3400

Control Data® 3400 Computer System
Preliminary Reference Manual

3400

Control Data® 3400 Computer System
Preliminary Reference Manual

This manual provides information for the machine-language use of the 3400 computer. Its intention is to describe the capabilities of the hardware. Options and constraints for programming are noted. Some programming examples are given to illustrate how instructions perform.

Other than using COMPASS mnemonics to abbreviate titles of instructions, no softwear systems are used in describing instructions.

The information is preliminary and subject to change.

December, 1963
Publication No. 282b
(Pub. No. 282b obsoletes Pub. No. 282a and includes all corrections and additions.)

Printed in United States of America

Address comments concerning this manual to:

Control Data Corporation
Technical Publications Department
501 Park Avenue
Minneapolis 15, Minnesota

RECORD OF CHANGE NOTICES

C. N. NO.	DATE ORIGINATED	DATE ENTERED	INITIALS	REMARKS

CONTENTS

Chapter VII – 3401 Console

Chapter VIII – 3655 High-Speed Line Printer

Chapter IX – 3691 Paper Tape Reader/Punch

Chapter X – 362X Magnetic Tape Controller

FIGURES

TABLES

CONTROL DATA® 3400 COMPUTER SYSTEM
PRELIMINARY REFERENCE MANUAL

CHAPTER 1
BASIC SYSTEM DESCRIPTION

The CONTROL DATA* 3400 is a solid-state, stored-program, general-purpose digital computing system, with large storage capacity and fast data transmission and computation speeds.

The 3400 system incorporates features of the CONTROL DATA 3600 Computer to provide program compatibility with this machine. With the several available options, a variety of configurations is possible.

3400 SYSTEM CHARACTERISTICS

- Stored-program, general-purpose computer
- Parallel mode of operation
- Single address logic
- 51-bit storage word (48 bits of data, 3 parity bits)
- Six 15-bit index registers
- Indirect addressing
- Magnetic core storage
  - 32,768 51-bit words (Standard)
  - 16,384 51-bit words (Optional)
- Input/Output
  - Transmission of 48-bit words (12-bit bytes)
  - Up to four separate bidirectional input/output channels
- System interrupt
- Flexible repertoire of instructions
  - Fixed point arithmetic (integer)
  - Single-precision, floating-point arithmetic (optional)
  - Logical and masking operations
  - Character handling instructions
  - Indexing
  - Storage searching
- Binary arithmetic
  - Modulus $2^{48} - 1$ (one's complement) for single precision operations
- Completely solid-state
  - Diode logic
  - Transister amplifiers
  - Ready access to circuits
- Console includes:
  - I/O Typewriter and display panel
- Inter-computer communication

3400 ↔ 3600	Via 3682 Satellite Coupler
3400 ↔ 3400	
3400 ↔ 160/160-A	Via 3682 Satellite and 3681 Data Channel Converter

* Registered trademark of Control Data Corporation.

BASIC 3400 SYSTEM

The basic 3400 system consists of a central computer, an input/output section, magnetic core storage, and a console. Over-all system operation depends on the integral operation of these elements. (See figure 1-1.)

Included in the basic computing system are operator and maintenance consoles. The consoles contain all the controls and indicators necessary to operate the system.

OPTIONS

For greater systems capability, the 3400 computing system may be expanded with several available options. These options are:

1) Floating point option (3410)

   This adds four single-precision, floating-point instructions.

2) 16,384 word storage

   The standard 3400 system includes 32,768 words of core storage. An optional system is available with only 16,384 words of core storage.

3) Additional Input/Output channels (3406)

   The basic system may be expanded to a maximum of four bidirectional I/O channels.

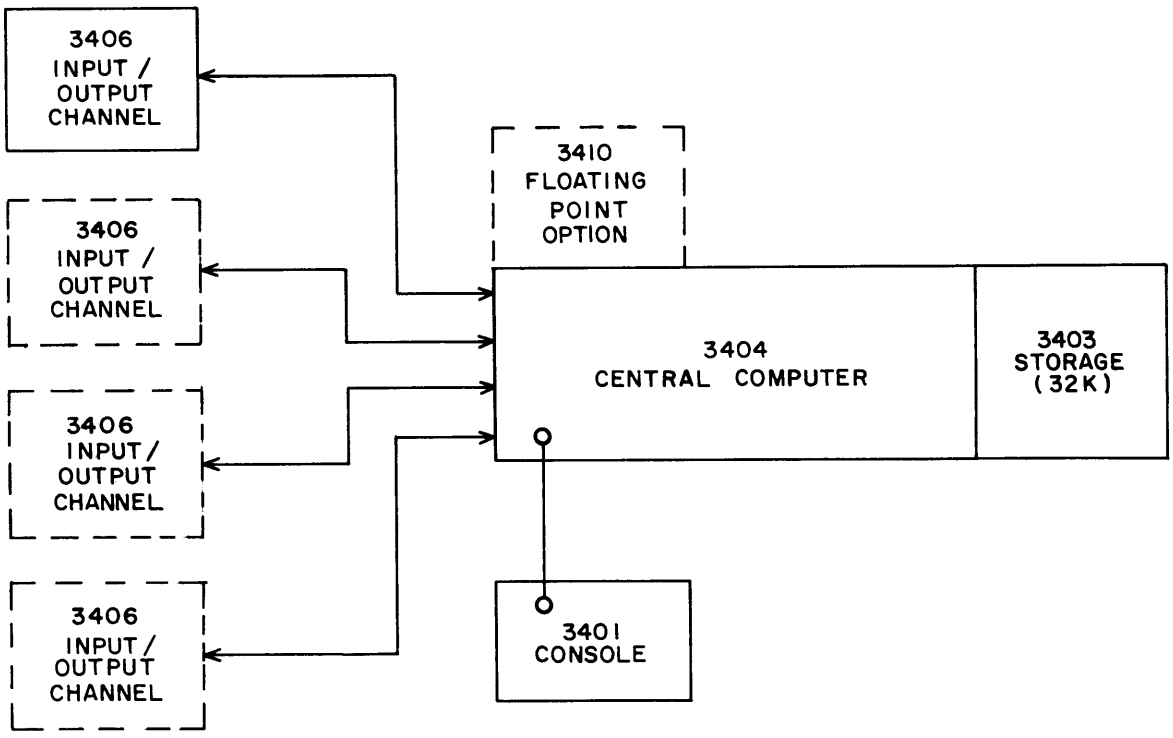


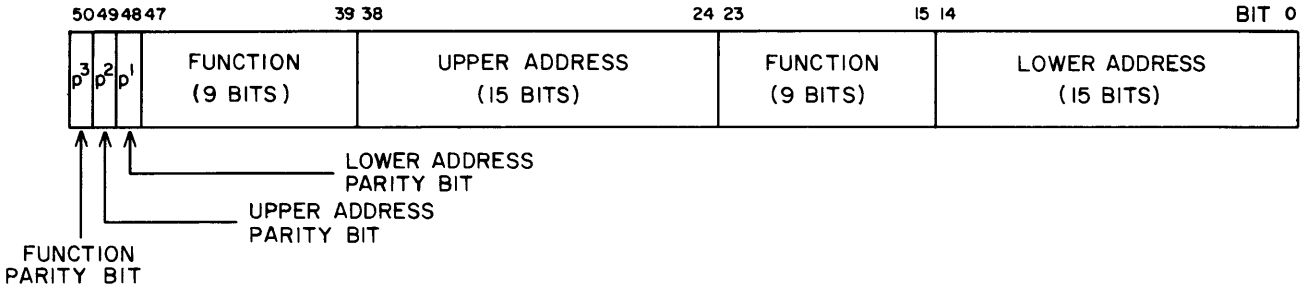
Figure 1-1. 3400 Computer System
(Dotted Lines Indicate Optional Additions)

CHAPTER II
STORAGE SECTION

The magnetic core storage section provides high-speed, random-access storage for 32,768 words. The storage section consists of the storage elements themselves, and the circuitry for addressing the storage elements.

STORAGE WORD

A storage word may be two 24-bit instructions, a single 48-bit instruction, or a 48-bit data word. Three parity bits are appended to each 48-bit word; thus a storage word is 51 bits in length. The format of a typical storage word is diagrammed below.



The storage word is divided into three portions:

1) a 15-bit lower address,

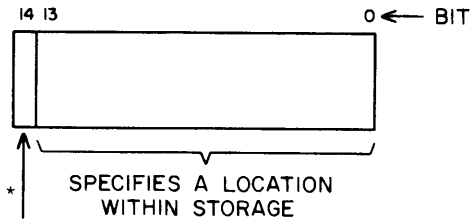
2) a 15-bit upper address, and

3) an 18-bit function portion, distributed in the storage word as diagrammed. A parity bit accompanies each of these portions when the word is stored. The parity bit (P1) associated with the lower address portion is placed in bit 48 of the storage word, parity bit P2 (upper address) is placed in bit 49, and parity bit P3 (function) is placed in bit 50.

When part of the word or the entire 51 bits is read from storage, the appropriate parity bit(s) accompanies the word and the computer checks the word for parity.

STORAGE ADDRESSING

The location of each word in storage is identified by an assigned number (address). An address consists of 15 bits interpreted as shown below:



The following storage locations are used with return jump instructions for interrupt processing (unless the

---

*1) In a 3400 system with 32,768 words of core storage: If this bit is a "0", it specifies the lower section of storage which contains addresses 00000_8--37777_8; if this bit is a "1", it specifies the upper section of storage which contains addresses 40000_8--77777_8.

2) In a 3400 system with 16,384 words of core storage, this bit is always "0". It specifies the one storage section which contains addresses 00000_8--37777_8.

interrupt category is removed by adding an option):

00007 } Category I Interrupt

00030
00031 } Category II Interrupt
00032
00033

00020 } Category III Interrupt

See Chapter 4 for additional information on the uses of these storage locations.

Address 000008 is reserved for use with a return jump instruction when the Restart switch is depressed. (See chapter 7.)

CHAPTER III
COMPUTATION SECTION

The computation section performs calculations and processes data in a parallel binary mode through the step-by-step execution of individual instructions. The instructions and data are stored in the storage section(s).

LOGICAL DESCRIPTION

Functionally, computation section may be divided into an arithmetic section and a control section.

Arithmetic Section

The arithmetic section performs the arithmetic and logical operations necessary for executing instructions. It consists primarily of several operational registers. The operational registers are described below. Table 3-1 lists the arithmetic properties of the registers.

A Register

Nearly all arithmetic and logical operations use the 48-bit A register (Arithmetic register). The contents of this register may be shifted to the right or to the left, separately or in conjunction with the Q register. In certain conditional instructions, the A register is used to hold control quantities which govern operations.

Q Register

The 48-bit Q register (Auxiliary Arithmetic register) assists the A register in performing arithmetic and logical operations. The contents of the Q register may be shifted right or left, separately or in conjunction with the A register. Q may also be used with the A register to form a double length register, AQ or QA. In addition to assisting the A register, certain instructions reference the Q register directly.

P Register

The 15-bit P register functions as a program address counter. The P register holds the address of each program step. After executing the instruction (or instructions) contained in the program step, the quantity in P is advanced by one to the address of the next instruction.

When a jump condition is met, the P register is set to the quantity specified by the execution address of

Table 3-1. Arithmetic Properties of Registers

Register	No. of Stages	Modulus	Complement Notation	Arithmetic	Result
A Register	48	$2^{48}-1$	one's	subtractive	signed*
Q Register	48	$2^{48}-1$	one's	----	signed
P Register	15	2^{15}	two's	additive	unsigned
Index Registers	15	$2^{15}-1$	one's**	----	----

* The result of an arithmetic operation in A satisfies $A \leq 2^{47}-1$ since A is always treated as a signed quantity. When the result in A is zero, it is always represented as 000...000 except when 111...111 is added to 111...111. In this case, the result is 111...111 (negative zero).

**Though the index registers have no arithmetic capabilities themselves, address modification using the index registers is performed modulus $2^{15}-1$ (one's complement).

the jump instruction. If the instruction is a return jump, the contents of P are stored before executing the jump, permitting a return to the program sequence after the jump is made.

Since the P register is a two's complement additive register, it can generate storage addresses in sequence from 00000 to 77777_8. When a count of 77777_8 is reached, the next count in P reduces its value to 00000. (Note that in generating storage addresses by adding the contents of an index register to a base quantity, address 77777_8 cannot be reached. Refer to the section on Address Modification Modes, (p. 3-5).

U Register

The 48-bit Program Control register (U) holds the program step while it is being executed. All operations necessary to execute an instruction are governed by the contents of this register.

B^1 - B^6 (Index Registers)

Six 15-bit index registers may be used to:

1) Hold quantities used as address modifiers.
2) Hold control quantities for certain instructions.

The index registers may also be explicitly referenced by certain instructions (refer to Repertoire of Instructions section).

Interrupt Register

Each interruptible condition in the system is connected to a particular bit position of the Interrupt register. The lower bit positions detect internal interrupt conditions such as overflow, divide fault, and exponent fault. The upper bit positions are interrupt lines coming from each of the four possible communication channels.

Interrupt Mask Register

This register enables testing of external interrupt lines and internal conditions. The bit positions of this register match the Interrupt register. Interrupt occurs on a condition if the bit of the Mask register is set to "1". The Internal Function instruction may be used to set or clear bits in this register.

Product Register

The product register contains the bit-by-bit logical product of the Interrupt register and the Interrupt Mask register (refer to the Interrupt section).

Bounds Registers (Upper and Lower)

Two bounds registers serve as a memory and jump lock-out. The Lower bounds register holds an 8-bit Lower bound address. The Upper bounds register holds an 8-bit upper bound address. The upper 8 bits of the 15-bit storage address S are compared with the contents of the bounds registers when bounds checking is in effect.

Control Section

The control section of the computer directs the operations required to execute instructions and establishes the timing relationships needed to perform these operations in the proper sequence. It also sends the preliminary commands necessary to begin the processing of input/output data.

The control section acquires an instruction from storage, interprets it, and sends the necessary commands to other sections. A program step may be a single 48-bit instruction or a pair of 24-bit instructions which together occupy a single storage location as a 48-bit word.

The program address counter, P, is a two's complement additive register. It provides program continuity by generating in sequence the storage addresses which contain the individual program steps. Usually, at the completion of each program step, the count in P is advanced by one to specify the address of the next program step.

The Program Control register, U, holds a program step while it is being executed. If the program step is a pair of 24-bit instructions, the upper instruction is executed first, followed by the lower instruction.

DESCRIPTION OF INSTRUCTIONS

A computer word consists of 48 bits and may be interpreted as one 48-bit data word, a 48-bit instruction, or two 24-bit instructions.

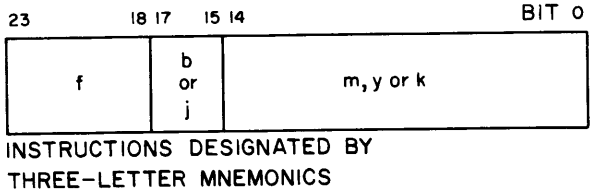
Most instructions designated by three-letter mnemonic codes are 24-bit instructions common to the 1604 and 3604 computers. These instructions are arranged in a 48-bit word; the higher order 24 bits are called the upper instruction and the lower order 24 bits are called the lower instruction.

Instructions which are not common to the 1604 computer and designated by mnemonic codes of three or four letters, differ in format and in word length (some are 24 bits; others are 48 bits).

Instruction formats are arranged in four major classes, according to differences in word length and the position of the function code within the format. A typical format from each class is outlined below. Designators used within these formats are explained at the end of this section. For a comprehensive description of instructions, refer to the Repertoire of Instructions section.

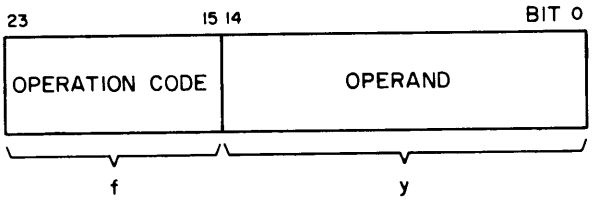
Class I

Class I instruction formats are 24 bits in length and have 6-bit function codes, 'f'. All instructions common to the 1604 and 3604 computers and designated by three-letter mnemonic codes are included in this category.



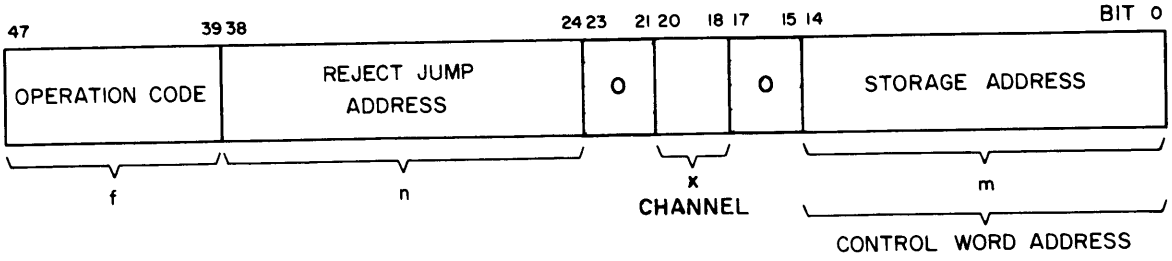
Class II

Class II instruction formats are 24 bits in length and have 9-bit function codes. All instructions in this category are designated by mnemonic codes of three letters.



Class III

Class III instruction formats are 48 bits in length and have 9-bit function codes. All instructions in this category are designated by mnemonic codes of four letters.



Class IV Miscellaneous

See pages 3-17, 3-18, 3-21, and 3-22.

DESCRIPTION OF DESIGNATORS

Designators used throughout the Description of Instructions section and in instruction formats are explained below. For specific interpretations of designators, refer to the individual instructions.

Designator		Use
b	Index	Specifies index register (B) used, or whose contents are used in the operation.
c	Connect and Function	Specifies codes used in Connect and Function instructions.
f	Function Code	A 6 or 9-bit code (depending on the operation) which specifies the operation to be performed.
j	Condition	Conditions operations in jumps and stops.
k	Unmodified Shift Count	Number of shifts to be executed.
K	Modified Shift Count	$[K = k + (B^b)]$
m	Unmodified Execution Address	Address of operand.
M	Modified Execution Address	$[M = m + (B^b)]$
v	Second Index	Specifies second index register (V) used, or whose contents are used in the operation.
w	Word Count	A 15-bit quantity which specifies the number of words to be processed in a transfer operation.
x	Channel Number	Specifies I/O channel; also used to specify channel whose status will be read or sensed.
y	Unmodified Operand	Used in execution address portion of instruction; specifies this address will be used as the operand.
Y	Modified Operand	$[Y = y + (B^b)]$

ADDRESS MODIFICATION

The portion of the instruction word designated by 'm', 'y', or 'k', is often termed the base execution address. The base execution address may be used as (1) a shift count, 'k', (2) an operand, 'y', (3) an address of an operand, 'm', in storage. The execution address may be modified or left unmodified depending on the index designator. The execution address is modified by adding the contents of the designated index register to the execution address. If left unmodified, the lower-case symbols 'k', 'y', or 'm', are used. If the address is modified, the symbols are capitalized.

The modified shift count is represented by:

1) $K = k + (B^b)$ where:

   K = modified shift count
   k = unmodified shift count (execution address)
   (B^b) = contents of index register b

If the index designator = 0, then K = k.

The modified operand is represented by:

2) $Y = y + (B^b)$ where:

   Y = modified operand
   y = unmodified operand (execution address)
   (B^b) = contents of index register b

If the index designator = 0, then Y = y.

The modified operand address is represented by:

3) $M = m + (B^b)$ where:

M = modified address of operand
m = unmodified address of operand (execution address)
(B^b) = contents of index register b

If the index designator = 0, then M = m.

ADDRESS MODIFICATION MODES

Three possible modes of address modification, as determined by the index designator, are:

1) b = 0 No address modification. The execution address is directly interpreted.

2) b = 1 - 6 Relative address modification. The execution address is modified as outlined previously. One's complement arithmetic is used in determining the modified execution address.

NOTE

Since one's complement arithmetic is used in determining the relative address, address 77776_8 is the highest address which can be generated. Address 77777_8 cannot be reached by indexing. For example, modifying execution address 77776 by adding, in one's complement arithmetic, an index value of 1, results in 00000. During address modification, the modified address will equal 77777_8 only if: (1) the unmodified execution address equals 77777_8 and b = 0, or (2) the unmodified execution address equals 77777_8 and $(B^b) = 77777_8$.

3) b = 7 Indirect addressing. A storage reference is made to the location specified by the execution address. The lower order 18 bits of the word at this storage location are interpreted as the index designator 'b' (3 bits) and execution address (15 bits) of the present instruction. The new index designator may refer to any one of the three modes.

EXECUTION OF INSTRUCTIONS

A program example and a step-by-step explanation of its execution is outlined below. Instructions from Class I are used in the example to help explain the use of various designators.

The storage reference is initiated at address 00300 (the address held in the P register). The 48-bit word is read from address 00300 and entered into U. Computer operation is now dependent upon the interpretation of the 24-bit instruction in the upper half of U.

The operation code, LDA, and the index designator, 0, are translated. The function of the LDA instruction is to load the A register with the contents of the designated storage location. Because the index designator is 0, the execution address is not modified. The translation of the operation code initiates the sequence of commands which executes the instruction, and the operand in address 00200 is loaded into A.

The lower instruction in U is translated. The ADD instruction causes the quantity in storage location M to be added to the contents of the A register. Since the index designator is not 0 or 7, the contents of the index register are added to the execution address to form M ($M = m + (B^b) = 00210_8 + 00100_8 = 00310_8$). The contents of storage address 00310 are added to the contents of the A register, completing the instruction. The contents of the P register are increased by one and the next program step at address 00301 is read from storage, entered into U, and translated.

Example:

Storage Address	Contents of Address					
	f	b	m	f	b	m
00300	LDA	0	00200	ADD	1	00210
.						
.						
.						
. $(B^1) = 00100_8$						

SYMBOLS

The following symbols are used in the Order of Instructions section.

A	The A register
A_n	The binary digit in position 'n' of the A register
B^b	Designated index register
LA	Lower address; execution address portion of lower instruction of a program step
Q	Auxiliary Arithmetic register
UA	Upper address
( )	Contents of a register or storage location
( )'	One's complement contents of a register or storage location
( )f	Final contents of a register or storage location
( )i	Initial contents of a register or storage location
#	A flag to denote the instruction must be located in the upper instruction position of an instruction word
v	The logical inclusive OR function
⊻	The logical exclusive OR function
^	The logical AND function

ORDER OF INSTRUCTIONS IN BASIC COMPUTER

Octal Code	Mnemonic Code	Name	Indirect Addressing	Storage * References	Address Modification	Number of Instruction Bits
	Inter-Register Transmission					
	IAQ	Interchange A and Q (00700554)	No	0	No	24
	Full-Word Transmission					
12	LDA	Load A	Yes	1	Yes	24
16	LDQ	Load Q	Yes	1	Yes	24
20	STA	Store A	Yes	1	Yes	24
21	STQ	Store Q	Yes	1	Yes	24
13	LAC	Load A, Complement	Yes	1	Yes	24
17	LQC	Load Q, Complement	Yes	1	Yes	24

* If indirect addressing is designated, at least one additional storage reference is required.

ORDER OF INSTRUCTIONS (Cont'd)

Octal Code	Mnemonic Code	Name	Indirect Addressing	Storage References	Address Modification	Number of Instruction Bits
	Address Transmission					
61	SAL	Substitute Address (lower)	Yes	1	Yes	24
60	SAU	Substitute Address (upper)	Yes	1	Yes	24
04	ENQ	Enter Q	Yes	0	Yes	24
10	ENA	Enter A	Yes	0	Yes	24
53	LIL	Load Index (lower)	Yes	1	No	24
52	LIU	Load Index (upper)	Yes	1	No	24
57	SIL	Store Index (lower)	Yes	1	No	24
56	SIU	Store Index (upper)	Yes	1	No	24
50	ENI	Enter Index	Yes	0	No	24
	ATI	Transmit A to Index (0074054b)	No	0	No	24
	Fixed Point Arithmetic					
14	ADD	Add	Yes	1	Yes	24
15	SUB	Subtract	Yes	1	Yes	24
24	MUI	Multiply Integer	Yes	1	Yes	24
25	DVI	Divide Integer	Yes	1	Yes	24
	Address Arithmetic					
11	INA	Increase A	Yes	0	Yes	24
51	INI	Increase Index	Yes	0	No	24
54	ISK	Index Skip	Yes	0	No	24
	Logical					
40	SST	Selective Set	Yes	1	Yes	24
41	SCL	Selective Clear	Yes	1	Yes	24
42	SCM	Selective Complement	Yes	1	Yes	24
43	SSU	Selective Substitute	Yes	1	Yes	24
44	LDL	Load Logical	Yes	1	Yes	24
45	ADL	Add Logical	Yes	1	Yes	24
46	SBL	Subtract Logical	Yes	1	Yes	24
47	STL	Store Logical	Yes	1	Yes	24

ORDER OF INSTRUCTIONS (Cont'd)

Octal Code	Mnemonic Code	Name	Indirect Addressing	Storage References	Address Modification	Number of Instruction Bits
	Shifting					
01	ARS	A Right Shift	Yes	0	Yes	24
02	QRS	Q Right Shift	Yes	0	Yes	24
03	LRS	Long Right Shift (AQ)	Yes	0	Yes	24
05	ALS	A Left Shift	Yes	0	Yes	24
06	QLS	Q Left Shift	Yes	0	Yes	24
07	LLS	Long Left Shift (AQ)	Yes	0	Yes	24
34	SCA	Scale A	Yes	0	No	24
35	SCQ	Scale AQ	Yes	0	No	24
	Replace					
70	RAD	Replace Add	Yes	2	Yes	24
71	RSB	Replace Subtract	Yes	2	Yes	24
72	RAO	Replace Add One	Yes	2	Yes	24
73	RSO	Replace Subtract One	Yes	2	Yes	24
	Storage Test					
36	SSK	Storage Skip	Yes	1	Yes	24
37	SSH	Storage Shift	Yes	2	Yes	24
	Search					
64	EQS	Equality Search	Yes		Yes	24
65	THS	Threshold Search	Yes		Yes	24
66	MEQ	Masked Equality Search	Yes		Yes	24
67	MTH	Masked Threshold Search	Yes		Yes	24
	Jumps and Stops					
22	AJP	A Jump	No	1*	No	24
23	QJP	Q Jump	No	1*	No	24
55	IJP	Index Jump	Yes	0	No	24
75	SLJ	Selective Jump	No	1*	No	24
76	SLS	Selective Stop	No	1*	No	24

* Return jump only

ORDER OF INSTRUCTIONS (Cont'd)

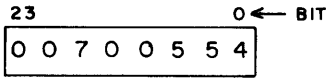
Octal Code	Mnemonic Code	Name	Indirect Addressing	Storage References	Address Modification	Number of Instruction Bits
	Input/Output					
74.0	CONN	Connect	No	0	No	48
74.1	EXTF	Function	No	0	No	48
74.2	BEGR	Read	No	0	No	48
74.3	BEGW	Write	No	0	No	48
74.4	COPY	Copy Status	No	0	No	48
74.5	CLCH	Clear Channel	No	0	No	48
74.6	CCWD	Change Control Word	No	0	No	48
77.2	CIS	Copy Interrupt Status	No		No	24
77.3	SEN	Internal Sense	No	0	No	24
77.4	CPR	Copy Product Register	No	0	No	24
77.0	INF	Internal Function	No	0	No	24
	Miscellaneous					
77.1	AUG	Augment	Yes	0		24
77.5	EUB	Enter Upper Bound	No	0	No	24
77.6	ELB	Enter Lower Bound	No	0	No	24
	LDC	Load Character (63bv 0006500m)	Yes	1	Yes	48
	STC	Store Character (63bv 0006505m)	Yes	2	Yes	48
	Illegal Codes					
26						
27	Return Jump to Address 00020					
74.7						
77.7						

INSTRUCTIONS ADDED BY FLOATING POINT OPTION

Octal Code	Mnemonic Code	Name	Indirect Addressing	Storage References	Address Modification	Number of Instruction Bits
	Single Precision Floating Point Arithmetic					
30	FAD	Floating Add	Yes	1	Yes	24
31	FSB	Floating Subtract	Yes	1	Yes	24
32	FMU	Floating Multiply	Yes	1	Yes	24
33	FDV	Floating Divide	Yes	1	Yes	24

INTER-REGISTER TRANSMISSION

Interchange A and Q (IAQ)



The 24-bit Interchange A and Q instruction interchanges the contents of the A and Q registers.

FULL-WORD TRANSMISSION

In Full-Word Transmission instructions, a 48-bit operand or data word is used in executing the instruction.

LDA Load A Op. Code 12

Replaces the contents of A with a 48-bit operand contained in the storage location specified by M. Negative zero is formed in A if the operand at M is equal to negative zero.

L AC Load A Complement Op. Code 13

Replaces the contents of A with the complement of a 48-bit operand contained in the storage location specified by M. Negative zero is formed in A if the operand at M is equal to positive zero.

LDQ Load Q Op. Code 16

Replaces the contents of Q with a 48-bit operand contained in the storage location specified by M. Negative zero is formed in Q if the operand at M is equal to negative zero.

LDQ Load Q Complement Op. Code 17

Replaces the contents of Q with the complement of a 48-bit operand contained in the storage location specified by M. Negative zero is formed in Q if the operand at M is equal to positive zero.

STA Store A Op. Code 20

Replaces the contents of the designated storage location, M, with the contents of A.

STQ Store Q Op. Code 21

Replaces the contents of the designated storage location, M, with the contents of Q.

ADDRESS TRANSMISSION

1) In the Address Transmission instructions, only the lower 15 bits of a 24-bit word instruction or data word are used.

2) In the LIU and LIL instructions, an index designation of "0" has no meaning and should not be used. If used, these instructions become pass instructions, but use some time in storage reference. The next instruction is then executed. Using "0" as an index designation does not constitute a fault.

LIU Load Index Upper Op. Code 52

Replaces the contents of the designated index register with the upper address portion of storage location 'm'. If b = 0 this instruction becomes a pass (do-nothing) instruction.

LIL Load Index Lower Op. Code 53

Replaces the contents of the designated index register with the lower address portion of storage location 'm'. If b = 0 this instruction becomes a pass (do-nothing) instruction.

SIU Store Index Upper Op. Code 56

Replaces the upper address portion of storage location 'm' with the contents of the designated index register. The remaining bits of the word in storage remain unchanged. If b = 0, (m_{ua}) is cleared.

SIL Store Index Lower Op. Code 57

Replaces the lower address portion of storage location 'm' with the contents of the designated index register. The remaining bits of the word in storage remain unchanged. If b = 0, (m_{la}) is cleared.

SAU Substitute Address Upper Op. Code 60

Replaces the upper portion of M with the lower order 15 bits of A. Remaining bits of M are not modified and the initial contents of A are unchanged.

SAL Substitute Address Lower Op. Code 61

Replaces the lower portion of M with the lower order 15 bits of A. Remaining bits of M are not modified and the initial contents of A are unchanged.

ENQ Enter Q Op. Code 04

The 15-bit operand, Y, is entered into Q and its highest order bit (sign bit) is extended in the re-

maining 33 bits. The largest positive 15-bit operand that can be entered into Q is 37777_8 (2^{14} -1) and its "0" sign bit will be duplicated in each of the remaining 33 bits of Q. Negative zero will be formed in Q if:

1) $(B^b) = 77777_8$ and $y = 77777_8$ or

2) $b = 0$ and $y = 77777_8$.

ENA Enter A Op. Code 10

The 15-bit operand, Y, is entered into the A register and its highest order bit (sign bit) is extended in the remaining 33 bits. The largest positive 15-bit operand that can be entered into A is 37777_8 (2^{14} -1) and the "0" sign bit will be duplicated in each of the remaining 33 bits. Negative zero will be formed in A if:

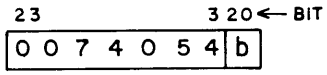
1) $(B^b) = 77777_8$ and $y = 77777_8$ or

2) $b = 0$ and $y = 77777_8$.

ENI Enter Index Op. Code 50

Replaces (B^b) with the operand y. If b = 0, this instruction becomes a pass (do-nothing) instruction.

ATI Transmit A to Index



Transmits the lower 15 bits of A to the index register specified by b. The quantity designated by b may have values 1--6; (b = 0, 7 is undefined and should not be used). The contents of the A register are not modified.

FIXED POINT ARITHMETIC

1) If the capacity of the A register, $\pm(2^{47}$ -1), is exceeded during the execution of the Fixed Point Arithmetic instructions (ADD, SUB, RAD, RSB, RAO, and RSO), an arithmetic overflow fault is produced. When executing the DVI instruction, if the result exceeds the capacity of the Q register, $\pm(2^{47}$ -1), a divide fault is produced (refer to appendix A).

2) The Multiply Integer instruction (MUI) uses the double register configuration QA. The least significant bit of the product is left in bit position A_{00}. The most significant bit may be in either A or Q, depending upon the magnitude of the product.

ADD Add Op. Code 14

Adds a 48-bit operand obtained from storage location M to contents of A. A negative zero may be produced by this instruction if (A) and (M) are initially negative zero.

SUB Subtract Op. Code 15

Obtains a 48-bit operand from storage location M and subtracts it from the initial contents of A. A negative zero will be produced if the initial contents of A are negative zero and that of storage location M are positive zero.

