

The IBM logo, consisting of the letters "IBM" in a bold, sans-serif font, is positioned inside a dark, textured square.

Systems Reference Library

IBM System/360 Basic Programming Support and IBM Basic Operating System/360 Programming Systems Summary

This publication describes the general function, capabilities, and application of the programming support of IBM System/360 at the 8K and 16K levels. At these levels of support, Basic Operating System/360 consists of a comprehensive set of commercial, scientific, and process control programming aids operating under supervisory control and coordination of an integrated control program. The IBM System/360 Basic Programming Support consists of programs and programming aids (distinct from Basic Operating System/360 or Operating System/360) providing programming support for small card or tape configurations of IBM System/360.

For a list of associated publications and their abstracts, see the *IBM System/360 Bibliography*, Form A22-6822.



Preface

This publication serves as a general introduction to the IBM System/360 Basic Programming Support and the IBM Basic Operating System/360. Because many combinations of programming and computing facilities are possible with IBM System/360, no attempt is made in this summary to relate the programming facilities to specific machine requirements. Only the *minimum* machine configurations (see Appendices) for each of the programming facilities are included.

Summary descriptions of programs are provided in this publication. More detailed information can be found in other publications. These publications are abstracted and listed by form number in *IBM System/360, Bibliography*, Form A22-6822.

Copies of this and other IBM publications can be obtained through IBM Branch Offices. A form has been provided at the back of this publication for readers' comments. If the form has been detached, comments may be directed to IBM Product Publications, Endicott, New York 13764.

Contents

IBM System/360 Basic Programming Support and IBM Basic Operating System/360 Programming Systems	5
Introduction	5
IBM Basic Programming Support (8K Card)	11
Card Assembler	11
Card Report Program Generator (RPG)	11
FORTRAN IV Card	12
IBM Basic Programming Support Basic Tape System	13
Component Programs of the Basic Tape System ..	13
Tape Assembler	13
Tape Report Program Generator	13
Input/Output Control System (IOCS)	13
Control Programs	14
System Service Programs	15
Basic Programming Support Sort/Merge (1 Channel) and Sort/Merge (2 Channels) Programs (8K Tape)	16
IBM Basic Programming Support Utility Programs	17
Autotest (Requires 16K Bytes of Main Storage)	18
Fortran IV Tape (Requires 16K Bytes of Main Storage)	19
IBM Basic Programming Support Paper Document Programs	20
Input/Output 1418/1428 Program and Input/Output 1412/1419 Program	20
Dual 1419 Input/Output Program	20
Input/Output 1231 N1 Program	20
IBM Basic Operating System/360	21
Control Program	21
System Service Programs	22
Processing Programs	23
Sort/Merge Programs	23
Utility Programs (8K Disk)	23
Utility Programs (16K Disk and 16K Tape)	24
Autotest (8K Disk)	24
Autotest (16K Disk/Tape)	25
IBM 1070 Process Communication Supervisor	25
Appendix A. Basic Programming Support Minimum Machine Requirements	27
Appendix B. Basic Operating System/360 Minimum Machine Requirements	29
Index	31

IBM System/360 Basic Programming Support and IBM Basic Operating System/360 Programming Systems Summary

Introduction

A wide variety of programming support is provided for use with IBM System/360:

1. Operating System/360.
2. Basic Operating System/360.
3. Basic Programming Support.
4. System/360 Model 20 programming support.

All are designed to minimize the time and effort required by the user to produce and process programs. Programming support for System/360 Model 20 users is summarized in *IBM System/360 Model 20 System Summary*, Form A26-5889; Operating System/360 is summarized in *IBM Operating System/360 Concepts and Facilities*, Form C28-6535.

Summary descriptions of Basic Programming Support (BPS) and Basic Operating System/360 (BOS) programs are presented in this publication. BPS is a group of programs, each of which can be used without dependence upon any other program. Each BPS program serves a specific and limited application for minimum card and/or tape configurations.

By contrast, BOS furnishes centralized control of *all* programs. Programs are stored on a tape reel (usually file protected) or a disk pack, thus providing a high degree of program security. At the direction of the user, these resident programs are retrieved by the control program and brought into main storage when needed. This overall control results in automation of system operations with a minimum of operator intervention.

Basic Programming Support

BPS programs (Figure 1) are specifically aimed at providing support for minimum card and tape configurations in which:

1. A limited number of different programs is used, and fast transition from one program to the next is not required.
2. Control-program main storage requirements are held to a minimum.
3. Distinct tape reels are ordinarily used for distinct applications to maximize speed of phase retrieval and other such control-program operations.

4. Control-program processing requirements are held to a minimum, thus furnishing the user more processing time for his application (e. g., a sort run) than any other system described in this publication.
5. Purely peripheral operations are common. Here, many utility programs are provided for both single and multiple file-to-file operations.

Other BPS programs support applications involving optical and magnetic character readers. Specific programs are provided for:

- IBM 1412 and 1419 Magnetic Character Readers.
- IBM 1418 and 1428 Optical Character Readers.
- IBM 1231 N1 Optical Mark Page Reader.

These programs also provide IOCS (Input/Output Control System) facilities for card readers, card punches, tapes, printers, etc., that are used in the paper document applications.

BPS furnishes translators for the following languages:

- Assembler.
- Report Program Generator.
- FORTRAN.

Most BPS programs require only 8,192 bytes of main storage. Others, such as the FORTRAN IV (Tape) program, require 16,384 main storage positions. Additional storage, above the minimum required, may be utilized by BPS programs.

Basic Operating System/360

Three resident systems are provided in BOS, each designed to provide operating system capabilities for 8K and larger System/360 configurations:

- Disk-Resident System (8K)—Figure 2.
- Disk-Resident System (16K)—Figure 3.
- Tape-Resident System (16K)—Figure 3.

The 8K and 16K disk-resident systems require program residence on an IBM 2311 Disk Storage Drive.

Figure 1. Facilities of Basic Programming Support

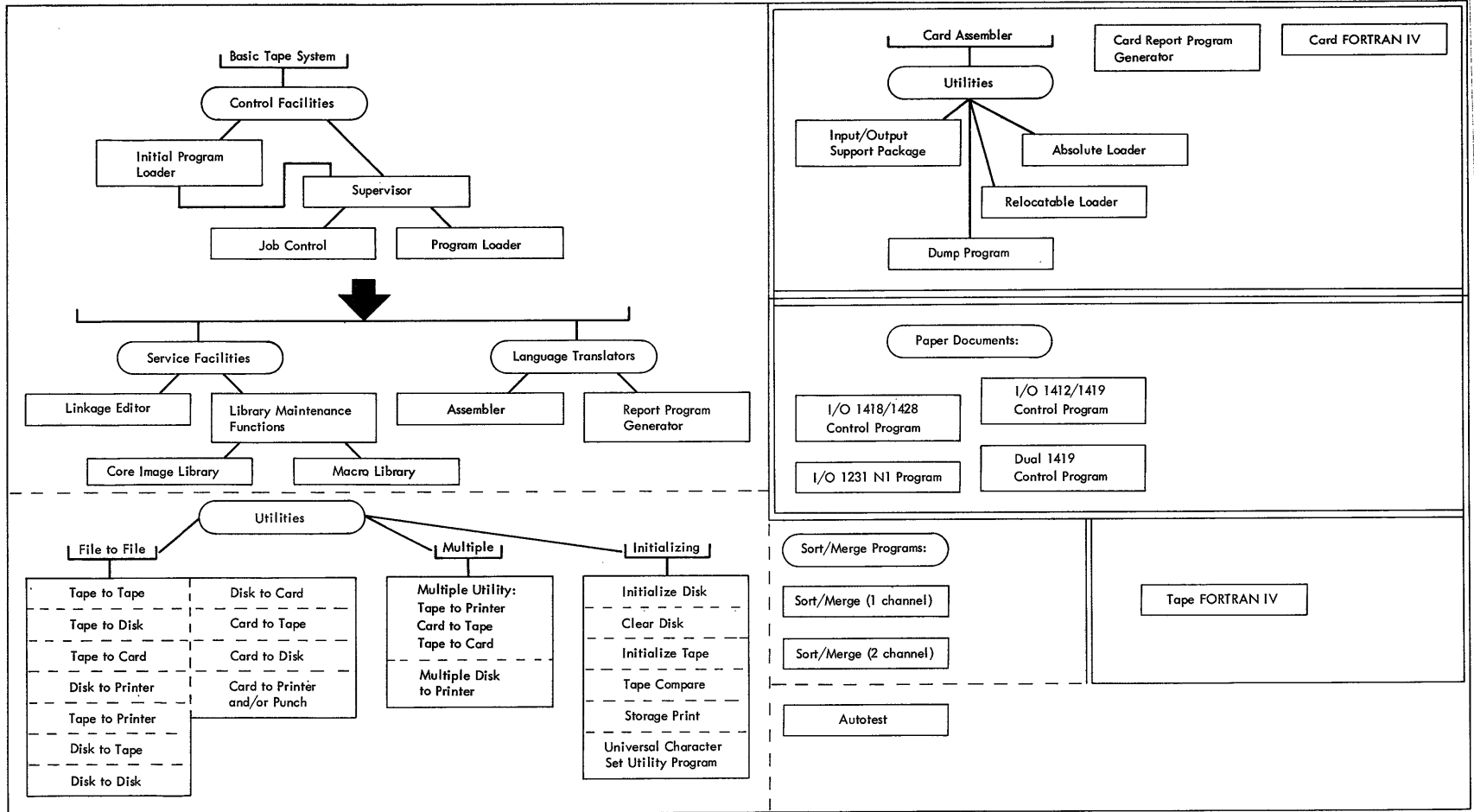


Figure 2. Facilities of Basic Operating System/360 (8K Disk)

