CONTROL DATA
CORPORATION

CONTROL DATA®
CYBER 170 SERIES
MODELS 172/173/174/175
COMPUTER SYSTEMS

NOS 1.0
GENERAL INFORMATION MANUAL

INTERRELATIONSHIP BETWEEN USERS AND MANUALS

NOS Publication	Applications [†] Analyst/ Programmer	Systems [†] Analyst	Operator [†]	Manager [†]
General Information Manual	Some	Some	Some	High
Reference Manual	High	High	Moderate	Some
Installation Handbook		High	Moderate	
Operator's Guide		High	High	
Time-Sharing User's Reference Manual	High	Moderate	Some	Some
Time-Sharing User's Guide	High	Some	Moderate	Moderate
NOS Instant Manual	High	High	Moderate	
Terminal User's Instant	High	Some	Some	
Export/Import Reference Manual	High	Moderate		Some
Batch User's Guide	High	Some	Moderate	Moderate
Text Editor Reference Manual	High	Some		Some

[†] Levels of interest

CONTROL DATA
CORPORATION

CONTROL DATA®
CYBER 170 SERIES
MODELS 172/173/174/175
COMPUTER SYSTEMS

NOS 1.0
GENERAL INFORMATION MANUAL

REVISION RECORD

REVISION	DESCRIPTION
A	Manual released.
(8-1-74)	

Publication No.
60435900

PREFACE

This manual provides a brief description of the CONTROL DATA® Network Operating System (NOS). It is intended as a general overview for computer center managers and as a guide to more information for programmers, analysts, and operations personnel.

The following publications provide reference material describing NOS, its related products, and the CDC CYBER 170 Series, Model 172, 173, 174, and 175 Computer Systems.

Control Data Publication	Publication No.
NOS Reference Manual	60435400
NOS Instant Manual	60436000
NOS Installation Handbook	60435700
NOS Time-Sharing User's Reference Manual	60435500
NOS Time-Sharing User's Guide	60436400
NOS Operator's Guide	60435600
NOS Terminal User's Instant Manual	60435800
NOS Export/Import Reference Manual	60436200
TRANEX User's Guide	60436500
TRANEX Reference Manual	60407900
TRANEX Operator's Guide Addendum	60408000
NOS Text Editor Reference Manual	60436100
NOS Batch User's Guide	60436300
Cyberlink Interchange Reference Manual	60373300
Cyberlink Interchange Operator's Guide	60408400
Modify Reference Manual	60281700
Modify Instant	60283000
BASIC Reference Manual	19980300
APL*CYBER Reference Manual	19980400
Update Reference Manual	60342500
COMPASS Reference Manual	60360900
FORTRAN Extended Reference Manual	60305600
COBOL Reference Manual	60384100
ALGOL Reference Manual	60329000
Sort/Merge Reference Manual	60343900
PERT/Time Reference Manual	60133600
SIMULA Reference Manual	60234800
SIMSCRIPT Reference Manual	60358500

SYMPL Reference Manual	60328800
Math Science Library Reference Manual	60327500
CYBER 170 Series/Model 172/173/174 Computer Systems Reference Manual	19981200
CYBER 170 Series/Model 175 Computer System Reference Manual	60420000

This product is intended for use only as described in this document. CDC cannot be responsible for the proper functioning of undescribed features or undefined parameters.

CONTENTS

FIGURES

NOS represents a major advancement in network data-processing capabilities. Network capability begins with the distributed processing computer design of the CDC CYBER 170 Series, Model 172, 173, 174, and 175 Computer Systems. Distributed processing enables the central processor to devote full time to its computational tasks (for example, input/ output is performed by separate, smaller computers). Thus, optimum use is made of each element in the system.