MUI Multiply Integer Op. Code 24

Forms a 96-bit product from two 48-bit operands. The multiplier must be loaded into A prior to execution of the instruction. The execution address specifies the storage location of the multiplicand. The product is contained in QA as a 96-bit quantity. The operands are considered as integers and the binary point is assumed to be at the lower order (right-hand) end of the A register.

DVI Divide Integer Op. Code 25

Divides a 96-bit integer dividend by a 48-bit integer divisor. The 96-bit dividend must be formed in the QA register prior to executing the instruction. If a 48-bit dividend is loaded into A, the sign of Q must be set (the sign of the dividend in A must be extended throughout Q). The 48-bit divisor is read from the storage location specified by the execution address. The quotient is formed in A and the remainder is left in Q at the end of the operation. Dividend and remainder have the same sign.

SINGLE PRECISION FLOATING POINT ARITHMETIC (Floating Point Option)

Floating Point range faults (overflow/underflow) occur if the exponent exceeds $\pm(2^{10}-1)$.

FAD Floating Add (FPO) Op. Code 30

Forms the sum of two operands packed in floating point format. A floating point operand is read from storage location M and added to the floating point word in A. The result is normalized, rounded, and retained in A at the end of the operation. Q contains only the residue of the rounding operation at the end of the sequence.

FSB Floating Subtract (FPO) Op. Code 31

Forms the difference of two 48-bit operands in floating point format. The subtrahend is acquired from storage address M and is subtracted from the minuend in A. The result is rounded and normalized if necessary and retained in A. The residue from the rounding operation is left in Q at the end of the sequence.

FMU Floating Multiply (FPO) Op. Code 32

Forms the product of an operand in floating point format with the previous contents of A also in floating point format. The operand is read from storage location M. The product is rounded and normalized if necessary and retained in A. The residue from the rounding operation is left in Q at the end of the sequence.

FDV Floating Divide (FPO) Op. Code 33

Forms the quotient of two 48-bit operands in floating point format. The dividend must be loaded into A prior to executing this instruction. The divisor is read from the storage location specified by M. The quotient is rounded and normalized if necessary and retained in A at the end of the operation. The residue from the rounding operation is left in Q at the end of the operation.

ADDRESS ARITHMETIC

In the Address Arithmetic instructions, only the lower 15 bits of the operand or data words are used.

INA Increase A Op. Code 11

Adds Y to A. The 15-bit operand, Y, with its highest order bit (sign bit) extended, is added to A.

INI Increase Index Op. Code 51

Increases (B^b) by the operand 'y'. If the b designator is zero, this instruction becomes a pass or do nothing instruction.

ISK Index Skip Op. Code 54

Compares (B^b) with 'y'. If the two quantities are equal, B^b is cleared and the lower instruction is skipped. If the quantities are unequal, (B^b) is increased by one. Counting in this instruction is performed in one's complement notation. (If $(B^b)_i = 777776_8$ and $y \neq 777776_8$, then $(B^b)_f = 00000$.) If b = 0, the (B^b) are taken to be zero. ISK is usually restricted to the upper instruction. If used as a lower instruction, no skip is possible.

LOGICAL

1) The LDL, ADL, SBL and STL instructions achieve their result by forming a logical product. A logical product is a bit-by-bit multiplication of two binary numbers:

$0 \times 0 = 0$ $\quad$ $1 \times 0 = 0$

$0 \times 1 = 0$ $\quad$ $1 \times 1 = 0$

2) A logical product is used, in many cases, to select specific portions of an operand for entry into another operation. For example, if only a specific portion of an operand in storage is to be added to (A), the operand is subjected to a mask composed of a predetermined pattern of "0's" and "1's". Forming the logical product of the operand and the mask causes the operand to retain its original contents only in those stages which have corresponding "1's" in the mask. When only the selected bits remain, the instruction proceeds to conclusion.

SST Selective Set Op. Code 40

Sets the individual bits of A to "1" where there are corresponding "1's" in the word at storage location M; "0" bits in the storage word do not modify the corresponding bits in A. In a bit-by-bit comparison of (A) and (M), four possible combinations of bits are possible.

1)	2)	3)	4)
$(A)_i = 1$	$(A)_i = 1$	$(A)_i = 0$	$(A)_i = 0$
$(M)_i = 1$	$(M)_i = 0$	$(M)_i = 1$	$(M)_i = 0$
$(A)_f = 1$	$(A)_f = 1$	$(A)_f = 1$	$(A)_f = 0$
$(M)_f = 1$	$(M)_f = 0$	$(M)_f = 1$	$(M)_f = 0$

SCM Selective Complement Op. Code 42

Individual bits of A are complemented where there are corresponding "1's" in the word at storage location M. If the corresponding bits at M are "0's", the associated bits of A remain unchanged.

1)	2)	3)	4)
$(A)_i = 1$	$(A)_i = 1$	$(A)_i = 0$	$(A)_i = 0$
$(M)_i = 1$	$(M)_i = 0$	$(M)_i = 1$	$(M)_i = 0$
$(A)_f = 0$	$(A)_f = 1$	$(A)_f = 1$	$(A)_f = 0$
$(M)_f = 1$	$(M)_f = 0$	$(M)_f = 1$	$(M)_f = 0$

SCL Selective Clear Op. Code 41

Clears individual bits of A where there are corresponding "1's" If the corresponding bits at M are "0's" the associated bits of A remain unchanged.

In a bit-by-bit comparison of (A) and (M), four possible combinations of bits are possible.

1)	2)	3)	4)
$(A)_i = 1$	$(A)_i = 1$	$(A)_i = 0$	$(A)_i = 0$
$(M)_i = 1$	$(M)_i = 0$	$(M)_i = 1$	$(M)_i = 0$
$(A)_f = 0$	$(A)_f = 1$	$(A)_f = 0$	$(A)_f = 0$
$(M)_f = 1$	$(M)_f = 0$	$(M)_f = 1$	$(M)_f = 0$

SSU Selective Substitute Op. Code 43

Substitutes selected portions of an operand at storage address M into the A register where there are corresponding "1's" in the Q register (mask). The portions of A not masked by "1's" in Q are left unmodified.

LDL Load Logical Op. Code 44

Loads A with the logical product of Q and the designated storage location, M. The operand can be in either Q or M.

ADL Add Logical Op. Code 45

Adds to A the logical product of Q and the quantity in location M; the mask may be in Q or storage. Once the logical product is formed, addition follows normal rules.

SBL Subtract Logical Op. Code 46

Subtracts from A the logical product of the Q register and the quantity in storage location M. The mask may be in Q or storage. When the logical product is formed, the subtraction proceeds in the normal manner.

STL Store Logical Op. Code 47

Replaces the bits in location M with the logical product of Q and A registers. Neither (A) nor (Q) is modified. The mask may be located in A or Q.

SHIFTING

1) The largest practical shift count for a 48-bit register is 48_{10}, for a 96-bit register, 96_{10}. If a shift greater than $177_8 = 127_{10}$ is attempted, the Shift Fault indicator will be set and the operand will be shifted according to the value of the lower 7 bits of K.

2) Shifts are not constant speed; that is, performing a shift of 46_{10} places, for example, takes longer than a shift of 1 place.

ARS A Right Shift Op. Code 01

Shifts contents of A to the right K places. The sign is extended and the lower bits are discarded. The largest practical shift count is 47_{10} since the register is now an extension of the sign bit.

QRS Q Right Shift Op. Code 02

Shifts contents of Q to the right K places. The sign is extended and the lower bits are discarded. The largest practical shift count is 47_{10} since the register is now an extension of the sign bit.

LRS Long Right Shift Op. Code 03

Shifts contents of AQ to the right K places as one 96-bit register. The A register is considered as the leftmost 48 bits and the Q register as the rightmost 48 bits. The sign of A is extended. The lower order bits of A replace the higher order bits of Q and the lower order bits of Q are discarded. The largest practical shift count is 95_{10} since AQ is now an extension of the sign of A.

ALS A Left Shift Op. Code 05

Shifts contents of A to the left K places, left circular. The higher order bits of A replace the lower order bits. The largest practical shift count, 48_{10}, returns the register to its original state.

QLS Q Left Shift Op. Code 06

Shifts contents of Q to the left K places, left circular. The higher order bits of Q replace the lower order bits. The largest practical shift count, 48_{10}, returns the register to its original state.

LLS Long Left Shift Op. Code 07

Shifts contents of AQ to the left K places, left circular, as one 96-bit register. The higher order bits of A replace the lower order bits of Q and the higher order bits of Q replace the lower order bits of A. The largest practical shift count, 96_{10}, returns AQ to its original state.

SCALE

1) Address modification does not apply. The index register is used to preserve the scale factor.

2) If b = 0, scaling is executed but the scale factor is lost.

3) If b = 7, indirect addressing is used and at least one storage reference is made.

4) If (A) or (AQ)i is already scaled or equal to positive or negative zero, K→B^b, and scaling is not executed.

5) If the lower 7 bits of the shift count are initially equal to 0, k→B^b, and scaling is not executed.

6) The Shift Fault indicator is not affected by this instruction.

SCA Scale A Op. Code 34

Shifts A left circularly until the most significant digit is to the right of the sign bit or until the lower 7 bits of k = 0. (Maximum shift = 177_8, although k may contain up to and including 77777_8.) The shift count (lower 7 bits of k) is reduced by one after each shift. The shift operation terminates when the lower 7 bits of k = 0 or the most significant bit is to the right of the sign bit. Upon termination, bits 00 -- 14 of k are entered in the designated index register.

SCQ Scale AQ Op. Code 35

Shifts AQ left circularly until the most significant digit is to the right of the sign bit or until the lower 7 bits of k = 0. (Maximum shift = 177_8, although k may contain up to and including 77777_8.) The shift count (lower 7 bits of k) is reduced by one after each shift. The shift operation terminates when the lower 7 bits of k = 0 or the most significant bit is to the right of the sign bit. Upon termination, bits 00 -- 14 of k are entered in the designated index register.

REPLACE

1) If the capacity of the A register, $\pm(2^{47}-1)$, is exceeded during the execution of the Replace instructions, an arithmetic overflow fault is produced.

RAD Replace Add Op. Code 70

Obtains a 48-bit operand from storage location M and adds it to the initial contents of A. The sum is left in A and is also transmitted to location M.

RSB Replace Subtract Op. Code 71

Subtracts A from M and places the result in both the A register and location M.

RAO Replace Add One Op. Code 72

Replaces the operand in storage location M with its original value plus one. The result is also placed in A.

RSO Replace Subtract One Op. Code 73

Replaces the operand in storage location M with its original contents minus one. The difference is also left in A; the original contents of A and M are destroyed.

STORAGE TEST

SSK Storage Skip Op. Code 36

Senses the sign bit of the operand in storage location M. If the sign is negative, the lower instruction is skipped. The contents of the operational registers are left unmodified. SSK is usually restricted to an upper instruction. If used as a lower instruction, no skip is possible.

SSH Storage Shift Op. Code 37

Senses the sign bit of the quantity in storage location M. If the sign bit is negative, the lower instruction is skipped. In either case the quantity is shifted left circular one bit. This instruction is usually restricted to the upper position. If used as a lower instruction no skip is possible. The contents of the operational registers are left unmodified.

STORAGE SEARCH

1) If b = 0 in the following instructions, only the word at storage location 'm' will be searched.

2) If b = 7, indirect addressing is used to obtain the execution address and 'b' designator.

3) If (B^b) = 0, no search is made.

4) The operands searched by these instructions may be in either fixed or floating point format.

5) Normally this is an upper instruction. If used as a lower instruction, no skip is possible.

EQS Equality Search Op. Code 64

Searches a list of operands to find one that is equal to A. The number of items to be searched is specified by B^b. These items are in sequential addresses

beginning at the location specified by 'm'. The search begins with the last address, $m + (B^b - 1)$. B^b is reduced one count for each word that is searched until an operand is found that equals A or until B^b equals zero. If the search is terminated by finding an operand that equals A, the lower instruction is skipped. The address of the operand satisfying this condition is given by the sum of 'm' and the final contents of B^b. Positive zero and minus zero are recognized as the same quantity. When EQS is used as a lower instruction, no skip is possible when $(B^b) = 0$, or when the condition is met.

THS Threshold Search Op. Code 65

Searches a list of operands to find one that is greater than A. The number of items to be searched is specified by B^b. These items are located in sequential addresses beginning at the location specified by 'm'. The search begins with the last address, $m + (B^b - 1)$. The contents of the index register are reduced by one for each operand examined. The search continues until an operand is reached that is greater than A or until B^b is reduced to zero. If the search is terminated by finding an operand greater than the value in A, the lower instruction is skipped. The address of the operand satisfying the condition is given by the sum of 'm' and the final contents of B^b. If no operand in the list is greater than the value in A, no skip is possible. If THS is used as a lower instruction, the next instruction will be executed when search terminates. In the comparison made here, positive zero is considered as greater than minus zero.

MEQ Masked Equality Search Op. Code 66

Searches a list of operands to find one such that the logical product of (Q) and (M) is equal to (A). This instruction, except for the mask in Q, operates in the same manner as an equality search.

MTH Masked Threshold Search Op. Code 67

Searches a list of operands to find one such that the logical product of (Q) and (M) is greater than (A). Except for the mask in Q, this instruction operates in the same manner as the threshold search.

JUMPS AND STOPS

Normal Jump

A jump instruction causes a current program sequence to terminate and initiates a new sequence at a different location in storage. The Program Address register, P, provides the continuity between program steps and always contains the storage location of the current program step.

When a jump instruction occurs, P is cleared and a new address is entered. In all jump instructions, the execution address 'm' specifies the beginning address of the new program sequence. The word at address 'm' is read from storage, placed in U and the upper instruction (first instruction of the new sequence) is executed.

Some of the jump instructions are conditional upon a register containing a specific value or upon the position of a Jump or Stop switch on the console. If the criterion is satisfied, the jump is made to location 'm'. If it is not satisfied, the program proceeds in regular sequence to the next instruction.

A jump instruction may appear in either position in a program step. If the jump instruction appears in the first (upper) part of the program step and the jump is taken, the second (lower) part of the program step is never executed. If the instruction appears in the lower part, the upper part is executed in the normal manner.

AJP A Jump Op. Code 22

Jumps to 'm' if the conditions of the A register specified by the jump designator 'j' exist. If not, the next instruction is executed.

j = 0	Jump if (A) = 0
j = 1	Jump if (A) ≠ 0
j = 2	Jump if (A) = +
j = 3	Jump if (A) = −

When (A) is negative zero the interpretation is:

j = 0	The jump is executed because, in this case, negative zero is recognized as positive zero.
j = 1	The jump is not executed when (A) = + 0 or − 0.
j = 2	The jump is not executed because the sign bit is a "1".
j = 3	The jump is executed because the sign bit is a "1".

For conditions specified by '1' when its value is 4 − 7, refer to the Return Jump section.

QJP Q Jump Op. Code 23

Jumps to 'm' if the condition of the Q register specified by the jump designator 'j' exists. If not, the

next instruction is executed.

j = 0	Jump if (Q) = 0
j = 1	Jump if (Q) ≠ 0
j = 2	Jump if (Q) = +
j = 3	Jump if (Q) = −

When (Q) is negative zero the AJP interpretation applies.

IJP Index Jump Op. Code 55

Examines (B^b). If this quantity is not zero, the quantity is reduced one count and a jump is executed to program step 'm'. The index jump can be used in the upper or lower instruction without reservation; it executes a normal jump upon satisfaction of the jump condition. If b = 0, execution continues with the next program step.

SLJ Selective Jump Op. Code 75

Jumps to 'm' if the condition of the Jump switches specified by 'j' exists. If not, the next instruction is executed.

SLJ	j = 0	Jump unconditionally (does not reference Jump switch setting).
SJ1	j = 1	Jump if Jump switch 1 is set
SJ2	j = 2	Jump if Jump switch 2 is set
SJ3	j = 3	Jump if Jump switch 3 is set

SLS Selective Stop Op. Code 76

Stops at present step in the sequence if the condition of the Stop switch specified by 'j' exists. (Stop switches are located on the maintenance panel of the computer. These are only active when the computer is in maintenance mode.) If the stop condition exists, the stop is executed, and the jump is executed unconditionally when the Go switch is pressed. If the stop condition is not satisfied, the jump is executed unconditionally.

SLS	j = 0	Stop unconditionally (does not reference Stop switch setting).
SS1	j = 1	Stop if Stop switch 1 is set
SS2	j = 2	Stop if Stop switch 2 is set
SS3	j = 3	Stop if Stop switch 3 is set

Return Jump

A return jump begins a new program sequence at the lower instruction portion of the program step to which the jump is made. At the same time, the execution address of the upper instruction of that program step is replaced with the address of the next program step in the main program. This instruction is usually an Unconditional Jump instruction and allows a return to the main program after completing the subprogram sequence (figure 3-1).

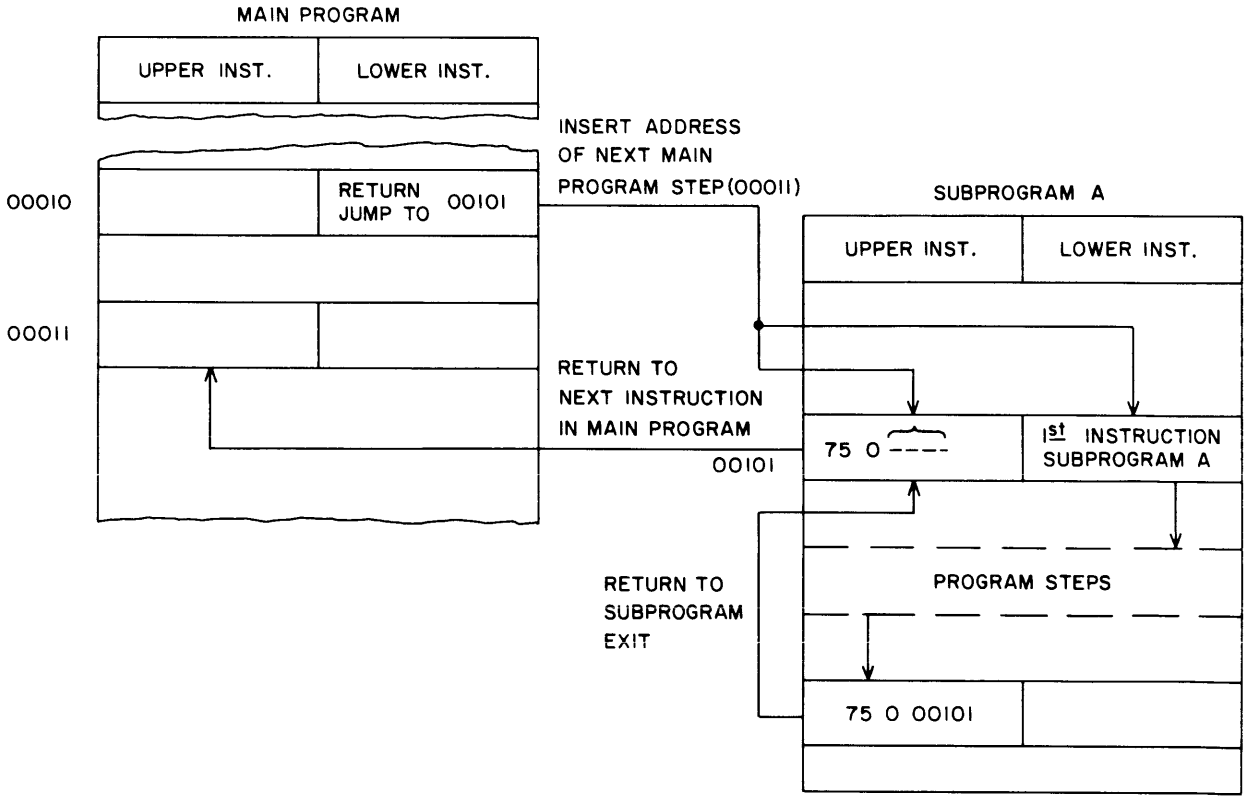


Figure 3-1. Return Jump

ARJ A Jump Op. Code 22

Executes a return jump to storage location 'm' if the condition of the A register specified by 'j' exists. If not, the next instruction is executed.

j = 4 Return jump if (A) = 0

j = 5 Return jump if (A) ≠ 0

j = 6 Return jump if (A) = +

j = 7 Return jump if (A) = –

For conditions specified by 'j' when its value is 0 – 3, refer to the Normal Jump section.

Note: If (A) = negative zero, refer to the AJP instruction in the Normal Jump section.

QRJ Q Jump Op. Code 23

Executes a return jump to storage location 'm' if the condition of the Q register specified by 'j' exists. If not, the next instruction is executed.

j = 4 Return jump if (Q) = 0

j = 5 Return jump if (Q) ≠ 0

j = 6 Return jump if (Q) = +

j = 7 Return jump if (Q) = –

Note: If (Q) = negative zero, refer to the AJP instruction.

SLJ Selective Jump Op. Code 75

Executes a return jump to storage location 'm' on condition 'j' where condition 'j' represents the setting of the Jump switches. If the condition is not satisfied, the next instruction is executed.

RTJ j = 4 Return jump unconditionally (does not reference Jump switches).

RJ1 j = 5 Return jump if Jump switch 1 is set

RJ2 j = 6 Return jump if Jump switch 2 is set

RJ3 j = 7 Return jump if Jump switch 3 is set

Note: The Jump switch is illuminated when it is in the Set position.

SLS Selective Stop Op. Code 76

Stops on condition 'j' and executes a return jump when the Go switch is pressed. If the stop condition is not satisfied, the stop is not executed and the return jump is executed unconditionally.

SRJ j = 4 Stop unconditionally; return jump on Go (does not reference Stop switches).

SR1 j = 5 Stop if Stop switch 1 is set; return jump on Go.

SR2 j = 6 Stop if Stop switch 2 is set; return jump on Go.

SR3 j = 7 Stop if Stop switch 3 is set; return jump on Go.

Note: The Stop switch is illuminated when it is in the Set position.

CHARACTER HANDLING

These instructions transfer 6 bits of information to/from a 6-bit portion of storage location 'm' and the lower six bits of the A register. On a load operation, the contents of 'm' remain unaltered. On a store operation, only the 6-bit portion of 'm' specified by the contents of v is altered.

LDC Load Character Op. Code 63bv0006500m

Transmit the 6-bit portion of the contents of M specified by the contents of index register v to the lower 6 bits of A. The remainder of A is cleared. (M) is not altered.

Contents of index register v equals:

00042, bits 47-42 of (M)
00036, bits 41-36 of (M)
00030, bits 35-30 of (M)
00024, bits 29-24 of (M)
00018, bits 23-18 of (M)
00012, bits 17-12 of (M)
00006, bits 11-06 of (M)
00000, bits 05-00 of (M)

Index register b is used for normal address modification. If b = 0, m = M, if b = 7, indirect addressing applies.

STC Store Character Op. Code 63bv0006505m

Store the lower 6-bit portion of the contents of A in the 6-bit portion of address M specified by the contents of index register v. A remains unchanged. The portion of (M) not affected by the byte is left in its original state.

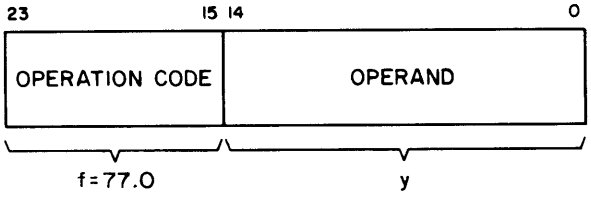
Contents of index register v equals:

00042, bits 47-42 of M
00036, bits 41-36 of M
00030, bits 35-30 of M
00024, bits 29-24 of M
00018, bits 23-18 of M
00012, bits 17-12 of M
00006, bits 11-06 of M
00000, bits 05-00 of M

Index register b is used for normal address modification. If b = 0, m = M; if b = 7, indirect addressing applies.

INTERNAL FUNCTION (*INF*)

The Internal Function instruction establishes the internal operating mode or executes the operation specified by the function code 'y'. These codes (values) for 'y' and their designated functions, are tabulated below.



Code	Function	Comments
	Select Interrupt On Channel Becoming Inactive:	Set the designated bit in the Interrupt Mask register. Clear the designated Interrupt register bit. Interrupt occurs when the desired channel becomes inactive (i. e. when the transition from active to inactive occurs) and the corresponding bit in the Interrupt Mask register is set.
00000	Channel 0	
00010	Channel 1	
00020	Channel 2	
00030	Channel 3	
	Clear Select Of Interrupt On Channel Becoming Inactive:	Clear the designated bit in the Interrupt Mask register. Clear the designated Interrupt Register bit.
00001	Channel 0	
00011	Channel 1	
00021	Channel 2	
00031	Channel 3	
	Set Interrupt Exit (lower instruction)	Sets the Interrupt Exit FF, enabling a return to the lower instruction of the main program step being executed when interrupt occurred.
00110	Category I	
00120	Category II	
	Clear Interrupt Exit (upper instruction)	Clears the Interrupt Exit FF enabling a return to the upper instruction of the main program step being executed when the interrupt occurred.
00111	Category I	
00121	Category II	
	Enable Interrupt From:	Set/Clear the designated bit in the Interrupt Mask register.
00200	Channel 0	
00210	Channel 1	
00220	Channel 2	
00230	Channel 3	
	Disable Interrupt From:	
00201	Channel 0	
00211	Channel 1	
00221	Channel 2	
00231	Channel 3	

Code	Function	Comments
	Select I/O Parity Error Interrupt:	
00300	Channel 0	
00310	Channel 1	
00320	Channel 2	
00330	Channel 3	Set/Clear the designated bit in the Interrupt Mask register.
	Clear Select of I/O Parity Error Interrupt	
00301	Channel 0	
00311	Channel 1	
00321	Channel 2	
00331	Channel 3	
00400	Clear Arithmetic Fault	Clears any arithmetic fault condition(s) existing in the Interrupt register. (Shift fault, Divide fault, Exponent Overflow fault, Exponent Underflow fault, Arithmetic Overflow fault.)
00410	Select Shift Fault Interrupt	
00411	Clear Select of Shift Fault Interrupt	
00420	Select Divide Fault Interrupt	
00421	Clear Select of Divide Fault Interrupt	
00430	Select Arithmetic Overflow Interrupt	Set/Clear the designated bit in the Interrupt Mask register.
00431	Clear Select of Arithmetic Overflow Interrupt	
00440	Select Exponent Overflow Interrupt	
00441	Clear Select of Exponent Overflow Interrupt	
00450	Select Exponent Underflow Interrupt	
00451	Clear Select of Exponent Underflow Interrupt	
00412	Clear Shift Fault	
00422	Clear Divide Fault	
00432	Clear Arithmetic Overflow Fault	Clear the designated bit in the Interrupt register.
00442	Clear Exponent Overflow Fault	
00452	Clear Exponent Underflow Fault	
00500	Select Time Interrupt	Set/Clear the designated bit in the Interrupt Mask register. Interrupt will occur only once per selection at a time interval corresponding to the power line frequency. For 60 cycle, interrupt will occur every 16 2/3 ms; for 50 cycle, interrupt will occur every 20 ms.
00501	Clear Time Interrupt	

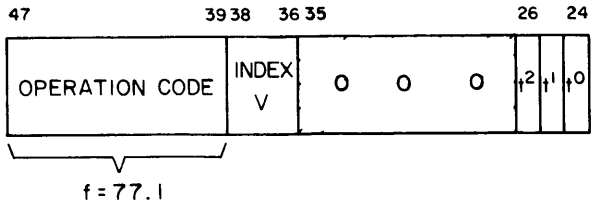
Code	Function	Comments
00510	Select Manual Interrupt	Sets the designated bit in the Interrupt Mask register. Manual interrupt will occur when this interrupt condition is selected and the Manual Interrupt switch is momentarily depressed. Interrupt will not occur if the Manual Interrupt switch is momentarily depressed and Manual Interrupt is selected some time later.
00511	Clear Manual Interrupt	Clears the designated bit in the Interrupt Mask and Interrupt registers.
00520 00521	Select Operand Parity Error Interrupt Clear Select of Operand Parity Error Interrupt	Set/Clear the designated bit in the Interrupt Mask register.
00530 00531	Select Instruction Parity Error Interrupt Clear Select of Instruction Parity Error Interrupt	Set/Clear the designated bit in the Interrupt Mask register.
00522 00532	Clear Operand Parity Error FF Clear Instruction Parity Error FF	
00540 00541	Select Out of Bounds Interrupt Clear Select of Out of Bounds Interrupt	Set/Clear the designated bit in the Interrupt Mask register. (Setting or Clearing the designated bit in the Interrupt Mask registers always clears the corresponding bit in the Interrupt register.)
00600	Set Interrupt Active	Sets the Interrupt Active FF which enables the Interrupt system. Interrupts will now occur if a particular bit in the interrupt Mask register is set when the interrupt condition occurs.
00601	Clear Interrupt Active	Clears the Interrupt Active FF. Regardless of the condition of bits in the Mask register, entrance into the interrupt routine does not occur when the interrupt system is inactive.
00602	Clear Interrupt System	Clears all bits in the Interrupt and Interrupt Mask registers. Also clears the Interrupt Active FF.

INTERNAL SENSE (*SEN*)

This instruction senses internal computer conditions specified by 'y'. These conditions do not include shift fault, arithmetic fault, etc. If this instruction is used in bits 00-23, it becomes a pass code. The program continues at P + 1. The code (values) for 'y' and their designated functions are tabulated below.

Code	Operation	
00000	Skip lower instruction if:	Channel 0 is inactive
00001		Channel 0 is active
00010		Channel 1 is inactive
00011		Channel 1 is active
00020		Channel 2 is inactive
00021		Channel 2 is active
00030		Channel 3 is inactive
00031		Channel 3 is active
00110	Skip lower instruction if:	Category I Interrupt Exit FF is set
00111		Category I Interrupt Exit FF is clear
00120		Category II Interrupt Exit FF is set
00121		Category II Interrupt Exit FF is clear
00210	Skip lower instruction if:	Category I Interrupt Mode FF is set
00211		Category I Interrupt Mode FF is clear
00220		Category II Interrupt Mode FF is set
00221		Category II Interrupt Mode FF is clear

AUGMENT (*AUG*)



The 24-bit Augment instruction may be used to perform one or more of the following operations:

1) Increase the capabilities of certain instructions by specifying additional operations to be performed.

2) Provide additional modification of the address portion of the lower instruction.

When this instruction is used in the lower position of a program step, no meaningful operation is performed. The instruction becomes a pass instruction and the program continues at P + 1.

When this command is used in the upper position of a program step (cases 1 and 2 above) the following operations occur:

1) Operations using index designator 'v' are performed.

2) The augment operation designators 't' are stored to condition the operation of the lower instruction being augmented.

Values for 'v' are assigned as follows:

Value	Operation
v = 0	If v = 0, this designator has no significance in the operation.
v = 1-6	If v = 1-6, address modification applies. The contents of the index register specified by 'v' are added to the address portion (bits 00-14) of the lower instruction to form M, Y, or K, whichever the case.
v = 7	If v = 7, multi-level indirect addressing rules apply. The quantity held in the address portion (bits 00-14) of the lower instruction is treated as a storage address (whether 'm', 'y', or 'k'). The lower 15 bits at this storage address are read from storage and are placed in the address portion of the original lower instruction.

Values for 't' are assigned as follows:

Designator	Value	
	If a "0"	If a "1"
$_t0$	Rounded arithmetic	Unrounded arithmetic
$_t1$	Normalized arithmetic	Un-normalized arithmetic
$_t2$	Use signed operand	Use magnitude of operand* (positive value)

Instructions which may be augmented are listed below. Designators which may be used when augmenting a given instruction are checked opposite that instruction.

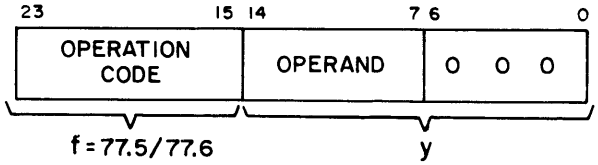
Instruction to be Augmented	$_t2$	$_t1$	$_t0$	v**
LDA	X			
LDQ	X			
STA	X			
STQ	X			
ADD	X			
SUB	X			
MUI	X			
DVI	X			
FAD	X	X	X	
FSB	X	X	X	
FMU	X	X	X	
FDV	X	X	X	

*$_t2$ applies to both fixed and floating point arithmetic.

** May be meaningfully used to augment most 24-bit instructions.

Upon completing the operations specified by the upper (Augment) instruction, the address portion of the lower instruction now contains a modified value (if 'v' specified address modification). When the instruction being augmented (the lower instruction) is executed, its index designator is interpreted in the normal manner. Indirect addressing or address modification is performed on the address modified by the Augment operation.

ENTER UPPER BOUND (*EUB*); ENTER LOWER BOUND (*ELB*)



The 24-bit bounds instructions (Enter Upper Bound = 77.5, Enter Lower Bound = 77.6) load 8-bit quantities into the upper and lower bounds registers. (When a bounds register contains all "1"s, its content is interpreted as 77600_8.)

Bounds checking on jump and memory addresses occurs only if the interrupt system is active and the bounds bit (bit 42) in the Interrupt Mask register is set. (Bit 42 is set with an Internal Function instruction.) Setting the Mask bit or activating interrupt may occur in any order. If the Mask bit is set

and interrupt is active, any one of the following operations will cause a Category I interrupt:

1) A jump to an address outside the bounds defined by the upper and lower bound addresses. (If a jump is attempted out of bounds, it will not be executed and interrupt will occur. Upon return to the main program after processing the interrupt, the next instruction will be executed.)

2) Writing in an address out of bounds.

3) Reading an instruction from out of bounds.