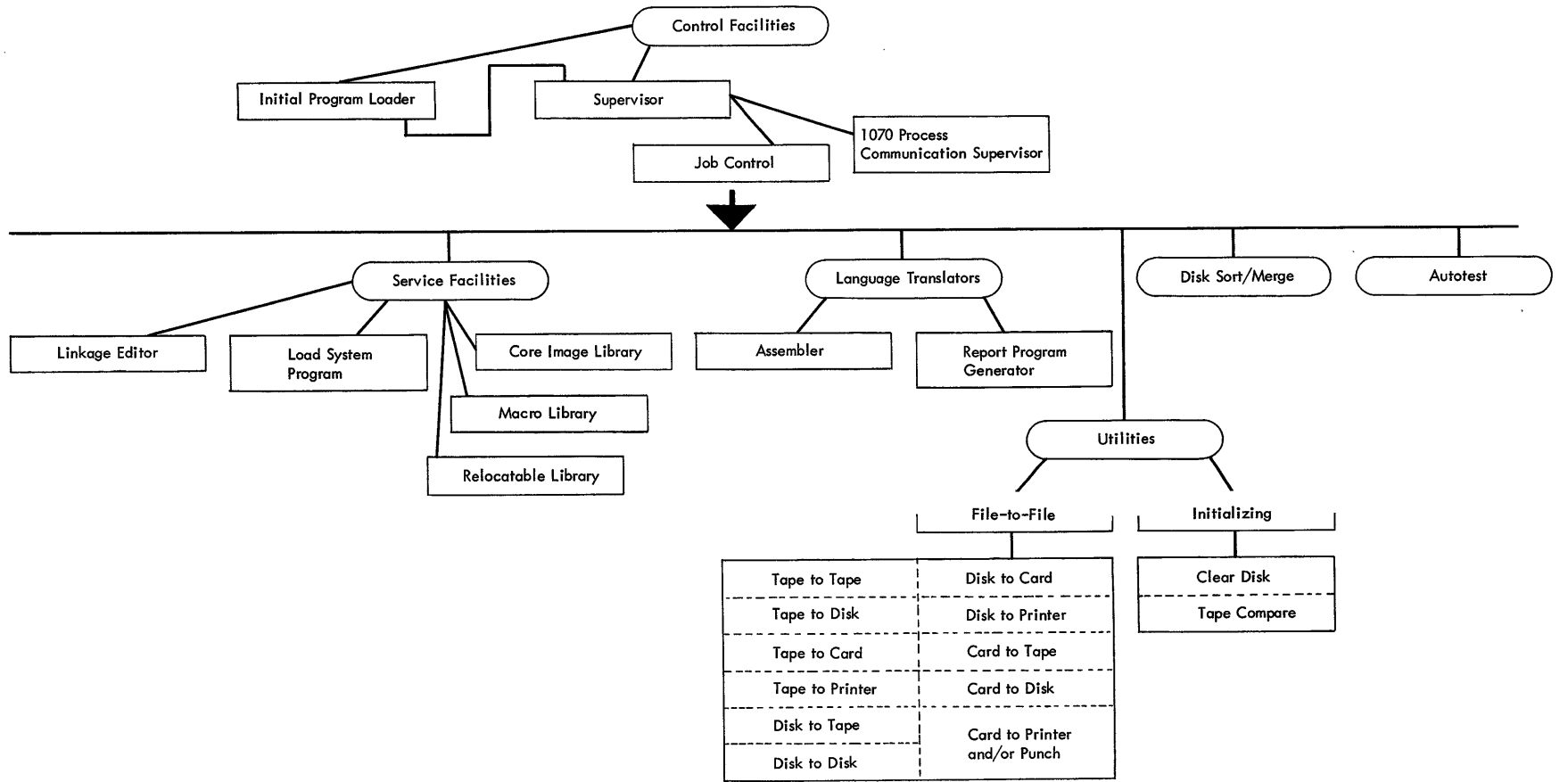
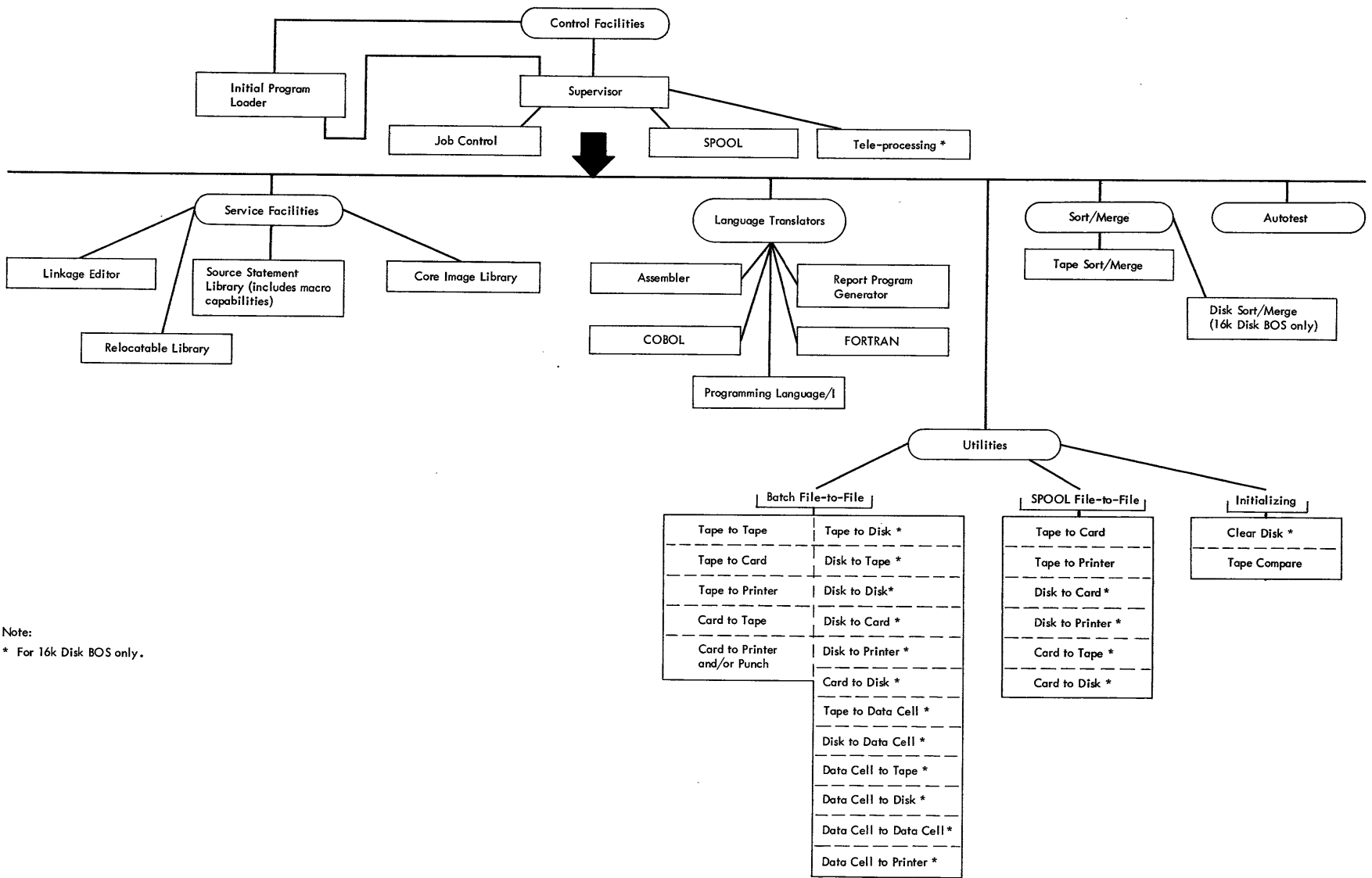


Figure 3. Facilities of Basic Operating System/360 (16 Disk and 16K Tape)



Note:
* For 16k Disk BOS only.

The 16K tape-resident system requires one IBM 2400-series magnetic tape drive for program residence.

BOS support is specifically aimed at small-to-medium tape or disk systems with the following operational requirements:

1. A limited number of different programs is used (8K disk), or a larger number of programs is used (16K disk or 16K tape); and, in either case, fast transition from one program to the next is often an installation requirement.
2. Control-program main storage requirements are held to a minimum (8K disk). Slightly more storage is used by the 16K tape or disk control programs to furnish additional functions, which can include:
 - a. Storage protection and interval timer support.
 - b. Scheduling of I/O devices and channels to achieve maximum overlap of computation and I/O operations.
 - c. SEEK overlap for multiple-disk or data-cell storage devices.
 - d. Automatic channel switching for tapes switchable over two channels.
 - e. Flexible correspondence of logical device names to physical I/O devices:
 - Permanent description of all attached I/O devices, whether operational or not.
 - Multiple names for a single physical device.
 - Multiple physical devices corresponding to a single name.
 - f. Shared-volume protection (disk packs or data cells used by the supervisor and/or problem programs).
 - g. Full system-to-operator and operator-to-system message capability using the 1052 console.
3. A single system volume can contain all control-program functions, language translators, sorts, user programs, etc. For fast program retrieval (and fast phase retrieval for multiphase programs), this is a major objective of the system volume layout.
4. Control-program processing requirements are small on the 16K Disk and Tape Systems, although not as small as in BPS. (The 8K disk system requirements are comparable to those of BPS.) The additional processing time is required for such functions as those listed in item 2.
5. Three comprehensive and efficient library services are offered in both the 8K and 16K systems:
 - a. Core image library, from which programs can be retrieved (with no further modification) and executed at once.