HARDWARE CONFIGURATION

Although the devices supported by NOS can be combined in a variety of configurations, most NOS systems use the basic components shown in Figure 1-1.
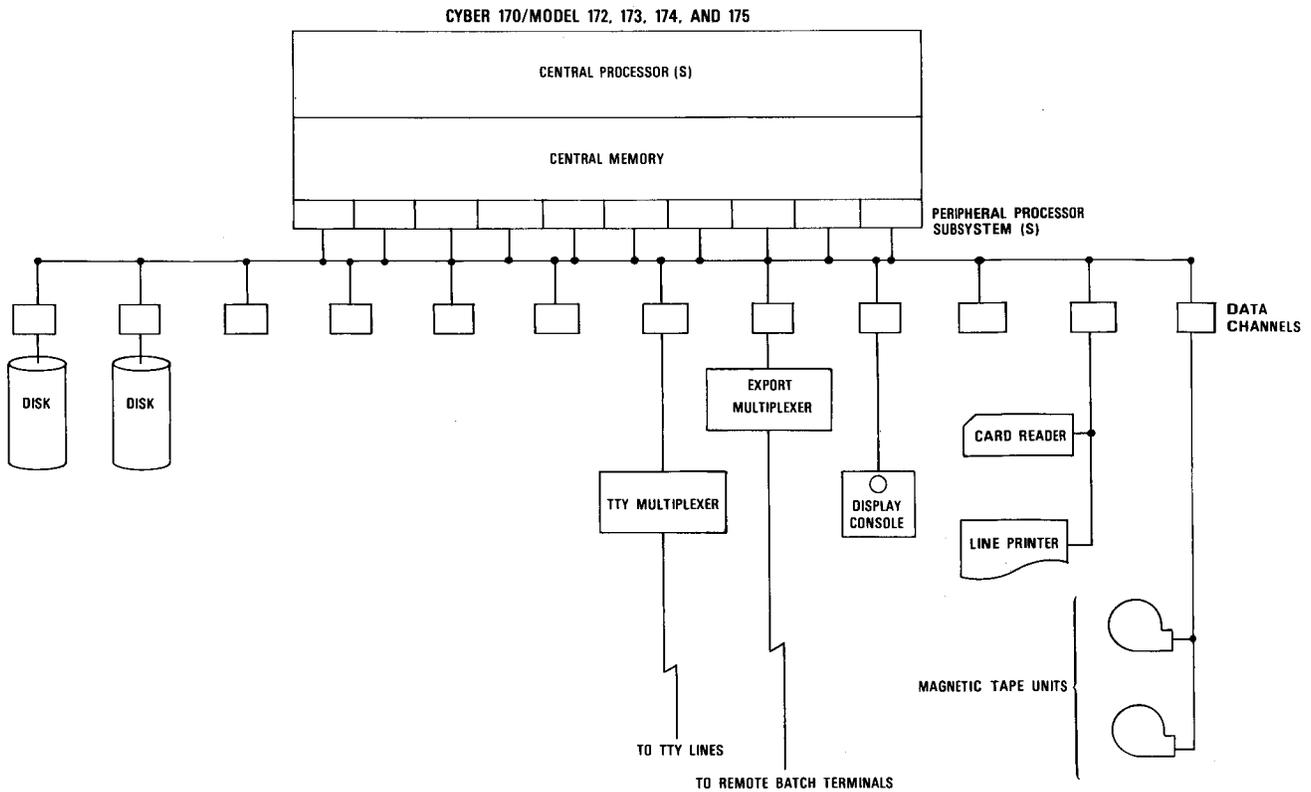

Figure 1-1. Basic NOS Configuration

CENTRAL PROCESSOR UNIT (CPU)

When an operation is to be performed on data, the relavant words are moved from central memory into the appropriate operating registers in the CPU. This unit is responsible for performing the high speed arithmetic operations of the computer (all multiplication, division, incrementing, indexing, and branching instructions, and some subtraction and logical operations). The central processor is linked to central memory through a block of 24 operating registers.

CENTRAL MEMORY

Currently active instructions and data blocks for a program are stored in central memory. Central memory consists of 32,768 to 262,144 60-bit words. In addition, many systems use extended core storage (ECS) as a mass storage supplement to central memory. The access time for information stored in ECS is slower than that for central memory, but faster than that for disk systems. As many as 2 million 60-bit words of ECS may be added to an NOS configuration.

PERIPHERAL PROCESSOR SUBSYSTEM (PPS)

The central processor shares processing responsibility and access to central memory with the PPS. The PPS consists of ten functionally independent computers; each equipped with its own memory (4096 12-bit words). All peripheral processors (PP) share common hardware for add/subtract, input/output, data transfer to and from central memory, and other necessary instruction execution hardware. NOS uses PPs primarily for transferring information between central memory and peripheral devices, and for system monitoring activities. The PPs relieve the CPU of the slower tasks associated with input/output, thus freeing the CPU to perform the relatively faster program calculations. A second PPS (consisting of four, seven, or ten PPs) can be added expanding the configuration to 14, 17, or 20 PPs.

DATA CHANNELS

For input or output, each PP accesses a peripheral device over a communication link called a data channel. Because several peripheral devices can be connected to each data channel, the number of devices at an installation is not necessarily limited to the number of channels available. Usually, only one PP at a time is permitted to use a given data channel for communication. However, a PP is not exclusively associated with a particular data channel and may use any data channel available for communication with an appropriate device.

DISPLAY CONSOLE

The system operator and NOS communicate through the system console, with its screen and associated keyboard. Two system programs provide the software interface between the console and the system. Their function is to maintain a current display of system and job status, and to process commands that the operator types at the keyboard. The system display program provides information pertaining to all jobs, whereas the job display program shows data from a single job. At the keyboard, the operator may assign equipment, exercise control over job scheduling and execution, initiate utility programs, and select displays.

PERIPHERAL EQUIPMENT

The many types of central site and remote devices NOS supports provide the flexibility required for the various processing modes.

Central site devices include:

- A card reader capable of reading 1200 cards per minute

- A card punch capable of punching 250 cards per minute

- Line printers capable of printing up to 2000 lines per minute

- Seven-track magnetic tape units with densities of 200, 556, and 800 characters per inch and transfer rates up to 160,000 characters per second, and 9-track magnetic tape units with densities of 800 and 1600 characters per inch and transfer rates of up to 320,000 characters per second

- Disk systems with capacities up to 894 million characters and data transfer rates up to 2.78 million bits per second

NOS controls communication between the central site and remote terminals via remote batch and time-sharing multiplexers. The remote batch network consists of low speed 200 User Terminals and 731-12 Remote Batch Terminals, medium speed 732-12 Remote Batch Terminals, and high speed 734 Remote Batch Terminals. Line printers for these terminals range from 300 lines per minute (low speed) to 600 lines per minute (high speed). Card readers range from 300 cards per minute (low speed) to 600 cards per minute (high speed). These terminals operate at transmission rates ranging from 600 to 4800 bits per second.

The time-sharing network consists of a variety of time-sharing terminals (10 to 60 characters per second) including the CDC 713 Display Terminal, the Teletype® Model 33, 35, 37, and 38 Terminals, the Memorex® 1240, IBM® 2741, Novar® 5-41, and Hazeltine® 2000 Terminals.

SYSTEM FEATURES

The software components of NOS consist of the operating system and its associated product set members. The operating system consists of programs that monitor and control the loading, execution, and output of user programs, manage storage assignments, handle input/output for peripheral devices, and control the sequence in which jobs are executed. The following are additional features.

- System parameters. System parameters allow each site to tailor NOS to its own needs by specifying such values as the CPU priority associated with all jobs of a given job type, the CPU time slice allotted to each job type, and the number of jobs that can reside simultaneously in central memory.

- Control statements. NOS provides a wide variety of control statements to accomplish such tasks as assigning a job to a certain account number and time limit, calling compilers into central memory, loading and executing jobs, creating permanent files and user library routines, and assigning peripheral equipment.

- Procedure file processing. A special NOS control language enables the programmer to save often-used sets of control statements on a procedure file. With this feature the user can define conditions under which a control statement is to be processed, specify the order in which processing is to occur, and initiate a series of complex operations with a single command.

- User control. The operating system provides two modes of user control: validation and profile. All users of the system are validated as their jobs enter the system. Interactive terminal users are validated via user number/ password. With nonterminal users, this validation is performed by an account card that contains the user number/password of the user. User profile control enables a site to set up three levels of user identification and job accounting. Related user numbers are grouped under one project number, and related project numbers are grouped under one charge number. Installations which select this feature require a user to enter his charge and project number before job processing proceeds. The feature can also be used to restrict the following.