The only permissible use of an address out of bounds (when bounds checking is in effect) is to read its contents as an operand.

Addresses which may be referenced are such that $B_L \leq S$ (upper 8 bits) $< B_U$ (where S is the storage address). Only bits 7 → 14 of S are used in the comparison for bounds checking. When the upper bounds register is set to + 0 (every bit = "0") the upper bound is disregarded.

Note that:

1) There is no MC for the bounds registers. They are program cleared when both registers are set to all 0's by 77.5 and 77.6 instructions.

2) If interrupt is active and bit 42 in the Interrupt Mask register is set (Out of Bounds Interrupt selected), then instructions 77.5 and 77.6 are illegal and cause a category III interrupt. This prevents changing the bounds when checking is in effect.

CHAPTER IV
INTERRUPT SYSTEM

The interrupt system provides for testing whether or not certain conditions (internal or external) exist without having these tests in the main program. (See the three categories of interrupts mentioned later in this chapter.) After executing each main program instruction, these conditions are sampled. If one of the conditions exists, execution of the main program halts. The contents of the Program Address register, P, are stored and an interrupt routine is initiated. This interrupt routine takes the necessary action for the condition and then jumps back to the next unexecuted main program step.

For each condition that can cause an interrupt, the program has two alternatives. It may select an interruptible condition, such that interrupt occurs when that condition occurs or it may choose to have the interrupt system ignore the condition. This is accomplished by not selecting interrupt on the condition. The program also has the choice of whether or not the interrupt system is to be used. The Internal Function instruction activates or deactivates the interrupt system and selects the desired interrupts.

LOGICAL DESCRIPTION OF INTERRUPT SYSTEM

Three registers are directly involved in the interrupt system:

1) a 48-bit Interrupt register

2) a 48-bit Interrupt Mask register

3) a 48-bit Product register

The bits in these registers are numbered from right to left in ascending order. The rightmost bit is numbered zero, and the leftmost bit (in a 48-bit register) is numbered 47.

Each of the 48 bits of the register is associated with a particular internal or external condition. This bit association is identical in all three 48-bit registers. See the Internal Function instruction for codes that set and clear bits in the Interrupt and Interrupt Mask registers. Bits are assigned as shown in Table 4-1.

Brief statements concerning the registers are given below:

Interrupt Register

Each of the internal conditions which can cause an interrupt is wired to a particular bit position of this register. Each bit position associated with external conditions receives an interrupt line from one of the four possible data channels.

Interrupt Mask Register

The programmer selects to have a given interrupt condition (internal/external) tested by setting the appropriate bit of the Interrupt Mask register.

Product Register

This register is the result of the AND of corresponding bits in the Interrupt and Interrupt Mask registers. Bits in the Product register are therefore the logical product of the Interrupt and Interrupt Mask registers. The Product register is fictitious in that it is not composed of flip-flops. Hence there are no instructions for clearing bits in this register. The "bits" in the Product register are "cleared" when one or both corresponding bits in the Interrupt and Interrupt Mask register = 0.

An interrupt results from a "1" in a bit of the Product register. To set to "1" any bit in the Product register requires that the Mask register bit must have been set to "1".

CATEGORIES OF INTERRUPTS

There are three categories of interrupts in this system.

Category I

This category includes all I/O interrupts and interrupts on internal conditions such as overflow and divide fault.

Table 4-1, Assignment of Bits

Bit	Assignment
01	Shift fault
02	Divide fault
03	Arithmetic overflow
04	Exponent overflow
05	Exponent underflow
10	Channel 0 inactive
11	Channel 1 inactive
12	Channel 2 inactive
13	Channel 3 inactive
20	Channel 0 I/O Transmission parity error
21	Channel 1 I/O Transmission parity error
22	Channel 2 I/O Transmission parity error
23	Channel 3 I/O Transmission parity error
30	Channel 0 External interrupt
31	Channel 1 External interrupt
32	Channel 2 External interrupt
33	Channel 3 External interrupt
42	Out of Bounds
43	Manual interrupt
44	Time interrupt
45	Operand parity error
46	Instruction parity error

A category I interrupt causes the following operations to occur:

1) Stops execution of the "main" program at the end of the instruction currently being performed.

2) Sets the Category I Interrupt Lockout flip-flop.

3) Sets the Category I Interrupt Mode flip-flop.

4) a) If the next instruction would have been a lower instruction, sets the Category I Interrupt Exit flip-flop and subtracts one from the Program Address register (P).

   b) If the next instruction would have been an upper instruction, the Category I Interrupt Exit flip-flop is cleared.

5) Performs a return jump to fixed address 00007.

The following operations occur the next time P is equal to 00007:

1) Clear Category I Interrupt Lockout flip-flop.

2) Clear Category I Interrupt Mode flip-flop.

3) If the Category I Interrupt Exit flip-flop is set, sets the Lower Instruction flip-flop. If the Category I Interrupt Exit flip-flop is clear, sets the Upper Instruction flip-flop.

Category II

This category of interrupt is performed when an attempt is made to execute one of the floating point instructions which are not available in the basic machine (30, 31, 32, 33). Addition of the floating point option (FPO) eliminates this interrupt category.

A category II interrupt causes the following operations to occur:

1) a) If the next instruction would have been a lower instruction, sets the Category II Interrupt Exit flip-flop and subtracts one from P.

   b) If the next instruction would have been an upper instruction, the Category II Interrupt Exit flip-flop is cleared.

2) Sets the Category II Interrupt Mode flip-flop.

3) Performs a return jump to the fixed address specified by the operation code (i.e., if the operation is FSB – operation code 31 – the return jump is to fixed address 00031).

The next time P is equal to the fixed address specified above, the Category II Interrupt Exit flip-flop is examined, and if it is set, the Lower Instruction flip-flop is set. If the Category II Interrupt Exit flip-flop is clear, the Upper Instruction flip-flop is set. The Category II Interrupt Mode flip-flop is cleared.

Category III

When an attempt is made to execute an illegal instruction, a return jump is always made to P = 00020. The program continues or halts, depending on the instruction at location 00020.

Programming Cautions

Note that the Category I interrupt is the only one with hardware lockout. Avoid endless loops caused by improper use of interrupts.

CHAPTER V
INPUT OUTPUT

Input/output facility for the 3400 system is provided by the Input/Output channel. It provides the method for bidirectional data exchange and for proper control of information transmission between the system and its various 3600-type external equipments.

A simplified block diagram of a system is presented in figure 5-1. For purposes of illustration, the system in figure 5-1 shows only one data channel. A basic system includes one I/O channel. Additional I/O channels, up to a total of four, may be added.

An I/O channel may control a maximum of eight external equipments. Typical external devices are line printers, punched card equipment, and magnetic tape equipment. The Connect instruction selects the equipments individually to communicate with the system via the I/O channel.

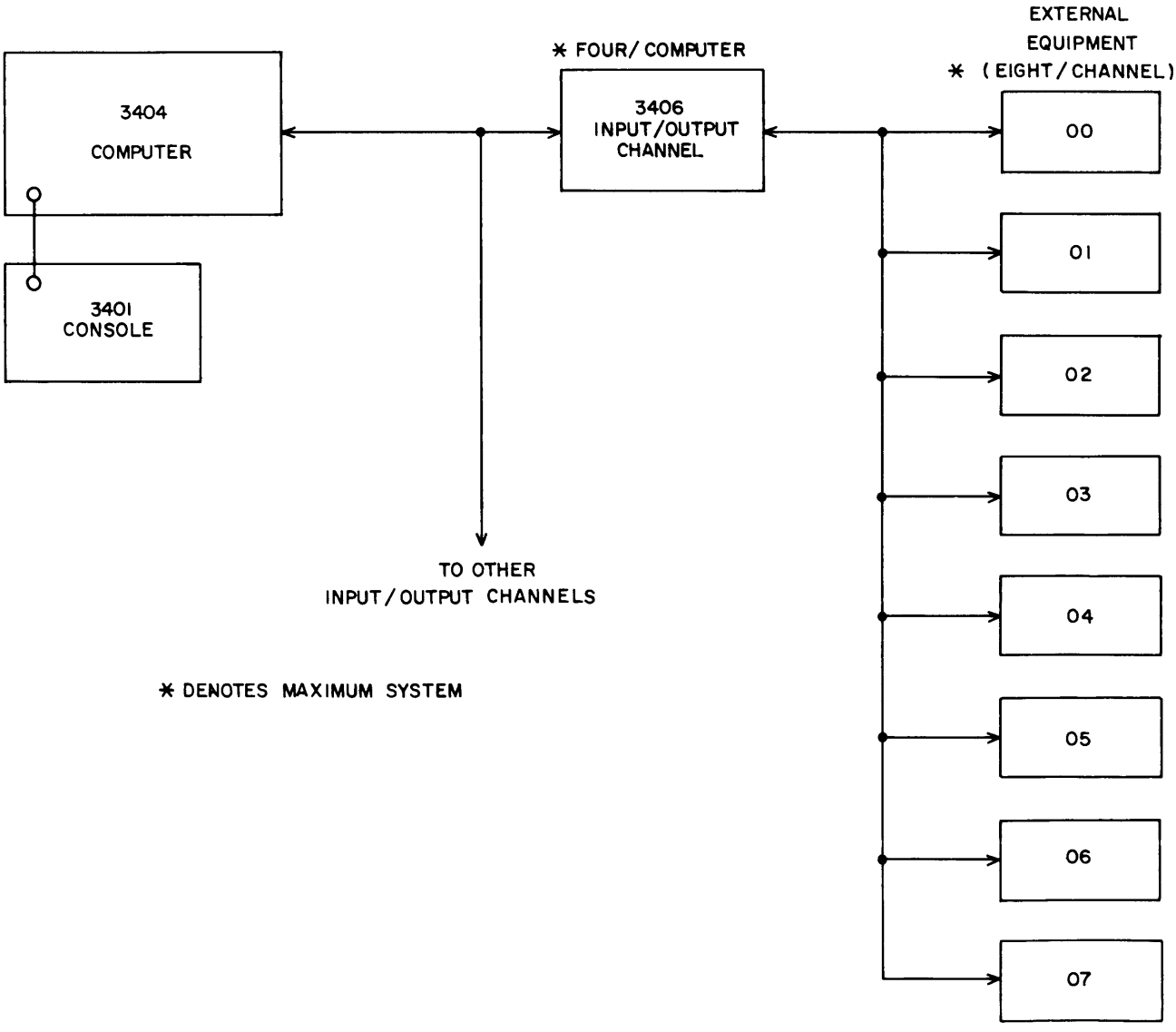


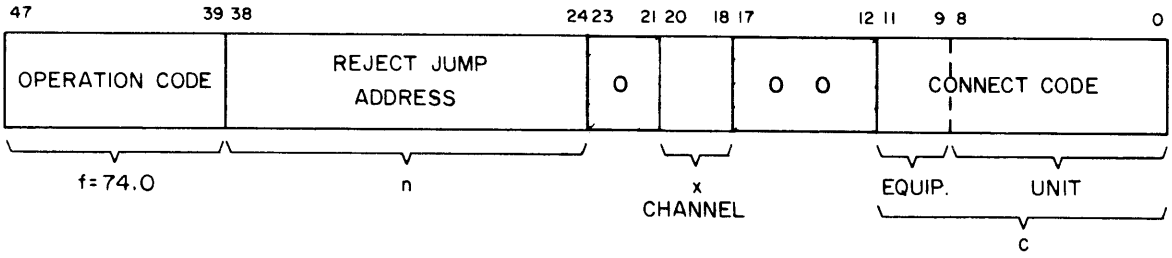
Figure 5-1. 3400 System

INPUT/OUTPUT INSTRUCTIONS

Seven instructions govern input/output operations in the system. These instructions establish operating modes within external equipments and provide for transferring data between the storage and communication modules. Provision is made for sampling the status of operating conditions in the external equipments, and for monitoring the progress of data transmissions.

The Connect instruction must be used to connect the external equipment to the system before data transmission can be initiated. If various operating conditions within the connected equipment are to be specified, a Function instruction is executed. After initial operating conditions have been established, the Read or Write instruction may be executed to transmit data into or out of the system automatically and independent of the main program. In addition, the Read or Write instruction specifies the address of the control word which contains all other information necessary to perform the operation; i.e., starting address, and word count. While input/output operations are in progress, the status of data transmission or operating conditions within the external equipments may be sampled by using the Copy Status instruction. The Clear Channel instruction is used to clear the designated channel at the start of a program or in the event of a program failure.

Connect (*CONN*)



An external equipment is connected to the system via an I/O channel. The digit 'x' specifies one of the four possible data channels attached to the computer.

Up to eight equipments may be attached to each I/O channel. The desired equipment is specified by the upper octal digit of the Connect code 'c'. The equipments are numbered 0 through 7 (octal). Each equipment, via an eight-position switch located on the equipment, may assume any one of the eight possible equipments codes.* Certain equipments, such as magnetic tape controllers, control more than one unit. A number specifies each unit. Unit numbers are the three lower octal digits of 'c' and may range from 000_8 to 777_8. Normally, the legitimate range of unit numbers is from 000_8 to 017_8.

The Connect instruction connects an equipment and/or unit to the computing system by specifying:

1) The 3-bit channel code (one of four channels)
2) The 3-bit equipment code (one of eight equipments assigned to the selected channel)
3) The 9-bit unit code, if any (one of 512 possible units; 16 possible units in all ordinary systems)

When the Connect instruction has been executed:

1) The Function instruction may be used to establish various operating modes within the equipment, if desired, and
2) The Read or Write instruction may be executed to transmit data from or to the designated equipment.

The external equipment remains connected until another Connect instruction is executed for the same channel, or the Clear Channel instruction is executed. Only one equipment may be connected to a channel at any one time. All four channels may have an equipment connected, and all may be active at once. Thus,

*If the operator has selected the same octal equipment number for more than one equipment on a particular channel, a Connect instruction will connect each equipment. Subsequent Read or Write instructions will reference each connected equipment resulting in loss of data or in a program malfunction.

four equipments may be simultaneously communicating with the system.

Under certain conditions, it may not be possible to connect the designated equipment. When one or more of these conditions occurs, a Reject signal is sent to the computer, and a jump is effected to the address specified by the 15-bit reject jump address.

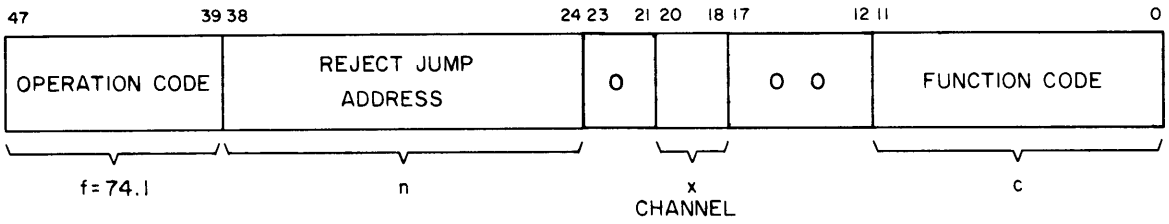
A Reject signal will be sent to the computer only under one or more of the following conditions:

1) Channel Busy: The selected channel is currently performing a Read or Write operation.

2) Unit Unavailable: The unit referenced is in use by another I/O channel. This may occur only if an equipment is multi-channel (such as a magnetic tape controller).

3) False Reference: Whenever the computer receives no response within 100 usec, it generates its own Reject signal and performs the jump. This case may occur if the referenced equipment is not attached to the specified channel or the equipment is inoperative.

Function *EXTF*



The Function instruction specifies operating conditions within an external equipment or a condition on which interrupt may occur. This instruction transmits a 12-bit Function code 'c' to the equipment connected to channel 'x'. This code specifies internal operating conditions of the referenced equipment. A list of codes for each external equipment is included in the associated reference manual.

If an equipment to which the Function instrument is directed has not been previously connected to the system via a Connect instruction, the Function code cannot be recognized, and a Reject signal will be generated. The Reject signal causes the program to jump to the 15-bit reject jump address. The following conditions or combination of conditions will result in a reject.

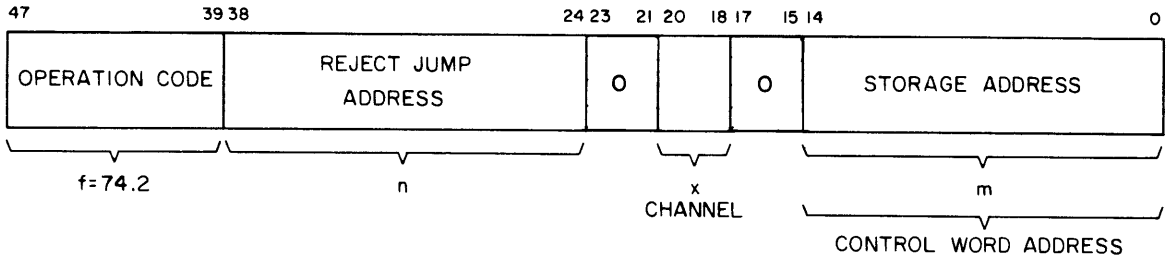
1) No Unit or Equipment Connected: The referenced device is not connected to the system and cannot recognize a Function instruction. If no response is received within 100 usec, the Reject signal is generated automatically by the computer.

2) Illegal Code: The Function code 'c' cannot be interpreted by the specified device. The Reject signal is generated by the external equipment after the attempted reference.

3) Equipment or Unit Busy or Not Ready: The device cannot perform the operation specified by 'c' without damaging the equipment or losing data. For example, a Write End of File code will be rejected by a tape unit if the tape unit is rewinding.

4) Channel Busy: The selected I/O channel is currently performing a Read or Write operation.

Read (*BEGR*)



The Read instruction initiates input activity on channel 'x'.

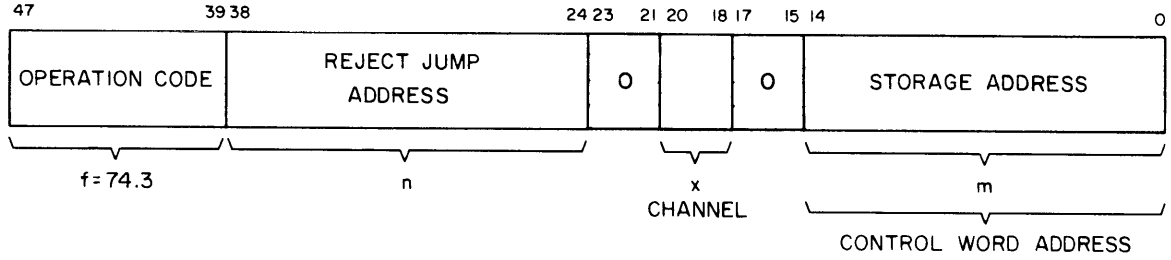
Unless a Reject signal is generated, the main computer program proceeds independently. If a Reject signal is generated, a jump is effected to the 15-bit reject jump address. A Reject signal is generated by a Channel Busy (the designated channel is currently performing a Read or Write operation).

If no Reject signal is given, the control word is fetched from the storage address designated by 'm'. The word count and starting address are placed in their respective registers in I/O channel 'x'.

This activity occurs during main computer program activity. All storage references required for a Read operation are spaced between the main program storage references.

The communication logic and the designated I/O channel control all input activity until the channel becomes inactive.

Write (*BEGW*)



The Write instruction initiates output activity on channel 'x'.

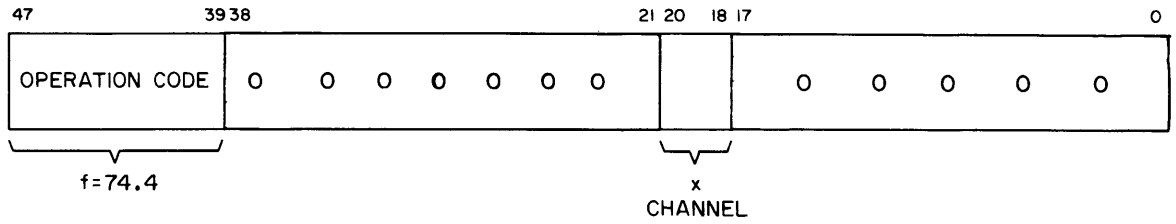
Unless a Reject signal is generated the main computer program proceeds independently. If a Reject signal is generated, a jump is effected to the 15-bit reject jump address. A Reject signal is generated by a Channel Busy (the designated channel is currently performing a Read or Write operation).

If no Reject signal is issued, the control word is fetched from the storage address designated by 'm'. The word count and starting address are placed in the proper registers in channel 'x'.

This activity occurs during main computer program activity. All storage references required for a Write operation are spaced between the main program storage references.

The communication logic and the designated I/O channel control all output activity until the channel becomes inactive.

Copy Status (*COPY*)



After an external equipment has been connected, the Copy Status instruction may be used to determine:

1) The operating conditions

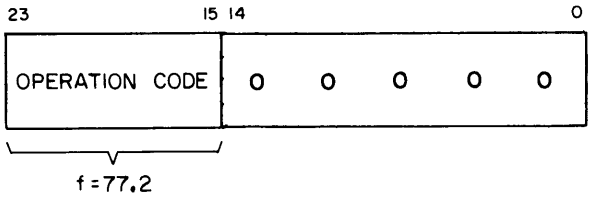
2) The progress of a data transmission

The external equipment issues a 12-bit status code to indicate operating conditions. (For a list of status codes, refer to the reference manual for each equipment.) This code is present at all times on lines from the external equipment to the data channel to which it is "connected".

The status of a data transmission may be determined by examining the word count and the current address. These quantities are held in registers in the I/O channel. Since the word count is reduced by one and the address is increased by one for each data word transmitted, the number of words processed in the operation may be determined at any time by examining these quantities. (The sum of the word count and the current address is always constant.)

The Copy Status instruction is used to determine if the data channel is busy, and to sample the status of the external equipment. This instruction transmits the current channel 'x' word count and address to A, and the channel status bits to Q. Bits are placed in A as follows: 00→14 = current address, 24→38 = word count. Bits are placed in Q as follows: 0→11 = channel status, 24→31 = channel interrupt lines.

Copy Interrupt Status (*CIS*)

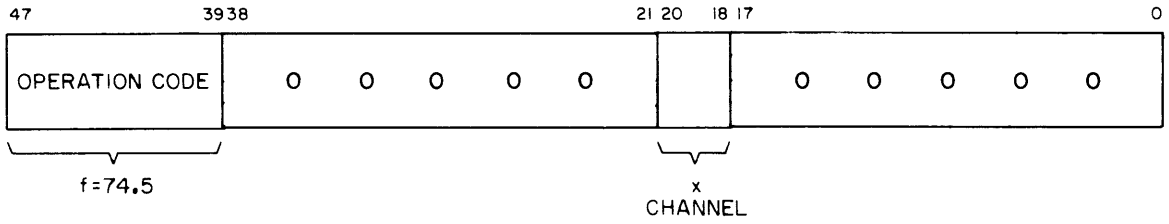


This 24-bit instruction transmits the current interrupt system status to the A register. This is equal to the contents of the Interrupt register. (Unused bits in A are cleared.)

Copy Product Register (*CPR*)

This 24-bit instruction has the same format as the Copy Interrupt Status instruction with the exception that f=77.4. This instruction transmits the contents of the Main Product register to the A register. (Unused bits in A are Cleared.)

Clear Channel (*CLCH*)



The Clear Channel instruction:

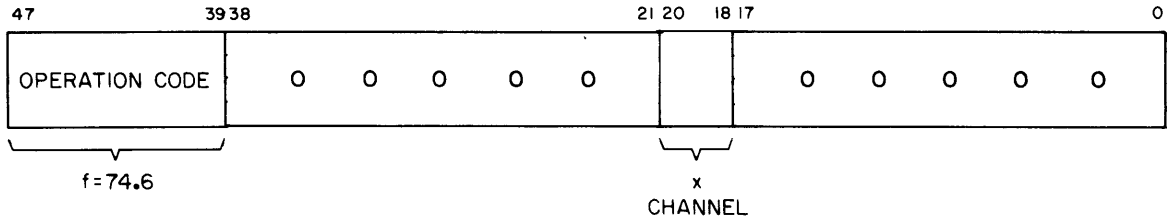
1) Disconnects all equipments from the specified channel, preventing any communication until a Connect instruction is executed.

2) Disconnects all units within an equipment.

3) Clears the channel control word to its original state; i.e., a control word containing all zeros. This cuts off any data transmission in progress.

This instruction assures a cleared channel prior to use when the previous condition of the channel is unknown, or removes a hang-up condition caused by a malfunction.

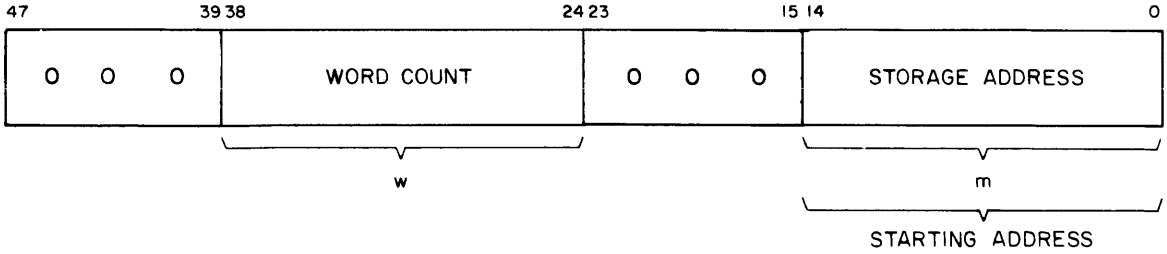
Change Control Word (*CCWD*)



This instruction interchanges the control word on a specified I/O channel with the contents of the A register.

CONTROL WORD

All data transmission between the system and external units is governed by control words associated with each I/O channel. The control word specifies the number of data words to be transmitted into or out of the system, and the starting address in storage for the list of words.



The control word, prior to the input or output operation, is located in storage. Its location is designated by a 15-bit control word address. A Read or Write instruction transmits the control word to the data channel. The control word specified a 15-bit starting address, 'm', from which the first output word will be stored. A 15-bit word count 'w' specifies the number of data words to be transmitted. The starting ad-

dress and word count are placed in their respective registers within the designated I/O channel.

When the current operation is complete (word count is zero), the I/O channel halts the input or output activity (disconnects the channel).

If the word count is not reduced to zero when address 77777 is referenced, the next word written into or read from storage will reference address 00000.

If a return to adress 00000 is not desired, the word count must be such that it is reduced to zero when address 77777 is reached.

AUTO LOAD

The auto-load feature provides a means by which data or instructions contained in an external storage medium may be automatically loaded into storage in the 3400 system. The 3404 performs an internal and external master clear, connects a data channel and an external equipment, and specifies a function. The 3404 then directs the channel to read to end of record, without fetching a control word from storage. The operator can specify the channel and equipment to be connected and the function to be executed by means of switches.

The 3404 enters the count of 40,000 (octal) into the Word Count register by setting the high order bit to a "1". The number 40,000 has no significance other than being large enough to insure that the word count will not be reduced to zero before the end of record is reached.

Any external storage medium may be auto-loaded, including magnetic tape, disc file, and punched cards. The magnetic tape produces an End of Record signal after each record. In the case of punched cards, each individual card constitutes a logical record. If several cards are to be auto-loaded, the first card must contain a "loader". This is a programmed Read instruction covering the remaining cards.

CHAPTER VI
PARITY

In the 3400 system, parity bits are generated and checked to determine one of two possible conditions:

1) Errors in I/O channel transmissions

2) Errors in storage of data or instruction words.

The presence of a parity error may indicate one or more of the above has occurred. Odd parity is used; that is, the parity bit is set such that the total number of ones in the word portion is odd.

In addition to the two cases listed above, the external equipment itself may generate and check parity on its internal operations. Treatment of this case is discussed later in this chapter. For operational details, refer to the Reference Manual for the specific equipment.

PARITY GENERATION

Parity bits are generated in the following cases:

1) On all words entering storage

2) On each 12-bit byte of data transmitted to an external equipment from the I/O channel

3) On each 12-bit byte of data transmitted to I/O channel from an external equipment (generated by the external equipment).

Data Parity Generation

The computer provides three parity bits along with the 48-bit instruction or data word on a Write into Storage operation. The format of this word with its associated parity bits is diagrammed below.

A parity bit is generated by the computer for each of the three portions of the instruction or data word.

Parity Generation For I/O Channel Transmissions

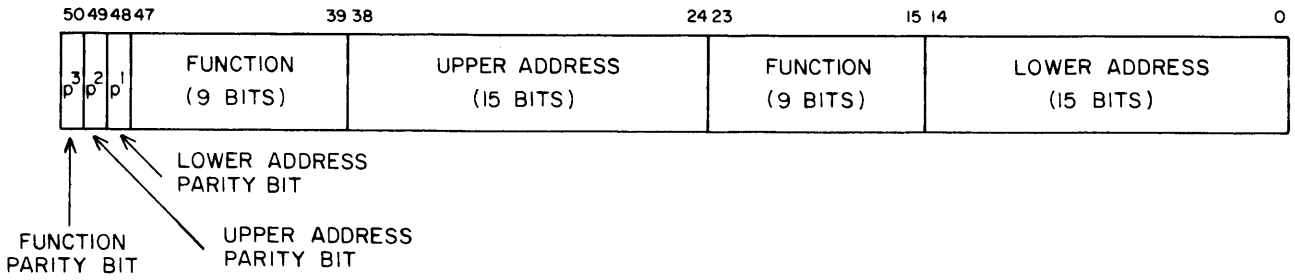
Each 12-bit byte of data being transmitted into an I/O channel from an external equipment has an accompanying parity bit. This parity bit is generated by the external equipment. The I/O channel generates a parity bit to accompany a 12-bit byte of data being transmitted to an external equipment.

PARITY CHECKING

Parity Checking on Storage or Transmission of Data

When an instruction or data word is read from storage by the computer, the computer checks for a parity error. The existence of a parity error may indicate one or more of the following:

1) An error may have occurred either when the word was initially sent to storage or when the word was read from storage.



2) The information was garbled in the storage read/write process itself.

Parity Checking on I/O Channel Transmissions

Each 12-bit byte of data being transmitted from an external equipment into the I/O channel is checked for a parity error by the I/O channel. A parity error indicates a transmission error has occurred. A 12-bit byte of data being transmitted to an external equipment is checked for a parity error by the external equipment.

PARITY ERRORS

When the parity errors occur, the manner in which they are handled depends on the computer. Descriptions of parity errors given below list the cases.

Operand Parity Error

A word read from storage for use as an operand by the computer or for use as an output word by an I/O channel is checked for parity. If an operand parity error occurs, and if the Operand Parity Error bit in the Interrupt Mask register is set (assuming interrupt is active), interrupt occurs. Otherwise the word is accepted regardless of error. In either case the Operand Parity Error indicator lights on the console.

Instruction Parity Error

An instruction word read from storage is checked for parity in the computer. If an instruction parity error occurs, the instruction will not be executed and the computer will stop. An instruction parity error lights an indicator on the console.

I/O Channel Transmission Parity Error

I/O channel transmission parity errors may occur during several operations involving the I/O channel. Each of these cases is outlined below.

1) If a parity error occurs in transmitting the 12-bit Function code to an external equipment from the I/O channel, the following actions take place:

   a) The function specified by the code is not executed.

   b) No external Reject signal is generated by the external equipment; the Reject signal is generated internally by the 3404.

   c) No Reply signal is sent to the computer from the external equipment.

   d) A Parity Error signal is sent to the I/O channel. This signal sets the I/O Parity Error bit in the I/O channel.

   e) In order to initiate operation, either a Restart (from the console) or a Clear Channel instruction must be executed to clear the condition.

2) If a parity error occurs in transmitting the 12-bit Connect code to an external equipment, the following actions occur:

   a) The connection specified by the Connect code is not effected.

   b) No external Reject signal is generated by the external equipment; the Reject signal is generated internally by the computer.

   c) No Reply signal is sent to the computer from the external equipment.

   d) No Parity Error signal is generated.

   e) In order to initiate operation, either an external master clear (from the console) or a Clear Channel instruction must be performed to clear the condition.

3) If a parity error occurs in transmitting a 12-bit byte of data from the I/O channel to an external equipment, the following actions occur:

   a) A Parity Error signal is returned to the I/O channel. This Parity Error signal sets the I/O Parity Error bit in the I/O channel.

   b) The Write operation continues as specified by the control word unless interrupt on I/O parity error is selected. (Refer to following discussion on interrupt selection on I/O parity errors.)

4) If a parity error occurs in transmitting a 12-bit byte of data into the I/O channel from an external equipment, the following actions occur:

   a) The I/O Parity Error bit in the I/O channel is set.

   b) The Read operation continues as specified by the control word unless interrupt on I/O parity error is selected. (Refer to following

discussion on interrupt selection on I/O parity errors.)

c) In order to initiate subsequent operation, either an external master clear (from the console) or a Clear Channel instruction must be performed to clear the condition.

Interrupt Selection on I/O Parity Error

Interrupt may be selected to recognize parity errors occurring in Input/Output operations. Selecting interrupt causes the I/O Parity Error bit in the I/O channel to be examined. The conditions which set this bit are:

1) Data parity error (48-bit words read from storage)

2) Data transmission parity errors:

   a) Parity error on Function code

   b) Parity error on 12-bit data transmission to external equipment from channel

   c) Parity error on 12-bit data transmission from external equipment to channel

Selecting interrupt to recognize one of these occurrences is accomplished as follows:

1) Execute Internal Function instruction (77.0 XXXXX - Select Interrupt on I/O Parity Error).

2) Activate interrupt system via Internal Function instruction (77.0 XXXXX - Set Interrupt Active).