- b. Relocatable library, from which mathematical subroutines and IOCS modules can be retrieved by the linkage editor (either automatically or in response to simple control statements).
- c. Source-statement library (16K) or macro library (8K), from which language translators can retrieve source text (e.g., macro definitions) to be inserted into source programs during compilations.

Each installation can add to, delete from, display, or punch any or all of these libraries, both at initial-generation of the system and thereafter as its application mix changes.

6. Other principal characteristics of the 16K systems are:
 - a. All five language translators: Assembler, Report Program Generator, COBOL, FORTRAN, and Programming Language/I.
 - b. Concurrent peripheral operations (SPOOL) in a 32K or larger system.
 - c. Concurrent teleprocessing applications (in a 32K or larger disk system).

In each of these resident systems, all installation programs are built, cataloged, and retrieved by the control program. The control program provides common services for:

1. Program loading.
2. System/360 interruption facilities.
3. Scheduling and execution of all I/O operations.
4. File initialization and label processing.
5. Operator communication.
6. End-of-job and job-transition control.
7. I/O device errors.

During system operation, a portion of the supervisor (a component of the control program) remains in main storage at all times. All other programs reside in external storage (disk or tape, depending upon the resident system used), and are loaded into main storage for execution when needed. Both IBM and user-supplied programs are cataloged into the libraries of the resident device by the control program. Cards identify the cataloged programs to be executed; they are read from a card reader by job control before each job starts.

Each cataloged program is loaded into main storage and executed in a specified sequence. For example, the execution of an object program can be initiated as soon as it is assembled or compiled. Hence, in this BOS environment, a minimum of operator intervention

is required, and a high degree of program security is maintained.

Infrequently used control-program routines are loaded, when needed, from external storage into a transient main storage area included in the supervisor. The main storage requirement of the operating system is thereby kept to a minimum. When a routine is loaded into the transient area, it overlays the routine that was previously stored there. The overlaid routine may subsequently be called in from external storage by the control program when again required.

Any operating system can retrieve programs randomly from system residence, even if the latter is a tape drive. However, far greater efficiency is obtained when program residence is on an IBM 2311 Disk Storage Drive; transition from one program to the next is achieved in minimum time. The 8K and 16K disk BOS systems furnish this (and other) advantages of disk system residence.

The 16K tape BOS provides operating system facilities for tape-oriented configurations. Optimum operation of this system is achieved when frequently used programs are stored at the beginning of the resident tape, just after the transient control-program routines.

Language translators available in BOS are:

- Assembler.
- Report Program Generator.

- COBOL (16K disk or 16K tape only).
- FORTRAN (16K disk or 16K tape only).
- Programming Language/I (16K disk or 16K tape only).

The 16K disk and 16K tape BOS systems will provide the additional capability of SPOOL (Simultaneous Peripheral Operations On Line). SPOOL facilities are designed to maximize total job throughput. For example, if one program does not require the full I/O capacity of the system, other peripheral operations can concurrently use the I/O facilities that would otherwise be idle. Also, a program that normally uses slow speed I/O devices (such as printers, card readers, and card punches) can direct its output to high speed I/O devices (such as magnetic tape or disk units). Later, a SPOOL operation can transfer this data from the high-speed to a low-speed I/O device if concurrent programs do not need this pair of I/O devices.

SPOOL operations, therefore, have the effect of:

1. Achieving a better balance between the processing and I/O capacity of the system.
2. Allowing the processing requirements within an installation to proceed without doing slow speed I/O operations on-line.

16K disk BOS also provides the facility to perform a Tele-processing application concurrently with a stack of other jobs.

Card Assembler

The card assembler provides for translation of programs written in basic assembler language into IBM System/360 machine-language object programs.

The assembler language is a flexible, symbolic language that is machine-oriented and applicable to both commercial and scientific problems.

This card assembler is a two-pass program. Input to pass 1 consists of source program statements (written by the problem programmer) punched into cards. Pass 1 partially translates the source program assembler statements into object code. The output of pass 1 is recorded on some external storage medium, which is then used as input to pass 2. (Either punched cards or magnetic tape can be used. If punched cards are used, the system is called the *Card Option System*; if tape is used, the *Tape Option System*.) The final object program is then punched into cards or written on magnetic tape, as directed by the user.

In the card assembler, there is no separate control program as in the basic tape system and BOS. Rather, I/O utility subroutines are called by the user program to perform I/O services, thus freeing the user from detailed programming of I/O functions. Other utilities perform service functions such as loading object decks and dumping main storage in the event of a program error. No macro instruction capabilities are available in this card assembler.

Utility Programs

Input/Output Support Package

The input/output support package consists of a modular set of subroutines that enable the user to use input/output devices. (A *module* in the input/output support package is a logical sequence of coding that either sets up or executes one I/O function.) These are routines to:

1. Read or punch a card.
2. Write on the message (or printer) device.
3. Sense information from a device.
4. Single space on the message or printer device.
5. Skip to channel 1 on the printer.
6. Read or write tape.

7. Write a tape-mark.
8. Rewind tape.
9. Backspace tape a record or file.
10. Forward space tape a record or file.
11. Read tape backward.

Loaders

Once the object program is produced, it must be loaded into the system for processing. The absolute and relocatable loaders are provided for this purpose.

The absolute loader loads each program segment into storage at the address assigned to it by the assembler. Control is then transferred to one program segment (possibly several such segments have been loaded) to begin program execution.

The relocating loader can load program segments into storage at locations other than those assigned by the assembler. It completes linkages among the segments so that each segment can refer to certain addresses in other segments.

Dump Program

The dump program provides a listing of contents of:

1. All or part of main storage.
2. The general registers.
3. The floating-point registers.

The dump program provides a number of basic formats for this listing. Information concerning the existing conditions of the program environment can be obtained at an intermediate point or at the normal end of job execution.

If the program is stopped at an intermediate point so that a "snapshot" of existing conditions can be made, then the program continues undisturbed after the snapshot operation has been completed.

Card Report Program Generator (RPG)

RPG is a problem-oriented, easy-to-use programming language, whose principal function is to accumulate data from existing files and generate reports from this data. Program specifications written by the prob-

lem programmer are compiled by the RPG translator. RPG compilation produces an object deck containing:

1. Machine language instructions.
2. Storage area assignments.
3. Linkages to routines for checking and for input/output operations.

Some of the capabilities of the card RPG compiler are:

1. The object program can obtain data records from as many as three card input files.
2. The object program can match records in as many as three card files to govern processing of the report.
3. Input records can be checked for sequence.
4. The object program can search tables for data. It can use this data to produce the report.
5. The program can branch to a subroutine that has been written in a language other than RPG, perform calculations, and return to the RPG program.
6. The report can be produced on as many as three printer or punch output files.

Here, as in the BPS card assembler, the compilation process produces an independent program which

needs no additional IBM-supplied routines in order to be loaded and executed.

The source program is entered into the system (in punched card form) along with control cards and the RPG processor. The RPG processor then compiles (or generates) the object program. The user can then specify that an object deck of the program be produced, or that execution of the program is to begin, or both. Hence, object programs can be retained for use in later runs without recompilation.

FORTRAN IV Card

This program is similar to the FORTRAN IV (tape) Basic Programming Support package (see section devoted to this program). In the FORTRAN IV (card) package, however, disk and tape units are not supported. The FORTRAN IV card compiler obtains source program input from cards and produces a punched-card object deck output.

The FORTRAN IV (card) language is a subset of the Operating System/360 FORTRAN language.

IBM System/360 Basic Programming Support Basic Tape System

For small tape systems, the basic tape system offers the following functions:

- Perform all supervisory and physical I/O functions, independent of all problem programs.
- Compile user programs (problem programs) written in the job-oriented RPG language.
- Assemble problem programs written in assembler language.
- Assemble control programs for use with, and controlling the operation of, problem programs.
- Link together separately assembled program sections and/or subroutines into a single program that can be executed without reassembling.
- Add user-written macro definitions to the system tape so they are available for inclusion in problem programs at assembly time.
- Build a tape of problem program(s) in loadable form, to enable batch execution.
- Maintain the system tape.

System Tape

A system tape, tailored to the installation, is built by using a master tape (available from IBM) and a set of control cards (supplied by the user and describing his requirements). The major functions of the system tape are assembling, RPG compiling, and preparation of problem programs for execution. Note that the system tape is not used when these problem programs are being executed. However, problem programs can be edited onto a tape for subsequent loading, thus providing limited tape-resident facilities.

Component Programs of the Basic Tape System

Tape Assembler

The tape assembler, which is substantially more powerful than the basic assembler and offers macro capabilities, translates source programs written in a subset of the Operating System/360 assembler language. In

general, one machine instruction is produced for each assembly language statement provided by the programmer. However, macro instructions can be used to specify entire routines for commonly used operations. For example, IBM provides macro definitions to support IOCS and supervisory functions. Users can provide their own macro definitions (e. g., to compare two data fields) and add them to the system library.