  - The use of a project number to certain times of the day

  - The amount of CPU time a project is allowed to accumulate

  - The amount of time users are allowed to be connected to the system under a specific project number

- Permanent file management. The user can retain data for future use. With the user number/password system employed by NOS, a user's permanent files can be protected from unauthorized access. Several levels of permission can be granted to other users. Indirect and direct modes of file access are supported. By indirect mode, NOS creates a working copy of the file which the user can modify without affecting the contents of the original file. By direct mode, the user interfaces with the actual file.

- Families of permanent file devices. NOS groups the permanent file devices available to users into families. Normally, a system has only one family of permanent file devices. However, because families are interchangeable between NOS systems, several families may be active on one system. If, for example, system B provides backup service for system A and system A fails, system A users could continue processing because their family of permanent file devices could be transferred to system B without interrupting current operations on system B.

- Magnetic tape subsystem. The user can record data in any of eight magnetic tape formats. External BCD (7-track), ASCII (9-track), or EBCDIC (9-track) conversion modes for labeled or unlabeled tapes may be used. With ANSI standard labels, the user can define any of the following file structures.

      Single file, single reel
      Single file, multireel
      Multifile, single reel
      Multifile, multireel

- Auxiliary devices. If a user requires access to certain permanent files at infrequent intervals, NOS provides auxiliary devices as a mass storage supplement to his normal family of permanent file devices.

- Analytical aids. NOS furnishes informative and diagnostic messages, and maps and dumps of central memory to aid the programmer in debugging his programs.

- Editing capabilities. Special editing programs allow the modification and maintenance of system and user programs.

- Dayfile. NOS maintains a chronological record, called a dayfile, of each job processed and any problems encountered. This information is available to both the programmer and the system operator.

- Checkpoint/restart. The user can capture the running environment of a critical job, enabling him in the event of system or job failure to restart it from some intermediate point of execution.

- Disable terminal control. During normal time-sharing operations, the user can enter commands to terminate or suspend a program waiting for input, executing, or generating output. The disable terminal control feature allows a program to lock out terminal operator-initiated interrupts during critical phases of execution.

- Product set. NOS accepts programs written in many programming and special application languages. An extensive product set (refer to section 4) provides APL, BASIC, and FORTRAN time-sharing languages, a text editor, several batch language processors, and special applications programs. Conversion aids and diagnostic routines are also available.

All work to be performed by NOS is submitted in the form of jobs. Jobs are the user's means of communicating with NOS. Typically, a job consists of program, data, and control statements or commands.

Although program and data statements are optional, control statements or commands are required in all jobs. They identify the user, specify the system resources the job will require, and define the operations the system is to perform. The structure and origin of a job vary according to the type of processing to be performed.

TIME - SHARING JOB PROCESSING

The time-sharing executive controls the interaction between NOS and the user at a time-sharing terminal. The user enters his job via any one of the supported interactive terminals. Each terminal has a keyboard that resembles a standard typewriter keyboard. It provides the normal alphanumeric and special character keys plus several control keys. The output device may be a teletypewriter printer, cathode ray tube display, or paper tape punch.

A time-sharing job consists of commands or control statements, and optionally, program and data statements entered at the terminal. Time-sharing commands are entered one at a time, and may precede or follow program or data statements. They specify the program and file environment for the job.

A time-sharing job can be divided into the following phases.

- Log in

- Subsystem selection

- Program/data entry and execution

- Log off

The user logs into the system by following the telephone dial-up procedure for his particular terminal. He may either select one of the following subsystems or through validation have the subsystem selected automatically.

- BASIC. Used to compile and optionally execute programs written in the Time-sharing BASIC programming language.

- FTNTS. Used to compile and optionally execute programs written in the FORTRAN Extended (time-sharing option) programming language.

- BATCH. Used to process the control statements available to local and remote batch users.

- EXECUTE. Used to execute previously compiled programs.

At this point, the user may enter program or data statements, or any valid time-sharing command or control statement.

