements in the talking path. These gas tubes are arranged into switches of the type shown on Fig. 3. Application of one half of the breakdown voltage on an input and an output wire causes the gas tube joining these wires to fire and connects the wires for speech transmission. Only one side of the transmission circuit is switched, the other side being grounded. Large numbers of these switches are connected together to form the complete network. A typical connection would be through one tube in a concentrator switch, then through six tubes in the switching network, and finally through one tube in a concentrator switch to the other telephone. Fig. 4 shows the physical form of the switching network. The gas tube crosspoints and the control circuits are assembled in plug-in packages which are then inserted in the cabinet. This permits easy maintenance and growth.

Logic

In general, the circuitry used for the processing of control information in the electronic switching system can be characterized as asynchronous and dc coupled, using germanium-alloy junction transistors as the active elements.

[2] W. A. Depp and M. A. Townsend, "Cold cathode tubes for audio frequency signaling," *Bell Sys. Tech. J.*, vol. 32, pp. 1371-1391; November, 1953.

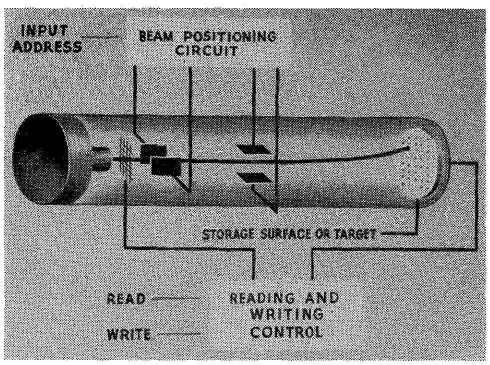


Fig. 5—Barrier grid store block diagram.

It is constructed of small general purpose circuit packages which are interconnected in accordance with the system control operations to be performed.

Logical operations are performed by conventional semiconductor diode AND and OR gates. To permit standardization of the design of these gate circuits, transistor gate amplifiers are placed in chains of diode logic to maintain appropriate signal levels. Bit storage associated with the logic circuitry consists of groups of double-ended flip-flops. These transistor flip-flops are coupled to the diode logic circuitry and to external circuitry composed of relays, gas tubes, or magnetic cores through specially designed amplifiers. Other miscellaneous circuit packages include inverters, emitter followers, pulse stretchers, and cable pulsers.

Most of the logic circuit packages are based upon grounded emitter transistor configurations in which the transistors are held out of saturation by self-biasing or diode feedback arrangements. This permits high-speed pulse operation at reasonable gains (0.5-μsec rise and fall times with current gains of approximately 15 in the gate amplifier).

The logic circuit packages are physically laid out on printed wire boards with all components crimped around the edge and dip soldered. The circuit boards are all jack mounted in the equipment. Both cards and jacks are coded to prevent improper interchange of packages. A shoe is provided on each card using transistors, the removal of which permits the transistors to be tested without unsoldering their leads.

The Temporary Memory

Two general types of memory are provided, temporary memory and permanent memory. The temporary memory is used to record data which must be changed in the course of processing a telephone call. Conversely, the information recorded in the permanent memory is not changed during a call, although numerous references to this permanent information may be required. The distinction between these two forms of memory is therefore based not on reading but rather on their writing characteristics. In the temoparary memory, information may be recorded in little over 1 μsec whereas several minutes are required to change information stored in the permanent memory.

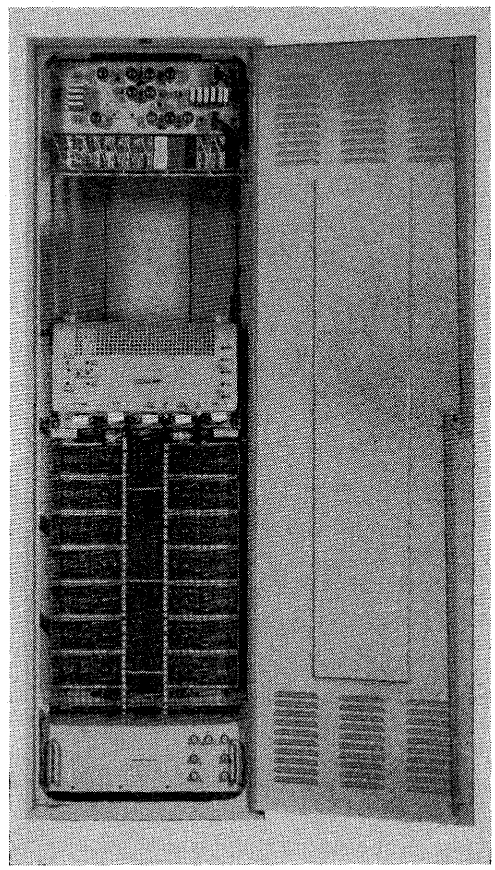


Fig. 6—An early model of the barrier grid store.

The temporary memory device used in the electronic switching system is the barrier grid tube.[3] This is an electrostatic storage tube wherein binary bits of information are recorded as electrostatic charges. An electron beam and electrostatic deflection plates provide access to the individual storage areas on a mica target. Writing is controlled by manipulation of the electric field at the mica surface.

The complete barrier grid store, shown in Fig. 5, consists of the barrier grid tube, deflection circuits to position the electron beam, circuits to turn the electron beam on and off and to pulse the target, an amplifier for the output signal, and control circuits to cause the various operations to occur in the correct sequences and at the right times. The barrier grid stores now operating are capable of storing 16,384 bits in a 128 by 128 array. In addition, the functioning times are quite short: 0.4 μsec to deflect the beam; 0.7 μsec to erase, of which only 0.3 μsec is required to read; and 0.7 μsec to write. The typical cycle is to read and then write at a given address, and this requires less than 2 μsec. Fig. 6 shows an early model of a store having these characteristics.

The Permanent Memory

In the electronic switching system there is a need for millions of bits of storage with access for reading in no

[3] M. E. Hines, M. Chruney, and J. A. McCarthy, "Digital memory in the barrier-grid storage tubes," *Bell Sys. Tech. J.*, vol. 34, pp. 1241-1264; November, 1955.

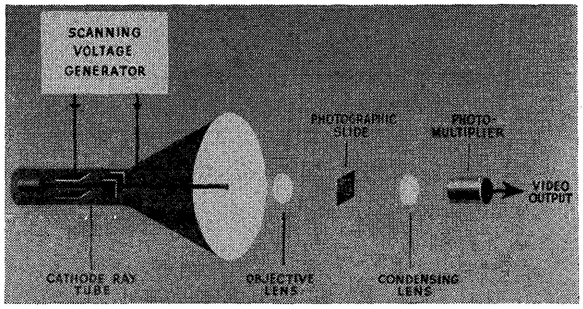


Fig. 7.—Television flying spot scanner.

more than 2.5 μsec. This information is of a semipermanent character and, for reliability, must not be lost by any failure of an electronic element. Storage of a photographic character was developed to meet this need. A group of photographic plates can store vast amounts of information as transparent or opaque spots and, with optical interrogation, an electrical malfunction cannot cause a loss of the stored information. Fig. 7 shows an example of a type of photographic storage, the flying spot scanner used in television. This represents a method for converting the transparency of a selected area on a photographic emulsion into an electrical signal. As shown in Fig. 8, the flying spot store uses similar techniques to read a number of photographic areas simultaneously.

In the photographic memory under development, over forty photographic areas 1.5 inches square are used to hold approximately three million bits of information. Each area is an array, 256 by 256, and individual spots are approximately 0.006 inch in diameter. Each photographic area has associated with it a lens to focus the spot of light from a cathode-ray tube. By use of a multiplicity of lenses in front of the same cathode-ray tube, a number of photographic areas can be interrogated simultaneously. Quick random access to any of the 65,536 words is achieved by electrostatic deflection of the cathode-ray beam.

In order to achieve practical photographic storage it is necessary to find means for positioning the small spot of light on the face of the cathode-ray tube to an accuracy and reproducibility of less than 1/1000 of an inch and in times of the order of a millionth of a second. This problem has been solved successfully. The basis of the solution is an optical-electrical feedback system which uses mechanical edges as the references for spot positioning. Fig. 9 is a photograph of an early model of the flying spot store.

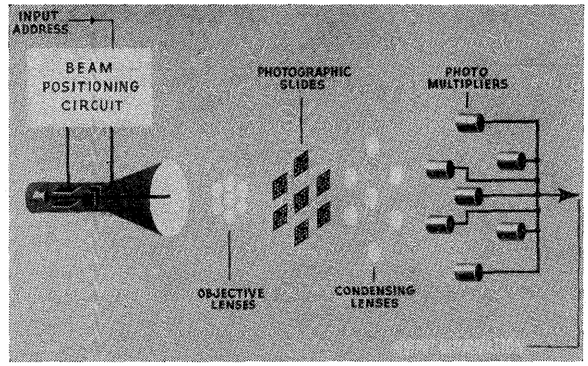


Fig. 8—Flying spot store block diagram.

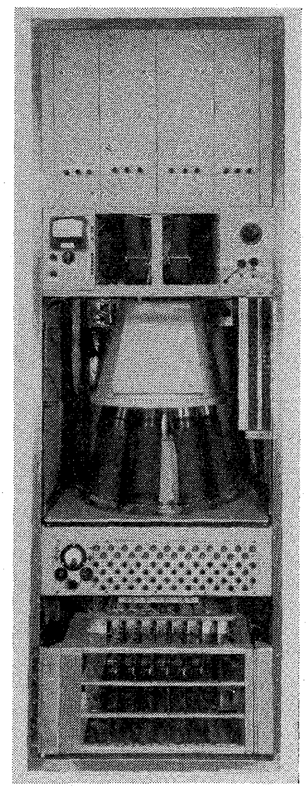


Fig. 9—An early model of the flying spot store.

Control by a Stored Program

The use of a stored program in place of wired logic is made possible by high-speed, random access, large capacity, permanent memory. The complexities of a telephone office are such that hundreds of thousands of bits of program are required. The real-time nature of the system requires fast random access, even though the stored information is changed infrequently. This substitution of memory for logic makes the remaining wired logic become a general purpose unit for interpretation and routing of order words. Thus, the machine problems are solved not by "logic" but by looking up the answers in the "back of the book." The stored order words thus control the system sequences and decisions for both telephone calls and internal trouble detection and location. The advantages of the use of a stored program are reduced system compleixty, fewer variations in manufactured units, reduction in wired options, simplified engineering of particular installations, and great flexibility. This flexibility permits the addition of new features and changes in control sequences with little or no change in hardware. The electronic switching system can be programmed to perform a wide variety of complex tasks and to render telephone service in a wide variety of ways. Changes in operation or provisions of new services are achieved by modification of stored information rather than by changes in wired connections. This same ease of modification is also useful for changes in line translation records which are also recorded in the permanent memory. These records give, for each directory number, the network terminal at which it appears and all class of service information.

Special Services

The application of electronic techniques to telephone switching make possible new services; services which have been impractical heretofore due to either economic or technical reasons. One example of what could be done is abbreviated dialing where one dials a preliminary "one" followed by a single digit. Each of the ten possible digits represents a preselected, frequently called, telephone number. You might choose to have a 1 represent a nearby friend; 2, your office; 3, a relative living thousands of miles away, etc. The system would recognize the type of call and, using its photographic plates, translate your 1X into the actual telephone number, whether it be a local call or not. The translations would, of course, have been previously been recorded in the system. This and many other services are made technically feasible by the use of a stored program and the use of electronic memory.

Conclusion

The introduction of electronic techniques into telephone switching represents a major change in the art. Both the new types of devices and new types of telephone system organization offer important advantages. Perhaps the most important result will be the increased flexibility and new services that this will make possible.

Traffic Aspects of Communications Switching Systems

JOSEPH A. BADER†

To DESIGN a communications switching system which provides a satisfactory grade of service at minimum cost, an understanding of the nature of the offered traffic is necessary.

For many years, telephone traffic engineers have been studying the problem of providing sufficient equipment to meet time-varying demands at a given level of service. The experience gained in these studies and the analytical results derived may prove valuable in the planning and use of modern digital data processing equipment for real-time applications.

Major Components of Traffic Variation

In order to design a switching system to meet a specified grade of service, an estimate of the average traffic offered to the switching system must be made. The reliability of this estimate depends on the magnitude of the components of variation present.

Variations such as seasonal, day-to-day, and hour-to-hour are not easily susceptible to an analytic approach. They are usually determined empirically for each exchange or area. Typical seasonal, daily, and hourly traffic patterns are shown in Figs. 1 to 3. These patterns are usually stable so that in choosing the average busy season, busy day, busy hour traffic as a base for engineering, we can be reasonably assured that the busy hour traffic offered during the rest of the year does not greatly exceed our engineered estimate. Of course, events occur such as earthquakes, snow storms, disasters, etc., which provide a common cause for call origination. This results in traffic "peaking" well above the engineered level with a resulting degradation of service. Under such extreme conditions, special overload-control procedures are usually initiated which tend to spread the peaked demand over a longer period of time.

The remaining component of variation is the instantaneous variation of the number of calls offered per unit time during the busy hour. Fig. 4 shows the variation in the number of calls offered per 24-second interval for a 10-minute measurement period. From these data, a frequency distribution giving the fraction of 24-second intervals containing x call originations was constructed (Fig. 5). If calls arrive at random, then the distribution per unit time should be Poissonian. To test the data for randomness of call arrivals, a graph of the Poisson and the sample distribution was constructed (Fig. 5). By inspection, the agreement appears to be quite close. It is evidence of this kind which lends assurance to the assumption of random-call input which is basic to most traffic theory. Accordingly, we can compute the magnitude of the instantaneous variation of offered traffic as the first step in determining the engineered capacity of the system.

Holding-Time Variations

The next step is to investigate the length of time required to serve the offered calls. The service time, generally called the holding time, ranges from a fraction of a second for certain switching equipments to several minutes

† Bell Telephone Labs., Inc., New York, N.Y.

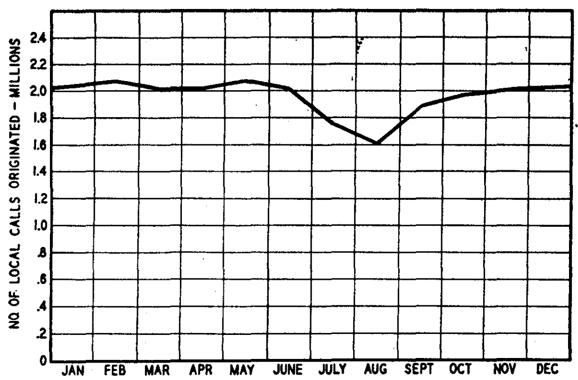


Fig. 1—Seasonal variation in daily local calls in Boston, 1934.

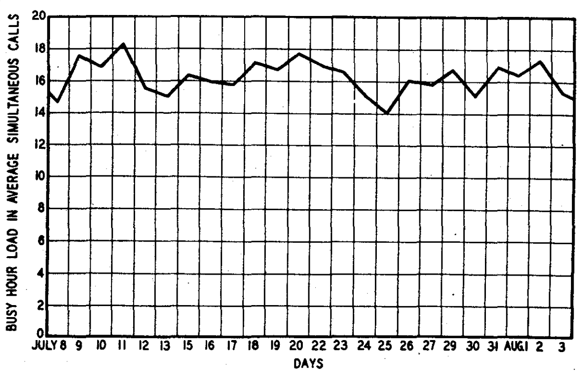


Fig. 2—Day-to-day busy hour variation in load, Newark, 1918.

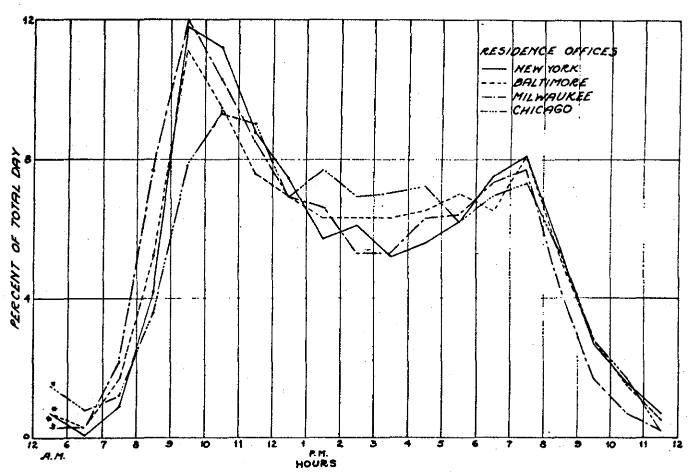


Fig. 3—Hourly traffic distribution. (*) values shown are per cents of total traffic handled in 12 MIDNIGHT-6 A.M. period.

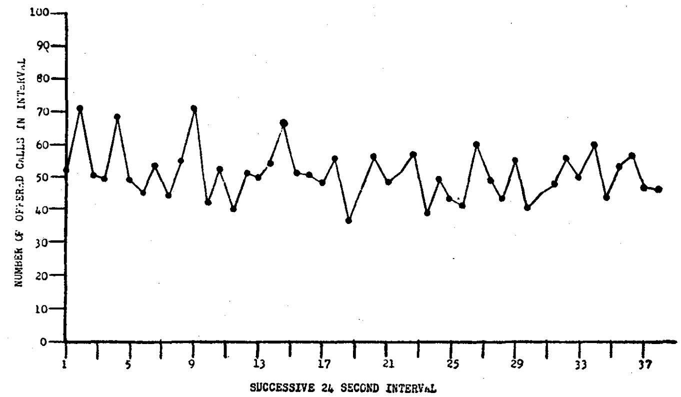


Fig. 4—Variation in number of calls offered per 24-second interval, Asbury Park, 1957.

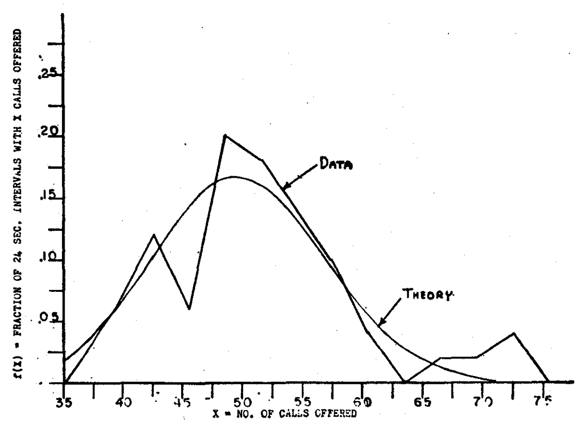


Fig. 5—Comparison of Poisson theory to measured number of calls offered.

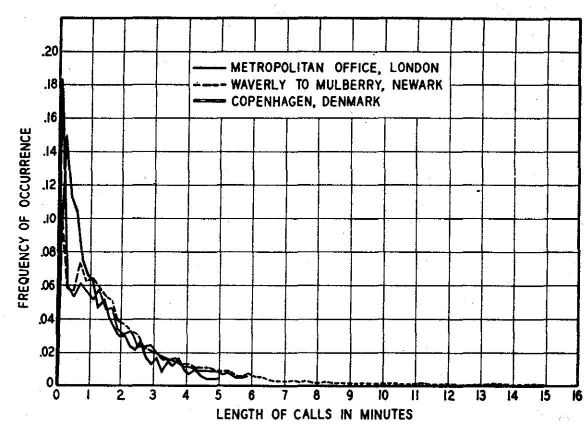


Fig. 6.

for talking paths. To expedite theoretical treatment, one of the following two assumptions is usually made with regard to call lengths or holding times: 1) holding times are distributed according to the exponential distribution, or 2) holding times are constant. The assumption of exponentially varying holding times is very well confirmed where local-call conversation and control time are examined. Fig. 6 shows the distribution of such times as measured in Newark, N.J., in a past traffic study. The agreement between data and theory is obviously good. Based on evidence of this kind, an exponentially varying holding time is assumed for equipment held during the entire conversation.

Constant holding time is exhibited by equipments whose function is to perform rapid switching operations prior to the beginning of conversations. For example, in one type of telephone switching system, a so-called marker performs the function of connecting the subscriber's line to an outgoing trunk. The marker then releases and is immediately available to another subscriber. The time required for this function is essentially constant.

Traffic Usage

Having determined the number of offered calls and the server's holding time, we can define a third parameter called traffic load, or usage. This is the product of the number of calls and the average length of each call. Thus, if a system is offered 1000 calls in the busy hour, each of average length 100 seconds, the offered load is 100,000 call seconds. In dealing with a system which is offered a large number of calls per hour, it is more convenient to deal with call hours per hour. In our example, then, the offered load would be 100,000/3600 or 27.8 call hours per hour. The traffic unit of one call hour per hour has been named the erlang in honor of A. K. Erlang, Danish mathematician, who pioneered in traffic theory. The offered load expressed in erlangs represents the average number of simultaneous calls that would be in progress if sufficient servers were always available. We can see this from Fig. 7 which shows the variation in the number of calls present on a trunk group between two central offices during the busy hour. During this hour, 246 calls were carried. Greatest number of calls in progress was 16 and this occurred four times during the hour. Average number of simultaneous calls in progress was 9.5. This means that on the average during this hour, the trunk group carried 9.5 call hours.

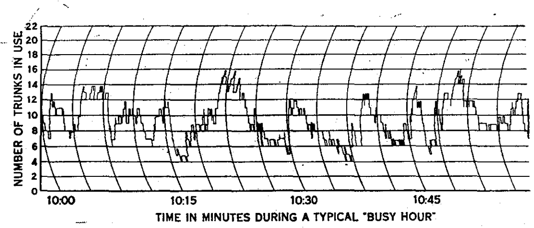


Fig. 7.

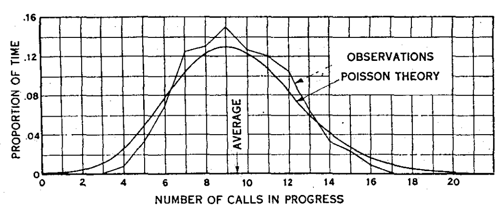


Fig. 8.—A comparison of theory with the observations in the hour above.

As indicated previously, the Poisson distribution correctly describes the variation in the number of random-call arrivals per unit time. Under certain assumption, however, the Poisson also describes correctly the probability of finding at a random instant a given number of calls on a group of servers. Fig. 8 shows the Poisson with average 9.5 and an empirical distribution of the number of calls in existence each 30-second interval taken from the data of Fig 7. In this case, the Poisson closely predicts the proportion of time x calls that are present; this is also clearly the fraction of calls arriving and finding x calls ahead of them in the system. If x exceeds the number of servers provided, newly arriving calls will fail in obtaining immediate service. The proportion of such failures is a commonly used criterion for the adequacy of service. Modification of the assumptions underlying the Poisson gives rise to other formulations. In particular, the assumption regarding the behavior of calls failing to find an idle server immediately is of primary interest.

Defection Ratio

The behavior of calls which fail to find an idle server immediately can be expressed in terms of the defection ratio j. This is the ratio of the rate at which waiting calls abandon before being served, to the rate at which they are served. j, of course, can assume values from 0 to infinity. However, in Telephone Traffic Engineering it is usual to find only three different values of j assumed. These values and their physical interpretation are as follows. 1) $j=0$ corresponds to the case in which unserved calls wait indefinitely for service. In telephone traffic parlance, this is called the "lost calls delayed" assumption. 2) $j=1$ corresponds to the case in which calls wait no longer than their holding time and then abandon. If an idle server becomes available, a call seizes the server and uses it for the remaining part of its holding time. This is the "lost calls held" assumption. 3) $j=$ infinity corresponds to the case in which calls are not willing to wait at all for servers and immediately abandon. This is the "lost calls cleared" assumption.

Loss Engineering

A group of servers engineered solely on the basis of "expected proportion of calls which fail to receive immediate service" is said to be engineered on a "loss" basis, and formulas used to predict the proportion of calls failing to find an idle server immediately are called "loss" formulas. Loss formulas and their underlying assumptions are listed in Fig. 9. The assumption of "infinite sources" is, of course, never quite realized in practice but where the rate of arrival of calls is nearly independent of the number of calls momentarily being served, this assumption can be used with confidence. The list in Fig. 9 is by no means complete. However, the formulas tabulated are those most widely used for engineering telephone switching equipment. Graphs of these loss formulas are shown in Figs. 10 to 12. By means of these curves, the traffic engineer is able to solve a wide range of "loss-engineering" problems. The following examples demonstrate the use of these curves.

Example 1

How many trunks should be provided at P.01 service if the load offered is 10 erlangs?

Solution: Past experience indicates that for trunk groups with no provision made to reroute overflow calls,

c = Number of Full Access Trunks

a = Load Submitted in Average Simultaneous Calls

$$= \frac{\text{(Number of Calls per Hour)(Average Holding Time in Seconds)}}{3600}$$

j	Lost Calls Assumption	Usual Designation of Formula	Frequency Distributions, $f(x)$		Probability of Delay, P
			When $x \leqq c$	When $x \geqq c$	
0	"Delayed"	Erlang "C"	$\dfrac{\frac{a^x e^{-a}}{x!}}{1-P(c,a)+\frac{a^c e^{-a}}{c!}\cdot\frac{c}{c-a}}$	$\dfrac{\frac{a^x e^{-a}}{c!\,c^{x-c}}}{1-P(c,a)+\frac{a^c e^{-a}}{c!}\cdot\frac{c}{c-a}}$	$C(c,a)=\dfrac{\frac{a^c e^{-a}}{c!}\cdot\frac{c}{c-a}}{1-P(c,a)+\frac{a^c e^{-a}}{c!}\cdot\frac{c}{c-a}}$
1	"Held"	Poisson	$\dfrac{a^x e^{-a}}{x!}$	$\dfrac{a^x e^{-a}}{x!}$	$P(c,a)=\sum_{x=c}^{\infty}\dfrac{a^x e^{-a}}{x!}$
∞	"Cleared"	Erlang "B"	$\dfrac{\frac{a^x e^{-a}}{x!}}{1-P(c+1,a)}$	0	$B(c,a)=\dfrac{\frac{a^c e^{-a}}{c!}}{1-P(c+1,a)}$

Fig. 9—Familiar telephone-traffic formulas assuming infinite sources.

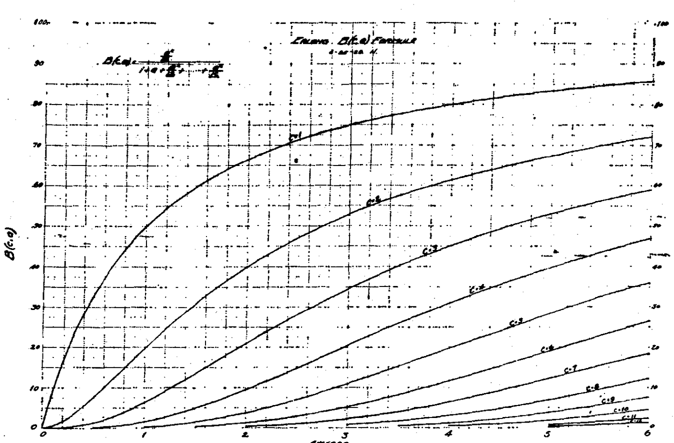


Fig. 10—Erlang "B" load vs loss curves.

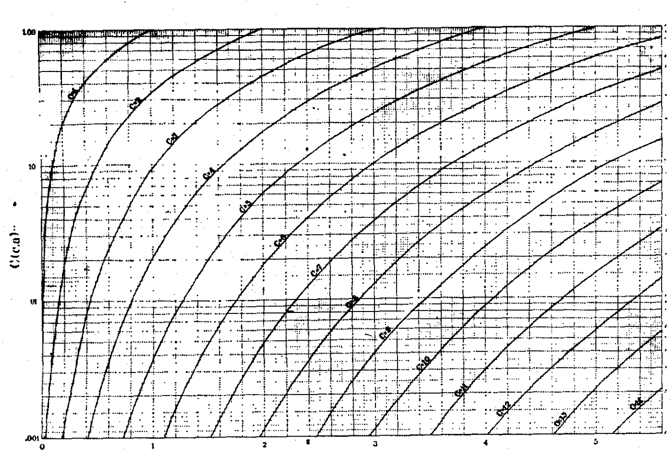


Fig. 11—Erlang "C" load vs loss curves.

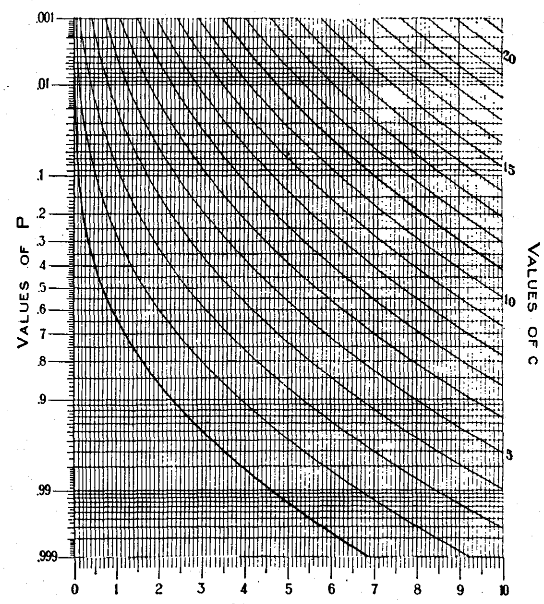


Fig. 12—Poisson load vs loss curves.

the lost calls held assumption applies reasonably well. Therefore, the Poisson loss formula is chosen here. From Fig. 12 we see that 19 trunks are required.

Example 2

Measurement indicates that a group of 10 dial-pulse registers are giving P.03 service. How many registers must be added in order to give P.01 service? Assume holding times are exponential and lost calls are delayed.

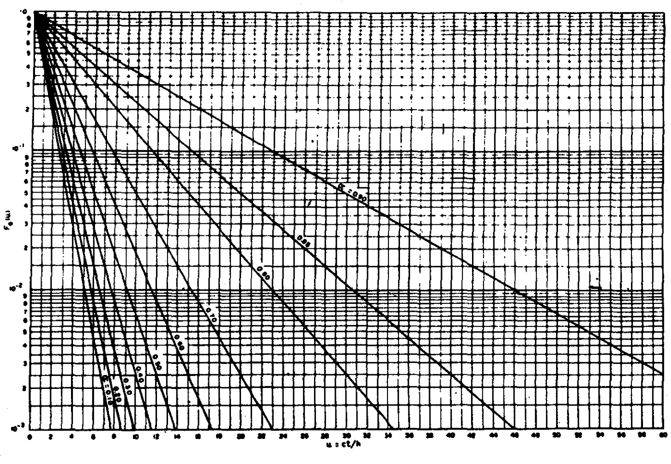


Fig. 13—Probability of delays exceeded with exponential calls served in order of arrival.

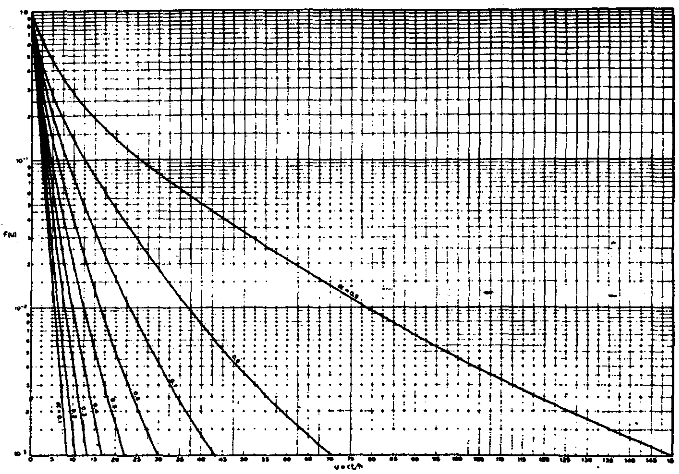


Fig. 14—Probability of delays exceeded with delayed exponential calls served in random order.

Solution: Fig. 11 shows that for 10 registers operating at P.03, the offered load is 4.84 erlangs. In order to give P.01 service, 11.3 registers are needed. Therefore, add two registers.

Delay Engineering

In many instances the traffic engineer is concerned with the length of delay as well as the probability of a delay. (Speed of dial-tone service is a typical example.) Theoretical formulations have been derived under the assumption of lost calls delayed for both exponential and constant holding times. In general, the length of a delay depends on 1) the offered load, 2) the number of servers, 3) the distribution of holding times, and 4) the queue discipline.

Theoretical descriptions of delay have been worked out for three queue disciplines. They are 1) "first come, first served," that is, waiting calls are served in the order in which they arrive; 2) "last come, first served"; and 3) "random service," in which waiting calls are served at random. Since the "last come, first served" queue discipline maximizes delays, it is never realized in telephone switching systems. There are, however, systems in which the other two disciplines are realized. Graphs of delay distributions for the queue disciplines 1) and 2) are shown in Figs. 13 and 14. The graphs show the conditioned distribution of delays. That is, they yield the probability of a delay greater than t *given* that a call is delayed at all. In order to get the unconditioned probability of a delay greater than t, we multiply by the probability of a delay $C\ (c,\ a)$ shown in Fig. 11. It should be noted that the average delay is the same for each discipline. From these curves, a traffic engineer can determine the number of equipments required to provide a given grade of delay service. The following examples demonstrate the use of these curves.

Example 3

A certain computer has a steering circuit which directs single, incoming pulses to an idle arithmetic unit. Pulses arrive at random and each pulse is acted on by the arithmetic unit for an average of 5 μsec. Pulses which arrive when all units are busy are held in a storage unit. Waiting pulses are served in the order of their arrival. The holding time of the steering circuit itself may be ignored. What is the capacity of the computer in pulses per hour if five arithmetic units are supplied and no more than 3 per cent of the delayed pulses are to be delayed longer than 10 μsec?

Solution: From Fig. 13, we find the occupancy, that is, the ratio of the offered load to the number of servers, of each arithmetic unit, is alpha = 0.65 so that $a = 3.25$ erlangs, since $a = \dfrac{N\bar{t}}{3600}$, we have $N = \dfrac{11700}{5 \times 10^{-6}} = 2.34 \times 10^{9}$ pulses per hour.

Example 4

A single toll booth is provided at the entrance to a bridge. Cars arrive at the toll booth at random at a rate of 0.15 per second. The time required to collect a toll is exponential with an average of 5 seconds; the booth is located such that cars cannot defect from the waiting line. What fraction of the cars are delayed more than 30 seconds in reaching the toll booth?

Solution: The load in erlangs offered to the single toll booth is $a = 0.15 \times 5 = 0.75$ erlangs.

The occupancy alpha $= \dfrac{a}{c} = 0.75.$

From Fig. 13 we find that $P(>6) = 0.23$ so that 23 per cent of the delayed cars are delayed more than 30 seconds. Multiplying by $C(1.75) = 0.75$ we have 19 per cent of all cars delayed more than 30 seconds.

Example 5

A counter in a department store is manned by three clerks. The time required to serve a customer is distributed exponentially with an average of 3 minutes. Customers receive a number indicating the order of their

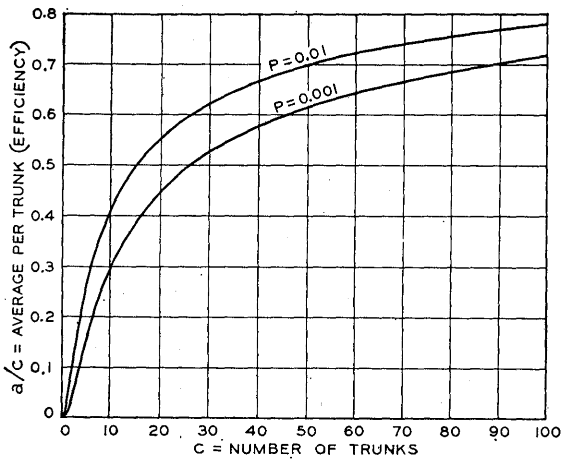


Fig. 15.

arrival so that service in the waiting line is very nearly first come, first served. Observations indicate that during the normal busy hour, 10 per cent of the delayed customers are delayed at least 6 minutes in acquiring the services of a clerk. If one clerk is added, what per cent of the customers would be delayed at least 6 minutes?

Solution: Using $P(>6)$ and Fig. 13, determine the offered load a. Since $\frac{a}{c} = 0.62$, we have $a = 1.86$. If one clerk is added, the new occupancy is alpha $= \frac{a}{c} = \frac{1.86}{4} =$ 0.47. From Fig. 13 we find that 1.40 per cent of the delayed customers would be delayed at least 6 minutes. The addition of one clerk resulted in over a 7 to 1 improvement in the per cent delayed at least 6 minutes.

Overload Performance

So far we have considered the problem of engineering equipment to accommodate an average busy hour load at a given grade of "loss" or delay service. Related to this problem is that of balancing efficiency against overload capacity. A brief discussion of this problem might be of interest.

The efficiency of a trunk group or average load per trunk is defined to be the ratio of the load carried to the number of trunks. Fig. 15 shows the relationship between efficiency and group size at engineered losses of P.01 and P.001. From the curves, it is clear that large groups are more efficient than small groups. On the other hand, the higher the efficiency, the less margin available percentage-wise for small overloads. Fig. 16 shows the relationship between the increase in loss from P.01 against group size when the offered load is over engineered level. The ideal balance between efficiency and overload margin depends on additional factors such as the purpose for which the system is to be used and the environment in which it is to function.

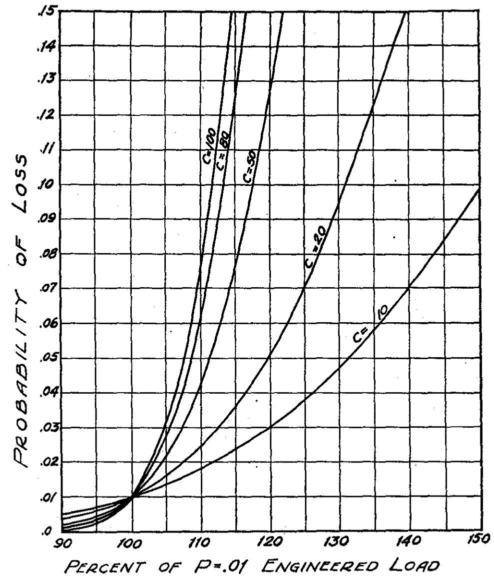


Fig. 16—Relative ability of different sizes of trunk groups to carry overloads.

Conclusion

Some of the fundamental traffic aspects of switching systems and some formulas by which probabilities of delays and losses may be calculated have been displayed. Working curves have been shown by which many traffic-engineering problems can be solved. Examples are given which illustrate the application of the curves in practical situations.

Bibliography

[1] Brockmeyer, E., Halstrom, H. L., and Jensen, A. *The Life and Works of A. K. Erlang.* Copenhagen: Copenhagen Telephone Company, 1948.

[2] Molina, E. C. "Application of the Theory of Probability to Telephone Trunking Problems," *Bell System Technical Journal,* Vol. 6 (1927), p. 461.

[3] Fry, T. C. *Probability and Its Engineering Uses.* New York: D. Van Nostrand Company, 1928.

[4] Thorndyke, F. "Applications of Poisson's Probability Summation," *Bell System Technical Journal,* Vol. 5 (October, 1926), p. 604.

The Use of the IBM 704 in the Simulation of Speech-Recognition Systems

G. L. SHULTZ†

Introduction

THE TERM speech-recognition device usually brings to mind a machine capable of duplicating the function of a human listener. Such a device not only would have to be capable of receiving and classifying acoustic stimuli, but also would have to be able to extract from these stimuli the message the speaker intended. To do this, a recognition machine must be familiar with the language statistics, and indeed, the entire human environment. Many years of investigation will be required before such a human replacement can be achieved.

However, more limited man-to-machine communications systems can be defined, and *acoustic* (as distinguished from speech) recognition devices can be built in the near future. Articulation tests with nonsense syllables show that listeners can agree upon, and classify, speech sounds on the basis of the acoustic signal with little use of language redundancy. Surely, then, a set of measurements exists by which a machine could likewise classify these sounds. Since classification of speech sounds is a necessary part of even the most comprehensive recognition system, our efforts are first turned to this task.

The study by introspection how we, ourselves, classify speech sounds is not very successful. We need to "look at" speech. The sound spectrograph was developed by Bell Laboratories to produce visible speech. An example of its display is shown in Fig. 1. The more familiar voltage amplitude vs time function is shown at the top of the figure. Directly below, and aligned in time, is the sound spectrogram of the same utterance. The frequency is scaled along the ordinate. Intensity at a given frequency and time is depicted by the blackness of the mark. Rules for reading these displays have been developed and reported in the literature by Bell Telephone Laboratories, M.I.T., Haskins, and others. These rules are being presented to us in a unified course conducted by Prof. Morris Halle at M.I.T.

Note that the spectrogram is divided into segments of no activity, horizontal bar structure, and areas of striation. The horizontal bars, termed formants, characterize the vowels, (*r*), (*l*), and nasal consonants (*m*), (*n*), (ŋ). The irregular striated areas characterize the fricative or noise-like consonants (*s*), (ʃ). A vertical blank area, followed by a sharp vertical line, is the distinguishing property of stops or plosives (*t*), (*p*), (*k*). The presence of a heavy horizontal bar at the bottom of the spectrogram indicates voicing. Shown here, the first word consists of an unvoiced fricative, a changing vowel sound, and a terminating unvoiced stop. The second word is voiced throughout and begins and ends with a nasal sound characterized by abrupt transitions in the formants of the middle vowel. The third word is the same as the first. Although we have only mentioned some rules for classifying vowels, fricatives, and stops, such rules have also been developed for subdividing these three classes into the approximately 40 basic elements of speech.

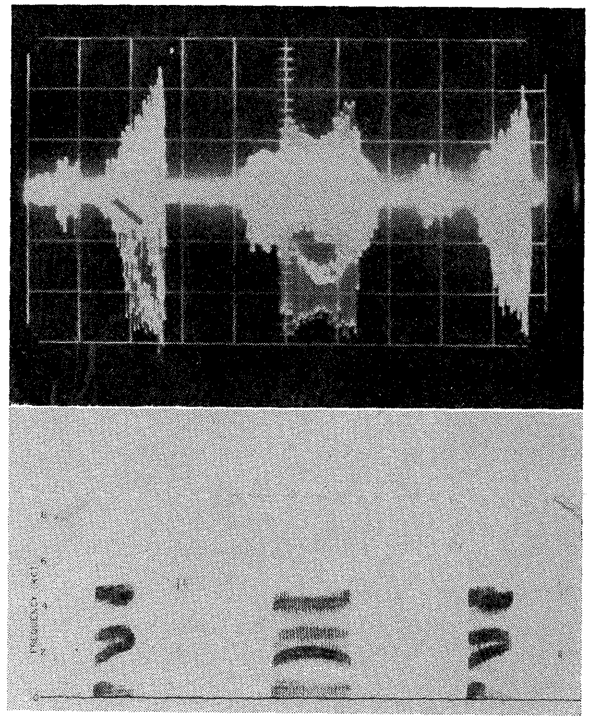


Fig. 1—Comparison of amplitude vs time function and sound spectrogram.

These qualitative rules must be operationally or quantitatively defined in solving the problem of mechanical recognition. For example, just what circuit would be able to identify a striated area? Even after quantitative rules have been defined, a set of physical properties will result whose ranges of variation with context and speaker must be determined. This calls for a statistical approach with its consequent data-handling problems. Further, the number and complexity of these speech properties require a versatile system of analysis. To accomplish this analysis, special advantage is taken of techniques made possible by the advent of the large-scale digital computer.

A computer can be programmed to duplicate any of the measurements of speech signals now used in speech studies.

† IBM Corp., Yorktown Heights, N.Y.

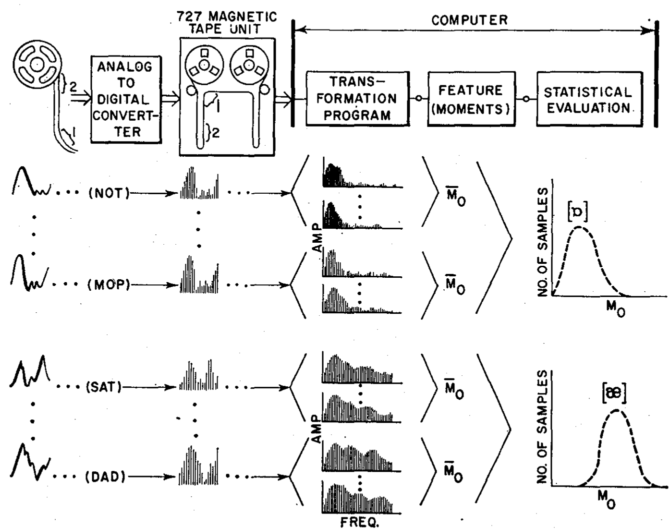


Fig. 2—System of analysis.

The computer is especially well suited for the large-scale reduction and analysis of data. The flexibility of the computer enables a recognition system to be evolved through easily inserted program modifications. Further, the inertia associated with the construction of specialized equipment is eliminated.

When a set of speech properties has been found, and a successful system based on this set has been duplicated in the computer, *then* the system can be embodied in the circuits of a speech-recognition machine.

System of Analysis

Using the computer as a central tool, we have built up the system of analysis as outlined in Fig. 2. In order to gather a large number of like speech events, a device is required to edit these events from continuous speech. Once a speech event has been selected, the acoustic wave must be converted to a digital form satisfying the input requirements of the computer.

The initial program routines are designed to aid in determining which speech properties are most significant with regard to recognition. First, the proposed property is computed for a large number of speech events. Then, the statistical distribution of these measurements is estimated and listed by like speech event.

We have divided the measurement of speech properties into two operations. The transformation block in Fig. 2 contains a basic program which yields a quantitative pattern of the acoustic signal. The many properties of this pattern are then explored by a set of simpler features programs.

Analog-to-Digital Converter

The input system of equipment required for the computer analysis consists of two machines, an Editor and a Coder, each with dual-track, audio-tape devices. Fig. 3 is a photograph of the Editor (left) and Coder (right). The Editor aids in the selection of speech events from continuous speech. The Coder converts these selected events to digital form. Care has been taken to make this system automatic so that the main concern of our work can be the study of speech events rather than their preparation. The desired speech event is selected by the Editor under push-button control. Editing is accomplished by transferring the section of speech containing the speech events of interest to an endless tape, or loop. Synchronizing pulses are placed on the second tracks of both the input and endless tapes as the speech is transferred. When the loop is read, two electronic delays, as shown in Fig. 4, are initiated by the loop synchronizing pulse. One delay is adjusted to extend from the loop synchronizing pulse to the start of the speech event. A second delay, oscilloscope-sweep length, is then adjusted to the duration of the speech event. This second delay also keys the selected portion into earphones, permitting simultaneous sight and sound adjustments.

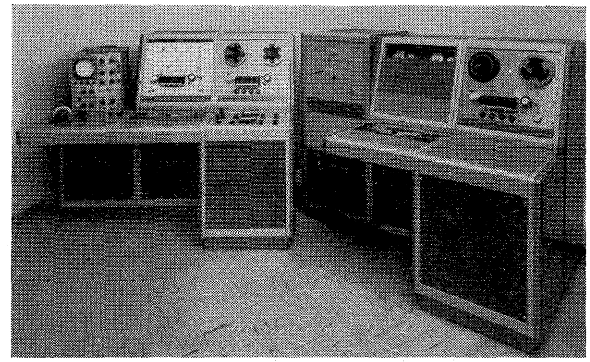


Fig. 3—Editor and Coder.

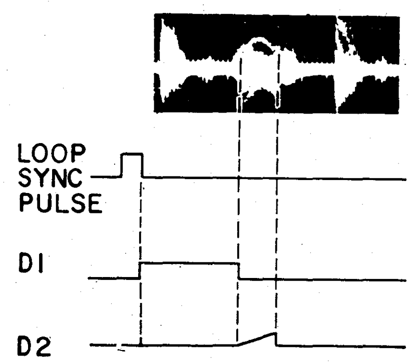


Fig. 4—Edit timing delays.

After the delays are properly set in the endless tape operation, they are initiated once more by the synchronizing pulse on the second track of the input tape. This final timing sequence properly writes an editing pulse on this track. Hence, the result of the editing operation is an audio tape with speech recorded on one track and editing pulses located on the second track opposite the selected speech events.

Fig. 5 shows the entire Editor-Coder operation in block diagram form. It has been split here for convenience into the audio and pulse tracks. As mentioned, the audio signal is recorded once on the Editor, and during the editing op-

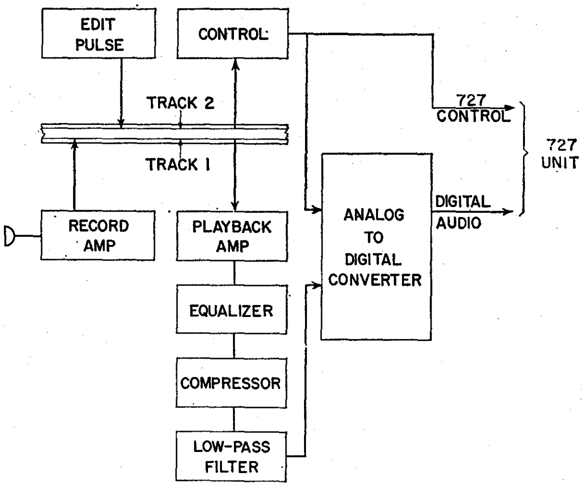


Fig. 5—Editer-Coder audio and timing.

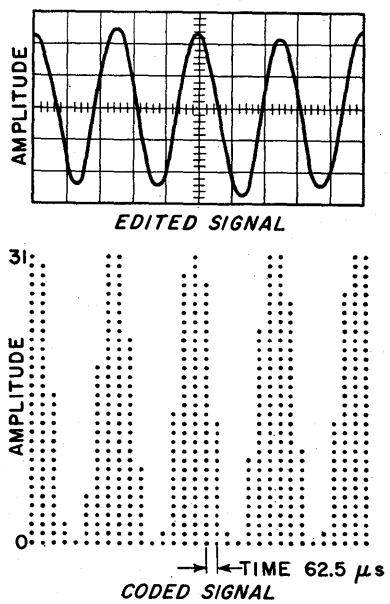


Fig. 6—Selection accuracy.

eration a pulse is affixed on the second track of this tape opposite the speech events selected. This tape is then played back in the coder where the audio information is *continuously* passed through a playback amplifier, an equalizer, a compressor, and a low-pass filter before entering the analog-to-digital converter. Upon reading a pulse from the second track of the tape, the converter, externally triggered at the prescribed 727-tape character rate of 16 kc, converts and speech-signal amplitude at each sample point to an 11-bit binary number. Simultaneously, the editing pulse causes control circuits to bring the 727-tape unit up to speed and properly write the converted speech signal.

One 727-tape record is made for each speech event selected. At present, only the five most significant bits are written on digital tape since this accuracy yields a sufficiently low quantizing noise value. The speeds (7½ or 15 ips) of the input tape in the Editor and the playback tape of the Coder can be arranged to produce effective sampling rates of ½, 1, or 2 times the tape character rate.

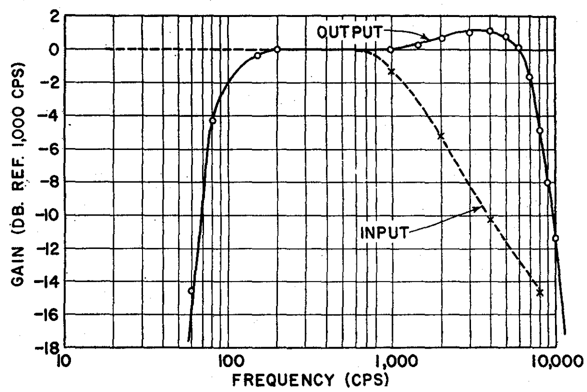


Fig. 7—Frequency responses of Editor-Coder.

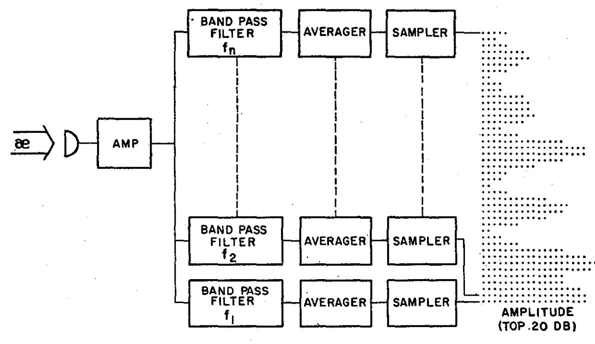


Fig. 8—Diagram of filter bank with spectrum.

The selection accuracy of the Editor is shown in Fig. 6. The 2-kc sine wave is shown at the top of the figure as it was selected in the editing operation. Below is shown a print out of this sine wave sampled at 16 kc. The accuracy of selection is primarily limited by tape-speed variation during delay 1. For a delay of 1 second from the synchronizing pulse to the start of the speech event, the start is located with an accuracy of approximately 2.5 msec.

The audio system response is shown in Fig. 7. Here the input signals were attenuated at a rate of 6 db per octave above 1000 cycles. With this input and the high frequency emphasis circuit inserted, the over-all audio response between half-power points is from 85 to 7500 cycles.

System of Programs

A set of programs has been written to implement the general analysis system. These programs have been tested together on a set of vowel phonemes.

The first transformation program was written to compute spectra. Essentially this program simulates a bank of band-pass filters as depicted in Fig. 8. The output of each filter is averaged for a certain period of time and this average output is sampled periodically. At each sample time the output amplitudes of the entire bank of filters are plotted as a function of the center frequency of each filter. The resulting graph is shown to the right of the figure. In this graph, then, we have two of the constituents of the spectrogram, namely, frequency and amplitude.

A series of these graphs at adjacent sample times would display amplitude and frequency as a function of time. The

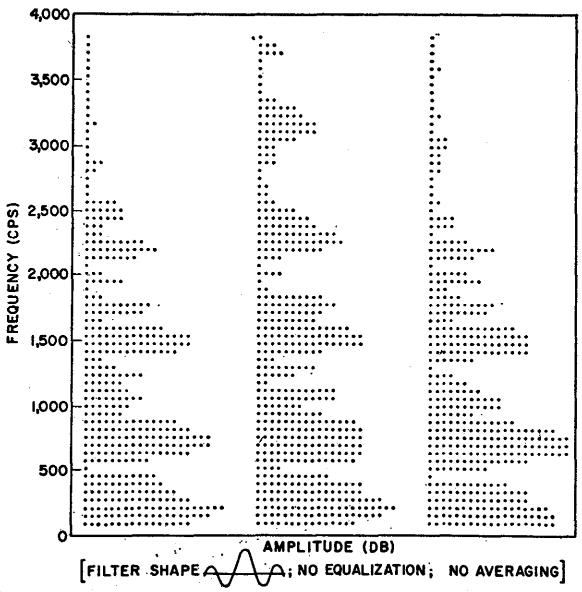


Fig. 9—Spectra produced by original program.

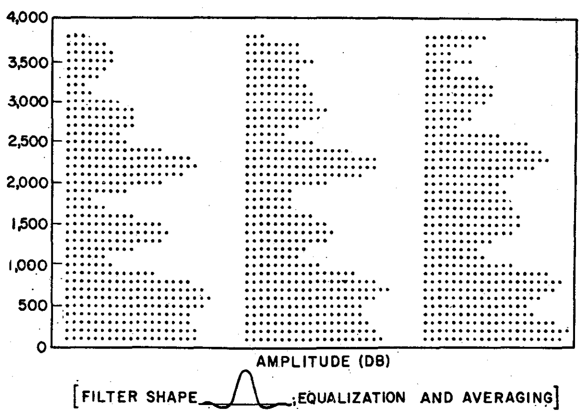


Fig. 10—Spectra produced by modified program.

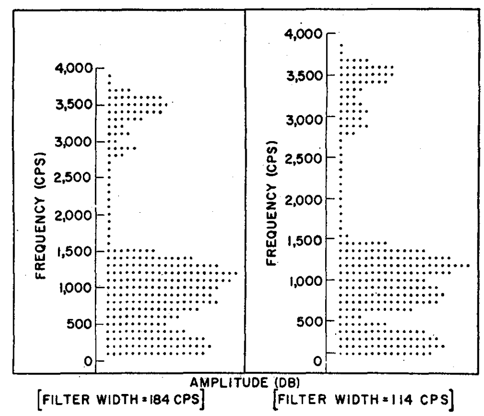


Fig. 11—Spectra compared for two filter widths.

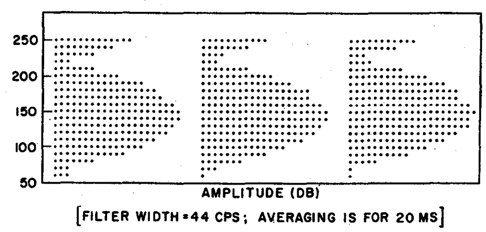


Fig. 12—Use of spectrum program for pitch finding.

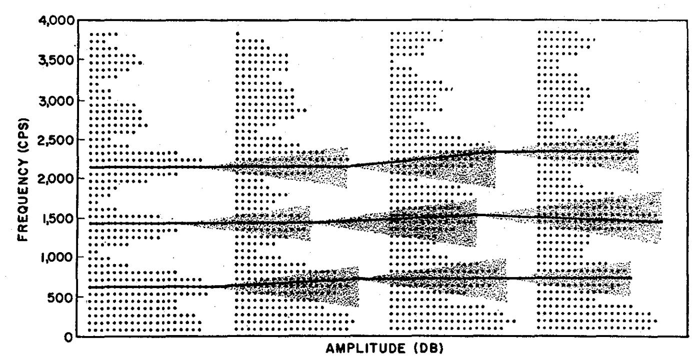


Fig. 13—Formant tracking.

program was so written that the number of filters, the center frequencies of each filter, the shape of each filter, the averaging time, and the sampling rate could be modified by program parameter cards.

The flexibility of this arrangement was of great advantage in obtaining curves that provide a much-improved display of formant structure. Spectra resulting from the program as originally written are shown in Fig. 9. The filter widths were 200 cps and the weighting function was a rectangle which produced a high side-lobed frequency response. Note that formant structure is masked by the additional contributions of these side lobes. Furthermore, formant position with time is not uniform since there was no time averaging. Finally, the amplitudes decrease with higher frequencies, making the third formant quite low. By contrast, Fig. 10 shows a much-improved display. Here, a cosine weighting function produced a filter frequency response with low side lobes. The formant structure is well defined for this 184-cps width. Time averaging of 20 msec resulted in a smooth flow of formant position with time. High-frequency emphasis of 6 db per octave above 1000 cps yielded a well-defined third formant. The general equations computed by this program are listed in Appendix I.

Further investigations of filter widths were made using this modified program. Fig. 11 compares a spectra obtained with filter widths of 114 cps and 184 cps. The (∞) sound has a first and second formant so closely situated in frequency as to become one broad peak for filter widths of 184 cps. The 114-cps width resolves this broad peak into the two formant peaks.

We are continuing to modify and improve this program. Conceivably, an ultimate filter might be varied dynamically in accordance with the speaker's pitch so as to provide the best display for each speaker. Fig. 12 represents a pitch-finding effort already made. To get the display shown, we used the spectrum-analysis program described to simulate a filter bank of closely-spaced, narrow-width filters.

The first feature program was written to compute the low-order moments of the spectra. This concise mathematical computation (Appendix II) permitted the system of programs to be tested while our input equipment was being constructed. Studies indicated that moments might provide a means of distinguishing the front from back vowels. A sufficient number of samples has yet to be tested.

In contrast to the mathematical statement of the moment program, a formant-tracking program can be defined only after a series of program modifications has been made based on experience of a rudimentary program.

Such a "starting" program is now being written. Fig. 13 depicts the procedure followed in this program. First, a peak is defined. Then, the peaks are examined further to determine what peaks are considered formants. In order to detect a "bar" structure, the peaks must be tracked with time; that is, after a major peak is located in a given spectrum, it must be confirmed in succeeding spectra before it is established as a formant. We can emphasize this by pointing out that only after a threshold of formant duration has been determined through experience, can we ignore spurious indications such as the third peaks appearing here in the second and third spectra. By taking full advantage of the 704, we hope to evolve a highly-detailed, formant-extraction method.

The final stage of our system is the statistical evaluation program. For a collection of measurement values, the statistical evaluation program, as it is now written, can develop the frequency table, can sample mean and standard deviation, cumulative probability function, and probability density function. The term evaluation will have more meaning as experience with this program grows. For example, when we find ourselves consistently performing further data reduction, and routine evaluation tasks, then these tasks should be inserted in the statistical program.

Conclusion

I have assumed, here, the role of correspondent, reporting the result of the highly cooperative effort of my associates, Messrs. Welch, Wimpress, and Wilser. During the past year, we have built the equipment and written a first system of programs. This effort has been supported in part by the Office of Naval Research.

It is our belief that this system of analysis will provide an efficient means of experimental study. Through this study we hope to contribute to the understanding of speech.

Appendix I

The spectrum-analysis program solves the following:

$$A_i = 10 \log_{10} \left\{ \frac{E_i}{N} \sum_{j=1}^{N} \left[\left(\sum_{k=1}^{S} \sigma_k W_k \sin \frac{2\pi k f_i}{R} \right)^2 + \left(\sum_{k=1}^{S} \sigma_k W_k \cos \frac{2\pi k f_i}{R} \right)^2 \right] \right\}.$$

The power-frequency characteristics are determined in this equation through the simulation of the power outputs from a bank of band-pass filters. The weighting functions of the filters are products of the function, W_k, which determines the filter shape, and the sinusoid, $\sin (2_\pi k f_i)/R$, which determines the location of the pass band. S is the segment, or summation, interval in input-time samples, and σ_k is the ac amplitude of the kth input-time sample of the acoustica signal. The filter power output, A_i, for the ith frequency, f_i, is averaged for N segments of speech, which corresponds to approximately two pitch periods (20 msec). These coefficients A_i are further modified by E_i, the frequency equalization factor. R is the sampling rate in samples per second. The equations used to compute the two filter shapes discussed are:

Shape	W_k	*Width**
Rectangular	1	$\frac{R}{2S} \sim f.$
Cosine	$1/2\ [1 + \cos(2\pi k/S - \pi)]$	$\frac{R}{S} \sim f$

* Width between 3-db points of main lobe.

Appendix II

The moment program solves the following:

$$M_p = \frac{\sum_{i=1}^{N} (Q_i - Q_0)^p A_i}{\sum_{i=1}^{N} A_i},$$

where M_p is the pth moment about point Q_o in a spectrum of Q equidistant coefficients, A_i, located at points N_i.

Discussion

M. Martin (General Electric Co.): What is the sampling rate used? How many samples are used with each filter to determine the frequency spectrum?

Mr. Shultz: The audio is sampled at the required 727-tape character rate of 16 kc. Since the two audio tape systems each have two speeds, *effective* sampling rates of 8, 16, and 32 kc can be achieved. The number of samples depends on sampling rate and filter width. For the 44-cycle filter for extracting pitch and for a sampling rate of 8 msec, 160 samples are required.

J. R. Barley (Du Pont): Can you transfer the numbers 0 to 9 to digital tape?

Mr. Shultz: Yes. Speech events from 30 msec to 5 seconds can be edited from continuous speech. A microphone input has been provided on the CODER to allow direct conversion of speech to digital tape.

L. S. Bearce (U.S. Naval Research Lab., Washington, D.C.): Would you comment in regard to the feasibility and practicality of a speech recognition system that would operate in real time?

Using your program in the 704, how much increase, if any, in computing speed and complexity do you think will be necessary?

Mr. Shultz: A bank of analog filters would operate in real time. Once we have completely investigated and specified a filter bank through this program, then it might be desirable to shorten our analysis time by building a bank of analog filters.

The spectrum analysis program produces spectra at a rate of 200 times real time. For example, sound of 200-msec duration would require 40 seconds of computation. This delay is due largely to serial computation of the power coefficient of each frequency.

An Automatic Voice Readout System

C. W. POPPE† AND P. J. SUHR‡

THE rapid development of electronic data processing systems has created a need for improved man-machine coupling equipment. In systems where the human operator is part of the control loop, the need for optimizing this coupling is even greater. To date, most display devices have utilized the human sense of sight. Additonal opportunities for improving the man-machine coupling lie in the use of the other senses, particularly the sense of hearing. The automatic voice readout systems (AVRS) described here makes use of this sense of hearing. This permits the operator the simultaneous use of his sense of sight for other forms of data display.

Typical Application

One example of an application of the AVRS is in air traffic control systems. Such systems are, in reality, closed loop control systems and many of the well-known closed loop techniques may be applied to the analysis of them. Fig. 1 shows a very simple example of the complete loop.

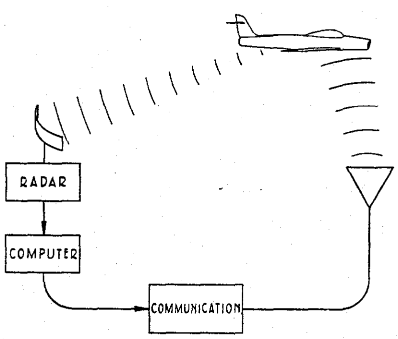


Fig. 1—Closed loop control system.

The position of the aircraft is obtained by the radar and fed to the computing element. The computing element, depending on the system, may range anywhere from a sophisticated electronic system to the human controller. The guidance commands which are outputs of the computing element must be communicated to the pilot for the necessary aircraft position correction. As automatic computers such as form part of the FSQ-7, GPA-37, or TSQ-13 systems become operational, the capability to control large numbers of aircraft at high data rates requires automatic, high speed communications. Where these systems are being used today with human operators relaying guidance commands, over-all system performance may degenerate due to time lags or operator error. This is one of the big reasons for the great interest and activity in the development of automatic data links.

† Fairchild Camera and Instrument Corp., Syosset, N.Y.
‡ Marine Div., Sperry Gyroscope Co., Roosevelt Field, N.Y.

History

During the development of the Tactical Air Control System, AN/TSQ-13, at Air Force Cambridge Research Center, a project was initiated to develop a voice readout unit for the digital data display [1], [2]. This unit gated together previously recorded words to form the desired voice message. As the readout unit progressed, it was felt that as a computer readout it could also be used to transmit the output of the ground-controlled-intercept-return-to-base computer to the aircraft in the absence of the digital data link. One problem always present in the system tests was the shortage of digital data link-equipped aircraft. It was and still is obvious that it will be many years before the majority of our operational aircraft are so equipped. For this reason an automatic voice data link which requires no additional equipment in the aircraft can be used to match the ground data processing equipment, now becoming operational, to the present day operational aircraft. In addition, it will serve to provide automatic data link capability for those aircraft which, for various reasons, may never be equipped with digital data links, as well as a back-up for the digital data link in the event of equipment malfunction. It should be pointed out here that the automatic voice data link does not significantly decrease the bandwidth nor speed up the data rate over that of the manual operator system, as do the digital links. It does, however, provide the same automatic, error free advantages as the digital links while providing improved intelligibility over the manual link. Moreover, the innate redundancy in spoken languages provides the ability to transmit through some fading and certain forms of jamming.

Returning briefly to the discussion of the history of the automatic voice readout system, a breadboard unit was constructed at AFCRC and flight tests were run at L. G. Hanscom Field in mid-1955. Reports from pilots taking part in these tests showed that the clear enunciation of the words and the crisp concise message made for more intelligible and efficient communication. During the last two years improved techniques have been incorporated into a model of this automatic voice readout system at Fairchild Controls Corporation. It is this equipment which is discussed here.

Description

The inputs to the automatic voice readout system are the output commands from the computer. In some cases these are serial digital or parallel digital, while in other cases they may be shaft position or analog voltage outputs. The readout system converts these inputs into a series of messages, the variable portions of which are controlled by the inputs. These messages are then transmitted to the aircraft

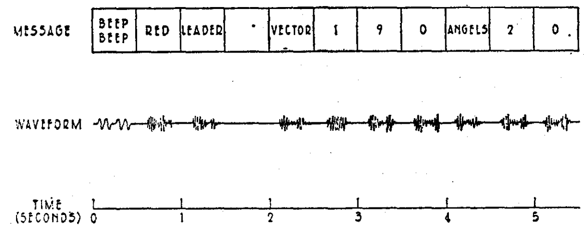


Fig. 2—Message structure.

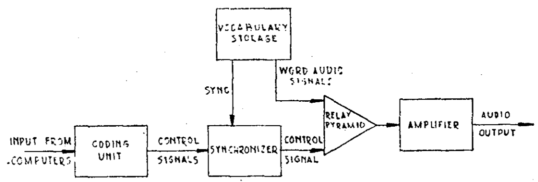


Fig. 3—AVRS block diagram.

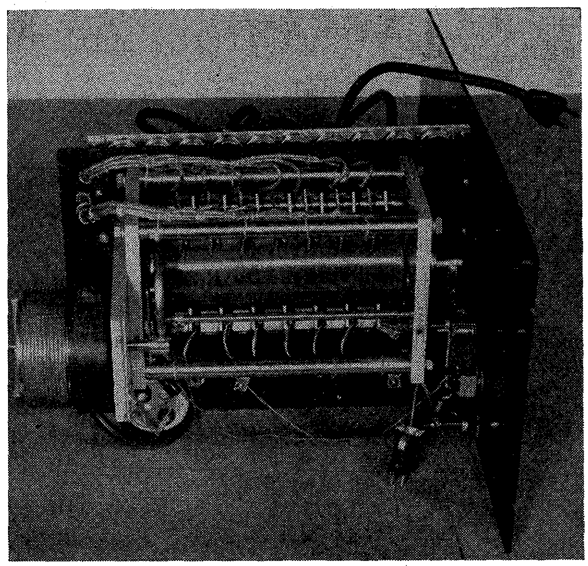


Fig. 4—Vocabulary storage unit.

using a standard voice channel transmitter such as the GRC-27. Fig. 2 shows a typical message structure. It is composed of words, one-half second or less in duration. An unlimited number of message formats may be programmed and the variable portions controlled by the input signals. In the example chosen, the aircraft address illustrated as red leader, the command heading shown as one-nine-zero degrees and the altitude shown as 20,000 feet are the variable portions of the message. In addition, one or more message formats may be selected by the computer outputs so that in addition to a command-heading-command altitude message, a range-bearing message, a landing assignment message, or any other prescribed format may be transmitted. The double beep tone transmitted at the beginning of each message serves to alert the pilot and to provide an indication of the message beginning. Fig. 3 shows a block diagram of the basic voice readout equipment. It is composed of five basic units.

The word storage unit is the heart of the system. It is shown in the photograph of Fig. 4. It consists of a magnetic drum rotating at two revolutions per second and a set of recording-playback heads. The drum is approximately two and one-half inches in diameter, four inches long and has a nine ten-thousandths thickness coating of magnetic oxide. The magnetic heads are spaced one and one-half thousandths from the drum face and may be stacked, ten tracks per inch. Such stacking allows about 35 tracks in the full drum length. Storing one word per track provides a vocabulary of 35 words. The basic numerals, "zero" through "nine," the programmed words such as "vector," "angels," "range," "bearing," and twenty call signs can be included in the 35-word vocabulary. It is to be noted that vocabularies up to 100 words are possible on a single drum. Such large vocabularies would be required where large numbers of aircraft call signs were to be used.

All words stored in the vocabulary are recorded in phase so that during any one-half second interval all words are being reproduced simultaneously at the magnetic heads. This then allows the sequential switching between heads to form the complete message.

The switching between heads to form the message is performed by a relay pyramid. The five digit control signals required to control the pyramid are supplied from the coding unit and the synchronizer. The output of the relay pyramid drives the audio amplifier which in turn feeds the transmitter or other output devices. The audio amplifier is of standard design requiring only the equalization necessitated by the magnetic playback heads.

The synchronizer serves as the basic timing unit within the system. Synchronization pulses are received every drum revolution, or ½ second, and are used to advance the synchronizer to the next word interval. The simplest synchronizer consists of a six-bank, twelve-position stepping switch. The incoming control signals from the coding unit are thus sampled in sequence and the programmed words such as "vector" and "angels" are sampled at the proper interval.

The coding unit will vary considerably depending on the types of computer outputs which must be accepted. These must be converted into the five digit parallel code used to drive the relay pyramid. Diode matrices are used to provide the digital coding. Coded commutators may be employed where the computer outputs are shaft positions.

An additional unit is generally required where several computers are feeding the same voice data link. This sampling unit which is not shown in the block diagram must sample the various outputs and feed them one at a time to the coding unit. It is conceivable that the computer outputs could be sampled, coded and transmitted in sequence, but in all probability a priority system will be required. This will allow more efficient use of the command channel since it will be necessary to transmit only when a command is required. As an example, the original test model voice data link was programmed to transmit only when a heading change of ten degrees or more was required. With such a system, it is believed that up to six aircraft may be controlled over a single channel. It is worth noting that a multichannel voice data link (*i.e.* a system capable of transmitting several different channels simultaneously) could use a single word storage unit.

Optical Unit

One current program at Fairchild Controls Corporation is development of an optical vocabulary storage unit. Such a unit employing film strip for audio storage would provide an easy means of changing vocabularies by simply inserting a different film disk. The size and weight of the unit would also be reduced.

Applications

A large number of additional applications exist for the automatic voice readout system. In air traffic control work, it may be used to advantage with GCI, Return-to-Base and GCA computers. Where voice communications *must* be used but transmitted from a remote site, the commands may be transmitted digitally from the command center to an automatic voice data link located at the remote transmitter site. The voice messages are then transmitted to the aircraft. The resultant system would greatly reduce the number of land lines or voice channels required from command center to remote transmitter.

Another important use could be in a multilingual area. Vocabularies could be stored in several languages for use in areas, for example, where NATO forces were stationed. The complaint against the Frenchman's English or the American's French could be eliminated.

The original concept of this idea is still an important application. In many radar data processing systems, cathode-ray-tube displays are used to read out computer data. Other data can be obtained upon request and may be displayed on an auxiliary status board. To obtain such data, the operator generally must divert his attention from his primary display. Often his dark adaptation is disturbed. The reading out of this auxiliary data through a voice readout would eliminate these undesirable points.

A combination data readout and communication application is the transmission of weather conditions from unmanned weather stations. Such stations, located on planned flight paths, could transmit temperatures, barometric readings and other important weather data to enroute aircraft either on interrogation or at periodic intervals.

It is believed that these applications represent only a few of the possible applications of analog and digital to audio conversion systems.

Bibliography

[1] Krauss, B. F. "An Oral Readout System for Digital Data," Master's thesis, Massachusetts Institute of Technology, Cambridge, Mass., May 23, 1955.
[2] Pastoriza, J., and Poppe, C. "Audio Presentation of Aircraft Data," Air Force Cambridge Research Center, Bedford, Mass., Systems Digital Group Technical Proposal No. 4, June 21, 1954.
[3] Begun, S. J. *Magnetic Recording.* Murray Hill, N.J.: Murray Hill Books, Inc., 1955.

Experiments in Processing Pictorial Information with a Digital Computer

R. A. KIRSCH†, L. CAHN†, C. RAY†, AND G. H. URBAN†

I. Introduction

In almost all digital data processing machine applications, the input data made available to the machine are the result of some prior processing. This processing is done manually in many applications. Thus, such inputs as punched cards, magnetic tape, and punched paper tape often are the result of a manual processing operation in which a human being is required to inspect visually an array of printed characters and to describe these data in a form capable of being processed by machine. In recognition of the importance of automating such operations, many investigations have been undertaken to devise automatic character sensing equipment. Suppose, however, that we attempt to view such efforts in proper perspective. We find a more fundamental problem that has, heretofore, failed to receive the attention that it warrants. The problem is one of making directly available to a computer pictorial information which would ordinarily be visually processed by human beings before being fed to a data processing system. This pictorial information may range from such highly stylized forms as printed characters, diagrams, schematic drawings, emblems, and designs through less stylized forms in cartoons and handwritten characters to such highly amorphous forms as photographs of real objects, *e.g.*, people, aerial views, and microscopic and telescopic images.

† National Bureau of Standards, Washington, D.C.

In recognition of the importance of pictorial sources of data for a data processing system, experiments were undertaken at the National Bureau of Standards to determine whether automatic processing techniques might be applied to pictorial information in order to reduce the amount of human intervention required during the input process. In considering this problem, new areas of the application of automatic data processing techniques for processing pictorial information have appeared. It had not been suspected that automatic data processing techniques were applicable in some of these areas, even if human intervention were allowed. The type of information with which these investigations are concerned ranges from the stylized to the amorphous forms previously mentioned. In the NBS experiments described in this paper, the equipment used consists of the general-purpose digital computer SEAC, to which are attached an input scanner for sensing pictures and copying them into the computer memory, and a cathode-ray-tube output display for reproducing processed pictorial information from the computer memory.

II. Description of the Experimental Equipment

A. SEAC

The experiments described here were performed on SEAC during a period when the capability of the computer for performing logical data processing operations was being enhanced by the addition of several new features. The state of SEAC at the time of most of the experiments described here was that of a 1500-word memory computer with an average time of 250 microseconds for performing a three-address instruction. Although faster computers exist, SEAC was found to have one decided advantage over these machines, namely, its availability for experimental use and modification on some frankly exploratory ventures. The restriction of having to account for every minute of use on a more powerful machine would have been a serious deterrent to the production of the experimental results described here.

B. The Scanner

In order to feed pictorial information into SEAC, it was considered adequate to construct a simple mechanical drum scanner which could digitalize the information in a picture and feed it into SEAC in a few seconds. The scanner is shown in Fig. 1. The photograph to be scanned is mounted on a drum about two thirds of an inch in diameter. As the drum rotates, a photomultiplier and a source of illumination mounted on a lead screw progresses along and scans the whole picture with a helical scan. The pitch of the lead screw is such that the photomultiplier assembly progresses 0.25 mm along the picture for each revolution of the drum. Between the drum and the photomultiplier, and in the image plane of the optical system, there is an opaque mask with a square optical hole of such a size that a square area, 0.25 mm on a side of the picture, illuminates the photomultiplier at each instant. A strobe disk mounted on the same shaft as the drum produces optical pulses each 0.25

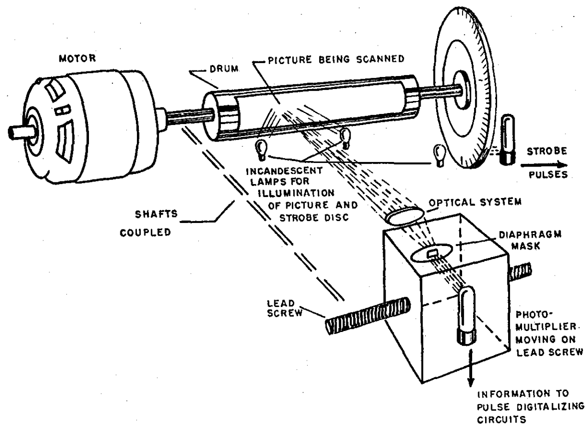


Fig. 1—The scanner.

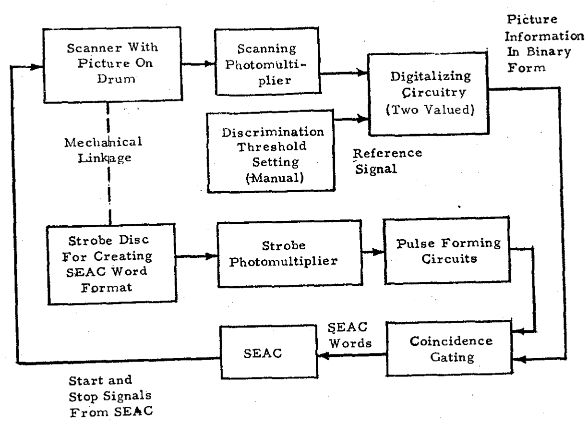


Fig. 2—The scanner connections to SEAC.

mm of drum rotation. These optical pulses are arranged in the format of SEAC input words, *i.e.,* multiples of 44 binary digits. The time for the scanner to scan one photograph is 25 seconds.

C. Method of Input

The scanner was first connected to SEAC in November, 1956. As far as SEAC is concerned, the scanner is just another input device, and it may be selected by the computer interchangeably with such other input and output devices as a printer, magnetic tapes, etc. This is shown in Fig. 2. At any time during the operation of a program if the input of photographic information is called for, SEAC starts the drum rotating. The analog signal from the scanning photomultiplier is compared with a dc reference signal that has been manually determined with a potentiometer setting. If the light reflected from the 0.25-mm square being scanned is less than that needed to produce a signal equal to the reference signal, then when a strobe pulse occurs, a binary 1 is fed to SEAC. If a sufficiently white spot is being scanned, a binary 0 is fed to SEAC.

The result of this operation is that in 25 seconds (or less) SEAC can, upon demand, call for all (or any part of) a picture to be fed into its memory. The whole picture is 44 mm by 44 mm and is thus digitalized into 176 by 176 or 30,976 binary digits, each binary digit representing the blackness of a unit square 0.25 mm by 0.25 mm in the picture. The elementary squares cover the whole picture and are nonoverlapping. The entire picture with one binary digit per square occupies 704 words of SEAC memory. One way of recognizing several different levels of grayness is to use several scans of the picture made with different manual settings of the discriminator threshold. The mechanical precision of the equipment is such that on successive scans of the same picture the scanner reproduces its scan with a maximum discrepancy of less than 0.25 mm at any point in the picture.

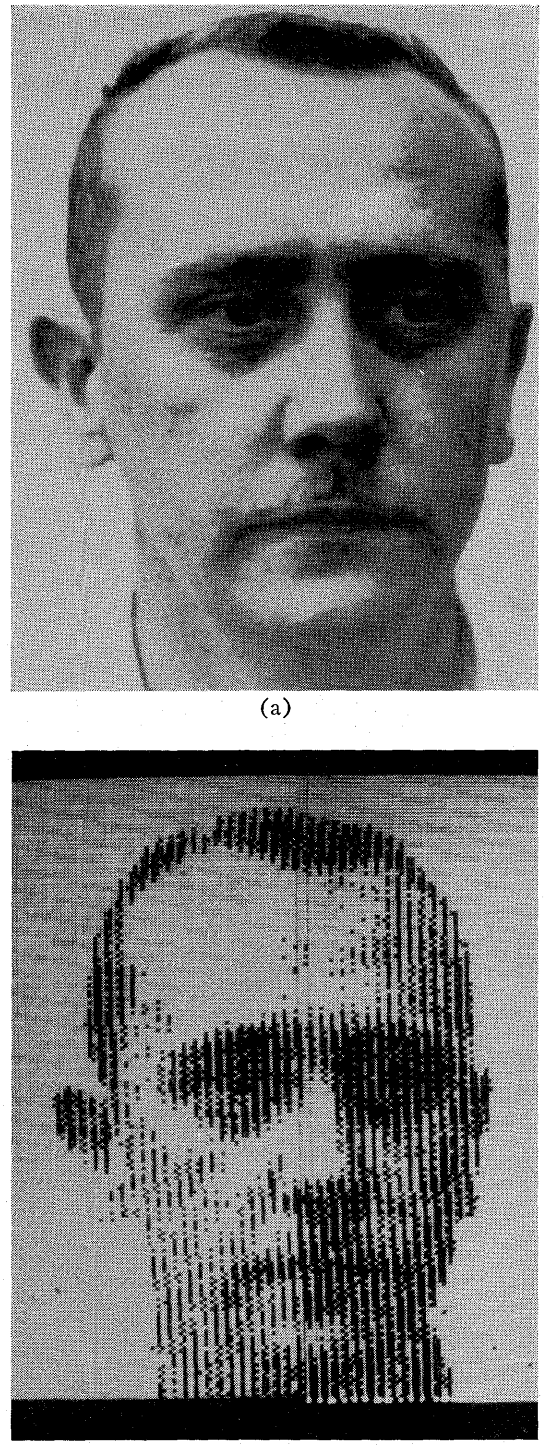
D. Method of Output

As soon as the first picture was fed into SEAC, an uncomfortable fact became apparent. SEAC could store pictorial information in its memory, but the machine users could not "see" the picture in the SEAC memory except by the very time-consuming procedure of printing the contents of the computer memory on a typewriter and attempting to interpret the numerical information. Fortunately, however, there was conveniently available a piece of equipment well suited to the task of producing pictorial output from the SEAC memory. This equipment was capable of decoding two ten-bit fields in a SEAC memory word and producing two analog voltages corresponding to the two sections of the word. There could be up to 96 such words decoded. The rate of presentation of the output signals was 23 kc.

The obvious course was to connect these two analog voltage outputs to the horizontal and vertical inputs on an oscilloscope and thus to plot points on the face of the scope corresponding to the information presented by a computer program for purposes of display. As it turned out, the display equipment was more than adequately fast for SEAC, since no program could generate useful display information that would change at a 23-kc rate.

To display a picture that had been fed into the computer from the scanner, it was then necessary to write a program which derived a pair of coordinate numbers for each binary 1 in the picture in the SEAC memory in such a way that these coordinates corresponded to those of the point in the picture from which the binary 1 was generated. The program then displayed a spot on the output scope corresponding to the spot on the original picture.

As an example of the use of the output display routine, Fig. 3(a) shows a picture fed into the scanner and Fig. 3(b) shows the same picture reproduced from the output display. To produce this picture, an artifice was used which allows the visual effect of a continuous gray scale to be produced on a single scan. As seen in Fig. 2, there is a manual discrimination threshold setting. Ordinarily, this setting determines the level at which black is distinguished from white. To scan the picture of Fig. 3 this discrimination threshold was varied with a sawtooth waveform at a frequency approximately one half that of the strobing frequency. The result was to produce the familiar "halftone" effect in which the density of uniformly black spots is proportional to the blackness of the original picture. The picture display routine produced the output photograph.



(a)

(b)

Fig. 3—(a) Grey-scale photograph. (b) Computer halftone reproduction of (a).

III. Experimental Investigations

With the equipment that has been described, a series of experiments were initiated with the goal in mind to program SEAC to recognize patterns of the type recognizable by human observers. As a first step toward this goal, it was decided to produce a library of picture processing subroutines which an investigator could use in programming pattern recognition. This section describes the library of routines that were written.

In pattern recognition the aim is to reduce the amount of information to the minimum necessary for recognizing one pattern from a group of patterns. In these experiments the approach was to develop a library of computational processes for simplifying patterns in order to obtain their most significant features. Preliminary experiments were concerned with determining those manipulations which would prove to be the most informative. The compilation of discrete routines for the performance of these elementary manipulations would provide the basis of a flexible system for simulating many widely diversified pattern identification logics. After determining the intended course of his pattern analysis, the programmer needs only to refer to this file and to select those routines which in combination will best serve his purpose.

Furthermore, it should be possible for the analyst to sit at the computer console and to draw from this tape library several routines which he will select pragmatically after studying the results of any preceding operations, and thus guide the computer step by step toward recognition of the pattern being studied. We distinguish two forms of output in these routines—numerical data and transformed pictures. Numerical data can be read directly from the computer but pictorial information must be converted before it can be displayed by the picture output scope. The picture display routine is used in such cases.

One of the simplest routines in the library counts the total black area in a pattern. This program examines each bit in sequence and tallies when the bit represents a black area. Advantage is taken of the fact that in many patterns there will be numerous words that are all black or all white. By comparing whole words against constants of all zeros or all ones, much time is saved. The area counting process requires approximately thirty seconds on SEAC.

Pictorial information is an extremely informal sort of information to feed into a computer; it is not stylized in the same sense as numerical or alpha-numerical information. This is one of the fundamental differences that must be faced in operating upon pictorial information. For one thing, pictorial information often contains a good deal of redundancy. The pictures obtained from the SEAC picture input equipment require about 30,000 bits of computer storage. However, the total number of bits of information in the pragmatic sense of the term is probably somewhat less than 100. In view of this fact, a routine was written to provide the data necessary for efficient encoding of pictures. The routine analyzes pictures and tabulates the number of runs of each length of continuous black or white points as they appear in the picture. On the basis of these run lengths it is hoped that we will be able to determine an efficient method of statistical encoding for these pictures.

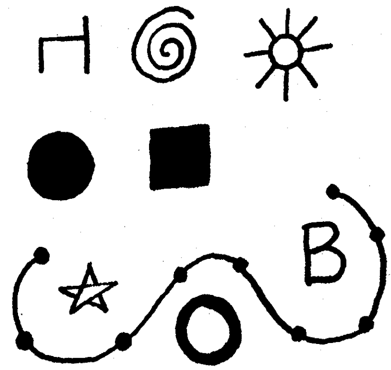


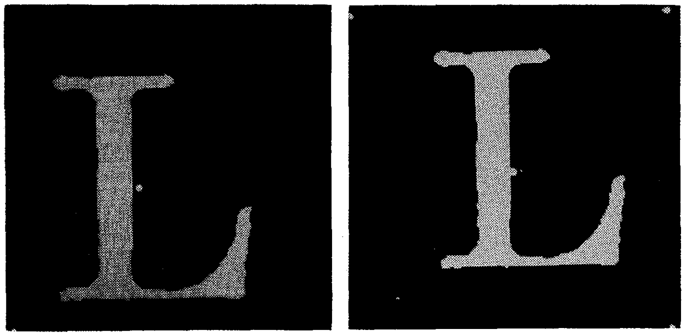
Fig. 4—"Blobs" counted by SEAC.

Object	No.	Area (0.25 mm squares)
	1	56
	2	163
	3	142
	4	187
	5	206
	6	82
	7	404
	8	142
B	9	80

Fig. 5—Areas for each of the objects in Fig. 4.

One possibility is the use of the so-called Shannon-Fano codes [5]-[8] which assign symbols of varying lengths to the different run lengths to minimize the total code length required for any given picture. Preliminary investigation shows that code compressions of the order of at least from 6 to 10 times are possible.

Another routine counts the number of separate noncontiguous black objects (blobs) and measures their separate areas [9]. Due to the restricted memory capacity of SEAC, the routine cannot analyze a full-size input image. Therefore, the image is compressed in the horizontal direction by a factor of four. The routine is devised to scan sequentially through the image until a black point is found, then to move systematically through the blob, counting its points and erasing them until the blob is completely removed from the image. Because the compression in one dimension is not linear, the computed areas are only approximately proportional to the original areas. The blobs are traced so that objects with re-entrant profiles and nonsimply connected objects will be recognized as single objects. The time required for SEAC to count blobs in an



(a) (b)

Fig. 6—(a) Letter L with center of gravity. (b) Letter L translated.

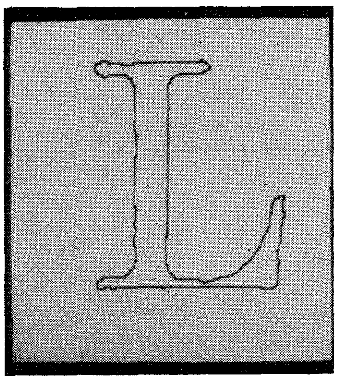
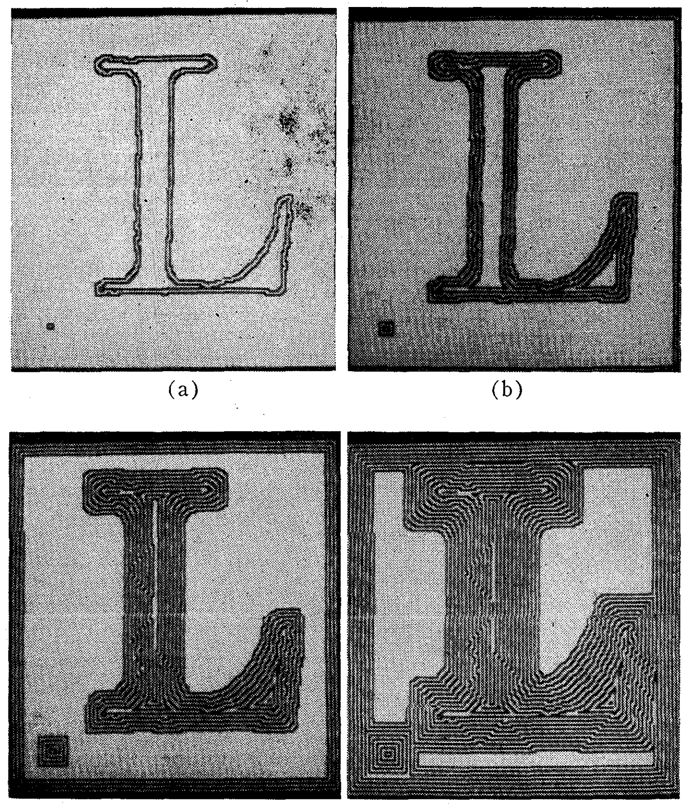


Fig. 7—Boundary of letter L.