Tape Report Program Generator (RPG)

RPG is a problem-oriented, easy-to-use programming language, whose principal functions are to accumulate data from existing files, and/or update these files, and/or generate reports from this data. Some of the capabilities of the tape RPG compiler are:

1. Produce listings.
2. Perform a wide variety of calculations.
3. Use multiple files.
4. Search tables.
5. Update files.

Thus it is possible to produce reports ranging from simple listings from cards or tape to complete jobs such as payroll and accounts receivable. Special coding sheets are used by the programmer to describe the job to be performed. The programmer describes the type of input data, the calculations to be performed, and the type of output desired. The information from the coding sheets is compiled to produce object programs that are ready for execution.

Input/Output Control System (IOCS)

The capability for logical record handling is provided via I/O macro instructions. An IBM-supplied macro definition in the system library is provided for each macro instruction. Physical record handling for assembled programs (and also programs compiled by tape RPG) is controlled by the physical IOCS section of the supervisor.

A *logical record* is one that is identified from the standpoint of its content, function, and use rather than its physical attributes. On the other hand, a *physical record* is identified by the manner or form in which it is stored and retrieved.

Logical I/O macro instructions are generated into each assembled object deck. These routines then use routines in the supervisor for the physical transfer of records.

Control Programs

Supervisor

In the basic tape system, the entire supervisor always resides in main storage during problem program execution. (In the Basic Operating System/360 packages, only the nucleus of the supervisor remains in main storage.) The supervisor is generated during assembly, either with problem programs or separately. One supervisor can be used with each set of programs, or a generalized supervisor can be used for all programs of an installation.

The functions performed by the supervisor are:

1. Interruption handling.
2. Channel scheduling.
3. Device error recovery.
4. Operator communication.
5. Program retrieval.
6. End-of-job indication.

All functions except interruption handling are available to the problem programmer by issuing macro instructions. The programmer is relieved of considering machine interruption conditions because of the interruption handling functions of the supervisor.

A communication region is provided in the supervisor area of main storage. This region contains information useful to both problem programs and the supervisor. Information such as the system configuration and addresses for certain user routines (i. e., timer routine, program check routine, etc) are contained in the communication region.

Job Control

Job control prepares jobs to be run. This function is performed between jobs, and job control is not present while jobs are being run.

The job-control program is distributed in relocatable form. It must be relocated, via the linkage editor, to run with the assembled supervisor just like any problem program.

The functions obtained from job control by means of control cards are:

1. Indicate the name of the job to be run.
2. Assign required input/output device addresses to symbolic names.

3. Allow restarting of previously checkpointed jobs.
4. Edit and store label information for later use by the tape label routines.
5. Place today's date in the communication region.
6. Set user program switch indicators for use by the problem program.
7. Store machine characteristics in the communication region.
8. Cause a machine wait (PAUSE) prior to execution of the next job.
9. Log (or print) job-control cards.
10. Indicate execution is to begin.

Execution of job control is required for each job only if a system tape is used. Thus, if the supervisor is loaded from cards, job control need be used only when I/O assignments must be changed, etc.

A restart facility is available in job control for checkpointed programs. By use of the CHKPT (Checkpoint) macro instructions, checkpoint records can be obtained as a program is processed. These records provide sufficient information so that the program can be restarted from specific points. This allows restarting a long program's execution at some intermediate point rather than requiring that the entire program be reprocessed. For example, the program might be interrupted because some other program of high priority requires immediate processing.

Initial Program Loader (IPL)

If a system tape is used, the IPL program must be loaded from this tape as follows: the load-unit switches on the system console are dialed to the unit number of this tape, and the load key is pressed.

If a system tape is not used, the IPL program is loaded from cards in the same manner (i. e. the unit number of the card reader is dialed into the load-unit switches).

Besides loading itself, IPL provides the following functions:

1. Clears all of main storage except that used by the IPL program.
2. Clears the general registers.
3. Loads the supervisor and program loader.
4. Stores the channel and unit number of the IPL device for the appropriate symbol in the symbolic device entry table. (This information is obtained from the loading procedures.)
5. Places the machine in wait state to allow operator intervention. This allows an object-time change of control-card reader assignment.

6. Places the machine in the problem program state and transfers control to the program loader for loading problem programs or job control.

Program Loader

The program loader loads absolute object decks into main storage without modification. That is, there is no relocation requirement or external-symbol linkage in the program to be loaded. In either of the latter situations, the program must first be linkage-edited before it is loaded.

Under supervisor direction, the program loader uses the input/output functions of the supervisor to load programs from either card or tape. After loading is completed, the program loader branches to the program just loaded. The program loader can be used during program execution by issuing a FETCH macro instruction. When this macro instruction is issued, the program loader must be available in core storage. Therefore, it must not be destroyed by the problem program until the last overlay has been read.

Initially, the program loader is brought in by IPL along with the supervisor. The program loader is located adjacent to the last location of the supervisor area.

System Service Programs

Linkage Editor

The linkage editor, which is contained on the system tape, combines separately assembled modules into one operable program. It also performs any relocation necessary for loading an assembled program with the program loader. Compiled programs, therefore, can be prepared for loading into any core storage locations with full backward/forward referencing between symbols in different control sections and with limited referencing between any separately assembled modules.

The linkage editor is not always needed. For example, if there is only one phase in the assembly, originated beyond the supervisor and program loader,

then the functions of the linkage editor are not required.

Libraries

Two libraries are stored on the system tape:

1. The core image library.
2. The macro library.

The core image library is a group of programs, each comprising one or more phases. Each phase is the image of (i. e., identical to) its form in main storage. The smallest complete unit that can be referenced is a phase.

The core image library contains IBM programs that provide for operation of the Tape Assembly System. These programs are:

1. Job Control.
2. Linkage Editor.
3. Assembler.
4. Library Maintenance Programs.
5. Library Service Programs.
6. Loadable Tape Programs.
7. RPG Programs.

The macro library is a group of both IBM and user macros. A macro directory is provided that contains identifiers for the macros in the corresponding macro library sections.

Library Maintenance Programs

Library functions are used for maintaining programs and macro definitions on the system tape, and for adding user-written macro definitions to the system tape. Two major library functions are provided:

1. *Maintenance* functions are used to enter or delete phases in the core image library and macro definitions in the macro library.
2. *Service* functions for the core image and macro libraries translate information from a particular library to printed (displayed) or punched output. The directory of each library can be displayed.

Basic Programming Support Sort/Merge (1 Channel) and Sort/Merge (2 Channels) Programs (8K Tape)

The IBM System/360 BPS 8K Tape Sort/Merge Programs enable the user to sort files of random records, or merge multiple files of sequenced records into one sequential file. The control-data information can be contained in as many as twelve fields in each record. The programs assume that input records for a sort operation are in random sequence. However, if any inherent sequencing exists within the input file, the programs take advantage of it. Records can be sorted or merged into ascending or descending sequence, and an individual sequence can be specified for each control-data field. The output sequence for a merge-only operation must be the same as the input sequence.

There are two sort/merge programs in the basic programming support (8K tape) environment:

1. Sort/merge (1 channel).
2. Sort/merge (2 channels).

These sort/merge programs are generalized programs in the form of assembled object decks when they are received by the user. In addition, the facility to create a program tape is provided. Control statements tailor the generalized sort/merge program to the user's specific application. The control statements are punched into cards and inserted into the program deck before it is loaded, if the program is loaded from cards. Alternatively, control statements are read from the card reader if the program itself is loaded from magnetic tape.

The 2-channel sort/merge program provides better performance than the 1-channel program when:

1. Two selector channels are available in the system.
2. One selector channel and 16,384 or more positions of core storage are available in the system.

The sort/merge program:

1. Translates mnemonic control-card information that describes the file parameters for each input and output file.
2. Sorts a single input file.
3. Merges a minimum of one to a maximum of five input files.
4. Allows multivolume input and/or output.
5. Provides for specification of an alternate input drive and an alternate output drive.
6. Provides for input from and output to 7-track and/or 9-track magnetic tapes.
7. Provides checkpoint, interrupt, and restart procedures during the merge phase (phase 2) of a sort operation.
8. Provides exits to user-written routines.
9. Prints out:
 - a. The control card information (optional).
 - b. Record counts at the end of phase 1 and phase 2 (optional).
 - c. Necessary diagnostics.
10. Provides the option to bypass unreadable data blocks or to indicate the need for operator intervention.
11. Sequence-checks the records during the final merge pass.
12. Sorts on each control field independently, permitting the user to specify either ascending or descending sequence for each individual field.
13. Processes standard System/360 volume and file labels.
14. Provides standard System/360 IOCS error procedures.

IBM Basic Programming Support Utility Programs

IBM supplies several types of utility programs to perform:

1. Transfer of information from one I/O device to another (file-to-file).
2. Transfer of information among several I/O devices (multiple file-to-file).
3. Initializing of tape or disk volumes and the universal character set feature.
4. Main storage print.

Each program handles a particular type of job (e. g., the tape-to-printer program will print any file on any single printer). No symbolic assembly is necessary for the operation of a program. To handle a specific job, the generalized program is modified by control information entered from cards or from the printer keyboard.

Control cards are free-form in that the parameters can be punched in any order. The programs assume certain options if they are not specifically selected on utility-control cards.