NOS commands, control statements, and time-sharing language specifications enable the user to communicate with the system and his programs. Through the input/output statements of the time-sharing languages, the user interacts with his program during execution. By interacting with the system and his program, the user examines the results of each job step and makes his next entry accordingly. He continues to enter commands or control, program, and data statements until he elects to terminate the job.

Included with the time-sharing module is the Text Editor program. Text Editor enables the user to edit program, data, or text files. Edit commands can be used to locate, modify, or replace lines or strings of characters within lines.

BATCH JOB PROCESSING

A batch job consists of a control record, and optionally, program and data records. Each record, in turn, consists of one or more control, program, or data statements. In addition to control statements, the control record may contain system control language commands and expressions.

Batch jobs differ considerably from time-sharing jobs. In a batch job all control statements are grouped and submitted as a control record. In the same manner any program or data statements that are submitted are grouped into records and placed after the control record in the job. Before submitting the job, the user defines all job steps and their sequence. At job termination the user examines the outcome of each job step and program results.

Batch jobs may be submitted at the central site or from remotely located terminals.

LOCAL BATCH JOBS

Local batch jobs are submitted at the central site directly via the card reader, or indirectly via a mass storage or magnetic tape file. The job can use the following NOS products. (Refer to section 4 for general descriptions.)

- ALGOL
- APEX
- APL
- BASIC
- COBOL
- COMPASS
- CYBER Record Manager
- FORTRAN Extended
- Math Science Library
- Modify
- Pert/Time
- SIMSCRIPT

- SIMULA
- Sort/Merge
- SYMPL
- Update

As NOS accepts the job, it stores the deck image on mass storage where it resides until the job's central memory requirements are met. When the job gains access to the CPU, NOS processes the control statements one at a time in the order specified by the job. During execution, NOS accumulates all listable and punched-card data on output files on mass storage. When the last control statement is processed and all job-related activity ceases, NOS terminates the job and routes the output files to the appropriate central site line printer or card punch.

REMOTE BATCH JOBS

Remote batch jobs may be submitted to the central site via remote batch or time-sharing terminals. NOS supports the 200 User Terminal and the 731-12/732-12/734 Remote Batch Terminals for remote batch operations. Each terminal provides an entry/display station, controller, line printer, and card reader. The user constructs and NOS processes remote batch jobs in the same manner as local batch jobs. The NOS products available for local batch jobs can also be used in remote batch jobs.

The user communicates with the central site via NOS Export/Import. Export/Import provides a set of commands which the user enters at the keyboard to submit a job via the card reader and list job output on the line printer.

Batch jobs can also be submitted to the central site via any time-sharing terminal. The job is created during a normal time-sharing session. The user may elect to precede all control, program, and data statements with line numbers. Otherwise, all statements are typed exactly as they appear on punched cards. As with standard batch jobs, system control language statements, and expressions may also be included in the control record.

By entering a time-sharing command, the user submits the job for batch processing. Special directives which the user can insert in the job direct NOS to reformat the job to conform to standard batch structure.

Once the job is reformatted, it is processed in the same manner as any other batch job. There is no direct communication between the job and the time-sharing terminal while the job is being processed. The user may specify that output generated during execution be directed to a central site output device or to a specific remote batch terminal. If the user wishes to receive job output at his time-sharing terminal, he can set up permanent files for that purpose.

TRANSACTION JOB PROCESSING

Through TRANEX the user processes transactions submitted to the central site via remote transaction terminals.

A transaction is defined as the communication of an administrative message, such as the information handled by a savings and loan office. Although there is no industry standard terminal, TRANEX is capable of supporting many types of terminals having the following characteristics.

- Buffered. Buffered terminals permit the setting up of buffer areas for data transfers.

- Multidrop. Multidrop terminals permit more than one terminal on a time-sharing network line.

- Polled. Each terminal is identified by a unique polling code. The executive polls each terminal to determine if it is ready for data transmissions.

With the ASCII code compatible terminal interface the user can enter the transaction subsystem from a time-sharing terminal. After logging in, the terminal may be switched to transaction mode (in which data is sent and received by the transaction subsystem) and back to time-sharing mode (in which data is sent and received by the time-sharing subsystem).

TRANEX provides an interface through which the user can create, access, and maintain one or more data bases and generate output to one or more transaction terminals.

Installations that require even greater processing flexibility than NOS provides may utilize the NOS-to-SCOPE communication feature of the CDC Cyberlink Interchange.

The Cyberlink Interchange is a subsystem of NOS that enables the user to route batch jobs over high-speed, wide-band telecommunication lines to any CDC SCOPE system operating within the CDC CYBERNET network. CYBERNET is a nationwide network of CDC CYBER 70 and 6000 series computer systems providing a wide variety of hardware/software configurations.

Figure 3-1 shows the basic components involved in NOS/SCOPE communications.

With Cyberlink the user can:

- Issue a single command to process batch jobs at a SCOPE system.

- Issue directives to reformat a job constructed at a time-sharing terminal into a batch job image.

- Save files on either system.

- Construct a job composed of records aggregated from both systems.

- Access special CYBERNET hardware, such as plotters and film printers.

- Access special CYBERNET applications programs for solving mathematical, statistical, financial, and engineering problems.
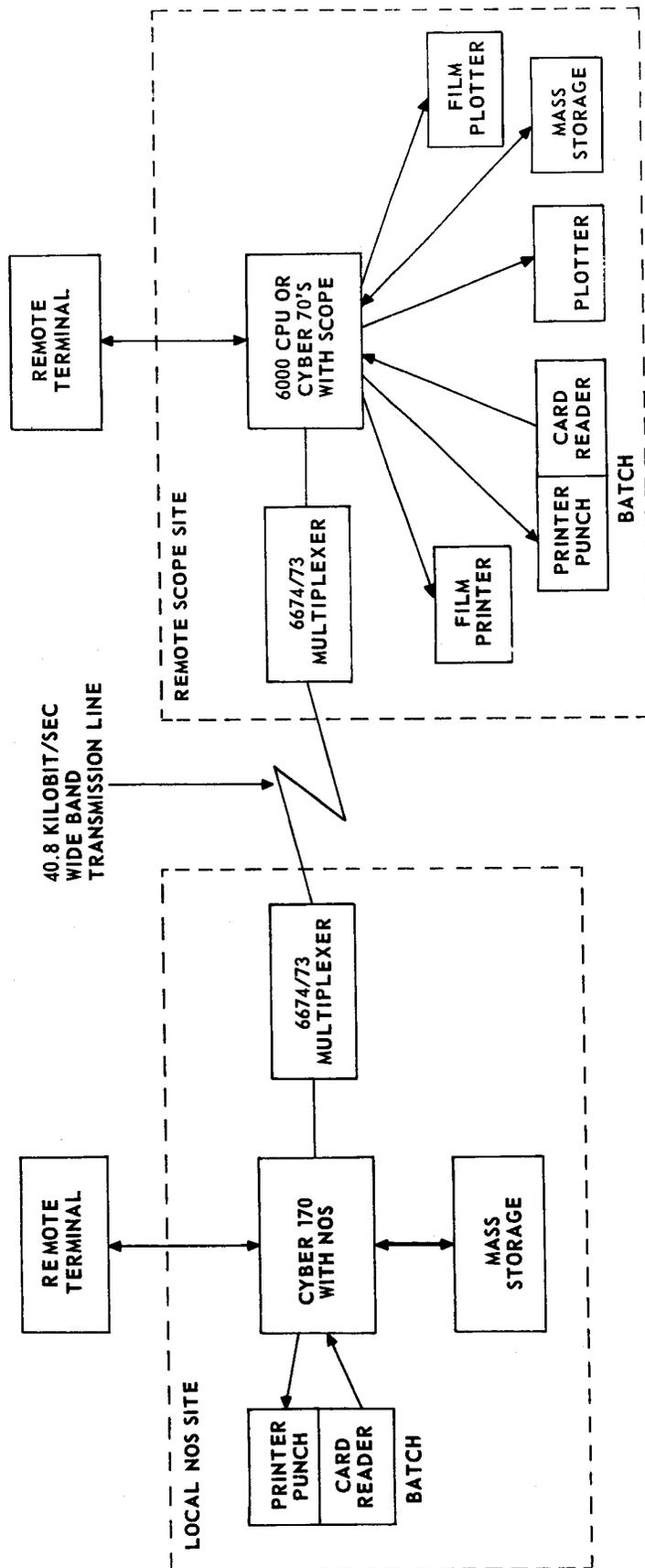
Figure 3-1. Components of NOS/SCOPE Communications

This section describes some of the software products available to the NOS user.