When interrupt is selected and one of the above conditions occurs, an interrupt routine is entered. Within the interrupt routine, the computer isolates the cause of interrupt to an I/O parity error. An I/O parity error constitutes a major machine malfunction. (Refer to chapter 4 and appendix A.)

External Equipment Parity Error

Parity errors within an external equipment (e.g., longitudinal parity error during a Read/Write operation on magnetic tape) are handled differently than the I/O parity errors. These parity errors indicate a difficulty in reading or writing information on the equipment's medium.

If a parity error occurs within the external equipment, a Parity Error signal is returned to the I/O channel via one of the twelve status lines.

Once the interrupt routine has isolated the cause of interrupt to a particular equipment, a Copy Status instruction must be executed to determine the particular condition causing interrupt (e.g., longitudinal parity error). Refer to chapter 4.

CHAPTER VII
3401 CONSOLE

The 3401 console includes various switches, a System Status Display panel, a speaker, and an electric typewriter. Through the use of these devices, the operator can control and monitor the computer and associated peripheral equipments.

SWITCHES

Console switches and their operations are listed in table 7-1.

Table 7-1. Console Switches

Switch	M or L*	Function
Selective Jump 1 Selective Jump 2 Selective Jump 3	L L L	Pressing the Selective Jump switches provides the manual conditions for executing a program jump on the Selective Jump Instruction (75). The Selective Jump switches are illuminated when depressed.
Auto Load	M	Pressing the Auto Load switch provides for automatically loading storage with information from a given external equipment.
Emergency Off	M	Pressing the Emergency Off removes power from the entire system.
Manual Interrupt	M	Pressing the Manual Interrupt switch forces the computer into an interrupt routine if: 1) The Manual Interrupt bit in the Interrupt Mask register is set, and 2) The interrupt system is active.
End-of-Record	M	When depressed, this switch ends an End-of-Record signal to the data channel if the typewriter attached to the data channel is "connected" and read and data signals are present.
Equipment Number	L	This eight-position (0-7) switch designates the console typewriter as equipment N. Bits 9, 10, and 11 of the Connect code must match the octal setting of this switch. (This switch is mounted inside the console on the typewriter logic chassis.)
Typewriter On/Off	L	This switch is located near the typewriter keyboard. It must be in the On position for any typewriter operation.

*M - Momentary, L - Locking

Table 7-1 Continued

Switch	M or L	Function
Fake Reply On/Off	L	When in the On position, this switch automatically selects the typewriter and gives a fake Reply to Read/Write signals on the data channel cabled to the typewriter. This switch is for maintenance use only.
Restart	M	When this switch is depressed, the following events occur: 1) Stop computer and issue on External MC. This stops all computer and I/O activity. 2) Clear Interrupt Active. 3) Restart computer with a return jump to address 00000_8.
System Active On/Off	L	When this switch is on, a "running time" meter (located in the power module) accumulates the time that power is on and the System Active switch is on. If both the System Active and the Maintenance Mode switch (located on the maintenance panel) are off, all other switches except power control will be locked out. When the System Active switch goes from Off to On, both Internal and External Master Clears are issued. When the System Active switch goes from On to Off, the computer stops. Any I/O activity continues until normal termination. The System Active switch is illuminated when in the On state.

INDICATORS

By means of lights, the System Status Display panel indicates certain operating conditions in the computer system. Figure 7-1 shows the System Status Display panel. Table 7-2 gives the conditions indicated by the lights. All displays are active when the condition arises, whether the computer is running or stopped. For definitive descriptions of various fault conditions, refer to appendix A.

Channel 0 Active	Channel 0 Parity Error	Computer Running	Arithmetic Overflow	Shift Fault	Read Next Instruction	Execute Cycle	Storage Overload 1	Temperature 1	Circuit Breaker
Channel 1 Active	Channel 1 Parity Error	Program Stop	Exponent Overflow	Divide Fault	Upper Instruction	Operand Cycle	Storage Overload 2	Temperature 2	Temperature
Channel 2 Active	Channel 2 Parity Error	Interrupt Active	Exponent Underflow	Instruction Parity Error	Lower Instruction	Extended Cycle		Temperature 3	Interlock Bypassed
Channel 3 Active	Channel 3 Parity Error	Interrupt Mode		Operand Parity Error	Maintenance Mode	Jump Condition		Temperature 4	Terminator Power Fault

Figure 7-1. System Status Display Panel

Table 7-2. Display Panel Indicators

Display	Color	Condition When Illuminated
Channel 0 Active Channel 1 Active Channel 2 Active Channel 3 Active	Green	The channel active indicator(s) lights when an I/O channel(s) is performing a read or write operation.
Channel 0 Parity Error Channel 1 Parity Error Channel 2 Parity Error Channel 3 Parity Error	Red	These indicate an I/O channel transmission Parity Error has occurred in one or more I/O channels. (See Chapter 6 of this manual.)
Computer Running	Green	This indicates the computer is running and/or I/O operations are taking place.
Program Stop	Red	The Program Stop indicates that the computer is no longer executing program instructions. (If the Computer Running light comes on and then goes out, but the Program Stop light does not come on, it indicates a "deep-end" condition: i.e., there is an electronic malfunction in the computer.)
Interrupt Active	Orange	The interrupt system is enabled, permitting examination of selected interrupt conditions in an interrupt routine when these conditions occur.
Interrupt Mode	Orange	The computer is processing an interrupt routine.
Arithmetic Overflow	Orange	The absolute value of the sum or difference of two fixed point integers is $\leq 2^{47}$.
Exponent Overflow	Orange	The value of the exponent formed during a floating point add, subtract, multiply, or divide is $> 2^{10} - 1$ (1777_8).
Exponent Underflow	Orange	The value of the exponent formed during a floating point add, subtract, multiply, or divide is < negative $2^{10} - 1$ (-1777_8).
Display	Color	Condition When Illuminated
Shift Fault	Orange	A register shift of more than $177_8 = 127_{10}$ places is specified in a Shift instruction.
Divide Fault	Orange	1) The absolute value of the quotient resulting from a Divide Integer instruction is $\geq 2^{47}$. 2) A fixed point or floating point divide by zero is attempted.
Instruction Parity Error	Red	A parity error occurs when reading a 48-bit instruction word from storage.

Table 7-2. (Continued)

Display	Color	Condition When Illuminated
Operand Parity Error	Red	A parity error occurs in a word read from storage for use as an operand by the computer or for use as an output word by an I/O channel.
Read Next Instruction	White	The computer is reading a 48-bit instruction word from storage.
Upper Instruction	White	The computer is executing the upper instruction portion of a 48-bit word.
Lower Instruction	White	The computer is executing the lower instruction portion of a 48-bit word. (Only the Upper Instruction indicator is illuminated when the computer is executing a 48-bit instruction.)
Maintenance Mode	Red	The Maintenance Mode switch on the maintenance panel is active.
Execute Cycle	White	The computer is executing an instruction which requires no memory reference.
Operand Cycle	White	The computer is executing an instruction which requires referencing memory.
Extended Cycle	White	The computer is executing an instruction which requires two cycles.
Jump Condition	Green	The computer is about to execute a Jump instruction and the condition for jumping has been met.
Storage Overload 1 Storage Overload 2	Red Red	A current overload has occurred in the storage stack(s).
Temperature 1 Temperature 2 Temperature 3 Temperature 4	Orange	The temperature in one or more cabinet in the system is exceeding the normal range.
Circuit Breaker	Red	A circuit breaker(s) in the system has tripped due to a current overload.
Temperature	Red	The temperature in a section of the system has reached a point where the equipment may be harmed. The motor-alternator is turned off.
Interlock Bypassed	Orange	The Interlock Bypassed switch on the maintenance panel has been depressed. Computation proceeds regardless of the temperature in any part of the system.

Table 7-2. (Continued)

Display	Color	Condition When Illuminated
Terminator Power Fault	Red	If the terminator power fluctuates or drops, this indicator lights.
Type-In	White	This indicator is located next to the typewriter. When this light comes on, it indicates that the computer is executing a read instruction and the input operation may begin via the typewriter.

TYPEWRITER

A 731 Selectric typewriter is used as a program monitor device in the computer system. The typewriter is cabled to an I/O channel and is operated using computer instructions pertinent to external equipments. These instructions include Connect, Function, Copy Status, Read, and Write. (For a description of switches and indicators related to typewriter operations, see the preceding section on switches and indicators in this chapter.)

The typewriter codes in tables 7-3 and 7-4 are used for both input and output operations.

CONNECT

Connect Equipment N (*NXXX*)

Bits 9, 10, and 11 of the Connect code must match the octal setting of the Equipment Number switch. If the switch setting and these bits do not agree, the typewriter will not be connected. No signals will be returned to the I/O channel. A new Connect code not equal to NXXX clears a previous connection unless a transmission parity error occurs or has occurred and has not been cleared.

FUNCTION

Set Interrupt On Abnormal Operation (*XX01*)

If bit 0 is present in the Function code (accompanied by a Function signal), an Interrupt on Abnormal Operation condition is established in the typewriter. (In all discussion of codes, bit 0 is in the rightmost position, and bits are numbered from right to left in ascending order.) This allows the typewriter to send an interrupt signal to the computer when either or both of two conditions occur:

1) End of Line

   The typewriter has reached the right margin of the line it is typing.

2) Type Parity Error

   A parity error has occurred in the typewriter logic. (See the Type Parity Error discussion under Status Lines for additional comments.)

The Interrupt signal is transmitted on the line which corresponds to the equipment number (N) of the typewriter.

Clear Interrupt On Abnormal Operation (*XX02*)

If bit 1 is present in the Function code, the Interrupt on Abnormal Operation condition is cleared.

Clear Interrupt (*XX04*)

If bit 2 is present in the Function code, the Interrupt signal is cleared. The presence of bit 2 in the Function Code also clears the interrupt signal if caused by a manual interrupt.

STATUS

Ready (*XXX1*)

If bit 0 is present in the status code, power has been applied to the 3692 control logic, the typewriter is turned on, and I/O operations may proceed.

Busy (*XXX2*)

If bit 1 is present in the Status code, it indicates that the typewriter is busy. When in this condition,

Table 7-3. 731 Typewriter Codes

Manifold Ten*

Lower Case	Code	Upper Case	Lower Case	Code	Upper Case
A	12				
B	01		0 (zero)	43	)
C	11		1	77	±
D	55		2	37	@
E	51		3	33	#
F	30		4	47	$
G	74		5	57	%
H	45		6	13	¢
I	16		7	53	& (and)
J	70		8	17	* (asterisk)
K	15		9	07	(
L	41		!	76	° (degree)
M	72		.	32	.
N	31		'	52	"
O	42		;	50	:
P	54		,	14	,
Q	10		/	44	?
R	56		=	34	+
S	46		Dash -	04	Underline _
T	75		Space	60	Space
U	35		Backspace	61	Backspace
V	36		Tab	62	Tab
W	02		C. R.	63	C. R.
X	71		U. C.	64	U. C.
Y	40		L. C.	66	L. C.
Z	73				

* Named after the 731 type font.

Table 7-4. Connect, Function, and Status Codes

CONNECT	
Connect Equipment N	NXXX
FUNCTION	
Set Interrupt On Abnormal Operation	XX01
Clear Interrupt On Abnormal Operation	XX02
Clear Interrupt	XX04
STATUS	
Ready	XXX1
Busy	XXX2
End of Line	XX4X
Upper/Lower Case	XXX4
Type Parity Error	2XXX

the typewriter is doing one of the following operations:

1) Typing a character
2) Backspacing
3) Doing a carriage return or tab
4) Shifting to the upper case
5) Shifting to the lower case
6) Processing a 00, 65, or 67 code.

   If one or both 6-bit frames of the 12-bit word received by the typewriter during a normal output (Write) operation equal 00, 65, or 67, nothing is typed or spaced corresponding to that code. The time interval required to process these codes is approximately 3 usec.

Upper/Lower Case (*XXX4*)

If bit 2 is present in the Status code, the typewriter is in upper case. If bit 2 is not present in the Status code, the typewriter is in lower case.

End of Line (*XX4X*)

If bit 4 is present on the status lines, it indicates the typewriter has reached the end of a line. Any further codes received by the typewriter will all be typed in the end character location unless a carriage return is executed.

Type Parity Error (*2XXX*)

If bit 10 is present in the Status code, a type parity error has occurred. This indicates some logic has failed in the typewriter control and the character typed is not necessarily the one specified by the code. A type parity error can only occur during an output operation. The Type Parity Error signal drops when an I/O master clear or a Clear Channel instruction is performed. It also drops if a Carriage Return or Clear Interrupt Function code is received.

PROGRAMMING

The general order of events when using the typewriter for an input/output operation via an I/O channel is:

1) Set tabs, margins, and spacing. Turn on the typewriter, logic control power.
2) Clear
3) Connect
4) Check status
5) Function
6) Write/Read

Set Tabs, Margins, and Spacing

All tabs, margins, and paper spacing must be set manually prior to the output operation. A tab may be set for each space on the typewriter between the margins.

Clear

There are two types of clears which may be used to clear all conditions existing in the typewriter control. These are:

1) External Master Clear.
   This signal is sent out on all I/O channels. In the case of the typewriter, it clears all functions, control logic, and an Interrupt signal if one exists. The external MC clears the Interrupt condition selected by a XX01 function code.

2) Clear Channel Instruction.
   This instruction performs the same operation as an external master clear, except it normally only applies to one I/O channel. (See this reference manual for a detailed description of this instruction.)

Connect

The 12-bit portion of the Connect instruction (accompanied by a Connect signal) connects the typewriter to an I/O channel. A Reply signal is returned to the channel when the connection has been made. There is no external reject under any circumstances.

Check Status

The programmer may wish to check the status of the connected typewriter before proceeding. This is done with a Copy Status instruction. (See this reference manual for a detailed description of Connect, Function, Copy Status, Bit Sensing, and Write instructions.) Status information is returned to the I/O channel on five of twelve status lines. The Bit Sensing instruction may be used to determine the status of the connected typewriter. If the programmer is certain of the status of the typewriter, this operation may be omitted.

Function

The 12-bit portion of the Function instruction (accompanied by a Function signal) performs a certain operation (depending on the code). These codes are listed in table 7-4. Since only 1 bit of the Function code needs to be interpreted to perform a specified operation, it is possible to combine operations using one code. For example: $XXX6_8 = (XXX110)_2$ would Clear Interrupt on Abnormal Operation and Clear Interrupt.

Example:

XXX4	Clear Interrupt
XXX2	Clear Interrupt on Abnormal Operation
XXX6	Clear both

Write

A Write instruction starts the output operation using the codes listed in table 7-3. The upper 6 bits of the 12-bit data word received from the I/O channel are translated and the character matching the code is typed. Then the lower 6 bits of the code are translated, the character is typed, and a reply is returned to the data channel.

Note that the typewriter must be in the lower case to type letters. If the typewriter is in the upper case when a letter is to be typed, the typewriter will space. No letter will be typed, but the operation will continue as in a normal output.

The typewriter is automatically placed in lower case at the beginning and end of every Write (output) operation (upper case can still be selected when the Write operation begins).

The typewriter keyboard does not have a lockout during an output operation. If a key, space bar, etc., is accidentally pressed during output, the typewriter may miss a character or type something other than the normal output character.

Read (*input*)

A Read instruction starts the input operation using the codes listed in table 7-3. When the type-in indicator on the console lights, the operator may enter information on the typewriter.

Input is character mode only. Six bits of information are entered into bit positions 0-5 of the 48-bit input word. Bit positions 6-47 are filled with zeros. Thus, the first three 12-bit words the I/O channel receives from the typewriter are all equal to zero. The last 12-bit word the I/O channel receives from the typewriter has a six-bit code in bit positions 0-5 (bit positions 6-11 = zeros).

During a Read operation, codes are entered corresponding to the symbols typed, except for the capital letters A through Z. If the typewriter is in the upper case during a Read operation, the code corresponding to the letter will be sent to the channel, the letter will not be typed, and the typewriter will space.

The mechanical operations, such as backspace or carriage return, send their corresponding codes to the I/O channel. For example: shifting from lower to upper case enters a 64_8; shifting from upper case to lower case enters a 66_8.

CHAPTER VIII
3655 HIGH-SPEED LINE PRINTER

The 3655 Printer is a peripheral equipment with capabilities for printing 1000 lines/minute. A maximum of 64 characters and 120 columns are available.

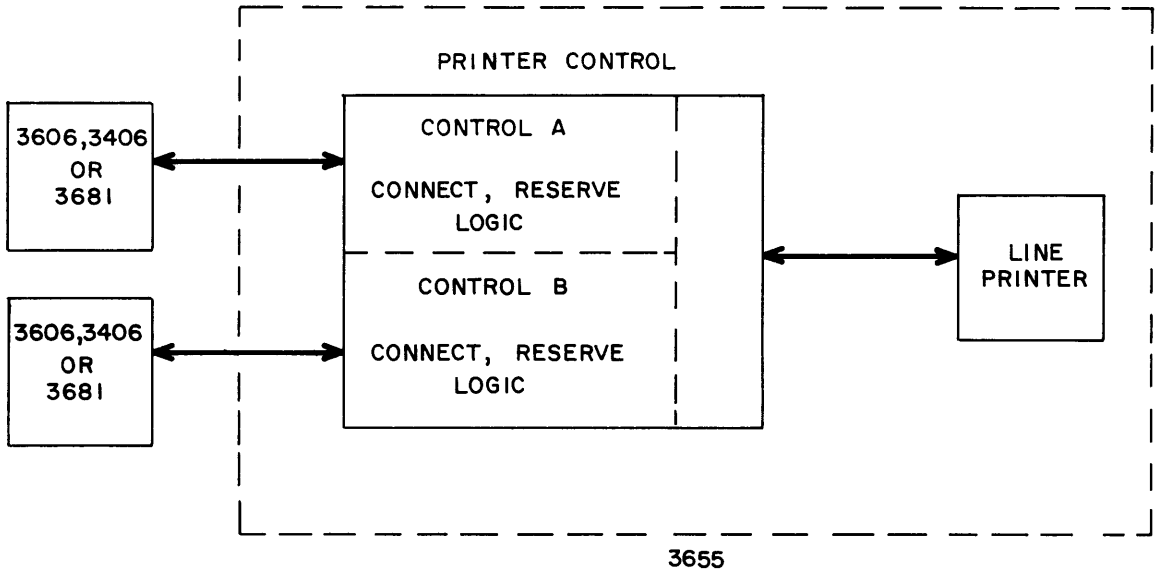


Figure 8-1. Typical Printer Configuration

TYPICAL PRINTER CONFIGURATION

The printer contains two controls, A and B. A 3606 type* data channel may be physically attached to each control. Only one control may communicate with the printer at any given time. Each control in the 3655 contains reserve logic such that the other control may be prevented from communicating with the printer.

SWITCHES

Equipment Number Selection Switch (2)

An 8-position Equipment Number Selection switch (one on each control) designates the control as equipment Number N. (N may have any value from 0---7.) Any interrupts will be transmitted to the data channel on interrupt line N.

INDICATORS

Connect (2)

A Connect indicator (one on each control) lights when the control is connected to a data channel. The indicator goes out when the control is no longer connected to the channel.

Reserve (2)

A Reserve indicator (one on each control) lights when the printer is reserved by a control. This indicator goes out when the control is no longer reserving the printer.

Transmission Parity Error (2)

A Transmission Parity Error indicator (one on each control) lights when a transmission parity error is

* This includes the 3406 I/O channel, 3606 data channel, and the 3681 data channel converter.

detected in connect, function, or data codes. This light goes out when a Master Clear is performed on the control.

CODES

All Connect, Function, and Write operations are performed by the 12-bit octal codes listed in tables 8-1 and 8-2.

Connect

Connect Equipment N (N000)

This 12-bit portion of the Connect instruction, accompanied by a Connect signal, connects control A/B and reserves the printer for the control. Bits 9, 10, and 11 (N) of the connect code must match the setting of the Equipment Number Selection switch on the control. If N does not match the Equipment Number Selection switch setting, or the printer is not in a Ready condition, the connection is not made and no signals are returned to the data channel.

If a transmission parity error occurs in the connect code, the transmission parity error indicator on the control is illuminated. The connection is not made and no signals are returned to the data channel.

Function

All functions will be rejected if the printer is not Ready, except in the case of the 0040 code.

Clear Reserve (0040)

If bit 5 is present in the function code, the reserve for a control is cleared. This enables the other data channel to connect and reserve the printer.

Select Interrupt (0030)

This selects interrupt in the printer. The printer will return an Interrupt signal via a control to the data channel when one line has been printed after establishing the interrupt condition and the printer is ready to receive more data. The Interrupt signal will remain up until the interrupt condition is cleared.

Table 8-1. Connect, Function, and Status Codes

CONNECT	Connect Equipment N	N000
FUNCTION	Clear Reserve	0040
	Select Interrupt	0030
	Clear Interrupt	0031
	Advance Paper One Line	0001
	Advance Paper Two Lines	0002
	Advance Paper to Select Line (format channel 7)	0003
	Advance Paper to Top of Form (format channel 8)	0004
	Suppress Paper Advance	0006
	Clear Format Selection	0010
	Select Format Channel 1	0011
	Select Format Channel 2	0012
	Select Format Channel 3	0013
	Select Format Channel 4	0014
	Select Format Channel 5	0015
	Select Format Channel 6	0016
STATUS	Printer Ready	Bit 0
	Printer Reserved for Other Control	Bit 11

Table 8-2. 1612 Line Printer Codes

(These codes are in external BCD)

CHARACTER	OCTAL CODE	CHARACTER	OCTAL CODE
:	00	-	40
1	01	J	41
2	02	K	42
3	03	L	43
4	04	M	44
5	05	N	45
6	06	O	46
7	07	P	47
8	10	Q	50
9	11	R	51
0	12	V	A 52
=	13	$	53
≠	14	*	54
≤	A 15	↑	A 55
%	A 16	↓	A 56
[	A 17	>	A 57
blank	20	+	60
/	21	A	61
S	22	B	62
T	23	C	63
U	24	D	64
V	25	E	65
W	26	F	66
X	27	G	67
Y	30	H	70
Z	31	I	71
]	A 32	<	A 72
,	33	.	73
(	34	)	74
⌜	A 35	≥	A 75
≡	A 36	¬	A 76
∧	A 37	;	A 77

Clear Interrupt (0031)

This clears the interrupt condition in the printer that was established by an 0030 function code. It also clears the Interrupt signal if one is present.

Advance Paper One Line (0001)

This code advances paper one line in the printer without printing. This function is self-clearing when the paper advance is complete.

Advance Paper Two Lines (00020

This code advances paper two lines in the printer without printing. This function is self-clearing when the paper advance is complete.

Advance Paper to Select Line (0003)

This code advances paper according to the holes punched in channel 7 of the format tape. This function is self-clearing when the paper advance is complete.

Advance Paper to Top of Form (0004)

This code advances paper according to the holes punched in channel 8 of the format tape. This function is self-clearing when the paper advance is complete.

Suppress Paper Advance (0005)

This code suppresses the paper advance after a line has been printed. This allows the next line to be printed directly over the first line. This function is self-clearing after the first line has been printed after initiating this function; i.e., when the second line has been printed over the first, paper advances according to the selected format.

Select Format Channel 1 (0011)

Select Format Channel 2 (0012)

Select Format Channel 3 (0013)

Select Format Channel 4 (0014)

Select Format Channel 5 (0015)

Select Format Channel 6 (0016)

Function codes 0011 through 0016 select a particular channel of the format tape for advancing paper in the printer. Any combination of one or more format channels, including all six format channels, may be selected for advancing paper. See the section on Format Tape Preparation in this chapter for more detailed information on format tape preparation and use.

Clear Format Selection (0010)

This function code clears all format channels selected by 0011 through 0016 codes.

Status

Printer Ready (Bit 0)

If bit 0 is present on the status lines, the printer is ready; i.e., selected by a control and not busy. When the printer is ready it can accept function or data codes. The printer is busy when it is printing or advancing paper.

Printer Reserved for Other Control (Bit 11)

If bit 11 is present on the status lines, the printer is reserved for the other control. The data channel receiving bit 11 on its status lines cannot select the printer until the other control clears its reservation.

PROGRAMMING

The general order of events involved in programming the printer when used with the printer adapter is:

1) Printer preparation
2) Clear (not usually needed)
3) Connect
4) Check status
5) Function
6) Write (print)

Printer Preparation

See the 1612 manual for operating instructions.

Clear

Before using the printer, it may be necessary to clear certain conditions existing in the control(s) and the printer. There are three types of clears that may be used:

1) Power On Mc - - A 40 msec Master Clear is sent to the printer when power is applied to the adapter. This clears all function logic in the printer. It also clears Transmission Parity Error and Reserve FFs in each control.

2) Clear Channel Instruction - - This performs a MC on the control physically attached to the data channel over which the MC is issued. This MC clears the current connection and reservation the control has with the printer. It also clears the Transmission Parity Error signal if one is present.

3) External MC - - This performs the same operation as the Clear Channel instruction, except this clear goes out on all channels.

Connect

Control A and Control B in the adapter each contain connect and reserve logic. When the connect code is received by a control, the connect logic sets (providing the Equipment Number Selection switch and N of the connect code match and no transmission parity error has occurred). Only when the connect logic sets (connection made) can Reply, Reject, and Status signals return to the data channel (or data channel converter). After the connection is made, the reserve logic sets and reserves the printer if the other control does not have its reserve logic set. When the operation is complete, a Reply is returned to the data channel.

If, for example, Control B is reserving the printer and Control A attempts to connect and reserve the printer, a Reject will be returned by Control A to the data channel. Control A will be "connected" in the sense that information will be available in bits 0 and 11 of the status lines, but any other communication with the printer is impossible.

See the description of the connect code in this chapter for additional information.

If both data channels attempt to connect Controls A and B and reserve the printer simultaneously, a priority circuit will allow only one connect at a time to be processed.

Check Status

See the description of the status codes in this chapter. When a Connect instruction is executed and if (1) a transmission parity error or (2) a mismatch of N and the setting of the Equipment Number Selection switch has NOT occurred, information will be available on the status lines. Thus, if a Connect instruction results in a Reject, bit 11 of the status lines will indicate the printer is reserved for the other control.

Function

See the description of the function codes in this chapter. A Reply is returned to the data channel when the desired function has been selected or executed. If a transmission parity error occurs in the function code, the adapter control ignores the function and no signals are returned to the data channel. No signals are returned to the data channel if the adapter control is not connected to the data channel. If the printer is not in a ready condition when the Function instruction is executed, a Reject will be returned to the data channel in the following case:

If the function code = 0040 (Clear Reserve) a Reply will be returned to the data channel regardless of whether the printer is ready or not ready.

If the adapter control receives a function code and the control is connected but does not have the printer reserved, a Reject will be returned to the data channel.

Write (*print*)

See the description of the Write instruction in this manual and the printer codes in table 8-2.

If the adapter control has been connected and has reserved the printer, the Write operation may begin. The 12 bits received by the adapter control from the data channel (highest 12 bits of the 48-bit word in storage) are disassembled, converted from 6-bit internal to 6-bit external BCD codes, and sent to the printer memory in separate 6-bit quantities. Bits 6 - - -11 correspond to the leftmost column on the print wheel; bits 0- - -5 correspond to the next column. When the printer memory has accepted two 6-bit character codes, the control returns a Reply to the data channel. The operation continues until one or more 48-bit word(s) (8 characters/48-bit word) has been sent to the printer.

The printer can print a maximum of 120 characters/line. If the buffer length exceeds 120 characters (720 bits) the Write operation will overlap the printer memory and will produce garbled output.

When the Write signal drops, the printer prints one line and advances paper according to the selected format. A separate Write instruction must be used for each line printed. When the Write operation terminates, the printer only prints according to the codes it has received; i.e., if the buffer length is less than 120 characters, the remainder of the line will be left blank.

FORMAT TAPE PREPARATION

Paper is always automatically advanced a single line following a printing operation unless suppressed by a programmed 0006 instruction. The Function instructions can control single or double line advancing. The Function instructions also select the format and monitor channels which provide for multiple line spacing.

An 8-level paper or Mylar tape in the printer controls the format and monitor channels. This tape advances one frame each time the paper is advanced one line. The selected level of the tape is then examined. If a hole is present the paper advance operation terminates. If there is no hole, line spacing continues.

The format tape, approximately 6 1/2 inches in length, is joined in a continuous loop. This tape contains 66 frames, corresponding to the 66 lines on the printed page. During any spacing operation, the format tape is advanced one frame each time the paper advances one line; thus it makes a complete revolution each time a page passes through the print head.

In addition to the eight channels, the format tape includes a row of feed holes. These engage cogs on a metal drum and drive the tape. Nine metal brushes read the format tape. One brush rests on the drum; the other eight rest on the format tape, making no electrical contact with the drum. When a hole in the format tape allows a brush to touch the drum, the circuit is completed. Paper spacing will stop if that particular channel has been selected to govern the format. Therefore, by the selection of one of the eight tape channels, the spacing of paper may be stopped at any point.

Paper motion is started in two ways. Channels 1 through 6, the monitor channels, provide a means of extending the automatic single space feature to include more than one line. The function codes which select channels 7 and 8, the format channels, will also initiate the spacing operation. Channels 7 and 8 are self-clearing; channels 1 through 6 remain selected until cleared by the function code.

Holes may be punched in the channels of the format tape in any desired pattern, with one exception: channel 7 must always contain only one hole punched in the first frame. Channel 7 may be selected by function code 0003, or by pressing the Eject button on the control panel.

Channel 8 also contains only one hole. This hole may be punched in any frame corresponding to the desired format. In the example shown in figure 8-2, the hole in channel 8 is in the fifth frame; consequently, the first line was printed on the fifth line of the paper.

Figure 8-2 shows a sample page layout. On the left is the format tape. (Figure 8-2 is not drawn to scale.)

The hole in channel 7 corresponds to the top of the page. In this example assume that the operation begins at that point. Next, program the function code 0004 which spaces the paper to the first printed line (a distance of 5 lines). Channel 8 will be automatically cleared.

Channel 3 contains holes in every third frame and channel 2 contains holes in every second frame. This is a convenient arrangement because the form contains a number of triple spaces and double spaces. After channel 8 is cleared, channel 3 must be selected. This allows printing in lines 7, 10, and 13, producing a double-spaced main heading, and triple-spacing to the column headings and the first tabulated figures. When line 13 has been printed, clear the format selection (0010) and reselect channel 2 (0012). This gives double spacing.

The columns are totaled eight lines from the bottom of the page. To do this, punch a hole in the 59th frame of channel 4. When the columns have been printed, channel 3 must be cleared and channel 4 selected by EXF instructions.

The signature appears three lines below the total. Reselect channel 3 to provide for triple-spacing. Channels 3 and 4 are then cleared simultaneously by a programmed 0010 function code. Select channel 8 to space to the first line of the next page.

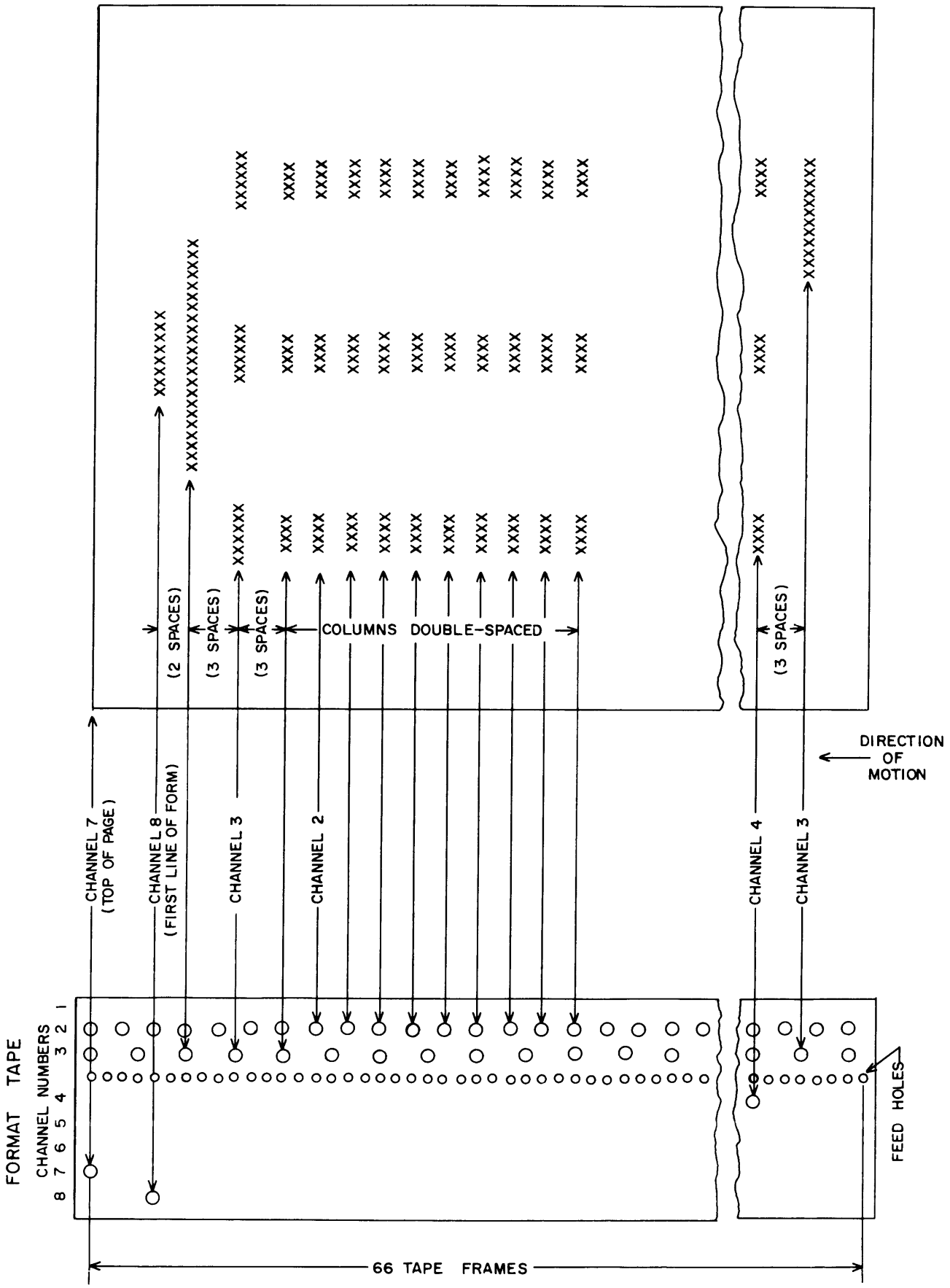


Figure 8-2. Sample Page Layout

CHAPTER IX
3691 PAPER TAPE READER/PUNCH

The CONTROL DATA 3691 Paper Tape Reader/Punch contains the CONTROL DATA 350 Paper Tape Reader, the Teletype BRPE-11 Paper Tape Punch, and a common control unit, all located in a desk type cabinet. A 3600-type bi-directional data channel controls all I/O and related activities in the 3691.