Eleven file-to-file programs are provided. These are:

1. Tape to tape.
2. Tape to disk.
3. Tape to card.
4. Tape to printer.
5. Disk to tape.
6. Disk to disk.
7. Disk to card.
8. Disk to printer.
9. Card to tape.
10. Card to disk.
11. Card to printer and/or punch.

The two BPS multiple file-to-file utilities are:

1. Multiple utility (any combination of the following can be performed).
 - a. Tape to printer.
 - b. Card to tape.
 - c. Tape to card.
2. Multiple disk to printer (16K storage required). With this program, up to three disk-to-printer operations can run concurrently.

The initializing utilities are:

1. *Initialize Disk*. This program can prepare up to five complete disk packs for use on IBM 2311 Disk Storage Drives.
2. *Clear Disk*. This program clears one or more areas of a disk pack; the area to be cleared can be as small as one track or up to a maximum of a complete disk pack.
3. *Initialize Tape*. This program places volume labels on tape.
4. *Tape Compare*. The tape compare program compares files from two or more tapes to ensure that the files are identical.
5. *Universal Character Set Utility*. This program conditions the IBM 2821 Control Unit for a change in chain or train arrangement on the associated IBM 1403 Model 2, 3, or N1 Printer that uses the universal character set feature.

The storage print utility program produces a print-out of storage and registers to aid the user in locating a program malfunction.

Autotest (Requires 16K Bytes of Main Storage)

Autotest provides debugging capabilities for assembled decks as they are being tested. All Autotest services are requested by control cards, whose removal restores a program to its normal condition. These control cards also facilitate remote testing. The programmer can preplan his test and have it run with a minimum of operator intervention. In addition to this, a user can batch as many individual test jobs as he desires and get extensive diagnostics and testing services in each with just one load procedure.

For each program tested, Autotest does the following:

- Clears core storage before the object program is loaded.
- Monitors the execution of the object program.
- Prints out main storage contents if the program comes to an abnormal end of job.

Although Autotest tests programs which will operate in 8K bytes of main storage, each Autotest run requires 16K bytes of main storage.

FORTRAN IV Tape (Requires 16K Bytes of Main Storage)

FORTRAN IV is a language for solving scientific and engineering problems. Problem solutions can be stated in a concise, mathematical notation, which is widely used and easily understood.

The System/360 BPS FORTRAN IV Tape System operates independently of any other programming system. This FORTRAN system provides an extensive range of capabilities including:

- The FORTRAN compiler.
- Facility for executing FORTRAN object programs.
- System maintenance facilities to simplify tailoring a system to individual installation requirements.
- A variety of system library facilities, including mathematical and service (utility) subprograms.

- A simplified set of operating instructions and diagnostic messages that minimize operator intervention during program processing and machine operation.

To simplify the compilation and execution of FORTRAN programs, the FORTRAN system provides the user with a flexible, yet concise, control language. This language is expressed by control statements which specify:

1. Run type (e. g., one or more source programs).
2. The processing to be performed (e. g., compile and execute).
3. The output desired (e. g., source program listing).

If certain control information is not specified by the user, the FORTRAN system operates with a set of assumed specifications.

IBM Basic Programming Support Paper Document Programs

Input/Output 1418/1428 Program and Input/Output 1412/1419 Program

The 1418/1428 control program is a complete set of supervisory routines for System/360 applications using the IBM 1418 or 1428 Optical Character Readers. The supervisor used is supplied as a source deck written in IBM System/360 basic assembler language. It is combined with the user's 1418 or 1428 processing programs to provide supervisory facilities for most typical optical character recognition (OCR) applications.

The 1412/1419 control program provides the same facilities for magnetic character recognition as does the 1418/1428 control program for optical character recognition. The 1412/1419 package uses either an IBM 1412 or 1419 Magnetic Character Reader.

Both of these program packages provide the following functions:

- Supervises execution of user-supplied channel programs (channel command words) for card readers and punches, printers, magnetic tape units, and disk drives.
- Guarantees maximum OCR and MICR document throughput by maintaining control over a document buffer area distinct from the document input area.
- Handles all interruption conditions.
- Provides communication between the user's problem program and the machine operator via program-generated messages.

The input/output 1418/1428 and 1412/1419 programs include:

- *Supervisor*. Includes I/O routines for 1418 and 1428 Optical Character Readers or 1412 and 1419 Magnetic Character Readers to handle all of the functions just described.
- *Job Control*. Prepares programs for execution, including symbolic device address assignment.
- *Program Loader*. Loads the user's problem program into main storage and, for job processing, loads the next program upon request.
- *IPL Loader*. Loads the supervisor and program loader into main storage to begin processing.

Dual 1419 Input/Output Program

An input/output dual 1419 program is provided that allows operation of two 1419's. The functions of this program are basically the same as those of the input/output 1412/1419 program.

Input/Output 1231 N1 Program

The input/output 1231 N1 program is a set of core-resident, independent, I/O subroutines and a supervisor program which provide simplified I/O control of the IBM 1231 N1 Optical Mark Page Reader (OMPR) and other I/O devices. The OMPR provides the facility of direct entry of data into a System/360 without the necessity of initial conversion of the data into punched cards.

The IBM 1231 N1 Optical Mark Page Reader reads positional marks made by a #2 lead pencil on paper documents. These marks are read and converted into machine acceptable form by the 1231 N1 and entered directly into the data processing system.

Some of the applications suitable for OMPR processing are:

- Order Entry.
- Inventory Control.
- Payroll.
- Accounts Payable.
- Route Accounting.
- Market Surveys.
- Policy Applications.
- Medical Records.

The input/output 1231 N1 program and user-supplied processing routines combine to provide a complete data-handling program for an OMPR application. Although the program is primarily oriented toward providing maximum 1231 N1 performance, it also provides read/write and control capabilities for tape units, printers, punches, etc, which might be associated with the particular OMPR application.

IBM Basic Operating System/360 is provided at the:

1. 8K level for disk.
2. 16K level for disk.
3. 16K level for tape.

The 8K disk resident version is designed to provide operating-system capabilities for 8K and larger System/360 configurations that include one or more IBM 2311 Disk Storage Drives. Similarly, the 16K tape-resident or disk-resident versions provide operating-system capabilities for 16K or larger configurations. In all three of these packages, on-line residence storage (magnetic tape for 16K tape, 2311 disk for 8K disk and 16K disk) is provided for all programs. Depending on the requirements of the particular application, each system can be expanded to include all processing programs used in a particular installation, or it can be cut down to a minimum system to control a single program.

The three Basic Operating System/360 programs (8K Disk, 16K Disk, and 16K Tape) are similar. Major differences are pointed out in text. For example, the 8K disk system offers translators for the assembler and report program generator languages. The 16K tape or disk systems offer, in addition, the COBOL, FORTRAN, and Programming Language/I translators.

These Basic Operating System/360 packages are designed for small or medium systems that do not require the expanded facilities provided by Operating System/360.

The 8K and 16K disk-resident systems are provided specifically for configurations including one or more IBM 2311 Disk Storage Drives. In addition to stacked job processing, concurrent Tele-processing applications, using terminals linked to an IBM 2701 Data Adapter unit or IBM 2702 Transmission Control Unit, are supported in the 16K *disk*-resident system. At least 32K bytes of main storage are required for concurrent Tele-processing. This Tele-processing facility is not available with the 8K *disk*-resident or 16K *tape*-resident packages.)

Also, limited concurrent peripheral operations (SPOOL) are included in the 16K disk-resident or 16K tape-resident systems for configurations with at least 32K bytes of main storage.

System Generation

The Basic Operating System/360 is delivered from IBM in either magnetic-tape or disk-pack form.

In some cases the system may be used as received. Ordinarily, the user will build a system tailored to the requirements of his particular installation.

Control Program

The control program constitutes the framework of each Basic Operating System/360. It prepares and controls the execution of *all* other programs. The components of each control program are:

1. *Supervisor*. This program handles all input/output operations, interruption conditions, and other control functions for all problem programs. Part of the supervisor resides in main storage at all times. Processing time is divided between the supervisor and the program being executed. This is true of user programs as well as the other IBM-supplied components of the system.

The physical IOCS routines of the supervisor handle the scheduling and supervise the execution of channel programs. The problem program (or logical IOCS within the problem program) supplies the channel programs and issues physical I/O macro instructions to request their execution.

The supervisor starts the I/O operation and returns control to the problem program. When the operation is completed, the supervisor checks for and handles any device error conditions. Thus, the user's program need not contain any I/O device error routines.

The checkpoint/restart routines of the supervisor provide a means of recording program status at desired points so that the program can be restarted at that point at a later time. The problem program resumes processing after each checkpoint. In response to CHKPT (checkpoint) macro instruction, the checkpoint routine writes the problem program, along with other information needed to restart the program, onto magnetic tape. The restart routine can reload the program later and resume processing

at the point of interruption. The restart program can reposition magnetic tape files prior to resuming program execution.

The supervisor transfers control of the system to job control at the end of each job step, providing transition between job steps and between jobs.

The storage print routine of the supervisor can provide a printout of core storage, and all registers, in the event of an abnormal end-of-job condition.

Some of the supervisor routines are loaded into main storage during system initialization (see *IPL Loader*, item 3 following). These routines are never overlaid and remain in main storage throughout execution of a stream of jobs. Other routines of the supervisor are called into main storage from external storage (tape or disk) only when their particular functions are needed. These are called *transient routines*. They are loaded into an area of main storage called the *transient area* and overlay any previous routine in the area. This allows numerous supervisor functions to be provided while using a minimum amount of storage.