(a) (b) (c) (d)

Fig. 8—(a) Letter L custered and complemented two times. (b) Letter L custered and complemented four times. (c) Letter L custered and complemented eight times. (d) Letter L custered and complemented sixteen times.

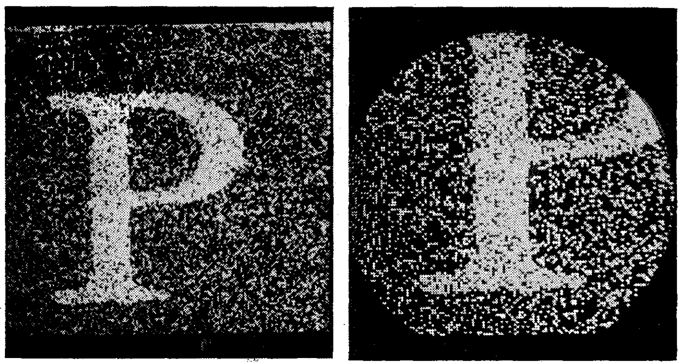
image depends upon the number of blobs and their total areas. An example of the use of the blob counter on the picture of Fig. 4 is shown in Fig. 5, on the preceding page. The area of each blob has not been corrected for the factor of 4 compression.

A second blob counter, which operates with full precision but does not give the areas of the individual blobs, has been planned, but not yet written. This routine examines each bit and its immediate neighbors, generating a tally of each new occurrence of a blob and adjusting when parts of blobs merge or diverge. Nonsimply connected blobs require special checking. There is no limit to the size of the image that can be handled by this routine. The time required on SEAC to process any single pattern from the scanner would be one minute.

Another simple code computes the center of gravity of a pattern and translates the pattern rectilinearly so that the center of gravity is at the midpoint of the image. Fig. 6(a) shows a pattern before translation and Fig. 6(b) shows it after translation. The boundary of the square and the center of gravity are shown as bright spots in the pictures. The translating routine can use any given set of coordinates to determine the shift.

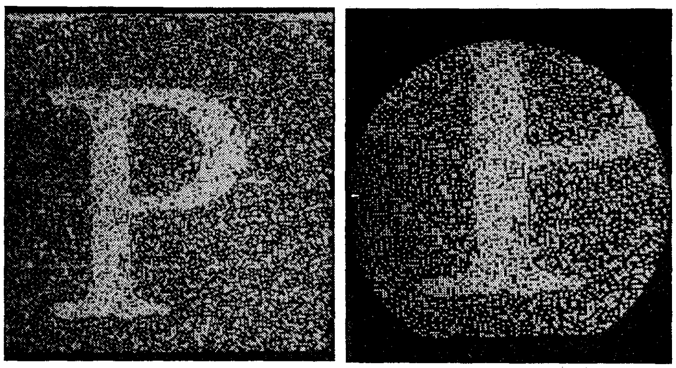
Fig. 7 illustrates the result of a routine which computes what might be called a "first derivative" of the pattern. Each 3-bit square is examined; if all nine bits are black the center bit is replaced by a white bit. The computer operates on three rows at a time, forming logical products of all combinations of each bit with its adjacent bits. These products are, in turn, logically multiplied to produce the final result in which black bits remain only when one or more neighboring bits are white. The effect of this so-called "custer" operation is to preserve the boundaries of a pattern and to erase all the internal areas [10]. Fig. 7 shows the result of computing the boundaries of the letter in Fig. 6. Notice that the boundaries contain most of the significant information of the original picture but require fewer bits. The time required for one "custering" operation is seven seconds.

It was suggested that a thin line representation of certain patterns could be obtained by computing the boundary, reversing the image (*i.e.,* forming the binary complement), recomputing the boundary, etc. Upon consideration it became obvious that this treatment would not produce the proposed result, but the idea aroused considerable curiosity as to what it would produce. As a result of performing this process, it was found, in fact, that the image became more complex and difficult to analyze. In Fig. 8 there are four pictures of the "custered" letter L of Fig. 7 after it was "custered" and complemented a varying number of times. It is interesting to note that the small dust



(a) (b)

Fig. 9—(a) Letter P with random noise.
(b) Expanded section of (a).



(a) (b)

Fig. 10—(a) Fig. 9(a) custered and complemented.
(b) Expanded section of Fig. 10(a).

speck in the lower left-hand corner of Fig. 7 grows much larger in successive pictures, and merges with the L in Fig. 8(d). The next set of figures shows a very surprising result of the custer and complement operation. A program was first written which superimposes controlled random noise on a photograph of a letter P to produce Fig. 9. This noisy letter was then custered 50 times to produce Fig. 10. The "noisy" letter in Fig. 9 is still clearly recognizable in Fig. 10 even after being operated on 50 times. Thus, the "custer and complement" is a process which seemingly preserves the information in a "noisy" picture but degrades the information in a "clean" picture.

In addition to these routines, some other simple routines have been written. These include routines to superimpose pictures, to smear pictures by translating and superimposing them, to magnify pictures so as to make their fine detail visible, to record pictures in permanent form on tapes and wire, and a routine to analyze a set of pictures to determine the number of spots that are always black or always white in the set of pictures.

These routines perform elementary operations, although in some cases, such as "custer and complement," the results of the operation have far from trivial explanations.

Some applied investigations have been undertaken to utilize the potential of this new equipment. Such problems as character recognition, aerial mapping [11], and automatic encoding of the chemical information in chemical-structure diagrams [12] are included.

Much interest has been developed in simulating character recognition logics by some of the preliminary experiments such as those of Greanias, *et al.,* of IBM [2]. The present picture scanning equipment adds to the practicality of simulating recognition logics with computers by providing a convenient source of data input. We plan to simulate some of the logics that have been proposed such as the "peek-a-boo" system, in which characters are recognized by locating the key points in a picture that characterize letters uniquely.

An attempt is being made to program the computer to generate elevation contour lines from aerial photographs. The relative elevation of ground points can be determined from two aerial photographs of the same area taken from different points in space by measurements of the apparent displacement of the elevated points. Some simplifying restrictions were made to reduce the problem to a convenient size for a first attempt. The photographs to be used were assumed to have been restituted to correct for tilt of the focal plane and oriented so that the overlap or shift of ground points occurs only in the x direction. It is much less involved to deal with one-dimensional shifts than the two-dimensional ones.

The contours are determined in three steps. First, similar areas on the two photographs are identified by checking units of 44 bits for the correspondence of at least 39 of the pairs of bits. Groups of bits that are mostly ones or zeros are not compared. Next, the pictures are shifted with respect to each other to bring them into alignment. Finally, all points with the desired parallax are computed and stored in the contour diagram picture. The first two steps of this program have been completed. The parallax determination and the preparation of output photographs remains to be done.

Another intriguing problem is to find a way that a machine could "look at" a diagram, such as a chemical structure diagram, and characterize it uniquely. The work to date has not been concerned with the more symbolic information that appears on structure diagrams, such as element symbols, double bonds, etc. We have attempted to treat only simple nets composed of vertices and bonds drawn between them. The connection pattern has been treated as a topological net and we are not concerned with such things as size of angles, length of lines, width of lines, and line breaks. The program we have been working on will first locate most of the vertices by counting the number and extent of clumps of "black" spots in each line of a picture in both vertical and horizontal tracings. Where these numbers change between successive lines a vertex is indicated. Then, starting at a vertex the bonding pattern could be traced from vertex to vertex. Thus far, the programming is in a preliminary state. The actual coding and handling of the "housekeeping" procedures remains to be done.

IV. Some Unsolved Problems

Thus far the discussion has been concerned with a report of experimental investigations. To those familiar with the application of data processing techniques to new fields, it should be apparent that such experimental investigations generally lead to the formulation of new problems in the two areas of higher performance equipment design and proper utilization of such equipment. The problems in automatic processing of pictorial information that occur in these two areas will be formulated here. To the knowledge of the authors, no solutions to these problems are available.

A. The Development of a High Performance Picture Scanner for Computer Input

In using a digital computer to process pictorial information, it is unthinkable that any large quantity of pictorial information should be scanned and stored on conventional computer storage media like magnetic tapes unless a tremendous amount of reduction of information has first taken place. The maxim that "one picture is worth ten thousand words" is probably overly optimistic for fairly common sources by about three decimal orders of magnitude, if a "word" means 25 to 50 bits.

This estimate is based on a comparison of the number of binary digits needed to describe highly stylized information in pictorial form and in such a form as to describe only the "meaning" conveyed by the picture. This means that in applications where it is either not possible or not practical to encode (and thereby reduce) information in a photograph, the best way to store a photograph is in its original form.

However, to make such information available to a computer, equipment is needed that will mechanically handle photographic information and that will be able to sense the information for input to a computer. The mechanical handling can probably be solved in many ways. Such devices as a microfilm rapid selector might be appropriate.

For the optical scanning of the photographic information, however, it appears that a device with performance somewhat better than conventional cathode-ray-tube flying spot scanners is required. If we assume an average document size of 8×10 inches then the scanner must be able to resolve a field of the order of $10^3 \times 10^3 = 10^6$ spots. Although it is seldom necessary to scan a whole pictorial source with this resolution, any section of a picture must be capable of being resolved with this precision. The data rate for a computer like SEAC should be 1 megacycle. Thus, if it is necessary to scan a whole field, it should be possible to scan the 10^6 bits in one second.

It is anticipated that with such a scanner, the computer would first direct the scanner to locate information. The most straightforward way to do this would be to have a defocused scan or perhaps several different levels of defocusing. Thus a field of $2^{10} \times 2^{10}$ bits might first be scanned with a raster of 32×32 spots. Since each spot would cover in turn an area of another 32×32 elementary spots, it would be necessary to be able to get something of the order of a 10-binary-digit reading of the light value from any spot. It would be unreasonable to expect an *accuracy* in such a reading, but such *precision* would be required. In other words, upon successive scans of the same large square array of 32×32 elementary spots, the same reading should be obtained within 1 part in 2^{10}, even though the reading itself is not accurate. The inaccuracy can be compensated in the computer programs.

In addition to the performance of the actual scanner, it would be desirable that the computer be able to specify certain types of front-end processing which would be done by suitable fast analog equipment, *e.g.*, time differentiation of the scanning signal and insertion of logarithmic response functions. Certain simple analog operations performed on the scanner signal can save a great deal of complex processing by the digital computer.

With a scanner such as this and with suitable mechanisms for motion of the photographs, it would be possible to get the photographic information into a computer at a rate comparable to the present processing rate of the computer. The next problem would be how to use this information.

B. The Effective Use of a Picture Processing Computer

It was stated at the outset of this paper that an aim of the present investigations is to automate some of the visual processing of information done by human beings. The beginning of the processing operation is, logically, the statement of requirements. Therein lies the rub! We can state our requirements to a human being and expect some intelligent performance but we do not know how to do this with a computer program. There seems to be fairly universal agreement among people as to what constitutes a picture of an automobile, a letter "e," or the President of the United States. This ability to recognize patterns is not learned the way the ability to multiply numbers is learned. Most of the pattern recognition ability of people exists on a nonarticulate level. In order to program a computer to duplicate this pattern recognition ability, it is necessary to make explicit the techniques that people use and then instrument these techniques in computer programs [13].

To accomplish this we require an automatic programming technique in which macroscopic patterns can be defined in terms of more simple ones [14]. This type of technique which would assume the nature of an automatic compiler would eventually enable a programmer to describe familiar objects in terms of other more simple but nevertheless familiar objects. Thus a linguistic formalism would be constructed which would continue to approach closer at successive levels of approximation to the formalism used by people in describing pictorial information.

Experiments are being initiated at NBS on the construction of such a compiler, however, there are no results available yet.

V. Areas of Application of Picture Processing Techniques

Much of the work described in this report was motivated by the promise of application in the solution of important problems. Other areas of application have suggested themselves as the work progressed. We discuss below those classes of applications.

A. Analysis of Pictures

In this class of applications it is desired to subject a picture to an analysis, the result of which is to produce some alphabetical or numerical data. The picture itself has, in principle, no value for retention after the analysis. It is desired to abstract some information from the picture and store this.

The first application in this area occurs in information retrieval of the type practiced in the U.S. Patent Office. Documents containing drawings and schematics are to be stored for purposes of subsequent reference by a computer. If a drawing can be coded so that an equivalent one can be reconstructed from the code, this is sufficient for storage purposes. Obviously, such pictorial considerations as the quality of the lines in a circuit diagram need not be preserved in the code. Thus we are led to the attempt to recognize by machine such configurations as chemical structure diagrams, electronic circuit schematics, and drawings of mechanical configurations. These problems show promise of yielding to the type of investigations described here.

Within the same class of analysis applications fall the problems that involve counting objects in a picture. Here we have such cases as the counting of particles in microphotographs of metallic structures, classification of particles in biological preparations, and analysis of tracks of nuclear particles. In the area of astronomy, knowledge comes mainly from photographs taken through telescopes or other instruments and the analysis of these pictures now requires considerable time in order to generate a rather large body of data. Picture processing techniques might be used for such problems as computing star positions and proper motions, evaluating star brightness or magnitudes, and automatically setting up star catalogs.

B. Transformation of Pictorial Information

In this class of applications, information is to be prepared for visual consumption by human beings. Generally, the information is originally in such a form that it cannot be used by human beings. The question will be left unanswered whether human beings may be replaced by automatic processes as visual consumers of the information produced in the applications in this class.

The first such application occurs in photogrammetry. A stereo pair of aerial photographs is to be processed to produce an elevation contour map. By techniques based on principles described, it is believed possible to use a digital computer to generate elevation contour maps. If investigations in pattern recognition proves successful, it may even be possible to superimpose cultural information upon maps, the whole process occurring automatically.

Another application of these techniques to the transformation of pictures was suggested to the authors by M. L. Minsky of M.I.T. Picture processing techniques could be used to develop a good reliable set of photographs for the planet Mars. We know that one of the main reasons that a good photograph of Mars doesn't exist is that there very rarely are conditions of perfect seeing where the entire disk of the planet is clearly visible and all of the details on it are plainly visible.

There are, however, several million frames of motion picture film that were taken of the planet Mars during its opposition. By an analysis of these photographs, abstracting those bits that represent good clear seeing in any one frame and putting them together to form a composite photograph of the disk of the planet, we may be able to get a good reliable map of the true features of the planet.

C. Simulation of Picture Processing Systems

In this class of applications, the digital data processing system in conjunction with its picture input and output is used to simulate the behavior of a system or the model of a system that processes pictorial information.

The most obvious use of such simulation techniques occurs in applications to character recognition studies. Fairly complex character recognition devices can have their behavior simulated by a data processing system. In this way devices can be "flown on the ground" without the necessity of costly construction of apparatus.

A more unusual application of such simulation techniques occurs in the field of experimental psychology. In attempting to explain human vision, theories have been propounded which are subject to analysis by simulation techniques. Many operations that have neurophysiological counterparts can be programmed on the type of research facility described here [15],[16]. It is to be hoped that the eventual use of general purpose picture processing simulation techniques will aid in encouraging the formulation of more ambitious theories of the functioning of the human visual process.

VI. Conclusions

In this paper a new type of research facility has been described which allows complex investigations to be made into the nature of pictorial information and into ways in which computers may be programmed to process such information with the same comparative ease that characterizes human processing of visual information.

The apparatus described here as well as the experiments performed are strictly of a research nature. Consequently no conclusion should be drawn as to the practicality of such processing as is described here. Before such applications become practical it will be necessary at least to solve the type of problems described here, namely, those of the design of high performance scanning equipment and the development of automatic techniques for the recognition of visual patterns.

The applications of automatic pictorial information processing techniques described here are not meant to represent the most important applications that can be anticipated. They are, rather, meant to illustrate typical classes of processing techniques that can be automated if experiments of the type outlined here lead to successful conclusions.

VII. Acknowledgment

The work described in this paper has largely resulted from a project sponsored by the Rome Air Development Center, USAF, and has also been supported in part by a cooperative U.S. Patent Office—National Bureau of Standards program to explore the potential mechanization of patent search operations. The results reported have been achieved through the contributions of a number of individuals. In particular, J. L. Pike and M. A. Fischler, helped to design, build, and debug the scanner. This work was asisted by G. Crowther and O. Hall. Some of the experimental SEAC picture processing programs were written by R. B. Thomas. The interest and suggestions of several other members of the Data Processing Systems Division were also of assistance in the formulation of some of the ideas expressed here.

Bibliography

[1] Glauberman, M. H. "Character Recognition for Business Machines," *Electronics,* Vol. 29 (February, 1956), pp. 132-136.
[2] Greanias, E. C., Hoppel, C. J., Kloomok, M., and Osborn, J. S. "Design of Logic for Recognition of Printed Characters by Stimulation," *IBM Journal of Research and Development,* Vol. 1 (January, 1957), pp. 8-18.
[3] Rabinow, J. "Report on DOFL First Reader," Diamond Ordnance Fuze Laboratories, Washington, D.C., Report TR-128, November 26, 1954.
[4] Stone, W. P. "The Alphanumeric Character Reader—RADC, ARDC, USAF," RADC-TN-56-194, AD 103-237.
[5] Fano, R. M. "The Transmission of Information," Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, Mass., Technical Report No. 65, 1949.
[6] Huffman, D. A. "A Method for the Construction of Minimum-Redundancy Codes," Proceedings of the IRE, Vol. 40 (September, 1952), pp. 1098-1101.
[7] Michel, W. S. "Statistical Encoding for Text and Picture Communication," AIEE Conference Paper CP 57-723, May 1, 1957.
[8] Shannon, C. E. "A Mathematical Theory of Communication," *Bell System Technical Journal,* Vol. 27 (July, 1948), pp. 379-473 (October, 1948), pp. 623-656.
[9] Dell, H. A. "Stages in the Development of an Arrested Scan Type Microscope Particle Counter," *British Journal of Applied Physics,* Supplement, No. 3 (1954), p. 156.
[10] Kovasznay, L. S. G., and Joseph, H. M. "Image Processing," Proceedings of the IRE, Vol. 43 (May, 1955), pp. 560-670.
[11] Strand, O. N. "Mathematical Methods Used to Determine the Position and Attitude of an Aerial Camera," U.S. Ordnance Test Station, 1585, NAVORD Report 5333, March, 1956.
[12] Ray, L. C. and Kirsch, R. A. "Finding Chemical Records by Digital Computers," *Science,* Vol. 126 (October 25, 1957), pp. 814-819.
[13] Selfridge, O. G. "Pattern Recognition and Modern Computers," *Proceedings of the Western Joint Computor Conference* (March, 1955), pp. 91-93.
[14] Dinneen, G. P. "Programming Pattern Recognition," *Proceedings of the Western Joint Computer Conference* (March, 1955), pp. 94-100.
[15] Clark, W. A., and Farley, B. G. "Generalization of Pattern Recognition in a Self-Organizing System," *Proceedings of the Western Joint Computer Conference* (March, 1955), pp. 86-91.
[16] Farley, G., and Clark, W. A. "Simulation of Self-Organizing Systems by Digital Computer," IRE Transactions on Information Theory, No. P61T-4 (September, 1954), pp. 76-85.

Discussion

V. M. Wolantis (Bell Telephone Labs.): Do you find it satisfactory to distinguish only between black and white rather than a number of shades of gray?

Mr. Kirsch: No, that is often not adequate. For purposes of looking at schematic diagrams or printed characters, two values of light intensity are adequate. However, for looking at the most interesting types of pictorial information, people's faces and so on, we should like to distinguish several shades of gray. What I showed you was a poor compromise between having full gray scale rendition and having only the black and white rendition. In the photograph of the person's face, what we were doing was sacrificing some of our resolution in order to get some gray scale information, but certainly ideally we would like to have considerably more gray scale information.

Mr. Wolantis: To what extent would a TV camera help solve the problem you mentioned near the end of your talk?

Mr. Kirsch: A TV camera would help. We would like to scan a million-bit picture in a second, but we also want to be able to deflect to any spot or zone on the picture with an access time of a few microseconds.

For the purposes of using the picture as a computer memory, which is after all what we were proposing that the high performance scanner do, we would want considerable reproducibility of scan. We would want the scanning device to go back and look at the same picture any number of times over the course of a machine computation, and be able to copy the same information; I don't know whether a standard TV camera type of scan is capable of that type of performance.

Mr. Rellis: Can you state time required to execute some of the programs cited?

Mr. Kirsch: Yes. The custer and complement for one cycle takes seven seconds and thirty seconds to take the area of a whole picture. To generate the coordinates of the output point from the information in the memory for a whole picture of 30,000 bits takes about one minute.

The most time consuming of the routines is the blob counter, which is a function of the complexity of the image being blob counted, and here that routine will take anywhere from about a minute to perhaps three minutes, or even more. However, these numbers are a reflection not of the intrinsic complexity of the processes, but rather of SEAC computation time, and an indication of that speed is that the SEAC add time is about a quarter of a milli-second. So you can see that a faster machine certainly would be able to do the processing more rapidly. It turns out, however, that by way of impedance matching the operator's thinking time with the computer, this is not really too slow. The machine can test out an idea at about the rate a person can generate it.

Mr. Rellis: Has any consideration been given to the similarity between multiply-custered black objects on white, and fingerprints?

Mr. Kirsch: Yes, many people have suggested this, and although we haven't done any serious investigation, some people have also suggested that perhaps by doing inverse custering on a fingerprint, one might get a picture of the criminal's face.

Optical Display for Data-Handling System Output

JAMES OGLE[†]

Introduction

DUE TO long experience in the field of business machines, Burroughs has always been alert to the fact that human operators are important links in most computing and data-handling systems. Because of its versatility, this link fills in where unattended techniques are still insufficient, where decisions must be made in problems of extremely variable or unpredictable nature, and where monitoring and manual intervention may be needed due to functional failure.

One type of display device has been developed at the Burroughs Research Center for conveying to operating personnel the output information of data-handling systems. These devices utilize an optical medium known as a lenticular screen. Such a medium consists of a large number of very small lens surfaces, either cylindrical or spherical, placed substantially in one plane of the lenticular screen and having a common focal plane in which are recorded in a space-sharing fashion the image elements of prerecorded messages.

Electronic engineers may be more familiar with the lenticular structure as used in the Lawrence television tube than with optical lenticular screens. The functional similarities are there, but the methods of design treatment, the various degrees of freedom, and the techniques of creating physical embodiments are sufficiently different to preclude considering the electronic and optical lenticulars as equivalent.

There have been numerous commercial applications of the lenticular medium. Its first major appearance was in decorative, dispersing, window glass not uncommon about 50 years ago. Here the cylindrical ribs, 0.1 to 0.5 inch wide, served the purpose of presenting a repeated series of prism angles effecting a transverse dispersion. No precision was required. The same principle is used in some rear-projection screen designs and in single azimuth dispersion of light sources such as sealed-beam lamps.

These applications utilize lenticular screens without need for any precise focal plane and no image information is carried by the screen itself. The minimal precision allows some of these to be made of molded glass. In other uses of the medium, the focal plane of the lenticules has an important function and carries prerecorded image information. Here the additional precision usually prescribes that the lenticular elements be molded or cast in plastic. We will mention a few; their geometry will become clear when we describe the principles of lenticular output-data displays.

† Burroughs Corp., Paoli, Pa.

In one type of three-dimensional photography, the lenticular screen is used for displaying consecutively different aspects of an object as might be seen by walking around the object, each aspect being visible only to an observer's eye viewing the display at the angle from which the aspect was photographed. Within certain audience limitations, the two eyes of the observer can see two different aspects, thus giving true binocular vision of the object. Likewise, a simpler use is that where unrelated images are made to appear consecutively as an observer's angle changes. These unrelated images may be messages which appear to flash on and off, eyes that wink, and other more or less sophisticated material.

Burroughs Lenticular Displays

An optical design is usually interesting in that it can be operated in either direction, in other words, the ray trace is usually reversible. This is the case in the lenticular screen. We note immediately that if an illuminator is substituted for the observer in the last examples given, only that image which the observer was able to see will be illuminated (see Fig. 1). Now if a device is constructed containing at its rear end a number of lamps or a plurality of filaments, there will be associated with each illuminator discrete areas, one for each lenticule, in the focal plane of the lenticules. In the case of cylindrical lenticules, these will be narrow lines; in the case of spherical lenticules, these will be small areas the shape of which will depend upon the configuration of the filaments. Now if the focal plane of these lenticules carries a photographic emulsion and if a stencil or negative is interposed between a light source and a lenticular screen and if this light source is made to occupy consecutively the positions of the filaments in the final device, it will be sufficient to make as many exposures as there are messages, substituting a new negative each time the light source is moved to a new position. An appropriate diffusing screen placed over the developed lenticular will allow the illuminated image to be visible to a sufficient audience.

Technique and Design Considerations

A number of difficulties is encountered in the use of this technique. First is the art of engraving master-die surfaces in order to generate with adequate precision and optical quality the minute surface elements involved. Another is that of designing a system in which there is, in general, only one surface to work with. Ordinarily an optical device, such as a photographic lens, will have a number of air-to-medium surfaces, thus enabling the optical engineer to introduce the necessary corrections to

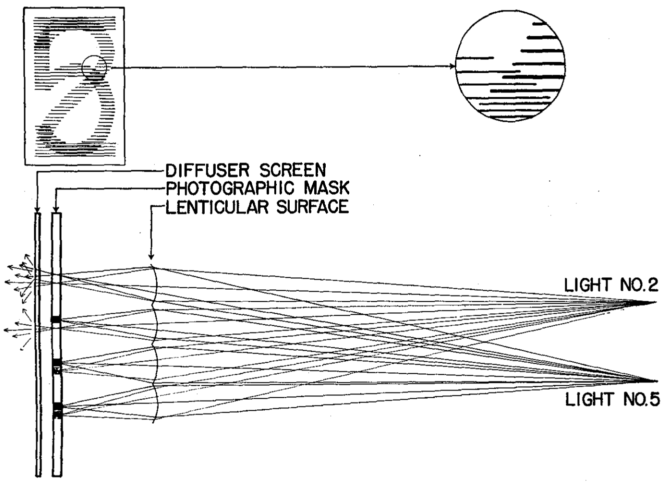


Fig. 1—Geometrical schematic of one-azimuth lenticular digit display. Two of the ten light sources are shown with their associated ray trace and image records. Lenticular screen carries cylindrical surfaces with horizontal axes.

assure that a substantial number of discrete image points can be resolved through the desired pupil. We have found that cylindrically-ribbed (*i.e.,* single azimuth) lenticular media can be made to resolve up to 10 images with adequate lack of crosstalk for a display purpose.

A practical technique for constructing two azimuth lenticular screens consists of placing two cylindrically-ribbed screens face to face with the axes of the cylinder of one screen at right angles to those of the other. The adjoining boundaries of the cylindrical elements now define square cellular pupils within each of which the refractive performance is similar to that of a spherical element (see Fig. 2).

Two azimuth systems of crossed lenticular structure can resolve in excess of 20 separate channels.

We have found that the quality of a screen image as seen by an observer does not yield readily to analysis into quantitative information of brightness, contrast and crosstalk, and that partly empirical designing and direct observation of models are still the most reliable ways of obtaining a conclusive analysis of a new application.

The light-handling capacity of these displays is of some interest because its analysis will largely determine the class of light source which is necessary today. If we consider the image-record plane, it is immediately clear that the luminance of any image area can never exceed that of the source divided by the number of messages. Considering further that the relative aperture of each lenticule is limited by optical-design considerations, we find that a further luminance reduction will occur as the outcoming flux is dispersed in order to satisfy a suitable audience, (except in certain special cases where a restricted audience can be tolerated). Further, certain clearance margins in the optical geometry must be allowed. These must take into consideration the mechanical tolerances of the light sources themselves, the mechanical tolerances of the assembly, the geometrical optical aberrations of the rudimentary lenticules, the phenomena of physical optics (since these lenticules are usually very small), and the remaining imperfections in the manufacture of the lenticules. When all these factors are taken into consideration, as well as the absorption factors which are inevitable in a system involving several surfaces, we find that the only practical illuminators for applications where the ambient light is of an office level (30 to 60-foot candles) are tungsten filaments and that further, for practical reasons, these tungsten filaments should be operated at low enough temperature so as to obtain consistently acceptable life. Typical embodiments specify filament currents from 80 to 200 ma and wattage from 0.5 to 1.5.

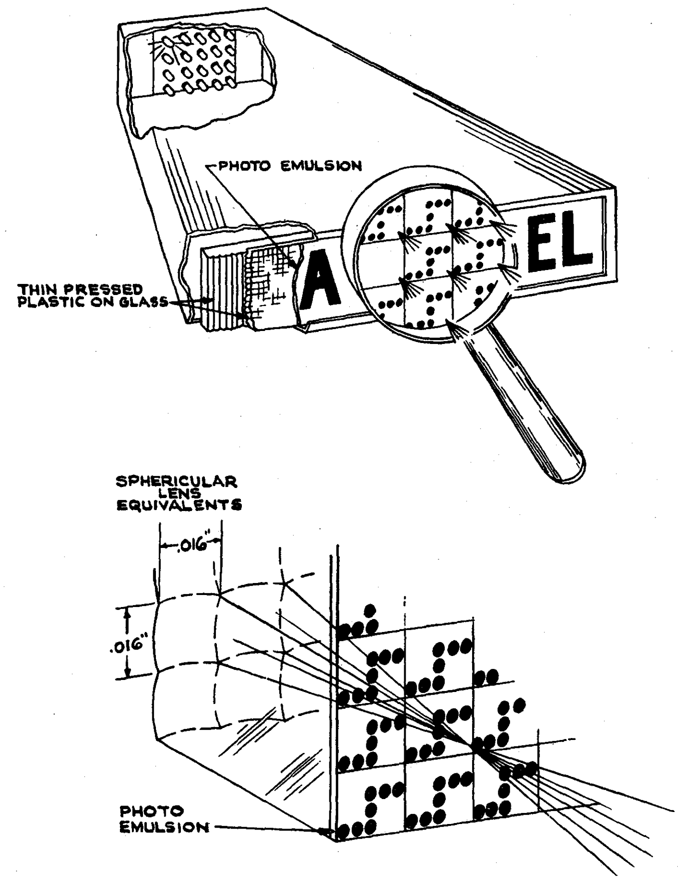


Fig. 2—Two-azimuth lenticular display. Ray trace through screen shown for "equivalent sphericles."

Logic and Economy

The information-handling capacity of such a device is of some interest. If there are N channels or resolved images in each of L "sphericules" in the crossed lenticular system, then the total information recorded in this system should be NL bits. Clearly, the same information could be handled in a system having N objectives provided that each of such objectives was able to resolve L bits of image information. We are familiar with the 35-bit (7×5) minimum matrix for the presentation of an individual alphanumeric character. One also finds immediately that reading reliability and operator acceptance increase rapidly if the number of bits per alphanumeric image is increased, say by a factor of at least four, and continues to improve appreciably until good graphic freedom obtains with a matrix of 20×28.

The economical question then becomes one of determining whether for a particular application it is more desirable to utilize N objectives (in N small projectors) each resolving L bits, or L equivalent sphericules each resolving N bits. We find that a numerical display will be slightly more economical when built with a lenticular than it would be built with a projector battery and that this advantage increases as the amount of information per channel increases.

An example of a large information display developed for a link in the SAGE system is illustrated in (Fig. 2). There are 20 channels in this display. The used screen area is six square inches and each channel carries 3600 bits per square inch. This particular display was developed to replace a battery of signal lights and represents, we believe, a tool which the human engineer can use to good advantage to satisfy what might be called the system requirements of a human operator. Interestingly, the human link in a system is the one least susceptible to redesign. With substantial training, an operator can learn to utilize unfamiliar information codes, but in the final analysis we will always find that, as an information-receiving center, the operator is a noisy device. It behooves us, therefore, to address such a device with signals which will activate the various functional aspects of the individual and also supply a sufficiently large amount of information so that these recognition and reaction functions can operate reliability. Thus, one might argue that the signaling of one out of 20 pre-established messages constitutes relaying information of no more than five bits for each message, whereas the fully flexible tool in this instance utilizes over 20,000 bits and that the device is thus extremely redundant. However, when these highly redundant patterns can be generated basically by one molding operation and one photographic operation for each message, the potential economy is self-evident.

Control complexity of the lenticular displays is naturally greater than that of coded displays. In the previous example some means must be provided to deliver more than five control lines to the display. One solution is to apply power to one out of N (in this case 20) leads. $N+1$ leads must then be brought out of the device. For maximum application freedom we prefer to bring out both ends of each filament. This may allow the user to simplify his decoder. Due consideration must be given to sneak paths. In general, the glow of a filament in a sneak path will be negligible if the voltage across the filament is no greater than one third of the operating voltage. The tolerable sneak path will vary depending upon the type of filament, the type of display and the application requirements. The individual application should be examined. Relay decoders have been used successfully to drive 10-channel digit displays where sneak paths included three filaments in series.

Conclusion

The lenticular medium lends itself to space-sharing display purposes in many applications. If the lenticular art is mastered, good manufacturability can be obtained. Power requirements are not negligible, the devices being best suited to low-voltage, high-current applications. Where electrical requirements can be met, excellent graphic freedom is obtained and this can materially assist the human engineer.

Devices for Reading Handwritten Characters

T. L. DIMOND†

In THE LAST five years, much thought and effort have gone into the development of printed character-recognition devices. Varying degrees of success have been achieved. In some cases, ingeniously distorted type faces have been required. One might wonder why all this interest exists. The answer is simple. Character-recognition devices help reduce the substantial cost of getting information into forms that computers can understand.

However, in creating devices that read printed or even typed characters, we are not reaching back far enough toward the origin of the information in the majority of the cases. Only a little reflection will show that nearly all of the information used by business data processing computers originates in the minds of humans. What is needed are methods and devices which will allow these people to produce, by simple and inexpensive means, the *initial* expression of their information in a form suitable for machine reading. Without these, there will be many situations, especially where the volume of input information is large in comparison to the amount of processing, where computers cannot be proven even if they cost little or nothing.

† Bell Telephone Labs., Inc., Murray Hill, N.J.

We have an example of this input problem in the Bell System, where toll switchboard operators are producing two billion toll tickets per year. These are the records of long-distance calls handled by operators. They are 2½ × 5-inch pieces of paper, each containing 20 to 30 characters of information needed for processing. While there are plans for ultimately eliminating these tickets by improvements in switchboards, they will be with us for a long time. Some idea of the magnitude of this input can be given by stating that these two billion paper tickets produced each year would make a pile 200 miles high or, if laid end to end, a strip 150,000 miles long. What is more significant, it is estimated that it would cost about $32,000,000 per year to transcribe this information to cards by means of keypunching.

A broad look at possible methods by which humans can communicate with machines, including computers, reveals the following situation.

First, the human can communicate by physical actions (generally involving the fingers) on keys, levers, dials, etc. Of these, the telephone dial is undoubtedly the one used in the greatest number. On the other hand, the key is used on a greater variety of machines, including typewriters, teletypewriters, calculating machines, keypunches, and switchboard operator key sets. Second, the human can communicate through physical action which produces a document without the intervention of a machine (disregarding the pencil). This document, in turn, is used to control a machine. Mark-sense cards exemplify this method. Third, it seems probable that it will be possible some day to produce machines which can interpret the human voice reciting numerals and letters. Possibly we may half facetiously suggest that ultimately the human mind can directly control machines.

When one examines the methods just mentioned in comparison with handwriting, one must conclude, however reluctantly, that it is pretty hard to beat handwriting as a ready, economical, fast, and accurate means of expression. Consequently, this discussion deals with two different methods by which handwritten characters may be read. The first falls in the category, mentioned above, of control by physical action and involves a new device which permits real-time communication with machines as characters are written by a stylus. The second falls in the category of communication through documents, and consists of simple methods and devices by which handwritten characters can be automatically read.

Let us consider the problems encountered in automatic recognition of handwritten characters. To simplify the discussion, it will be confined first to numerals. Of course, the problem can be greatly simplified if it is permissible to adopt an entirely new set of characters created specifically for easy machine reading. For example, characters in the set of Fig. 1 could easily be recognized by a machine scanning vertically and horizontally. The patent literature discloses many such sets of symbols. They have the obvious and common disadvantages that writers must learn them and become proficient and accurate in their use, and that they cannot be understood by the uninitiated who occasionally come in contact with them. Personal experience indicates that it would be very difficult to persuade people to adopt them and that promoters of such systems are viewed as enthusiastic but misguided.

Mark-sense marks cannot be considered as special symbols because it is the mark position rather than the shape that carries the information. Mark sensing has the disadvantages of occupying considerably more space than ordinary numerals and of being slow for humans to read.

If the idea of special sets of symbols is rejected, nothing remains but regular Arabic numerals. The problems which are encountered in reading these will now be surveyed.

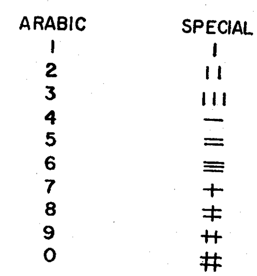


Fig. 1—Special handwritten symbols for machine reading.

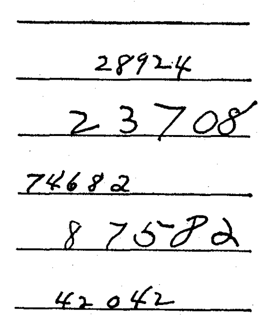


Fig. 2—Numerals written without constraint.

Fig. 2 shows a collection of Arabic numerals chosen from those produced by a random group of people. An examination of these forces the conclusion that some degree of control must be placed on their writing not only to enable machine reading but to reduce "sloppiness" which makes even human reading difficult. The variable factors which must be dealt with by an automatic reading device are location, size, orientation, and shape.

Location is important, if for no other reason, because it generally defines the meaning. For example, a given number appearing in one place on a form may indicate revenue and in another, expense. Also, of course, the machine's problem is greatly eased if it knows exactly where the numeral is located. It may be remarked that while the information content is in the shape rather than size or orientation, nevertheless the machine must be able to recognize shape in the presence of variation in size and orientation.

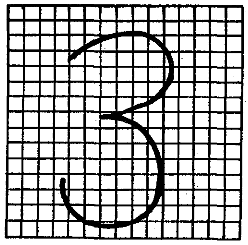


Fig. 3—Cartesian-coordinate grid for character recognition.

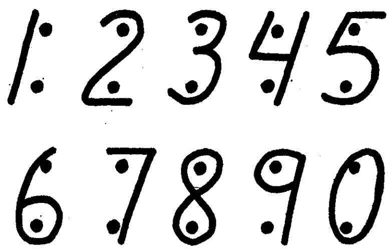


Fig. 4—Numerals with dot constraint.

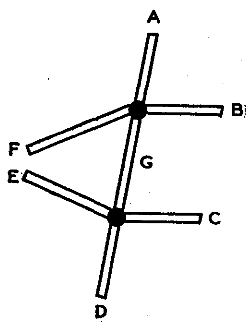


Fig. 5—Set of bipolar coordinates for character recognition.

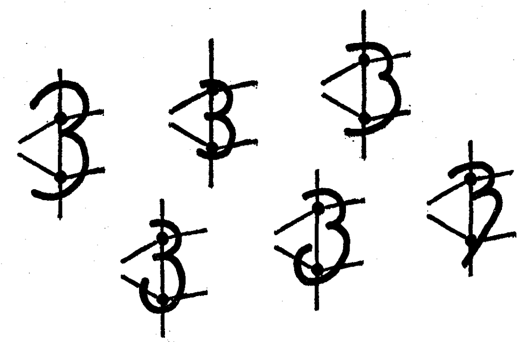
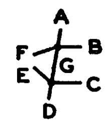


Fig. 6—Range of variation permissible.



ALLOWED CONFIGURATIONS	CRITERIAL AREA						
	A	B	C	D	E	F	G
1: 1:	0	0	0/1	0	0/1	1	0
:1 :1	0	1	0/1	0	0/1	0	0
2	1	1	0	1	0/1	0	1
3 3 3	1	1	1	1	0/1	0	0/1
4 4 4 4	0	0/1	0/1	0	0/1	1	1
5	1	0	0/1	0/1	0	1	0/1
6 6 6 6	0/1	0	0/1	1	1	1	0/1
7 7 7	1	1	0/1	0	0/1	0	0/1
8 8 8	0/1	1	1	1	0/1	1	1
9 9 9 9	1	1	0/1	0	0/1	1	0/1
0 0	0/1	1	1	1	0/1	1	0

Fig. 7—Truth table for numerals.

Any plan to control writing must take into account the methods by which the symbol is to be recognized. The method of recognition, generally proposed, is to examine the character in a field of Cartesian coordinates as in Fig. 3. The presence or absence of a mark in each rectangular cell is indicated by a television-scanning technique to a computer which can operate on the information with all its considerable resources. The cells may or may not be contiguous. Study of this plan indicates that unless very rigid writing controls are employed, this is a hard and expensive way to recognize characters.

At this point we may conclude that these things are needed to read handwritten numerals automatically and at reasonable cost: 1) a means of constraining writing which does not seriously affect writing habits, 2) a mode of machine examination of the symbols, under which the symbols appear invariant with reasonable changes in location, size, orientation, and shape, and 3) compatibility between 1) and 2).

A simple solution is now described which encompasses these three needs. First, the constraint is provided by means of two dots around which the numerals are written as shown in Fig. 4. The naturalness and ease of this method are obvious from the figure. Second, in order for the numerals to appear invariant, the machine examines them in two polar coordinate fields with the two dots as origins. The machine is able to recognize the numeral by sensing which of the radius vectors in the particular set of Fig. 5 are traversed by the lines making up the numerals. (The two left-hand vectors are moved out of their horizontal positions to avoid the ends of 3's and 5's.) The use of the same dots both for the origins of the polar coordinate sets and for controlling the writing makes it possible for simple machines to recognize numerals even though they vary quite widely in location, size, orientation and shape. In Fig. 6, the numeral 3 is shown to be invariant with a wide range of these four variables.

It remains to be shown that the set of radius vectors crossed by each numeral is unique. This is done in Fig. 7 in which a binary 1 indicates a necessary cross, a binary 0, a necessary noncross, and 0/1, indifference to a cross. A point to be noted in the left-hand column of this figure is the considerable tolerance of this method for the vagaries of humans. For example, 1 may be written either to the right or left of the dots since the associated transversals are each unique and can both be interpreted as 1. The nu-

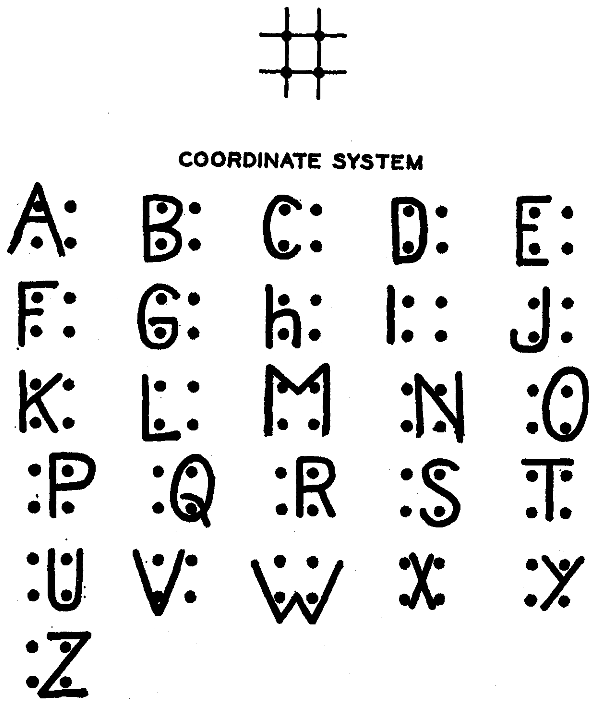


Fig. 8—Four-dot letter restraint—method 1.

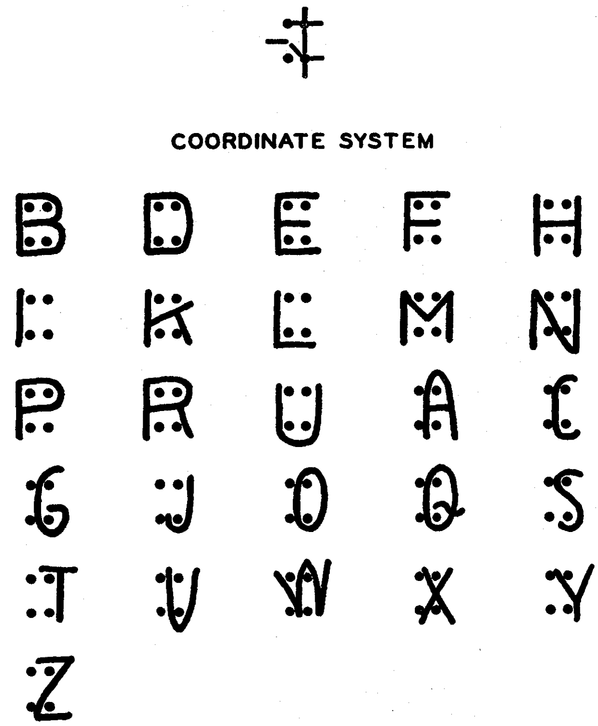


Fig. 9—Four-dot letter constraint—method 2.

meral 7 may be written with the vertical leg either between the dots or to the right. However, a closed top 4 cannot be permitted because it will appear the same as a 9. There are many different truth tables that could be devised following the philosophy indicated in Fig. 7 but trading among allowable variations in the formation of the different numerals. Still others could be designed by adding a "goof" detection feature which detects combinations not corresponding to any numeral. The logic for the latter would be quite extensive, however, because there are 2^7 or 128 possible combinations of which a minimum of 10 are legitimate.

So far only Arabic numeral recognition methods have been disclosed. The question naturally arises as to whether the basic methods of controlling writing by dots and invariance in polar coordinate fields can be extended to include letters.

There are several ways of accomplishing this result. Two will be discussed briefly. The first, involving four dots, is shown in Fig. 8. The basic idea here is that the first half of the alphabet is written about the left two dots, and the rest, with a few exceptions, about the right two dots. It will be noted that with a few exceptions which are necessary to attain uniqueness of each character, all the letters are regular upper-case block or drafting type. H is lower case to avoid confusion with K. G and Q are somewhat specially formed. In passing, it should be noted that block letters seem preferable to script because script writing is more likely to be undecipherable even by humans. Evidence of this is the statement, "please print," on the job applications which some of you have filled out.

Another set of characters is shown in Fig. 9. Here advantage is taken of the fact that 13 of the 26 letters begin with a vertical line, while 13 do not. In this embodiment the letters G, K, Q and W are slightly peculiar.[1]

To mechanize any of the logical methods described above, it is only necessary to devise a machine which can detect marks in the long, narrow areas (hereafter called criterial areas) corresponding in position to the radius vectors. An obvious way of doing this is with electro-optical scanning. Alternatively, a very simple reader can be made by providing a sensing head made with printed wiring as shown in Fig. 10. In this figure, each criterial area is made up of one long, narrow conductor connected to a source of potential, and another, parallel to it, used as a sensing element and connected to a translator. When the head is properly placed on a piece of paper on which a numeral has been written around dots with a conductive lead pencil, the mark on the paper closes circuits between the two parts of each of the criterial areas[2] which it crosses. Hence, certain of the seven leads from the seven criterial areas will be energized causing the translator to energize a different output lead for each different character.

A translator using transistor logic and based on the truth table of Fig. 7 is shown in Fig. 11. The input leads A to G connect to the seven criterial area conductors similarly designated at the top of Figure 7. The RS lead connects to a contact which is closed to reset the translator

[1] Proposed by Dr. L. A. Kamentsky of Bell Telephone Labs., Inc.

[2] Subsequent to the presentation of this paper, U. S. Patent 2,741,312 issued to R. B. Johnson was called to the author's attention. It discloses the use of the two dots and of the radial areas for sensing conduction through the mark constituting the numeral. Relays are operated to control a card punch.

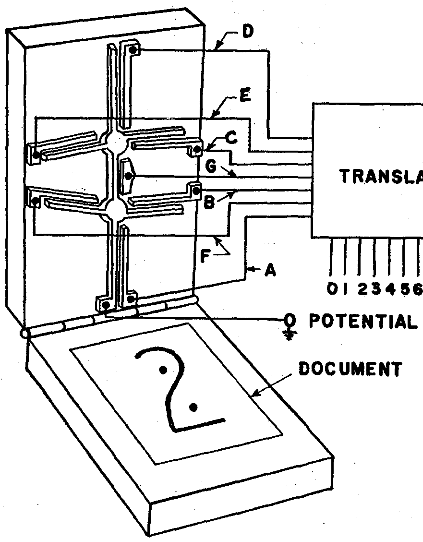


Fig. 10—Reader.

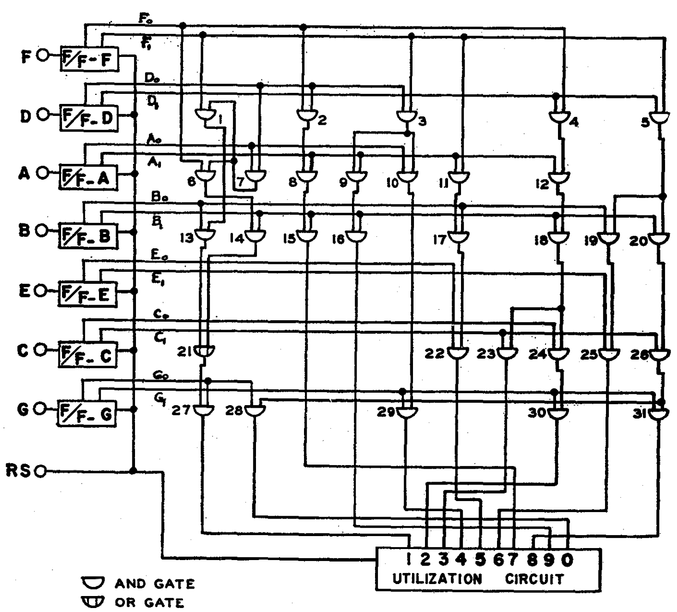


Fig. 11—Translator circuit.

by restoring flip-flops F/F-A to F/F-G to a normal condition in which they energize the leads designated with a zero subscript. The recognition of the numeral 2 will now be followed. Conductors A, B, D, E, and G will have been energized by conduction through the conducting pencil mark although the energization of E is immaterial to correct recognition of the numeral 2. Flip-flops of corresponding designation energize leads of subscript 1. The end result is that leads A_1, B_1, D_1, E_1 and G_1 are energized as well as leads C_0 and F_0. Leads D_1 and F_0 open gate 4. Gate 12 is opened by the output of gate 4 and lead A_1; gate 18 by gate 12 and lead B_1; gate 24 by gate 18 and lead C_0; gate 30 by gate 24 and lead G_1. Gate 30 energizes lead 2 to indicate recognition of that numeral. The flip-flops are not strictly necessary, but they aid by furnishing ample power to drive the logic circuitry.

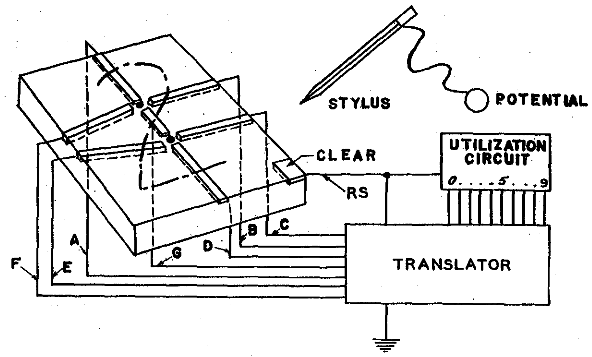


Fig. 12—*Stylator.*

As mentioned before, a method will be described which permits automatic recognition *as the characters are written.* A novel device for performing this function for numerals is shown in Fig. 12. A writing surface is provided on which there are two guide dots surrounded by a set of criterial areas consisting of seven conductors embedded in a plastic plate. As a numeral is written with a stylus connected to a source of potential, the stylus energizes, one at a time, the conductors in the criterial areas involved in the numeral. The combination of areas energized causes certain flip-flops in a translator such as that in Fig. 11 to operate and drive the rest of the translator to indicate the correct numeral. The flip-flops are necessary because the criterial areas are not all energized simultaneously. Alternatively, seven relays may be used to replace the electronic translator of Fig. 11. This device has been tentatively christened a *Stylator,* meaning stylus translator or interpreter.

The problem arises in connection with the *Stylator* of informing it when the writing of a character has been completed. This is necessary because in some cases the character changes from one to another during the writing process and because the translator must be returned to normal before a new character is written. A simple way of incorporating this feature is to provide another conductor in the writing plate which, when touched by the stylus, causes the memory and translator circuits to return to normal as soon as the character already written has been recorded. This conductor may extend around the whole perimeter of the plate so that it can be touched by a continued stroke of the stylus.

In the devices just described, no advantage is taken of the information residing in the sequence in which the criterial areas are crossed. Of course, this information cannot be recovered from a character already written but it is readily available in the case of the *Stylator.* This added information is so meaningful that the two-dot system can be used for letters as well as numerals.

Several uses have been suggested for the *Stylator.* It is a competitor for key sets in many applications. It has been successfully used to control a teletypewriter. It is attractive in this application because it is inexpensive and does not require a long period for learning to use a keyboard.

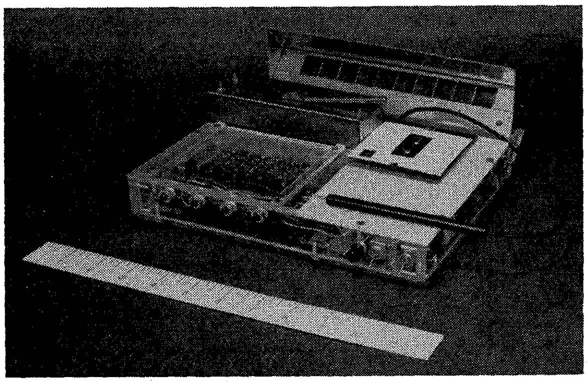


Fig. 13—Number reader and *Stylator.*

If the criterial areas are used to control the frequency of an oscillator, an inexpensive sending device is obtained which may be connected to a telephone set to send information to remote machines.

Fig. 13 shows a combined number reader and *Stylator* which can successively read four separate numerals from a sheet of paper as well as recognize numerals as they are written. The following set of rules will help in using the *Stylator.*

1) Clear by touching the stylus to the small area at the lower right.
2) Make open top 4's.
3) Keep ends of 3's and 5's out of area between segments E and F in Fig. 5.

Conclusion

Methods of constraining the writing of characters for machine reading and machines for reading such characters has been described. A new device called a *Stylator* has been disclosed which permits real-time recognition of characters as they are written on a platen.

Acknowledgment

The author wishes to mention that Dr. L. A. Kamentsky of the Bell Laboratories designed the first logic circuits, constructed the first model and furnished many valuable ideas as well, and that W. W. Gulden of the Cincinnati and Suburban Bell Telephone Company designed and constructed the small model shown in Fig. 13.

Discussion

E. A. Etling (RCA Service Co): What thought, if any, have you given to the development of a system which places no constraint on the handwriting of characters, other than broadly defined statistical bounds?

Mr. Dimond: The polar scanning technique can be extended to systems involving different methods and degrees of constraint. I think that some degree of restraint is desirable because it requires people to form their characters more carefully than they would if they followed their normal habits, which may be so sloppy that the characters are sometimes unrecognizable even by humans.

P. Hersh (General Ceramics): How does the Stylator distinguish between the "early" versions of a character and the final (correct) one?

Mr. Dimond: In these demonstration machines, a separate segment is touched by the tip of the stylus after writing is complete to cause the information stored in the translator circuitry and indicated by lamps to be wiped out. In a commercial machine, the touching of the segment would first cause the information stored in the translator to be transferred to some sort of memory and would then restore the translator to normal. To minimize the effort required, the segment could consist of a border surrounding the platen. It could then be touched with the stylus by continuing the last stroke in writing a character.

M. J. Stoughton (Sears Roebuck): What controls are you contemplating for reducing operator errors?

Mr. Dimond: Errors may result either because the operator writes a wrong number or because she forms it incorrectly. Errors of both sorts may be reduced by training. Nothing can be done in the design of the machine to prevent entirely the former. In this respect, the problem is the same as with a keyboard. In some cases the error can be detected by a system of control totals.

There are some things that can be done to minimize errors due to incorrect formation. More criterial areas and a more able translator would help. If the degree of accuracy required justifies it, mentally computed check numbers can be used. Suppose, for example, the number 13 is to be recorded. The operator also records 24 which is obtained by adding 1 to each digit of the number 13. At some later stage the two numbers are automatically subtracted to check that a difference of 11 is obtained.

L. C. Oesterich (U.S. Navy): You are apparently adapting this reading device to your toll ticket problem. Please outline the system.

Mr. Dimond: The proposed reading device is one of the contenders for solving the toll ticket problem. It would be used in the following manner. The tickets furnished to the operators would have dots preprinted on them around which the operators would write the characters, most of which are numerals. These tickets would be gathered up periodically and sent to a processing center where they would be fed automatically into a device which would read the information and record it on cards or magnetic tape for further processing.

E. Nassell (Electronic Associates): Wouldn't all cases where you consider transmission be much faster handled by a ten-key keyboard of some sort? And as cheaply? As I see it, the best use for reading characters handwritten is when the original recording must be made remote from accessibility to transmitting or recording equipment. Since a form is necessary in any case, doesn't it appear the conventional existing methods of electrographic sensing would remain superior?

Mr. Dimond: If we consider only the originating device, there is probably not much difference in speed or cost between a ten-key keyboard and a Stylator. Two other factors appear to tip the balance in favor of the Stylator. First, preliminary tests indicate that better accuracy is obtained in writing than in keying. Second, a written document can be made at the same time the Stylator is used as a sending device, if the Stylator is designed to use capacitive coupling through the paper rather than direct coupling between the stylus and the criterial areas. If it is necessary to transmit letters as well as numerals, the keyboard would undoubtedly cost more than a Stylator and would require more skill.

I assume that in using the term electrographic sensing, Mr. Nassell refers to what is more commonly known as mark sensing. The main difficulty with mark sensing is that 10 to 20 times more space per character is required than for written characters.

Automatic Registration in High-Speed Character Sensing Equipment

ABRAHAM I. TERSOFF†

Introduction

GREAT strides have been made in the field of data processing machinery. A wide variety of equipments has been developed which will quickly and accurately perform assorted operations on data fed into them. However, far less progress has been made on the task of efficiently providing these equipments with input data. The existence of a severe language barrier has generally necessitated the use of human operators to translate all data into machine language when it is first fed into a data processing system. In an attempt to efficiently overcome this language barrier, a number of different character sensing systems have been developed to serve as automatic man-machine links.

One of the problems faced by most character sensing systems is the relatively inexact fashion in which human beings tend to position information on a document, due both to economic considerations and to the flexibility of the human beings who normally operate on this information. Of great help under such circumstances is the ability of the character sensing equipment to register automatically and accurately on the specific information selected for processing. In fact, it is just such general flexibility which tends to distinguish an equipment suitable for field use from one capable of operating satisfactorily only under laboratory conditions.

Scanning of the Document

The automatic registration system to be described has been successfully employed in a number of different Analyzing Readers.[1] These high-speed character sensing equipments all utilize a high resolution scanner (usually mechanical) and photomultipliers to convert the optical image received from the document into electrical signals. As shown in Fig. 1, the document to be read is moved past a reading station, and an image of the information on the document is focused onto a scanning disk containing perhaps twenty to forty radial slits, each 0.010 inch wide. This disk is normally caused to rotate at a speed of ten to fifteen thousand rpm. Immediately in back of the scanning disk, and swept by light projected through the radial slits, is a fixed plate containing a slit 0.010 inch wide. As the document is moved horizontally past the reading station, an image of the information on it moves across the system of intersecting slits. A two-dimensional scan of the information is thus obtained by means of a beam of light passed by what is effectively a "flying aperture."

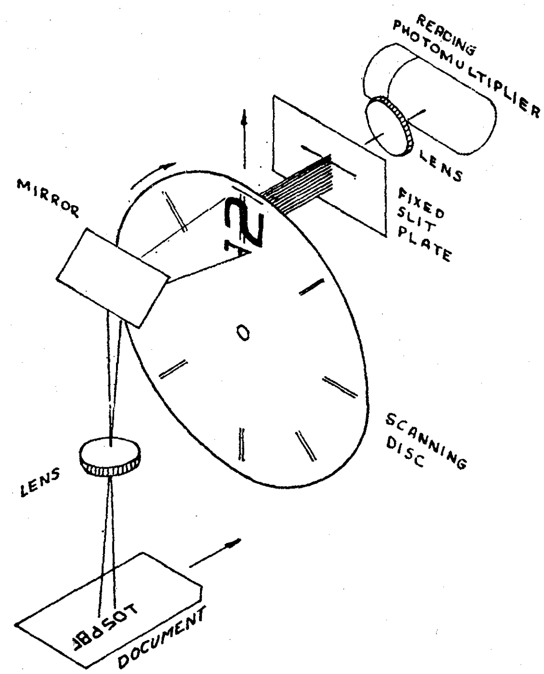


Fig. 1—Simplified view of scanner shows rapid scanning along one vertical axis of moving character image.

Considerations such as the horizontal speed of the document, the height of the field to be scanned on the document and the size of characters to be read will combine to determine the magnification ratio of the optical system employed, the length of the slit in the fixed plate, the number of slits in the scanning disk and the speed of rotation of the disk. Under normal circumstances, more than twenty vertical scans are made across each character, with adjacent scans overlapping slightly. In designing the scanning system, care is taken to insure that the effective cross section of the scanning beam will be substantially less than the narrowest line element normally occurring in the characters to be read.

The moving spot of light passed by the intersecting slits is focused onto a photomultiplier, where it is converted into an electrical signal whose amplitude is always proportional to the intensity of the spot of light. Since the length of the fixed slit is made slightly less than the chord between adjacent radial slits at that point, each scan across the fixed slit will produce a "black pulse" in the photomultiplier output at the point where light is completely blocked from the fixed slit. (See Fig. 2.) During the re-

† Intelligent Machines Res. Corp., Alexandria, Va.
[1] Trademark, Intelligent Machines Res. Corp., registered U.S. Patent Office.

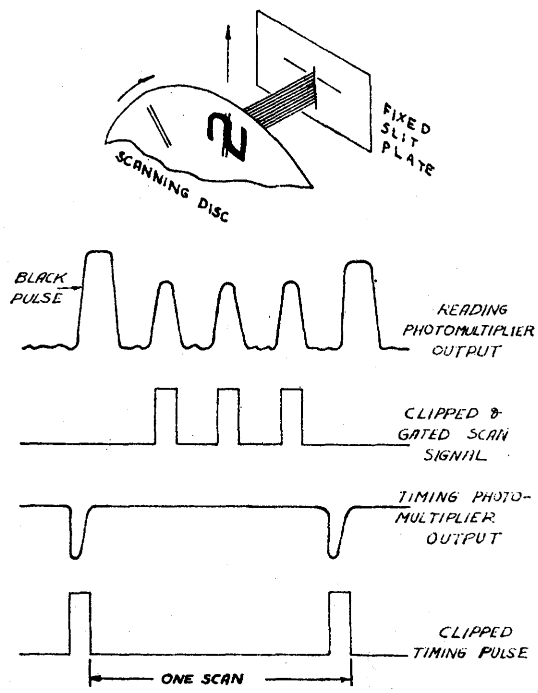


Fig. 2—Development of a clipped scan signal and timing pulse.

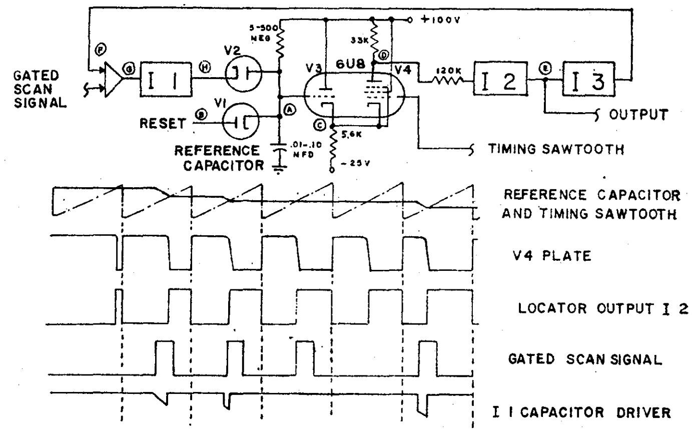


Fig. 3—Locator circuit marks time-in-scan of the earliest signal it receives.

mainder of each scan, pulses will occur only when the scanning of a line element of a character produces a marked diminution in the intensity of the light passed by the intersecting slits. This photomultiplier output is then fed to a video channel, where it is amplified and clipped at the voltage levels (+ 15 and − 25 volts) used in subsequent logical units. Also developed in the video channel is a feedback voltage which holds the amplitude of the "black pulse" constant, compensating for variation in document reflectivity and photomultiplier sensitivity.

A second photomultiplier and an exciter lamp, placed on opposite sides of the scanning disk, are used in a similar fashion to produce a timing pulse at the end of each vertical scan (see Fig. 2.)

Basic Operation of Locator Circuit

The operation of the aforementioned locator circuit, which has been used for automatic registration in machines with the above scanning system, can be analyzed with the aid of Fig. 3. In essence, the voltage established across the reference capacitor is compared with the timing sawtooth. This establishes the point within the scan cycle to be "marked for registration" by the switching of the locator circuit's output. Switching occurs at the earliest point in the scan cycle at which the information being tracked is detected on the document. A feedback loop causes the circuit to keep switching at this point until it is either moved to a point even earlier in the scan cycle by new information or reset to the end of the scan cycle at the end of reading.

Referring to Fig. 3, the signals at points B, E, F, and G are always at either + 15 or − 25 volts. The units I1, I2, and I3 are simple inverters whose outputs are clipped at + 15 and − 25 volts. They are designed to provide an output which is always the exact inverse of their input. The key potential in this circuit is the one established at point A by the reference capacitor. When the locator is not in operation, + 15 volts is applied to reset input point B, causing V1 to conduct until the capacitor is charged to the same potential. With point A at + 15 volts, V3 conducts sufficiently heavily to bring its cathode, point C, to + 15 volts. Under these circumstances, V4 will normally be cut off, conducting only during that brief period at the end of each scan when the timing sawtooth on its grid rises above the tube's cut off potential. Only during this period of conduction will the potential on the plate of V4, point D, be sufficiently low to bring the output of I2 up to + 15 volts. At all other times point E will be held at − 25 volts. Since the signal at point F is the exact inverse of that at point E, the AND gate input to I1 is enabled during almost the entire scan. Only the absence of a gated scan signal keeps point G at − 25 volts and point H at + 15 volts, preventing V2 from conducting.

To put the locator into operation, the potential of point B is reduced to − 25 volts, effectively removing V1 from the circuit. The potential of point A still cannot rise above + 15 due to the clamping action of V2, whose cathode is normally at + 15. However, whenever a gated scan signal is detected during that portion of scan when point E is at − 25 volts, point G will go to + 15 volts and I1 will try to drive point H to − 25 volts. It can seldom do this, since the reference capacitor normally holds the plate of V2 above − 25 volts. Inverter I1, however, will conduct to the limit of its capacity, discharging the reference capacitor through V2 and driving the potential at point A down. Tube V3 will then conduct less heavily, the potential at C will drop to approximately that at A, V4 will conduct for a longer interval at the end of the frame, and input F will disable the AND gate in each frame earlier than before. This process will continue, with I1 driving the potential at A lower and lower, and the output of I2 getting up to + 15 earlier and earlier in the scan, until the output of I3 holds off the AND gate from the earliest point in the scan that a gated signal is detected until the end of the scan. The potential at point A will then remain relatively constant, causing the leading edge of the locator output, point E, to continue marking the earliest point in the scan that a gated scan signal was detected. This situation will continue until the potential at A is either driven still lower by the detection of a gated scan signal even earlier in the scan, or reset to + 15 volts by the signal applied at point B.

It was stated earlier that once set, the potential at point A will remain relatively constant until either driven still lower or reset to + 15 volts. The primary reason this potential does not remain constant is that current drawn by the grid of V3 tends to accumulate on the reference capacitor and slowly drive point A negative. To counteract this effect, a large resistance is connected between this point and a much more positive potential in order to draw electrons away from point A at approximately the same rate that grid current is supplying them. In general, it is safer to choose a resistance value which causes the potential at point A to drift very slowly positive, since it will simply be driven back down as soon as the output of I3 fails to hold off the AND gate while a gated scan signal is present. If, however, the potential at point A is permitted to drift in a negative direction, causing the locator output to mark erroneously a point earlier and earlier in the frame, nothing can drive the potential at A and the locator output back until the unit is completely reset.

It should be noted that the speed with which the potential at point A can be set or reset to the proper values is determined both by the capacitance of the reference capacitor, and by the rate at which electrons can be supplied by I1 and drawn off by the reset pulse.

Application of Locator Circuit

The basic method of operation of the locator circuit has been described above. In practice, the circuit is sometimes made to operate in slightly different fashions for different applications. For example, in one application the scan signals used to set the locator could simply be those used in the basic character analysis program. Other applications may require that the locator be completely positioned and performing its registration function before the information to be analyzed reaches the primary scanning station. In such case, the locator can be positioned by scan signals obtained from this same information as it passes one or more prescanning stations. Using the mechanical scanning system previously described, it is relatively simple to scan simultaneously a number of lines displaced horizontally or vertically from each other on the document. In all applications, however, the basic goal is to register in a known manner on the characters to be read, and to do so with sufficient precision to permit an analysis of the various strokes comprising the characters. An example of the general manner in which characters are normally distinguished from one another is provided by the table in Fig. 4. Here, eleven different stroke criteria are employed

	0	1	2	3	4	5	6	7	8	9
LONG VERTICAL LEFT	+		−	−	−	−	+	−		−
LONG VERTICAL RIGHT	+	+	−	+	+	−	−	+		+
HORIZONTAL TOP	+	−	+	+	−	+	−	+	+	+
HORIZONTAL MIDDLE	−	−	+	+	+	+	+	−	+	+
HORIZONTAL BOTTOM	+	+	+	+	−	+	+	−	+	−
SHORT VERTICAL UPPER LEFT & LOWER RIGHT	+	−	−	−		+	+	−	+	+
SHORT VERTICAL UPPER RIGHT & LOWER LEFT	+	−	+	−	−	−	−	−	+	−
SHORT VERTICAL LEFT & RIGHT SIMULTANEOUSLY	+	−	−	−		−	+	−	+	+
SHORT VERTICAL UPPER LEFT				−	+					
LONG VERTICAL LEFT & RIGHT SIMULTANEOUSLY	+								−	
MIDDLE PROJECTING RIGHT					+					−

+ CONDITION MUST BE DETECTED

− CONDITION MUST NOT BE DETECTED

Fig. 4—Character is identified by correct combination of detected and not detected conditions.

to provide a minimum of two differences between any pair of numbers in a particular type face. This program would not be possible if the lower, middle and upper portions of the characters could not be identified fairly precisely.

The character identification program shown in Fig. 4 is the one actually employed in the Scandex[2] character sensing system, which automatically processes imprinted gasoline credit card invoices. This equipment reads the customer's account number, as imprinted on the reverse side of an invoice card, and punches it into the same card. Since carbon paper is used to make this impression, we are sometimes confronted with smudges adjacent to the characters to be read. In addition, the vertical registration of the account number on the invoice card varies appreciably, being affected by the registration of the credit card during the embossing operation, the position of the credit card relative to the invoice in the imprinter, and the precision of the Scandex card feeding mechanism. To achieve accurate registration on the account number despite the presence of smudges, the locator is programmed to track a point two-thirds below the tops of the characters. This is accomplished, as shown in Fig. 5, by employing two simultaneous scans spaced a character width apart, summing their scan signals, and operating on this sum. Thus, small spurious information will not affect the location system. With the point-in-scan marked by the locator circuit serving as a reference, the characters are then divided into top, middle, and bottom thirds by appropriate time measuring circuits.

Another Analyzing Reader, in which three locator circuits are incorporated, is presently being employed to process automatically utility company billing stubs prepared on tabulating machines. An example of such a stub is shown in Fig. 6. In this application it is necessary that the machine record the first two and last four digits on the top line, the four digits on the second line, and the total on the bottom line of the stub. Here the problem is one of

[2] Trademark, Farrington Manufacturing Co., registered U.S. Patent Office.

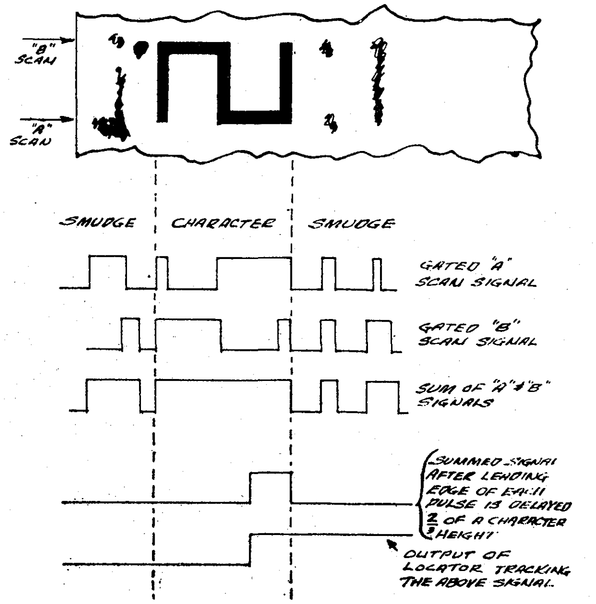


Fig. 5—Scandex locator tacks point two thirds below top of combined scan signals.

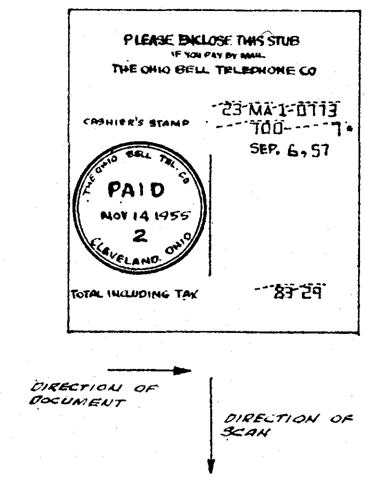


Fig. 6—Locator circuits track characters in first, second, and last lines.

variation in position of the lines to be read, both with respect to each other and to the top and bottom of the stub. This variation in positioning results from the paper form being fed, both in the tabulator and in the cutter, by pins substantially smaller in diameter than the holes in the paper. To correct this variation, the top of each of the lines to be read is tracked by a different locator circuit, as shown in Fig. 6 and, in greater detail, Fig. 7. To prevent that locator which tracks the second line from moving up into the top one, its input is gated so that it can only track scan signals more than a character height away from the first vertical locator.

A third Analyzing Reader employing locator circuits is an automatic page reader developed for an agency of the Department of Defense. This equipment is required to read double-spaced, typewritten information line by line, automatically advancing the page vertically at the end of

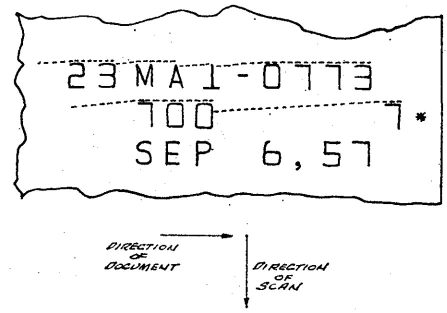


Fig. 7—Locator circuits track adjacent lines on billing stubs.

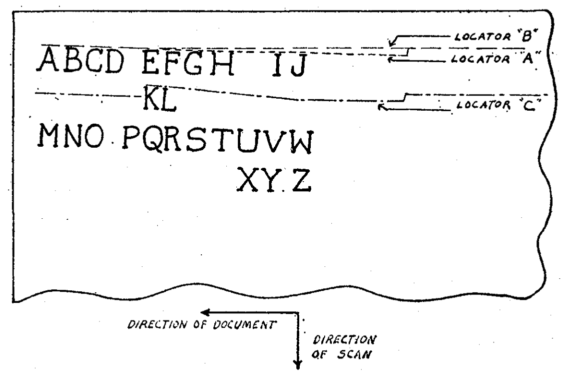


Fig. 8—Locator circuits track normal and substitute characters in spite of tilt.

each line. It is also required to read any "substitute characters" typed between lines in place of the characters immediately above them. (See Fig. 8.) One serious registration problem encountered in this application was the lack of control over distance from the top of the page to the first line on the page. A second problem was the fact that the typed lines are not always parallel to the top of the page. To overcome these problems, a scan capable of covering four single-spaced lines is used, and three locator circuits are employed for tracking purposes. Locator A normally tracks the tops of characters. However, when no characters are detected for a specified length of time, it tracks locator B instead. Locator B, in turn, tracks locator A for a number of characters at the beginning of each line, after which its input AND gate is disabled for the remainder of that line. Locator C will normally track to within a specified distance of locator A, this distance being such as to put it in the middle of the "substitute character" zone. However, whenever it detects a signal earlier in the scan than this point, yet more than a character height later in the scan than locator A, it will track that signal.

In this application, locator B is used to remember the vertical position of the beginning of each line. With this as a reference, the machine can tell where to look for the beginning of the next line even if the lines are tilted. Since it is essential that the point-in-scan marked by locator B not drift significantly during the relatively long time that its input AND gate is disabled, its reference capacitor is made ten times as large as normal.

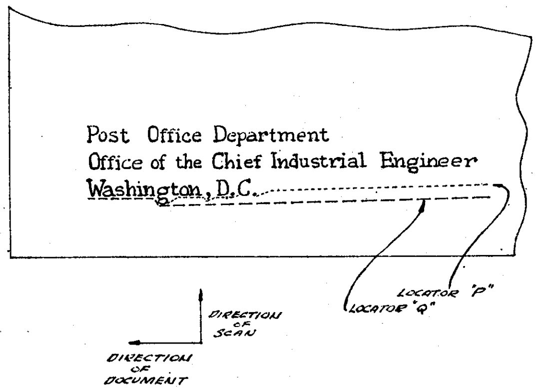


Fig. 9—Locator circuits with slow and fast positive drift of the reference capacitor voltage.

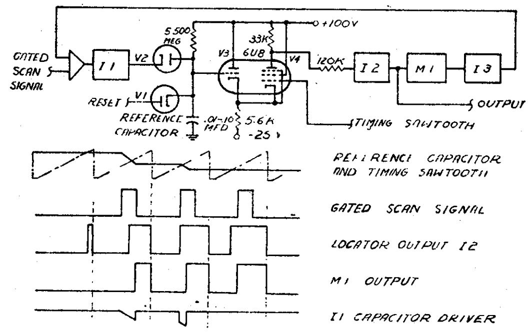


Fig. 10—Locator circuit marks a specified interval before the time-in-scan of the earliest signal it receives.

A fourth Analyzing Reader employing locator circuits is an automatic mail sorter being developed for the Post Office Department. In this machine, programming considerations make it desirable that we be able to register on the bottoms of the various characters in the bottom line, as with locator P in Fig. 9. This is achieved by connecting a relatively low resistance between the reference capacitor and + 100 volts. In this case, 18 megohms is used, causing the reference voltage to drift in a positive direction relatively quickly when not being set down by input scan signals. The danger in this approach is that the locator might occasionally drift up to the previous line, and confuse it with the bottom one. To prevent this happening, a second locator, with a much larger "pull-up" resistance (in this case, 500 megohms) is made to track the same characters. This locator will behave more like locator Q in Fig. 9, drifting upward at a much slower rate. Now, by using locator Q as a reference, an alternate input for locator P can be developed to prevent its drifting to a point more than three fourths of a character height later in the scan than locator Q. As can be seen, this permits locator P to follow the bottoms of those characters in the bottom line very closely without any danger of its drifting into the previous line.

An interesting problem encountered in this same mail sorter application was the need for determining immediately the fact that we have begun reading a new line in a staggered address. A normal locator circuit tracking the bottom of the bottom line on a document will, in such a case, simply drop down to this new line. How, then, do we determine that this has occurred? The technique presently employed is to have an additional locator track a point roughly three fourths of a character height below the line being read. The presence of a scan signal earlier in the scan than this point indicates that we have begun scanning a new line in a staggered address.

A locator circuit can be made to track a specified distance ahead of the input scan signal by simply delaying the leading edge of its output pulse by a fixed amount before inverting it and feeding it back to the AND gate. Unit M1 in Fig. 10 is used to accomplish this delay. As can be seen from Fig. 10, the leading edge of the M1 output pulse, and hence the trailing edge of the I3 output pulse, follow the leading edge of the locator output pulse by a fixed interval. The presence of M1 does not, however, alter the basic operation of the locator circuit itself. Whenever the AND gate's output is positive, V2 will conduct and cause the locator to mark a point earlier and earlier in the scan. It is only I3's disabling of the AND gate that prevents scan signals from improperly advancing the locator output all the way to the beginning of the scan. Scan signals can advance the locator only if they occur while the output of I3 is positive. We see, then, that scan pulses will continue to advance the locator until the trailing edge of I3's output is brought to the same point in the scan as the leading edge of the earliest scan pulse. Of necessity, the leading edge of the locator output will then be marking a point-in-scan which is the specified distance ahead of the earliest scan signal detected.

Conclusions

In most high-speed character sensing systems, automatic registration is extremely helpful in: 1) overcoming poor registration and extraneous marks on the document, 2) compensating for tilt of the information to be read, and 3) accurately dividing the characters to be read into vertical zones for purposes of stroke analysis. A special locator circuit is employed in a number of IMR Analyzing Readers to provide such automatic registration. Minor modifications of the circuit permit the control of such characteristics as the speed with which it can be set, the speed with which it can be reset, the timing stability of its switching point, once set, and the degree to which it can register on misaligned characters. Registration on a variety of types of information, either in restricted zones or anywhere on the document, is possible. The usefulness of this locator circuit has been thoroughly established through its application to various Analyzing Readers presently under development or in use in the field.

Acknowledgment

The basic locator circuit was developed by David H. Shepard of Intelligent Machines Research Corporation. Techniques for applying it to different models of Analyzing Readers reflect the contributions of numerous members of the IMR staff.

The National Cash Register High-Speed Magnetic Printer

J. SEEHOF†, M. ARMSTRONG†, G. FARLEY†, M. LEINBERGER†, M. MARKAKIS†, AND S. SMITHBERG†

I. Introduction

THE magnetic printer is designed to operate in conjunction with an electronic data processing system. It operates on the principle of recording a latent magnetic image on a special paper, which is essentially magnetic tape with a white topcoat. The image is in the form of an alphanumeric (or other) character which is subsequently made visible by exposure to a ferromagnetic powder attracted to the magnetized portions of the paper. The powder is coated with a thermoplastic resin and requires a heating operation to fix it to the paper. An example of the print is illustrated in Fig. 1.

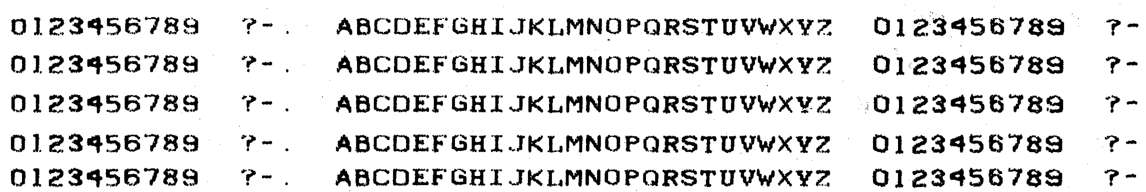


Fig. 1—Sample of print.

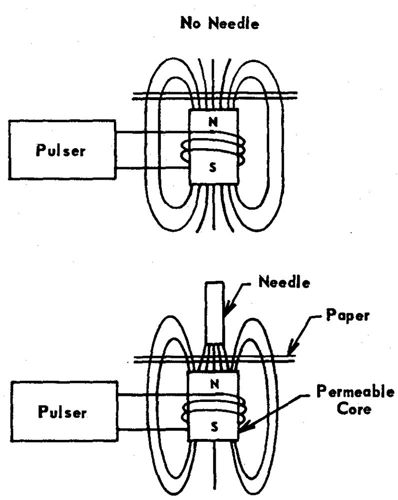


Fig. 2—Effect of permeable needle on field strength.

II. Principles of Operation

Printing is accomplished by utilizing the fact that a weak diffuse field will not magnetize the paper while a strong concentrated one will. Field concentration is obtained, as illustrated in Fig. 2, by placing a permeable needle directly above the area which is to be printed. Print may then be made visible by exposing the area to a black permeable powder.

Dynamic operation of the printer is illustrated in Fig. 3. The magnetic field is established by a coil wound around a permeable bar. It is energized with pulses of a length directly proportional to the length of vertical scan to be recorded. Illustrated is a three-by (three vertical needle sweeps per letter) format where the letter T is formed by pulsing the bar when needle 1 passes over the top area, while needle 2 is traversing the entire bar, and when needle 3 passes over the top area. The needles are arranged in the form of a helix on a drum rotating over the bar and are spaced so that only one needle is within the field of the bar at any one time.

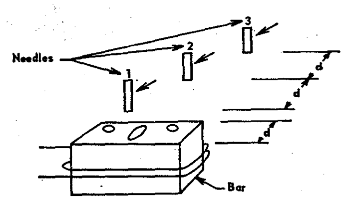


Fig. 3—Three-by system.

A completed breadboard of the magnetic printer (Figs. 4-6) consists of a spiral track of needles arranged around the surface of a drum. Directly over the drum supported by a fixture, is a permeable bar. The magnetic field which places the latent image on the paper is established between the needle points and the bar. During the time that a particular needle point is passing under the bar, a magnetic flux path is established between the needle and bar only when the coil is energized. Seven vertical scans are allotted for each character (seven-by system). The scan (bar) is energized for the full height of a given character (width of bar) if a full vertical line is to be recorded for that portion of the character, or the scan is energized for only a portion of the time as needed.

As an example, consider the formation of the letter L.

† Electronics Div., National Cash Register Co., Hawthorne, Calif.

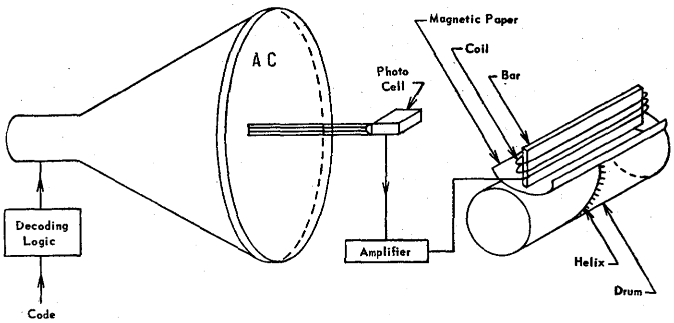


Fig. 4—Block diagram of magnetic printer system.

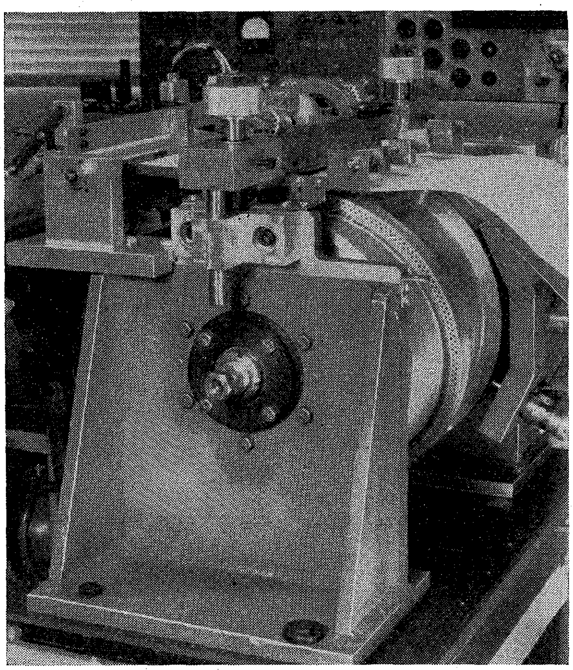


Fig. 5—Close-up view of the drum, bar, and paper.

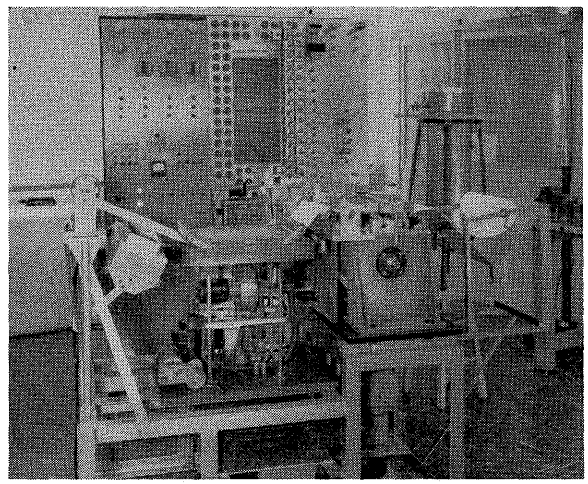


Fig. 6—Breadboard model of magnetic printer.

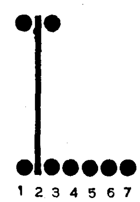


Fig. 7—The letter L.

The second scan is a full height scan, whereas the other six are energized for only portions of each pass over the bar. The resulting image appears in Fig. 7.

The encoder which has been utilized in conjunction with the printer is a cathode-ray-tube device. Over the face of the CRT is placed a photonegative mask which has on it all of the characters which are to be used in the printer (60 or more). The mask is easily replaceable with other sets of characters or symbols. In back of the mask is a photocell which converts the light energy from the phosphor on the face of the CRT to electrical pulses. When a character is to be printed,[1] the binary coded representation of the character enters the electronic input circuitry of the printer. A decoding matrix then directs the horizontal and vertical deflection plates of the CRT to a certain portion of the face of the tube, this point being at one corner of the character (on the mask) which is to be scanned. The CRT beam then scans the character in seven sequential vertical sweeps in synchronism with the passage of the seven needles past the bar in the printer. The photocell senses the beam when it illuminates the transparent portions of the mask. In like manner, when the light energy is interrupted by the opaque portions of the mask, the photocell lies dormant. The electrical signals from the photocell are amplified and converted into current pulses for the coil and magnetic field pulses from the bar. This CRT encoder has been utilized for some time and has proven very reliable. Various other types of encoders, such as a magnetic core matrix are feasible and would probably be less expensive.

As the drum rotates over the bar, a full line of characters can be printed. Since the width of the helix depends upon the diameter of the drum for a given letter size, one can design either a large drum to scan 120 characters per revolution or a smaller drum to scan 40 characters per revolution with three spirals (each with an independently pulsed bar) wound on its surface. To achieve equal speeds, the smaller drum would have to rotate three times as fast. The present system scans 40 characters per revolution and 80 characters per line at a speed of 1200 characters per second or 15 lines per second. Higher speeds than this have not been attained with the present breadboard due to mechanical vibrations set up upon rotating the drum at high speeds. There is no reason to assume that increases in speed by at least a factor of ten cannot be attained with better drum design.

The paper that is used is formed by a series of layers:

1) Base kraft paper.

[1] Along the edge of the drum is placed a magnetic layer. On this layer are placed pulses in line with the first needle for a revolution, each needle beginning a character, and each needle on the drum. Three separate reading heads are utilized. Thus the printer can "signal" the encoder or paper feed when a line begins, when a character begins, or when the needle begins scanning the bar.