The control unit of the paper tape station is shared by both the reader and punch. The reader and punch cannot concurrently operate since the control unit can be connected to only one data channel.

The punch perforates tape at speeds up to 110 characters per second. Paper tape of either seven or eight levels may be punched.

The reader converts information on punched tape to electrical signals. Five, seven, or eight-level tapes of standard widths can be read photoelectrically at rates up to 350 characters per second. The reader can be programmed to operate start/stop, a character at a time, or to read continuously. A switch on the unit must be set corresponding to the number of the tape levels being read.

SWITCHES AND INDICATORS

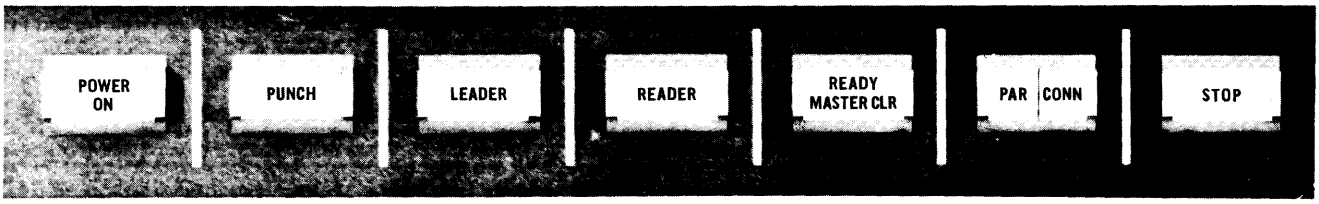


Figure 9-1. 3691 Controls

Power

This switch applies power to the logic chassis, blowers, and Punch and Reader switches.

Punch

This switch applies power to the punch motor.

Leader

This momentary switch causes the punch to feed tape until the switch is released.

Reader

This switch turns the reader motor and exciter on and off.

Master Clear

This momentary switch makes the station ready after loading or reloading.

Stop

Pressing the momentary Stop switch causes the station to become not Ready. Any reader/punch tape motion will stop.

Ready

A Ready indicator (located in the MC switch) lights when the connected unit is in a ready condition.

Parity

A Parity Error indicator lights when a transmission parity error is detected in Connect, Function, or Data codes. A Transmission Parity Error signal is returned to the data channel if a parity error occurs in Function or Data codes. A MC clears the light and error signal.

Connect

A Connect indicator lights when the reader/punch is connected.

Equipment Number Selection

An eight-position switch (positions 0-7) is located in the logic chassis. Bits 9, 10, and 11 of the Connect code must correspond to the setting of this switch. All interrupts are transmitted to the data channel on one of eight interrupt lines corresponding to the setting of the Equipment Number Selection switch.

Table 9-1. Operating Codes and Status Responses

Read		Punch	
N000	Connect Reader	N001	Connect Punch
0001	Read Assembly Mode	0001	Punch Assembly Mode
0002	Read Character Mode	0002	Punch Character Mode
0005	Clear	0005	Clear
0020	Interrupt on Ready and Not Busy	0020	Interrupt on Ready and Not Busy
0021	Release Interrupt on Ready and Not Busy	0021	Release Interrupt on Ready and Not Busy
0022	Interrupt on End of Operation	0022	Interrupt on End of Operation
0023	Release Interrupt on End of Operation	0023	Release Interrupt on End of Operation
0024	Interrupt on Abnormal End of Operation	0024	Interrupt on Abnormal End of Operation
0025	Release Interrupt on Abnormal End of Operation	0025	Release Interrupt on Abnormal End of Operation

Status Responses	
XXX1	Station Ready
XXX2	Station Busy
XXX4	Punch Tape Supply Low
XX1X	Reader Last Device
X2XX	Interrupt due to Ready and Not Busy
X4XX	Interrupt due to End of Operation
1XXX	Interrupt due to Abnormal End of Operation

CODES

Connect

Connect Reader (N000)

Bits 9, 10 and 11 (N) must match the setting of the Equipment Number Selection switch. Bits 0, 1 and 2 must be "0" to connect the reader and enable all status lines pertinent to the reader.

Connect Punch (N001)

Bits 9, 10 and 11 (N) must match the setting of the Equipment Number Selection switch. Bit 0 must be a "1" and bits 1 and 2 "0" to connect the punch and enable all status lines pertinent to the punch.

Function

Read Assembly or Punch Assembly Mode (0001)

This code will allow the following to occur (see code 0005, Clear, for other ways of establishing this mode).

Read Assembly Mode

Starting with the first frame containing a 7th level hole, the first frame of tape is placed in the upper 6 bits of the first byte, the second frame is placed in the lower 6 bits of the first byte, and a Reply signal is sent. This sequence continues with odd frames being placed in the upper 6 bits of each byte and even frames completing the byte. The Word Mark signal comes up with the Data signal during the last 12-bit byte of each word. When the last byte has been transferred, the Word Mark signal drops, signifying that the next frame will begin a new word, and should contain a 7th level hole. If the control hole does not appear, the reader clutch is dropped to stop tape motion, and an End of Record signal is sent to the data channel.

Punch Assembly Mode

The first frame is punched with the 7th level hole and information contained in the upper 6 bits of the first 12-bit byte. The lower 6 bits go to the second frame and so on. Thereafter, the 7th level hole is punched only with the first 6 bits of the byte received after a Word Mark signal; i. e., the first byte of a word.

Read Character or Punch Character Mode (0002)

This code will allow the following to occur:

Read Character Mode

Information goes directly from the tape frame to the lower order bits of the last byte. Replies with no information are returned for each Data signal which does not have an accompanying Word Mark.

Punch Character Mode

The eight lower order bits of a computer word are transferred directly to the paper tape. Replies are returned for each Data signal which does not have an accompanying Word Mark, with no action occurring.

Clear (0005)

This code automatically sets the paper tape reader/punch to the Assembly mode. Also, any interrupt selections are cleared.

Interrupt on Ready and Not Busy (0020)
Release Interrupt on Ready and Not Busy (0021)

Two conditions are responsible for this interrupt – ready and not busy.

1) A ready condition exists for a device if its associated Power switch is on, the Ready light is on, and the device is connected.

2) A not busy condition exists if neither a punch nor reader cycle is in process.

The interrupt active signal may be cleared by one of the following:

1) 0020 (Interrupt on Ready and Not Busy)

2) 0021 (Release Interrupt on Ready and Not Busy)

3) 0005 (Clear)

Code 0022 Interrupt on End of Operation
Code 0023 Release Interrupt on End of Operation

Code 0022 or 0023 or 0005 will clear the interrupt on End of Operation signal.

Reader Interrupt on End of Operation

An interrupt will occur when:

1) The Channel Busy line drops for either Assembly or Character mode, or

2) A reader not ready condition exists.

Punch Interrupt on End of Operation

For either Assembly or Character mode an interrupt occurs when:

1) The Channel Busy signal is down and the cycle is complete, or

2) A punch not ready condition exists.

Code 0024 Interrupt on Abnormal End of Operation
Code 0025 Release on Abnormal End of Operation

The Interrupt on Abnormal End of Operation signal may be cleared by any of the following function codes:

1) 0024 (Interrupt on Abnormal End of Operation)

2) 0025 (Release Interrupt on Abnormal End of Operation)

3) 0005 (Clear)

Reader Interrupt on Abnormal End of Operation

For either Assembly or Character mode:

1) An interrupt will occur if tape motion failure occurs, or

2) A reader not ready condition develops.

Punch Interrupt on Abnormal End of Operation

For either Assembly or Character mode:

1) An interrupt will occur if the paper tape supply is low at the end of a punch cycle, or

2) A punch not ready condition develops.

Status

Station Ready (XXX1)

The device connected is ready if its associated Power switch is on and the Ready light is on.

Station Busy (XXX2)

Indicates that either a punch or read operation is in process.

Punch Tape Supply Low (XXX4)

If the punch is connected, this bit is present if the quantity of remaining paper tape is less than may be necessary for a medium output.

Reader Last Device (XX1X)

Reader was the last equipment performing an operation.

Interrupt – Ready and Not Busy (X2XX)

This bit indicates that the interrupt has occurred due to the ready and not busy condition.

Interrupt – End of Operation (X4XX)

This bit indicates that the interrupt was generated due to the code 0022 selection and the end of operation.

Interrupt – Abnormal End of Operation (1XXX)

This bit indicates that the interrupt was generated due to the code 0024 selection and the abnormal end of operation.

PROGRAMMING NOTES

Reader

Read, Data, and Channel Busy signals from the data channel pull the reader clutch. A Data signal must reappear no less than 0.8 ms after a Reply signal to avoid dropping and repicking the clutch. Dropping the Read signal will drop the clutch.

Punch

The punch motor is activated by a combination of Write, Data, and Channel Active signals. Punching begins three seconds after activation and will continue without time loss if punch is reselected within three seconds after a punch cycle.

INTRODUCTION TO CHAPTER X

The 3400 system may use either the 342X or the 362X magnetic tape controller. The 342X controller has the following capabilities:

3421 Two Read/Write controls to control a maximum of four 604/607 magnetic tape units.

3422 Two Read/Write controls to control a maximum of six 604/607 magnetic tape units.

3423 Two Read/Write controls to control a maximum of eight 604/607 magnetic tape units.

The following information in chapter X discusses the 362X magnetic tape controller. The 342X is identical to the 362X except that the 362X can have up to four controls and can handle up to sixteen tape units. Bits 03-05 of the 3400 Connect instruction should all equal "0" when a 342X tape controller is used.

The 604 tape handler has all the capabilities of the 607 tape handler with the exception that tape speed during Read/Write operations is one-half that of the 607.

CHAPTER X
362X MAGNETIC TAPE CONTROLLER

The 362X tape controller connects and handles computer operations on 606 or 607 tape units. It has capabilities for handling up to four bi-directional input/output data channels. Each channel may communicate with any one of the 16 (maximum) tape units if the desired unit is not in use or reserved by some other channel. Four channels may communicate with four different tape units simultaneously.

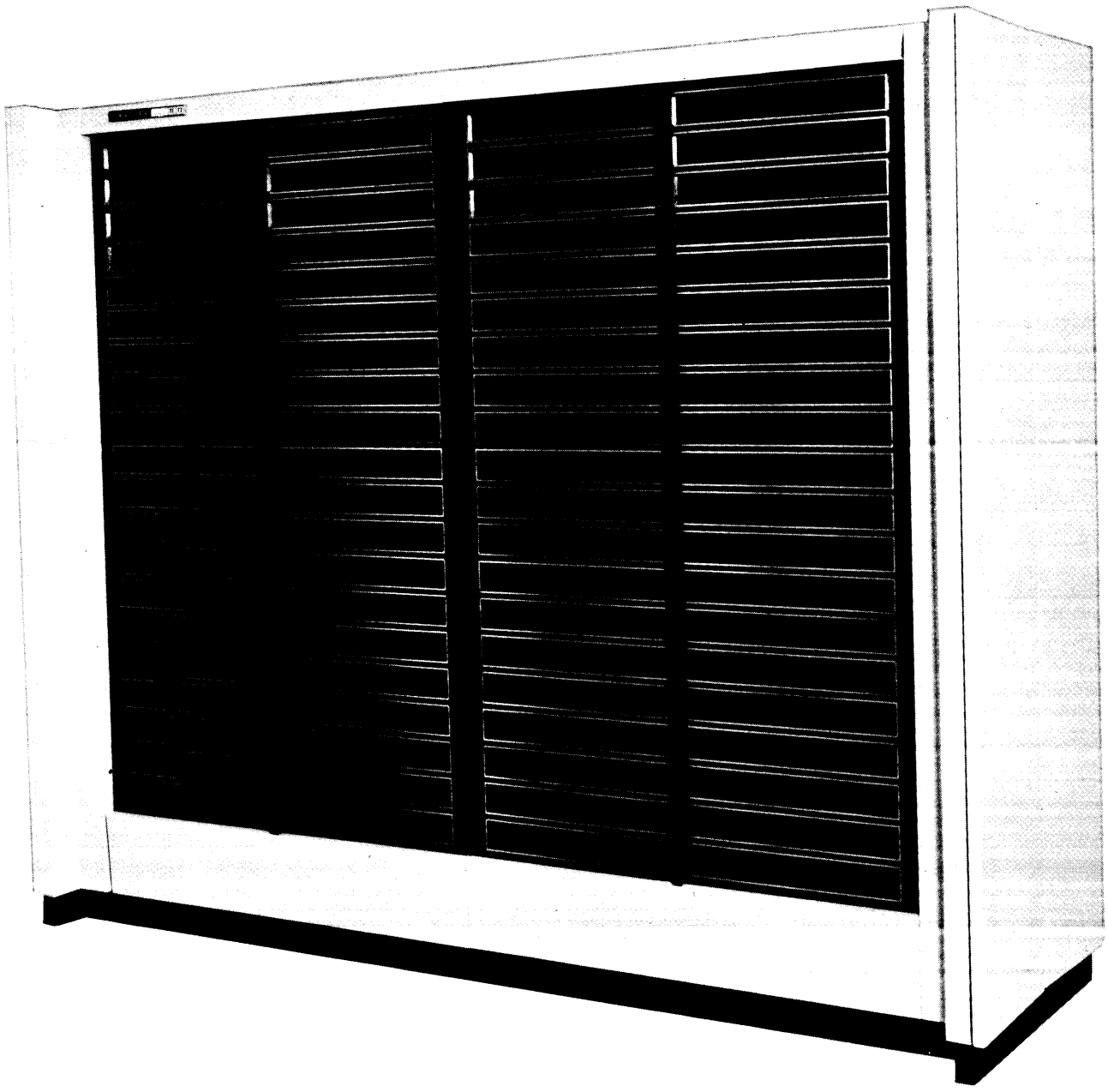


Figure 10-1. 362X Cabinet

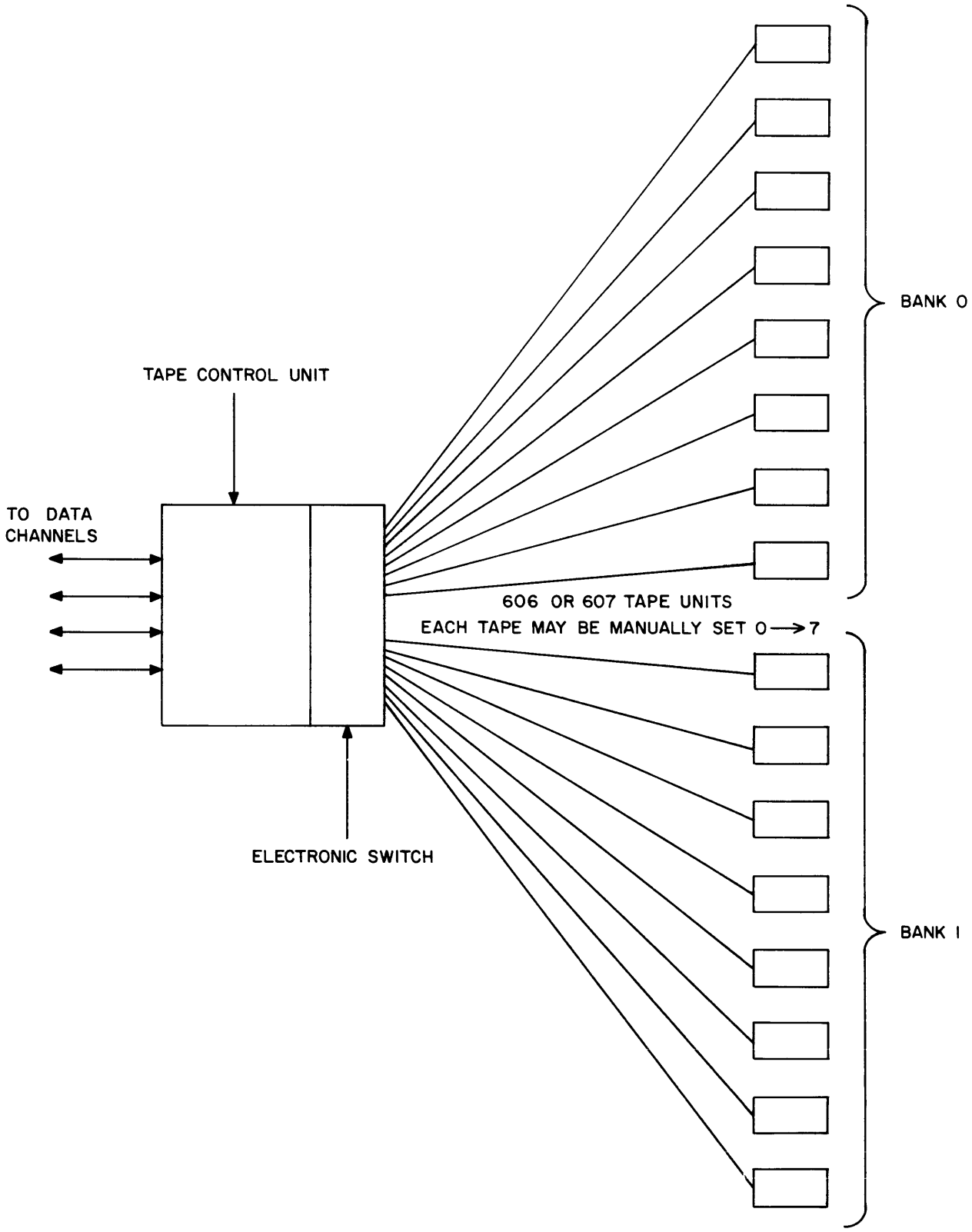


Figure 10-2. 362X Magnetic Tape Controller

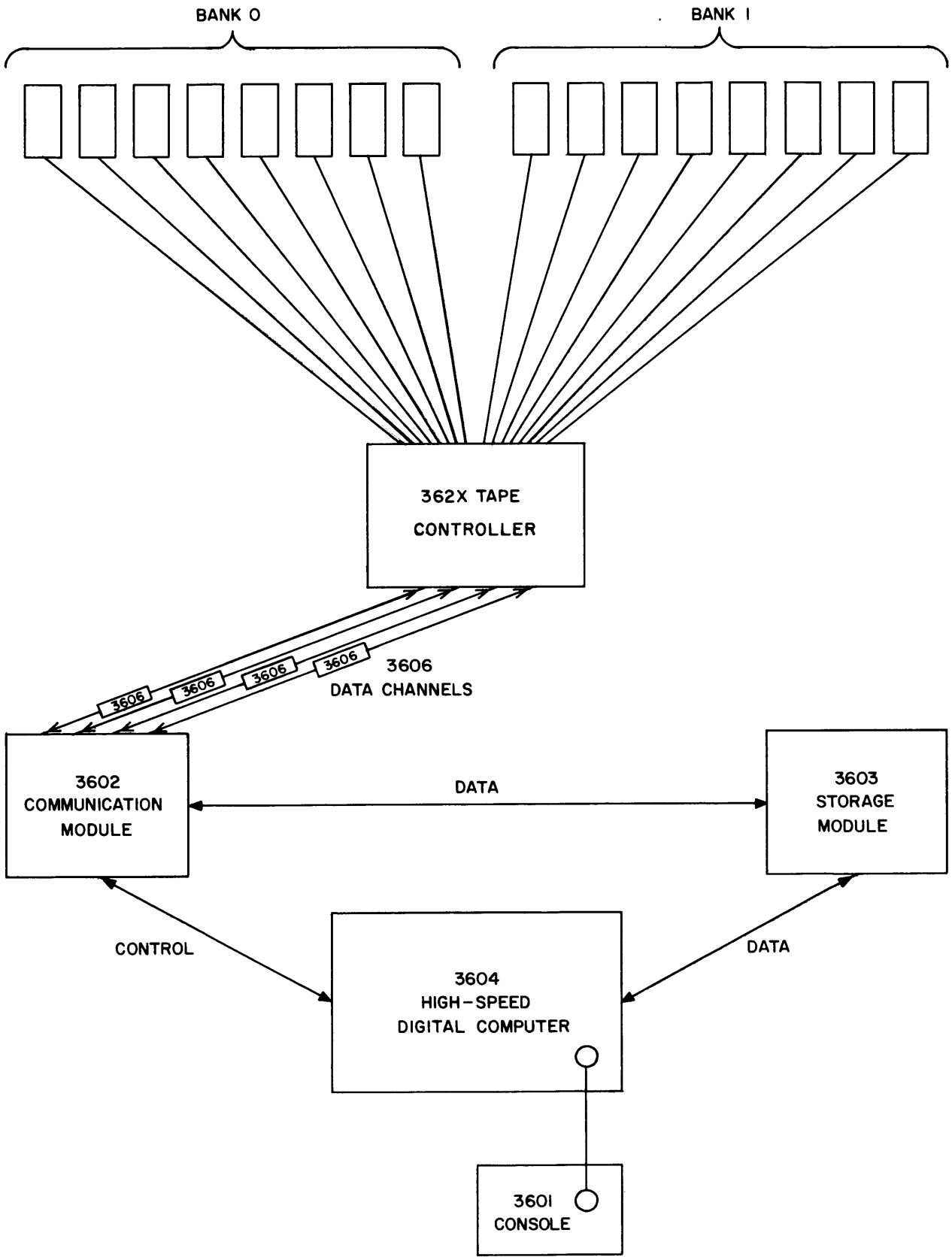


Figure 10-3. 362X as Related to the Basic 3600 System

UNIT SELECTION

Tape units physically attached to the 362X may be in one of three conditions:

1) Connected to a data channel

2) Reserved by a data channel

3) Unconnected and unreserved

Each of the four channel controls contains unit-connect and unit-reserve logic for each 606 or 607 tape unit. Once a channel connects a particular tape unit (i.e., establishes a communication path) that unit remains reserved for that channel, even though the channel connects another tape unit for some operation. No other channel has access to the reserved tape unit until the channel reserve for the tape unit is cleared. An unconnected and unreserved tape unit may be connected and reserved by any channel.

It is possible for a channel to reserve all tape units so no other channel may have access to them.

After a channel connects a tape unit and reserves it, any other connections that channel makes automatically clear the previous unit connection, but not the reservation. If more than one channel attempts a connection simultaneously, a scanner sequentially processes these attempts.

The 16 tape units are divided into banks 0 and 1, and each tape unit is manually set at any number from 0-7 in each bank. (Tapes in bank 0 are referred to as 00-07; bank 1 tapes are referred to as 10-17.) No more than one tape in each bank may be set to any given number. When the setting on a tape unit is changed, the connection and reservation is cleared for that tape unit.

The 362X may be connected to a maximum of four 12-bit data channels from any source. These may be:

1) 3606 data channel.

2) 3681 data channel converter from the 160 or 160-A.

3) Channels with the same characteristics from other equipments.

All four channels do not necessarily have to come from the same equipment.

SWITCHES AND INDICATORS

Equipment Number Switch

An eight-position (0-7) Equipment Number switch is associated with each control. The setting of this switch designates the control as equipment number N. Any interrupts coming from the control will be transmitted on one of the eight interrupt lines corresponding to the setting of the Equipment Number switch.

When a control is connected to a tape unit, a white indicator in the switch lights. (This indicator also lights when a connection cannot be made because the tape unit is reserved for another control.)

If a transmission parity error occurs during a Function, Read, or Write operation, a red indicator in the Equipment Number switch lights.

Longitudinal Parity

Seven Longitudinal Parity indicators are associated with each control. At the end of an operation involving longitudinal parity checking, none of these indicators should be lit. If one or more are lit, it indicates a longitudinal parity error has occurred.

Write

The Write indicator is illuminated during Write and Write End of File Mark operations. The Write indicator remains on until the Write operation terminates.

Vertical (VERT)

A Vertical Parity Error indicator lights if a vertical parity error occurs during an operation. This light is illuminated until a new record is begun.

Interrupt (INT)

This indicator lights when interrupt occurs. This light is illuminated until the Interrupt signal drops.

BCD

This indicator lights when BCD mode is selected or an end of file mark is written on tape.

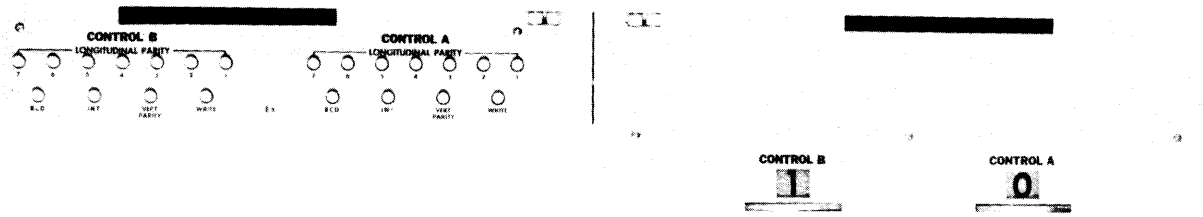


Figure 10-4. 362X Switch and Indicator Panel

DATA TRANSMISSION

Operations on tape include: Read, Reverse Read (607 only), Write, Write End of Record, Write File Mark, Search File Mark Forward or Backward, Backspace, Skip Bad Spot, Rewind, and Rewind Unload. A complete list of Function and Status Reply codes is given in table 10-1.

All information is transmitted between the tape controller and the data channel in odd parity, making a total of 12 information bits plus a parity bit. Information is written on or read from tape in odd transverse (vertical) parity for binary, even transverse parity for BCD. The end of record check character makes the total number of "1" bits in each of the seven longitudinal tracks even. Read or Write operations can be performed at any one of three densities: 200 characters/inch, 33 μsec/frame; 556 characters/inch, 12 μsec/frame; 800 characters/inch, 8 μsec/frame (607 only). Density and binary or BCD formats should be chosen or checked on the status lines before a Read or Write operation.

During a Write operation in BCD mode, the 362X automatically changes any word of all "0's" and writes it as 12_8 on tape. When reading a 12_8 from tape in BCD mode, the 362X changes it to a word of all "0's". If the 5th bit is a "1", the 6th bit is complemented during Read and Write operations on tape (the 5th bit remains unchanged). This converts internal BCD codes to external BCD codes when writing and vice versa for reading.

If the 3604 is running in 1604 mode, the conversion from internal BCD to external BCD will not occur during a Write operation. Similarly, the conversion from external BCD to internal BCD will not occur during a Read operation.

The tape controller has an interrupt feature which allows any tape unit to interrupt the equipment which is communicating with it. The program may select to be interrupted on occurrence of the following conditions:

1) Ready and Not Busy
2) End of Operation
3) Abnormal End of Operation

In order to read, write, etc., on any tape unit, it is necessary to:

1) Clear the desired control.
2) Connect the tape unit.
3) Sense the status of the connected tape unit.

There may be times when only step 2 is necessary, but normally all three steps are essential.

Table 10-1. Function Codes and Status Replies

TAPE MOTION			
Rewind	0010	Write End of File Mark	0015
Rewind Unload	0011	Skip Bad Spot	0016
*Backspace	0012		
Search End of File Mark Forward	0013		
Search End of File Mark Backward	0014		

FORMAT			
Release	0000	Clear	0005
Binary	0001	800 BPI Density	0006
Coded	0002	Clear Reverse Read	0040
556 BPI Density	0003	Set Reverse Read	0041
200 BPI Density	0004		

INTERRUPT			
Interrupt on Ready and Not Busy	0020	Release Interrupt on End of Operation	0023
Release Interrupt on Ready and Not Busy	0021	Interrupt on Abnormal End of Operation	0024
Interrupt on End of Operation	0022	Release Interrupt on Abnormal End of Operation	0025

STATUS REPLIES			
XXX1	Ready	X1XX	Density ("1" in bit 6 indicates 556 BPI, "0" in bit 6 indicates 200 BPI)
XXX2	[Channel (and/or) Control (and/or) Unit] Busy	X2XX	Density ("1" in bit 7 indicates 800 BPI)
XXX4	Write Enable	X4XX	Lost Data
XX1X	File Mark	1XXX	Longitudinal Parity Error
XX2X	Load Point	2XXX	Vertical Parity Error
XX4X	End of Tape	4XXX	Reserve Reject

* If a Backspace operation is executed when Reverse Read is set, tape is moved in a forward direction.

CODES

Clear

Prior to the initial use of the tape controller, the system should be cleared. There are five possible ways of clearing the controller:

1) Clear Channel (100 μsec)
   This instruction: a. Clears all activity in the data channel.

   b. Clears the present connection control X may have with a tape unit.

   c. Releases all tapes (i.e., clears reserve logic) which control X may have reserved.

   d. Performs a MC on control X Read, Write, and Function logic. (No status signals are available to the data channel after executing this instruction.)

2) Clear (2 μsec)
   This instruction: a. Clears the present connection control X may have with a tape unit. Control X remains "connected" in the sense that Status signals are still available for the data channel.

   b. Releases all tape units control X may have reserved.

3) Release
   This instruction clears only the connection and the reservation for the connected tape unit. It does not clear the reservations for any other tape units.

The latter two Function instructions (Clear and Release) can only be used after a control is connected to a tape unit.

4) Power On MC
   When power is applied to the 362X, all tape units connected and reserved by all controls are cleared. Logic in all controls is also cleared. No Status signals are available to the data channel after power is applied.

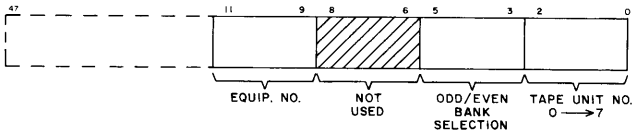
5) External MC
   This clears all tape units connected and reserved by all controls. It also clears the logic in all controls. No Status signals are available to the data channel after executing this operation.

All Clear operations (except the Release instruction) place the 362X in binary format.

Connect

The computer sends a 12-bit Connect code over a specified data channel to the tape controller which connects the desired unit.

Four manual eight-position switches on the tape controller (one for each channel) lock out all Connect codes except the ones having the correct bit combinations in bits 9, 10, and 11.



Bits 0, 1, and 2 may have octal values of 0-7. These bits determine which of the eight tape units the 362X will communicate with. Bits 3, 4, and 5 determine the bank (0 or 1) of tape units. The value held in bits 9, 10, and 11 must match the switch setting on the 362X or the connection will not be made. Bits 6, 7, and 8 are not used. If none of the controllers or any other equipments physically connected to the communication module via data channels have the proper switch setting, or a parity error occurs in the Connect code, an Internal Reject is generated within the computer.

Status

After the desired tape unit has been connected, it is usually necessary to check the status of the connected tape unit before attempting any further operations. This is done with the Copy Status and Bit Sensing instructions.

The 48-bit Copy Status instruction can:

1) Place the 12-bit Status code from the 362X in B^b.

2) Place the control word address in the lower order 18 bits of the Q register.

3) Place the control word in the A register.

The 48-bit Bit Sensing instruction allows the programmer to check the registers in which the Copy Status instruction has placed the information described above. A complete list of Status Reply codes is given in table 10-1. For example: The presence of bit 6 in the Status Reply code indicates that the tape unit is set to operate in 556 BPI density. The programming example on page 10-13 shows how the Bit Sensing instruction is used. This instruction can also examine the current word count (part of the control word placed in A by the Copy Status instruction) to determine the progress of information transfer.

Status Definitions

Ready (XXX1)

A Ready indicator on the tape unit lights when it is in a ready condition; i.e., power has been applied and the tape unit is in Automatic mode. When in Automatic mode, the tape unit is controlled by the tape controller.

The Ready signal is not present when an operator manually controls the 606 or 607 from the console of the tape unit.

File Mark (XXIX)

Non-Chaining operations: This signal is present when the tape unit has searched for and located an end of file mark. It is also present immediately after writing an end of file mark. Signal drops when:

1) reading/writing begins on a new record, or

2) a Backspace, Search End of File Mark Forward, or Search End of File Mark Backward operation is initiated.

Chaining operations: If the tape unit "chains over" a file mark during a chaining Read operation, the File Mark signal will appear. This signal will remain up throughout the remainder of the chain operation. This signal drops when:

1) reading/writing begins on a new record (new chaining/non-chaining operation), or

2) a Backspace, Search End of File Mark Forward, or Search End of File Mark Backward operation is initiated, or

3) a MC is executed.

Write Enable (XXX4)

This signal is present only when the file protection ring is on the tape reel. When this signal is absent, it is impossible to write on tape, although information may be read from the tape.

[Channel (and/or) Control (and/or) Unit] Busy (XXX2)

If the tape unit is ready, this signal is present:

1) During and 5 ms after any operation which results in tape motion (Read, Write, etc.).

2) When the data channel begins executing or is executing a Read/Write instruction.

This signal will not be present if:

1) The tape unit is not ready.

2) The channel begins executing or is executing a Read/Write instruction and/or:

   a) Lost data has occurred in a previous operation.

   b) Interrupt On Abnormal End of Operation has occurred in a previous operation and the Interrupt signal is still present.

Load Point (XX2X)

This signal is present when the tape is at load point. The signal drops when tape motion begins again.

End of Tape (XX4X)

This signal is present when the end of tape marker is detected. The signal drops when tape has been rewound past the end of tape marker; i.e., the end of tape marker is sensed during rewind.

Density (X1XX)

See table 10-1.

Density (X2XX)

See table 10-1.

Lost Data (X4XX)

This signal appears during a Write operation if the tape controller is ready to accept information but the Data signal from the 3606 is absent (Write signal present).

When the Lost Data signal appears during a Write operation, tape motion stops. Further Write operations are impossible until the Lost Data signal is cleared with a new Function or Connect code.

The Lost Data signal also appears during a Read operation when the tape controller has data ready for output, but the Data signal from the 3606 is absent (Read signal present).

If the Lost Data signal appears during a Read operation, reading continues until the end of the record. Further Read operations are impossible until the Lost Data signal is cleared with a new Function or Connect code. (Any legal Function code listed in table 10-1 will clear the Lost Data signal.)

The Lost Data signal is meaningless when the tape controller is attached to a 160/160-A via a 3681 adapter. However, this signal must be cleared if Read/Write operations are to continue.

Longitudinal Parity Error (1XXX)

This signal indicates that a longitudinal parity error has occurred during a Read/Write operation. This signal drops when a new record is read or written. A Clear Channel instruction, external master clear, or a power on master clear causes this signal to drop.

During a chaining operation, the Longitudinal Parity Error signal appears on a status line when an error has occurred during reading/writing a record. This signal will remain up during the remainder of the chaining operation. This signal drops when a new chaining/non-chaining Read/Write operation begins or one of the three types of master clears is executed.

Vertical Parity Error (2XXX)

This signal indicates a vertical (transverse) parity error has occurred during a Read/Write operation. This signal drops when reading begins on a new record. A Clear Channel instruction, external master clear or a power on master clear causes this signal to drop.

If a vertical parity error occurs during chaining, the Vertical Parity Error signal drops when reading/writing begins on a new record during the same chaining operation. The signal also drops when a new chaining/non-chaining Read/Write begins or one of the three types of master clears is executed.

The Vertical Parity Error signal also appears when an end of file mark is read in Binary mode.

Reserve Reject (4XXX)

This signal is present when the last Connect operation performed was rejected because the tape unit to be connected was reserved by another control.

Function and Format

The following information refers to the Function and Format codes used with the Function instruction; i.e., the lower 12 bits which specify the particular operation. An octal 0 in bits 3, 4, and 5 of the Function code indicates format, an octal 1 in bits 3, 4, and 5 indicates tape motion, and an octal 2 in bits 3, 4, and 5 indicates interrupt.

The proper tape unit must be connected before a Function instruction can be issued. If an error occurs in the Function code (assuming the proper unit is connected) a Parity Error signal appears on a transmission parity error line, and the computer issues an internal reject after 100 μsec. The program continues at a reject jump address.

Once a function (backspace, rewind, etc.) is initiated on a tape unit on a given channel, it is possible to connect another tape unit on that same channel, perform some operation on the second tape unit, and reconnect the first tape unit before or after the operation on the first tape unit terminates.

Function

Rewind (0010)

A 0010 code rewinds tape at high speed (225-400 inches/second) to load point. Any further Rewind instructions when tape is at load point will have no effect. A Load Point signal appears on a status line when the operation is complete.

Rewind Unload (0011)

A 0011 code rewinds tape at high speed until all the tape is on the supply reel. All further operations on this tape are locked out until the tape has been reloaded manually.

Backspace (0012)

A 0012 code backspaces tape one record length. If the load point occurs other than at the beginning of

the tape, this code will backspace tape from load point to one record length behind the load point. It will also backspace tape from an end of record check character back to load point if there are no other record check characters in between. Simultaneous backspace operations are possible on different units, but separate Connect and Backspace instructions are required for each.

Search End of File Mark Forward (0013)

Search End of File Mark Backward (0014)

A 0013/0014 code searches forward/reverse until an end of file mark is detected. A File Mark signal appears on a status line when the operation is complete. If no file marks are detected, tape motion continues until load point or end of tape markers are reached and then terminates.

Write End of File Mark (0015)

A 0015 code writes 17_8 as an end of file mark (even transverse parity) in both binary and BCD format. Writing an end of file mark does not change the current format.

Skip Bad Spot (0016)

A 0016 code moves and erases tape six inches in a forward direction. Vertical and longitudinal parity checking is performed during the operation to insure that tape is being erased. If a parity error occurs, a Vertical/Longitudinal Parity Error signal will appear on a status line when the operation is complete.

If Interrupt on End of Operation is selected, interrupt will occur when the Skip Bad Spot operation is complete.

All previously mentioned codes (0010-0016) result in a reject if attempted when the read/write control is busy.

Format

Release (0000)

A Release code clears the existing unit connection and reserve logic for a tape unit. It does not clear the reserve logic on any other tape units reserved by the control.

Binary (0001)

A 0001 code allows all information to be written/read in binary notation. A parity generator makes the total number of "1" bits odd in the transverse (vertical) direction on a total of seven separate tracks on tape. The end of record check character makes the longitudinal number of bits in each of the seven tracks even. During Read or Write operations, a constant transverse parity check is made. A parity error is indicated on a status line if a vertical parity error is detected. When the end of record check character is written/read, a longitudinal parity error check is performed.

Coded (0002)

A 0002 code allows all information to be written/read in binary coded decimal notation. A parity generator makes the total number of "1" bits even in the transverse (vertical) direction on the seven tracks on tape. Constant parity checks during Read/Write operations are performed in the same manner as in binary format.

556 BPI Density (0003)

200 BPI Density (0004)

A 0003/0004 code permits all information to be written or read onto or from tape at 556/200 density (556 density: 556 6-bit characters + parity bit/inch; 200 density: 200 6-bit characters + parity bit/inch).

Clear (0005)

A 0005 code clears all the tape unit reservations a channel has made and the existing unit connection. It is desirable to issue this code when a channel has completed all operations on one or more tape units. This permits other channels to gain access to these units.

800 BPI Density (0006)

A 0006 code permits all information to be written or read onto or from tape at 800 density: 800 6-bit characters + parity bit/inch.

Set Reverse Read (0041)

This code is used for a reverse Read operation. (See Reverse Read section.)

Clear Reverse Read (0040)

This code clears the condition established by the 0041 Format code.

All of the Format codes (0000-0006, 0040, 0041) result in a reject if attempted when the read/write control is busy.

Interrupt

All desired interrupt instructions must come before a Read or Write operation, but can occur during any other operation. The eight-position Equipment Selection switch determines which line the Interrupt signal is transmitted on. For example: If the Equipment Selection switch on control X is set to 5, any interrupts coming from control X will be transmitted on interrupt line 5. Any new interrupt instruction clears the existing Interrupt signal. This signal is also cleared by releasing all three possible interrupts, or doing a master clear.

Interrupt on Ready and Not Busy (0020)
Release Interrupt on Ready and Not Busy (0021)

The 0020 code allows a tape unit to send an Interrupt signal out on a channel when this tape unit is in a ready and not busy condition; i.e., when power is applied, the unit is in Automatic mode, and all tape motion has ceased. Release Interrupt on Ready and Not Busy code 0021 clears this condition.

Interrupt on End of Operation (0022)
Release Interrupt on End of Operation (0023)

A 0022 code allows a tape unit to send an Interrupt signal out on a channel approximately 200 μsec after an end of record check character is read by a tape unit, a file mark has been located in a search file mark forward or backward operation, load point has been detected during a rewind operation or a Skip Bad Spot operation has been completed. Release Interrupt on End of Operation code 0023 clears this condition.

During a chaining operation, interrupt will not occur until the data channel is not busy; i.e. until the last record has been written/read in the chaining operation.

Interrupt on Abnormal End of Operation (0024)
Release Interrupt on Abnormal End of Operation (0025)

A 0024 code allows a tape unit to send an Interrupt signal out on a channel after an abnormal operation occurs. These abnormal operations are:

End of Tape, File Mark, Load Point, Vertical Parity Error, Longitudinal Parity Error, Lost Data, and Connected Tape Unit Becoming Not Ready. In all but the last case, the interrupt occurs when one or more of these conditions is encountered and an end of record check character is written/read by the tape unit. In the case of interrupt on Connected Tape Unit Becoming Not Ready, interrupt occurs immediately when the connected tape unit goes from a Ready to a Not Ready condition (e.g. if the power is turned off on the tape unit, etc.). Interrupt on Connected Tape Unit Becoming Not Ready will not occur during:

1) a Connect operation; or
2) when an 0000 Function instruction (Release Connected Unit) is being executed.

During a chaining operation, interrupt will occur at the same time as in a non-chaining operation if one or more abnormal condition arises as described previously.

A new Read/Write operation cannot start until the Interrupt signal is cleared by one of the methods mentioned previously in this chapter under the Interrupt section. Release Interrupt on Abnormal End of Operation code 0025 clears this condition.

If Interrupt on Abnormal End of Operation is selected, chaining is selected, and an interrupt condition occurs during reading a record, the Interrupt signal will be returned to the 3606 when the end of record is read. When the interrupt occurs, the data channel will wait after receiving the last reply.

The 3604 will enter an interrupt routine (see the Interrupt chapter in this manual) and process the interrupt. A Stop Channel Activity instruction (in the 3604 interrupt routine) will terminate data channel activity and store the present word count, etc. When the Interrupt signal is cleared (in the 3604 interrupt routine), reading may be initiated by a new Read instruction.

If the new Read signal is returned to the tape controller 45 μsec or less after the interrupt occured, the Read operation will be non-stop. If the time exceeds 45 μsec, tape motion will stop and restart. If no Read instruction is executed by the 3604 after interrupt, tape motion will halt.

If chaining is selected and an interrupt condition occurs during writing a record, the Interrupt signal will be returned to the 3606 when the record is completed. When the interrupt occurs, the data channel will wait after receiving the last reply.

The 3604 will enter an interrupt routine (see the Interrupt chapter in this manual) and process the interrupt. A Stop Channel Activity instruction (in the 3604 interrupt routine) will terminate data channel

activity and store the present word count, etc. When the Interrupt signal is cleared (in the 3604 interrupt routine), writing may be initiated by a new Write instruction.

If the new Write signal is returned to the tape controller 2.5 ms or less after interrupt occurred, the Write operation will be non-stop. If the time exceeds 2.5 ms, tape motion will stop and restart. If no Write instruction is executed by the 3604 after interrupt, tape motion will halt.

WRITE

After the 606 or 607 has been connected and format chosen, the programmer should check status for a Write Enable (bit 2 in the Status Reply code). If this signal is not present, it indicates that the protective ring is missing from the tape reel. It is possible to write on the tape only when this condition no longer exists. (If the Write Enable is not present, the 3606 hangs up.) If the Write Enable is present, accompanied by the Ready signal, the Write operation may begin. For information on the eight available options when the word count is reduced to 0, refer to the Control Word, Input/Output chapter.)

If the Write signal comes up within 2.5 ms after the last Write operation, a non-stop Write is initiated. The total delay before actual writing on tape can begin again is approximately 4 ms, versus a normal delay of approximately 7 ms.

READ

In a Read operation, the order of events is similar to the Write operation. After connecting, checking status, and choosing the proper format, the Read operation begins. It is not necessary to check for a Write Enable since Read operations from tape are possible when the protective ring is not present on the tape reel. The non-stop Read is similar to the non-stop Write operation. (For information on the eight available options when the word count is reduced to 0, refer to the Control Word, Input/Output chapter.)

During Read and Write operations, the programmer may choose to check status periodically. He may also program one or more interrupts to let the computer know when the present operation is complete.

Reverse Read (607 Tape Units Only)

The 362X can read information in a reverse direction from tape. Six-bit frames are read from tape and assembled into 12-bit bytes and sent to the data channel. When a 48-bit word is read in a reverse direction from tape and entered into storage, it is identical to the 48-bit word which was initially written on the tape from storage. There is no change made in the final order of the bits during a Reverse Read operation.

To initiate a Reverse Read operation (assuming format, etc. have already been selected and all tape motion has stopped), a 362X controller must first receive the Function code 0041 (Reverse). When the Read instruction is executed in the 3604, the Reverse Read operation will begin; i.e., data will be available to the data channel. A Reverse Assembly signal will be sent to the data channel from the 362X to indicate that the 12-bit bytes should be assembled into a 48-bit word in reverse order.

Data transfer will continue until the word count in the control word equals zero. Tape motion continues in a reverse direction and stops at the gap between the current record and the record check character of the next record (unless chaining or non-stop read has been selected).

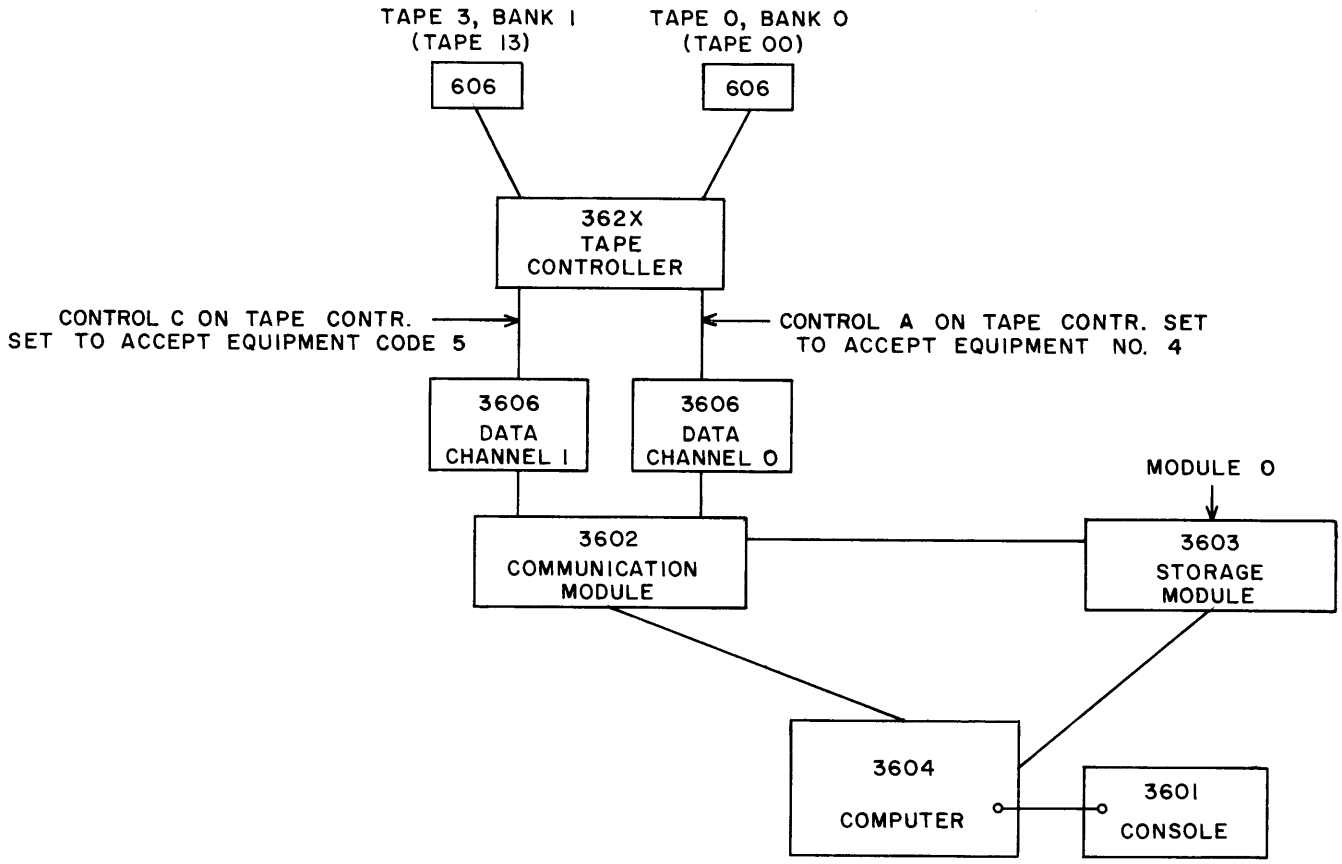
Vertical and longitudinal parity checking occur as in a normal Read operation with one exception: if the first frame read is a record check character, no vertical parity check is made on that character. Vertical parity checking is performed on all remaining frames.

If a Reverse Read is attempted from load point, there will be no tape motion. The Read operation will halt indefinitely.

Parity errors and interrupts may be handled as if the operation were a normal read. An End-of-Record signal is returned to the data channel when a record gap is reached. Chaining and non-stop read operations are also handled as if the operation were a normal read.

The Function code 0040 (Clear Reverse) should be issued when the Reverse Read operation terminates.

PROGRAMMING EXAMPLE



100_8 words are stored in storage module 0 at locations 00100-00177. Write them on tape 0, bank 0 (556 density, BCD). Tape 00 is at load point. Also, a record of 300_8 words is stored on tape 13 (written in BCD, 556 density). Store the record in storage module 0, starting at location 00200. Tape 13 is also at load point. Rewind both tapes to load point sometime after the Write and Read operations are complete. Halt the program if any of the above operations cannot be executed.

Do console MC.
Start program at address 10000.

Storage Address	Contents of Address	
10000	745-----00------	CLEAR CHANNEL (Channel 0, Comm. Mod. 0)
10001	7402000000	CONNECT Channel 0 Comm. Mod. 0 to Tape 00
10002	74411---00------	COPY STATUS on Tape 00, Load in B^1
10003	BIT SENSE	Sense bit 2 of Status code in B^1 for Write Enable
10004	76000000	HALT
10005	BIT SENSE	Sense bit 0 of Status code in B^1 for Ready
10006	76000000	HALT
10007	7411000400--0002	SET BCD MODE
10010	7411000400--0003	SET 556 DENSITY
10011	7431000400030000	WRITE (Contains control word address)
10012	745-----01------	CLEAR CHANNEL (Channel 1, Comm. Mode. 0)
10013	7402000001--5013	CONNECT Channel 1 Comm. Mod. 0 to Tape 13
10014	74431---01------	COPY STATUS on Tape 13, Load in B^3
10015	BIT SENSE	Sense bit 0 of Status code in B^3 for Ready
10016	76000000	HALT
10017	7411001601--0002	SET BCD MODE
10020	7411001601--0003	SET 556 DENSITY
10021	7421001601030001	READ (Contains control word address)
	Continue with main program. Tapes 00 and 13 remain connected.	
20000	76000000	HALT
30000	10000100--000100	CONTROL WORD (Write operation)
30001	10000300--000200	CONTROL WORD (Read operation)
30004	7413000400--0010	REWIND Tape 00
30005	7413000501--0010	REWIND Tape 13
30006	76000000	HALT (End of program)

Do a console MC. This clears all registers needed in this example. The program starts at address 10000 with a 48-bit Clear Channel instruction. This clears all equipment connected to communication module 0, data channel 0. The Connect instruction (address 10001) connects tape unit 00 to communication module 0, data channel 0. (The switch on control A on the tape controller is set to accept equipment #4.) The Copy Status instruction (address 10003) loads the 12-bit Status code into B^1. A Bit Sense instruction (address 10003) senses bit 2 of the Status code for a Write Enable. (If this signal is not present when attempting a Write operation it indicates that the protective ring is not present on the tape reel.) Another Bit Sense instruction (address 10005) senses bit 0 of the Status code for a Ready signal. The program halts if either a Write Enable or a Ready signal is not present.

The two Function instructions (addresses 10007, 10010) set the control A logic in the tape controller to BCD mode and 556 density. When the computer executes the Write instruction (address 10011) it sends the 18-bit control word address to the communication module. The 3602 reads the 48-bit control word from address 30000 and initiates output from storage module 0 to tape 00.

The main program continues at address 10012 when the control word address has been transmitted to the communication module. The same procedure as above follows for tape 13, except that the operation is a Read from Tape 13 on data channel 1, and there is no need to sense for a Write Enable. The Set BCD Mode instruction is necessary since the Clear Channel instruction has set control C logic in the tape controller to the Binary mode. The switch on the 362X on control C is set to accept equipment code 5.

When the computer reaches address 10021, it executes the Read instruction and sends the 18-bit control word address to the communication module. Then the communication module begins the Read operation, and the main program continues.

When the program reaches address 30004, it sends the 12-bit Rewind instruction (0010) to tape 00. If tape 00 is busy (indicated by a Reject signal in this case) the program halts at this step and executes reject jumps to address 30004. When the Reply is returned (indicating tape 00 is in a rewind condition) the program continues at address 30005. When the instruction to rewind tape 13 is completed, the program stops at address 30006.

INTRODUCTION TO CHAPTER XI

Since the 3406 input/output channel and the 3606 data channel have identical operating characteristics, the 3681 may be used with the 160/160-A to communicate with 3400-type external equipments.

CHAPTER XI
3681 DATA CHANNEL CONVERTER

The 3681 data channel converter simulates a 3606 data channel, allowing a 160/160-A computer to communicate with 3600-type external equipments. The 160/160-A may perform Connect, Function, Read, Write, and Status operations on these equipments via the 3681.

The 160/160-A transmits codes to the 3681 prior to starting any operation on an external equipment (see table 11-1 for 3681 Select and Status codes). These codes establish conditions in the 3681 such that the proper 3600-type signals accompany 160/160-A Input/Output operations. (These 3600-type signals are Connect, Function, Data, Read and Write.)

A parity bit is added to the 12 bits coming from the 160/160-A when exchanging information with external equipments (odd transverse parity on transmission).

Two 3681 adapters can be run from one 160/160-A by using Select codes 6XXX or 5XXX.

SWITCHES

A Code Selection switch determines which Select codes the 3681 will recognize; e.g., if the switch is set to position 5, the 3681 responds to 5XXX codes; in position 6, it responds to 6XXX codes. Interrupt 40 corresponds to position 6, interrupt 30 to position 5.

NON-SATELLITE/SATELLITE SYSTEMS

Non-Satellite

Figure 11-1 shows two possible configurations. The 160/160-A can communicate with up to eight 3600-type equipments such as the 362 tape controller, card reader, card punch, etc. A second adapter can be connected to additional 3600-type devices.

The 160-A will respond to interrupts from the external equipments.

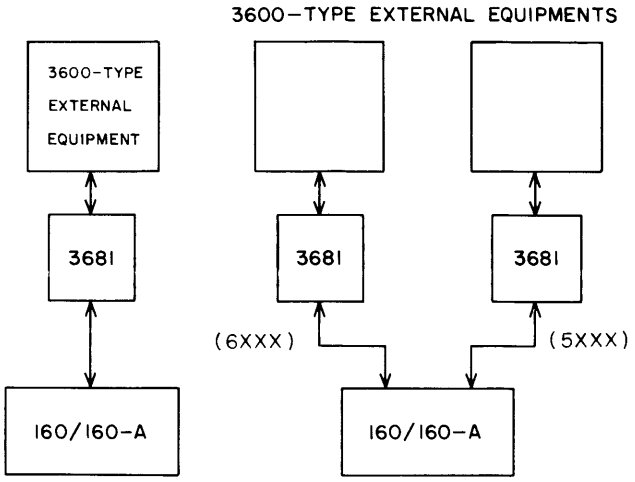


Figure 11-1. Non-Satellite System

Satellite

Figure 11-2 shows the 3681 and 160-A as related to a satellite system. See the 3682 chapter in this reference manual for detailed satellite information and programming.

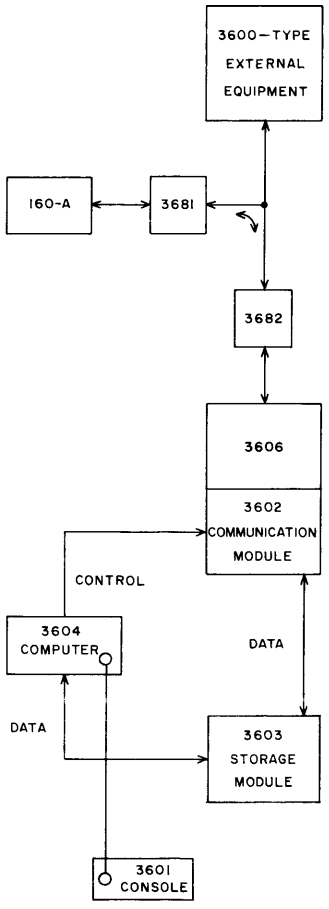


Figure 11-2. Satellite System

Table 11-1. Select and Status Codes

160/160-A Select Codes	
6001 Channel Status	Gates the channel status information. An Input instruction in the 160/160-A transfers this information to the computer where it can be checked to determine the current conditions of the external equipment (see Channel Status codes).
6002 Equipment Status	Gates the Status Reply codes. An Input instruction transfers this information to the 160/160-A where it can be checked. See the equipment reference manuals for lists of Status Reply codes.
6003 Master Clear	This code is similar to the 3604 Clear Channel instruction. When issued, the 3681 holds a 100 μsec master clear on all 3600-type equipments attached to it.
6004 Connect Initiate	Must be sent out to the 3681 prior to a Connect code. The Connect Initiate generates a pseudo Connect signal when the Connect code is transmitted to the external equipment.
6010 Function Initiate	Must be transmitted to the 3681 before a Function code. This code generates a pseudo Function signal when the Function code is transmitted to the external equipment. See the equipment reference manuals for Function codes.
6020 Read	Must be transmitted to the 3681 prior to performing an Input operation from a 3600-type external equipment. This code provides the pseudo Read signal needed by the 3600-type equipment.
6040 Write	Identical to the 6020 code except that it provides the Write signal during output. It must be transmitted to the 3681 prior to a Write operation.
6200 Clear Functions	Clears all functions but does not clear an interrupt in the 3681.
6400 Clear Functions and Interrupt	Clears all functions and an interrupt in the 3681 caused by a reject.

Channel Status Codes			
XXX1	Reject *	XX4X	Interrupt Equipment #3
XXX2	Transmission Parity Error	X1XX	Interrupt Equipment #4
XXX4	Interrupt Equipment #0	X2XX	Interrupt Equipment #5
XX1X	Interrupt Equipment #1	X4XX	Interrupt Equipment #6
XX2X	Interrupt Equipment #2	1XXX	Interrupt Equipment #7

* A transmission parity error, a reject, or any of the eight interrupt lines will interrupt the 160-A.

PROGRAMMING

The order of events in communicating with external equipment through the 3681 is:

1) Connect the desired equipment.
2) Check status of the connected equipment.
3) Perform the desired operation(s).
4) Clear the 3681 Read or Write by a new function or a 6200 Clear Functions code.

Connection

Before using an equipment, it must be connected by the proper Connect code and Connect signal. This is done with a 75XX EXF instruction in the 160/160-A which sends a 12-bit Connect Initiate code to the 3681 adapter. (This code provides a pseudo Connect signal when the 3681 transmits the 12-bit Connect code to the external equipment.) An Output instruction from the 160/160-A following the EXF code transmits the Connect code to the external equipment.

It is necessary to check for a reply or reject each time when:

1) Connecting a 3600-type external equipment.
2) Executing a function on a 3600-type external equipment.

Status

160

A reply or reject from the external equipment will cause the program to continue in the 160. It is therefore necessary to check the status of the equipment referred to by a Channel Status when a resume or reject is involved. If neither a reply or reject is returned by the external equipment, the 160 program halts.

160-A

A transmission parity error, a reject, or any one of the eight interrupt lines will enable an interrupt to the 160-A, provided the interrupt lockout is not in effect. (The 75XX EXF instruction sets the interrupt lockout which remains set until one instruction after a 0120(CIL) instruction is executed.)

A Channel Status code (6001) gates the current channel status to the 160-A where it can be examined to determine which condition(s) caused the interrupt.

If neither a reply or reject is returned by the external equipment, the 160-A program halts.

Channel Status (6001)

A 75XX EXF instruction transmits this code to the 3681. A following Input to A instruction transfers the channel status information into the 160/160-A where it can be checked (see table 11-1 for Channel Status codes.)

Equipment Status (6002)

A 75XX EXF instruction sends this code to the adapter. A 7600 INA instruction transfers the Status code to the 160/160-A (see the equipment reference manuals for Status Reply codes.)

Function

A 75XX EXF instruction in the 160/160-A must precede executing function(s) on the external equipment. This instruction sends the 6010 Function Initiate code to the 3681. An Output instruction from the 160/160-A then transmits any number of Function codes to the external equipment.

It should be noted that a separate EXF instruction in the 160/160-A is required when more than one code is sent only to the 3681 (i.e., Connect, Initiate, Equipment Status, etc.). The 75XX EXF instruction generates its own Output Resume, but I/O instructions depend on the external equipment for resumes. For example, it is not possible to send codes which first set the external equipment to some condition for a Read operation and have the same instruction send the code (6020) for the Read operation. The code for setting the read will not generate an Output Resume, and the 3681 will halt indefinitely.

Any new EXF instruction clears all previous selections in the adapter.

Read (6020)

Before doing an Input operation, it is necessary to use the 75XX EXF instruction to transmit the 6020 code to the 3681. An Input instruction (72XX) executes the Read operation. If an end of record is read while the read command is still in effect (3681 in Non-Satellite mode), or a Busy signal is not returned 25 μsec after the Read signal is sent to the external equipment, an Input Disconnect is sent to the 160/160-A. This terminates the Input operation. Clear the Read code with a new function or a master clean when the operation is complete (the Read signal remains up until cleared and may cause difficulties if not cleared).

Write (6040)

Write is similar to the Read operation. The 75XX EXF instruction sends the 6040 code to the 3681. When the Output Resume is returned to the computer, a 73XX instruction executes the Output operation. Clear the Write code when the operation is complete.

Word Mark (Output)

A Word Mark signal accompanies the fourth 12-bit byte of information sent to the external equipment. This signal comes up with the fourth data signal and drops when the data signal drops.

Word Mark (Input)

A Word Mark signal is sent to the external equipment when the data signal requests the fourth 12-bit byte of information. This signal comes up with the fourth data signal and drops when the data signal drops.

Programming Example

A 160-A is connected to a 3681. The switch on the converter is set to position 6 (6XXX codes). The converter is connected to control A on a 362X tape controller (control A is set to accept equipment 0). A 606 tape unit (bank 1, tape 0 = tape 10) is connected to the controller.

100_8 words are stored in locations 1300, 1377 in the 160-A. Write them on tape 10 in 556 density, BCD. Tape 10 is at load point. Rewind when the Write operation is complete. Halt the program if any steps in the above operation are not completed or an Interrupt is generated. Refer to the 160-A Programming Manual (publication 145f) for a detailed discussion of instructions used in this program.

- Set direct, indirect, and relative bank controls to 0.
- Do MC.
- Start program at address 1000.

LOCATION	CONTENTS	COMMENTS	
1000	7500	EXF Constant	Connect controller
1001	6004	G = Connect Initiate	
1002	7366	Output (FWA = 1202)	
1003	1203	G = LWA + 1 = 1203	
1004	0120	Clear Interrupt Lockout	
1005	0001	No Operation	Check controller for Ready
1006	7500	EXF Constant	
1007	6002	G = Controller Status	
1010	7600	INA (12-bit Controller Status → A)	
1011	0201	LPN (Check for bit 0 = Ready)	
1012	6102	Non-Zero Jump Forward	
1013	7700	HALT if Ready Not Present	
1014	7600	INA (12-bit Controller Status → A)	Checks controller for Write Enable
1015	1005	LPD (Check for bit 2 = Write Enable)	
1016	6102	Non-Zero Jump Forward	
1017	7700	HALT if Write Enable Not Present	
1020	7500	EXF Constant	Set controller to BCD
1021	6010	G = Function Initiate	
1022	7347	Output (FWA = 1200)	
1023	1201	G = LWA + 1 = 1201	
1024	0120	Clear Interrupt Lockout	
1025	0001	No Operation	Set controller to 556 Density
1026	7345	Output (FWA = 1201)	
1027	1202	G = LWA + 1 = 1202	
1030	7500	EXF Constant	Write on Tape 10
1031	6040	G = Write	
1032	7346	Output (FWA = 1300)	
1033	1400	G = LWA + 1 = 1400	
1034	0120	Clear Interrupt Lockout	Check controller and wait for Not Busy
1035	0001	No Operation	
1036	7500	EXF Constant	
1037	6002	G = Controller Status	
1040	7600	INA (12-bit Controller Status→→ A)	
1041	0202	LPD (Check for bit 1 = 0)	
1042	6002	ZJF (Zero-Jump Forward)	
1043	6503	NZB (Non-Zero Jump Backward)	

Programming Example (cont'd)

LOCATION	CONTENTS	COMMENTS	
1044	0120	CIL (Clear Interrupt Lockout)	
1045	0001	No Operation	Rewind
1046	7500	EXF Constant	Tape 10
1047	6010	G = Function Initiate	to Load
1050	7323	Output (FWA = 1203)	Point
1051	1204	G = LWA + 1 = 1204	
1052	0120	CIL (Clear Interrupt Lockout)	
1053	0001	No Operation	Halt
1054	7700	Halt Program	
0005	0004	Logical Product Operand	
0040	----	Store P on Interrupt 40	
0041	7500	EXF Constant	
0042	6001	Channel Status	
0043	7600	INA (Channel Status → A)	
0044	7700	HALT	
1070	1202		
1071	1200		
1072	1201		
1073	1203		
1100	1300		
1101	----		
1200	0002	Set Controller to BCD	
1201	0003	Set Controller to 556 Density	
1202	0010	Connect Code (Control A, Tape 10)	
1203	0010	Rewind Tape 10	
1300	----		100_8 Words Stored in
1377	----		Locations 1300 → 1377

Program Explanation

Begin by doing a computer master clear and setting the direct, indirect, and relative bank controls to 0.

Address 1004 contains a Clear Interrupt Lockout instruction which enables an interrupt to halt the program one program step after doing a CIL. If an interrupt is generated during the program, the program address (P) at the time of the interrupt is stored at location 0040. The current channel status is displayed in the A register before halting the program.

Address 1006 contains another EXF instruction which sends the Equipment Status code to the 3681. This code makes the controller Status Reply available to the 160-A. The 7600 instruction at address 1010 reads the 12-bit Status Reply code into the A register in the 160-A. Logical Product instructions at 1011 and 1015 check the code for Ready and Write Enable signals.

The program starts at address 1000 with an EXF instruction. This transmits the 6004 Connect Initiate code at address 1001 to the 3681. The following Output instruction (7366) at address 1002 sends the Connect code at address 1202 to the tape controller. The 3602 makes the connection and returns a Reply, or it returns a Reject if it cannot make the connection.

Another EXF instruction at 1020 transmits the Function Initiate code (address 1021) to the adapter. The following Output instruction at address 1022 (7347) sends a code to the tape controller which sets control A to the BCD mode. A Clear Interrupt Lockout instruction at address 1024 again enables any interrupts to the 160-A.

Instructions at addresses 1026, 1027 set control A in the tape controller to 556 BPI density.

Another EXF instruction at address 1030 sends the Write code (6040) to the 3681. (A separate EXF instruction is needed since the 3681 cannot itself generate an Output Resume unless the code is preceded by an EXF instruction.) The Output instruction at address 1032 begins the Output operation, tape motion starts, and writing continues until all 100_8 words have been written on tape. Another Clear Interrupt Lockout instruction (address 1034) enables any interrupts which may have occurred during the operation.

The program waits until the controller is not busy (1034-1043) and then continues at address 1046 with another EXF instruction which sends the Function Initiate code to the adapter. The Output instruction at address 1050 transmits the Rewind code (0010) found at address 1203 to the tape controller. After doing a Clear Interrupt Lockout (address 1052) the program halts at address 1054.

INTRODUCTION TO CHAPTER XII

The 3400 system may use the 3682 Satellite Coupler for communication between 3400 and 160-A systems. The 3682 may also be used for communication between two 3400 systems. The information in chapter XII applies to the 3400 system as well as the 3600 system.

CHAPTER XII
3682 SATELLITE COUPLER

The 3682 satellite coupler is a peripheral device which allows communication between two 3606 data channels or between a 3606 data channel and a 3681 converter. (The 3681 converter is used with a 160 or a 160-A.) In addition to providing a bi-directional 12-bit data path between the two 3606 channels, it contains eight flags which may be set or cleared by either channel. These flags may be sensed or may be selected to cause interrupts. (The same flags may be used in an identical manner when communication takes place between a 3606 data channel and a 3681 adapter.)

A block diagram of the 3682 is shown in figure 12-1. The 3682 may be considered as consisting of two divisions, A and B. Each division contains control logic, a 10-bit Interrupt Mask register, and input/output hardware. The two divisions share a common Flag register.

Bits in the Flag register may form logical products with corresponding bits in either Interrupt Mask register. When the logical product of corresponding bits equals one, an interrupt is sent to the computer which set up the interrupt condition.

Figure 12-2 shows typical 3682 configurations.

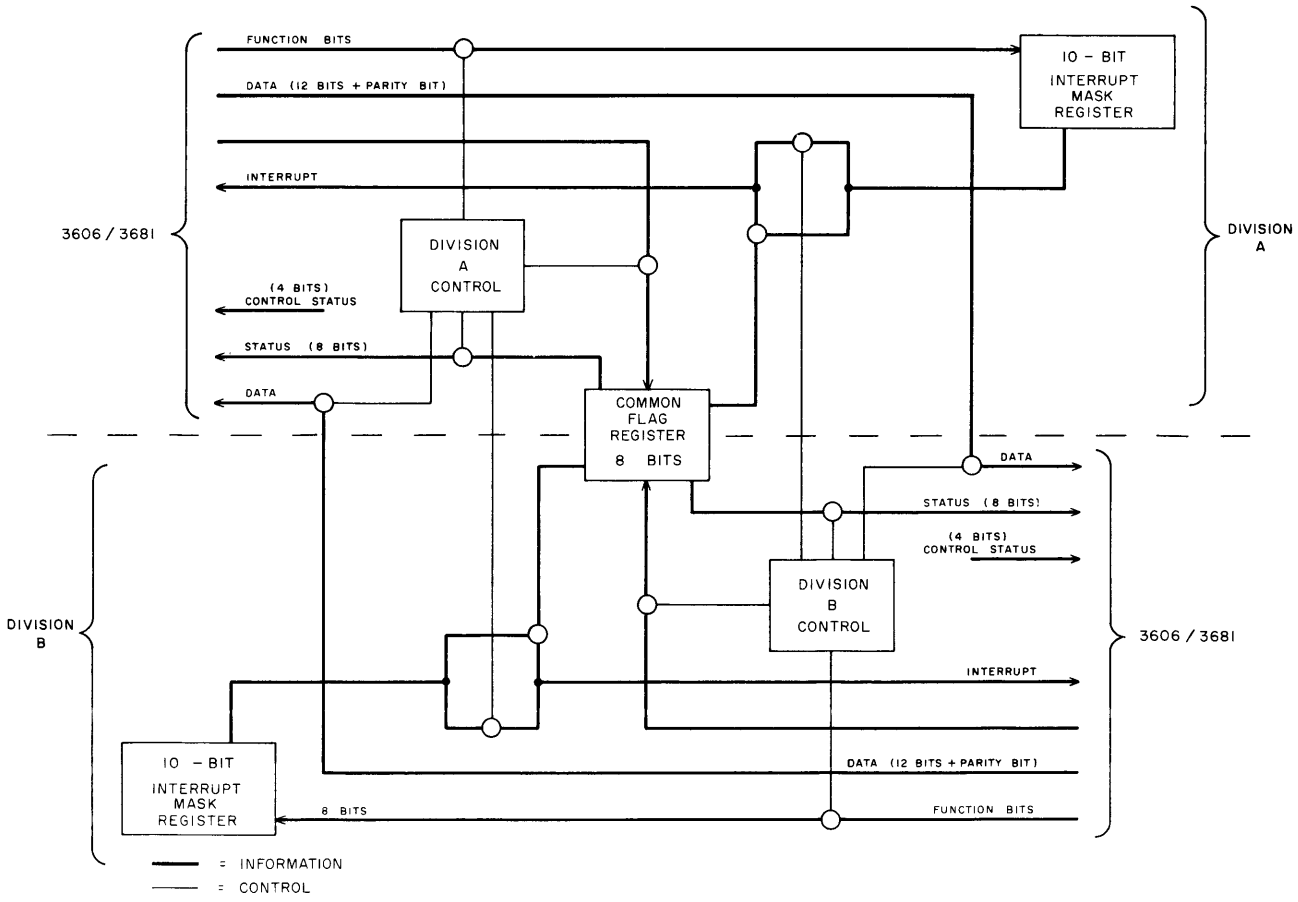


Figure 12-1. 3682 Block Diagram

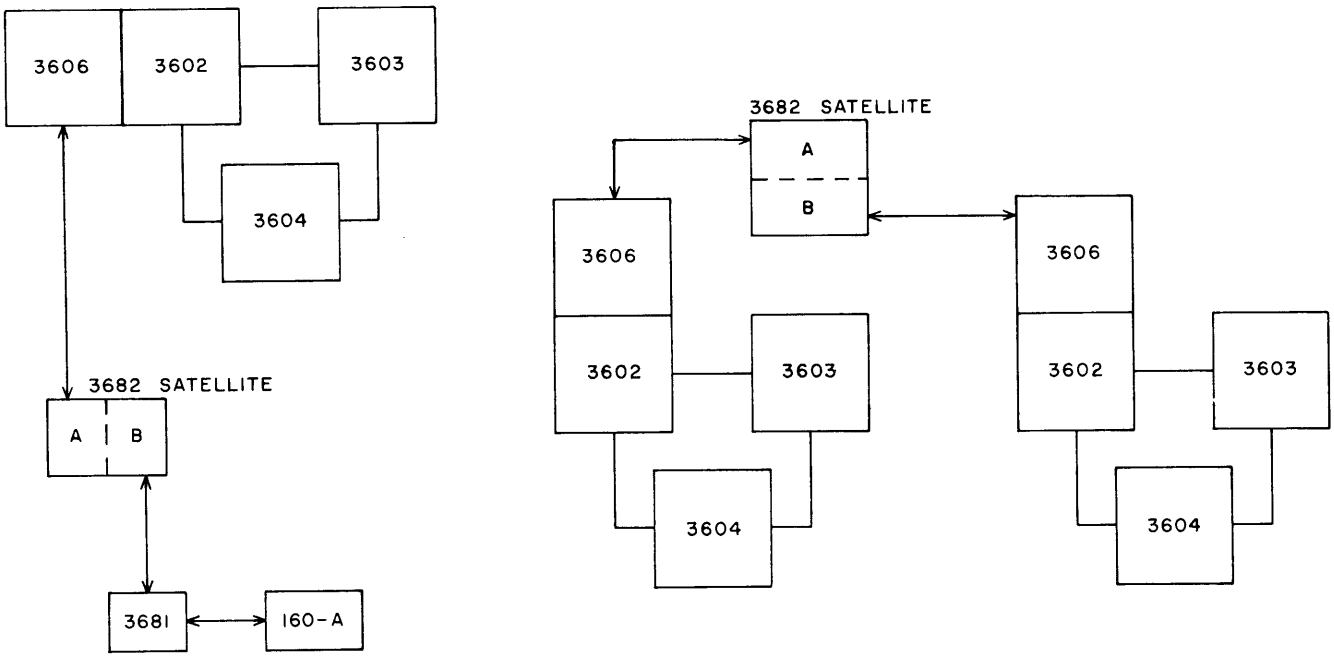


Figure 12-2. Typical 3682 Configurations

SWITCHES

An eight-position switch on each division designates the division as one of eight optional peripheral equipments (e.g., if the switch on division A is set to 3, that division responds to the 12-bit Connect code 3000_8).

INDICATORS

Transmission Parity Error (2)

A Transmission Parity Error indicator in a division lights if a parity error occurs in the division during a Connect, Function, or Write operation. This light remains on until a MC is applied to the division in which the parity error occurred.

Connect (2)

A Connect indicator in a division lights when the division is connected to a data channel or data channel converter. This light remains on until the division is no longer connected.

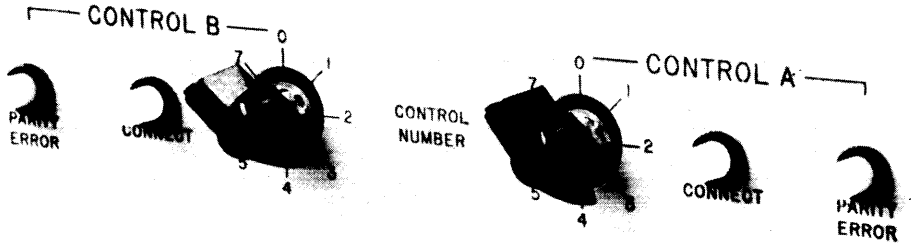
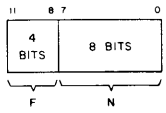


Figure 12-3. 3682 Switch and Indicator Panel

FUNCTION CODES

The 12-bit Function code is divided into a 4-bit F portion and an 8-bit N portion. Each of the 8 bits in N has a corresponding flag (e.g., bit 5 in N corresponds to flag 5).



The function codes (in binary) are as follows:

F = 0001 Set the flags corresponding to the "1" bits of N.

F = 0010 Clear the flags corresponding to the "1" bits of N.

F = 0100 Set Interrupt Mask register corresponding to "1" bits of N.

F = 1000 Clear Interrupt Mask register corresponding to "1" bits of N.

F = 0000, N = 00000001
Select interrupt on other division in abnormal condition (transmission parity error or computer not running).

F = 0000, N = 00000010
Clear abnormal interrupt.

F = 0000, N = 00000100
Select interrupt if other division is active (read or write lines up).

F = 0000, N = 00001000
Clear other division active interrupt.

STATUS LINES

The twelve status lines are assigned as follows:

Bit 0 -- Flag 0
Bit 1 -- Flag 1
Bit 2 -- Flag 2
Bit 3 -- Flag 3
Bit 4 -- Flag 4
Bit 5 -- Flag 5
Bit 6 -- Flag 6
Bit 7 -- Flag 7
Bit 8 -- Other division, computer running
Bit 9 -- Other division, read
Bit 10 - Other division, write
Bit 11 - Other division, transmission parity error

PROGRAMMING

The 3682 may be used with the 3681. See the 3681 chapter for detailed 3681 programming information.

The order of events involved in programming the 3682 when used as a link between two 3606 data channels or a 3681 and a 3606 (figure 12-1) is:

1) Clear
2) Connect
3) Check status
4) Functions, Read/Write
5) Clear flags and interrupts when complete if they serve no further purpose.

Clear

There are three different types of clears used in the 3682:

Power On MC — This clears all the flags when power is first applied to the 3682.

MC — If the 3600 system is physically attached to a division of the 3682 via a 3602 and a 3606, an external master clear from the 3600 system clears all selected interrupt conditions in that division. If the 160/160-A is physically attached to a division of the 3682 via a 3681, a master clear from the 160/160-A clears all selected interrupt conditions in that division.

Function Clear — Selected flags and interrupts may also be cleared by a Function code, but the division of the 3682 about to be cleared must be connected before a function can be executed. (See comments concerning F codes in the section on Function Codes.)

The type of clear needed at the end of a program involving the 3682 depends on the specific situation, but the operator must clear flags and interrupts either at the end of a program or at the beginning of a new program.

Connect

After clearing common flags by a Power On MC and division interrupts by master clears, each division must be connected.

12-bit Connect codes are transmitted to the divisions of the 3682 by the computers desiring a communication path. The Connect code from each computer must match the switch setting on its division of the 3682. (See the 3681 chapter for information on generating a 12-bit Connect code when using the 3681 with a 160/160-A).

The order and time of connecting the divisions of the 3682 are not critical, but each division must be connected before a communication path between computers is established.

A Reply is returned to the equipment connecting a division when the Connect operation is complete. If a parity error occurs during transmission of the Connect code (odd parity on transmission; 12-bit code +1 parity bit), a Reply is not returned. A Parity Error signal is returned to the computer attempting the connection.

Status

Twelve status lines return from each division (through a data channel, etc.) to the computer attached to it. When one computer has been connected to a division, it can check:

1) Flags	If bit 5 is present on the status lines, it indicates flag 5 is set.
2) Other division, computer running	If bit 8 is present, it indicates that the computer associated with the other division is running (but not necessarily connected).
3) Other division, transmission parity error	If bit 11 is present, it indicates that a transmission parity error has occurred while the other computer was attempting a Connect, Function, or Write operation on its division.
4) Read (other division)	If bit 9 is present on the status lines (both computers connected to the 3682), it indicates that the other computer is sending a Read signal to the 3682.
5) Write (other division)	If bit 10 is present on the status lines (both computers connected to the 3682), it indicates that the other computer is sending a Write signal to the 3682.

Function

When a division of the 3682 has been connected, 12-bit Function codes may set up certain operating conditions within the 3682. (See page 12-3 for the F portions of the Function codes). These conditions are:

$F = 0001_2$	If bit 8 is present in the F portion of the 12-bit Function code (bits numbered from right to left in ascending order), all flags are set which correspond to the "1" bits in the N portion of the code (e.g., the 12-bit code 0405_8 sets flags 0 and 2).
$F = 0010_2$	If bit 9 is present in the F portion of the 12-bit Function code, all flags are cleared which correspond to the "1" bits in N (e.g., the 12-bit code 1122_8 clears flags 1, 4, and 6).
$F = 0100_2$	If bit 10 is present in the 12-bit Function code, a Mask register is set corresponding to the "1" bits in N. (Each division has its own Mask register, but shares the common Flag register.) When the flag that corresponds to the "1" bit in N is set by either computer, an interrupt is sent to the computer that set up the interrupt conditions (e.g., Function code 2003_8 selects interrupt when flag 0 = 1 and flag 1 = 1).

$F = 1000_2$ If bit 11 is present, the computer that selected the interrupt in a division clears that condition (e.g., Function code 4003_8 clears interrupt when flag 0 = 1 and flag 1 = 1). This code does not clear the flags which were set to cause the interrupt. These may be cleared by a Power On MC or the Clear Flag code ($F = 0010_2$).

$(F = 0000, N = 00000001)_2$
This selects an abnormal interrupt condition (bit 0 = 1 in N, all other bits in the code = 0). If the other division has a computer not running, a transmission parity error (on an attempted connect), or both, an interrupt will be returned to the computer which selected the interrupt.

$(F = 0000, N = 00000010)_2$
This clears the abnormal interrupt condition selected by a computer. (Bit 1 in N = 1, all other bits = 0.)

$(F = 0000, N = 00000100)_2$
Interrupt is selected by a computer if the computer in the other division has a read or write line up.

$(F = 0000, N = 00001000)_2$
This clears the condition set up by $(F = 0000, N = 00000100)_2$.

When a particular interrupt condition is met, the Interrupt signal remains up until a code clears the interrupt.

Read/Write

Information transfer takes place when one computer does a Write operation and the other computer does a Read. The exchange terminates when: (1) the computer doing the Write operation drops its Write signal, or (2) the computer doing the Read operation no longer requests information. An End of Record signal (Input Disconnect in the case of the 160-A) is sent to the computer doing the Read operation when case 1 occurs. In the latter case, the data channel doing the Write operation waits indefinitely.

EXPANDED 3600 SATELLITE

Figure 12-4 shows an expanded satellite system. The 3606 data channel may communicate with a total of eight 3600-type external equipments. The 160-A may also communicate via the 3681 with a total of eight 3600-type external equipments, such as the 3620 tape controller, the 3642 card punch, etc.

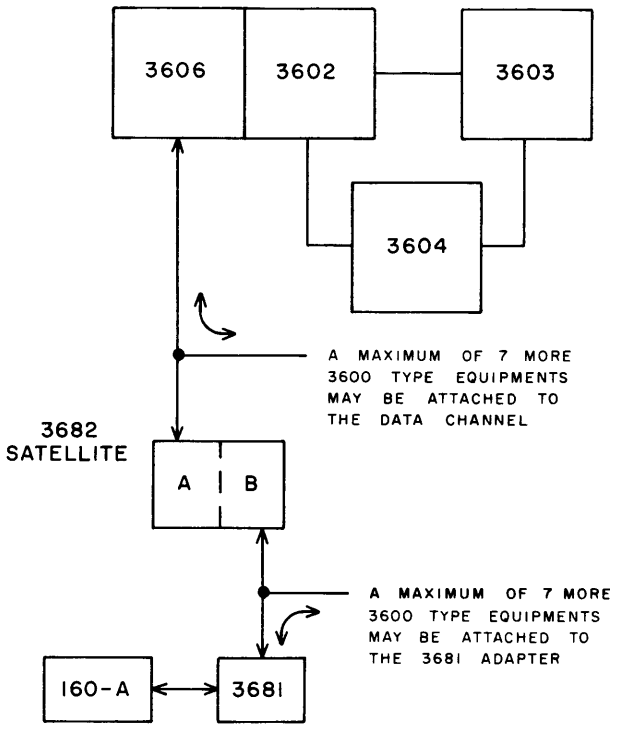


Figure 12-4. Expanded Satellite System

PROGRAMMING EXAMPLE

Since the programming possibilities of a satellite system are beyond the scope of this manual, a general programming example is included here. The desired results may be achieved in many ways other than those presented.

Problem: Transfer 100_8 words from the 3600 system to the 160-A via the 3682 satellite. Notify the 160-A by bringing up a write line in division A prior to the word transfer. (This will interrupt the 160-A and it will enter into a predetermined interrupt routine.) Refer to figure 12-5.

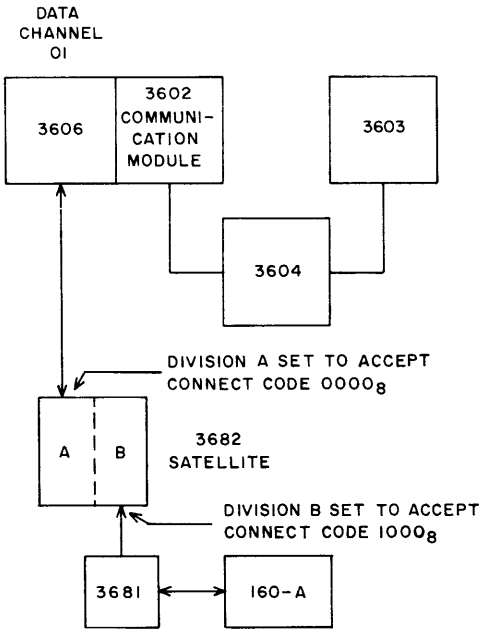


Figure 12-5. Satellite System Programming Example

Initial Conditions:

1) The 3606 data channel is designated channel 01.

2) The 3606 is physically attached to division A of the 3682. Division A is set to accept Connect code 0000_8.

3) The 160-A is physically attached to the 3681. The 3681 is physically attached to the 3682, division B. Division B is set to accept Connect code 1000_8.

4) Both the 160-A and the 3600 system are running.

The 3600 system and 160-A must be connected to their respective divisions of the 3682 prior to information transfer. In addition, the 160-A must have division B selected to interrupt the 160-A when the 3600 system brings up its write line. The 160-A and 3600 programs proceed as follows:

160-A PROGRAM	3600 PROGRAM
START	START
A Connect division B via the 3681 (Connect code = 1000_8; see 3681 Reference Manual).	
B Select interrupt on division B if division A is active (i.e., read/write line up).	
160-A waits for interrupt from 3682, division B.	C Connect division A via the 3606 data channel (Connect code = 0000_8). 3600 continues main program D 3600 reaches the point in its program to transfer 100_8 words to the 160-A. The write line (+ Data signal) comes up which interrupts the 160-A.
E The 160-A enters an interrupt routine. It does a status check, determines that the desired operation is Input, and brings up Read and Data signals to the 3682 via the 3681.	3600 waits for Read and Data signals to appear in division B.
F The transfer of information from the 3600 to the 160-A occurs. (The 3600 does a Write operation and the 160-A does an Input or Read operation.) The 3600 waits for the 160-A to generate a Data signal each time before the 12-bit word transfers occur.	
G The operation terminates when the 3600 drops its Write signal. This sends an End of Record signal to the 3681 which terminates the Input operation by sending an Input Disconnect to the 160-A. (The 160-A could also keep a word count and terminate the Input operation after it had stored 100_8 words, but the word count in the 3602 must match or the data channel may halt indefinitely.)	
H 160-A program continues independent of the 3600.	3600 program continues independent of the 160-A.

INTRODUCTION TO CHAPTER XIII

The 3400 system may use either the 3446 or 3644 card punch. The 3446 is identical to the 3644 with the following exceptions:

1) The 3446 is a single control version of the 3644, i.e. it can be cabled to only one input/output channel.

2) Being a single control device, the 3446 has no reserve capabilities.

It is strongly urged that the user program the 3446 as if it were a 3644. This involves such things as releasing the punch when all operations are complete, etc. The 3446 will respond in the normal manner to the release code and all other 3644 codes, even though it does not contain all the capabilities of the dual-control 3644. A program written for the 3644 will run on the 3446 with no alteration. Then, if the user decides to switch to the 3644, there will be complete compatibility.

CHAPTER XIII
3644 CARD PUNCH CONTROLLER

The CONTROL DATA 3644 Card Punch Controller adapts either an IBM 523 or 544 Card Punch for use with a 3606-type data channel. The controller contains all necessary logic for connecting, punching, and error checking. Two data channels may be physically connected to the controller as shown below, though only one may use it at a time.

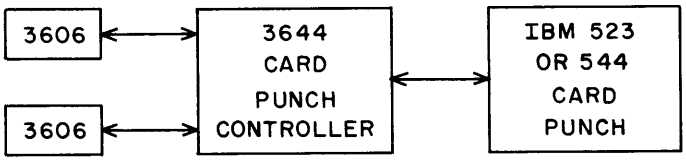


Figure 13-1. Typical 3644 Configuration

The controller contains a 12 x 80 "turn around" core memory which holds the data to punch one card. The data channel loads the memory on a 12-bit column by column basis beginning with the lower order column. The memory is unloaded and data is punched on a row by row basis.

HOLLERITH CONVERSION

In the normal mode of operation each 12-bit byte from the channel is considered as two characters in internal BCD. The controller converts BCD characters to Hollerith characters. Thus in the first 12-bit byte the upper 6 bits are translated into a Hollerith code which is then stored in column 1 of the memory. The lower 6 bits of this 12-bit byte are translated into a Hollerith code which is stored in column 2. Forty 12-bit bytes fill the 80-column card. Table 13-1 lists the codes received, resulting Hollerith punches and the characters represented.

After the Negate BCD to Hollerith conversion function is performed, data is punched just as received. Thus the first 12-bit byte specifies the punching for column 1, etc.

HOLE COUNT COMPARISON

Both the card punches used with the 3644 controller have a read station behind the punch station. As data from the channel is loaded into memory a count of one's is made, where one's = holes to be punched. This count is stored until the card is punched and read by the read station (during the next punch cycle). A total hole count is prepared as the card is read. Comparing the stored one's count and the total hole count reveals an error if they are not equal. The program can be notified of the error by interrupt, but in any case the compare error status line comes up.

TIMING

When the IBM 523 is used rates are:
- 100 cards per minute
- 600 milliseconds for complete punch cycle

When the IBM 544 is used rates are:
- 250 cards per minute
- 240 milliseconds for complete punch cycle

Regardless of the punch being used, loading the turn around memory requires 1400 μsec for Hollerith punching and 1600 μsec for binary punching. To maintain maximum punch rate the channel should send out data to the 3644 for the next card as soon as the controller becomes "not busy" at the end of the punch cycle for the current card. For the 523 with Hollerith punching, the channel becomes not busy approximately 598.6 ms after loading memory and with binary punching not busy occurs 598.4 ms after loading memory. For the 544 with Hollerith, not busy occurs 238.6 ms after loading and with binary cards the not busy condition occurs 238.4 ms after loading.

The time required to transfer a byte from the channel to controller memory is approximately 20 μsec per byte when information is punched in binary form. For information punched in Hollerith the transfer time is approximately 35 μsec per byte.

Table 13-1. BCD/Hollerith/Card Codes

Internal BCD Code	Char	Card	Internal BCD Code	Char	Card
00	0	0	40	(minus) -	11
01	1	1	41	J	11, 1
02	2	2	42	K	11, 2
03	3	3	43	L	11, 3
04	4	4	44	M	11, 4
05	5	5	45	N	11, 5
06	6	6	46	O	11, 6
07	7	7	47	P	11, 7
10	8	8	50	Q	11, 8
11	9	9	51	R	11, 9
12		8, 2	52	-0	11, 0
13	=	8, 3	53	$	11, 8, 3
14	(dash) -	8, 4	54	*	11, 8, 4
15		8, 5	55		11, 8, 5
16		8, 6	56		11, 8, 6
17		8, 7	57		11, 8, 7
20	+	12	60	(Space)	Blank
21	A	12, 1	61	/	0, 1
22	B	12, 2	62	S	0, 2
23	C	12, 3	63	T	0, 3
24	D	12, 4	64	U	0, 4
25	E	12, 5	65	V	0, 5
26	F	12, 6	66	W	0, 6
27	G	12, 7	67	X	0, 7
30	H	12, 8	70	Y	0, 8
31	I	12, 9	71	Z	0, 9
32	+0	12, 0	72		0, 8, 2
33	.	12, 8, 3	73	,	0, 8, 3
34	)	12, 8, 4	74	(	0, 8, 4
35		12, 8, 5	75		0, 8, 5
36		12, 8, 6	76		0, 8, 6
37		12, 8, 7	77		0, 8, 7

INDICATORS AND SWITCHES

3644

Reserve A/ Reserve B	I*	Indicates the channel reserving the punch. Comes on following connect for which reply was satisfied. Goes off by release or clear.
Parity Error	I	Indicates a parity error in transmission from the data channel to the controller. Turned off by master clear or channel clear.
Compare Error	I	Indicates detection of difference in pre-punch one's count and post-punch hole count. Turned off by issuing a new function.
Punch Not Ready	I	Indicates punch is not in operable condition.
Fail to Feed	I	Indicates a card failed to feed from hopper to pre-punch station; also causes Punch Not Ready condition. Turned off by manually advancing cards from hopper.
Equipment Number	S	Two such switches, one for each channel going to the controller.

523

Start	S	Causes punch to start and advance a card one cycle. At end of punch operation press twice to unload cards from stations.
Stop	S	Causes punch to stop and become not ready. Turns off Ready light.
Reset	S	Causes punch to become ready after Stop was pressed to reload hopper or unload stacker. Does not advance cards.
Ready	I	Indicator comes on after pressing Start unless hopper is empty, stacker is full or chip box is full. Indicator flashes as cards are punched.

544

Start	S	Causes punch to start and advance a card. When cards are initially loaded, pressing Start once advances cards into all stations. Pressing twice unloads cards at stations at end of punch.
Stop	S	Causes punch to stop and become "not ready". Turns off Ready indicator.
Reset	S	Causes punch to become "ready" after Stop was pressed, for example, to reload hopper or unload stacker. Does not advance cards.
Transport	I	Indicates when there are cards at any of the stations.
Ready	I	Indicator comes on after pressing Start unless stacker is full, hopper is empty or chip box is full.

* I = indicator, S = switch.

Table 13-2. 3644 Codes

CONNECT	
Connect Equipment N	N000
FUNCTION	
Release Reserve	0000
Negate BCD to Hollerith Conversion	0001
Release Above	0002
Check Last Card	0004
Clear	0005
Interrupt on Ready and Not Busy	0020
Release Above	0021
Interrupt on End of Operation	0022
Release Above	0023
Interrupt on Abnormal End of Operation	0024
Release Above	0025
STATUS	
Punch Ready	XXX1
Punch Busy	XXX2
Fail to Feed	X1XX
Interrupt Due to Ready and Not Busy	X2XX
Interrupt Due to End of Operation	X4XX
Interrupt Due to Abnormal End of Operation	1XXX
Compare Error	2XXX
Reserve Reject	4XXX

CODES

Connect

Connect Punch, Equipment N (N000)

The N part of the Connect code must match the setting of the Equipment Number switch for the controller section (either A or B). If the controller can be connected to the channel, a Reply is returned to the channel. Otherwise a Reject is returned. Once a channel is connected to the controller it has the punch reserved until a master clear occurs or until that channel issues a Release code. Due to the reservation, any attempts by the other data channel to connect the controller will result in a reject being sent back to that channel. The reservation holds despite the fact that the reserving channel may currently be connected to another equipment.

Although a connect may result in a reject, it does make status information available to the channel attempting the connect. The controller will gate status bits to the rejected channel until that channel attempts to connect another equipment.

Function

Release Reserve (0000)

This code, when received by the connected or reserved controller channel, releases the reservation of the punch to allow other channel to use it.

Negate BCD to Hollerith Conversion (0001)
Release Above (0002)

After this code, data is punched in the form received; the first byte will be punched in column 1 with bit 11 in row 12, bit 10 in row 11, etc.

Unless this Function operation has been performed, the controller performs the conversion of bytes to Hollerith characters.

Check Last Card (0004)

After the data channel has sent out data for the last card and it has been punched, the card is positioned at the post-punch read station. The Check Last Card code advances the punch one cycle and performs the total hole count check on the last card. Also, in case of a feed failure. this code should be used to check the last card prior to manual intervention.

Clear (0005)

A Clear code removes any interrupt selection or interrupt condition, and re-establishes the conversion to Hollerith mode of operation. This code does not release a reservation.

Interrupt on Ready and Not Busy (0020)
Release Above (0021)

This code allows an interrupt to be sent to the computer via the data channel when a new operation can be started. Usually the interrupt is interpreted as signalling the completion of a manual operation. The punch is ready if (1) cards are in the hopper, punch, and read stations, (2) the stacker is not full, and (3) the chip box is not full. The controller becomes not busy at the end of a punch cycle if the controller memory is not reloaded. The interrupt response for this condition is cleared by a reselection (code 0020), the release of selection (code 0021), or the Clear code (code 0005).

Interrupt on End of Operation (0022)
Release Above (0023)

The interrupt will occur when (1) all information

has been transferred, the channel is no longer busy, and the punching of current record is complete, (2) if the punch becomes not ready, or (3) a comparison error is detected. The interrupt response is cleared by codes 0022, 0023, or 0005.

Interrupt on Abnormal End of Operation (0024) Release Above (0025)

The interrupt will occur at the end of a punch cycle when one of the following conditions exist: (1) comparison error, (2) feed failure, or (3) punch not ready. The interrupt response is cleared by codes 0024, 0025, or 0005.