2. *Job Control*. This program runs between jobs and prepares the system for execution of all other programs. Job control is loaded by the supervisor whenever it is needed.
3. *IPL Loader*. This program loads the supervisor into main storage when system operation is initiated. The IPL loader is loaded from the system residence unit by dialing its address on the load-unit switches on the system console and pressing the load key.

Data Management

Basic Operating System/360 provides a number of routines to facilitate handling input and output data files. These facilities are collectively referred to as *Data Management*. The system features that provide the data management functions are described in the sections *Logical IOCS* and *Label Processing*.

Logical IOCS -

IBM furnishes a comprehensive set of macro definitions to create, access, and maintain data files. Descriptive macro instructions in the user's program generate the data and program logic for these files. (In 8K Disk BOS, the macro instructions must be assembled immediately preceding the rest of the problem program. Thus, they occupy an area of core storage between the user's program and the supervisor.)

Each imperative macro instruction issued by the programmer causes a branch to the proper instruction (in the generated IOCS logic) for the requested service.

The services provided by logical IOCS include:

- Request physical I/O operations as necessary by issuing the required physical IOCS macro instructions. The necessary channel programs are generated from the descriptive IOCS macro instructions.
- Supply logical input records to, or accept logical output records from, the problem program. This includes blocking and deblocking logical data records (fixed length or variable length) from larger physical blocks. (*Logical record* refers to the individual unit of a data file; *physical record* refers to the block of logical records read or written as a single string of information.)
- Switch between two I/O areas to provide time for processing while records are being read or written.
- Handle end-of-file and end-of-volume conditions.
- Construct and maintain file organization structures. This includes additions and deletions to files and the construction and use of index tables for processing files.

Label Processing

Disk and tape label processing capabilities are included in the Basic Operating System/360 to provide:

1. Assurance that the correct editions of disk and tape data files are provided for input and (in the case of multipack or multireel files) that this input is provided in the correct sequence.
2. Assurance that areas of disk storage or tape reels designated for output contain no current information. If usable, new labels are written for the output areas or reels.

The actual label processing is performed by transient routines assembled as part of the supervisor during initial system generation. These routines are loaded into the transient area of the supervisor and executed in response to macro instructions in the problem program.

System Service Programs

The system service programs:

1. Generate the system.
2. Create and maintain the libraries.
3. Edit programs onto a residence device before execution.

Once a system has been built with complete system-generation capability, other minimum systems can be built that do not include the system service programs. Such minimum systems still require residence in external storage (tape or disk, depending upon the system used).

The system service programs are:

1. *Linkage Editor*. The linkage editor:
 - a. Links separately compiled decks.
 - b. Relocates these decks as required.
 - c. Resolves external references.
 - d. Includes modules from the relocatable library as necessary.

All programs are edited onto the resident disk (disk system) or a utility tape (tape system) by the linkage editor. These programs can then be permanently placed in the core image library of the system, requiring only control statements to call them for execution. Alternately, they can be executed at once, then overlaid by new programs.

2. *Librarian*. This group of programs maintains the libraries and provides printed and punched output from them. Three libraries, all residing on tape or disk, are available.

- a. *Core Image Library*. All programs cataloged in the system (IBM-supplied and user programs) are loaded from this library by the system-loader routine of the supervisor.

- b. *Source Statement Library (16K Disk or 16K Tape)*. This library contains IBM-supplied and user-defined source statement books such as macro definitions. A *book* is an arbitrary collection of 80-byte records that is cataloged in the source-statement library under a single name.

Books are maintained in compressed format on the resident volume to conserve space and improve their speed of retrieval. Complete books may be added or deleted from the library (but not individual records). These books can be copied, for example, into assembly source programs or COBOL source programs.

The Basic Operating System/360 (8K disk) does not have the full source-statement library facilities. Rather, its *Macro Library* stores IBM-supplied and user-defined macro definitions in resident packs built to provide program-assembly capability.

- c. *Relocatable Library*. This library stores object modules for subsequent linkage into complete programs.

3. *Load System Program (8K Disk only)*. This is an independent program that is loaded from cards. It has its own IPL, supervisor, and job-control programs. The load-system program builds a resident system from *cards*. This program can be used to build minimum systems for specialized applications. If two disk drives are available, the librarian can be used instead of the load-system program to build specialized systems.

Processing Programs

All user programs are run within the BOS environment under control of the supervisor. A minimum resident tape or disk pack may consist of only the control program and one or more user programs. For other applications, the resident tape or disk pack can contain the control program and the linkage editor. This system can edit user programs from cards or tape onto a system tape or disk pack, and then into main storage for execution. A full system may include all user programs and the following IBM-supplied programs:

1. *Language Translators: Assembler, COBOL, FORTRAN, Report Program Generator (RPG), and Programming Language/I (PL/I)*. Basic Operating System/360 (8K disk) offers only the assembler and report program generator languages.
2. *Tape Sort/Merge or Disk Sort/Merge*.
3. *Utilities*.
4. *Autotest*.

Sort/Merge Programs

The IBM Basic Operating System/360 sort/merge programs provide the user with the ability to sort files of random records, or merge multiple files of sequenced records, into one sequential file. The control-data information can be contained in as many as twelve fields in each record. The records can be sorted or merged into ascending or descending sequence. An individual sequence can be specified for each control-data field. The output sequence for a merge-only operation must be the same as the input sequence.

The sort/merge program is a set of generalized modules (in the relocatable library) which must be tailored at execution time to each application. The user furnishes appropriate parameters on control cards; the tailored sort/merge program is built in the core image library; and it is then automatically executed as a sequence of overlays from this library.

The 8K disk system and 16K disk system furnish disk sort/merge programs; both 16K systems (tape and disk) furnish tape sort/merge programs.

Utility Programs (8K Disk)

Eleven of these utility programs comprise modules in the relocatable library. The other two utility programs (*Clear Disk* and *Tape Compare*) are disk-resident programs located in the core image library.

Each utility program handles a particular type of job. (That is, the tape-to-printer program will print any tape file on any single printer.) The appropriate mod-

ules are selected for each job, and the generalized program is then modified by control cards. The tailored utility program is then ready for use.

The eleven file-to-file utility programs are:

- Tape to Tape.
- Tape to Disk.
- Tape to Card.
- Tape to Printer.
- Disk to Tape.
- Disk to Disk.
- Disk to Card.
- Disk to Printer.
- Card to Tape.
- Card to Disk.
- Card to Printer and/or Punch.

The initializing utilities are:

- *Clear Disk*. The area to be cleared can be as small as one track or up to a maximum of a complete disk pack.
- *Tape Compare*. The tape-compare program compares two files from two or more tapes to ensure that the files are identical.

Utility Programs (16K Disk or 16K Tape)

Thirteen utility functions provided in Basic Operating System/360 (8K disk) are also available for Basic Operating System/360 (16K disk). Seven of these utilities are also available in the 16K tape system (i. e., those not using disk or data cell storage devices). Though the 8K and 16K utilities have the same functions, the utility programs for the 16K packages are designed specifically for the 16K Basic Operating System/360 environment.

The eleven file-to-file programs are:

- Tape to Tape.
- Tape to Disk (16K disk only).
- Tape to Card.
- Tape to Printer.
- Disk to Tape (16K disk only).
- Disk to Disk (16K disk only).
- Disk to Card (16K disk only).

- Disk to Printer (16K disk only).
- Card to Tape.
- Card to Disk (16K disk only).
- Card to Printer and/or Punch.

The initializing programs are:

- Clear Disk.
- Tape Compare.

In addition, the following utilities are provided for the IBM 2321 Data Cell Drive in the 16K Disk Basic Operating System/360:

- Tape to Data Cell.
- Disk to Data Cell.
- Data Cell to Tape.
- Data Cell to Disk.
- Data Cell to Data Cell.
- Data Cell to Printer.

Autotest (8K Disk)

This is the principal debugging aid of the 8K Disk Basic Operating System/360. It resides in the core-image library on the system disk. Its testing services are available to any program:

1. Assembled by the IBM Basic Operating System/360 Assembler (8K Disk).
2. Run under control of the IBM Basic Operating System/360 Supervisor (8K Disk).

Autotest provides full testing services to programs that are either punched in cards or written in the relocatable library.

Autotest jobs can be intermixed with other non-Autotest jobs. Certain job-control cards request and further specify each Autotest job (e. g., request dynamic monitoring of the execution of a user program). If one of the tested programs is trapped in an unending loop or destroys some vital portion of the control program, Autotest provides special operating instructions and control cards to continue the run.

The Autotest concept permits remote testing of programs. A programmer can preplan his test to run with a minimum of operator intervention. Furthermore, the user can convert his program to its non-Autotest state easily, without reassembly, by removing the control cards.

Autotest provides the following optional services for any test-job:

Autopatch. Instructions can be exchanged, added, or deleted without reassembling or computing linkage addresses.

Display. Any portion, or all of main storage can be dumped in a variety of formats whenever a designated address in the user's program (the *test-point address*) is reached during the execution.

Panel. The programmer can have the contents of his registers, etc, printed at a designated address (the test-point address) during execution of his program. He can choose one of the following four printouts:

- General registers (0-15).
- Floating point (0-6) and general registers.
- Permanent storage assignments (bytes 0-127), and general registers.
- Permanent storage assignments, general, and floating point registers.