ALGOL

The ALGOL compiler translates source code written in a procedure-oriented ALGO-rithmic Language. The ALGOL language is primarily used for scientific applications.

APEX

APEX is a mathematical programming system that uses a matrix generator to arrive at optimal solutions to large linear programming problems.

APL

APL is an interactive interpreter that provides a large set of predefined functions used to solve a variety of mathematical and scientific problems. The name APL is an acronym derived from A Programming Language. APL can be used in time-sharing and batch jobs.

BASIC

The BASIC compiler translates source code written in Beginner's All-purpose Symbolic Instruction Code. The BASIC language is easy to learn, yet powerful enough to perform most of the operations available in more complex languages. The compiler can be used for time-sharing or batch jobs. Language specifications enable the time-sharing user to interact with his program during execution.

COBOL

The COBOL compiler translates source code written in COmmon Business Oriented Language. The COBOL language is designed to simplify the programming of business data processing operations. It is compatible with ANSI COBOL.

COMPASS

The COMPrehensive ASsembly System translates machine-dependent source code written in the COMPASS assembly language. Most NOS system routines and product set programs are written in COMPASS.

CYBER RECORD MANAGER

The CYBER Record Manager provides a common file access interface for FORTRAN Extended, COBOL, COMPASS, and Sort/Merge products. NOS also supports the in-dexed sequential and direct access file management capabilities of the CYBER Record Manager.

FORTRAN EXTENDED

FORTRAN Extended is a procedure-oriented language commonly used to solve mathematical and scientific problems. The FORmula TRANslation Extended language can be used to compile batch or time-sharing FORTRAN programs. It is compatible with FORTRAN IV.

MATH SCIENCE LIBRARY

The Math Science Library provides extensive mathematical problem solving capabilities. A FORTRAN programmer with a given problem can research the general area of mathematics involved, study his particular problem, and follow the library's documented direction concerning which of the 400-plus routines will best solve his problem.

MODIFY

Modify is a utility program that provides the user with a method of developing and maintaining a system of programs. With Modify, the user can create, edit, and manipulate program library files.

PERT/TIME

PERT/Time is a special application program that utilizes a time-oriented network structure which managers use to plan, monitor, and evaluate projects and programs.

SIMSCRIPT

The SIMSCRIPT compiler translates source code written in FORTRAN-like SIMulation SCRIPT. With SIMSCRIPT the user can prepare both simulation and nonsimulation programs required for solving mathematical and operations research problems involving many variables.

SIMULA

The SIMULA compiler translates source code written in ALGOL-like SIMUlation LAnguage. SIMULA is a general-purpose programming language for the simulation of discrete even systems.

SORT/MERGE

Sort/Merge is a special application program that accepts input from magnetic tape or mass storage and constructs, (according to user specifications) sorted output on magnetic tape or mass storage. Sort/Merge can be used for sort-only, merge-only, and sort-and-merge operations.

SYMPL

The SYsteM Programming Language is used by system programmers in writing com-
pilers and system software. SYMPL is a procedure oriented, ALGOL-like language
that is readable and concise. It uses self-explanatory English words and the familiar
notations of algebra and logic.

UPDATE

Update is a utility program similar to Modify. It is used to create, edit, and mani-
pulate program library files.

COMMENT SHEET

MANUAL TITLE ___CONTROL DATA® CYBER 170 NOS 1.0___

___General Information Manual___

PUBLICATION NO. ___60435900___ REVISION ___A___

FROM: NAME:___

BUSINESS
ADDRESS:___

COMMENTS:

This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number references and fill in publication revision level as shown by the last entry on the Record of Revision page at the front of the manual. Customer engineers are urged to use the TAR.

NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

FOLD ON DOTTED LINES AND 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
Technical Publications Department
4201 North Lexington Avenue
St. Paul, Minnesota   55112

ARH220

FOLD

FOLD