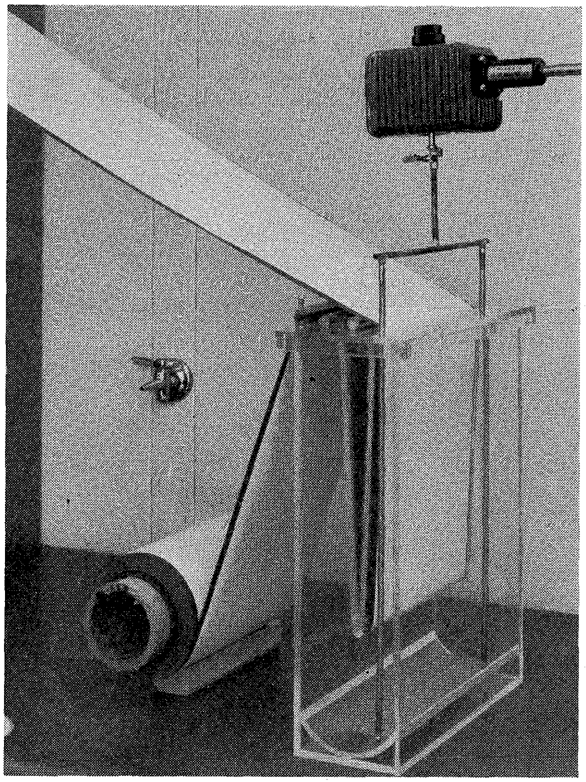


Fig. 8—Experimental working model of a liquid inking chamber.

2) Magnetic oxide layer (can be permanently magnetized).
3) White topcoat.
   a) Forms a contrasting surface for the black ink.
   b) Forms a multilith surface.

The printer produces only one copy at the time of printing. However, since the paper contains a multilith topcoat, it can be utilized to make any number of copies at a later time by a simple multilith operation.

Inking is accomplished by passing the paper through a liquid (Freon-113) suspension of iron particles (mapico black coated with resin) which is in a state of rapid agitation. This operation can be accomplished with the apparatus illustrated in Fig. 8 at speeds greater than 15 lines per second. For off-line operation a continuous feedthrough would be utilized. For on-line operation, since it is not desirable that the paper stop for any length of time in the inking solution (overinking would occur), a take-up reel would be utilized so that the paper would be passed through the inker in long passes (10 feet or more). As soon as a pass through the inker was completed the paper would stop, the agitation would stop, and therefore further inking would cease. When another ten feet of paper built up, the printer would signal the agitator to begin again and another ten feet of paper would be inked.

III. Design Parameters and Experimental Results

A. Needles

1. Needle Permeability: The flux density, B_n, concentrated in the needle is a measure of the print quality that can be obtained since it is directly proportional to the field magnetizing the paper. A certain minimum B_n is necessary as a print threshold, and as high a value as possible should be obtained consistent with the other variables. Since $B_n \approx \mu_n$ (needle permeability) H_B (field due to bar), it can be seen that keeping all other parameters constant, the higher μ_n is, the higher B_n will be. The effective permeability μ', is not only a function of the material utilized but also of the length to diameter ratio of the needle. The graph in Fig. 9 illustrates how important this ratio is.[2]

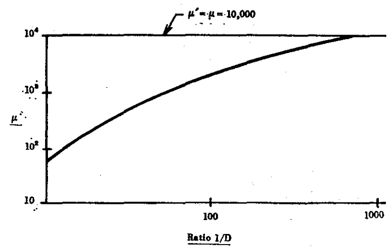


Fig. 9—Effect of length to diameter ratio of needle on effective permeability.

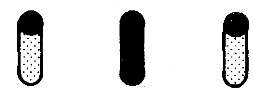


Fig. 10—Needle retentivity and trailing.

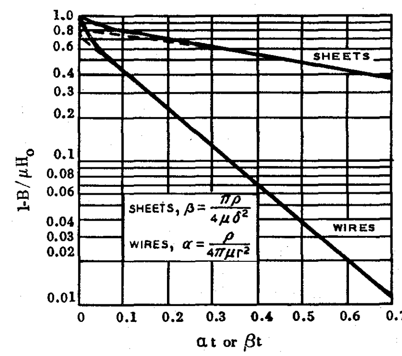


Fig. 11—Build-up of flux in wires and sheets subjected to sudden constant field.

If the needle utilized has "hard" magnetic characteristics (permanent magnet properties), and retains part of its field after being pulsed, it will continue to print in a weaker fashion if this field is above the threshold value of B_n. To eliminate this "trailing" effect, "soft" magnetic materials should be used. For example, if trailing occurs, the letter T would look as shown in Fig. 10.

2. Needle Response Time: In te megacycle region, the needle response time will be primarily dependent upon eddy current effects. These effects are in turn functions of permeability, resistivity, and geometry. The graph[3] in Fig. 11 illustrates the effects of these variables for wires (needles).

[2] R. M. Bozorth, "Ferromagnetism," D. Van Nostrand Co., Inc., New York, N.Y., p. 848; 1951.
[3] *Ibid.*, p. 784.

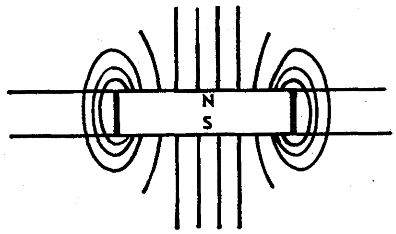


Fig. 12—Field distribution for magnetized paper area.

The larger the resistivity, and the smaller the diameter and permeability, the quicker will be the rise time of the needle. The same general argument will apply to the decay of the fields. An example follows:

$$H = 0, \qquad t = 0$$

$$H = H_n, \qquad t \geqq 0$$

Needle—mu-metal ($\mu \approx 4 \times 10^4$).

Let $1 - \dfrac{B_n}{\mu H_B} = 0.1$ (*i.e.*, 90 per cent of rise is attained)

$$\alpha t = 0.35.$$

Case I:

$$r = 0.005 \text{ inch}$$

$$t = 0.44\ \mu\text{sec.}$$

Case II:

$$r = 0.060 \text{ inch}$$

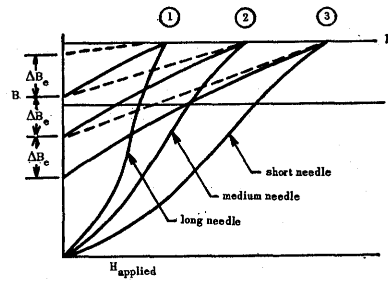
$$t = 66\ \mu\text{sec.}$$

If the needle response time is too slow, a transient "trailing" effect analogous to permanent retentivity will be present.

3. Needle Shape: Fig. 12 is a lateral view of a piece of magnetic paper containing a pulsed portion. The area which will most attract the permeable powder is that portion exhibiting maximum field strength. It is apparent that this will occur not at the center of a pulse, but at the edges, where a short return path for the magnetic flux is available. Print density from a large needle will suffer by having central unprinted areas, while a very small needle will have lesser area coverage on the paper. The effect of powder diameter should be appreciable as the particle size approaches printed area dimensions.

4. Effect of Needle Length: Curves 1, 2, and 3 of Fig. 13 illustrate the general behavior to be expected of the hysteresis curves as needle length is shortened while diameter is kept constant. The heavy lines depict the static magnetization-demagnetization curve while the dotted line indicates the mode of demagnetization in a dynamic (high frequency) case. After a short period of time, B will drop to the normal static case.

a) The needle of Curve 1 will exhibit strong "transient retentivity and also permanent retentivity which



B_t = Threshold flux density at which print just becomes visible on paper.
B_p = Flux density needed for dense printing.
ΔB_e = Transient flux density remaining after each pulse due to eddy currents.

Fig. 13—Effect of nedle length on hysterisis characteristics.

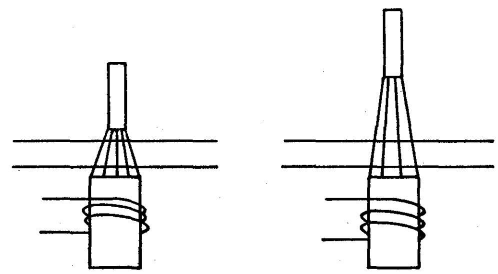


Fig. 14—Paper field definition and concentration as a function of paper-needle distance.

will cause light print to appear on the paper even if the bar is not being pulsed.

b) Needle 2 will exhibit weak "transient" trailing and no permanent retentivity.

c) Needle 3 should show no "transient" trailing or permanent retentivity.

These effects were all demonstrated by experiment, and strong printing with no trailing was obtained with soft iron, piano wire, and mu-metal of 0.10 and 0.05 inch in length.

5. Needle Distance from Paper: As the needle is moved away from the paper surface, a marked diminishing of field strength and definition is apparent (see Fig. 14). Thus, optimum printing conditions will be present when the needle is as close as possible to the paper surface.

6. Overlap of Needle Fields: In a three or multiple-by system depicted in Fig. 3, it is possible for the fields of successive needles to cancel each other if they overlap. Fig. 15 shows what proper and improper needle spacing can do in a two-by system.

7. Needle Pulsing: Six mu-metal needles were wound with twenty-five turns of wire each and connected in series through a commutator so that the pulses received were the same that were ordinarily fed to the bar (see Fig. 16). Experiments were conducted to determine the effect of:

a) Needle pulsing alone.

b) Needle and bar pulsing with same and opposite polarities.

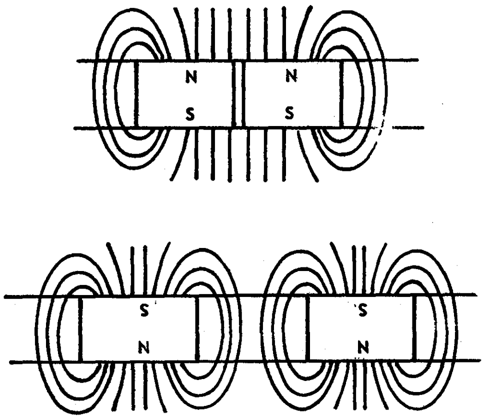


Fig. 15—Paper field patterns as affected by needle overlap. (a) Improper. (b) Proper.

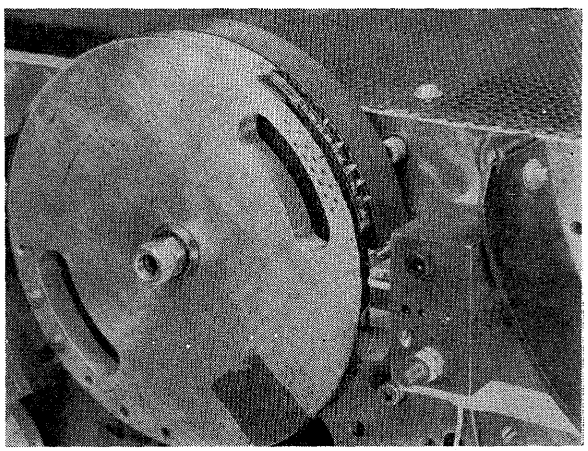


Fig. 16—Close-up of needle pulsing equipment.

c) Needle pulsing in the presence of a nonmagnetic (lucite) bar.

All results were positive but it was felt that the slight improvement in print quality did not warrant the extra trouble inherent in mechanizing the commutator system.

The purpose of this experiment was to determine whether the directed field emanating from the needle point itself, instead of the bar, would improve definition of print. Very little improvement resulted.

8. Needle Materials Tested: The following materials (available in needle form) were tested to determine optimum results with respect to permeability, retentivity, and response time:

a) Welding rod (commercial grade)
b) Piano wire
c) Phonograph needles
d) Ferrite
e) Mu-metal
f) Laminated mu-metal
g) Pure iron wire (reagent grade)
h) Pressed powered carbonyl iron (GQ-4).

The best over-all characteristics were obtained with mu-metal which exhibits high permeability (response at low

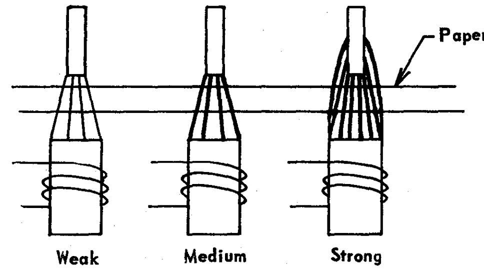


Fig. 17—Effect of bar field on paper field concentration and definition.

field strengths), low permanent retentivity (no permanent trailing), and fair eddy current lowering (slight "transient" trailing found at high field strengths). Transient trailing (eddy current effects) is apparent with all needles at high field strengths, but decreasing the length to diameter ratio of the needle almost completely eliminates this effect.

As a result of these experiments and analysis of design parameters, the needles utilized in the printer were fabricated of short (50-mil length, 7-mil diameter) pieces of mu-metal wire. The wire was then mounted in nonmagnetic stainless steel cylinders (60-mil outside diameters with a 7-mil hole drilled down the center) to give rigidity and to keep the wires away from the surface of the aluminum drum. The cylinders had no noticeable effect upon the print quality.

B. Bar

1. Field Strength of Bar: Weak bar fields will cause no print at all until the threshold of B_n is exceeded. Very strong fields will cause printing over a wide area with loss of definition, even though field strength is increased. An optimum field strength exists (see Fig. 17).

The effect of varying the current through the bar from 0.5 amp to 3.0 amp is illustrated by Fig. 18, next page.

2. Effect of Stray Field from the Bar: Because of stray field from the bar, weak print may appear both above and below the desired line unless special precautions are taken. This effect was substantially decreased by winding the bar up to its very tip, thus effectively directing the field in the desired direction. Fig. 19 illustrates the two cases.

3. Bar Distance from Paper: Since $B_n \approx \mu H_B$, and since H_B does not vary much with small distances from the bar, one would expect approximately the same definition of field in the two cases. Gradual diminishing of field strength as the bar is moved from the paper should also occur (see Fig. 20).

4. Electronic Response Time and Bar Permeability: The rise time of the electronic circuitry pulsing the bar should be kept to a minimum. Besides being a function of various circuit parameters, the electronic rise time is markedly affected by the L/R ratio of the pulsing coil. The number

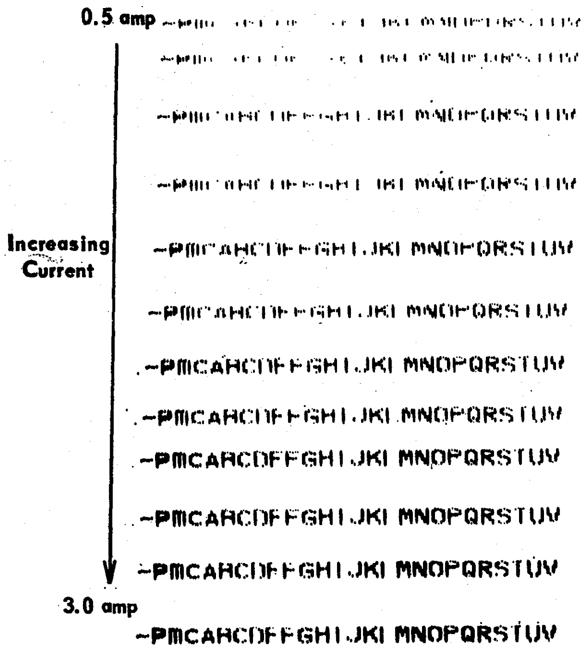


Fig. 18—Effect of bar current.

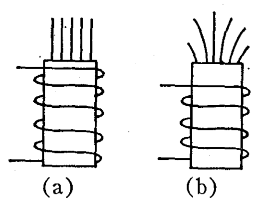


Fig. 19—Bar field. (a) Properly wound. (b) Improperly wound.

of turns about the bar should be made an optimum with respect to print quality, since not only rise time but H_B is a direct function of the inductance of the coil. To obtain a given H_B, (or a given number of ampere-turns) the current should be maximized consistent with other variables and the number of turns held to a minimum. Bar permeability should be as large as possible to minimize number of ampere turns necessary.

5. Bar Response Time: The magnetic response time of the bar lags behind the pure electronic response, and, if too slow, may cause transient "trailing" of the same type as in Section III-A, 2 (needle eddy-currents).

6. Bar Retentivity: As long as the retentivity $(H_B)_{\text{ret}}$ is below the threshold value for printing, this may aid somewhat in acting as a "bias." Instead of pulsing the bar from $H_B = 0$ to $H_B > (H_B)_{\text{threshold}}$, one may decrease the eddy currents by pulsing from $(H_B)_{\text{ret}}$ to $H_B > (H_B)_{\text{threshold}}$. The same type of analysis would apply to needle retentivity.

7. Bar Material: Bars constructed of the following materials were tested:

a) Ferrite
b) Mu-metal
c) Laminated mu-metal

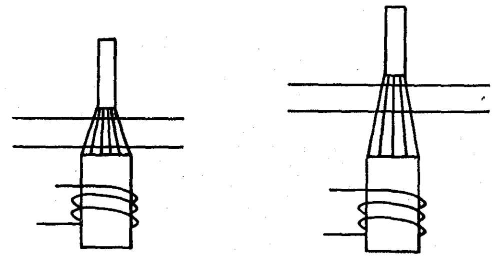


Fig. 20—Paper field concentration as a function of paper-bar distance.

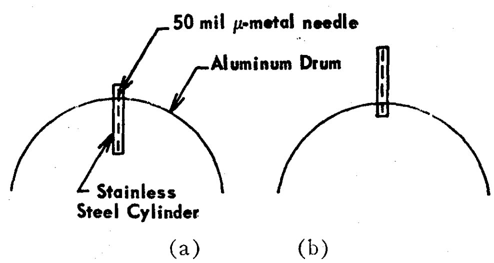


Fig. 21—Effect of drum mass. (a) Poor print. (b) Good print.

d) Pressed carbonyl iron (all types)
e) Cold rolled steel.

Good print was attained with all bars tested, but ferrite gave consistently better results. At present speeds, ferrite gives acceptable results (as do the others). At higher speeds, on the order of 10,000 characters per second, further experimentation may be necessary to provide a high enough response time.

The final bar utilized in the printer was 0.1 inch wide, 0.875 inch high, and 8 inches long. 100 turns of wire were wound in four layers of 25 turns each, concentrated on the upper ⅜ inch of the bar (to keep the field well defined). Continuous runs were not made, but it is anticipated that heat effects with concomitant insulation breakdown of the wire will occur under these conditions if precautions are not taken. Magnet wire with high temperature characteristics will be necessary.

C. Drum

The large mass and high electrical conductivity of the aluminum drum used in the prototype magnetic printer are the cause of *large* eddy current effects, so that it has been found necessary to mount the mu-metal needles as far from the drum surface as possible with present equipment. No print at all was obtained when the needles were at the surface of the drum, while marked improvement was obtained by moving the needles away from the drum surface as indicated in Fig. 21.

Other approaches which would eliminate this effect are:

1) Fabrication of a nonconductive drum.
2) Coating the aluminum drum with a nonconductive layer at least two inches thick.

D. Aluminum Paper Guides

Fig. 22 illustrates the geometrical configuration of paper guides, bar, needles, and paper; Fig. 23, the print obtained from this system.

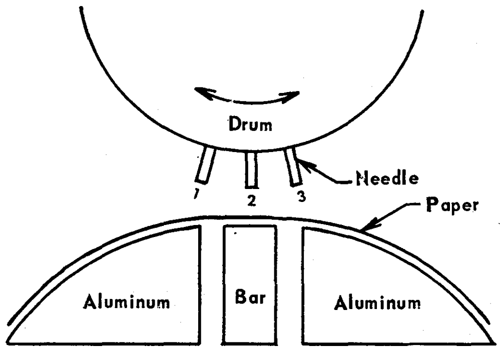


Fig. 22—Effect of paper guides.

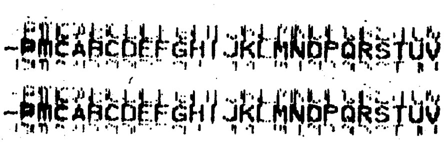


Fig. 23—Print with paper guides.

While needle 2 of Fig. 22 is pulsing, needles 1 and 3 are so close to the paper surface that the stray field from the bar causes weak print to appear both above and below the desired line. Letters below are displaced one needle to the left and those above, one needle to the right.

For example, while needle 2 is tracing out the center scan of the letter T, needle 1, which is displaced to the left and which is beneath the line, is tracing the same scan. Needle 3 is doing the same thing above the line and displaced one to the right.

A slight amount of retentivity is also apparent. This is attributed to the presence of the high conductivity aluminum guides with consequent eddy currents. These two effects were eliminated by utilizing the following system illustrated in Figs. 24 and 25.

By causing the paper to be bent sharply over the end of the bar, it is possible to eliminate the effect of the adjacent needles, since print threshold is very sensitive to distance of the needles from the paper. The lines seen above and below the print are due to an "edge effect" of the bar. Extra strong lines of force emanate from this region and the effect was easily eliminated by placing a 5-mil mylar shim over the bar surface. This sufficiently lowered the field of the edge to a value below the print threshold but did not substantially lower the over-all field strength.

E. Paper

1. Paper Response Time: Since the magnetic layer in the paper is composed of finely powdered ferrite particles, with consequent minor eddy-current problems, it is probable that the paper response time is appreciably faster than that of the bar and needles.

2. Paper Magnetic Fields: In order to most strongly attract the printing ink (fine magnetic powders), the magnetic particles in the paper should have characteristics of maximum retentivity.

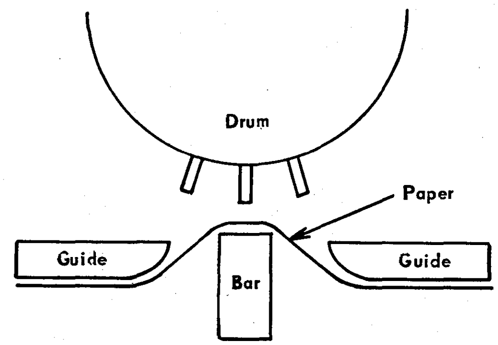


Fig. 24—New guide system.

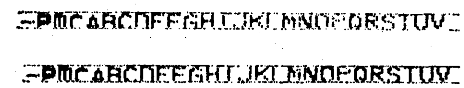


Fig. 25—Print with new guide system.

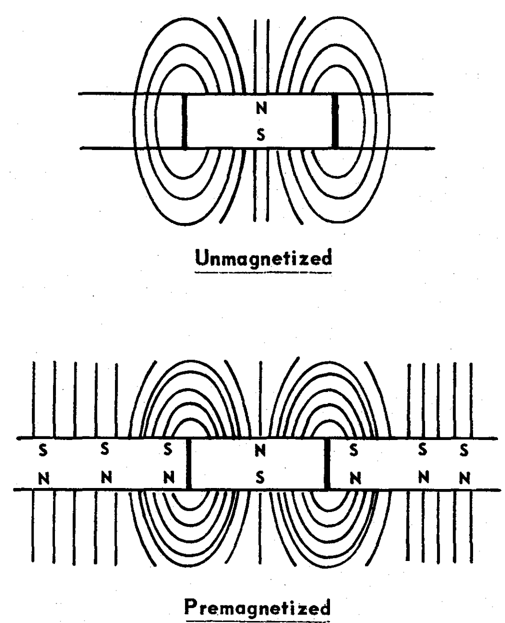


Fig. 26—Paper field patterns as affected by premagnetization.

3. Paper Orientation and Magnetization: Since the ferrite particles in the magnetic layer are asymmetric, they should be oriented during manufacture so that the axis of maximum permeability lies in the direction of the field which is to be applied (*i.e.*, perpendicular to the plane of the paper).

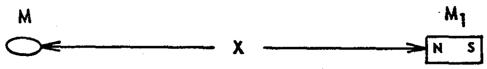
Premagnetizing the paper in a direction opposite to that of the applied field should enhance printing density by shortening the flux return path (see Fig. 26).

F. Inking

1. Liquid Inking: Investigations determined that inking in a liquid medium did not have to be conducted under static conditions. Agitation by a Fisher Vibrastirrer yields greatly improved results. Movement of paper through the liquid medium does not cause smearing or other deleterious effects on print quality. An experimental inking chamber is illustrated in Fig. 8.

2. Gas Phase (Dust) Inking: All experimentation in this area yielded poor results with a large amount of background inking. There is also a serious danger of explosion when utilizing small iron particles in a system where static charges can be built up.

3. Inks and Field Attraction: If an unmagnetized particle is brought into the field of a permanent magnet, the force of attraction is as follows.[4] (See Fig. 27.)



$$F(\text{attraction}) = \frac{6MM_1}{X^4}$$ where M = induced magnetic, M_1 = permanent magnet

Fig. 27—Attraction of permeable particle by a magnetic field.

The larger the permeability of the particle, the larger are M and the attractive force. At the same time it is desired to minimize the X^4 term by using fine particles to allow a minimum distance of approach. Thus, since fine particles have lowered permeabilities,[5] an optimum ink particle size must be experimentally determined in order to maximize (F/mass) of the particle.

Particle shapes and interactions with surrounding fluids will influence the rate and density of deposition upon a magnetized paper surface. Maximum covering power per particle is also desirable for best printing contrast.

4. Solvents: Freon-113 is an excellent dispersing agent because it has the following properties:

a) Volatile
b) Nontoxic
c) Noninflammable
d) Nonviscous.

[4] *Ibid.*, p. 729.
[5] *Ibid.*, p. 45.

IV. Present Characteristics and Future Capabilities

The present characteristics and future capabilities are listed below.

1) Speed—1-2000 characters per second at present: 20-50,000 characters per second ultimately.
2) Type face—External and replaceable fonts of 64 or more characters each; unlimited number and kind of characters are available with the restriction that they must fall within the space usually reserved for capital letters.
3) Legibility—Comparable to typewriter print.
4) Noiseless—Except for paper feed.
5) Low maintenance—No mechanical moving parts except rotating drum. No breakdown due to solenoids and moving hammers, no wear of type face, no ribbon debris.
6) Serial operation—No line buffer necessarily needed. Continuous (nonintermittent) off-line paper feed possible.
7) Copies—Unlimited number available from a multilith master which is printed directly. Only one copy immediately available unless printing is conducted in tandem.
8) Immediate visibility—Not available: lag of ten feet of paper if used for on-line operation with take-up reel; about two feet for off-line operation with steady travel through inker.
9) Character size—Present machine has ten characters per inch, each 0.1 inch high, width of printing line 80 characters. All of these parameters can be easily varied.

On-Line Sales Recording System

J. S. BAER†, A. S. RETTIG†, AND I. COHEN†

Introduction

THIS PAPER presents an equipment description of a pilot on-line Sales Recording System currently in operation for the Associated Merchandising Corporation as part of a research project. This pilot system comprises point-of-sale units connected to a central computer by means of an Input-Output Buffer Unit. The operating characteristics of this system are such that they may be extended to include, for example, inventory and production control, and the handling of transportation reservations.

Operation of the pilot unit which started in April, 1957, has continued with highly satisfactory results. The Sales Recording System has maintained an average "up-time" record of over 97 per cent for the past nine months.

Sales Recorder

The point-of-sale unit or Sales Recorder (Fig. 1) consists of a keyboard for manual input, in combination with a character display, and a procedure indicator. There is a punched tag reader for automatic input of the sales person number, customer number, and merchandise stock number. An output printer provides for a three-part sales check at the point of sale. The entire unit is packaged over a cash drawer and is contained within a vented aluminum housing, which opens completely for servicing.

Fig. 1—Sales Recorder.

The keyboard (Fig. 2), is of the type commonly referred to as a ten-key keyboard. Actually, it consists of fourteen keys—ten numeric and four control keys. The four control keys are: Enter, Nonmerchandise, Clear, and Total; they allow the operator to control the procedural input to the machine.

The keyboard is designed to provide both manual and electrical interlocks. When one key is depressed, another key cannot be depressed, and two keys cannot be depressed at the same time. The actual depression of the key is only for the first position of the key stroke. From that point on, the key is mechanically pulled down, and is held down until the Input-Output Buffer Unit has recognized the character being input as a legitimate one. When this routing is complete, the key is allowed to return to its normal position. Each time a numeric key is depressed, that number is shown in a lighted character display window of the point-of-sale unit. In this manner, a sales person can verify any item prior to striking the Enter key, which clears the display.

Fig. 2—Sales Recorder controls and indicators.

† Electronic Data Processing Div., Industrial Electronic Products, RCA, Camden, N.J.

The output of the keyboard is four bits with parity. These five bits, plus an additional control bit, are the information levels sent to the Input-Output Buffer. Operation of the keyboard can be at the sustained rate of six characters per second.

The procedure indicator provides a visual indication to the operator of where in the procedure the operator should be at any given time during input of information to the Input-Output Buffer. The indicator consists of a plastic engraved drum driven by a stepping switch. The drum advances one step each time the Enter key is depressed. Upon depression of the Nonmerchandise key, the drum advances to the nonmerchandise field. Upon depression of the Total key, the drum advances to its original start position.

The punched-tag reader was designed to accept the twenty-five column, Dennison tag, or abbreviation of it. When a sales person or customer number tag is read, the tag is returned after read-in, so that it may be retrieved by the operator. In the case of the merchandise tags, however, the tag is read and retained in the machine, which prevents littering the counter with merchandise tags. Since all of the tag information has been recorded on magnetic tape, the merchandise tags may be removed from a bin and discarded at the end of the day. The first character of any tag indicates which of the three types it is and if the wrong tag is inserted, the logic of the machine is such that the tag reader stops, and the Clear key must be depressed in order to retrieve the tag. The tags are read one step or character at a time, 7-1/2 characters per second. Again, as in the keyboard, the tag reader stops in a reading position and the output is verified as a valid binary number, prior to advancing to the next reading position. An electrical interlock between the tag reader and the keyboard prevents keyboard operation when a tag has been inserted in the reader. The actual reading of each character on the tag is accomplished by "sensing" the tag with five pins so connected through linkages as to actuate switches in the presence of holes.

When the Total key is depressed, the Sales Recorder is placed in the output mode causing the cash drawer to open and allowing the print-out of the sales check to begin. However, in a credit type sale, if the purchaser did not have a good credit rating, a Hold button will light, providing a bad credit indication to the operator, and preventing print-out. Print-out will proceed if the operator presses the Hold button.

The same mechanism that provides the character display for the keyboard is also the principle portion of the output printer. The output printer consists of ten numerical print wheels fabricated from nylon. These wheels are set up sequentially directly from the information sent by the Input-Output Buffer Unit. When a line of print has been set up, a print platen is released, the line is printed, and the paper advances to the next print position. The speed of this operation averages about six characters per second. An automatic overprint provides a visual indication on the check for credits or C.O.D. transactions.

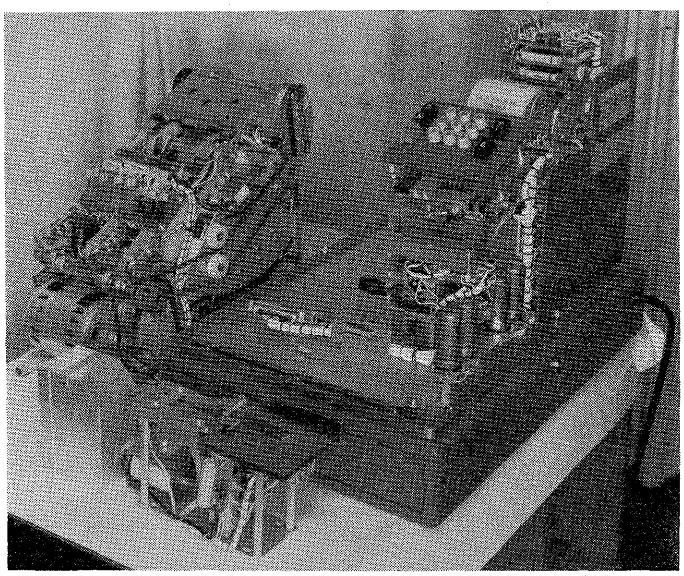


Fig. 3—Partially assembled unit with the Procedure Drum and keyboard in place and the tag reader and printer shown separately.

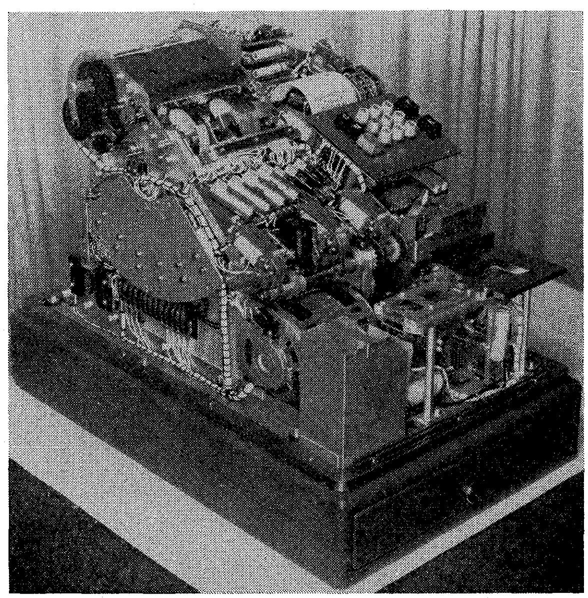


Fig. 4—Complete assembly with the cover removed.

The paper used consists of a three-part sprocket-fed preprinted form. The first copy is obtained through an ink ribbon impression and the back copies are carbonless paper, and require no ribbon for printing. Since the form is preprinted, the paper feed machanism is programmed in conjunction with the printer so as to print only in the correct blocks or spaces on the check and not in the preprinted portions.

The printer, keyboard, and tag reader are all separate units that plug into the base assembly. Fig. 3 gives an exploded view of the Sales Recorder while the complete assembled unit is shown in Fig. 4.

Buffer

The Buffer Unit is a multiplexing device which allows the transfer of data to and from as many as ten Sales Recorders. Working on a time-sharing basis, the buffer permits independent operation of each Sales Recorder. A block diagram of the Buffer Unit is shown in Fig. 5.

The communication between the Sales Recorder and the Buffer Unit consists of information flow within a closed loop.

At both receiving ends of the loop the character is checked for parity errors. If wrong parity is sensed, the character is rejected and an error displayed at the Sales Recorder. The keyboard of the Sales Recorder remains locked until the error is cleared. As mentioned earlier, the return of the correct character from the Buffer Unit to the Sales Recorder unlocks the keyboard so that another character may be entered. This double check, together with the relatively high power used in the transmission and the low impedance at both ends, makes the system very reliable from the point of view of the data exchanged, and insensitive to noise and crosstalk.

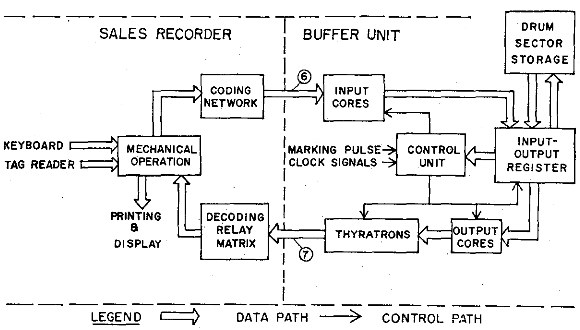


Fig. 5—Block diagram of the Buffer Unit.

The two information trunks are actually contained in a single cable that connects each Sales Recorder, via a junction box, to its designated input terminal at the Buffer Unit. To best describe the functioning of the Buffer Unit, its operation will be divided into two modes:

1) Input: receiving information from the Sales Recorder—or entering transaction data, and
2) Output: sending the processed information to the Sales Recorder—or the printing of the sales slip.

Let us consider the input mode. The 6-bit binary character in the form of voltage levels created by either depressing a key or reading a character from a tag in the tag reader is transmitted by cable directly to the input magnetic core bank. (See Fig. 5.) There is a bank of input cores associated with each Sales Recorder. The cores serve the dual function of noise suppression and gating.

Also associated with each Sales Recorder is a portion of a magnetic drum called a sector. That part of the drum containing all the sectors is designated as the Sector Channel. Each sector has a storage capacity of 360 5-bit characters. The sector is divided into preassigned fields corresponding to 1) the items listed on the Procedure Drum for input information, and 2) those items required for output information. These variable fields, once established for the desired application, are fixed in length. Thus, for instance, there is a three-character field for the sales person number. If insertion of a fourth digit is attempted, the Clear light at the Sales Recorder will light, indicating that the capacity of the field has been exceeded.

A special indexing track on the drum, called the Marking Pulse Track, provides the indexing mark within each sector to indicate the location of the character last operated on and the field starting position.

A magnetic core shift register in synchronism with the magnetic drum provides a read-out pulse that transfers the information from the input cores to the Input-Output Register at the time that the sector associated with the respective Sales Recorder is accessible. Since the arrival of the information from the Sales Recorder is asynchronous with respect to the read-out pulse, core logic is provided to assure that a complete character is actually placed in the Input-Output Register. Once in this Register, the information is checked for parity, and if an error is sensed an error control flip-flop is set.

If a numeric character is present, it is written on the sector at the location specified by the marking pulse. However, if the character in the register is one of the six operational commands, the matrix in the control unit is activated, so that the specified command level is generated. These command levels enable their related logic to perform the required function. In either event, a parity error prevents the processing of the character in the register.

The information transferred at the start of the sector cycle remains in the Input-Output Register for a time interval equivalent to a sector period, approximately 4.1 msec. All operations pertaining to the character received are executed within this period. Therefore, it is the Input-Output Register that is being time shared, allowing the sequential sampling of information in each input core bank.

Clock signals are provided so that synchronous timing with the magnetic drum occurs.

Just prior to clearing the Input-Output Register, near the end of the sector period, the contents are transferred to the output core bank. If the error control flip-flop was set, the Input-Output Register is cleared before the contents are transferred, so that the output core bank contains all zeros. Shortly thereafter, the information is read out of the cores to fire their associated thyratrons, thus forming the return character which activates the decoding relay matrix in the Sales Recorder. This in turn terminates the transmission of the character to the Buffer Unit, and also extinguishes the thyratrons by removing their plate voltage. As was the case for the input cores, there is a bank of output cores and thyratrons associated for each Sales Recorder.

The information levels transmitted by the Sales Recorder remains present for several complete auxiliary memory cycles, about 130 msec, at which time an error indication is made, unless the correct character echo is returned earlier. Since all zeros are returned to the Sales Recorder when parity error is detected, the information transfer is not interrupted. Therefore, a second or third chance is afforded to correctly process the transmitted character, and thereby minimize the possibility of transient errors stopping the operation.

Let us now consider the output mode where information is being printed on the sales slip. In the output mode, the information to be transferred to the Sales Recorder originates from the sector storage. However, a character is only transmitted when a request for information is made by the Sales Recorder, and then only one character is transferred per request. The buffer, in processing this request, will allow a character in sequence, as indicated by the marking pulse, to be transferred from the respective sector to the input-output register. Once in this register, it can be read into the output cores and the thyratrons fired in much the same manner as is done during the input mode.

The central computer processes a transaction only if the input of data from a Sales Recorder has terminated. This is signified by a total symbol entered at the beginning of the respective sector when the total key at the Sales Recorder is depressed. Before the buffer can respond to the Sales Recorder's request for a character, the processing of its associated sector by the computer must be completed. After the central computer has signaled that this processing is completed, it has no further control over the respective sector.

The seventh line in the output trunk furnishes the overprint level to the Sales Recorder. Another thyratron per Sales Recorder is provided and operated by a special control flip-flop. This control storage is activated by a program controlled character coming from the sector storage.

Each sector may be in a different operational state at a given time independent of one another, and thus there is no interference or interruption of communication between a Sales Recorder and its associated sector storage.

Recorder Central

The central computer, referred to as Recorder Central, is a general-purpose internally programmed digital device with a fixed order code. As shown in Fig. 6, it comprises a magnetic drum, an arithmetic unit, a control unit including a clock and control pulse generator, a small high-speed magnetic core memory, and an operator console to provide program and operator control. The magnetic drum contains, in addition to the sector channel, a random access stock and credit reference file, the program storage, the work space for transaction processing, and the necessary timing tracks. The small high-speed memory of twenty-character capacity is used for all operations, except transfers within the drum.

The computer is a one-address, variable word, numeric machine. An instruction word consists of an order code of two characters, and an address area consisting of four characters. The order code was specially designed to facilitate file processing as well as rapid calculation. The order code contains instructions for communication between a Sales Recorder sector and the computer, arithmetic computations, decision and control operations, file processing, and console input and output via paper tape and monitor printer.

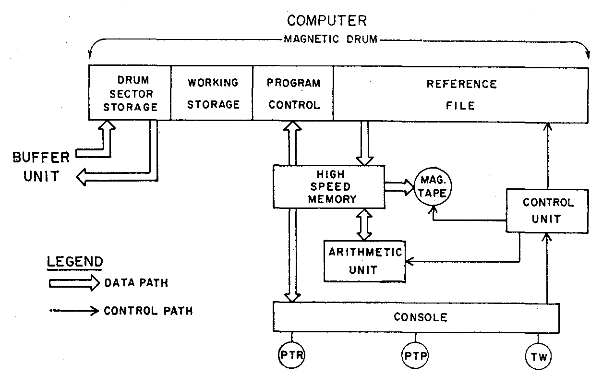


Fig. 6—Block diagram of the Recorder Central.

Upon recognition of the Total symbol by the computer, the entire contents of that sector are transferred to the working storage. The sequence of words and their positions within the sector remain the same. Thus, transaction processing requires a minimum of editing and rearrangement for output printing.

The input data in the working storage is analyzed to determine how the transaction is to be processed. If the transaction requires the verification of a customer's credit, the customer's charge number is processed against a credit exception file. All stock numbers are passed against the stock reference file to determine price and city, state and federal tax information. These data are obtained for all merchandise items sold in the transaction. Prices are extended, subtotal and total calculated, and required information listed for printing on the sales slip, after which the contents of the working storage are written out to the transaction record magnetic tape. After receiving a check signal from the tape station, the information in the working storage is then transferred to the respective Sales Recorder sector and the Input-Output Buffer notified that transaction processing is completed.

To determine the beginning and end of the variable sized items on the drum processed by the Recorder Central, item markers are used. The working storage can be changed in layout to represent any kind of sales check or business form corresponding to the Sales Recorder sector layout. The ability to vary the reference storage message sizes to conform to variable word requirements allows great efficiency to the use of the drum.

Access to the file storage is hastened by avoiding long indexing searches. Messages within the file may be either extracted, deleted, changed, or added, by separate orders. Variable sized criteria may be used with these instructions to extract desired information. Thus, one may be interested in all swim suits, in all bikini swim suits, in all bikini swim suits with blue polka dots, by adjusting the criterion accordingly.

To gain access to the desired data of a message, a mathematical transformation on the criterion of the message is used. This allows minimum delay in locating the messages and thus speeds up over-all transaction processing. More important, however, by avoiding the use of indexing routines, messages can be entered into the reference storage or extracted from it without the requirement of prior sorting and collating. Thus, the problems of file maintenance are considerably simplified and external processing is appreciably reduced.

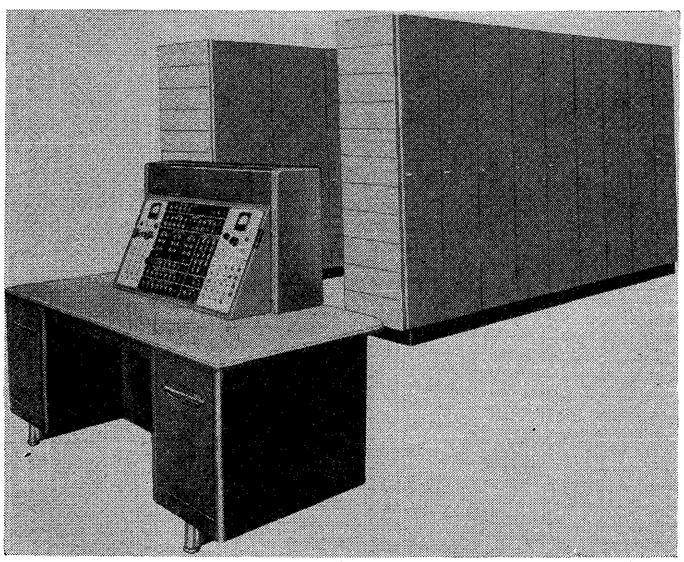


Fig. 7—Recorder Central and Input-Output Buffer Racks with console and power supply.

The control console may be used to monitor system operation or to provide means for manual control of the Recorder Central. A view of the Recorder Central is provided in Fig. 7. All control flip-flop indicators in the machine are displayed as an aid in maintenance and program debugging. All areas of the computer may be interrogated from the console. Information can be introduced either manually or by means of paper tape. A monitor printer is provided for printing the contents of the high-speed memory or any portion of the auxiliary memory when desired. Marginal checking facilities are also controlled from the console.

All transaction processing operations are carried out automatically. If information concerning daily transactions is required at any time by the Electronic Data Processing System the current transaction record magnetic tape can be remotely disconnected, and a new tape connected, by the Sales Recording System.

The Recorder Central contains many built-in checking features. Redundancy checking is used throughout the equipment to determine errors in transmission of characters and to isolate their sources. Arithmetic operations are repeated and results compared. Orders are checked before they are carried out. These, among others, are designed to insure against incorrect processing. However, in a system used for on-line processing the ability to maintain continuous operation is of paramount importance. Thus, the Recorder Central is designed to attempt to overcome any error a fixed number of times before it will stop operation. This will discriminate between transient errors and those due to catastrophic breakdown. In the latter case, a complete set of machine status indicators is available at the console, specifying exact portions of an order in which failure had occurred for ease and rapidity of maintenance. Plug-in type module construction is used to facilitate troubleshooting, preventive maintenance, and replacement in case of failure.

Conclusion

The Sales Recording System represents a great step forward in providing the means for data integration within a department store. The Associated Merchandising Corporation's research installation has demonstrated beyond a doubt that on-line Sales Recording systems are a reality. Here, for the first time, a variety of transaction types, as broad as the store desires, may be processed directly from a point-of-sale unit, with complete computation performed by a fast, accurate, and versatile high-speed computer, including an automatically printed sales slip.

But more than this, the point-of-sale unit can be used to either interrogate the reference file and thereby gain immediate access to any desired information, or actually enter new reference information, directly from its remote location.

It is important to note that this information need *not* be in any ordered form. Furthermore, a magnetic tape record, made for each transaction entered, provides a direct and reliable means of furnishing the information to an Electronic Data Processing System for data handling. The Electronic Data Processing System can have access to the magnetic tape at any suitable time without interrupting transaction processing by Recorder Central.

Acknowledgment

The authors wish to acknowledge the contributions of Albert Burstein, Edward Damerau, Robert Grapes, Andrew Ling, and Felipe Tanco in preparing this paper.

Discussion

J. T. Wallace (Eastman Kodak Co.): Is the Dennison tag reader available separately and what is the general means of reading the tag?

Answer: Yes, it is. The reading is accomplished by pins going through the holes and consequently making switches.

N. J. Dean (Ramo-Wooldridge Corp): What is the capacity, speed, and form—tape, drum, etc—of the random access reference file?

Answer: The random access reference file which is a portion of the drum has a capacity of approximately 72,000 characters. The speed is approximately 1500 rpm with a character rate of slightly less than 100 kc.

Question: How many characters in a complete transaction? If more than one, is the entire message buffered before presentation to the computer?

Answer: There could be up to 360 characters. That is the full capacity of a sector, there being 10 sectors for the sector storage.

The normal transaction will have a smaller percentage and therefore the sector capacity was laid out so that you could process a maximum of five merchandise transactions and five nonmerchandise type transactions per sale.

An entire message is retained in the sector until the "Total" indication has been given. At this time, the complete sector is transferred ino the "Working Storage."

Mr. Dunham: Do the marking pulses and clock signals originate from the magnetic drum?

Answer: Yes.

Question: Will this point-of-sale unit include customer billing?

Answer: Ultimately it could, by providing all the necessary information to an Electronic Data Processing System.

Question: How do you plan to handle the sales slip; namely, is a copy to be returned to the customer—a generally accepted billing practice?

Answer: One copy of the sales slip is given to the customer at the time of purchase. The other two are retained by the store; one to be included with the monthly statement for charge sales, and the other for record purposes.

Question: I would like an idea of the general size and cost of a complete system for a medium size department store.

Will one computer handle any number of sales recorders? How many pieces of intermediate equipment are required?

What provisions would be made for leaving an alternate means of recording sales in the event of computer failure for even a short time during the day?

Answer: For cost figures our sales department should be contacted.

The number of Sales Recorders that may be handled by a single computer is a function of the computer handling rate as well as the random access rate to the reference file.

A paper tape punch could be used as a stand-by unit to capture the required information for future data processing, while the sale may be handled by normal sales book method.

Question: What is the expected cost on each point-of-sales recorder?

Answer: For cost figures our sales department should be contacted.

Question: Do you intend to use telephone lines to connect a remote point-of-sales recorder to files and computer?

Answer: Long-distance hookup, that is, beyond the confines of a store, has not been considered as yet.

Question: How many point-of-sales recorders can be attached to your system?

Answer: The system is a pilot system, and therefore provision for only ten Sales Recorders was made.

Question: What is the maximum number of input-output units you can couple to the described system?

Answer: For this pilot system, thirty Sales Recorders were set as the maximum, and still maintain an average transaction rate of one a minute with an average queueing rate of one second for the transaction processing.

Question: What provisions are made for multiplexing or sequencing between the many point-of-sale units and the central computer?

Answer: Each Sales Recorder has a sector allocated to it. When the sector comes under the read heads, the associated Sales Recorder will receive information if it's in the output mode or furnish information if it's in the input mode.

Question: Can a department store economically justify the use of these Sales Recorders?

Answer: This is a question for the individual department store to answer.

Question: How much time is required to process a transaction at the point of sale? Did you find that this time requirement was in any way objectionable to the customer?

Answer: The transaction time varies with the data to be entered, of course. However for average length transactions a rate of less than one minute per transaction may be sustained. This was found to be in no way objectionable to the customer.

Question: If the point-of-sales unit can enter new reference information, as on credit, can a customer operate one point-of-sales unit in the absence of the salesman and thus improve his credit rating prior to making a large purchase?

Answer: No. Although special coding is provided to allow entry of data from the Sales Recorder, it requires a knowledge of the code (nonmerchandise code) and a special sales person number. Adequate programming checks are provided to accomplish this.

Organization of Simulation Councils, Inc.

THIS YEAR Simulation Councils, Inc., accepted with great pleasure S. N. Alexander's (1957 EJCC Chairman) invitation to hold its annual meeting in conjunction with the Eastern Joint Computer Conference. This allowed members of the regional groups to meet at a national conference where all members had a definite interest. The Simulation Councils participated in two of the formally scheduled sessions, and in addition, four informal sessions were held at which papers were presented by its members.

The following background information is provided to familiarize members of the AIEE, ACM, and IRE with the objective of the Simulation Councils.

Simulation Councils, Inc., is an organization formed to improve communications at the working level concerning methods of using simulation techniques and equipment to facilitate the study, design, test, and analysis of physical systems. These objectives are quite broad and naturally are very much related to the use and application of electronic computation.

The beginning of this organization occurred in November, 1952, when a group of persons associated with the operation of general purpose analog computers decided to hold informal meetings for discussing the details of problems and methods for their solution, equipment, and new ideas. It was decided that the minutes of such meetings would be published in an informal newsletter which would be made available to all those interested, including those who were unable to attend.

The group holding this first meeting evolved into the Western Simulation Council. Following the same concept and procedures, five other Councils were organized: the Midwestern Simulation Council, Eastern Simulation Council, Central Simulation Council, Southeastern Simulation Council, and the Canadian Simulation Council. All these Councils have joined into an international group which is called Simulation Councils, Inc. The purpose of this group is to provide a medium for publishing the minutes of the meetings of all regional groups, and to arrange annual meetings to which all the regional groups are invited.

The entire concept of the Simulation Councils is based on informality. Membership in any one of the regional groups may be obtained by attending a meeting and signing the attendance register, or by requesting the Chairman of that Council to place your name on the mailing list.

Meetings are held approximately every second month with from 20 to 100 members in attendance. The meetings usually are held at the computing facilities of organizations interested in simulation. An attempt is made to choose different hosts for each meeting so members may visit a wide variety of simulation and computational facilities. Meetings usually begin with brief talks on a subject selected to promote discussion, and an attempt is made to encourage all present to join in the discussions. This informal discussion is the essence of the Simulation Councils effectiveness; everyone gets a chance to compare his technique and equipment with others having similar interests and problems.

The *Simulation Council Newsletter,* as informal as the Council discussions, has appeared every month since November, 1952. At first it was privately published by mimeograph, but increased demand made this impractical and since April, 1955, it has appeared as a separate and editorially autonomous section of *Instruments and Automation.* Individual issues of the *Newsletter* are as different as the wide variety of subjects and the personalities of people discussing them. The result is that while reporting month after month the offhand remarks as well as the serious thoughts of those developing the techniques and designing the equipment for simulation, the *Newsletter* reflects progress in the allied fields of analog and digital computation and data processing as well.

Anyone interested in the Simulation Councils is invited to contact the Steering Committee Chairmen of the various regional groups listed below.

Western Simulation Council
Dov Abramis, Convair, Pomona, Calif.

Midwestern Simulation Council
Warren Jackson, Standard Oil Co., Midland Building, Cleveland, Ohio

Eastern Simulation Council
Hideo Mori, Hydel, Cambridge, Mass.

Southeastern Simulation Council
Robert Johnson, Georgia Institute of Technology, Research Area 4, Atlanta, Ga.

Central Simulation Council
James Pierce, Beech Aircraft, Wichita, Kan.

Canadian Simulation Council
F. W. Pruden, Analog Computation and Simulation Group, Mechanical Engineering Division, National Research Council, Ottawa, Ont., Canada

Abstracts of Papers

Papers presented by members of the Simulation Councils, Inc., at the four informal sessions during EJCC are abstracted on the following page.

1) *Physical Simulation in Airplane Control System Problems,* P. G. Hurford, McDonnell Aircraft Corp. This paper describes the use of physical simulation including the pilot as an aid in the solution of lateral control problems of modern jet fighters. To accomplish this, an analog computer and control system are combined with a movable chair which imparts to the pilot the "feel" of rolling motions. The effects of various chair, control systems, and airframe parameters are determined with this system.

2) *Design and Utilization of a Three-Axis Simulator,* M. Paskman and R. Edwards, Aircraft Armaments, Inc. The time and cost savings resulting from utilization of a three-axis simulator are discussed. The design evolution of a three-axis, gimballed system is described, including specifications, mechanization, and analysis of the system. Test procedures are outlined, after which the use of a general purpose analog computer to solve typical control system problems is described.

3) *Synthesis of Closed-Loop System by Means of Analog Computers with Real Gyros or Accelerometers in the Loop,* B. W. McFadden, Micro Gee Products, Inc. Presented are the advantages and techniques in the synthesis of a closed-loop system (such as an autopilot or autonavigational system) by including the real gyro or accelerometer in the loop with an analog computer. Also given are the characteristics of a simulation table with a threshold of less than one second of arc which makes it possible to determine closed-loop performance about null under the influence of real, small discontinuous nonlinearities such as friction, noise, and deadband.

4) *A Discussion of the Procedures and Practical Problems Relating to Real-Time Simulation Using Control System Hardware,* Eaton Adams, Jr., Convair, Fort Worth, Texas. The techniques and problems involved in real-time flight simulation using hardware are described. A three-degree-of-freedom missile trajectory problem, including nonlinear aerodynamics, is used as an example. Problems of analysis and correlation with analytic results when nonlinear hardware is included in the simulation are discussed, along with other difficulties arising in such a simulation.

5) *A New Dead Time Simulator for Electronic Analog Computers,* Millard Brenner and Jerome D. Kennedy, Electronic Associates, Inc. Up to now, the accurate simulation of dead time, performance of auto and cross correlations, and long-term arbitrary function storage have been difficult to achieve on an electronic analog computer. This paper deals with the applications and operational principles of the SIMULAG, a variable delay, multichannel magnetic tape unit developed by Electronic Associates, Incorporated.

6) *A. C. Diode Function Generators,* C. L. Cohen and D. S. Peck, Nuclear Products—Erco Division of ACF Industries, Inc. Circuits employing silicon diodes as used in aircraft flight simulators are described. The circuits involved perform function generation such as slope changing, limiting, jump functions, absolute value generation, and magnitude selection.

7) *Future of the Transistor Analog Computer,* Robert Bruns and George Milligan, Jet Propulsion Laboratory, California Institute of Technology. A portable analog computer employing transistors has been developed to study the possibility of analog computers employing semiconductor devices only. The transistor operational amplifier and servomultiplier circuits for this computer are described, along with the performance capabilities of these components. The direction of future analog computer design is discussed.

8) *Open-Shop Operation of a Large Real-Time Simulator,* Stanley Rogers, Convair, San Diego, Calif. The problem of maximizing the use of simulators under open-shop conditions is reviewed. Topics include scheduling, operator training, equipment reliability and maintenance, problem-checking methods, and suggested future trends for large simulators.

9) *A Real-Time Simulation System for Use with an Analog Simulator,* Robert M. Beck and Max Palevsky, Packard-Bell Computer Corp. Reported are some newly developed digital devices which can be employed with analog computers. These devices include an extremely high speed incremental computer for computing nonlinear analytic functions, a function generator that employs photographic techniques for storage, and a 0.01 per cent analog-to-digital and digital-to-analog converter. The converter also performs multiplication and division within the conversion process.

10) *Simulation of Aircraft Landing Gear Dynamics on the Geda Analog Computer,* P. J. Hermann, Goodyear Aircraft Corp. A simulation of the vibration dynamics of a landing gear has been set up in order to study the effects of braking torques and other influences in exciting the landing gear into undesirable vibration amplitudes. Assumptions in formulating the physical model of the landing gear are discussed, as well as the development of the equations of motion. The difficulties in determining numerical values for the various constants are related, and the results of the computer simulation are presented.

11) *Direct Simulation on Analog Computers through Signal Flow Graphs,* Louis P. A. Robichaud, Canadian Armament Research and Development Establishment. For physical systems which can be considered as made up of combinations of various elements, it is not necessary to write out the equations before determining the analog computer circuit. A procedure is presented for going directly from a physical system to an analog computer representation of that system. Each physical unit is represented by a computer unit such that all terminal variables are in evidence in that unit as they are in a matrix or flow graph representation of the physical unit. The computer units then are interconnected in a manner similar to the physical units.

12) *The Use of Quaternions in Simulation of the Motion of Rigid Bodies,* A. C. Robinson, Wright Air Development Center. In simulation involving real three-dimensional coordinate transformations, it is customary to represent orientation by Eular angles or direct cosines. The use of the four-parameter quaternion notation which avoids both "gimbal lock" and redundancies is outlined. It is shown that from the standpoint of accuracy and amount of equipment required, the quaternion method is superior to either of the two usual techniques. Other advantages and limitations are set forth.

13) *On the Loop and Node-Analysis Approach to the Simulation of Electrical Networks,* Joseph Otterman, University of Michigan. Simulation of an electrical network on an analog computer by the loop or node-analysis approach is quite often unsatisfactory because of hidden regenerative loops. Such regenerative loops arise when the computer setup contains more integrators than the order of the differential equation describing the network. Instability may result because the actual computer components in the loop containing an excess integrator depart from ideally exact, prescribed values. Presented is a procedure for tracing the loop currents in such a way that there is one-to-one correspondence between the number of integrators in the simulation setup and the count of independent energy-storing elements in the network, *i.e.,* the order of the differential equation describing the network. The generality of the procedure proves that it is always possible to trace the loop currents in such a way that excess integrators are avoided. A parallel procedure for node analysis is discussed briefly.

14) *Application of High Speed Compressed Time Scale Computer to Engineering Problems,* Joseph Miasnick, General Electric Co. Application is made of a high-speed compressed time scale computer to certain engineering problems. The place of this type of computer in the over-all problem-solving picture is described. Specific application is made to the simulation of the performance of a starter-engine combination. The simulation is used to determine the effect in performance of varying cutout speed, torque-speed characteristics, and engine application. The computer also is used to determine the inertia of the test stand.

15) *Simulator for Use in Development of Jet Engine Controls,* Emile S. Sherrard, National Bureau of Standards. Recommendations and cost estimates for a simulator capable of representing a twin-spool, after burning, turbojet engine and its controller are given. The simulator is intended as a tool to be used by a group of engineers engaged in developing engine control systems. The simulator is designed to determine stability and performance of the engine control system, to be useful in both early and late stages of the development, and to operate in real time so control hardware may be operated with a simulated engine.

16) *Ducting Air-Flow Characteristics By Electrical Circuit Measurements,* William H. Sellers, General Electric Co. The well-known dynamical analogies are developed and reviewed for electrical, mechanical, and acoustical systems. The analogous electrical circuit then is derived for a physical system consisting of a cylindrical tube, containing a butterfly valve arbitrarily located within the tube. An actual circuit consisting of inductances, capacitors, etc., is then constructed and the values of voltages and currents are recorded by a brush recorder. These recorded values are analogous to the pressure and flow of the fluid within the duct. Photographs are presented depicting the actual wiring and measurement arrangements. Also, a comparison solution, obtained by a standard analog computer, relates the relative error and reliability of the direct circuit measurement method.

17) *The Real Time Simulation of Turbo-Jet and Turbo-Prop Engines and Controls,* D. L. Dresser, Allison Division, General Motors Corp. Some problems faced by the designer of control systems for turbo-jet and turbo-prop engines are outlined. The use of the analog computer as a tool in designing controls for future engines and testing hardware is discussed. Two specific examples of full-scale simulation are presented. The first one is a turbo-jet simulation based upon component turbine and compressor date, and the second one is a turbo-prop simulation which uses test-stand dynamometer data. Also, two linearized procedures are given. Computer techniques for representing a variety of nonlinearities are described.

18) *Noise and Statistical Techniques in Analog Simulation,* Henry Low, The Ramo-Wooldridge Corp. The role of the analog computer in obtaining the impulse response of a linear system is discussed, including the adjoint method for obtaining the impulse response of time-varying linear systems. Techniques for treating nonlinear systems on an analog computer are discussed. The types of statistical quantities corresponding to random inputs to the analog computer are described, and an illustrative example is worked out. The theory of operation of various types of analog computer noise generators is explained. Techniques for measuring the statistical characteristics of low-frequency noise generators are presented.