Occurrence of Panel and Display. The programmer can control the occurrence of a panel or display in a programmed loop.

Methods of Specifying Addresses. Disk Autotest provides three methods of specifying main storage addresses for the autopatch, display, and panel control cards. Any name-field symbol in a source program can be referenced as an address if an assembler symbol table is provided to Autotest. Hexadecimal or decimal addresses can also be used.

Utilities. Any Basic Operating System/360 Utility or special routine written for the user's installation that is in either the core image library or the relocatable library can be executed within Autotest. A JOB card must be used to call these programs.

Card-to-Tape (Variable) Utility. Autotest provides a special utility for building variable-or undefined-length records on tape.

Phase Fetch Order Listing. The user can print a listing of all the phases fetched by his program in the order of usage.

Symbolic Dump. This type of dump relates main storage data (at the time of the dump) to source program symbols. To obtain a dump in this format, an assembler symbol table must be provided to Autotest.

Normal End-of-Job Dump. A programmer can optionally dump the contents of main storage at normal end-of-job. If he requests a dump, the contents are always printed in hexadecimal representation. In addition, the contents can be printed in one or more of the following formats:

- Alphameric.
- Symbolic.
- Mnemonic representation of operation codes.

Abnormal End-of-Job Dump. An abnormal end-of-job dump is always printed; the programmer can select the format for it. These formats are the same as for the normal end-of-job dump.

Autotest (16K Disk/Tape)

The IBM Basic Operating System/360 16K Autotest program is available in two versions—tape and disk. By system generation, the user builds the 16K Basic Operating System/360 Autotest into the core image library.

The 16K Basic Operating System/360 Autotest can be used to test any program processed by the linkage editor of the 16K Basic Operating System/360. Consecutive tests can be run on the same program with different sets of data. Assemble and test is possible also (i. e., a program can be "Autotested" immediately following an assembly).

Autotest jobs can be intermixed with non-Autotest jobs. All output from an Autotest job will be on the auxiliary printed-output file of the 16K Basic Operating System/360.

Communication between the problem program and Autotest is through supervisor calls. The type of Autotest action to be performed, and associated data, is stored in a patch table built by the linkage editor in the lowest part of user core. This table is variable and a function of the type and number of Autotest requests. Functions performed by this Autotest are similar to those obtained through the Basic Operating System/360 (8K disk) Autotest program.

IBM 1070 Process Communication Supervisor

The 1070 process communication supervisor has the ability to monitor many remote processing facilities on a real-time basis. This includes:

1. Constant scanning of process terminals.
2. Control of all transmission to and from the remote stations.
3. Evaluation of data.
4. Execution of analytical and corrective subroutines in the event of error conditions.

Recognizing the formidable programming task for a system of this scope, IBM has developed the IBM System/360-1070 Process Communication Supervisor. This supervisor relieves the user of much programming effort by providing efficient pretested routines to gather and evaluate data, and to effect proper control action at remote stations.

The process communication (PC) supervisor operates in a Basic Operating System/360 (8K disk) environment. Relying on the BOS supervisor to handle ordinary physical and logical I/O operations (i. e., for cards, disk, etc.), the PC supervisor is specialized to the process control aspects of the user's program. Together with the BOS supervisor, the PC supervisor provides the process control user with complete control of communication lines and access to all I/O capabilities of the system.

To the BOS supervisor, the PC supervisor is simply a problem program to be loaded by means of a JOB card. Once gaining control, the PC supervisor becomes the primary system supervisor, intercepting all interrupts and passing control to the appropriate routines. (In many cases, these routines are the normal ones of the 8K disk BOS supervisor.)

The PC supervisor is designed for dedicated process control environment; no provisions are made for time-sharing with other, independent programs.

The significant features of the System/360-1070 process communication supervisor are:

- Initiation of all communication line operations.
- Handling of all interrupts from the communication lines and arranging the execution of the various routines on a priority basis.
- Comprehensive error-checking capabilities with automatic diagnostic printouts.
- Subroutines to handle standard data from the process, performing code conversion when necessary.
- Simulation of a real-time clock and a series of interval timers.
- Complete access to all of the facilities offered by the Basic Operating System/360 (8K disk) that are not offered by the PC Supervisor.
- Capability of being tailored to a particular installation during assembly, thereby saving time and core storage.

Appendix A. Basic Programming Support Minimum Machine Requirements

Card Assembler

- System/360 with 8,192 bytes of main storage.
- Card Read Punch (IBM 2540 or IBM 1442).

Card Report Program Generator

To generate an RPG object program:

- System/360 with 8,192 bytes of main storage.
- One card reader (2540, 1442, 2520-B1, or 2501)
- One card punch (if object program card deck is desired) (2540, 1442, 2520-B1, 2520-B2, or 2520-B3).
- Decimal arithmetic feature.

To execute an RPG object program:

- System/360 with 8,192 bytes of main storage.
- I/O units required by the object program.
- Decimal arithmetic feature.

Basic Tape System

To generate a system tape:

- System/360 with 8,192 bytes of main storage.
- Two magnetic tape units (2400 series); one must be 9-track. (If a 7-track tape unit is used, it must have the data conversion feature.)
- One card reader (2540, 1442, 2501, or 2520).
- One multiplexor or one selector channel.

For assembly of a problem program or a supervisor:

- System/360 with 8,192 bytes of main storage.
- Three magnetic tape units (one must be 9-track).
- One card reader (2540, 1442, 2501, or 2520).
- One printer (1403, 1404, 1443, or 1445).
- One multiplexor or one selector channel.

To execute an object program:

- System/360 with 8,192 bytes of main storage.
- One multiplexor or one selector channel.
- One card reader (2540, 1442, 2501, or 2520).
- I/O devices required by the problem program.

Sort/Merge (8K Tape)

- System/360 with 8,192 bytes of main storage.
- Three 2400 series magnetic tape units (7 and/or 9 track).
- One multiplexor or one selector channel.
- One 1403, 1404, or 1443 Printer or one 1052 Printer Keyboard.
- One card reader (2540, 1442, 2501, or 2520).

Utilities

- System/360 with 8,192 bytes of main storage (16,384 bytes required for multiple disk-to-printer program).

For program loading:

- One 2540 Card Read Punch or one 2520 Card Read Punch or one 1442 Card Read Punch or one 2501 Card Reader.

For program operation:

- I/O devices required by the specific utility program. 1052 Printer Keyboard (for multiple utility programs).

Universal Character Set Utility Program

- System/360 with 8,192 bytes of main storage.
- One card reader (1442, 2540, 2501, or 2520).
- Universal Character Set Feature on appropriate printers and 2821 Control Units.
- Control units (2821) and printers (1403) required by the applications.

Autotest

- System/360 with 16,384 positions of main storage.
- One card reader (2540, 1442, 2501).
- One 2400 series magnetic tape unit (9-track), used for the Autotest system tape.
- One 1443 or 1403 Printer.

FORTRAN IV (Card) or FORTRAN IV (Tape)

- System/360 with 16,384 bytes of main storage.
- One 1442 or 2540 Card Read-Punch.
- One 1443 or 1403 Printer.
- From three to ten 2400 series magnetic tape units, depending on the type of program processed. (This item for tape systems only; not for Card FORTRAN IV.)
- Scientific Instruction Set.

Input/Output 1418/1428

- A System/360 with 8,192 bytes of main storage.
- A card reader (2540, 1442-N1, 2501-B2, or 2501-B3).
- Either a 1418 or 1428.
- Direct-control feature or external-interrupt feature to utilize the external interrupt lines.
- Appropriate control units.

Input/Output 1412/1419

- A System/360 with 8,192 bytes of main storage.
- A card reader (2540, 1442-N1, 2501-B2, or 2501-B3).
- Either a 1412 or 1419.

- Direct-control feature or external-interrupt feature to utilize the external interrupt lines.
- Appropriate control units.

Dual 1419 Input/Output

- A System/360 Model 30 or 40 with at least 16,384 positions of main storage.
- Multiple 1442 or 2540's.
- Two 1419's (*on the multiplexor channel*).
- Two 1403 Printers (with selective tape listing feature and the universal character set feature). The printer may be operated with or without the selective tape listing feature. Each printer is attached to a *separate* selector channel.
- One multiplexor and two selector channels.
- Two 2400 series magnetic tape units (one read-while-write control unit: 2404 or 2804).

Input/Output 1231 N1

- System/360 with 8,192 bytes of main storage.
- One 1231 N1.
- One Card Reader (1442-N1, 2540, 2501-B1, or 2501-B-2).

Appendix B. Basic Operating System/360 Minimum Machine Requirements

8K Disk Resident System

For control programs:

- System/360 with 8,192 bytes of main storage.
- One multiplexor or one selector channel.
- One card reader (1442, 2501, 2520 or 2540).
- One 2311 Disk Storage Drive.

For system service programs:

1. Librarian

Additional features required:

- Load System—none.
- Linkage Editor—none.
- Library Service—Punched output (1442, 2520, or 2540 card punch)
—Printed output (1403, 1404, or 1443 printer).
- Library Organization—none, except COPYS function (one additional 2311 Disk Storage Drive).

2. Assembler and RPG

Additional features required:

- Assemble-and-execute—none
- Punch card output—(1442, 2520, or 2540 Card Punch).
- Printed listing—(1403, 1404, or 1443 Printer).
- Decimal Arithmetic Feature—(for RPG only).