Status

Information is constantly available on these lines when a controller channel is connected to a data channel.

Punch Ready (XXX1)

The punch is ready when it can be used by the data channel. This involves several conditions including: (1) stacker not full, (2) chip box not full, and (3) cards present in hopper, pre-punch, punch, and post-punch read stations.

Upon pressing the Punch Stop switch, the punch becomes not ready at the end of the current punch cycle. In this case the punch is made ready by pushing the Reset switch (this does not advance cards). Once ready, the punch remains continuously ready until one of the aforementioned conditions arise to prevent further operation. The punch becomes not ready only at the end of a card cycle. Note that when an I/O chain operation is in process it is possible at the end of a card cycle for the status to be busy and not ready.

Punch Busy (XXX2)

The controller is busy when the data channel is busy with a Write operation or the punch is busy. The channel becomes busy upon initiation of a Write operation although actual punching has not yet begun. A Check Last Card code also makes the punch busy and it remains so until the hole count check is completed.

Fail to Feed (X1XX)

A feed failure means that when a punch cycle was initiated, a card did not feed from the hopper into the pre-punch station. If a feed failure occurs the computer should wait until the not busy line comes up and then issue a Check Last Card code.

Interrupt Due to Ready and Not Busy (X2XX)

This bit indicates that Interrupt on Ready and Not Busy (0020) was selected and the ready and not busy conditions now exist.

Interrupt Due to End of Operation (X4XX)

This bit indicates that Interrupt on End of Operation (0022) was selected and the end of operation condition now exists.

Interrupt Due to Abnormal End of Operation (1XXX)

This bit indicates that Interrupt on Abnormal End of Operation was selected and the condition now exists.

Compare Error (2XXX)

The card punched on the previous cycle did not have total hole count equal to one's count. The bit remains up until another code is issued to the controller.

Reserve Reject (4XXX)

The punch is reserved by the other channel.

PROGRAMMING

A typical order of steps in programming the punch is:

Clear (by external master clear or clear channel)
Connect
Function
Write
Copy Status (to determine satisfactory completion of operation)
Function (check last card)
Function (release punch)

The Clear could also be accomplished after the Connect by Clear code 0005. Part of the need for a Clear arises from the two channel feature. When one channel finally releases the punch to the other channel, the function selections of the first still prevail. The Clear enables the newly connected channel to begin with previous selections cleared.

The word count given by the Write instruction may specify fewer, the exact number, or more bytes than required to punch one card. If fewer bytes are speci-

fied one card is punched, and it contains the data sent to the punch. The timing changes are minor since only the time to load memory is reduced.

If the exact number of bytes to fill the card are specified, one card is punched and timing is as specified. If more bytes are specified than can be punched on one card, the following occurs. After enough bytes are received to punch one card, the controller stops the transmission of bytes and punches one card. Then the controller accepts remaining bytes and punches a second card, etc.

The Check Last Card code advances the card to the post-punch read station and makes the comparison of the one's count and the hole count.

When one channel has completed its use of the punch, the Release Reserve code is necessary to remove the reservation and allow the other channel use of the punch. An external master clear or Clear Channel code will also release any reservation.

INTRODUCTION TO CHAPTER XIV

The 3400 system may use either the 3447 or the 3649 card reader. The 3447 is identical to the 3649 with the following exceptions:

1) The 3447 is a single control version of the 3649, i.e. it can be cabled to only one input/output channel.

2) Being a single control device, the 3447 has no reserve capabilities.

3) The 3447 has no Input to A capabilities.

It is strongly urged that the user program the 3447 as if it were a 3649. This involves such things as releasing the reader when all operations are complete, etc. The 3447 will respond in the normal manner to the release code and all other 3649 codes, even though it does not contain all the capabilities of the dual-control 3649. A program written for the 3649 will run on the 3447 with no alteration. Then, if the user decides to switch to the 3649, there will be complete compatibility.

CHAPTER XIV
3649 CARD READER CONTROLLER

The CONTROL DATA 3649 Card Reader Controller consists of a basic CONTROL DATA 405 Card Reader with additional logic. The additional logic provides for operation of the 3649 with the 3600 type bi-directional data channels. This chapter covers only the controller logic as related to the card reader. The 405 manuals should be consulted for information concerning the card reader.

All information on a card is read twice, checked and then loaded into the reader buffer memory. The operator has the choice of reading data as it appears on the cards or translating Hollerith cards to BCD format. If Hollerith cards are to be read and translated, two Hollerith characters are sent to the computer per byte (6 bits for each character). 51-column cards may be read by setting the 51 Column switch (51-column cards can be read in binary format only). The 405 card reader can read 1200 80-column cards/minute or 1600 51-column cards/minute.

A Gate Card code allows the 3649 to perform a limited sorting operation. As the cards are read they are normally directed to the primary stacker. The use of the Gate Card code will direct selected cards to the secondary stacker. A Function code must be issued for each card that is to be directed to the secondary stacker. With care, cards may be added to the input tray and removed from the primary stacker while the reader is in operation. Cards may not be removed from the secondary stacker during operation.

Input to A mode is accomplished by setting the Card Input Mode switch on the 3601 console to select the reader for the Input to A operation. The reader must be in an unreserved condition for the Input to A selection. Card motion and each card cycle are initiated by the presence of the Card Input to A select line and the request from the computer.

OPERATION AND PROGRAMMING

This section describes the operating controls, necessary preliminary conditions, and basic programming of the 3649 card reader controller.

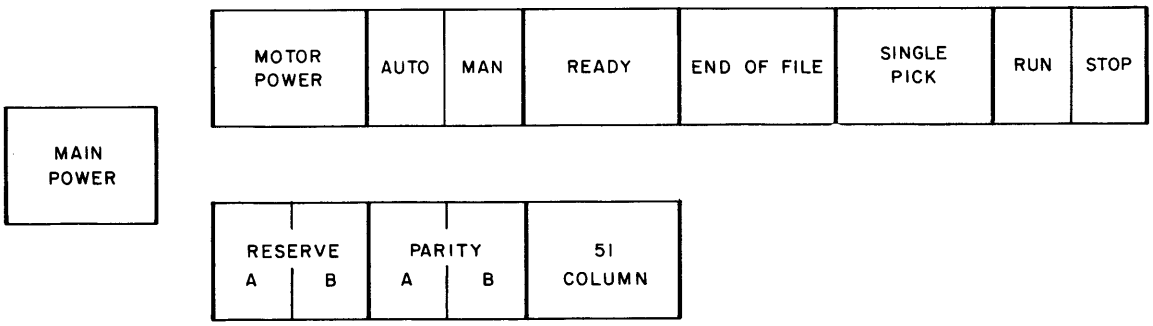


Figure 14-1. 3649 Control Panel

Switches

Main Power (I)*

Controls all primary power and photocell light source. Light indicates when main power is on.

Motor Power (I)

Controls drive motors, vacuum-pressure system, and input tray-stacker vibrators. The On condition is displayed by the indicator light.

* Switches followed by (I) are illuminated when activated.

Auto/Man (I)

In the AUTO position the reader is reserved for computer use and the Auto light is on. In the MAN position, the indicator is on and the reader is reserved for adding cards to the input tray or removing cards from the stacker tray. Maintenance may also be performed in the manual condition.

End of File (I)

This switch will cause an End of File signal to be generated when the information from the last card has been transferred to the computer. The indicator light is on when the switch is set.

Ready (I)

When set, the information from the first card will be loaded into memory. This switch must be set to establish a ready condition after switching from MAN to AUTO. The indicator light remains on during the ready condition until the hopper is empty or an abnormal condition developes.

Single Pick

Allows a single card to be cycled through the reader and loaded into memory when Auto/ Man switch is in the MAN position.

Run/Stop (I)

Disables all functions of the reader when the Auto/ Man switch is in MAN position. In the ready condition the Run light indicates that cards are being processed through the reader. The Stop light indicates that no cards are being processed.

51 Column (I)

When this switch is pressed the indicator will light to signify that short (51-column) cards will be read. When the indicator light is not on, 80-column cards may be read.

Equipment Number Selection

An eight-position switch (one for each control) determines whether or not this control will respond to a programmed Connect Code on the associated channel. This switch is located inside the logic chassis.

Any interrupts are transmitted to the data channel on the line corresponding to the setting of this switch.

Indicators

Reserve A/B

This light indicates which channel is reserving the read station.

Parity A/B

This light indicates a transmission parity error has occurred in a Connect or Function code. If a transmission parity error occurs in a Function code, a transmission Parity Error signal is returned to the data channel. A MC from the data channel clears the Transmission Parity Error signal. The MC turns off the indicator light when a transmission parity error occurs in either a Connect or Function code.

Card Reader Preparation

The following steps are necessary to prepare the 3649 for use:

1) Turn Equipment Number Selection switch to desired position.

2) Place cards in input tray.

3) Press Main Power switch on.

4) Press Motor Power switch on.

5) Press Man/Auto switch to AUTO.

6) Set End of File switch to OFF unless the hopper load is a complete file.

7) Press Ready switch.

8) Issue master clear from the computer.

9) The 3649 is now ready and the Ready switch should be illuminated.

Reader Conditions

The Reader is in a Normal condition when:

1) Card in input tray
2) Primary or secondary stacker not full
3) Reader buffer memory has been loaded.
4) No feed failure
5) No pre-read error
6) No compare error
7) Auto/Man switch in AUTO position.
8) Reader connected

Codes

All connections and operations in the 3649 are controlled by 12-bit Connect and Function codes (table 14-1). Status codes are constantly available when the reader is connected. In all discussion of codes, bit 0 is in the rightmost position of the code.

Connect

Connect Reader Controller Equipment N (N000)

These 12-bit portions of the 48-bit Connect instruction,* accompanied by a Connect signal, connect the desired read station via either data channel. Bits 9, 10 and 11 (N) must match the Equipment Number

Table 14-1. Connect, Function, and Status Codes

CONNECT	
Connect Reader Controller Equipment N	N000
FUNCTION	
Release Reserve	0000
Negate Hollerith to Internal BCD Conversion	0001
Release Negate Hollerith to Internal BCD Conversion	0002
Set Gate Card	0004
Clear	0005
Set Interrupt on Ready and Not Busy	0020
Release Interrupt on Ready and Not Busy	0021
Set Interrupt on End of Operation	0022
Release Interrupt on End of Operation	0023
Set Interrupt on Abnormal End of Operation	0024
Release Interrupt on Abnormal End of Operation	0025
STATUS	
Reader Ready	XXX1
Reader Busy	XXX2
Binary Card in Reader Memory	XX14
End of File	XX1X
Stacker Full or Jam	XX2X
Hopper Empty	XX4X
Fail to Feed	X1XX
Interrupt Ready and Not Busy	X2XX
Interrupt End of Operation	X4XX
Interrupt Abnormal End of Operation	1XXX
Read Compare or Pre-Read Error	2XXX
Reserve Reject	4XXX

* See Chapter 5 for a detailed description of the Connect code.

switch setting on the 3649 logic chassis for a particular channel. A reply is returned to the 3604 if the unit can be connected. When a channel has connected the read station, that station remains reserved until released via a Function code (0000). This prevents the other channel from using that station. Any Connect code that does not match the Equipment Number switch setting will clear a present connection.

Function

Release Reserve (0000)

When the read station has been reserved, it remains reserved until released via a Function code (0000) or a master clear.

Negate Hollerith to Internal BCD Conversion (0001)
Release Negate Hollerith to Internal BCD Conversion (0002)

The presence of a 7 and 9 punch in column one indicates that the card will be read as a binary card. The absence of a 7 and 9 punch indicates a Hollerith card, and initiates automatic Hollerith to BCD conversion. The automatic conversion may be negated by the 0001 code.

Set Gate Card (0004)

This code provides for sorting a card that has been read into memory and placing the selected card in the secondary stacker. This code must be issued ≤ 1.5 ms after the card has been read by the data channel. A Gate Card code must be issued for each card to be gated.

Clear (0005)

This code clears any interrupt selection, interrupt response or negate selection.

Set Interrupt on Ready and Not Busy (0020)
Release Interrupt on Ready and Not Busy (0021)

The primary reason for this interrupt is to signal the computer when it can start an operation. The interrupt also indicates the completion of a manual operation. Equipment is ready when the reader is in the normal condition (described in section on Reader Conditions). The interrupt signal for this condition is cleared by Set (0020), Release (0021) or Clear (0005).

Set Interrupt on End of Operation (0022)
Release Interrupt on End of Operation (0023)

Interrupt occurs when Read, Data, and Channel Busy signals drop. If a portion of a card is read from memory, Interrupt on End of Operation occurs when Read, Data, and Channel Busy signals drop. The remainder of the card is discarded. The status lines must be checked to determine the cause for the read termination. The interrupt signal may be cleared by Set (0022), Release (0023) or Clear (0005).

Set Interrupt on Abnormal End of Operation (0024).
Release Interrupt on Abnormal End of Operation (0025)

This Interrupt will occur when a failure occurs in a normal condition. The interrupt signal may be cleared by Set (0024), Release (0025) or Clear (0005).

Status Codes

Reader Ready (XXX1)

The ready condition exists if the memory has been loaded with the information from the first card and the information from that card is ready to be unloaded from memory. This condition drops when the reader enters a not normal condition.

Reader Busy (XXX2)

This status remains during the interval that memory is loaded. After the first card is read, the reader will be busy from initiation of the channel busy and ready condition until:

1) the channel busy drops and memory is reloaded or

2) the unit becomes not ready. Any operation that can no longer continue will make the reader not ready.

Binary Card in Reader Memory (XXX4)

The presence of a 7 and 9 punch in column 1 on a card flags a binary card. The read station senses information in this column when loading memory. Bit 2 is present after memory is loaded and the binary card is read.

End of File (XX1X)

This condition exists when the End of File switch is ON, the input tray is empty and memory is unloaded. When the input tray does not contain the last card of a file, the switch should be placed in the OFF position. The switch does not over-ride abnormal conditions.

Interrupt-Ready and Not Busy (X2XX)

This bit indicates that Interrupt on Ready and Not Busy has occurred.

Interrupt-End of Operation (X4XX)

This bit indicates that Interrupt on End of Operation has occurred.

Interrupt-Abnormal End of Operation (1XXX)

This bit indicates Interrupt on Abnormal End of Operation has occurred.

Read Compare or Pre-Read Error (2XXX)

Bit 10 indicates that either a comparison error was detected during the transfer of card information to memory or a read amplifier was not functioning properly prior to reading the information from the card to memory.

Reserve Reject (4XXX)

Code 4XXX indicates that the reader is reserved for the opposite channel.

Program Timing

The reader may send up to 40 translated or 80 untranslated 12-bit words to the data channel from each card. Figure 14-2 shows the timing for each card cycle. On-line, the chart enables a programmer to make full use of the computer between card columns and to reconnect the reader at a time that will insure full operation.

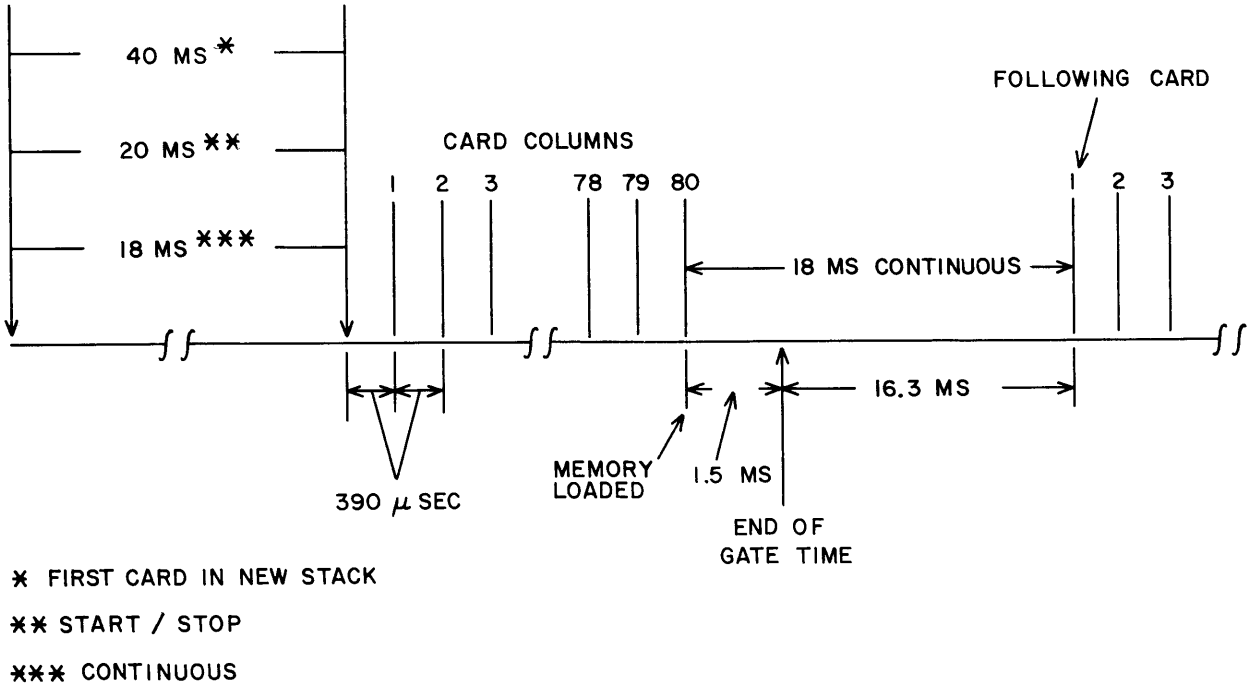


Figure 14-2. Card Timing Chart

Reader Codes

Table 14-2 lists the Hollerith codes required to produce internal BCD codes $00\text{-}77_8$.

Table 14-2. BCD / Hollerith / Codes

Internal BCD Code	Char	Card	Internal BCD Code	Char	Card
00	0	0	40	(minus) -	11
01	1	1	41	J	11, 1
02	2	2	42	K	11, 2
03	3	3	43	L	11, 3
04	4	4	44	M	11, 4
05	5	5	45	N	11, 5
06	6	6	46	O	11, 6
07	7	7	47	P	11, 7
10	8	8	50	Q	11, 8
11	9	9	51	R	11, 9
12		8, 2	52	-0	11, 0
13	=	8, 3	53	$	11, 8, 3
14	(dash) -	8, 4	54	*	11, 8, 4
15		8, 5	55		11, 8, 5
16		8, 6	56		11, 8, 6
17		8, 7	57		11, 8, 7
20	+	12	60	(Space)	Blank
21	A	12, 1	61	/	0, 1
22	B	12, 2	62	S	0, 2
23	C	12, 3	63	T	0, 3
24	D	12, 4	64	U	0, 4
25	E	12, 5	65	V	0, 5
26	F	12, 6	66	W	0, 6
27	G	12, 7	67	X	0, 7
30	H	12, 8	70	Y	0, 8
31	I	12, 9	71	Z	0, 9
32	+0	12, 0	72		0, 8, 2
33	.	12, 8, 3	73	,	0, 8, 3
34	)	12, 8, 4	74	(	0, 8, 4
35		12, 8, 5	75		0, 8, 5
36		12, 8, 6	76		0, 8, 6
37		12, 8, 7	77		0, 8, 7

APPENDIX SECTION

APPENDIX A
INTERRUPTABLE CONDITIONS AND FAULTS

Under certain internal conditions in the execution of a computer program, faults may occur. Most of these fault conditions are associated with a particular bit of the Interrupt register, and may be tested in an interrupt routine. In most cases, a fault condition does not stop operation, but a visual indication of a fault occurrence is provided on the console.

SHIFT FAULT

When a register shift of more than $177_8 = 127_{10}$ places is specified by the shift count in a Shift instruction, a bit in the Interrupt register is automatically set to "1". The Shift instruction shifts according to the value of the lower 7 bits of K. The Interrupt register bit remains set until:

1) an Internal Function instruction (Clear Arithmetic Fault or Clear Interrupt System) is executed, or

2) a manual internal master clear is performed.

A shift fault lights an indicator on the console.

DIVIDE FAULT

Bit 02 of the Interrupt register is automatically set to "1" when:

1) The absolute value of the quotient resulting from a Divide Integer instruction is $\geq 2^{47}$.

2) A fixed point or floating point divide by zero is attempted.

This Interrupt Register bit remains set until:

1) The Internal Function instruction (Clear Arithmetic Fault or Clear Interrupt System) is executed, or

2) A manual internal master clear is performed.

A divide fault lights an indicator on the console.

EXPONENT OVERFLOW FAULT

A bit position of the Interrupt register is automatically set to "1" when the value of the exponent formed during a floating point add, subtract, multiply, or divide is $> 2^{10} - 1$ (1777_8). This Interrupt Register bit remains set until:

1) The Internal Function instruction (Clear Arithmetic Fault or Clear Interrupt System) is executed, or

2) A manual internal master clear is performed.

An exponent overflow fault lights an indicator on the console.

EXPONENT UNDERFLOW FAULT

A bit position of the Interrupt register is automatically set to "1" when the value of the exponent formed during a floating point add, subtract, multiply, or divide is < negative $2^{10} - 1$ (-1777_8). This Interrupt Register bit remains set until:

1) The Internal Function instruction (Clear Arithmetic Fault or Clear Interrupt System) is executed, or

2) A manual internal master clear is performed.

An exponent underflow fault lights an indicator on the console.

ARITHMETIC OVERFLOW FAULT

A bit position of the Interrupt register is automatically set to "1" when:

1) The absolute value of the sum or difference of two fixed point integers is $\geq 2^{47}$. This Interrupt Register bit remains set until:

   a) The Internal Function instruction (Clear Arithmetic Fault or Clear Interrupt System) is executed, or

   b) A manual internal master clear is performed.

An arithmetic overflow fault lights an indicator on the console.

I/O PARITY ERROR

When an I/O Transmission Parity Error occurs in an I/O channel, one of four bits is set in the Interrupt register. This bit remains set until:

1) An Internal Function instruction (Clear I/O Parity Error or Clear Interrupt System) is executed, or

2) A manual internal master clear is executed.

OPERAND PARITY ERROR

A word read from storage for use as an operand by the computer or for use as an output word by an I/O channel is checked for parity. A bit position of the Interrupt register is automatically set to "1" if a parity error occurs.

This Interrupt Register bit remains set until:

1) The Internal Function instruction (Clear Interrupt System or Clear Parity Error) is executed, or

2) A manual internal master clear is performed.

An operand parity error lights an indicator on the console.

MANUAL INTERRUPT

When the momentary Manual Interrupt switch on the console is pressed, a bit position of the Interrupt register is set to "1" only if the corresponding bit of the Mask register is set.

This bit in the Interrupt register remains set until:

1) The Internal Function instruction (Select Manual Interrupt) is executed.

2) The Internal Function instruction (Clear Manual Interrupt) is executed.

3) The Internal Function instruction (Clear Interrupt System) is executed.

4) A manual internal master clear is performed.

INSTRUCTION PARITY ERROR

If a parity error occurs when reading an instruction from storage, the Instruction Parity Error bit in the Interrupt Register is set to "1". This bit remains set until:

1) An Internal Function instruction (Clear Parity Error or Clear Interrupt System) is executed, or

2) A manual internal master clear is executed.

OUT OF BOUNDS

Bit position 42 of the Interrupt register is automatically set to "1" when:

1) The interrupt system is active, and

2) Bit position 42 of the Interrupt Mask register is set to "1" (check bounds), and

3) A write reference is attempted out of bounds (the bounds addresses are defined by the contents of the two 8-bit bounds registers), or a jump is attempted out of bounds, or an attempt is made to read an instruction from out of bounds.

This Interrupt register bit remains set until:

1) The Out of Bounds Mask register bit is cleared by an Internal Function instruction 77.000541, or

2) An attempt is made to set the Out of Bounds Mask Register bit 42 with an Internal Function instruction 77.000540, even though bit 42 is already set, or

3) An Internal Function instruction (Clear Interrupt System) is executed, or

4) A manual internal Master Clear is performed.

APPENDIX B
INDEX TO INSTRUCTIONS (OCTAL CODES)

Octal Code	Operation	Page	Octal Code	Operation	Page
75	Selective Jump	3-16, 3-17	77.3	Internal Sense	3-21
76	Selective Stop	3-16, 3-17	77.4	Copy Product Register	5-5
77.0	Internal Function	3-18	77.5	Enter Upper Bound	3-22
77.1	Augment	3-21	77.6	Enter Lower Bound	3-22
77.2	Copy Interrupt Status	5-5			

APPENDIX C
INDEX TO INSTRUCTIONS (MNEMONIC CODES)

Mnemonic Code	Operation	Page
INI	Increase Index	3-12
ISK	Index Skip	3-12
LAC	Load A, Complement	3-10
SSK	Storage Skip	3-14
SST	Selective Set	3-12
SSU	Selective Substitute	3-13
STA	Store A	3-10
STC	Store Character	3-17
STL	Store Logical	3-13
STQ	Store Q	3-10
SUB	Subtract	3-11
THS	Threshold Search	3-15

APPENDIX D

CONTROL DATA 3400 COMPUTER INSTRUCTIONS

CONTROL DATA 3400 COMPUTER INSTRUCTIONS

SYMBOLS

k = Address portion of instruction
K = k + (B^b), Modified shift count
m = Address portion of instruction
M = m + (B^b), Modified operand address
y = Address portion of instruction
Y = y + (B^b), Modified operand
* = 48-bit instruction

b = Designator for index register
j = Designator for 22, 23, 75, 76
# = Restrict instruction to upper
' = Complemented
( ) = Contents of
NI = Next instruction
+ = Floating point option

FULL WORD DATA TRANSMISSION

LDA	12	Load A	(M)$\longrightarrow$A
LAC	13	Load A complement	(M)'$\longrightarrow$A
STA	20	Store A	(A)$\longrightarrow$M
LDQ	16	Load Q	(M)$\longrightarrow$Q
LQC	17	Load Q complement	(M)'$\longrightarrow$Q
STQ	21	Store Q	(Q)$\longrightarrow$M

PARTIAL WORD DATA TRANSMISSION

* LDC 63 Load character ($Byte_{0-7}$)$\longrightarrow$$A_{05-00}$
(63 b v0006 50 0 m)

* STC 63 Store character (A_{05-00})$\longrightarrow$$Byte_{0-7}$
(63 b v0006 50 5 m)

SAU 60 Substitute address upper
(A_{14-00})$\longrightarrow$$M_{UA}$

SAL 61 Substitute address lower
(A_{14-00})$\longrightarrow$$M_{LA}$

INTER-REGISTER TRANSMISSION

IAQ 00 Interchange A and Q (A)$\longrightarrow$Q, (Q)$\longrightarrow$A
(00 7 00554)

ATI 00 Transmit A to index (A_{14-00})$\longrightarrow$$B^b$
(00 7 4054b)

INDEXING

ENI	50	Enter index	y$\longrightarrow$$B^b$
INI	51	Increase index	y + (B^b)$\longrightarrow$$B^b$
LIU	52	Load index upper	(m_{UA})$\longrightarrow$$B^b$
LIL	53	Load index lower	(m_{LA})$\longrightarrow$$B^b$
SIU	56	Store index upper	(B^b)$\longrightarrow$$m_{UA}$
SIL	57	Store index lower	(B^b)$\longrightarrow$$m_{LA}$

ISK # 54 Index skip:
(B^b) $\neq$ y: (B^b) + 1$\longrightarrow$$B^b$, continue
(B^b) = y: 0$\longrightarrow$$B^b$, skip lower instruction

IJP 55 Index jump:
(B^b) $\neq$ 0: (B^b) - 1$\longrightarrow$$B^b$, jump to m
(B^b) = 0: continue

SHIFTING

ARS	01	Shift (A) right by K
QRS	02	Shift (Q) right by K
LRS	03	Shift (AQ) right by K
ALS	05	Shift (A) left by K
QLS	06	Shift (Q) left by K
LLS	07	Shift (AQ) left by K

ARITHMETIC

FIXED POINT

ADD	14	Add to A	(A) + (M)$\longrightarrow$A
SUB	15	Subtract from A	(A) - (M)$\longrightarrow$A
MUI	24	Multiply integer	(M) (A)$\longrightarrow$QA
DVI	25	Divide integer	(QA) / (M)$\longrightarrow$A, rem Q
RAD	70	Replace add	[(M) + (A)]$\longrightarrow$M & A
RSB	71	Replace subtract	[(M) - (A)]$\longrightarrow$M & A
RAO	72	Replace add one	[(M) + 1]$\longrightarrow$M & A
RSO	73	Replace sub. one	[(M) - 1]$\longrightarrow$M & A

SCA 34 Scale A Shift (A) left until $A_{47} \neq A_{46}$ or k = 0; (k - no. of shifts)$\longrightarrow$$B^b$
SCQ 35 Scale AQ Shift (AQ) left until $A_{47} \neq A_{46}$ or k = 0; (k - no. of shifts)$\longrightarrow$$B^b$

FLOATING POINT

+	FAD	30	Floating add.	[(A) + (M)]$\longrightarrow$A
+	FSB	31	Floating sub.	[(A) - (M)]$\longrightarrow$A
+	FMU	32	Floating mult.	(A) (M)$\longrightarrow$A
+	FDV	33	Floating div.	(A) / (M)$\longrightarrow$A

LOGICAL

SST 40 Selective set
Set (A_n) to 1 for $(M_n) = 1$
SCL 41 Selective clear
Clear (A_n) to 0 for $(M_n) = 1$
SCM 42 Selective complement
Complement (A_n) for $(M_n) = 1$
SSU 43 Selective substitute
$(M_n) \longrightarrow A_n$ for $(Q_n) = 1$
LDL 44 Load Logical L (Q) (M) $\longrightarrow$ A
ADL 45 Add logical [(A) + L (A) (M)] $\longrightarrow$ A
SBL 46 Sub. logical [(A) − L (Q) (M)] $\longrightarrow$ A
STL 47 Store logical L (Q) (A) $\longrightarrow$ M

INPUT/OUTPUT

* CONN 74.0 Connect
* EXTF 74.1 Function
* BEGR 74.2 Read
* BEGW 74.3 Write
* COPY 74.4 Copy status
* CLCH 74.5 Clear channel
* CCWD 74.6 Change control word

NO MEMORY REFERENCE

ENQ 04 Enter Q Extend sign Y, Y $\longrightarrow$ Q
ENA 10 Enter A Extend sign Y, Y $\longrightarrow$ A
INA 11 Increase A Extend sign Y,
Y + (A) $\longrightarrow$ A
EUB 77.5 Enter upper bound
ELB 77.6 Enter lower bound

GENERAL

INF 77.0 Internal function
AUG 77.1 Augment
CIS 77.2 Copy interrupt status
SEN 77.3 Internal sense
CPR 77.4 Copy product register

MEMORY TEST

SSK# 36 Storage skip (M) neg: skip lower instruction; (M) pos: continue
SSH# 37 Storage shift (M) neg: skip lower instruction, left 1; (M) pos: continue, left 1
EQS# 64 Search (B^b) words, if (M − 1), or (M − 2), etc. = (A) skip lower instruction; ≠ A, continue
THS# 65 Search (B^b) words, if (M − 1), or (M − 2), etc. > (A) skip lower instruction; ≦ A, continue
MEQ# 66 Search (B^b) words, if L (Q) (M − 1), or (M − 2), etc. = (A) skip lower instruction; ≠ A, continue
MTH# 67 Search (B^b) words, if L (Q) (M − 1), or (M − 2), etc. > (A) skip lower instruction; ≦ A, continue

A AND Q TEST

AJP 22 Jump to m on condition j
QJP 23 Jump to m on condition j

SELECTIVE JUMP AND STOP

SLJ 75 Jump to m on condition j
SLS 76 Stop on j, and jump to m

j	22	23	75	76
0	(A) = 0: Jump	(Q) = 0: Jump	Jump	Stop: Jump
1	(A) ≠ 0: Jump	(Q) ≠ 0: Jump	Key 1: Jump	Key 1: Stop: Jump
2	(A) Pos: Jump	(Q) Pos: Jump	Key 2: Jump	Key 2: Stop: Jump
3	(A) Neg: Jump	(Q) Neg: Jump	Key 3: Jump	Key 3: Stop: Jump
4	(A) = 0: Ret. Jump	(Q) = 0: Ret. Jump	Ret. Jump	Stop: Ret. Jump
5	(A) ≠ 0: Ret. Jump	(Q) ≠ 0: Ret. Jump	Key 1: Ret. Jump	Key 1: Stop: Ret. Jump
6	(A) Pos: Ret. Jump	(Q) Pos: Ret. Jump	Key 2: Ret. Jump	Key 2: Stop: Ret. Jump
7	(A) Neg: Ret. Jump	(Q) Neg: Ret. Jump	Key 3: Ret. Jump	Key 3: Stop: Ret. Jump

CUT ALONG LINE

COMMENT SHEET

CONTROL DATA 3400

PRELIMINARY REFERENCE MANUAL – PUB. NO. 282b

FROM NAME : _____

BUSINESS ADDRESS : _____

COMMENTS: (DESCRIBE ERRORS, SUGGESTED ADDITION OR DELETION AND INCLUDE PAGE NUMBER, ETC.)

NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

FOLD ON DOTTED LINES AND STAPLE

STAPLE

STAPLE

FOLD

FOLD

FIRST CLASS
PERMIT NO. 8241

MINNEAPOLIS, MINN.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
8100 34TH AVENUE SOUTH
MINNEAPOLIS 20, MINNESOTA

ATTN: TECHNICAL PUBLICATIONS DEPT.

CUT ALONG LINE

FOLD

FOLD

STAPLE

STAPLE