3. Disk Sort/Merge

Additional features required:

- One 1403, 1404, 1443 Printer or 1052 Printer Keyboard.

4. Utilities

Additional features required:

- One 1403, 1404, 1443 Printer or 1052 Printer Keyboard.
- The I/O units used by the particular utility program.

5. Autotest

Additional features required:

- System/360 with 16,384 bytes of main storage.
- One Printer (1403, 1404, or 1443).

16K Disk Resident System

For control programs:

- System/360 with 16,384 bytes of main storage.
- One multiplexor or one selector channel.
- One card reader (1442-N1, 2501, 2520-B1, or 2540). See Note 1.
- One card punch (1442-N1, 1442-N2, 2520, or 2540). See Note 1.
- One printer (1403, 1404, or 1443). See Note 1.
- One 1052 Printer Keyboard. Required for satisfactory system performance. The system can operate without this device.
- One 2311 Disk Storage Drive. See Note 2.

Notes: 1. One 2400 Series Magnetic Tape Unit may be substituted for this device (7-track or 9-track; if 7-track is used, the data-convert feature is required for card reader and card punch functions.)

2. System generation requires an additional disk drive or a 2400-Series Magnetic Tape Unit (7-track or 9-track; if 7-track is used, the data-convert feature is required).

For assembler:

Additional features required:

- None.

For RPG:

Additional features required:

- Decimal Arithmetic.

For COBOL:

Additional features required:

- Decimal Arithmetic.
- Floating Point (if required by the COBOL source program).

For FORTRAN:

Additional features required:

- Floating Point.

For Programming Language/I:

Additional features required:

- Floating Point.
- Decimal Arithmetic.

For Autotest:

Additional features required:

- None.

For Disk Sort/Merge:

Additional features required:

- None.

For utility programs:

Additional features required:

- I/O units required for the particular utility function.

16K Tape Resident System

For control programs:

- System/360 with 16,384 bytes of main storage.
- One multiplexor or one selector channel.
- One card reader (1442-N1, 2501, 2520-B1, or 2540). See Note 1.
- One card punch (1442-N1, 1442-N2, 2520, or 2540). See Note 1.
- One Printer (1403, 1404, or 1443). See Note 1.
- One 1052 Printer Keyboard. Required for satisfactory system performance. The system can operate without this device.
- One 2400 Series Magnetic Tape Unit (9-track) for system residence and three 2400— Series Magnetic Tape Units (see Note 2) for language translator functions.

An additional tape unit (see Note 2) is required for compile and execute functions.

Notes: 1. One 2400 Series Magnetic Tape Unit (7-track or 9-track) may be substituted for this device. Also see Note 2.

2. If a 7-track tape unit is used, the data-convert feature is required for card reader and punch functions.

For assembler:

Additional features required:

- None.

For RPG:

Additional features required:

- Decimal Arithmetic.

For COBOL:

Additional features required:

- Decimal Arithmetic.
- Floating Point (if required by the COBOL source program).

For FORTRAN:

Additional features required:

- Floating Point.

For Programming Language/I:

Additional features required:

- Floating Point.
- Decimal Arithmetic.

For Autotest:

Additional features required:

- None.

For Tape Sort/Merge:

Additional features required:

- None.

For utility programs:

Additional features required:

- I/O units required for the particular utility functions.

IBM 1070 Process Communication Supervisor

To assemble 1070 Process Communication Supervisor and user's routines:

- The machine configuration for assembly is the same as that needed for assembling with the Basic Operating System/360 (8K Disk).

To execute an object program using the 1070 Process Communication Supervisor:

- System/360 with 16,384 bytes of main storage.
- Standard Instruction Set.
- IBM 2701 Data Adapter Unit (handles up to four lines) and/or IBM 2702 Transmission Control (handles up to 31 lines).
- One selector channel.
- One multiplexor channel (no I/O device can operate in burst mode on the multiplexor channel).
- IBM 1052 Printer Keyboard.
- Interval Timer Feature.
- Minimum requirements of the Basic Operating System/360 (8K Disk).

Appendix A. Basic Programming Support Minimum		
Machine Requirements	27	
Appendix B. Basic Operating System/360		
Minimum Machine Requirements	29	
Autotest (BPS)	18	
Autotest (8K Disk-BOS)	24	
Autotest (16K Disk/Tape-BOS)	25	
Basic Operating System/360	21	
Basic Operating System/360 (Introduction)	5	
Basic Programming Support Basic Tape System	13	
Basic Programming Support (Introduction)	5	
Basic Programming Support (8K Card)	11	
Basic Tape System (BPS)	13	
Card Assembler	11	
Card Option (Card Assembler)	11	
Card Report Program Generator	11	
Checkpoint (BOS)	21	
Checkpoint (BPS)	14	
Component Programs of the Basic Tape System	13	
Control Program (BOS)	21	
Control Programs (Basic Tape System-BPS)	14	
Core Image Library (BOS)	23	
Data Management (BOS)	22	
Dual 1419 Input/Output Program	20	
Dump Program (Card Assembler)	11	
FORTRAN IV Card	12	
FORTRAN IV Tape (BPS)	19	
Input/Output Control System (IOCS)	13	
Input/Output Support Package (Card Assembler)	11	
Input/Output 1231 N1 Program	20	
Input/Output 1418/1428 Program and Input/Output 1412/1419 Program	20	
Introduction	5	
IPL (Basic Tape System-BPS)	14	
IPL (BOS)	22	
Job Control (Basic Tape System-BPS)	14	
Job Control (BOS)	22	
Label Processing (BOS)	22	
Language Translators (BOS)	23	
Librarian (BOS)	23	
Libraries (Basic Tape System-BPS)	15	
Library Maintenance Programs (Basic Tape System-BPS)	15	
Linkage Editor (Basic Tape System-BPS)	15	
Linkage Editor (BOS)	23	
Loaders (Card Assembler)	11	
Load System Program (BOS)	23	
Logical IOCS (BOS)	22	
Logical Record	13	
Minimum Machine Requirements (BPS)	27, 28	
Minimum Machine Requirements (BOS)	29, 30	
Paper Document Programs (BPS)	20	
Physical Record	13	
Preface	2	
Process Communication Supervisor (8K Disk-BOS)	25	
Processing Programs (BOS)	23	
Program Loader (Basic Tape System-BPS)	15	
Relocatable Library (BOS)	23	
Sort/Merge (BOS)	23	
Sort/Merge (BPS)	16	
Source Statement Library (BOS)	23	
Supervisor (Basic Tape System-BPS)	14	
Supervisor (BOS)	21	
System Generation (BOS)	21	
System Service Programs (Basic Tape System-BPS)	15	
System Service Programs (BOS)	22	
System Tape	13	
Tape Assembler (BPS)	13	
Tape Option (Card Assembler)	11	
Tape RPG (BPS)	13	
Utility Programs (BPS)	17	
Utility Programs (8K Disk-BOS)	23	
Utility Programs (16K Disk and 16K Tape-BOS)	24	
Utility Programs (Card Assembler-BPS)	11	



International Business Machines Corporation

Data Processing Division

112 East Post Road, White Plains, N. Y. 10601

READER'S COMMENT FORM

S/360 BPS and BOS/360
Programming Systems Summary C24-3420-0

- Your comments, accompanied by answers to the following questions, help us produce better publications for your use. If your answer to a question is "No" or requires qualification, please explain in the space provided below. All comments will be handled on a non-confidential basis. Copies of this and other IBM publications can be obtained through IBM Branch Offices.

- | | Yes | No |
|--|--------------------------|--------------------------|
| ● Does this publication meet your needs? | <input type="checkbox"/> | <input type="checkbox"/> |
| ● Did you find the material: | | |
| Easy to read and understand? | <input type="checkbox"/> | <input type="checkbox"/> |
| Organized for convenient use? | <input type="checkbox"/> | <input type="checkbox"/> |
| Complete? | <input type="checkbox"/> | <input type="checkbox"/> |
| Well illustrated? | <input type="checkbox"/> | <input type="checkbox"/> |
| Written for your technical level? | <input type="checkbox"/> | <input type="checkbox"/> |
- What is your occupation? _____
 - How do you use this publication?

As an introduction to the subject?	<input type="checkbox"/>	As an instructor in a class?	<input type="checkbox"/>
For advanced knowledge of the subject?	<input type="checkbox"/>	As a student in a class?	<input type="checkbox"/>
For information about operating procedures?	<input type="checkbox"/>	As a reference manual?	<input type="checkbox"/>
 - Other _____
 - Please give specific page and line references with your comments when appropriate. If you wish a reply, be sure to include your name and address.

COMMENTS:

- Thank you for your cooperation. No postage necessary if mailed in the U. S. A.

Fold

Fold

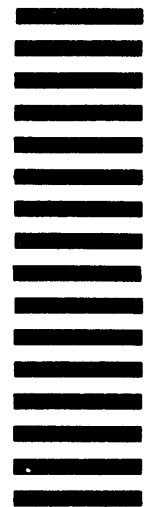
FIRST CLASS
PERMIT NO. 170
ENDICOTT, N. Y.

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY

IBM Corporation
P. O. Box 6
Endicott, N. Y. 13764

Attention: Product Publications, Dept. 171



Fold

Fold

Cut Along Line

IBM S/360

Printed in U. S. A.

C24-3420-0



International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N. Y. 10601

Additional Comments: