GC33-5008-1

Systems

DOS Version 4
System Generation

DOS Release 27.1

IBM

Second Edition (September, 1973)

This is a reprint of GC33-5008-0 incorporating changes released in Technical Newsletter GN33-8723 (dated November 1, 1972).

This edition relates to Version 4, Release 27.1, of DOS and to subsequent Version 4 releases until otherwise indicated in new editions or technical newsletters. Changes are continually made to the information herein; before using this publication, consult the IBM System/360 and System/370 Bibliography, GA22-6822, for the editions that are applicable and current.

Major changes and additions are:

- Support for the System/370 Model 135
- RMS is replaced by RMSR (Recovery Management Support Recorder)
- Support for the Time-Of-Day (TOD) clock
- EREP enhancements
- Support of new devices (2596, 3330, 3410, 3420, 3505, 3525)
- BTAM and QTAM support for the 3330 Disk Storage
- BTAM support for the 3270 Information Display System.

Comments concerning the contents of this publication may be addressed to IBM Laboratory, Publications Dept., P.O. Box 24, Uithoorn, The Netherlands. Comments become the property of IBM.

Preface

The information in this publication is of particular interest to anyone responsible for the planning and/or implementation of system generation and maintenance for DOS. This includes installation managers, system analysts, programmers, and machine operators.

Familiarity with the following system programs and facilities is invaluable when using this publication:

- the control programs (supervisor, job control, and IPL),
- the system service programs (linkage editor and librarian) and
- the Input/Output Control System (IOCS) logic modules.

The control program and the system service programs are described in the System Control and Service publication. The IOCS logic modules are described in the Supervisor and I/O Macros publication. In addition, the DOS Version 4 manual should be consulted for those features of DOS which are new with Version 4.

The storage estimates given in this publication are within 15% of actual requirements.

MODULAR ORGANIZATION OF THIS MANUAL

This publication is in a modular format that enables you to discard unwanted modules and to insert system generation modules from program product documents. Thus, you can tailor your system generation document for your installation.

Although the pages are numbered consecutively throughout the book, each module is a self-contained document. A numbered tab on the right-hand edge of the page indicates the beginning of each new module. Each module has a Module Outline and, where length warrants, a Module Index.

The total publication has a General Contents and a General Index. These refer you to the individual module numbers. Figure numbers throughout the publication are in the form: Figure 1-3, where 1 is the module number and 3 is the figure number within the module.

Each module in this publication is a collection of information about a particular component or related topics. The modules, numbered 1-35, are grouped as follows:

- Module 1: Planning and procedures information
- Modules 2, 3: Emulator programs
- Modules 4-11: I/O components
- Modules 12-23: Languages
- Modules 24-26: Sort/merge programs
- Module 27: System aids (CLTEP)
- Modules 28, 29: Teleprocessing
- Modules 30-35: Utilities

Information contained in the appendixes of previous editions is now located as follows. Three are located in Module 1:

1. IPL Control and ASSGN Statements
2. DOS System/370 Distribution Program and BPS Messages
3. Optimum Assignments of workfiles

The other two appendixes: Master Macro List and Master Transient List have been deleted, but the components still contain this information. The macros and transients are listed in the General Index.

Individual library block counts for preassembled IOCS modules, subroutines, and macros have been deleted. To obtain this information, execute the DSERV program with the sort operation.

Closely related publications are:

DOS Version 4, GC33-5007

DOS System Programmer's Guide, GC24-5073

IBM System/360 Basic Programming Support, Distribution Program Specifications and Operating Guide, GC21-5001

IBM System/360 Basic Programming Support, DASD Utility Programs Operating Guide, GC24-3392

DOS and TOS Utility Programs, GC24-3465

IBM System/360 Basic Programming Support, DASD Utility Programs Specifications, GC24-3363

References are made in this publication to the following:

DOS BTAM, GC30-5001

QTAM Message Control Program, GC30-5004

IBM System/360 Disk and Tape Operating Systems, COBOL Programmer's Guide, GC24-5025

IBM System/360 Disk Operating System, FORTRAN IV Programmer's Guide, GC24-6397

1401/1440/1460 DOS Emulator on Models 135/145/155, GC33-2004

1410/7010 DOS Emulator on Models 145/155, GC33-2005

General Contents

Module 1: Planning and Procedures

Modular Outline

Modular Index

Introduction

The object of DOS system generation and maintenance is to tailor the IBM-supplied system to the requirements of your installation. This tailoring process is accomplished by:

1. Planning the supervisor functions and the contents of libraries necessary to support the applications desired by your installation.

2. Implementing your plan by performing a system generation procedure.

IBM supplies the DOS system on either magnetic tape(s) or disk pack(s). These distributions are further discussed in Disk Operating System Distributions.

The disk operating system is composed of three libraries:

The source statement library contains IBM-supplied macro definitions. When the desired parameters are chosen, the macros can be assembled. For convenience, the source statement library also contains sample problems and system generation job streams that can be retrieved as needed.

The relocatable library contains IBM programs that have not been assigned addresses for execution and assembled macros from the source statement library. These assembled macros perform input and output procedures for IBM-supplied programs. These assembled macros (Logical IOCS modules) can also be used by problem programs when applicable.

The core image library contains programs that are ready for execution. System control programs and system service programs are always shipped in the core image library. Where it is necessary for system generation purposes, an assembler program is provided. The system control programs must always be part of the system. The librarian programs are a key set to the system and should be carefully considered before ever removing them from the system.

During system generation, work with the IBM-supplied system to tailor it to your individual needs by adding to and deleting from the IBM-supplied and your own libraries (source statement, relocatable, and core image).

Planning System Generation

Proper and detailed planning saves on total system generation time. Thus, it is important to perform a very thorough job. Figure 1-1 illustrates an overview of the planning process.

Planning system generation consists of:

- Planning the options and estimating the size of a supervisor. This entails selecting from the programming services provided by IBM, those options to be included in the supervisor, and estimating the cost of these services in terms of bytes of storage.

- Planning the contents, organization, and ultimate size of the system and/or private libraries. This entails distributing the storage space available (on the disk packs) between the libraries ultimately desired for day-to-day use. Major points of consideration are:

a. the size of the system and, if desired, private core image, relocatable and source statement libraries

b. workfile space needed to assemble a supervisor, and to accommodate the linkage editor, which is needed to catalog the components selected to the system core image library

c. standard assignments (allocation of space) for workfiles (assemblies), and linkage editing needed for everyday operation.

CONSIDERATIONS AND DECISIONS

The following considerations and decisions should be made before system generation:

- Select supervisor options by coding a set of supervisor macro instructions (see Planning a Supervisor).
- Determine the programs to be in the core image library of each operational pack (e.g., COBOL, FORTRAN, etc).
- To utilize the problem determination procedures recommended in DOS Version 4 Messages, GC33-5009, during system generation, include applicable aids from the following list in your DOS system:

  Tape Error Statistics and EVA (Magnetic Tape Error Analysis)

  FETCH/LOAD, I/O, GSVC and QTAM Traces, and the Transient Dump (Problem Determination Serviceability Aids, PDAIDs)

  DUMPGEN (DOS Stand-Alone DUMP Generator, with formatting DUMP option)

  LVTOC and DKPR

  Online Terminal Test (Teleprocessing Aids)

  LSERV (Label Cylinder Display Program)

- Determine the assembler to generate a new supervisor. (See the sections on Assembler D and Assembler F. Volumes 2 and 3 of the 2311 system have the 14K assembler supplied in the core image library.
- If the system supports both the batched-job foreground and private core image library options, give consideration to which IBM-supplied programs are to be placed in a private core image library or libraries. Under such a system, the linkage editor executes in any partition. You can link-edit most IBM-supplied programs for execution in a batched-job foreground partition and place them in a private core image library assigned to that partition if enough core storage is available to execute the link-edited program. If desired, the version to execute in the background partition may be retained in the system core image library.

  You must place the supervisor, $$A$SUP1, in the system core image library. A supervisor cataloged to a private core image library cannot be used, because the IPL retrieval program searches the system core image directory only. If the requested phase is a $$R phase (RAS transient), the system only searches the core image directory. If the requested phase is a $$A or $$B phase (transient), the system first searches the system core image directory, then, if necessary, the private core image directory assigned to the partition; if the phase is not found, the system enters the wait state with an error message of 04W (X'04E6') in bytes 0 and 1 of low main storage.

When you request other phases, the system searches both the system and private directories, if necessary. If the phase starts with $, the system first searches the system core image directory and, if it does not find the phase, it then searches the directory of the private core image library, assigned to the partition. If the requested phase does not begin with a $, the order is private core image directory, then system core image directory.

The ERP transient $$ANERSE supports the error recovery for the 2245 Kanji Printer. If you do not have this printer you may delete $$ANERSE from the core image library of your operating system.

- Determine which modules are to be deleted from the relocatable library of each operational pack. Deleting from the relocatable library allows for expansion of the system core image library to hold a greater number of components. Refer to the module for each component for its sizes.

- Also determine if the macro definitions used to build the supervisor and IOCS modules are to be deleted from the source statement library. Retaining the macros in the source statement library facilitates building a new supervisor and new IOCS modules.

- With one disk drive you may prefer to maintain only enough room in the relocatable library of the operational pack to contain the modules for building the largest component in the system. This small relocatable library permits temporary insertion of any component in relocatable form. It can then be immediately linkage edited into the system core image library and then deleted from the relocatable library. When the relocatable library is subsequently condensed, only the updated core image form of the component remains, thus conserving disk-storage capacity. Reducing the size of the relocatable library allows expansion of the system core image library. The expanded system core image library allows a greater number of components to be contained in a single systems volume.

- Copy and restore programs are necessary to transfer the resident system from tape to disk, from disk to tape, from disk to cards, and from cards to disk for maintenance and backup purposes.

- The procedures for the configurations shown in this publication assume the system packs to be initialized with the VTOC on cylinder 199, when using either a 2311 or 2314/2319, or on cylinder 403 when using a 3330, and the work packs to be initialized with the VTOC on cylinder zero or 199 (for a 2311, 2314, or a 2319) or on cylinder 0 or 403 (for a 3330).

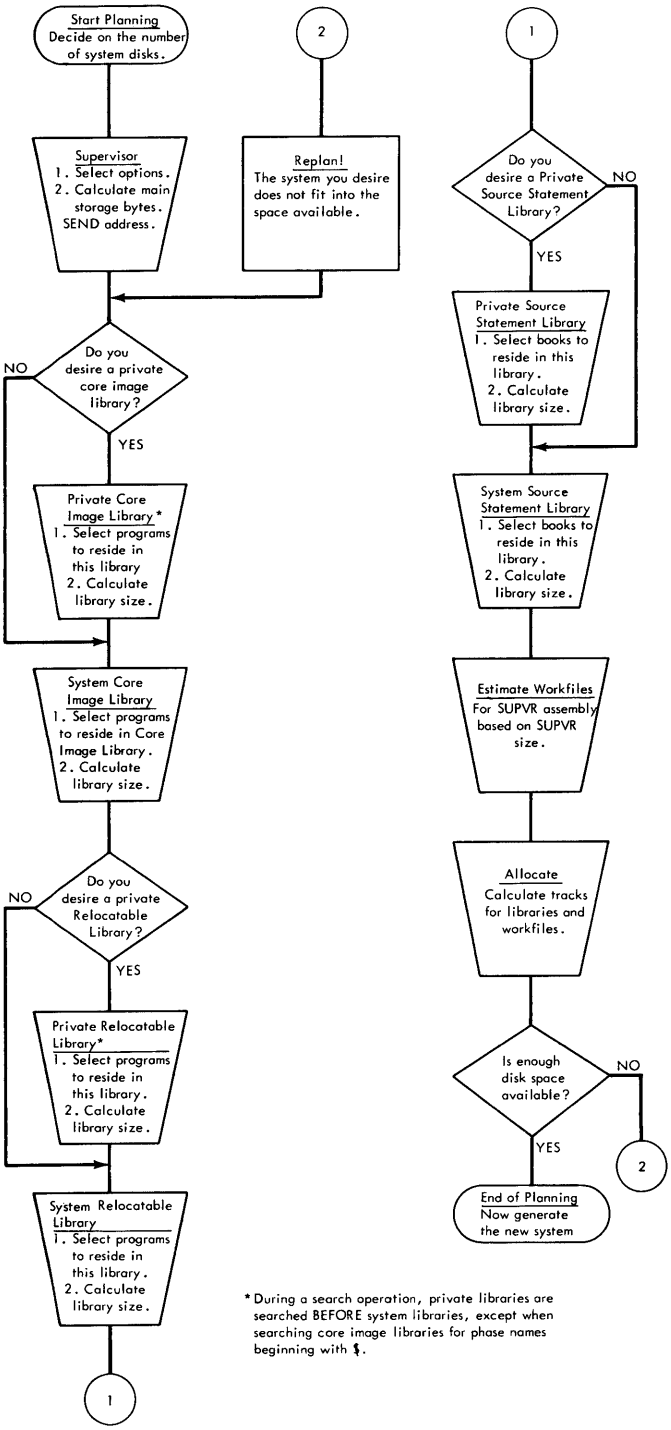


Figure 1-1. INTRODUCTION--Planning Summary

MAPS AND LISTINGS PRODUCED DURING SYSTEM GENERATION

All linkage editor output on SYSLST from the system generation procedure and any future updates (including maps produced by the linkage editor) must be retained. These maps provide necessary information on the level of the system and the load address (relocation) of each component. Similarly, supervisor assembly listings should be retained. These maps and listings are used by the systems programmer and the field engineer maintaining the system.

System Configuration

This section presents the minimum system configuration required to operate the disk operating system. The system control programs and basic IOCS must always be present to execute any other programs.

MINIMUM MACHINE REQUIREMENTS

A System/370 CPU Model 135, 145, or 155.

Standard instruction set. See Note 1.

Two I/O channels (one multiplexer, and one selector or the integrated file adapter). See Note 2.

One Card Reader (1442, 2501, 2520, 2540, or 3505). See Note 3.

One Card Punch (1442, 2520, 2540, or 3525). See Note 3.

One Printer (1403*, 1404**, 1443, 2245, or 3211). See Note3.

* 1403 Model STL if not attached to a 2821.
** if not attached to a 2821.

One 3210/3215 Console Printer Keyboard, attached to the multiplexer channel.

One 2311 Disk Storage Drive, or

One 2314 Direct Access Storage facility, or

One 2319 Direct Access Storage Facility, or

One 3330 Disk Storage (See Note 4).

Note 1: Language translators may require extended instruction sets.

Note 2: Telecommunication devices should not be on the same selector channel as SYSRES.

Note 3: One 7-track 2400 series or one 7-track 3420 magnetic tape unit, or one 9-track 2400 or 3400 series magnetic tape unit, or a disk extent may be substituted for this device. If 7-track tape units are used, the data-conversion feature is required, except when substituted for a printer. Neither the tape unit nor the disk extent can be substituted for the card reader as the communication device during system IPL time.

Note 4: The 3330 Disk Storage requires a high-speed selector channel.

Planning a Supervisor

The supervisor is a control program that provides specialized services to programs executed from the problem program area of main storage. This program is composed of a group of assembled macros. The options selected in each of the supervisor generation macros determine the size of the assembled supervisor. The size of each option is identified in Supervisor Main Storage Requirements.

The SEND macro determines the end of the supervisor. If a supervisor is assembled with a SEND address larger than the previous supervisor, some portion of the supervisor is overlaid by programs linkage edited to the previous SEND address. After the new supervisor is loaded, relinkage edit these programs so that the new load address follows the newly assembled supervisor.

Note that at each IPL, you must include ADD and ASSGN statements for each device until a system supervisor is built that describes your machine configuration and standard I/O assignments (see DVCGEN and ASSGN macros).

To calculate the size of the supervisor, list all of the options that are chosen and then determine the sum of the sizes obtained from the storage requirements (see Supervisor Main Storage Requirements).

When choosing supervisor parameters, check to see if the chosen options are compatible. Also, check to determine which parameters automatically provide the support for another.

With the IBM-supplied volume, each installation normally generates a more efficient supervisor using the supervisor generation macro instructions. The macro instructions describe the machine configuration, standard I/O assignments, and standard processing options.

MNOTES DURING SUPERVISOR ASSEMBLY

During supervisor assembly, the assembler may generate one or more MNOTEs. You may choose to ignore some by accepting the assumed values given in the MNOTEs. For others, you may have to interrupt procedures, modify one or more specifications in a macro or macros, and reassemble the supervisor.

Some general procedures for resolving MNOTEs are:

1. Go to the DIAGNOSTICS section at the end of the supervisor assembly listing. This section includes references to the MNOTEs generated during assembly. Each MNOTE reference is in the form

   ```
   statement-number  IJYQ37  MNOTE STATEMENT
   ```

2. Using the statement number, go back into the listing and examine each of the MNOTEs. A severity code precedes the message portion (the higher the code, the more severe the error). Each MNOTE is listed after the macro with which it is associated.

3. Determine the reason for the MNOTE. The MNOTE message indicates the parameter in question and usually provides a clue to the type of discrepancy. Some errors to look for are:

   - Misspelled items or misspunched numbers.

- Parameters that are incompatible with other parameters in this or another macro.

- Parameters that are outside the valid limits.

4. Make any necessary changes and reassemble. If you have no changes to make (in other words, the assumed values listed in the MNOTEs are acceptable), continue with the generation procedures. Some errors are so severe, however, that no object deck is generated, and you must correct one or more macros and reassemble before continuing.

Example of MNOTE Resolutions

The following example of a single supervisor assembly contains macros and the MNOTEs generated by the macros. Each set is followed by a discussion of how the MNOTE can be resolved. Because several macros need to be modified to achieve the desired results, it is necessary to reassemble.

```
1.  FOPT  PTO=YES,CCHAIN=NE,IDRA=YES,JA=(1000,270),                X
          DASDFP=(0,7,2321),CBF=YES
```

a. 3,CCHAIN SPECIFICATION INVALID-'NO' ASSUMED

b. 3,INCORRECT THIRD SUBOPERAND OF DASDFP, DASDFP=NO ASSUMED

c. 3,THE CHANNEL SPECIFICATION IS ABOVE MAXIMUM ALLOWABLE VALUE, DASDFP=NO ASSUMED

d. 3,JA SPECIFICATION INVALID 'NO' ASSUMED

e. 3,CBF SPECIFICATION INCORRECT - 1 BUFFER ASSUMED

f. 3,PTO=YES INVALID IN A BATCH SYSTEM - 'NO' ASSUMED

Discussion:

a. Specification for CCHAIN is misspelled. The default is the desired result, however. If this were the only error, there would be no need to reassemble.

b. This is another keypunch error (2331). Repunch the card with a corrected third operand.

c. The valid range of channels is 0-6. This parameter must be repunched for DASDFP.

d. There are two reasons why the JA parameter is invalid. First, the numbers themselves are too large - the maximum is 255 per partition. Second, MPS=BJF must be specified in the SUPVR macro if SIO (Start I/O) count is to be supported in the foreground. Repunch both items with the proper entries.

e. The CBF parameter must be specified as NO or as a decimal number in the range 1-50. If you are not satisfied with the assumed value, repunch and reassemble.

f. For PTO to be valid, you must specify MPS=YES or BJF in the SUPVR macro. Repunch the SUPVR macro specification and reassemble.

2. PIOCS BMPX=YES

3,BMPX SPECIFICATION INVALID WITH MICR SUPPORT - 'NO' ASSUMED

Discussion: MICR=1419 was specified in the SUPVR macro. Assuming that the MICR specification was intentional, you can merely accept the assumed value of NO for BMPX. If this were the only error, you would not need to reassemble.

3. ALLOC F1=14K,F2=14K

7,THIS MACRO ONLY VALID IN MPS SYSTEM-THIS MACRO IGNORED

Discussion: The SUPVR macro is the real cause of error (assuming you want an MPS system). Repunch the specification in the SUPVR macro for MPS=YES or BJF and reassemble.

4. IOTAB BGPGR=250,F1PGR=250,F2PGR=250,IODEV=256

a. 3,BGPGR SPECIFICATION ABOVE MAXIMUM-'10' ASSUMED

b. 3,IODEV SPECIFICATION ABOVE MAXIMUM-'10' ASSUMED

c. 3,F1PGR IS INVALID PARAMETER WHEN MPS=NO

d. 3,F2PGR IS INVALID PARAMETER WHEN MPS=NO

Discussion:

a. The number of programmer Logical Unit Blocks (LUBs) for all partitions cannot exceed 222 when MPS=BJF or 244 when MPS=YES or NO. Also see the discussion for c and d.

b. The maximum specification for I/O devices attached to the system is 255. Accept the default, or repunch the macro specification with an acceptable value and then reassemble.

c. and d. Repunch the SUPVR specification to MPS=YES or BJF. You must also repunch the programmer LUB specifications in the IOTAB macro, since the sum of the LUB specifications for the background and foreground partitions exceeds the maximum as described in a.

5. SEND

.

.

.

NUCEND EQU *

1, WARNING - IF ADDRESS OF NUCEND IS GREATER THAN ADDRESS OF SYSEND, SEND ADDRESS SPECIFICATION WAS TOO SMALL

Discussion: This message is always printed and should be checked after each supervisor assembly. With no address specified in the SEND macro, the addresses, NUCEND and SYSEND, will probably change when you reassemble with one or more macros changed from the previous run.

Be sure the NUCEND address is not greater than the SYSEND address (if the SEND address is not specified, the addresses will be equal). Compare the two hexadecimal addresses (at the left margin of the

listing) to ensure that the SYSEND address is at least as large as the NUCEND address. Read the discussion of the SEND macro and make any adjustments you feel are necessary or desirable.

This section defines the ten macro instructions and their parameters required to generate an installation-tailored supervisor for the disk operating system. Figure 1-4 is a consolidated list of the supervisor generation macros.

RULES FOR USING SUPERVISOR GENERATION MACROS

1. The assumed value default for an omitted parameter is underlined in the following discussion and in Figure 1-4. Figure 1-5 shows the device type codes of the DOS supported devices used for system generation.

2. Material enclosed in braces { } indicates a programmer option. One of the enclosed values must be selected by the programmer.

3. Bracketed operands are optional, e.g., [n].

4. Replace the letter n in a parameter with a decimal number.

5. The name field must be blank. The operation field always contains the mnemonic operation code. The operand field contains the parameters.

6. Several parameters may be included on one line for all macros. Separate each parameter with a comma. No embedded blanks are permitted. Continuation cards are permitted (nonblank character in column 72; the continue column is column 16).

7. In the expression X'cuu', replace cuu with the hexadecimal number for channel and unit.

8. The macros must be issued in the following sequence: SUPVR, CONFG, STDJC, FOPT, PIOCS, ALLOC, IOTAB, DVCGEN, ASSGN, SEND.

9. The DVCGEN, ASSGN, and ALLOC macros are not required. They are specified if input/output tables (DVCGEN) are being specified, if standard assignments (ASSGN) are being made or if storage is allocated (ALLOC) for MPS at system generation time.

10. An END card and a /* card must follow the SEND macro instruction.

SUPVR

The Supervisor (SUPVR) macro instruction and its parameters define the system as disk resident with the ability to support certain desired functions, such as multiprogramming, MICR, or teleprocessing.

PARAMETERS FOR SUPVR

MPS={NO / YES / BJF} Specify if there is to be multiprogramming support. When YES or BJF is specified, the system generated is capable of supporting two foreground programs. YES or BJF must be specified if TP=QTAM. When BJF is specified, batched job environment will be supported

for both foreground partitions. Multiple communication regions are generated only if BJF is specified. MPS=YES is implied if MPS=BJF is supplied. This parameter is discussed further in the DOS System Programmer's Guide, GC24-5073.

TP={NO / BTAM / QTAM / QTAMn}
Specify if teleprocessing support is desired, and, if so, whether Basic or Queued Access Method (BTAM or QTAM) is desired. When QTAM is specified, SVC support for BTAM is also included. For multitasking, n is the maximum number of problem programs in the system at one time. n may have any value from 2 to 12. (For multitasking, AP must equal YES.) If AP=NO and QTQMn are specified, n must be 2.

MICR={NO / 1419 / 1419D}
Specify if the supervisor is to support magnetic ink character readers or optical reader/sorters. If 1255/1259/1270/1275s are to be supported, indicate 1419. 1419D indicates Dual Address Adapter 1419/1275s only; 1419 support does not provide 1419D support. If 1419s or 1255/1270/1275s are attached to the multiplexer channel, the PICCS parameter BMPX=YES is not supported. This parameter is discussed further in the DOS System Programmer's Guide, GC24-5073.

ASCII={NO / YES}
Specify YES if supervisor support of ASCII-Code is desired. If ASCII is omitted, NO is assumed.

AP={NO / YES}
Specify if there is to be multitasking support. Multitasking allows the execution of more than one program within a partition. MPS=YES and WAITM=YES are assumed if AP=YES. If the operand of this parameter (NO or YES) is misspelled, AP=YES is assumed. This parameter is discussed further in the DOS System Programmer's Guide.

EU={NO / YES}
Specify EU=YES if you require mixed parity tape processing for the emulators and the tape preprocessor and tape postprocessor programs. You do not need to specify EU=YES to run the programs if mixed parity tape processing is not required.

Note: For reasons of compatibility with earlier DOS releases, specification of EU=RELOC is accepted by the assembler. The code generated is the same as if EU=YES had been specified.

ERRLOG={YES / RDE}
Specify to include Recovery Management Support Recording (RMSR) and the Reliability Data Extractor (RDE). RMSR records device errors, statistical data, and cumulative error status for I/O devices on the SYSREC file.

RDE causes RMSR to gather hardware reliability data used by IBM personnel to evaluate hardware performance. The RDE option does not require any additional main storage, however, the operator must enter a reason code in reply to message 1I89A (see DOS Version 4 Messages, GC33-5009), each time he performs the IPL procedure.

ERRLOG=YES provides RMSR support only; specification of ERRLOG=RDE provides support for both RMSR and RDE. If ERRLOG is not specified, ERRLOG=YES is assumed.

Note: For reasons of compatibility with earlier DOS releases, specification of ERRLOG=(YES,RDE) is also accepted by the assembler with the effect that code is generated as if ERRLOG=RDE had been specified.

CONFG

The configuration (CONFG) macro instruction and its parameters define the system configuration and can specify generation of optional supervisor services. If the assumed options are all satisfactory, the only entry required is the CONFG macro itself without any parameters.

PARAMETERS FOR CONFG

MODEL={135 / 145 / 155} Specify the CPU model number. If this parameter is omitted, or if an incorrect model number is specified, supervisor generation is terminated.

Specify:

MODEL=135 for the IBM Model 3135 CPU
MODEL=145 for the IBM Model 3145 CPU
MODEL=155 for the IBM Model 3155 CPU.

DEC={NO / YES} Specify if the decimal feature is present.

FP={NO / YES} Specify only if the floating point feature is present.

PORT={NO / 155} Specify PORT=155 when running a 145 System on a Model 155 CPU, in order to prevent data loss. The PORT option will then reserve sufficient storage for the Model 155 extended logout area. If MODEL=135 or MODEL=155 is specified, PORT should not be specified.

Note: The storage protection feature has been made standard. The Interval Timer (at location 80) has also been made standard. If, however, you erroneously specify either SP= or TIMER= (irrespective of whether the operand is YES or NO), the assembler will accept your specification and merely react by generating an MNOTE, which need not be resolved.

STDJC

The Standard Job Control (STDJC) macro instruction and its parameters specify the standard settings for job control. If the assumed options are all satisfactory, the only entry required is the STDJC macro instruction itself, without any parameters. These standard options can be locally overridden by an OPTION statement, except LINES and DATE. LINES can be overridden by a SET command, DATE by a // DATE statement.

PARAMETERS FOR STDJC

DECK={YES / NO} Specify if language translators are to output object modules on SYSPCH.

LIST={YES / NO} Specify if language translators are to write source module listings and diagnostics on SYSLST.

LISTX={NO / YES} Specify if compilers are to write hexadecimal object module listings on SYSLST.

SYM={NO / YES} Specify if assembler is to output symbol tables on SYSPCH. SYM=YES may be specified if the PL/I (D) compiler is to produce a symbol and offset table listing, or if American National Standard COBOL is to produce a data division glossary.

XREF={YES / NO} Specify if assembler is to write symbolic cross-reference lists on SYSLST, or if American National Standard COBOL is to produce a cross-reference listing.

ERRS={YES / NO} Specify if compilers are to summarize all errors in source programs on SYSLST. Assembler and PL/I (D) always assume ERRS=YES.

CHARSET={48C / 60C} Specify either the 48- or 60-character set for PL/I (D) translator input on SYSIPT.

LOG={YES / NO} Specify for a listing of all control statements on SYSLST. LOG=NO suppresses the listing of all job control statements on SYSLST. Invalid statements and commands will be listed on SYSLST if it is assigned.

DUMP={YES / NO} Specify if a dump of the registers and main storage is to be written on SYSLST in case of an abnormal program end.

LINES={56 / nn} Specify the number of lines per page on SYSLST. The minimum is 30; the maximum is 99.

DATE= {MDY / DMY} Specify the format of the date MDY=month/day/year. DMY=day/month/year.

SPARM= {NO / YES} Specify if the &SYSPARM variable of the Assembler D (14K) is to be supported.

FOPT

This Optional Features (FOPT) macro and its parameters specify additional optional features that can be included in the supervisor. The following parameters of the FOPT macro are discussed further in the DOS System Programmer's Guide, GC24-5073: OC, IT, PC, TEB, TEBV, EVA, SKSEP, PTO, CBF, CCHAIN, TRKHLD, AB, WAITM, DASDFP, and SYSFIL. IT, TEB, EVA, DASDFP, TOD, and ZONE are discussed in DOS Version 4, GC33-5007.

PARAMETERS FOR FOPT

OC= {NO / YES} Specify if external interruptions (except timer) can be handled by problem programs. If YES is specified, the facility is available to all programs in MPS. The Tape Compare Utility program requires CC=YES. YES is required if emulator program operator services are to be requested through the INTERRUPT key. If RETAIN=YES or OLTEP=YES is specified, CC=YES is forced.

IT= {NO / BG / F1 / F2} Specify if the interval timer can be handled by problem programs. Timer support is available to only one program in MPS. QTAM requires F1.

PC= {NO / YES} Specify if a user program check routine will be used. Included in supplied supervisor because QTAM, FORTRAN IV, Basic FORTRAN, COBOL-D, PL/I (D), and RPG require PC=YES.

TEB= {NO / n} Specify if tape error statistics are to be accumulated and logged for the 2495 Tape Cartridge Reader, where n is the number of tape cartridge readers attached to the system. Choose a value of n that allows for possible future expansion of the system.

TEBV= {IR / CR} Specify if tape statistics are to be automatically accumulated in the PUB2 table and logged on the SYSREC file. For all standard labeled tapes, tape statistics are accumulated by volume. For unlabeled or nonstandard labeled tapes two types of error recording are available:

- Combined Recording (CR), and
- Individual Recording (IR).

When TEBV=CR is specified, the error statistics for all nonstandard and unlabeled tapes on a specific tape unit are accumulated until a standard labeled tape is mounted and opened on that unit. Then one recording of the statistics for the nonstandard, unlabeled tapes is made and the counters are reset in the PUB2 table.

Specify TEBV=IR to record tape error statistics on the SYSREC file and reset the PUB2 table counters at each OPEN for nonstandard and unlabeled tapes.

EVA={NO / (r,w)} Error volume analysis is supported (may be used with or without TEBV and TEB). r specifies the number of temporary read errors that can occur. When the number of temporary read errors that occurred exceeds the value of r, the operator is notified via a message on SYSLOG. The maximum value for r cannot exceed 254.

w specifies the number of temporary write errors that can occur. When the number of temporary write errors that occurred exceeds thevalue of w, the operator is notified via a message on SYSLOG. The maximum value for w cannot exceed 254.

SKSEP={NO / YES / n} Specify if SEEKs are to be separated from the remainder of channel programs. Seek separation allows other devices on the channel to be accessed (including other seeks) during the seek. YES indicates support for all DASD type devices specified by the DVCGEN macro at system generation time. n is the number of DASD devices to be supported and cannot be less than the number of DASD devices specified at system generation. The maximum is 254.

PD={NO / YES / n} The PD parameter must be specified if the problem determination programs (PDAIDS) are desired. PD=YES specifies that a minimum of 800 bytes is reserved for this function. Any amount between 800 and 10,240 bytes may be specified for n.

JA={NO / YES / (n_1,n_2,n_3)} Specify if Job Control Job Accounting Interface is to be supported..

JA=YES indicates support for each batch partition.

JA=(n_1,n_2,n_3) specifies additional support for SIO count of input/output devices, and JA=YES is generated. n_1, n_2, and n_3 indicate the number of I/O devices for which the number of SIOs are to be counted in the background, foreground 2, and foreground 1 partitions respectively. n may be any number from 0-255 and is independent of the number of I/O devices assigned to the system.

Job accounting tables for foreground partitions are generated only if MPS=BJF.

JALIOCS={NO / (s,l)} Specify for support of user-written job accounting routines containing LIOCS, and LIOCS with label processing.

s and l reserve a user save area and a label area equal to their specifications (i.e., if s equals 25, then a 25-byte save area is reserved). s can be any decimal number in the range 0 - 1024. If s is omitted, or if JALIOCS=NO, 16 bytes are still reserved in the supervisor. l can be any decimal number between 0 and 224.

PTO={NO/YES} Specify if the Physical Transient Overlap (PTO) feature is to allow the system to select tasks while Fetch is reading a fetched (or loaded) phase from the system residence file, or while I/O operations are being performed during Error Recovery Procedures (ERP). This overlap allows increased throughput. MPS=YES or BJF must be specified.

IDRA={NO/YES} Specify IDRA=YES if the Independent Directory Read-in Area (IDRA) feature is desired. IDRA=YES allows the system to read the directory containing the location of the phase specified in a FETCH or LOAD command into the IDRA. Thus, with the IDRA feature, throughput may increase when an error recovery procedure is in progress. IDRA=YES is valid only if MPS=YES or BJF in the SUPVR macro. If IDRA=YES is specified and PTO is not specified or equals NO, PTO=YES is assumed.

OLTEP={YES/NO} Specify OLTEP=YES if the on-line testing function is desired. OLTEP=YES allows the on-line test executive program to execute. If OLTEP=YES, then OC=YES is forced.

RETAIN={NO/YES} RETAIN is an OLTEP function that allows the OLTEP programs to be executed, initiated from a remote location. RETAIN is a problem determination tool used by IBM. If this function is to be used, specify RETAIN=YES and include the IBM 2955 Retain Communications Device in the PUB table. External interruptions are required for RETAIN operation; therefore, if RETAIN=YES is specified, OC=YES is forced. If RETAIN=YES is specified, OLTEP=YES is also forced.

The RETAIN function is available on the Models 135, 145, and 155 in the United States of America only.

PCIL={NO/YES} Specify if private core image libraries are to be supported. These have the same format as the system core image library on SYSRES. You may assign private core image library to any partition. Selection of the batched-job foreground (MPS=BJF) option and the private core image library option gives the ability to link-edit in any batched-job partition. You may link-edit many IBM-supplied programs to run in a foreground partition and place them in a private

core image library to be assigned to that partition. Specifying PCIL=YES and MPS=NO or YES provides private core image library support for the background only.

CBF={NO / n} Specify the number of I/O requests to be buffered under the following conditions:

1. The actual record to be written does not exceed 80 characters.
2. Data or command chaining is not performed.
3. The CCB associated with this operation does not indicate the acceptance of unrecoverable I/O errors, posting at device end, or user error routines.
4. The CCB does not request sense information.

Console buffering allows overlap of CPU processing with write operations to SYSLOG by satisfying the requestor's WAIT immediately, rather than at I/O completion time. When this option is selected, the number of CHANQ entries should exceed the number of CBF entries so that the buffering process is not bound by the number of CHANQ entries. If the assumed option is taken for CHANQ and CBF is selected, the number of CHANQ entries is assumed to be six more than the CBF entry. Numbers 1 through 50 are valid, and one is assumed if the operand is invalid.

CCHAIN={NO / YES} Specify if command chaining support for retry on I/O operations is to be handled. When command chaining and an error occurs, CCHAIN allows the user to retry at the last CCW executed. Normal retry would return to the first CCW in the channel program. To make use of this option, you must have the appropriate bit set in the CCB. CCHAIN=YES must be specified if data or command chaining of IBM 2495 Tape Cartridge Reader input is performed, or if command chaining is required to read or write multiple tape records.

TRKHLD={NO / n} Specify if the track hold function is to be supported. When processing sequential disk workfiles or updates to direct access files, specify if a hold is to be placed on the track of the record being read. The hold prevents anyone else that is using track hold from accessing that track. The maximum number of tracks that can be held at one time is 255. The default is 10 if n is an invalid parameter (nonnumeric or outside the range 1-255). MPS=YES or BJF must be specified if TRKHLD=n.

AB={NO / YES} Specify if the abnormal termination exit function is to be supported. The abnormal termination exit allows you to exit to a user's routine before an abnormal end-of-job causes a program to be cancelled. Specify YES for American National Standard COBOL I/O error recovery.

WAITM={NO / YES} Specify if the multiple wait function is to be supported. This function allows you to use the WAITM macro to wait for one of a number of events to occur. WAITM=YES is assumed if AP=YES.

DASDFP={(n,n[,2311 / ,2314 / ,3330])} / {,2321}

Specify if supervisory DASD file protection is handled where n,n indicates the range of channels to which DASDs may be attached. Either 2311 or 2314 indicates file protection for 2311, 2314, and 2319. 3330 indicates file protection for 2311, 2314, 2319, and 3330. Specification of 2321 indicates file protection for all DASD devices. If the third operand is omitted, file protection for 2311, 2314, and 2319 is automatically provided. DASDFP prevents you from writing outside the extents of your file in case of program error. Extents are protected to the nearest cylinder except for 2321, where they are protected to the nearest head bank. This feature does not protect the file from being overwritten.

Note: If a disk device type is specified by the third operand and this disagrees with the one as specified by the DISK parameter of the PIOCS macro, the value of this parameter will overrule the one specified by the third operand.

SYSFIL={{NO / YES}[,n_1,n_2]}

Specify if system input and system output (SYSRDR, SYSIPT, SYSLST, SYSPCH) files may be assigned to a disk device. Specification of YES gives support for all disk devices. If MPS=BJF in the SUPVR macro, this parameter supports foreground logical units when running in batched mode. If the emulator program parameter SYSIO=222 or SYSIO=333 is indicated, a value must be specified for SYSFIL.

n_1 = residual capacity for beginning of operator notification when SYSLST assigned to disk. $100 \leq n_1 \leq 65535$. If n_1 is omitted, 1000 is assumed.

n_2 = residual capacity for beginning of operator notification when SYSPCH assigned to disk. $100 \leq n_2 \leq 65535$. If n_2 is omitted, 1000 is assumed.

Note 1: If neither n_1 or n_2 is specified, the operand need not be placed between parentheses ().

Note 2: Specification of either SYSFIL=2311 or SYSFIL=2314 is also accepted by the assembler with the effect that code is generated as if SYSFIL=YES had been specified.

TOD={NO / YES} Specify if time of day clock support is desired. If YES is specified, GETIME support is provided. Every time a GETIME is issued, the date fields in the supervisor communication regions are updated, if necessary.

ZONE={NO / (EAST / WEST, hh,mm)} Specify the difference between Greenwich Mean Time (GMT) and local time in hours (hh) and minutes (mm). Use EAST for areas to the east of the meridian of Greenwich, Great Britain, and WEST for areas to the west of the meridian of Greenwich, Great Britain. The value used in the ZONE parameter is used to obtain the local time-of-day. If the parameter is not, or incorrectly specified, or if TOD=NO is specified, ZONE=NO is assumed.

You can change the zone value (default is GMT) every time you perform the IPL procedure, by specifying any zone value in the ZONE parameter of the SET control statement.

Note: Information on macro instructions STXIT, EXIT, and SETIME is contained in the DOS Supervisor and I/O Macros, GC24-5037, and DOS Version 4, GC33-5007.

PIOCS

The Physical Input/Output Control System (PIOCS) macro instruction and its parameters define the configuration requirements to be supported by physical IOCS. If the assumed options are all satisfactory, the only entry required is the PIOCS macro itself, without any parameters.

PARAMETERS FOR PIOCS

SELCH={YES / NO} Specify if selector channels are attached to the system. YES must be specified if MODEL=25.

BMPX={NO / YES} Specify if burst mode devices are supported on multiplexor channel. If YES is specified, unbuffered devices will run in burst mode. If 1419s or 1255/1259/1270/1275s are attached to the multiplexor channel, BMPX=YES is not supported.

CHANSW={NO / RWTAU / TSWTCH} Specify if channel switching tape control unit. For a 2404 or 2804, enter CHANSW=RWTAU. For a 2816 (with a 2403 or 2803) or a 3803 enter CHANSW=TSWTCH.

TAPE={NC / 9 / 7} Indicates required tape PIOCS support. If TEB, TEBV, or EVA is specified and TAPE is not, then TAPE=9 is assumed.

9 = nine track only.

7 = seven or nine track.

NC = No tape drives attached.

MRSLCH={NO / YES} Specify if MICR device is on the selector channel.

DISK={2311 / 2314 / 3330} Specification of 2311 or 2314, or omission of the parameter, gives PIOCS support for both 2311 and 2314/2319. Specification of 3330 gives support for all disk devices.

ALLOC

ALLOC F1=nK,F2=nK The Allocate (ALLOC) macro instruction specifies the storage partitioning of MPS, where n must be a multiple of 2. This macro is optional. Most IBM components require 14K bytes of background area. Foreground area must be a minimum of 14K to allow BJF processing. The ALLOC macro is discussed further in the DCS System Programmer's Guide.

System-Used Areas

Each foreground partition contains a save area for program name, old program status word, and registers. The background save area is located in the supervisor. All partitions contain a label area for label processing if the LBLTYP statement is used. Both areas (except for the background save area) are at the low end of the partition.

Save area length is 88 bytes or 120 bytes if the floating point feature (FP=YES) was specified in the CONFG macro.

Label area length is determined by the system according to the LBLTYP card specification:

- TAPE (standard tape labels) = 80 bytes
- NSD (nn) (nonsequential disk) = 84 bytes + 20 bytes per extent statement
- Omitted = 0

Save Area Consideration

If you have a background job that uses nearly all the partition and you plan to run that same job in the foreground, you may need a foreground partition larger than background. For example, assume you specified FP=YES, your background partition is 14K (14,336 bytes), and your background job, including the label area, is 14,290 bytes. Because of the save area, the job requires an additional 120 bytes in foreground and exceeds 14K. The foreground partition, then, will have to be at least 16K to run the job unchanged. This applies to both IBM and user programs.

IOTAB

The Input/Output Tables (IOTAB) macro instruction and its parameters define the area for the necessary device tables for the system. If the assumed options are all satisfactory, the only entry required is the IOTAB macro itself without any parameters.

PARAMETERS FOR IOTAB

BGPGR={10 / n} Specify the number of Logical Unit Blocks (LUBs) for programmer units, i.e., the number of symbolic programmer logical units (SYS000-SYSnnn) assigned to the background partition. The minimum number is 10. The maximum numbers are 222 when MPS=BJF and 244 when MPS=YES or NO. See Note, which follows.

F1PGR={5 / n} If MPS=YES or BJF, specify the number of symbolic programmer logical units (of the class SYSnnn) for F1. If MPS=NO, F1PGR is not valid, and zero is assumed. The minimum acceptable value when MPS=YES or BJF is 5. The maximum numbers are 222 when MPS=BJF and 244 when MPS=YES. See Note, which follows.

F2PGR={5 / n} If MPS=YES or BJF, specify the number of symbolic programmer logical units (of the class SYSnnn) for F2. If MPS=NO, F2PGR is not valid, and zero is assumed. The minimum acceptable value when MPS=YES or BJF is 5. The maximum numbers are 222 when MPS=BJF and 244 when MPS=YES. See Note, which follows.

JIB={5 / n} Specify the number of Job Information Blocks (JIBs) for the system. One is required for each logical unit temporarily reassigned by a // ASSGN statement that differs from standard system assignment (i.e., established at IPL time). One JIB is required for each alternate

logical unit assignment. One JIB is required for each open 2311, 2314, or 2319 extent with the DASD file-protect feature except for system input/output extents. Two JIBs are required for each open 2321, or 3330 extent with the DASD file-protect feature. The minimum value generated is 5. The maximum value generated is 255. This parameter is discussed further in the DCS System Programmer's Guide, GC24-5073.

CHANQ={6 or 6+CBF / n}

Specify the number of entries in the channel queue. The minimum value generated is 6. If the assumed option is taken and CBF is specified, the assumed option is six more than the CBF entry. The number of the channel queue should exceed the total number of I/O requests you wish to accumulate simultaneously. The start I/O commands for all channels are stored in this queue. The maximum value is 255. This parameter is discussed further in the DOS System Programmer's Guide.

IODEV={10 / n}

Specify the number of I/O devices attached to the system. The maximum is 255. The minimum value is 5.

Note: The sum of BGPGR + F1PGR + F2PGR LUBs must not exceed 222 when MPS=BJF and 244 when MPS=YES.

D2311={0 / n}
D2314={0 / n}
D2321={0 / n}
D3330={0 / n}
D2400={0 / n}
D3410={0 / n}
D3420={0 / n}

This parameter is used to reserve PUB2 Table space for devices that require more than the normal value of 12 bytes per device for each PUB2 Table entry.

For each device or drive (for 2314/2319 or 3330) listed at the left, specify n, where n is the number of devices or drives of that type attached to your system. If the device is not specified, the system assumes that no devices of that type are attached to the system. However, if no disk devices are specified, the system assumes D2311=2. Likewise, if no tape devices are specified, D2400=4 is assumed.

During system generation, the amount of space reserved for the PUB2 Table is calculated. An adequate amount of space is reserved for each device specified. Then, the total number of these devices is subtracted from the total possible number of I/O devices (IODEV=n) attached to the system, resulting in the number of devices for which the normal amount of PUB2 Table space is reserved.

Allow additional PUB2 Table space for future expansion. If there is not enough PUB2 Table space available for all the devices attached to the system at IPL time, the system is unable to perform the IPL procedure.

DVCGEN

The Device Generation (DVCGEN) macro instruction and its parameters define the physical input and output units attached to the system. This macro instruction is optional. The DVCGEN macro is further discussed in the DOS System Programmer's Guide.

PARAMETERS FOR DVCGEN

CHUN=X'cuu',DVCTYP=xxxxxx,CHANSW={NO|YES},MODE=X'ss'

CHUN=X'cuu' Specify the hexadecimal number of the channel and unit for the device.

DVCTYP=xxxxxx Specify the device type. Figure 1-5 contains the codes for each IBM-supported device.

CHANSW={NO|YES} Specify if the device is attached to more than one selector channel. This parameter indicates if the device can be switched (IBM 2816 (with a 2403 or 2803), 2804, 2404, or 3803 available).

MODE=X'ss'

1. 2400T9. MODE specifies the tape mode. X'C0' is the default value. For 800 bpi 9-track single density, specifying X'C8' saves time during tape OPEN.

2. 3410T9 or 3420T9. MODE specifies the tape mode. X'C0' is the default value.

3. 2400T7, 3410T7, or 3420T7. MODE specifies the tape mode. X'90' is the default value.

4. 2702. MODE designates the SADxxx command. X'00' is the default value.
   X'00' SAD0
   X'01' SAD1
   X'02' SAD2
   X'03' SAD3
   See IPL Control and ASSGN Statements for System Generation for other values of ss.

5. 2260 (Local) and 3270 (Local). MODE specifies the 1053, 3284, or 3286 printer when CHUN=X'cuu' refers to a 1053 attached to a 2848, a 3284 attached to a 3272, or a 3286 attached to a 3272, respectively. The operand must be entered as MODE=X'01'.

6. 1419/1255/1259/1270/1275. MODE designates the external interrupt bit associated with magnetic ink character readers or optical reader/sorters. The mode X'01' through X'20' corresponds to external interrupt PSWs bits 26 through 31, respectively. For dual address adapter 1419, this parameter is needed for both 1419P and 1419S.

   X'01' Device attached to external line 7.
   X'02' Device attached to external line 6.

X'04' Device attached to external line 5.
X'08' Device attached to external line 4.
X'10' Device attached to external line 3.
X'20' Device attached to external line 2.

7. 1018. MODE specifies whether the error-correction feature is present or not.
X'00' is the default value
X'01' feature is present.

RULES FOR USING DVCGEN

1. A separate DVCGEN macro instruction is required for each device. For a 2314, 2319, or 3330 each individual unit needs a DVCGEN card.

2. The total number must not exceed the total number of devices specified in the IODEV parameter of the IOTAB macro.

3. DVCGEN macros must be specified in ascending channel address sequence.

4. Switchable units (attached to more than one selector channel) must be defined only once on the lowest channel by which they are addressable. They cannot be redefined as nonswitchable units on the higher channel.

5. The sequence of the DVCGEN cards determines the priority of the devices on their channel. SYSRES should be the first DVCGEN card if it is to have the highest priority for that channel. Switchable units must be the last devices specified for each channel and must be on consecutive channels.

6. The specifications of these macros can be altered at IPL by ADD and DEL statements.

7. IBM 3210/3215 Console Printer-Keyboards that are not on-line but were defined by DVCGEN statements must be deleted by DEL statements when performing IPL from the card reader.

ASSGN

The Assign (ASSGN) macro instruction and its parameters assign symbolic device names (LUBs) to physical I/O devices (PUBs). A separate macro instruction is required for each symbolic device name with a standard system generation assignment. This macro instruction is optional.

PARAMETERS FOR ASSGN

SYSnnn,X'cuu'[,BG / ,F1 / ,F2] Symbolic name may be any of the following system logical units or programmer logical units:

SYSRDR SYSLST SYSREC
SYSIPT SYSLOG SYSRLB
SYSPCH SYSLNK SYSSLB
SYS000-SYS221 (if MPS=BJF) or
-SYS243 (if MPS=NO or YES)

X'cuu' is the hexadecimal number of the channel and unit to which the symbolic device is attached. A separate macro is required for each standard assignment desired.

BG, F1, or F2 specifies the partition to which the assignment is being made. To make standard foreground assignments at system generation time, MPS=YES or BJF must be specified in the SUPVR macro. The ASSGN macro is ignored if an invalid partition specification is entered.

Making a standard foreground assignment at system generation time is equivalent to making a permanent assignment after IPL (exception noted). When you use single program-initiated foreground partitions, the assignments are released at the end of the first job unless the HOLD command is issued. Also, when you unbatch a foreground partition, the assignments are released at the end of the job unless the HOLD command is issued.

System input/output units (SYSRDR, SYSIPT, SYSLST, or SYSPCH) that are assigned to a tape or DASD when the system is generated are unassigned by IPL. An unassigned device can cause a job to be canceled.

Notes

- During system generation, conflicting assignments (ASSGN macro specifications made in the supervisor assembly) across partitions are not checked.
- If SYSLOG is assigned to a foreground partition, it must also be assigned in the background partition.
- SYSIN and SYSOUT cannot be specified in the ASSGN macro.
- If SYSCLB is assigned, the assignment is ignored.
- Compile, link-edit, and go in a foreground partition is only allowed if a private core image library is supported and available. Therefore, SYSLNK can only be assigned to a foreground partition if PCIL=YES and MPS=BJF have been specified.

SEND (n)

The Supervisor End (SEND) macro instruction indicates the address at the end of the supervisor. n is a multiple of 8 and cannct be greater than 30,720.

The steps in this sectiон are a general guide tc the use of the SEND macro. By fcllowing this procedure, you can generate a desirable supervisor without having tc examine all aspects of the SEND macro. The full description of the SEND macrc follows these steps.

For a Multiprogramming Systeм

Step 1: Assemble the supervisor with no SEND address specified. The address of NUCEND will equal the address of SYSEND.

Step 2: If PTO is specified or assumed YES (if IDRA=YES, PTO is forced), be sure the PTA (physical transient area) is storage protected. To do this, add 400 (hex) to the PTA address, then subtract this sum from the address of EOSSP.

If the difference is zerc or pcsitive, the PTA is stcrage prctected (as in Figure 1-2C or 1-2D). Go to Step 3 if the difference is zero or positive.

If the difference is negative, the PTA is not storage protected (Figure 1-2B). In this case add at least the difference between ECSSP address + 8 and the PTA address tc the PPBEG address, and make the decimal equivalent the SEND address. Thus:

SEND address = PPBEG ADDRESS + (EOSSP address + 8 less PTA address)

The number of bytes added tc the PPBEG address is reserved for future expansion of the supervisor (Figures 1-2H and 1-2I)

Go to step 4 if the difference is negative.

Step 3: When the difference between the PTA+400 and ECSSP is zero or positive, examine for unused space (desirable if ycu wish to reserve room for future expansion of the supervisor without re-link editing your programs). Figure 1-2D shows unused space between the end of the supervisor and the beginning of the prcblem prcgram area. To determine how much unused space exists, subtract the address of PPBEG from the EOSSP address. If the difference is greater than zerc, that result is the number of bytes (in hexadecimal notation) of unused space. If you do not desire the extra space at this point, add the decimal equivalent to the PD=n specification in the FOPT macro.

This extra space is then available for use by IBM personnel in servicing the supervisor until the time you feel it is necessary tc use a portion of that space in expanding your supervisor. For example, if you are to have PDAIDs running in your system, PD=800 is the minimum specification. If you find X'158' (decimal 344) bytes of unused space between PPBEG and EOSSP, specify PD=1144 (800 + 344). Ccntinue to Step 4.

Step 4: If you found no reason tc modify a parameter in Step 2 or Step 3, continue with generation prccedures. Otherwise, make the desired change and reassemble. Make sure that the NUCEND address is equal to cr less than the SYSEND address (this could happen with the SEND address

specified). If the NUCEND address is greater than the SYSEND address, add at least the decimal equivalent of the difference to the specified SEND address and reassemble. You may wish at this time to consider Step 3 again. Also, if NUCEND is less than SYSEND (this can occur with the SEND address specified), some extra space has been generated and can be used to expand the supervisor. This can be left as is or added to the PD=n parameter as discussed in Step 3.

Note: To obtain more space for future expansion, you may add 2nK bytes (decimal notation, where n≥1) to the SEND address, provided you don't exceed the maximum of 30,720.

For a Non-Multiprogramming System

Step 1: Assemble the supervisor with no SEND address specified. The address of NUCEND will equal the address of SYSEND.

Step 2: Make sure that the LTA (Logical Transient Area) is storage protected. You should do this by adding 4B0 (hexadecimal notation) to the LTA address; then subtract the result from the EOSSP address.

If the result is zero or positive, the LTA is storage protected. Proceed with step 3.

If the result is negative, the LTA is not storage protected (Figure 1-2B). In this case add at least the difference between EOSSP address + 8 and the LTA address to the PPBEG address and make the decimal equivalent the SEND address. Thus:

SEND address = PPBEG address + (EOSSP address + 8 less LTA address)

The number of bytes added to the PPBEG address is reserved for future expansion of the supervisor.

Step 3: If you reassembled with a SEND address specified, you need only make sure that the NUCEND address is not greater than the SYSEND address. Subtract NUCEND address from SYSEND address. If the difference is negative, add this amount (decimal equivalent) to the SEND address and reassemble and repeat Step 3. In actual practice, this should not occur, but it is wise to check the addresses as a precaution.

Note: To obtain more space for future expansion, you may add 2048 bytes (decimal notation) to the SEND address.

Figure 1-2 illustrates supervisor storage generated by various combinations of n (SEND address), supervisor size, and storage protection boundary.

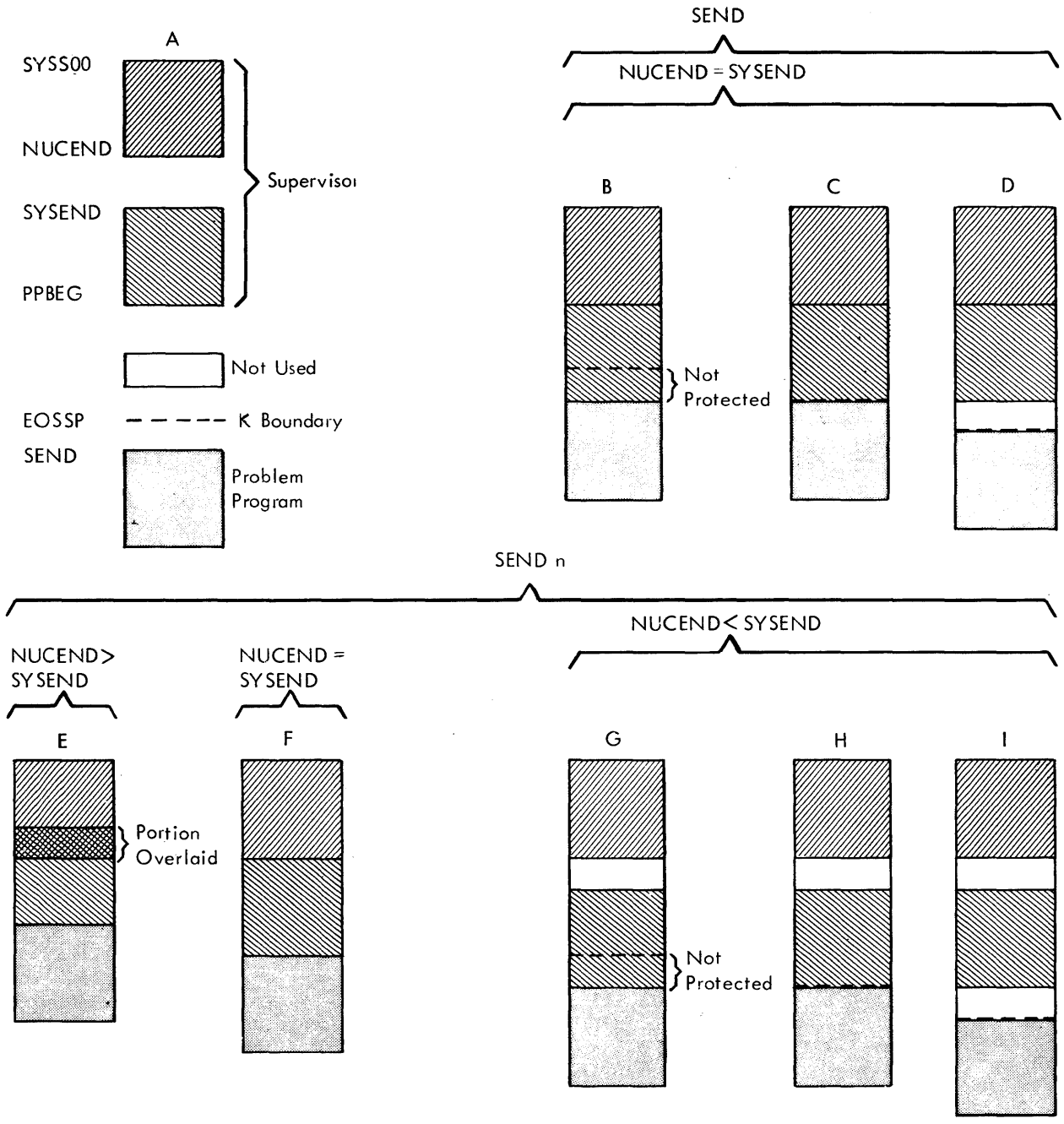


Figure 1-2. SUPERVISOR--Unprotected Supervisor Storage Generated

The supervisor consists of two parts:

1. The nucleus that extends from the address SYSS00 to NUCEND, and
2. The area that extends from the address SYSEND to PPBEG. It consists of:

- The Independent Directory Read-In Area (Optional, IDRA=YES),
- The Logical Transient Area (LTA),
- The Physical Transient Area (PTA),
- The Problem Determination Aids Area (Optional, PD=YES, or PD=n), and
- The Background Save area.

The End-of-Supervisor Storage-Protect (EOSSP) address is either:

- the first 2K boundary after the address of the LTA - if MPS=NO was specified, or
- the first 2K boundary after the address of the PTA - if either MPS=YES or MPS=BJF was specified.

SEND

If the operand n is not specified (Figure 1-2B), the address of NUCEND equals the address of SYSEND.

SENDn

If the operand n is specified (Figures 1-2E through 1-2I), the address of NUCEND can be greater than, equal to, or less than the address of SYSEND.

1. If NUCEND is greater than SYSEND (Figure 1-2E), a portion of the transient area overlays the supervisor nucleus. n must be increased, and the supervisor reassembled.

2. If NUCEND is equal to SYSEND (Figure 1-2G), the result is identical to the case where n was not specified (see Figures 1-2B, 1-2C, and 1-2D) and the effect is also the same.

3. If NUCEND is less than SYSEND (Figures 1-2G through 1-2I), the area between SYSEND and NUCEND is not used and is available for future expansion of the supervisor. The difference between SYSEND and NUCEND is the number of bytes the supervisor may expand without having to relinkage edit programs at the end of the supervisor.

SENDn, NUCEND<SYSEND (Figures 1-2G, 1-2H, and 1-2I)

1. If the EOSSP address is less than the PPBEG address, the area between these two addresses is not storage-protected. (Figures 1-2B and 1-2G). This happens if MPS=NO was specified, or if either MPS=YES or MPS=BJF was specified and the EOSSP address is somewhere within the area that contains the Physical Transient Area (PTA), the Problem Determination Aids Area and the Background Save Area.

2. If the EOSSP address is equal to the PPBEG address, the entire supervisor is storage-protected (Figures 1-2C and 1-2H).

3. If the EOSSP address is greater than the PPBEG address, the entire supervisor is storage-protected. The area between the PPBEG address and the EOSSP address is not used and is available for future expansion of the supervisor. The difference between the EOSSP address and the PPBEG address is the number of bytes the supervisor may expand without having to relinkage edit programs at the end of the supervisor. This expansion area is in addition to any area between SYSEND and NUCEND (Figure 1-2I).

MAXIMUM UNPROTECTED STORAGE

If a portion of the supervisor is not storage-protected (Figure 1-2B and 1-2G), verify that the unprotected area falls within the following limits.

	PD=NO	PD=YES	PD=n
MPS=NO PTO=NO FP=NO/YES	2344	3184	2384+n
MPS=YES/BJF PTO=NO FP=YES	1176	2016	1216+n
MPS=YES/BJF PTO=NO FP=NO	1144	1984	1184+n
MPS=YES/BJF PTO=YES FP=YES	120	960	160+n
MPS=YES/BJF PTO=YES FP=NO	88	928	128+n

Figure 1-3. SUPERVISOR--Maximum Unprotected Area between ECSSP and PPBEG (End of Supervisor) Address

Figure 1-3 shows the maximum number of bytes in the area between the EOSSP address and the PPBEG address that may be unprotected.

Operation	Operand	Explanation
SUPVR		Supervisor macro instruction.
	SYSTEM=DISK	Specify a disk-resident supervisor. SYSTEM=DISK is assumed if this parameter is omitted.
	MPS={NO / YES / BJF}	Specify if there is to be multiprogramming support. When YES or BJF is specified, the system generated is capable of supporting two foreground programs. YES or BJF must be specified if TP=QTAM. BJF must be specified if batched job environment is desired for foreground partitions.
	TP={NO / BTAM / QTAM / QTAMn}	Specify if teleprocessing support is desired and if so, whether Basic or Queued Access Method (BTAM or QTAM) is desired. When QTAM is specified, SVC support for BTAM is also included. n is the maximum number of problem programs in the system at one time. n may be any value 2 to 12. (For multitasking, AP must equal YES.)
	MICR={NO / 1412 / 1419 / 1419D}	Indicates whether the supervisor is to support magnetic ink character readers or optical reader/sorters. If 1412/1419s or 1255/1259/1270/1275s are attached to the multiplexor channel, the PIOCS parameter BMPX=YES is not supported.
	ASCII {NO / YES}	Specify YES, if supervisor support of ASCII-code is desired.
	AP={NO / YES}	Specify if there is to be multiprogramming within a partition (multitasking) support. Multiprogramming within a partition provides the ability to execute more than one program (multitasking) within a partition. MPS=YES and WAITM=YES are assumed if AP=YES.
	EU={NO / YES}	EU = YES must be specified if mixed parity tape processing is required for the emulators, the tape preprocessor and tape postprocessor programs.
	ERRLOG={YES / RDE}	YES provides Recovery Management Support Recording (RMSR). RDE provides both RMSR and the Reliability Data Extractor (RDE).
CONFG		Describes the hardware features.
	MODEL={135 / 145 / 155}	Specify the model number.
	SP={NO / YES}	Storage protection feature. YES must be specified for MPS or BJF.
	DEC={NO / YES}	Decimal feature.
	FP={NO / YES}	Floating point feature.
	PORT={NO / 155}	If you specify MODEL=145, then PORT=155 can be specified to provide support for the extended logout area of the Model 155. This allows the MCAR/CCH functions of the supervisor you are generating to be CPU independent.

Figure 1-4. SUPERVISOR--Macro Instructions (Part 1 of 4)

Operation	Operand	Explanation
STDJC		Specify standard settings for job control.
	DECK= { YES / NO }	Output of object modules of language translators on SYSPCH.
	LIST= { YES / NO }	Source module listings and diagnostics from language translators on SYSLST.
	LISTX= { NO / YES }	Hexadecimal object module listings from PL/I (D) and COBOL on SYSLST.
	SYM= { NO / YES }	Assembler and PLI (D) outputs symbol tables on SYSPCH: COBOL-D compiler outputs DATA DIVISION map on SYSLST. The American National Standard COBOL compiler produces a data division glossary.
	XREF= { YES / NO }	Assembler and American National Standard COBOL Compiler outputs symbolic cross reference lists on SYSLST.
	ERRS= { YES / NO }	COBOL, PL/I (D), FORTRAN, and Basic FORTRAN summarize all errors in source programs on SYSLST.
	CHARSET= { 48C / 60C }	Specify the 48- or 60-character set for PL/I (D) input on SYSIPT.
	LOG= { YES / NO }	Listing of all control statements on SYSLST.
	DUMP= { YES / NO }	Dump of registers and main storage on SYSLST.
	LINES= { 56 / nn }	Number of lines per page on SYSLST.
	DATE= { MDY / DMY }	Format of the date.
	SPARM= { NO / YES }	Support of Assembler variable symbol &SYSPARM.
FOPT		Specify optional support in the supervisor.
	OC= { NO / YES }	STXIT option is available for external interrupt (except timer). YES is required for tape compare utility program.
	IT= (NO / BG / F1 / F2)	STXIT option is available for interval timer interruption in the area specified. TIMER=YES is assumed.
	PC= { NO / YES }	STXIT option is available for program check interruption. Included in supplied supervisor because FORTRAN, COBOL-D, RPG, QTAM, PL/I and Autotest require PC=YES.
	TEB = { NO / n }	Specify if tape error statistics are to be accumulated and logged for the 2495 Tape Cartridge Reader where n is the number of tape cartridge readers attached to the system. Allow extra TEBSs for future expansion of the system.
	TEBV = { IR / CR }	Specify the type of error recording (Combined Recording or Individual Recording) to be performed for unlabeled or nonstandard tapes.
	EVA= { NO / (r,w) }	Specify if error volume analysis is supported. r is the read error parameter; w is the write error parameter; and n is the number of tape drives attached to the system.**
	SKSEP= { NO / YES / n }	Specify if SEEK's are to be separated from the remainder of channel programs. Seek separation allows other devices on the channel to be accessed (including other seeks) during the seek. YES indicates support for all DASD type devices specified by the DVCGEN macro of system generation time. n is the number of DASD devices to be supported and cannot be less than the number of DASD devices specified at system generation. The maximum number is 254.
	PD= { NO / YES / n }	Specify the number of bytes to be allocated to the problem determination programs. 800 is the minimum number of bytes that can be specified. Specification of YES provides the minimum number of 800 bytes.
	JA= { NO / YES / (n1,n2,n3) }	Specify if the job control job accounting interface is to be supported.

Figure 1-4. SUPERVISOR--Macro Instructions (Part 2 of 4)

Operation	Operand	Explanation
FOPT (Continued)	JALIOCS= { NO / (n1,n2) }	Specify for support of user-written job control job accounting interface routines containing LIOCS, and LIOCS with label processing.
	PTO= { NO / YES }	Specify if the physical transient overlap feature is to allow the system to select tasks while FETCH is reading a fetched (or loaded) phase from the system residence file, or while I/O operations are performed during error recovery procedures.
	IDRA= { NO / YES }	Specify if the independent directory read-in area (IDRA) is desired. If so, MPS must equal YES or BJF and PTO=YES is assumed.
	OLTEP= { YES / NO }	Specify if the on-line testing function is desired. If OLTEP YES, then OC YES is forced.
	RETAIN= { NO / YES }	Specify YES if the data link to the Remote Analysis Center is to be supported. The RETAIN function is available in the USA only.
	PCIL= { NO / YES }	Specify if private core image libraries are to be supported. This allows the user to assign a private core image library.
	CBF= { NO / n }	Specify if I/O requests are to be appraised for console buffering and indicate the number of buffers (1-50) to be generated.
	CCHAIN= { NO / YES }	Specify if command chaining support for retry on I/O operations is to be handled.
	TRKHLD= { NO / n }	When processing sequential disk workfiles or updates of direct access files, specify if a hold is to be placed on the track of the record being read. The hold prevents anyone else who is using track hold from accessing that track. The maximum number of tracks that can be held at one time is 255 and the assumed value is 10.
	AB= { NO / YES }	Specify if the abnormal termination exit function is to be supported. The abnormal termination exit allows the user to exit to a user's routine before an abnormal end of job causes a program to be cancelled. Specify YES for American National Standard COBOL I/O error recovery.
	WAITM= { NO / YES }	Specify if the multiple wait function is to be supported. This function allows the user to use the WAITM macro to wait for one of a number of events to occur.
	DASDFP= { NO / (n,n,{2311 / 2314 / 2321 / 3330}) }	Specify if supervisory DASD file protection is handled where n,n indicates the range of channels to which DASDs may be attached. Either 2311 or 2314 indicates file protection for 2311, 2314, and 2319. 3330 indicates file protection for 2311, 2314, 2319, and 3330. 2321 indicates file protection for all DASD devices.
	SYSFIL= { NO / YES / ({2311 / 2314}[,n1,n2]) }	Specify if system input and system output (SYSRDR, SYSIPT, SYSLST, SYSPCH) files may be assigned to a disk device. Specification of YES, 2311, or 2314 gives support for all disk devices n1 = residual capacity for beginning of operator notification where SYSLST assigned to a disk device. 100 n1 65535. If n1 is omitted, 1000 is assumed. n2 = residual capacity for beginning of operation notification where SYSPCH assigned to a disk device. 100 n2 65535. If n2 is omitted, 1000 is assumed. Note: If neither n1 nor n2 is specified, the operand need not be placed between parentheses ( ).
	TOD= { NO / YES }	Specify if the supervisor is to provide time-of-day clock support.
	ZONE= { NO / ({EAST / WEST},hh,mm) }	Specify the difference between Greenwich Mean Time and local time in hours (hh) and minutes (mm). Use EAST for areas east of Greenwich and WEST for areas west of Greenwich. If TOD=NO is specified, ZONE=NO is assumed.
PIOCS		Define options and configuration requirements to be included in physical IOCS.
	SELCH= { YES / NO }	Specify if selector channels are attached to the system.
	BMPX= { NO / YES }	Specify if burst mode devices on multiplexor channel is supported. If 1419s or 1255/1259/1270/1275s are attached to the multiplexor channel BMPX=YES is not supported.
	CHANSW= { NO / RWTAU / TSWTCH }	Specify if channel switching. For a 2404 or 2804 enter CHANSW=RWTAU. Enter CHANSW=TSWTCH for a 2816 (with a 2403 or a 2803, or a 3803.
	TAPE= { NO / 9 / 7 }	Specify 9- or 7-track tape. 7 indicates support for both. If TEB, TEBV, or EVA is specified and TAPE is not, then TAPE=9 is assumed.
	MRSLCH= { NO / YES }	Specify if MICR device is on the selector channel. (can only be specified for the 1419 single-address device).
	DISK= { 3330 / 2311 / 2314 }	Specification of 3330 gives PIOCS support for all disk devices. Specification of 2311, or 2314, or omission of this parameter, gives PIOCS support for 2311 and 2314 only.
ALLOC	F1=nK,F2=nK	Specify storage partitioning.

Figure 1-4. SUPERVISOR--Macro Instructions (Part 3 of 4)

Operation	Operand	Explanation
IOTAB		Define the necessary input/output tables for the system.
	BGPGR= {10 / n}	Specify the number of logical unit blocks (LUBs) for programmer units, i.e., the number of symbolic programmer logical units (SYS000 - SYSnnn).
	F1PGR {5 / n}	Specify the number of symbolic units of the class SYSnnn for F1.
	F2PGR= {5 / n}	Specify the number of symbolic units of the class SYSnnn for F2.
	JIB= {5 / n}	Number of JIBs for the system. Minimum value generated is 5.
	CHANQ= {6 / n}	Number of entries in the channel queue. Minimum value is 6.
	IODEV= {10 / n}	Specify the number of I/O devices attached to the system. The minimum value is 5.
	D2311 = {0 / n} D2314 = {0 / n} D2321 = {0 / n} D3330 = {0 / n} D2400 = {0 / n} D3410 = {0 / n} D3420 = {0 / n}	Specify the number of each of the devices listed that will be attached to your system. Specification of these parameters is required as each of these devices requires more than the minimum amount of 12 bytes for its PUB2 Table entry.
DVCGEN (See note 1)		Specify the physical I/O units attached to the system.
	CHUN = X'cuu'	Hexadecimal number of channel and unit.
	DVCTYP = xxxxxx	Specify the device type. See Figure 1 - 5.
	CHANSW= {NO / YES}	YES indicates that the device is attached to more than one selector channel (the device is switchable).
	MODE = X'ss'	1. 2400T9. MODE is used to specify the tape mode. X'C0' is the default value. 2. 3410T9 or 3420T9. MODE specifies the tape mode. X'C0' is the default value. 3. 2400T7 or 3420T7. MODE is used to specify the tape mode. X'90' is the default value. 4. 2702. MODE designates the SADxx command. X'00' is the default value. X'00' SAD0, X'01' SAD1, X'02' SAD2, X'03' SAD3, 5. 2260 (Local) and 3270 (Local). MODE is used to specify the 1053, 3284, or 3286 printer when CHUN = X'cuu' refers to a 1053 attached to a 2848 or to either a 3284 or a 3286 attached to a 3272. This operand must be entered as X'01'. 6. 1419, 1255, 1259, 1270, 1275 MODE designates the external interrupt bit associated with magnetic ink character reader. X'01' External line 7 X'04' External line 5 X'10' External line 3 X'02' External line 6 X'08' External line 4 X'20' External line 2 7. 1018. MODE specifies wether the Error Correction feature is present or not. X'00' is the default value. The operand must be entered as X'01' if the feature is present.
ASSGN		Assign LUBs to PUBs as standard system assignments.
	SYSnnn,X'cuu' [,BG / ,F1 / ,F2]	Symbolic unit is assigned a hexadecimal channel and unit number within a partition.
SEND	[n]	End of supervisor macro instructions. n = beginning address of the problem program area.

Note 1: Rules for using DVCGEN

1. A separate DVCGEN macro instruction is required for each device.
2. The total number of DVCGEN macros must not exceed the total number of devices specified in the IODEV parameter of the IOTAB macro.
3. DVCGEN macros must be specified in ascending channel address sequence.
4. Switchable units (attached to more than one selector channel) must be defined once. They are defined on the lowest channel on which they are addressable.
5. The sequence of the DVCGEN cards determines the priority of the devices on their channel. Switchable units must be the last devices for each channel, and must be on consecutive channels.
6. The specifications of these macros may be altered by IPL ADD and DEL statements. See IPL PLM, GY24-5086.

Note 2: Rules for using ASSGN

1. The ASSGN macro allows SYSRDR, SYSLST, SYSPCH, and SYSIPT to be assigned to a tape or DASD. However, IPL unassigns any such assignments.
2. SYSLOG must also be assigned in BG, if assigned in foreground partition.
3. SYSLNK cannot be specified in either foreground partition.

Figure 1-4. SUPERVISOR--Macro Instructions (Part 4 of 4)

Card Code	Actual IBM Device	Dev. Type X'nn'	Device Type
2400T9	9 - track Magnetic Tape Units } 2400 - series	50	Magnetic Tape Devices
2400T7	7 - track Magnetic Tape Units }	50	
3410T9	9 - track 3410 Magnetic Tape Units	53	
3410T7	7 - track 3410 Magnetic Tape Units	53	
3420T9	9 - track 3420 Magnetic Tape Units	52	
3420T7	7 - track 3420 Magnetic Tape Units	52	
2495TC	2495 Tape Cartridge Reader	51	Tape Cartridge Reader
1442N1	1442N1 Card Read Punch	30	Card Read Punches
2520B1	2520B1 Card Read Punch	31	
2596	2596 Card Read Punch	30	
3525RP	3525 Card Punch (with optional read feature)	23	
2501	2501 Card Reader	10	Card Readers
2540R	2540 Card Reader	11	
3505	3505 Card Reader	12	
2540P	2540 Card Punch	21	Card Punches
2520B2	2520B2 CardPunch	20	
1442N2	1442N2 Card Punch	22	
2520B3	2520B3 Card Punch	20	
3525P	3525 Card Punch	23	
1403	1403 Printer	40	Printers
1403U	1403 Printer with UCS feature	42	
1443	1443 Printer	41	
2245	Kanji Printer	44	
2260 (local)	1053 Printer with 2848 Control Unit. MODE operand must be entered as X'01'	C0	
3211	3211 Printer	43	
3277 (local 3270)	3284 or 3286 Printer with 3272 Control Unit. MODE operand must be entered as X'01'	B0	
3277B (local 3270)	3284 or 3286 Printer with 3272 Control Unit, attached in Burst Mode to a multiplexer channel. MODE operand must be entered as X'01'	B0	
1050A	3210, 3215 Console Printer Keyboards	00	Printer - Keyboards
UNSP	Unsupported Device	FF	Unsupported No burst mode on multiplexor channel
UNSPB	Unsupported Device	FF	Unsupported with burst mode on multiplexor channel
2311	2311 Disk Storage Device	60	DASD
2314	2314 Direct Access Storage Facility	62	
2314	2319 Disk Storage Facility	62	
2321	2321 Data Cell Drive	61	
3330	3330 Disk Storage	63	
1419	1255 Magnetic Character Reader	72	MICR - Magnetic Tape Character Recognition Devices
1419	1259 Magnetic Character Reader	72	
1419	1419 Magnetic Character Reader	72	
1419P	1419 Dual Address Adapter Primary Control Unit	73	
1419S	1419 Dual Address Adapter Secondary Control Unit	74	

Figure 1-5. SUPERVISOR--Device Code (Part 1 of 2)

Card Code	Actual IBM Device	Dev. Type X'nn'	Device Type
2701	2701/2715 Data Adapter Unit	D0	Teleprocessing lines A = SAD0 command when enabling the line B = SAD1 command when enabling the line C = SAD2 command when enabling the line D = SAD3 command when enabling the line
2702 A 2702 B 2702 C 2702 D	2702 Transmission Control Unit	D1	
2703	2703 Transmission Control Unit	D2	
2955	2955 Data Adapter Unit	D7	Data Link for RETAIN
1017	1017 Paper Tape Reader with 2826 Control Unit Model 1	78	Paper Tape Readers
1017TP	1017 Paper Tape Reader with 2826 Control Unit Model 2	D5	
2671	2671 Paper Tape Reader	70	
1018	1018 Paper Tape Punch with 2826 Control Unit Model 1	79	Paper Tape Punches
1018TP	1018 Paper Tape Punch with 2826 Control Unit Model 2	D6	
1419	1270 Optical Reader Sorter	72	Optical Readers
1419P	1275 Optical Reader Sorter	73	
1287	1287 Optical Reader	77	
1288	1288 Optical Page Reader	77	
2260	2260 Display Station	C0	Display Stations
3277 (local 3270)	3277 Display Station; MODE operand need not be entered	B0	
3277B (local 3270)	3277 Display Station; attached in Burst Mode to a multiplexor channel. MODE operand need not be entered	B0	
7770	7770 Audio Response Unit	D3	Audio Response Units

Figure 1-5. SUPERVISOR--Device Code (Part 2 of 2)

Supervisor Main Storage Requirements

This section contains the data required for estimating the size of the supervisor required for a generated system, and the amount of main storage required at object time for the supervisor macro instructions.

The supervisor size can determine the size of the available problem-program area. Note that the background problem-program area must be at least 14,336 bytes when the disk operating system is used. If Assembler F is part of the system, the minimum background area must be 45,056 bytes. The supervisor varies in size from system to system according to the options chosen by the user and to the machine configuration.

Figure 1-6 gives the main storage requirements for the base supervisors and the elements that can be included in a tailored supervisor. The base supervisor requirement (MODEL=135) for the batched job system is 10,896 bytes. Additional storage requirements must be added to the base requirement for each additional supervisor element desired that is not within the base requirement. For example, for OC=YES (FOPT) under batched job system, add 136 bytes to the base storage requirement (10,896 bytes). The base requirement for the MPS=YES supervisor is 11,766 bytes. For TP=BTAM under MPS=YES, add 568 bytes to the base requirement for the MPS supervisor. Thus, an MPS supervisor that includes BTAM requires 12,334 bytes (see Figure 1-6). (Note that QTAM requires an MPS supervisor.) Thus, by the time the supervisor is tailored to the installation requirements, it is usually larger than the base requirements.

Note the relationship between the actual number of bytes in the 14K supervisor shipped by IBM and the number of bytes specified in the SEND macro for the supervisor.

Size of Supervisor Shipped by IBM (in bytes)	Address Specified In SEND Macro (in bytes)
14,336	14,336

The SEND macro specifies the beginning of the problem program area to facilitate system generation. Depending on the combination of supervisory functions chosen, a supervisor greater than 14,336 bytes may be generated.

Note: All supervisor generation options are described earlier in this module under Planning a Supervisor.

Supervisor Element	Generation Operand	MFS=		
		NO	YES	BJF
Required Routines (Basic Size)-SUPVR SVC Interruption Handling System Loader (FETCH and LOAD) I/O Units Control Tables (LUBS, PUBS, and JIBS) General Entry and Exit Routines Communication Region Transient Area End of Job Step Physical IOCS (including Selector Channel Support) Storage Protection		10,896	11,766	12,240
Optional Routines				
ASCII	ASCII=YES	512	512	512
Magnetic Character Options	MICR=1419 (fcr 1255/1259/1270/1275, see Note 1)	1614	1592	1600
	MICR=1419D	1392	1480	1480
Teleprocessing Options BTAM	TP=BTAM	536	568	568
QTAM	TP=QTAM		1200	1200
	TP=QTAMn (Note 2)		1088+[A]	1088+[A]
I/O Error Log (Note 9) Option (RMSR and RDE)	ERRLOG=YES, or RDE	0	0	0
Multitasking	AP=YES (Includes WAITM=YES)		2214	2198
1401/1440/1460 and 1401/7010 Emulator	EU=YES MODEL=135, 145, or 155	168	176	176

Figure 1-6. SUPERVISOR--Main Storage Requirements (Bytes) (Part 1 of 5)

Supervisor Element	Generation Operand	MPS=		
		NC	YES	BJF
Configuration Options-CONFG				
Model 135 (Note 9)	MODEL=135	0	0	0
Model 145 (Note 9)	MODEL=145	552	552	552
Model 155 (Note 9)	MODEL=155	872	872	872
Decimal Feature	DEC=YES	0	0	0
Floating Point Feature	FP=YES	0	104	112
Portability (Note 9)	PORT=155	320	320	320
Functional Supervisor Options-FOPT				
User Option to Handle: Operator Comm.	OC=YES	136	160	168
User Option to Handle: Prog Ck	PC=YES	208	240	240
User Option to Handle: Int Timer	IT=BG,F1,F2(Note 12)	448	456	456
Operator Comm. + Int Timer	Note: If AP=YES when PC=YES, add 168 bytes. If OC=YES when any MICR support is included, add 32 bytes.	472	496	496
Prog Ck + Int Timer		488	520	528
Operator Comm. + Prog Ck		248	288	304
Operator Comm. + Prog Ck + Int Timer		512	536	544
Problem Determination Programs	PD=YES (Note 3)	984	992	1000
	PD=n (Note 3)	n+184	n+192	n+200
Seek Separation	SKSEP=YES or n (Note 4)	328+5n + [a]+[b]	328+5n + [a]+[b]	328+5n + [a]+[b]
Physical Transient Overlap	PTO=YES (Note 5)	--	336	352
Independent Directory Read/in Area (Note 5)	IDRA=YES		760	776
On-Line Testing	OLTEP=YES (Note 9)	416	440	438
RETAIN	RETAIN=YES	152	168	168
Private Core Image Lib.	PCIL=YES	248	296	384
Console Buffering	CBF=n n may be 1 to 50 buffers [if CHANQ is not elected--Note 6]	464+105n+ [7n]	520+105n+ [7n]	520+105n+ [7n]

Figure 1-6. SUPERVISOR--Main Storage Requirements (Bytes) (Part 2 of 5)

Supervisor Element	Generation Operand	MPS=		
		NO	YES	BJF
Job Accounting Interface	JA=YES (Note 12)	568	568	936
	JA=(n_1,n_2,n_3)	640+8 (n_1+n_2+n_3)	640+8 (n_1+n_2+n_3)	1040+8 (n_1+n_2+n_3)
	JALICCS=(s,l) (Note 11)	value from JA+ (s+1)-16	value from JA+ (s+1)-16	value from JA+ (s+1)-16
Time-of-Day Clock Support	TOD=YES	712	712	720
	ZONE= {(EAST,hh,mm) / (WEST,hh,mm)}	0	0	0
Multiple Wait (Note 7)	WAITM=YES	56	48	48
Abnormal Termination	AB=YES (without AP=YES)	352	336	344
	(with AP=YES)	--	616	600
Track Hold (Note 8)	TRKHLD=n (without AP=YES)	--	596 +12n	604 +12n
	TRKHLD=n (with AP=YES)	--	716 +12n	708 +12n
Tape Error Statistics by: Unit, Volume, and EVA (Note 10)	TEB=n	28+6n	36+6n	36+6n
	EVA=r,w	16	16	16
	TEBV=IR	0	0	0
	TEBV=CR	0	0	0
Command Chaining	CCHAIN=YES	32	32	32
Disk System Input and Output Files	SYSFIL=YES [,n_1,n_2]	416	416	632
DASD File Protection (with 2321)	DASDFP=n_1,n_2,dev	446+ 24(n_2-n_1)	430+ 24(n_2-n_1)	446+ 24(n_2-n_1)
(with 2311 or 2314)		374+ 24(n_2-n_1)	358+ 24(n_2-n_1)	366+ 24(n_2-n_1)
(with 3330)		432+ 24(n_2-n_1)	424+ 24(n_2-n_1)	424+ 24(n_2-n_1)
DASDFP (with 2321) including Disk SYSFIL		848+ 24(n_2-n_1)	832+ 24(n_2-n_1)	1048+ 24(n_2-n_1)
DASDFP (with 2311 or 2314) including Disk SYSFIL		776+ 24(n_2-n_1)	760+ 24(n_2-n_1)	976+ 24(n_2-n_1)
DASDFP (with 3330) including Disk SYSFIL		832+ 24(n_2-n_1)	800+ 24(n_2-n_1)	1016+ 24(n_2-n_1)

Figure 1-6. SUPERVISOR--Main Storage Requirements (Bytes) (Part 3 of 5)

Supervisor Element	Generation Operand	MPS		
		NO	YES	BJF
Job Control Options-STIDJC				
Assembler &SYSPARM support (Other Job Control options affect only the contents of the communications region, not its size.)	SPARM=YES	8	8	32
Physical IOCS Support-PIOCS				
Selector Channel Support	SELCH=NO	-16	-32	-32
Burst Mode on Multiplexor Channel	BMPX=YES	40	48	48
Channel Switching Tape Control				
2404 or 2804 or 3803	CHANSW=RWTAU	112	136	136
2816 only	CHANSW=TSWTCH	112	136	136
Tape Support (Note 10)				
7-track and 9-track, 7-track only, or 9-track only	TAPE=7 } TAPE=9 }	408	408	408
MICR device on selector channel	MRSLCH=YES	32	40	40
	DISK=2311	0	0	0
	DISK=2314	0	0	0
	DISK=3330	624	632	632
Allocate	ALLOC	0	0	0
Input/Output Tables-IOTAB				
Number of I/O devices on system	IODEV=n	8(n-10)	8(n-10)	8(n-10)
Number of programmer logical units	BGPGR=n	2(n-10)	2(n-10)	2(n-10)
	F2PGR=n		2(n-5)	2(n-5)
	F1PGR=n		2(n-5)	2(n-5)
Number of Channel Queue Entries	CHANQ=n (without AP=YES) (Note 6)	7(n-6)	7(n-6)	7(n-6)
	CHANQ=n (with AP=YES) (Note 6)	--	8(n-6)	8(n-6)
Number of Job Information Blocks	JIB=n	4(n-5)	4(n-5)	4(n-5)
Amount of extra space reserverved for the PUB2 Table (required for RMSR recording)	D2311=n1	+24n1	+24n1	+24n1
	D2314=n2	+24n2	+24n2	+24n2
	D2321=n3	+24n3	+24n3	+24n3
	D3330=n4	+32n4	+32n4	+32n4
	D2400=n5	+52n5	+52n5	+52n5
	D3410=n6	+60n6	+60n6	+60n6
	D3420=n7	+60n7	+60n7	+60n7
	IODEV=n	+12B	+12B	+12B
	(B=n-(n1+n2 +n3+n4+n5+n6+n7)			

Figure 1-6. SUPERVISOR--Main Storage Requirements (Bytes) (Part 4 of 5)

Note 1. A 1255/1259/1270 is addressed as a 1419 single address adapter machine i.e., MICR=1419.

Note 2. MPS=YES or BJF is required for TP=QTAMn. TP=QTAMn includes ETAM Supervisor support.

Telecommunications requires a minimum of two channels: one multiplexer channel and at least one selector channel. One channel is required for telecommunications and the other for the system resident device. (Telecommunications should not be on the same selector channel as SYSRES.) If AP=YES when TP=QTAMn, then the quantity A must be added to the basic storage requirement for TP=QTAMn. A=44+(n-2)12, where n is the value elected for TP=QTAMn. (For multitasking, AP must equal YES.)

Note 3. If PTO=YES when PD=n, add 8 bytes to the PD storage requirement. In addition, if any of the following options are elected along with PD, subtract 8 bytes from the PD storage requirement under MPS=NO, or MPS=YES and subtract 24 bytes from the PD storage requirement under MPS=BJF.

- Multitasking (AP=YES)
- Track Hold (TRKHLD=n)
- Abnormal Termination (AB=YES)

The number of bytes indicated is the current storage requirement, where n is a minimum of 800 bytes.

Note 4. When SKSEP=YES, n (in the formula) equals the number of DASD devices specified at system generation time. When SKSEP=n, n (in the formula) is the number of DASD devices supported as specified, but cannot be less than the number specified at system generation time. In either case, a (in the formula) is the 8 bytes required if DASDFP and/or SYSFIL options are selected, and b (in the formula) is the 8 bytes required if Teleprocessing (TP) option is selected.

Note 5. PTO=YES requires that MPS=YES or MPS=BJF for Physical Transient Overlap (PTO) support to be generated. If IDRA=YES and PTO is not specified, then PTO=YES is assumed.

Note 6. The selection of the CBF option results in extra channel queue usage. Consider this when requesting the number of CHANQ entries. Thus, specification of the CBF option and selection of the CHANQ default, which is 6 channel queue entries, results in the number of buffers specified being added to the CHANQ default. However, when both the CBF and CHANQ options are specified, the number of CHANQ entries desired should be increased by the number of buffers specified. Otherwise, the number of entries generated in the channel queue will be less than desired.

Note 7. WAITM=YES is assumed when AP=YES.

Note 8. Where n equals the maximum number of tracks (1-255) to be held at any given time by the entire system. The default is 10 if n is an invalid parameter (non-numeric or outside the range, 1-255).

Note 9. The MODEL=155 main storage requirements include the storage required by ERRLOG and OLTEP. The MODEL=155 storage estimate also includes the storage required by PORT.

Note 10. If EVA is specified and TAPE is not, TAPE=9 is assumed.

Note 11. If s is omitted or if its specification is invalid, then 16 bytes are reserved. If l is omitted or if it is invalid, 0 is assumed.

Note 12. The Timer feature is automatically generated if the interval timer (IT=BG, F1, or F2) and/or Job Accounting Interface (JA=YES, or JA=n1,n2,n3) are supported.

Figure 1-6. SUPERVISOR--Supervisor Main Storage Requirements (Bytes) (Part 5 of 5)

Supervisor size increases are not necessarily linear. For example, compare the size requirements of the separate entries CC=YES and PC=YES with the combined entry OC=YES,PC=YES. Combinations of elements may result in an actual supervisor size that is smaller than the calculated total derived for the same supervisor.

The I/O unit control tables in the IBM-supplied supervisor contain entries for up to 10 physical units and the first 10 programmer logical units. (See the DOS System Control and Service publication for a discussion of these tables.)
The I/O table provides six channel queue positions and five Job Information Blocks (JIBs). As a minimum, 12 I/O devices and program check interrupts can be included within a 14,336 byte supervisor.

A DOS supervisor generated with any of the following options requires a minimum of 14,336 bytes:

- Multiprogramming (See Note)
- Telecommunications
- DASD File Protection
- Disk System Input/Output
- 1255/1259/1419 Magnetic Character Readers or 1270/1275 Optical Reader/Sorters

Note: If multitasking is also specified, the minimum supervisor size becomes 16,384 bytes.

If foreground areas are not used, the batch-job supervisor would probably be more useful because the multiprogramming supervisor requires additional space and time to perform its functions.

Computing the Size of a Supervisor

As an example, assume a supervisor is generated using the macros shown in Figure 1-7. The size of this supervisor is determined as follows:

GENERATION OPERAND		MAIN STORAGE REQUIREMENT (bytes)
SUPVR		
MPS=YES		11.776
CONFG		
MODEL=135		0
DEC=YES		0
FP=YES		104
STDJC		0
FOPT		
IT=BG, PC=YES, OC=YES		536
CCHAIN=YES		32
DASDFP=(1,2,2321), SYSFIL=YES	[832 + 24(2-1)] =	856
TEBV=IR		0
JA=YES		568
PIOCS		
CHANSW=RWTAU		136
TAPE=7		408
BMPX=YES		48
DISK=3330		632
IOTAB		
JIB=10	4(n-5) =	20
CHANQ=10	7(n-6) =	28
F1PGR=8	2(n-5) =	6
F2PGR=8	2(n-5) =	6
IODEV=10	8(n-10) =	0
BGPGR=10	2(n-10) = 0 See Note.	
D3330=2		64
D3410=1		60
D3420=1		60
Total Number of Bytes		15.340

Note: If entry for BGPGR is below minimum requirement, n=10 is assumed. The following MNOTE is obtained for an entry of less than 10.

```
BGPGR SPECIFICATION BELOW MINIMUM - "10" ASSUMED
```

```
// EXEC ASSEMBLY
         SUPVR MPS=YES
         CONFG MODEL=135,DEC=YES,FP=YES
         STDJC LISTX=YES,LINES=46
         FOPT IT=BG,PC=YES,OC=YES,CCHAIN=YES,DASDFP=(1,2,2321),        X
               SYSFIL=YES,TEBV=IR,JA=YES
         PIOCS CHANSW=RWTAU,TAPE=7,BMPX=YES,DISK=3330
         ALLOC F1=14K,F2=14K
         IOTAB JIB=10,CHANQ=10,F1PGR=8,F2PGR=8,ICDEV=10,BGPGR=10,      X
               D3410=1,D3420=1,D3330=2,D2321=1
         DVCGEN CHUN=X'00C',DVCTYP=2540R
         DVCGEN CHUN=X'00D',DVCTYP=2540P
         DVCGEN CHUN=X'00E',DVCTYP=3211
         DVCGEN CHUN=X'01F',DVCTYP=1050A
         DVCGEN CHUN=X'190',DVCTYP=3330
         DVCGEN CHUN=X'191',DVCTYP=3330
         DVCGEN CHUN=X'192',DVCTYP=2321
         DVCGEN CHUN=X'180',DVCTYP=3410T9,CHANSW=YES
         DVCGEN CHUN=X'181',DVCTYP=3420T7,CHANSW=YES
         ASSGN SYSRDR,X'00C'
         ASSGN SYSIPT,X'00C'
         ASSGN SYSPCH,X'00D'
         ASSGN SYSLST,X'00E'
         ASSGN SYSLOG,X'01F'
         ASSGN SYSREC,X'190'
         ASSGN SYSLNK,X'191'
         ASSGN SYS001,X'191'
         ASSGN SYS002,X'191'
         ASSGN SYS003,X'191'
         ASSGN SYS004,X'191'
         ASSGN SYS007,X'192'
         SEND 16384
 END
/*
```

Figure 1-7. SUPERVISOR--Example for Computing the Size of a Supervisor

Planning an Operational Pack

An operational system is one used in day-to-day operations that contain a tailored supervisor and libraries appropriate to each customer's particular combination of system programs and application programs. A system maintenance volume is one used primarily to facilitate changes to programs supplied by IBM. Application programs can also be added to a maintenance volume for ease of program maintenance.

Change Distributions are those changes supplied by IBM to the IBM-shipped volume.

Proper planning is the key to successful system generations. The time spent in planning can save you frustration and valuable time. Planning should reflect the initial, intermediate and ultimate capacity of the core image, relocatable and source statement libraries. Enough workfile capacity must be available throughout system generation for assemblies and linkage-edit steps.

The module for each component should be used to determine its storage requirements. Figure 1-8 contains a summary of the library and partition requirements for all components shipped with the disk operating system. Adjust the requirements you derived from this figure for any program products (separately purchased components) you are including. The IBM-supplied disk operating system contains a 14K supervisor for the 2311, 2314/2319, and 3330 resident systems. This supervisor is described further in Supervisor Nucleus 14K-370N-SV-495.

STORAGE REQUIREMENTS FOR DISK OPERATING SYSTEM IBM-SUPPLIED PROGRAMS

All IBM-supplied programs used with disk operating system that are language translators or utilities execute in the background partition. Figure 1-8 lists the minimum size background partition required. In addition, you can link-edit certain of the language translators and utilities to execute in a batched-job foreground partition. Minimum partition size is the same as that for the background partition. However, note that the background save area is contained within the supervisor, but that either foreground save area is located at the beginning of the partition, thus reducing the partition size by the length of the save area. Therefore, program execution in a foreground partition may require 2K more bytes of storage than would be required for execution in the background partition.

Component	IBM Program Number 370N-	Part. Size	Core Image Library Blocks				Relocatable Library			Source Statement Library	
		bytes	Phase	2311	2314/ 2319	3330	Prefix	Module Blocks		Books	Blocks
System Control:	CL-453	10240							**840		
IPL			3	6	6	8	IJB	1	**		
Job Control		14336	10	29	29	30	IJB	10	**		
Linkage Editor		10240	8	16	16	19	IJB	4	**		
Librarian:		10240									
CORGZ			9	20	20	22	IJB	10	**		
MAINT			10	34	34	37	IJB	18	**		
CSERV			1	5	5	3	IJB	1	**		
DSERV, (Sorted)			6	10	10	10	IJB	2	**		
RSERV			1	4	4	5	IJB	2	**		
SSERV			1	4	4	5	IJB	2	**		
Supervisor:			1	14	14	21					
Macros:											
Communications										26	180
Generation:											
Basic										23	3687
MICR										1	237
TP										7	181
IOCS: See Note.										***2	9
Card (Reader/Punch)							IJC	64	154	2	313
Console										1	73
Device Independent (DTFDI)							IJJ	4	14	2	206
Printers							IJD	23	52	2	196
Serial Device										1	51
PIOCS (DTFPH)										1	55
Imperative Macros										27	306
Transients: (System and Common IOCS)			225	222	223	221					
MCAR/CCH			14	14	14	14					
Dumps:											
Standard System			6*	6	6	6	IJB	6	46		
Translating System			4*	4	4	4	IJB	4	31		
Aids:											
ESTVUT			1	3	3	3	IJB	1	14		
PDAID:		6144	15	18	18	17	IJB	2	38		
Transient			1	1	1	1					
DUMPGEN		10240	2	8	9	9	IJB	1	51		
LSERV		8192	1	5	5	6	IJB	1	18		
3211 Printer Support:		2048	2	2	2	2					
Transients			4	4	4	4					
UCSB Loads							IJB	7	32		
System Generation										5	334

*Transients
**The number of System Control Relocatable Library blocks includes only the IPL, Job Control, Linkage Editor, and Librarian Programs.
*** For BOS and BPS Compatibility.

Note: See Module 4, Compiler IOCS Modules, for the preassembled module names and usage.

Figure 1-8. Disk Operating System Program Requirements Summary Libraries (Part 1 of 6)

Component	IBM Program Number	Part. Size	Core Image Library Blocks				Relocatable Library			Source Statement Library	
		bytes	Phase	2311	2314/ 2319	3330	Prefix	Module Blocks		Books	Blocks
American National Standard COBOL:	360N-CB-482	55296	13	220	223	252	ILA	16	1584	1	24
Transients			3	2	2	2					
Assembler D :	370N-AS-465	14336	32	85	88	102	IJQ	26	567	6**	36
Assembler F :	360N-AS-466	45056	8	70	71	82	IJY	34	476	6**	36
COBOL:	360N-CB-452	14336	47	172	174	188	IJS	51	1327	1	25
Subroutines							IHD	42	162		
Transients			5	5	5	5					
COBOL Debug			1	4	4	5	IJS	2	23		
COBOL & PL/I (D) DASD	360N-CB-468									8	270
COBOL LCP	360N-CV-489	18432	25	62	62	70	IKL	33	461	1	38
Compiler IOCS Modules	370N-IO-476		Note: This program consists of a group of various pre-assembled IOCS modules used by American National Standard COBOL, PL/I (D), COBOL and RPG. The digits on this figure represent the number of modules used and the number of library books required by the various combinations of compilers desired.					222	1247		
ANS* COBOL								132	864		
ANS* COBOL & COBOL								158	1053		
ANS* COBOL & PL/I (D)								163	970		
ANS* COBOL & RPG								159	1032		
ANS* COBOL,COBOL,&PL/I(D)								181	1134		
ANS* COBOL, COBOL, & RPG								171	1081		
ANS* COBOL, RPG & PL/I(D)								186	1127		
ANS* COBOL,COBOL,PL/I (D) and RPG								194	1161		
COBOL								77	548		
COBOL & PL/I (D)								107	662		
COBOL & RPG								93	595		
COBOL,PL/I (D) & RPG								116	675		
PL/I (D)								70	400		
PL/I (D) & RPG								98	612		
RPG								43	298		
Required IOCS Modules								27	84		

*American National Standard
**These books are the sample problems that are common to both assemblers.

Figure 1-8. Disk Operating System Program Requirements Summary Libraries (Part 2 of 6)

Component	IBM Program Number	Part. Size	Core Image Library Blocks				Relocatable Library			Source Statement Library	
		bytes	Phase	2311	2314/ 2319	3330	Prefix	Module Blocks		Books	Blocks
FORTRAN IV (Basic):	360N-	10240	4	20	20	22	IJT	5	143	1	5
Subroutines	FO-451						IJT	37	149		
FORTRAN IV	360N- FO-479	40960	8	43	43	30	ILF	9	316	2	15
FORTRAN IV Subroutines	360N- LM-480						ILF	61	286		
PL/I (D):	360N-		110	278	282	304	IJX	115	2335	2	17
10K Variant	PL-464	10240					IJX	114	2297		
12K Variant		12288					IJX	115	2294		
Subroutines							IJK	95	363		
Transients			1	1	1	1					
RPG	360N- RG-460	10240	28	98	101	111	IJR	60	785	1	13
1401/1440/1460 Emulator	370N- EU-490	19180 Note 1	19		20	20	IIQ	27	235	30	4556
1410/7010 Emulator	370N- EU-490	28672 Note 1	18	19	19	19	IIR	25	238	29	2639
IOCS: (Note 2)	370N-										
Direct Access (Disk)	IO-454						IJI	24	187	3	883
Index Sequential Disk	IO-457						IJH	34	555	19	1772
MICR	IO-477		5*	5	5	5	IJU			7	479
Magnetic Tape (2400/3400)	IO-456						IJF	15	68	2	794
Optical Reader	IO-478		2*	2	2	2	IJM			7	609
Paper Tape	IO-458		4*	4	4	4	IJE			2	459
Sequential Disk	IO-455						IJG	32	162	12	3735

*Transients unique to that component.

Note 1: This is the minimum partition size for the smallest 1410/7010 program being emulated. For more information about storage requirements, refer to the emulator publication listed in the Preface.

Note 2: See Compiler IOCS Modules for the preassembled module names and usage.

Figure 1-8. Disk Operating System Program Requirements Summary Libraries (Part 3 of 6)

Component	IBM Program Number	Part, Size bytes	Core Image Library Blocks Phase	2311	2314/ 2319	3330	Relocatable Library Prefix	Module Blocks		Source Statement Library Books	Blocks
OLTEP	370N-		54	63	64	64	IJZ	58	429		
Transients	DN-481		4	4	4	4					
Configuration Data Sets (CDSs)			Note 1								
On-Line Tests (OLTs)			Note 2								
Sorts:	360N-										
Disk Sort/Merge	SM-450	10260	26	74	76	83	IJO	38	521	1	2
Tape and Disk Sort Merge:	360N- SM-483		40	69	71	77	ILH	101	987	3	6
Tape or 2311 Variant		10240					ILH				
2314/2319 Variant		22528					ILH				
Tape Sort Merge	360N- SM-400	10240	20	36	36	38	IJP	5	206	1	2
Supervisor (14K)	370N- SV-495	14336	1	18	19	21					
Teleprocessing (BTAM):	370N-		52	52	52	52	IJL	64	158	46	5029
Transients	CQ-469		29	29	29	29					
Teleprocessing (QTAM):	370N-						IJL	109	392	95	1899
Transients	CQ-470		22	22	22	22					

Note 1: This is the formula to calculate the number of core image library blocks necessary for the configuration data sets:

$$\frac{\text{number of devices on system} + 1}{6} = \text{number of blocks}$$

Note 2: The number of core image library blocks required for the On-Line Tests (OLTs) depends on the number of device types on the system and the number and size of the tests needed to test these devices. This information is in the write up that accompanies the tests.

Figure 1-8. Disk Operating System Program Requirements Summary Libraries (Part 4 of 6)

Component	IBM Program Number	Part. Size bytes	Core Image Library Blocks Phase	2311	2314/ 2319	3330	Relocatable Library Prefix	Module	Blocks	Source Statement Library Books	Blocks
Utilities:											
MFS Utility	360N-UT-471		3*	3	3					10	1187
Utilities Group 1: **	360N-UT-461	10240	44	122	125		IJW		657		
Assignment Disk			5	26	27		IJW	6			
Card to Disk			5	10	10		IJW	4			
Card to Printer and/or Punch			5	12	12		IJW	4			
Clear Disk			3	4	4		IJW	2			
Copy Disk to Card			2	7	7		IJW	3			
Copy Disk to Disk			2	7	8		IJW	3			
Disk to Card			5	11	11		IJW	4			
Disk to Disk			5	11	11		IJW	2			
Disk to Printer			5	12	12		IJW	2		3	3
Initialize Disk			4	14	15		IJW	5			
Restore Card to Disk			1	4	4		IJW	2			
VTOC Display:			1	3	3		IJW	2			
Transient			1	1	1						
Utilities Group 2:**	360N-UT-462	10240	47	103	104		IJW		453		
Card to Tape			5	10	10		IJW	4			
Copy Disk or Data Cell to Tape			2	7	7		IJW	3			
Data Cell to Tape			5	10	10		IJW	2			
Disk to Tape			5	10	10		IJW	2			
Initialize Tape			1	3	3		IJW	1			
Restore Tape to Disk or Data Cell			1	4	4		IJW	2			
Tape Compare			3	5	5		IJW	5			
Tape to Card			5	11	11		IJW	4			
Tape to Data Cell			5	11	11		IJW	2			
Tape to Disk			5	11	11		IJW	2			
Tape to Printer			5	11	12		IJW	4		2	2
Tape to Tape			5	10	10		IJW	4			
Utilities Group 3:**	360N-UT-463	10240	32	74	76		IJW		322		
Alternate Track Assignment Data Cell			5	17	17		IJW	6			
Clear Data Cell			3	4	4		IJW	2			
Data Cell to Data Cell			5	11	11		IJW	2			
Data Cell to Disk			5	11	11		IJW	2			
Data Cell to Printer			5	12	12		IJW	2		1	1
Disk to Data Cell			5	11	11		IJW	2			
Initialize Data Cell			4	8	10		IJW	5			

*Transients

** The number of relocatable library blocks for Group 1 includes the blocks required by common modules. The number of relocatable library blocks for Groups 2 and 3 represent the unique modules for that group. The common modules are discussed in their respective sections.

Figure 1-8. Disk Operating System Program Requirements Summary Libraries (Part 5 of 6)

Component	IBM Program Number	Part. Size bytes	Core Image Library Blocks				Relocatable Library			Source Statement Library	
			Phase	2311	2314/ 2319	3330	Prefix	Module	Blocks	Books	Blocks
System Utility Programs:	370N-UT-491	10240	35	113	118	126	IJW		615		
Assign Alternate Track Disk			5	26	27	29	IJW	6			
Assign Alternate Track Data Cell			5	17	17	19	IJW	6			
Clear Disk			3	5	5	5	IJW	4			
Clear Data Cell			3	5	5	5	IJW	4			
Copy Disk to Card			2	7	7	7	IJW	3			
Copy Disk to Disk			2	7	8	8	IJW	3			
Copy Disk or Data Cell to Tape			2	7	7	7	IJW	3			
Initialize Disk			4	16	16	17	IJW	5			
Initialize Data Cell			4	8	10	11	IJW	5			
Initialize Tape			1	3	3	3	IJW	2			
Restore Card to Disk			1	4	4	5	IJW	2			
Restore Tape to Disk or Data Cell			1	4	5	5	IJW	2			
VTOC Display			1	3	3	4	IJW	4			
Transients			1	1	1	1					
EREP	370N-UT-492	10240	68	98	59	46	IJB	35	503		

Figure 1-8. Disk Operating System Program Requirements Summary Libraries (Part 6 of 6)

Planning the Libraries

Two types of IBM libraries are:

1. System libraries

2. Private libraries

SYSTEM LIBRARIES

The system libraries are the core image, the relocatable, and the source statement. The private libraries are the private core image, the private relocatable, and the private source statement library. Private core image library support is a system generation option.

Core Image Library

Because the core image library contains the executable format of programs, it is the library in which you are most likely to keep your programs. Otherwise, the programs must continually be placed in the core image library before each execution (linkage edited). Therefore, during system generation, expand the size of the core image library to accommodate all the programs desired resident and on-line, both your programs and IBM's.

In addition, try to envision future space requirements, and provide this space if possible. Such planning can eliminate the need for another system generation. Thus, to expand the core image library means making the remaining libraries smaller on the pack. See Private Libraries for alternative considerations.

Before the size of a library is reduced, delete those items that are not to be used, or those items that were transferred to another library in the required format. It is recommended that backup of the system (a copy of the IBM-supplied system) be obtained for your protection in case something is accidentally removed and desired from the system. It is also recommended that after successful completion of segments of system generation, a copy (backup) of the partially generated system be obtained, i.e., upon the creation of a library, or the assembly of a supervisor. This permits you to return to a point other than the beginning of the procedure in case of an error. The components supplied in the IBM-shipped core image library facilitate system generation.

CORE IMAGE LIBRARY PHASES: All program phase names in the core image library(s) are composed of two four-character parts. The first four characters uniquely identify the program. The next four characters identify the phase of the program. The first phase of a program to be executed from the system core image library (cataloged on SYSRES) or a private core image library must be identical to the name specified in the // EXEC control statement. For RPG, the first four characters of the phase names are RPG1, although the processor is invoked by

```
// EXEC RPG
```

All IBM-supplied phase names begin with an alphabetic character (A-Z). Three classes of programs are exceptions: transients, job control and linkage routines, and IPL and supervisor program names.

Relocatable Library

All IBM-supplied components are shipped in the relocatable library. This library is the basis for the creation of a private relocatable library. Thus, it is from this library that most IBM components are directly or indirectly extracted (indirectly in the case of the existence of a private relocatable library and a system relocatable library).

RELOCATABLE LIBRARY COMPONENT NAMING CONVENTIONS: All DOS relocatable modules, including compiler subroutines, are assigned unique three-character prefixes. This convention facilitates handling of these modules when deleting, displaying, punching, copying, or merging modules either individually or by component. Figure 1-8 has a list of prefixes for IBM-supplied components.

COMPONENTS SHIPPED WITH TRANSIENTS: The following components are shipped in the relocatable library with transients. When cataloged to the core image library, the transients are automatically cataloged with the component phases.

Component	Number of Transients
OLTEP	3
PL/I (D)	1
VTOC Display	1

Source Statement Library

All IBM-supplied macro definitions are in this library. This library is the basis for the creation of a private source statement library. Thus, it is from this library that you may extract, directly or indirectly, IBM-supplied component macro definitions (indirectly in the case of the existence of both a private source statement library, and a system source statement library).

All macros reside in the A. sublibrary. All sample problems reside in the Z. sublibrary. LINKEDIT, DELETECL, DELETERL, and DELETESL also reside in the Z. sublibrary and contain the necessary link-edit and control statements, respectively, to linkage edit or delete selected IBM components. Also, the ILFMERG book resides in the Z. sublibrary.

LINKAGE EDITING OR DELETING IBM COMPONENTS: Among the macro definitions in the source statement library are four books that are used when generating a new system or altering an existing operational system: LINKEDIT, DELETECL, DELETERL, and DELETESL. These books contain the necessary control statements to selectively linkage edit and delete all IBM components.

The linkage edit and delete statements for each IBM component are also listed in their respective sections.

PRIVATE LIBRARIES

If more than one disk drive is available and the system supports private core image libraries, you may reduce the size of the system core image library by placing some programs in a private core image library or libraries. You should place phases beginning with a $ in the system core image library. You must place phases beginning with $$A, $$B, or $$R in the system core image library. When you request a phase, the system searches both the system and private core image directories, if necessary and applicable:

- If the requested phase is a $$A, or $$B phase (transient), the system first searches the system core image directory. If the phase is not found, the private core image directory assigned to that partition is then searched. If the phase is not found, the system enters the wait state with an error message of X'04E6' in bytes 0 and 1 of low main storage. A supervisor cataloged to a private core image library can never be used, because the IPL retrieval program searches the system core image directory only.

- For other phases starting with $, if the phase is not found in the system core image library, the private core image library assigned to that partition is then searched.

- If the requested phase does not begin with $, the private core image library assigned to the partition is searched first. If it is not found, the system core image library is then searched.

If the system is to support both the batched-job foreground and private core image library options, consider which IBM-supplied programs you want placed in a private core image library or libraries. Under such a system, the linkage editor executes in any partition. You can link-edit most IBM-supplied programs for execution in a batched-job foreground partition (if enough core storage is available in which to execute the link-edited program) and place them in a private core image library assigned to the partition.

In a nonbatched-job foreground system (MPS=NO or YES), private core image library support is available for the background partition only.

If more than one disk drive is available, it is not necessary to decrease the size of the relocatable and source statement libraries. They can be assigned to other disks and are then referred to as private libraries. Systems can be built with private and system libraries containing those items that best fit your needs.

LIBRARY SIZES

Choose the desired libraries, and then plan their precise content and size for daily use. Thus, you should know the initial, intermediate, and final sizes of the libraries throughout system generation, and plan the exact contents of each library that is created during system generation. These contents should be listed, along with their sizes, and then the total number of cylinders to be allocated can be calculated.

The contents of the libraries are identified in Attachment 1 of the Memorandum To Users that accompanies the IBM system that is shipped. The storage requirements (sizes) for these components and macro definitions are identified in the section for each component.

An alternate method for determining the number of cylinders to allocate for a library is given in the discussion Allocating Library Sizes. Note that once private library sizes have been allocated, they cannot be reallocated. Although excess room in the libraries may not offer maximum efficiency, in case of a calculation error, the benefits can be easily recognized when sufficient space is available.

One 2311 Disk Drive

When planning an operational system, decide upon the ultimate appearance of your libraries. It is most convenient to build your operational system on the IBM volume that contains the core image and relocatable libraries. Single disk drive users may want to build operational systems appearing as one of the following:

- Core image library, small system relocatable library, and/or small system source statement library

The small system relocatable library is for those who need the compiler subroutines and/or system IOCS modules on line at all times.

The small system source statement library can contain system control and logical IOCS macros. The system with the source statement library, supplied by IBM, can be used as an assembly pack.

Two or more 2311 Disk Drives, a 2314/2319 Direct Access Storage Facility, or a 3330 Disk Storage

Note: You may use the IBM 2319 Disk Storage Facility as the system residence device if you so desire.

For convenience, build your operational pack upon the IBM volume that contains the core image and source statement libraries. Multiple disk drive users may want to build an operational system appearing as one of the following:

- System core image library, private relocatable library, and private source statement library.
- System core image library, private relocatable library, system source statement library, and private source statement library.
- System core image library, system relocatable library, private relocatable library, and private source statement library.
- System core image library, system relocatable and system source statement libraries, and private relocatable and source statement libraries.
- System core image library, system relocatable library and system source statement librarary.
- System core image library, system source statement library, small private relocatable library (on the operational pack) and private relocatable library.
- System core image library, private core image library, private source statement library, system relocatable library.
- System core image library, system relocatable library, system source statement library, private relocatable library, private core image library.

IBM system control and system service programs are supplied in the system core image libraries of all volumes for all systems. All volumes (with the exception of volume 1 for the 2311 system) have the 14K Assembler D in the core image library.

If the assembled supervisor does not exceed the size of the IBM supervisor, relinkage editing and recataloging of the IBM-supplied

programs shipped in the core image library are unnecessary. Job control, linkage editor, MAINT, DSERV, and CSERV are self-relocating and need never be relinkage edited. Only the steps required to retrieve the sample problems, delete unwanted components, assign standard labels, assemble another supervisor, allocate for and linkage edit IBM components, and condense libraries are necessary to perform system generation.

When the number of tracks required for each library has been calculated, allocate a sufficient number of cylinders to each library of each operational system. Additional cylinders may be allocated to the core image library for application programs. Sometimes you must reallocate the libraries on your disk pack(s) during system generation to ensure sufficient workfile storage for assemblies and linkage-edit steps. For the library reallocation function, see DOS System Control and Service, GC24-5036, and DOS Version 4, GC33-5007.

It is usually advantageous for users with more than one available disk drive to define private core image, relocatable, and source statement libraries for an operational system during system generation.

Organization of a DOS System Pack

The organization of the system pack is as follows:

Name	Start Location, if Present
IPL Program	Track 0 of Cylinder 0.
System Volume Label	Track 0 of Cylinder 0.
System Directory	Track 1 of Cylinder 0.
Librarian Work Area	Tracks 2, 3, and 4 of Cylinder 0.
Transient Directory	Track 5 of Cylinder 0. (See Note)
Open Routine Directory	Track 6 of Cylinder 0.
Library Routine Directory	Track 7 of Cylinder 0.
Foreground Program Directory	Track 8 of Cylinder 0.
Problem Program Phase Directory	Track 9 of Cylinder 0.
Core Image Directory	Track 0 of Cylinder 1, on a 2311; Track 10 of cylinder 0 on a 2314/2319, or 3330.
Core Image Library	Beginning of the first available track following the core image directory.
Relocatable Directory, Optional	Track 0 of the first available cylinder following the core image library.
Relocatable Library, Optional	Beginning of the first available track following the relocatable directory.
Source Statement Directory, Optional	Track 0 of the first available cylinder following the previous library.
Source Statement Library, Optional	Beginning of the first available track following the source statement directory.

Label Cylinder

First full cylinder after the last system library.

Track	Provides Storage For
0	Background User Labels
1	Background Partition Standard (PARSTD) Labels
2	Foreground 2 User Labels
3	Foreground 2 Partition Standard (PARSTD) Labels
4	Foreground 1 User Labels
5	Foreground 1 Partition Standard (PARSTD) Labels
6 - End of cylinder	Standard labels

Volume Table of Contents

Location assigned by the user.

Note: The IPL procedure cannot be performed if either:

- the Transient Directory is located on a defective track while an alternate track has been assigned, or
- if part of the Supervisor Nucleus in the core image library is located on an alternate track.

This is caused by the fact that the bootstrap program does not contain any error recovery routines. Restrictions that had to be imposed on the size of this program did not allow the inclusion of any such routines.

Allocating Library Sizes

When the size of an existing library is reduced, it may be important to know the minimum size library that can be allocated. Once the minimum size library is calculated, it should be increased to accommodate any problem programs to be included in the library. The size of IBM components can be determined by referring to the storage requirements given in Figure 1-8 and to appropriate documentation for any program products to be included.

The following explanation illustrates how to calculate the number of tracks required for a core image, relocatable, or source statement library. The formula for computing the size of a library is:

Library (size in tracks) = LBA/nn

where: library = either core image, relocatable or source statement

LBA = the number of LIBRARY BLOCKS ACTIVE for the library of interest and is obtained from a DIRECTORY (SYSTEM or PRIVATE)

nn = LIBRARY LAST AVAILABLE ENTRY in the R (record) column

Using the sample PRIVATE DIRECTORY that follows, an example of how to compute a library size is:

Relocatable Library = LBA/nn

where: LBA = 9849
nn = 9 , thus
Relocatable Library = 9849/9 = 1,094.3 tracks

The relocatable library size computed does not include the tracks allocated for the directory. The directory size must be added to the relocatable library size computed. Thus,

Relocatable Library Allocation = Relocatable Library + Directory Allocated Tracks

Therefore,

Relocatable Library Allocation = 1,094.3+9
= 1,103.3 Tracks

For a 2311, cylinders = $\frac{\text{Library Size (Tracks)}}{10}$

For a 2314, cylinders = $\frac{\text{Library Size (Tracks)}}{20}$

For a 3330, cylinders = $\frac{\text{Library Size (Tracks)}}{19}$

For this example,

the Relocatable Library Allocation for a 2311 = $\frac{1103.3 \text{ Tracks}}{10}$
= 110.3 or 111 cylinders, rounded high

```
PRIVATE DIRECTORY                PRIVATE-RELOCATABLE

02/01/69                         ------DECIMAL------

                                      C  H  R  E
DIRECTORY STARTING ADDRESS           01 00 01
DIRECTORY NEXT ENTRY                 01 05 01 08
DIRECTORY LAST ENTRY                 01 08 09 19

LIBRARY   STARTING ADDRESS           01 09  01
LIBRARY   NEXT AVAILABLE ENTRY      111 03  04
LIBRARY   LAST AVAILABLE ENTRY      127 09  09


                                 STATUS INFORMATION
DIRECTORY ENTRIES ACTIVE            903

LIBRARY   BLOCKS ALLOCATED        11349
LIBRARY   BLOCKS ACTIVE            9849
LIBRARY   BLOCKS DELETED             00
LIBRARY   BLOCKS AVAILABLE         1500

AUTOMATIC CONDENSE LIMIT             00

LIBRARY   ALLOCATED CYLINDERS       127
DIRECTORY ALLOCATED TRACKS           09
```

SYSTEM DIRECTORY AND LIBRARY TRACK CAPACITIES

Figures 1-9 and 1-10 illustrate the DOS system library directory and track capacities.

	Directory Entries Per Track		
Device	Core Image Library (Phases)	Relocatable Library (Modules)	Source Statement Library (Books)
2311	144	180	160
2314/ 2319	270	340	270
3330	468	560	440

Figure 1-9. LIBRARIES--System Library Directory Capacities (Entries per Track)

	Core Image Library			Relocatable Library			Source Statement Library		
Device	2311	2314/ 2319	3330	2311	2314/ 2319	3330	2311	2314/ 2319	3330
Library Block Size(bytes)	1728	1688	1504	322	322	322	160	160	160
Blocks per Track	2	4	8	9	16	28	16	27	44

Figure 1-10. LIBRARIES--System Library Track Capacities

Optimum Assignment of Workfiles

Figure 1-11 indicates the optimum assignments of the symbolic units used as workfiles when assembling and linkage editing or compiling and linkage editing user programs. While SYSRES and SYSLNK must be assigned to disk units, SYSnnn can be assigned to either tape or disk units. Where split cylinders are recommended, Figure 1-11 gives the division of the tracks in each cylinder between the symbolic units.

For Figure 1-11, part 1, workfile assignments are as follows: a cylinder represents a 2311 disk drive. Thus two cylinders, one over the other, represent a 2311 two-disk drive system.

Within each disk drive, system workfile assignments are represented two ways: by horizontal lines and by vertical lines.

Horizontal lines denote whole cylinder allocations. Vertical lines denote split cylinder allocations. In either case, the number of cylinders to allocate depends upon the size of the assembly or compilation.

Whenever split cylinder assignments are recommended, the integer notations (0 - 4, etc.) within the disk drive indicate the number of disk tracks per cylinder assigned for the specified symbolic unit. Thus, Figure 1-11, part 1, shows that for a 2311 two-disk system, disk workfile assignments for Assembler (14K) are:

Disk drive number 1 (SYSRES)

SYSLNK
SYS003

Disk drive number 2

SYS001, tracks 0 - 4 of the cylinders allocated
SYS002, tracks 5 - 9 of the cylinders allocated

Interpreting the information collectively reveals that SYSLNK and SYS003 are assigned separate cylinder groups on disk 1, and SYS001 and SYS002 share the same cylinder group on disk 2. Thus, the cylinder group on disk 2 is split so that SYS001 and SYS002 each occupy 5 tracks of every cylinder in the group. To define the extents for workfiles, refer to the IBM publications DOS System Control and Service and DOS Version 4.

For Figure 1-11, Part 2 of 2:

- Tape speeds are given by an integer s where:

  s = the relative speed of the tape unit (the number 1 designates the fastest tape unit, the number 2 designates the next fastest, etc)

- Tape channels are given by an integer c where:

  c = the channel that the tape should be on.

- Where applicable, the assignments that give the best overall performance are indicated.

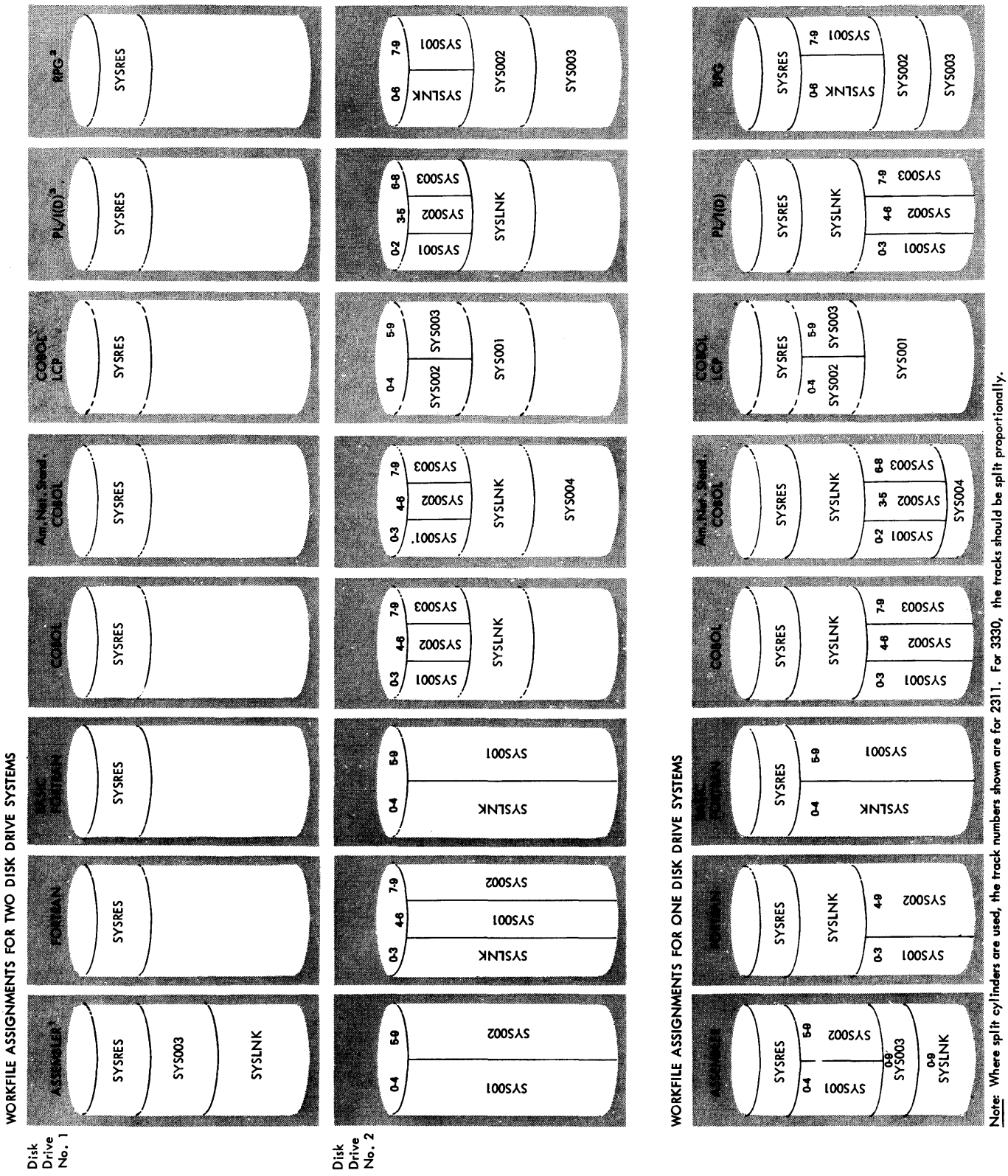


Figure 1-11. Optimum Workfile Assignments for Assembling and Linkage Editing or Compiling and Linkage Editing (Part 1 of 2)

Workfile Assignments for One Disk Drive with Three Tape Units

Symbolic Unit Requirements[5]		Assembler D[1]	FORTRAN	Basic[2] FORTRAN	COBOL	PL/I (D)	RPG	ANS* COBOL
	Tape[4] Speed (s)	3	1	Any	2	2	1	1
SYS001	Channel Requirement (c)	2	1	Any	2	2	2	2
	Tape[4] Speed (s)	2	2	Not Req'd	1	1	2	1
SYS002	Channel Requirement (c)	1	1	--	1	1	1	1
	Tape[4] Speed (s)	1	Not Req'd	Not Req'd	1	1	2	1
SYS003	Channel Requirement (c)	2	--	--	2	2	2	2
	Tape Speed (s)	Not Req'd	Not Req'd	Not Req'd	Not Req'd	Not Req'd	Not Req'd	3
SYS004	Channel Requirement (c)	--	--	--	--	--	--	2

*American National Standard

Note 1: The configuration that yields the best overall performance depends upon the storage available. For cases with over approximately 500 lines of output, the two disk drive configuration yields the best overall performance. For cases with under approximately 500 lines of output, the three tape drive configuration yields the best overall performance.

Note 2: Indicates best overall performance.

Note 3: PL/I (D) users with two disk drives, compilation times will be faster if the following assignments are made for SYS001, SYS002 and SYS003 in place of those given.

```
Assign SYS001 to drive 2, tracks 0-4 of the cylinders allocated  }
Assign SYS003 to drive 2, tracks 5-9 of the cylinders allocated  }  For 2311
Assign SYS002 to drive 1, tracks 0-9 of the cylinders allocated  }
```

The assignments for SYSRES and SYSLNK are the same.

Note 4: Relative tape speeds are indicated by 1, 2 or 3. 1 designates the fastest tape unit, 2 the next fastest, etc.

Note 5: SYSRES and SYSLNK must always be assigned to disk in a one disk drive, three tape drive configuration.

Figure 1-11. Optimum Workfile Assignments for Assembling and Linkage Editing or Compiling and Linkage Editing (Part 2 of 2)

Recorder File (SYSREC)

IJSYSRC is the filename of the recorder file that is used exclusively for output from the Recovery Management Support Recorder (RMSR) function. The data contained on IJSYSRC is edited and printed by the EREP program. The recorder file must be created after the first IPL procedure has been performed following supervisor replacement but before the first JOB card is read; it is defined by using the file definition statements of the system.

The RMSR function makes several types of recordings on the recorder file, in chronological order. The file is made up of records that contain information relating to:

- MCAR
- CCH
- Unit Check
- Counter Overflow
- Tape Volume Statistics
- IPL/EOD
- Miscellaneous Data Recorder
  (2715 error records, 3211 buffer error records,
  3330 non unit-check records)

The IJSYSRC file is defined as a disk extent for an IBM 2311, 2314/2319, or 3330 disk device, and cannot be a split cylinder file. The file should not be defined on an extent which includes a defective or alternate track, because data may be lost. The file definition must be included in the standard label area.

Creating and using the Recorder File

A minimum of ten tracks is required for the recorder file (SYSREC). The following file definition statements create the recorder file:

```
// OPTION STDLABEL
// DLBL IJSYSRC,'DOS RECORDER FILE'
// EXTENT SYSREC,,,,nnnnn,nnnnn
   ASSGN SYSREC,X'cuu'
   SET RF=CREATE
     .
     .
     .
// JOB FIRST
     .
     .
     .
     .
```

Note: The cards must be included in the standard label deck.

The recorder file is created when the first job statement (// JOB FIRST) is read.

These file definition statements must immediately follow the IPL procedure and precede the first job. We recommend that the file definition be permanently retained on the standard label track of the label cylinder Thus, once the file is created, recording can proceed at the beginning of each day without operator intervention; that is, at IPL time the recorder file is opened and updating continues. When the system is to be shut down at the end of the day, issue the Record On Demand (ROD) command to record

the statistical data and to ensure that no statistical data is lost. With RDE support, an EOD (End of Day) record is also written.

Note: The ROD command is not valid for recording teleprocessing statistical data. Refer to DOS BTAM, GC30-5001, and CTAM Message Control Program, GC30-5004, for teleprocessing procedures.

System Generation and Maintenance Procedures

Many techniques exist for generating and maintaining operational volumes. Each installation uses techniques dependent on its machine configuration and its selection of system and application programs. The techniques described here correspond to the following principal machine configurations relevant to system generation and maintenance:

1. At least two 2311 disk drives.
2. One 2311 disk drive.
3. One 2314/2319 direct access storage facility.
4. One 3330 disk storage.

The four examples given do not show the coding necessary to linkage edit and delete all IBM-supplied components. They are meant only to be examples and must be tailored to meet your needs. Additional control statements required for linkage editing and deleting any IBM-supplied components are given in their respective sections.

Disk Operating System Distributions

Figures 1-12 through 1-14 illustrate the various distributions of the IBM Disk Operating System. The procedures within this section (System Generation and Maintenance Procedures) will guide you through your system generation.

Note: The DLBL and EXTENT information for the 2311, 2314/2319, and 3330 distributions is found in Section 4 of the Memorandum to Users of the IBM Disk Operating System.

ALL IBM 2311 USERS

The IBM 2311 Disk Operating system is supplied on three volumes:

- Volume 1 contains a core image library and a system relocatable library.
- Volume 2 contains a core image library and a system source statement library.
- Volume 3, is optional depending on the components you ordered, contains a core image library, a private source statement library, and a private relocatable library.

Volume 3 Users

This SYSRES file (Volume 3) contains a core image library with the following:

- System Control and Service Programs (370N-CL-453)
- Assembler D 14K Disk and Tape Workfile (370N-AS-465)

This core image library is required only by single disk-drive users. Multiple disk-drive users may wish to use the MERGE function of the CORGZ program to access the private libraries on Volume 3. Use either the core image library of Volume 3 or one of another SYSRES to access the SYSSLB and SYSRLB files of Volume 3.

IBM 2311 Disk-Only Users

IBM 2311 disk-only users receive the volumes on 1316 disk packs. Volume 3 is optional and may not be required for your particular installation. These volumes are ready for system generation, but it is recommended that they be copied and retained for backup.

IBM 2311 Disk and 9-Track Tape Users

The 9-track tape reel for the 2311 system contains either Volumes 1 and 2 or all three Volumes. Volumes 1 and 2 are necessary for system generation. Volume 3 may not be necessary for your particular installation. See All IBM 2311 Tape Users. Retain this tape for backup.

IBM 2311 Disk and 7-Track Tape Users

The 2311 system for 7-track tape is supplied on one or two reels of magnetic tape. The first reel contains Volumes 1 and 2 and is shipped to all users. The second reel of tape contains Volume 3 and is shipped only if one of the components in either of the private libraries has been ordered. Retain these tapes for backup.

All IBM 2311 and Tape Users

Restore one or both of the Volume 3 private libraries (files) based on the components you desire. See either Section Four of the Memorandum to Users of IBM Disk Operating System or Attachment One of the Program Material List for IBM Disk Operating System. Section four and attachment one are identical in content and identify the various components of these libraries (files). See Restoring the IBM-Supplied System Tape to Disk for the procedure to restore the tapes to disk.

ALL IBM 2314 USERS

The IBM 2314 Disk Operating System is supplied on one SYSRES file. This file contains the core image library, relocatable library, and the source statement library.

IBM 2314-Only (and 2319-Only) Users

IBM 2314-only users receive the 2314 system on one 2316 disk pack. This pack is ready for your system generation procedure, but it is recommended that this pack be copied and retained for backup.

IBM 2314 and 9-Track Tape Users

The 2314 system is shipped on one reel of magnetic tape. This tape contains the entire SYSRES file for the 2316 disk pack. See Restoring the IBM-Supplied System Tape to Disk for the procedure to restore this tape. After the tape has been restored, retain it for backup.

IBM 2314 and 7-Track Tape Users

The 2314 system is shipped on two reels of magnetic tape. The first reel of tape contains the first half of the SYSRES file; the second reel contains the second half of the SYSRES file. Both 7-track tape reels are required to restore the entire 2314 system to a single 2316 disk pack. See Restoring the IBM-Supplied System Tape to Disk for the procedure to restore these tapes. Retain these tapes for backup after they have been restored to disk.

IBM 3330 AND 9-TRACK TAPE USERS

The 3330 system is shipped on one reel of magnetic tape. This tape contains the entire SYSRES file for the 3336 disk pack. See Restoring the IBM-Supplied System Tape to Disk for the procedure to restore this tape. After the tape has been restored, retain it for backup.

IBM 3330 AND 7-TRACK TAPE USERS

The first 3330 system is shipped on two reels of magnetic tape. The first reel of tape contains the first half of the SYSRES file; the second reel contains the second half of the SYSRES file. Both 7-track tape reels are required to restore the entire 3330 system to a single 3336 disk pack. See Restoring the IBM-Supplied System Tape to Disk for the procedure to restore these tapes. Retain the tapes after they have been restored to disk.

General System Generation Procedures

Each system generation job begins with a new IPL procedure, including the necessary ADD, SET, and ASSGN statements. Typically, each job consists of many job steps, including such librarian programs as:

- CSERV (core image library service) to punch out (or write on magnetic tape or disk) programs from the core image library.
- SSERV (source statement library service) to punch out (or write on magnetic tape or disk) macro definitions.
- RSERV (relocatable library service) to punch out (or write on magnetic tape or disk) the relocatable modules used to build IBM-supplied processor programs.
- DSERV (directory service) to display on SYSLST the current contents of one or more library directories and their remaining library capacities. The directory display may be either an alphabetically sorted listing or a listing of the entries in the order they appear in the directory.
- MAINT (library maintenance) to delete and/or catalog library elements, and also to condense and reallocate library extents.
- CORGZ (copy or merge) to selectively copy or merge library entries from one disk pack to another disk pack. CORGZ allows larger or smaller allocations for each library of the new pack.

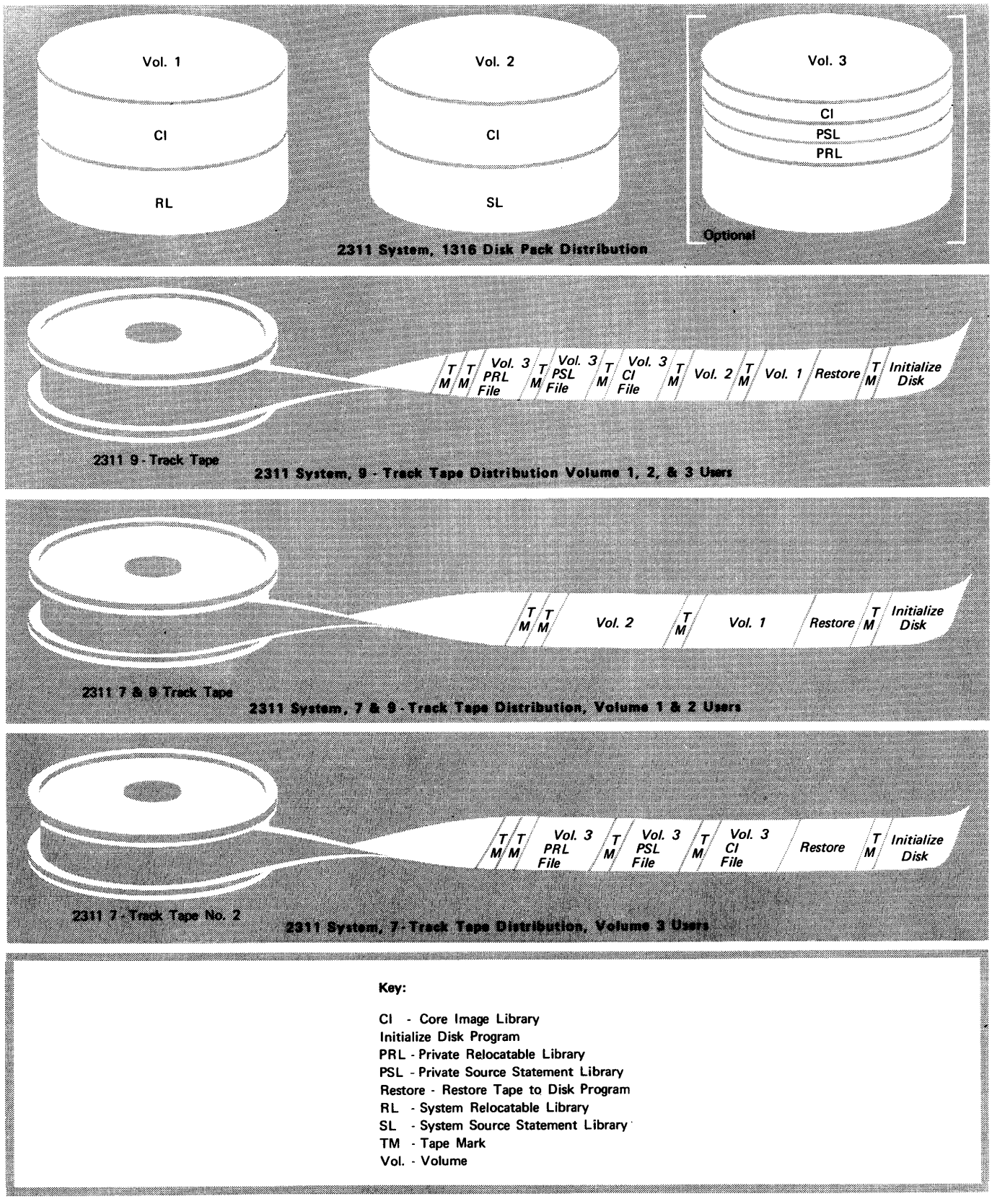
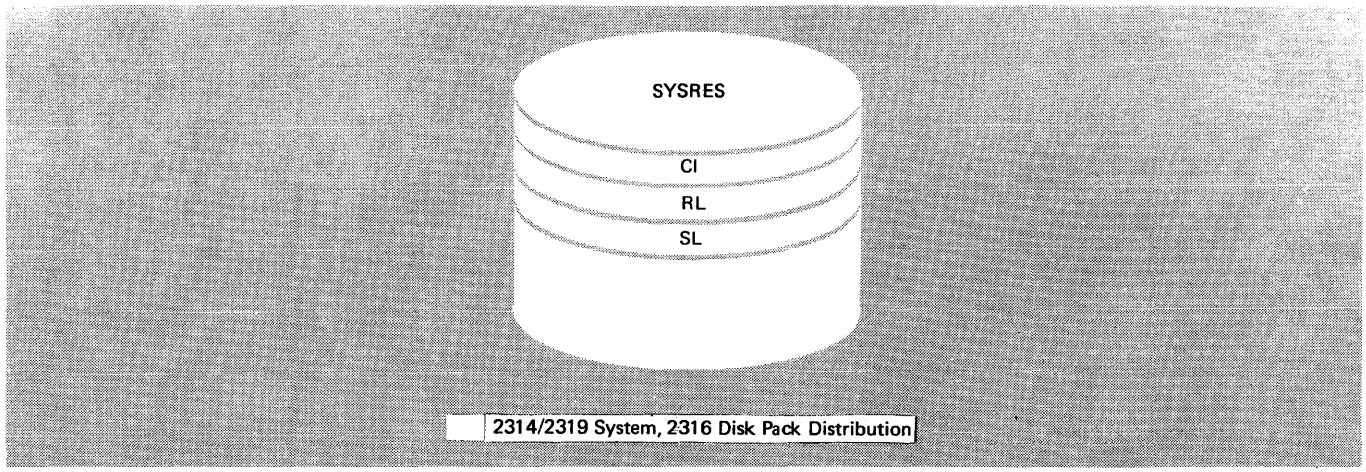
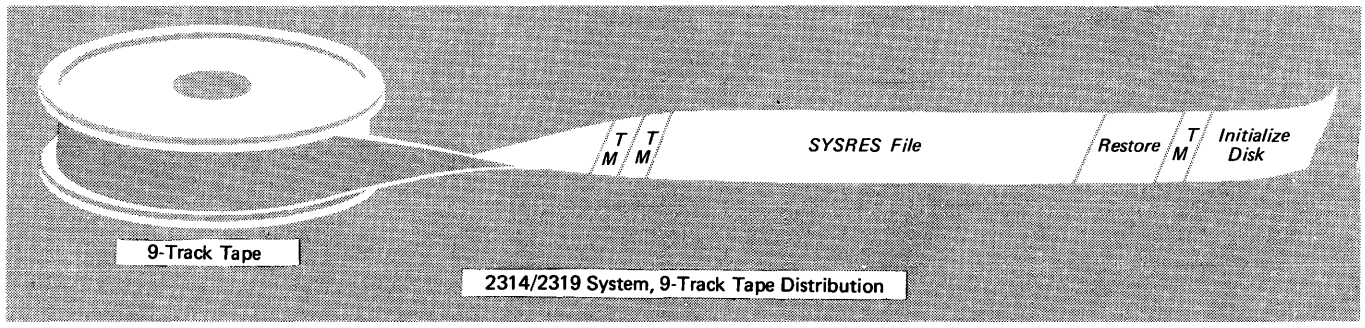
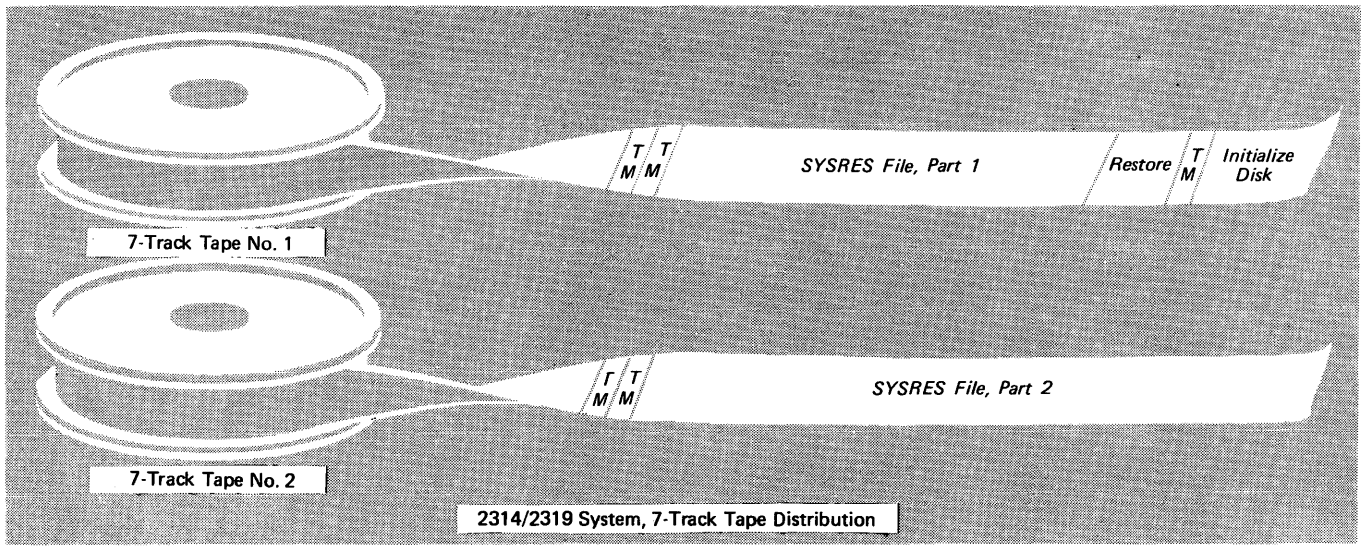


Figure 1-12. PROCEDURES--2311 Disk Operating System Distributions







Key:

CI - Core Image Library
Initialize Disk Program
Restore - Restore Tape to Disk Program
RL - System Relocatable Library
SL - System Source Statement Library
TM- Tape Mark

Figure 1-13. PROCEDURES--2314/2319 Disk Operating System Distributions

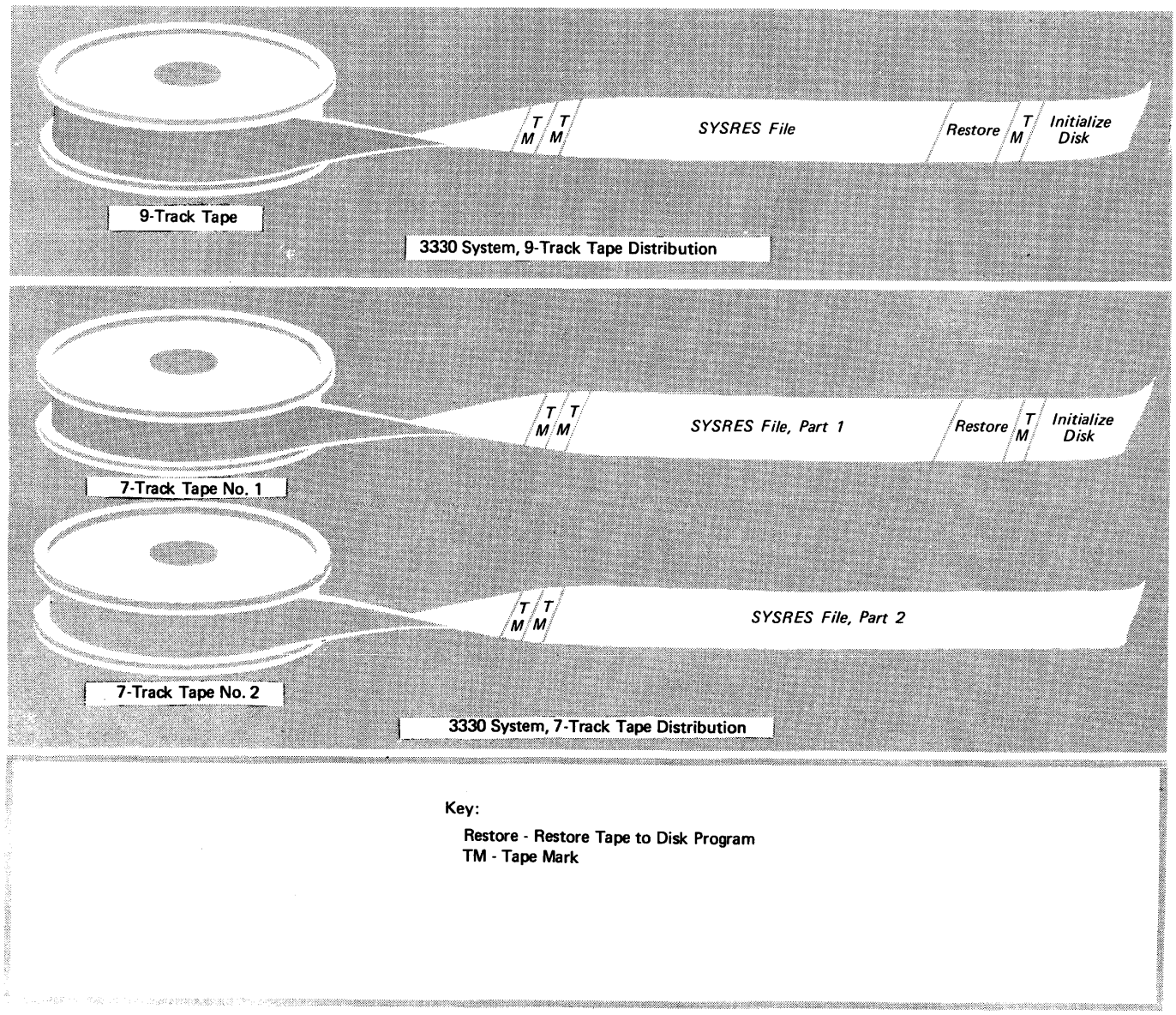


Figure 1-14. PROCEDURES--3330 Disk Operating System Distribution

The sequence of job steps depends on the configuration available and the operational packs being built. Certain activities are common:

1. The general system generation procedure is:

   a. Initialize the disk pack to contain the system.

   b. For disk and tape, restore the IBM-supplied tape onto disk. For two disk drives and no tape drives, copy the IBM-supplied disk to another disk to obtain backup.

   c. Perform the IPL procedure from the restored disk and create the recorder file.

   d. Retrieve sample problems.

   e. Create private libraries, if desired.

   f. Delete unwanted programs from the system (all libraries).

   g. Allocate library sizes required.

   h. Set standard labels for SYSLNK, SYS001, SYS002, and SYS003 if desired.

   i. Assemble supervisor into cards.

   j. Linkage edit and catalog supervisor and IBM components. (A supervisor cataloged to a private core image library can never be used because the IPL retrieval program searches only the system core image directory for phases beginning with $$.)

   k. Assemble IOCS modules.

   l. Catalog IOCS modules to the relocatable library.

   m. Delete supervisor and IOCS macros if not desired.

   Note: The operator needs to know the address of label CNLSVE in order to obtain a "special purpose dump". As he normally does not have access to the supervisor assembly listing, you should locate the address in this listing and ask him to write it down under the heading "special Purpose Dump" (Reference 10) in DOS Version 4 Messages, GC33-5009.

2. The supervisor generation macro instructions must be keypunched to form a single source deck.

3. DELETR cards are selected or prepared for each relocatable library component that is not needed in the system.

   On a system with at least two disk drives, you can copy selectively rather than delete (DELETR). To copy selectively, prepare the librarian cards in the form: COPYR IJx.ALL.

4. DELETS A.xxxxxxxx cards are selected or prepared for each macro definition that is not needed in the system. IBM-supplied macro definitions appear in each component section. They appear in the general Index under "macros, IBM-supplied" with pointers to the correct module number(s). Certain low-usage macro definitions, such as supervisor generation macros, may be retained on a system maintenance pack, rather than on the operational pack(s). The books, Z.DELETECL, Z.DELETERL, Z.DELETESL and Z.LINKEDIT, can be retrieved through SSERV. These books contain the necessary statements to delete or linkage edit selectively any components from the system. A pause card is read immediately before a deletion or linkage-edit job is performed. This allows you to enter END to perform the job or to type in CANCEL at the console printer keyboard to bypass that deletion or

linkage edit. To avoid going through the complete book performing selective jobs, choose only those cards needed from the book.

5. After a component is linkage edited into the core image library, the relocatable library space used to build the component can be freed by deleting the modules and condensing the library. The delete book Z.DELETERL can be used to perform this step. However, the relocatable library of the pack prior to updating a component must be rebuilt, by entering the appropriate modules either from cards, magnetic tape, or disk. On a system without magnetic tape this tradeoff must be carefully considered: disk tracks made available vs. speed and simplicity of component maintenance.

6. To use some compilers a certain collection of IOCS modules must be available in the relocatable library of each operational pack. These modules are preassembled and supplied in the relocatable library for the compilers that use them. Certain of these IOCS modules are linkage edited into each compiler object program. These modules are generated using the following macro definitions supplied by IBM:

   CDMOD — Card Reader/Punch

   PRMOD — Printer

   MTMOD — Magnetic Tape

   SDMODxx — Sequential DASD. SDMOD consists of ten similar macro definitions: SDMODFI for sequential disk with fixed input, SDMODFC for sequential disk with fixed output, etc. A complete description of SDMODxx is contained in DOS Supervisor and I/O Macros, GC24-5037, and DOS Version 4, GC33-5007.

   ISMOD — Index Sequential DASD

   DAMOD — Direct Access Method, DASD

   DIMOD — Device Independent Module

   With assembler language you can assemble these IOCS functions directly into their application programs, or the IOCS modules can be assembled separately and cataloged into the relocatable library. Separate assembly of IOCS modules requires no additional main storage or additional execution overhead in speed. These modules, shipped preassembled for IBM components, can also be used by any other program, if applicable. A separate assembly is preferable because:

   - Program assembly and reassembly time is minimized.
   - IBM supplied IOCS modules used by compilers may also be used by your application programs. Use of these modules reduces the assembly time of your application programs.
   - Use of preassembled IOCS modules facilitates program maintenance and standardization.
   - The xxMOD macro definitions just cited require a substantial number of cylinders in the source statement library.

   The corresponding generated modules ordinarily require fewer cylinders in the relocatable library. Thus, you may prefer to retain xxMOD macro definitions only on the system backup volume, cataloging a selection of generated modules onto each operational volume.

7. During each system generation and maintenance procedure, system libraries are periodically copied on magnetic-tape reels, disks, or cards to provide backup in case of subsequent specification errors or machine errors. These backup procedures may be omitted, but the

indicated maintenance procedures and any additional precautionary procedures desired should be followed.

The following IBM Basic Programming Support Utility Programs may be required for system generation and should be ordered with the initial distribution volume, depending on the configuration of the system. Each BPS utility must be loaded from cards by a separate IPL procedure.

- Distribution Program 360P-UT-208
- Initialize Disk 360P-UT-206
- Restore Tape-to-Disk 360P-UT-061
- Universal Character Set 360P-UT-048

Note: These utilities support neither the 3400 series magnetic tape units, nor the 3330 Disk Storage.

8. Perform a DSERV, or check a system directory printout to determine the contents and sizes of the system libraries during system generation. This procedure enables you to determine that enough blocks remain for linkage edit and catalog procedures. A DSERV requires the following control statements:

```
// JOB DSERV
// EXEC DSERV
   DSPLYS ALL
/*
/&
```

A printout of the system directory is provided automatically following a linkage edit with an OPTION CATAL specified or any // EXEC MAINT.

9. The initial system volume from IBM contains the volume serial number 111111.

If the IBM-supplied volume is a tape, initialize the disk pack with the volume serial number before restoring the tape to the disk. The standard labels (DLBL and EXTENT statements) shown for the examples are adequate for system generation. However, the EXTENT statement must be adjusted to reflect your volume serial number.

To use the standard labels for configurations with two disks, SYSLNK, SYS001, SYS002, and SYS003 can be assigned to the second disk. The standard labels shown in the following discussions assume the VTOC to be on Cylinder 199 of the residence volume (for 2311 and 2314/2319) or cylinder 403 (for a 3330). Following system generation, adequate standard label assignments should be set to support the installation's requirements.

10. Standard labels (OPTION STDLABEL) are defined on the system distributed by IBM for SYSLNK, SYS001, SYS002, SYS003, and SYSREC.

A reply of delete to the following message destroys the system residence file unless it is encountered during a MAINT reallocation run:

```
DOS.SYSTEM.RESIDENCE.FILE
4444A OVERLAP ON UNEXPIRED FILE
```

The core image library allocations on the IBM-supplied volume(s) are not sufficient to contain all of the system components. The adequacy of allocations can be determined through the use of Figure 1-8.

Refer to the section for each component for core image phase names, relocatable module names, and source statement macro names. Phases, modules, and macros are identified by component. Also included in the section for each component are those statements required for linkage editing and deleting.

11. If the installation-tailored supervisor does not exceed the SEND address of the IBM-supplied supervisor, the linkage editor, librarian, and assembler do not need to be linkage edited and cataloged again to the core image library.

Maintenance Procedures

The maintenance of disk operating system libraries can be performed with the MERGE function of the CORGZ librarian program. The MERGE function allows complete libraries or selective library entries to be merged into existing libraries. When the MERGE function is used, space availability is always a consideration. The following discussions apply to the core image, relocatable, and source statement libraries.

REPLACING IDENTICAL ENTRIES USING MERGE FUNCTION

When transferring entries that have identical names as existing entries of a library, the old entry (phase, module, or book name) is deleted from the library's directory, and the new entry is added to the end of the library's directory. The phase, module, or book is added to the end of the library.

ADDING UNIQUE ENTRIES USING MERGE FUNCTION

When uniquely named entries (phase, module, or book name) are transferred to an existing library, the names of the entries are added to the end of the library's directory, and the phase, module or book itself is added to the end of the library.

General Library Updating Techniques using the Merge Function of the CORGZ Librarian Program

The techniques presented here apply to the examples given. They are intended as a guide and do not necessarily satisfy all requirements. The method that you apply to maintain the libraries depends upon the library structure, and special requirements concerning the maintenance of your library.

COPYING SELECTIVELY TO MERGE LIBRARIES

When there are more desired entries in an existing library, it is faster to selectively copy (MERGE) to the library containing the most number of desired entries. Thus, the number of entries transferred and directory searches are kept to a minimum to save time. If there is insufficient space in a library to accommodate additional entries, unwanted entries can be deleted, the library condensed, and the new entries then added to the library.

DELETING UNWANTED ENTRIES AND MERGING AN ENTIRE LIBRARY WITH ANOTHER LIBRARY

A second technique is to apply the IBM-supplied DELETERL or DELETESL book to delete all unwanted entries from a library and copy the entire library

to another library, thus merging the two libraries. Using this technique requires that the library being copied to contains encugh space to accommodate the entire library being copied. If there is insufficient space, the library being copied to can be condensed or reallocated.

COPYING SELECTIVELY, MERGING TWO LIBRARIES TO CREATE A THIRD LIBRARY

If there is insufficient space to accommodate a merge of two libraries, a third library can be created tc contain selected entries frcm the two libraries being merged. This technique eliminates the need for condensing or reallocating an existing library. Note that this technique can be applied by using only two disk drives.

DELETING UNWANTED ENTRIES FROM TWO LIBRARIES AND MERGING BCTH LIBRARIES TC CREATE A THIRD LIBRARY

Ancther technique for merging two libraries is to delete unwanted entries from the two existing libraries, and merging the twc libraries, in their entirety, by copying them to create a third library. Note that this technique can be applied by using cnly two disk drives.

Copying the IPL Record

You may transfer the IPL record $$A$IPL2 between SYSRES and SYS002 (RES and NRS) in either direction. To do this, use the COPYI cr CCPY ALL statement under the MERGE function of the CORGZ program. COPYI copies only $$A$IPL2. COPY ALL copies $$A$IPL2 and the system core image, relocatable, and source statement libraries.

CONVENTIONS FOR MERGE EXAMPLES

The following conventions are used in the maintenance examples given here. If a macro definition called m is an entry in a library called L (L being any library and m being any macrc definiticn entry in any library L) then any macro definition entry m in any library L can be designated as Lm.

Assume that there are two libraries L called A and B, and that the macrc definition entries 1, 2, and 3 in library A are IBM-supplied, and 8 and 9 are user-defined macro definitions. Further, assume that you receive a new release of DOS with a source statement library, called library B, containing IBM-supplied macro definitions 1, 2, 3, 4, and 5. Thus, library A and library B can be represented as follcws:

Library A-- A1, A2, A3, A8, A9

Library B-- B1, B2, B3, B4, B5

For the libraries defined, macrc-definition entries 1, 2, and 3 in library B are identical in name to 1, 2 and 3 in library A, but a later version cr update. Macro definitions 4 and 5 in library B are new IBM-supplied macro definitions for the release received, and macrc definitions 8 and 9 in library A are user-defined. The structures of the libraries defined here are used in the examples that fcllcw.

Maintenance Examples using Merge Function

For each example that follows:

- The macro definitions updated are selected arbitrarily,
- Library A is assumed to reside on disk drive A, and library B on disk drive B,
- Library A is assumed to be your existing library, and library B the IBM-supplied maintenance update (new release),
- The techniques described:

  a. can be applied by using only two disk drives.

  b. applies to the maintenance of all libraries (core image, relocatable, and source statement).

The following examples illustrate several ways to update an existing library.

EXAMPLE 1A: Copying Selectively to Merge Two Libraries

If the majority of the macro definitions desired are in library A, and it is desired to update macro definitions A2 and A3, selectively copy (merge) macro definitions B2 and B3 from IBM-supplied library B to library A. When there are many more macro definitions in one library than the other, it is faster to selectively copy to the library with the greatest number of entries, provided there is enough space in the receiving library. If there is not enough space in library A, the unwanted macro definitions A2 and A3 must first be deleted from library A, and library A condensed before the transfer can be attempted.

A representation of the results of selectively copying macro definitions B2 and B3 from library B to library A follows:

Library A-- A1, __, __, A8, A9, B2, B3

Library B-- B1, B2, B3, B4, B5

Note that the macro definitions B2 and B3 are added to the end of the library updated (library A).

EXAMPLE 1B: Variation of Selective Copying to Merge Two Libraries

A variation of the technique described in Example 1A demonstrates the capability of transferring in any direction. Thus, if the majority of macro definitions are in library B, and it is desired to update macro definitions A1, A2, and A3, selectively copy macro definitions A8 and A9 from library A to library B.

A representation of selectively copying macro definitions A8 and A9 from library A to library B follows:

Library A-- A1, A2, A3, A8, A9

Library B-- B1, B2, B3, B4, B5, A8, A9

Note that macro definitions A8 and A9 are added to the end of the library updated (library B).

EXAMPLE 2A: Deleting Unwanted Entries from A Library and MERGING Libraries by Copying One to the Other

Assuming that the majority of macro definitions are in library A, an alternate method of accomplishing an update similar to that described in Example 1A follows.

Delete macro definitions B1, B4, and B5 from library B, and completely copy library B to library A. Thus, the IBM-supplied DELETESL book can

delete all unwanted macro definitions from the source statement library, and the library transferred in its entirety.

A representation of the results of deleting macro definitions B1, B4, and B5 from library B, and copying library B follows:

Library A-- A1, __, __, A8, A9, B2, B3

Library B-- __, B2, B3, __, __

Note that library B is added to the end of library A, and that library A in Example 1A is identical to library A in this example.

EXAMPLE 2B: Variation of Deleting Unwanted Entries from A Library and Copying (MERGING) One to the Other

A variation of the technique described in Example 2A that demonstrates the capability of transferring in any direction follows.

Thus, to add macro definitions A8 and A9 to library B, delete macro definitions A1, A2, and A3 in library A and then completely copy library A to library B. Again, the IBM-supplied DELETESL book can delete all unwanted IBM-supplied macro definitions from the source statement library.

A representation of the results of deleting macro definitions A1, A2, and A3 from library A and copying library A to library B follows.

Library A-- __, __, __, A8, A9

Library B-- B1, B2, B3, B4, B5, A8, A9

Note that user macro definitions A8 and A9 are added to the end of the library being updated (library B), and library B in Example 1B is identical to library B in this example.

EXAMPLE 3: Copying Selectively, Merging Two Libraries to Create a Third Library

If space is a problem, a third library called C can be created to be a combination of selected macro definitions from both library A and library B. Thus, if the desired macro definitions in library A are A8 and A9 (the user's macro definitions), and B1, B2, and B3 in library B (IBM-supplied), then user macro definitions A8 and A9 can be selectively copied (merged) from library A to library C, and IBM-supplied macro definitions B1, B2, and B3 can be selectively copied (merged) from library B to library C. The advantage of this technique is that the need for a condense or reallocation of an existing library is eliminated, and time is saved. However, the time required to initialize a third disk pack must be taken into consideration. A representation of the results of selectively copying library A and library B (merging) to create library C follows:

Library A-- A1, A2, A3, A8, A9

Library B-- B1, B2, B3, B4, B5

Library C-- B1, B2, B3, A8, A9

Note that library C requires less available space than if either library A or library B were merged with each other.

EXAMPLE 4: Deleting Unwanted Entries from Two Libraries and MERGING Both Libraries to Create a Third Library

A variation of selectively copying from two libraries to form a third library follows.

To retain user macro definitions A8 and A9 in library A, and update IBM-supplied macro definitions A1, A2, and A3 in library A, delete A1, A2,

and A3 from library A, and B4 and B5 from library B. Then completely copy (merge) library A and library B to create a third library C.

A representation of the results of deleting unwanted macro definitions from library A and library B, and copying (merging) both libraries to create a third library C follows:

Library A-- __, __, __, A8, A9

Library B-- B1, B2, B3, __, __

Library C-- B1, B2, B3, A8, A9

Note that library C requires less available space than if either library A or B were merged with each other. Also note that library C in Example 3 is identical to library C in this example.

CONSIDERATIONS FOR MERGING

The DLBL and EXTENT file definition statements must precede the MERGE control statement. The file name, logical unit, and direction of transfer for the following operations are indicated in Figure 1-15.

1. In merging to or from a system residence file (NRS), the modified or duplicate file name must be IJSYSRS, the logical unit must be SYS002, and the file ID (identification) must be identical to the ID supplied when the file was created.

2. In merging to a private relocatable library file, the file name must be IJSYSRL, the logical unit must be SYSRLB, and the file ID must be identical to the ID supplied when the file was created.

3. In copying from a private relocatable library file, the file name must be IJSYSPR, the logical unit must be SYS001, and the file ID must be identical to the ID supplied when the file was created.

4. In merging to a private source statement library file, the file name must be IJSYSSL, the logical unit must be SYSSLB, and the file ID must be identical to the ID supplied when the file was created.

5. In copying from a private source statement library file, the file name must be IJSYSPS, the logical unit must be SYS000, and the file ID must be identical to the ID supplied when the file was created.

6. When you merge to a private core image library file, the file name must be IJSYSCL, the logical unit must be SYSCLB, and the file ID must be identical to the ID supplied when the file was created.

7. When you merge from a private core image library file, the file name must be IJSYSPC, the logical unit must be SYS003, and the file ID must be identical to the ID supplied when the file was created.

		1	2	3	4	5	6	7
File Name Logical unit	SYSRES	IJSYSRS SYS002	IJSYSRL SYSRLB	IJSYSPR SYS001	IJSYSSL SYSSLB	IJSYSPS SYS000	IJSYSCL SYSCLB	IJSYSPC SYS003
Merge RES to NRS	from	to						
Merge NRS to RES	to	from						
Merge RES to PRV	from		to		to		to	
Merge NRS to PRV		from	to		to		to	
Merge PRV to RES	to			from		from		from
Merge PRV to NRS		to		from		from		from
Merge PRV to PRV			to	from	to	from	to	from

Figure 1-15. PROCEDURES--Merge Operation File Name and Logical Unit Identification

Diagnostic messages for erroneous assignments, file definitions, etc., are provided on SYSLST.

The following is an example of a job set up for the MERGE function. The sections of the job are bracketed and numbered 1 through 15. The explanations that follow the job are keyed to the job sections. The example assumes two disk drives with addresses of 190 and 191 that:

1. the MERGE function of the CORGZ librarian program is on SYSRES which is on 190 by virtue of an IPL

2. SYSRLB and SYSSLB are assigned to 191, SYS000 and SYS001 to 190, and SYS002 to 191.

3. Following this, SYS002 and SYSCLB are assigned to 191 and SYS003 to 190.

```
        // JOB EXAMPLE
        // ASSGN SYSRLB,X'191'
        // ASSGN SYSSLB,X'191'
        // ASSGN SYS003,X'191'
        // ASSGN SYS000,X'190'
        // ASSGN SYS001,X'190'
        // ASSGN SYS002,X'191'

Note 1  // DLBL IJSYSRL,'PRIVATE RL',99/365
        // EXTENT SYSRLB,111111,1,0,1500,100

Note 2  // DLBL IJSYSSL,'PRIVATE SL',99/365
        // EXTENT SYSSLB,111111,1,0,1600,100

Note 3  // DLBL IJSYSPC,'PRIVATE CIL',99/365
        // EXTENT SYS003,111111,1,0,1700,100

Note 4  // DLBL IJSYSPR,'PRIVATE RL TEST',99/365
        // EXTENT SYS001,111111,1,0,1300,100
```

```
Note 5   // DLBL IJSYSPS,'PRIVATE SL TEST',99/365
         // EXTENT SYS000,111111,1,0,1400,100

Note 6   // DLBL IJSYSRS,'SYSTEM RESIDENCE',99/365
         // EXTENT SYS002,111111,1,0,1,170

         // EXEC CORGZ

            NEWVCL RL=10(2),SL=10(2),CL=10(15)
Note 7      COPYR ALL
            COPYS ALL
            COPYC ALL

            MERGE PRV,PRV
Note 8      COPYR ALL
            COPYS ALL

            MERGE NRS,PRV
Note 9      COPYR ALL
            COPYS ALL

         /*

         // ASSGN SYS003,X'190'
         // ASSGN SYS002,X'191'

Note 10  // DLBL IJSYSCL,'PRIVATE CIL',72/365
         // EXTENT SYSCLB,111111,1,0,1700,100

Note 11  ASSGN SYSCLB,X'191'

Note 12  // DLBL IJSYSPC,'PRIVATE CL TEST',72/365
         // EXTENT SYS003,111111,1,0,1500,100

Note 13  // DLBL IJSYSRS,'SYSTEM RESIDENCE',72/365
         // EXTENT SYS002,111111,1,0,1,170

         // EXEC CORGZ

Note 14     MERGE PRV,PRV
            COPYC ALL

Note 15     MERGE NRS,PRV
            COPYC ALL

         /*

         /&
```

The following explanations are keyed to the sections of the job:

Note 1. File definition statements for a private relocatable library file which is created and updated.

Note 2. File definition statements for a private source statement library file which is created and updated.

Note 3. File definition statements for a private core image library file which will be created.

Note 4. File definition statements for a private relocatable library file from which modules are copied.

Note 5. File definition statements for a private source statement library file from which books are copied.

Note 6. File definition statements for a modified, or duplicate system residence file from which modules and books are copied. Note that this file could be the old SYSRES (with user programs).

Note 7. Creates private core image, relocatable, and source statement libraries on SYS003, SYSRLB, and SYSSLB and copies the relocatable and source statement libraries from the system residence file on SYSRES into them.

Note 8. Merges all modules and books from private relocatable and source statement libraries on SYS001 and SYS000 into the appropriate private libraries created on SYSRLB and SYSSLB.

Note 9. Merges all modules and books from the relocatable and source statement libraries of a modified, or duplicate system residence file on SYS002 into private libraries created on SYSRLB and SYSSLB.

Note 10. File definition statements for a private core image library file just created and to be updated.

Note 11. To merge to the private core image library just created, assign it to SYSCLB. You must assign the 'from' file to SYS003, and the file is a previously created private core image library. See the note under Considerations for Merging.

Note 12. File definition statements for a private core image library from which phases are copied.

Note 13. File definition statements for a modified or duplicate system residence file from which the phases of the core image library are copied.

Note 14. Merge all phases from the private core image library on SYS003 into the newly created private core image library on SYSCLB.

Note 15. Merges all phases from the core image library of a modified, or duplicate, system residence file on SYS002 into the newly created private core image library on SYSCLB.

For a more detailed description of the MERGE function see DOS System Control and Service, GC24-5036.

DOS System/370 Distribution Program and BPS Job Control Coding Specifications

Figure 1-16 provides job control information that is to be inserted in the program decks described in the sections Processing the Distribution Tape and Creating a Back-up of your Generated System Volume (2311 and 2314/2319 systems only).

```
// DATE = yyddd

    yy = 00-99 decimal (years)
   ddd = 01-366 decimal (days)
```

```
// ASSGN SYSxxx,x'cuu',dd[,X'ss']

   xxx = logical unit
   cuu = channel and unit of device

   • The channels used by the BPS supervisors are:

            Multiplexer (channel 0)
            Selector 1
            Selector 2
            Selector 3 and 4 (DOS distribution program only)

    dd = one of the following device type codes:

          C1-3210/3215 Console Printer-Keyboard

          D1-2311 Disk Drive
          D3-2314/2319 Disk Drive
          D4-3330 Disk Storage

          L1-1403 or 3211 Printer
          L2-1443 Printer

          R1-2540 Card Read-Punch (reading only), or 3505 Card Reader
          R2-2540 Using Punch-Read-Feed feature
          R3-1442 Card Read-Punch
          R4-2501 Card Reader
          R5-2520 Card Read-Punch

          T1-2400, 3410, or 3420 7-track Tape
          T2-2400, or 3420 9-track Tape

    ss =  90 for 7-track tape unit
          C0 1600 BPI
          C8 800 BPI
```

```
// VTOC STRTADR=(cccchhh),EXTENT=(yy)

   cccchhh = cylinder and head number of starting address
        yy = number of tracks allotted to VTOC in decimal (1-20)
```

```
VOL1 nnnnnn    nnnnnn = Volume Serial Number
```

Figure 1-16. PROCEDURES--Distribution Program and BPS Job Control Coding Specifications

Processing the Distribution Tape

The IBM-supplied system residence tape must be copied onto a disk pack before system generation can be performed. The pack that is to contain the system must be initialized with a volume label and a volume table of contents (VTOC) at cylinder 199 (2311, 2314/2319), or cylinder 403 (if a 3330 Disk Storage is used).

Initializing the Disk

To initialize the disk:

- Mount the distribution tape.
- Set write protection switch to R/W (3330 only).
- Place the following control cards in the card reader, in the sequence shown:

```
// JOB INTDSK
// DATE yyddd
// ASSGN SYSOPT,X'cuu',dd (disk) see Note 1.
// ASSGN SYS00n,X'cuu',dd (disk) see Note 2.
// ASSGN SYSLOG,X'cuu',dd
// EXEC
// UID nn see Note 3.
// VTOC STRTADR=(cccchhh),EXTENT=(yy) see Note 4.
VOL1nnnnnn see Note 5.
// END
```

Note 1: SYSOPT is required to specify the disk device to be initialized.

Note 2: n = 2,3,4, or 5. SYS002 through SYS005 are optional to specify additional disk devices to be initialized.

Note 3: Refer to the DOS Version 4 publication for a complete description of the UID control card.

For 2311 and 2314/2319, nn=:

IR Previously flagged tracks are to retain their flags without surface analysis, or

IA Denotes surface analysis on all tracks. Generation of home address (HA) and R0 records, preformatting of IPL records, writing of volume label and VTOC, or

IS For packs that have already been initialized this entry can be used to change the volume label(s) and the VTOC location.

For 3330 Disk Storage, nn=:

IQ Quick initialization. No surface analysis. No home address generation, only standard R0 generation. IPL records are preformatted, volume labels and VTOC are written, or

IS See above.

Note 4:

For 2311:	cccchhh=0199000	yy=1-10
For 2314/2319:	ditto	yy=1-20
For 3330:	cccchhh=0403000	yy=1-19

For each output disk specified (max. 5) a set of

```
// VTOC STRTADR=(cccchhh),EXTENT=(yy)
VCL1nnnnnn
```

cards are to be added.

Note 5: nnnnnn = Volume serial number
Columns 42-51 are reserved for user's identification. A label control set consisting of a VTOC control card and a VOL1 control card is required for each pack assigned. The label control sets apply to the assignments in the order as specified in the job stream and not in strict numerical sequence.

- Dial on the console the address of the tape unit containing the distribution tape into the CPU load address switches.

- Press LOAD.

- When the Wait light comes on, press START and EOF on the card reader.

- If message 4444A appears in SYSLOG, type in 4 blank, and press INTERRUPT to delete unexpired files, one by one. Type in 2 blank to delete all unexpired files at once.

- The message *END OF INTDSK* appears on SYSLOG when initialization is complete and the CPU enters the wait state. The DOS system can then be copied on the disk.

Bypassing the Initialize Disk Procedure

If the disk has previously been properly initialized, the Initialize Disk procedure can be omitted and the following procedure performed to bypass the initialize disk routine:

- Mount and ready the distribution tape.

- Place the following control cards in the card reader in the sequence shown:

```
// JOB INTDSK
// DATE yyddd
// ASSGN SYSLOG,X'cuu',dd
// FILES SYSIPT,1
```

- Dial the address of the tape drive containing the distribution tape in the CPU load address switches.

- Press LOAD.

- When the Wait light comes on, ready the card reader containing the control cards by pressing START and EOF.

- The tape forward-spaces past the initialization program and the following message appears on SYSLOG:

```
000C
4000A
```

Restoring the IBM-supplied System Tape to Disk

The DOS Distribution program that is part of the IBM-supplied system tape is a modified version of the restore program available with 360P-UT-208. The latter program, however, does not support the 3330 Disk Storage. Moreover, the DOS distribution Program can use alternate disk and tape drives, if necessary. The IBM-supplied system tape may contain files for more than one volume. You are informed on which device the file is to be restored and are given the option to restore or bypass each file.

Following initialization or bypassing initialization of the disk pack, the IBM-supplied system tape can be restored to disk by this procedure:

- Do not rewind the distribution tape.
- Clear the card reader.
- Set write protection switch to R/W (3330 only).
- Place the following control cards in the card reader:

Note to all 2311 users with 7-track tape: The assignment of SYS001 is not required for Volume 3.

```
// JOB DISRST
// DATE yyddd
// ASSGN SYS000,X'cuu',dd (required output disk)
// ASSGN SYS001,X'cuu',dd (alternate output disk, if used)
// ASSGN SYS002,X'cuu',dd,X'ss' (alternate input tape, if used)
// ASSGN SYSLST,X'cuu',dd
// ASSGN SYSLOG,X'cuu',dd
// EXEC
```

- Press LOAD on the console (IPL from the distribution tape).
- When the Wait light comes on, ready the card reader by pressing START and EOF.
- If message 4444A appears on SYSLOG, type in 4 blank, and press INTERRUPT to delete unexpired files, one by one. Type in 2 blank to delete all unexpired files at once.

The following three-part message is issued when each of the files of the IBM-supplied tape(s) is to be restored next on the assigned logical unit.

```
file name
                SYS000
THIS FILE FOR   SYS001
4307A
```

Either restore the file by typing in 2 blank and pressing INTERRUPT, or bypass the file by typing in 4 blank and pressing INTERRUPT.

The following message is issued whenever output is switched to another volume. This message gives the logical unit assignment on which the next file is to be restored.

```
                SYS000
SWITCHING TO    SYS001
4308A
```

Note to all multiple 2311 users requiring Volume 3: Volume 3 is restored to SYS000.

Be sure that an initialized pack is mounted on the designated drive. Type in 2 blank and press INTERRUPT to restore the volume. Type in

0 blank or 1 blank and press INTERRUPT to terminate the job. For a single-drive user, SYS000 and SYS001 refer to the same drive.

- If the message END OF VOLUME ON SYSIPT 3777A occurs while restoring a multivolume file, ready the next reel on SYSIPT reply 2 blank and press INTERRUPT to continue. This message will not be encountered if the second volume of a two-volume system distribution is mounted on an alternate drive assigned to SYS002.

```
// ASSGN SYS002,X'cuu',dd[,X'ss'] (tape)
```

- When the job is finished, the following message is printed on SYSLOG and the disk pack is ready for system generation:

```
DOS SYSTEM RESIDENCE FILE
nnnnnn RECORDS RESTORED FOR ABOVE FILE
3007
END OF JOB
3008
```

Creating a Back-up of your generated System Volume

At the end of system generation, you should copy the system volume for operational volume backup. To restore the files to a disk pack, the IBM BPS copy and restore or the distribution program utility programs should be used. Familiarity with the BPS utility publications as listed in the Preface of this publication is necessary. BPS Messages, appearing later in this module, lists the BPS messages.

CREATING A BACK-UP TAPE OF 2311 AND 2314/2319 SYSTEMS

The BPS program 360P-UT-208 can copy a 2311 or 2314 system pack to tape for backup. (For a more detailed information see IBM System/360 BPS Distribution Program Specifications and Operating Guide, C21-5001.) The resulting tape is a self-loading tape that is similar to the IBM-supplied system tape. The tape produced by this program has the following format:

- IPL
- Initialize disk program
- Tapemark
- IPL
- Restore program
- File identification record
- File label information
- Disk file (DOS system)
- Tapemark

To copy a system pack to tape, using the copy function of 360P-UT-208:

- Mount the system pack and a tape.
- The following job control cards must be placed in the program deck between the cards containing ID numbers C208 and D208 in columns 73-76:

```
// JOB DISCPY
// DATE  yyddd
// ASSGN SYSLOG,X'cuu',dd
// ASSGN SYSLST,X'cuu',dd
// ASSGN SYS000,X'cuu',dd (disk)
// ASSGN SYS001,X'cuu',dd[,X'ss'] (tape)
// EXEC
```

- Place the deck in the card reader.

- The following utility modifier card must be placed immediately following the program deck:

```
     column 8                                              column 53
     ↓                                                     ↓
// UDS 'field one of format 1 DASD file label 44-characters'
```

- Dial on the console the address of the card reader.
- Check the units and settings; then press the System-Reset key.
- When the Wait light comes on, press START and ECF on the card reader.
- Press the console load key.

Restoring the Back-up Tape to Disk, using the Restore Function of 360P-UT-208

See the section entitled: Restoring the IBM-supplied System Tape to Disk. The procedure described there can also be used to restore the back-up tape of your generated system to disk. Please note, however, that 360P-UT-208 does not support multivolume disks. After bypassing initialization the back-up tape can be restored to a disk pack (which should have been initialized previously).

CREATING A BACK-UP TAPE OF A 3330 SYSTEM

A 3330 system pack cannot be copied to a tape using a BPS utility (for instance, 360P-UT-208), because BPS does not support the 3330 Disk Storage.

A back-up can be obtained under DOS using the Copy and Restore Disk function of 370N-UT-491 (see the DOS Version 4 manual listed in the Preface of this manual).

The user can copy his system to tape in two ways:

- Copy volume (complete system pack)
- Copy file

The first method is the simpler of the two: all files on the system pack, plus any other data, are copied to tape(s). The second method needs specific file information for every file to be copied; moreover, only one file can be copied per tape volume. Generally speaking, the second method requires more tape volumes than the first.

Copying a System Disk to Tape

1. Before the copy disk function can be performed, module IJWCDT must be link-edited and cataloged into the core image library, if this has not been done yet. Put the following cards in the card reader, in the sequence shown:

```
// JOB LINKEDIT COPY DISK UTILITY
// OPTION CATAL
 INCLUDE IJWCDT
// LBLTYP NSD(10)
// EXEC LNKEDT
/&
```

2. To copy your generated system:

   - Mount the system pack.
   - Mount the back-up tape on a tape drive and ready the device.
   - Place the following cards in the card reader, in the sequence shown:

```
// JOB COPY DISK VOLUME TO BACK-UP TAPE
// ASSGN SYS004,X'cuu'     DISK INPUT (VOLUME TO BE COPIED)    (Note1)
// ASSGN SYS005,X'cuu'     TAPE OUTPUT (BACK-UP TAPE)
// ASSGN SYS003,UA
// UPSI 00100
// EXEC CDKTP
// UCR TV,A=(1504),E=(3330)     (Notes 2 and 3)
/&
```

Note 1: cuu denotes the channel and unit address.

Note 2: The parameter A= in the UCR card specifies the most common physical record length of the area to be copied to obtain the best performance. (1504 is the maximum length of a record in the core image library.)

Note 3: The entry E= in the UCR card specifies the device type of the disk storage device.

When the first tape volume is full, the following message will be printed:

```
BG 4140A NO ALTERN DRIVE ASSGN    UOUT    SYS005=cuu
```

The tape is rewound and unloaded by the system. As soon as a new tape volume is mounted and the drive is readied, reply as follows to continue the job:

```
newtap
```

Replacing a full tape volume by another tape may have to be done more than once. Not before the whole disk volume has been copied, will the system terminate the job with the message:

```
BG  EOJ COPY
```

provided the job name was indeed COPY.

Restoring the Back-up Tape to Disk

If your DOS system ever becomes defective, you must restore your back-up tape to a disk volume. In order to do this, first of all reload a disk pack with the IBM-supplied system tape so as to obtain a system under control of which the restore procedure can be executed.

Next, the following steps are to be performed:

1. Using the reloaded PID system, module IJWRTD must be link-edited and cataloged into the core image library. Use the following job control cards:

```
// JOB LINKEDIT RESTORE DISK UTILITY
// OPTION CATAL
  INCLUDE IJWRTD
// LBLTYP NSD(10)
// EXEC LNKEDT
/&
```

2. To restore your generated system:

   - Mount the back-up tape on a tape drive and ready the device.
   - Mount a disk pack (re-initialized if necessary) and ready the disk drive.
   - Place the following job control cards in the card reader, ready the reader, and start executing the job:

```
// JOB RESTORE BACK-UP TAPE TO DISK
// ASSGN SYS004,X'cuu'      BACK-UP TAPE            (Note 1)
// ASSGN SYS005,X'cuu'      OUTPUT DISK
// ASSGN SYS003,UA
// UPSI 10000
// DLBL UCUT,'(filename of the system residence file)',99/365 (Note 2)
// EXTENT SYS005,,1,0,nnnnn,mmmmm   (Notes 3 and 4)
// EXEC RTPDK
/&
```

Note 1: cuu denotes the channel and unit address.

Note 2: (filename of the system residence file) can be any name for this new system, for instance DOS.SYSRES.FILE. This name will be kept temporarily in the F1 label of the VTOC and will eventually be overwritten by the original filename of the copied system pack.

Note 3: nnnnn=00001 denotes the relative track address of the lower limit of the SYSRES file which normally starts at cylinder 0, track 1.

Note 4: mmmmm denotes the number of tracks occupied by the new file (Note 2). This number is not significant here, because all the back-up tapes will be restored regardless of this limit. When your copied system has been successfully restored, the original extent limits of the copied system are written in the system's F1 label in the VTOC.

If more than one back-up tape is involved, the following message will be displayed after end-of-reel is detected:

```
BG 8001D  IS IT EOF
```

Reply by typing in n (for 'no'). The tape will be rewound and unloaded after which the second back-up tape volume can be mounted. Meanwhile the system will print the following message:

```
BG 4140A   NO ALTERN DRIVE ASSGN   UIN   SYS004=cuu
```

After the tape drive with the newly mounted reel has been readied, reply by typing in:

```
newtap
```

The above procedure must be repeated for each tape volume to be restored. After the last tape has been restored, reply y (for 'yes') to the message

```
     BG 8001D   IS IT EOF
```

upon which the system displays the final message

```
     EOJ RESTORE
```

provided the jobname was indeed RESTORE.

3. As a last step, it is advisable to run a LVTOC job to check whether all files have been restored, and to keep track of the extents of the files. Before running this job, module IJWLTVB must be link-edited and cataloged into the core image library. Use the following job cards in the sequence shown:

```
// JOB LINKEDIT LISTVTOC UTILITY
// OPTION CATAL
  INCLUDE IJWLTVB
// LBLTYP NSD(10)
// EXEC LNKEDT
/&
// JOB LIST VTOC
// ASSGN SYS004,X'cuu'       SYSTEM DISK
// ASSGN SYS005,X'cuu'       PRINTER
// EXEC LVTOC
/&
```

Disk Operating System Sample Problems

The sample problems provided with the Disk Operating System (DOS) demonstrate to the user, particularly the operator, each component of the programming system. Although the problems are general and illustrative rather than detailed and exhaustive, they nevertheless serve as a minimal test of each user's programming system. The user's programming system is built from the more general system supplied by IBM.

The sample problems are designed to be run on a minimum system configuration including at least one 2311 disk drive (the system residence volume), a card reader/punch, printer, and a 3210/3215 Console printer-keyboard. The minimum background partition storage capacity required for each sample problem is given in Figure 1-8 and in the module for each component.

No data is required for any of the sample problems, except RPG, and PL/I (D). The requirements for each problem are described in the examples that follow.

All sample problems, except those for multiprogramming, are included in the IBM-supplied core image and source statement library volume. The multiprogramming sample problem is prepared by the user. These sample problems included in the source statement library volume are retrieved as card decks by the SSERV librarian program. Each card deck is either a source program or a set of control cards. Individual decks are preceded by a CATALS card and a BKEND card and followed by a BKEND card. Once the sample problems have been retrieved, they can be deleted from the user's operational system disk during system generation.

Retrieving the Sample Problems

The sample problems are retrieved from the second volume (core image and source statement library volume) during system generation. Although the order for retrieving the problems from the disk is optional, it is recommended that they be retrieved in the order of intended execution.

This facilitates the task of preparing the job stream. The following job step is necessary to selectively retrieve all of the sample problems:

```
// JOB SAMPLPRB
// EXEC SSERV
   DSPCH Z.FO1,Z.CB1,Z.RG1,Z.PL1,Z.AS1
   DSPCH Z.SM1,Z.UTTPPR1,Z.AS2,Z.SM2
   DSPCH Z.UTDKPR1,Z.AS3,Z.UTDCPR1
   DSPCH Z.AS4,Z.SM4,Z.UTTPPR2
   DSPCH Z.AS5,Z.SM5,Z.UTDKPR2
   DSPCH Z.AS6,Z.SM6,Z.UTDKPR3,Z.CB2,Z.DLCP12
   DSPCH Z.MCR1,Z.MCR2
   DSPCH Z.ORDC,Z.ORJT,Z.IIFSAMPL
/&
```

If you do not wish to retrieve a particular problem from the source statement library volume, delete the appropriate operand from the preceding DSPCH statement. All of the sample problems are in the Z sublibrary and all of that sublibrary can be retrieved with the DSPCH Z.ALL statement.

Card columns 73-80 in each sample problem deck (except RPG) contain an identification number and a sequence number. The identification number for the RPG sample problem deck is punched in columns 75-78. Card columns 1-5 contain the sequence number for RPG. Although these numbers can identify individual sample problems, it is recommended that the cards be machine-interpreted. These decks are punched in the order in which they are retrieved from the source statement library volume (preferably the intended order of program execution). To prepare each sample problem for execution, the user must:

- Remove all CATALS and BKEND cards. If the RPG or PL/I (D) sample problem is retrieved, remove the *END SOURCE DECK/BEGIN INPUT CARDS card from the programs.
- Punch and insert the necessary job control cards for each program to be run, as shown in the examples.

The programs are compiled or assembled, linkage edited, and executed with a minimum of operator intervention.

Notes: All six assembler sample programs will run under either Assembler D or Assembler F.

Do not attempt to run any sample problems for components or devices you do not have. Each sample problem is further discussed in the section covering the IBM-supplied program for the sample problem intended.

The sample problems can be run as separate jobs; but, when possible, they should be run as successive job steps within an operating system environment. A PAUSE card placed at the end of each sample problem to be run in successive job steps allows the operator to make any necessary changes in device assignments. The order for running the sample problems is not completely arbitrary. The assembler, sort/merge, and utility programs should be run consecutively; the output of one program becomes the input to the next program. The compiler and IOCS sample problems can either precede or follow the other programs. The multiprogramming example should be run last when the programs are run as successive job steps.

Physical and Logical I/O Assignments

It is assumed that the user has made assignments for these logical I/O functions:

SYSLOG	SYSPCH	SYSREC	SYS003
SYSRDR	SYSLST	SYS001	SYS004
SYSIPT	SYSLNK	SYS002	

The preceding assignments are for the background problem program area only. The assignments necessary for the multiprogramming sample problem are included in the input test data for that problem.

Unique disk extents should be assigned initially to SYSLNK, SYS001, SYS002, and SYS003. The extents defined during system generation should be valid for sample problem execution.

If standard assignments are missing, the necessary ASSGN cards should be inserted at the beginning of the job stream. After each job step of a sample problem, certain additional logical I/O assignments or reassignments may be required. Examples of such reassignments are shown in Figures 1-17 and 1-18. When reassignments are necessary, the user must furnish:

- The channel and unit number (X'cuu') for each tape or disk extent.
- The proper disk pack serial number and data cell volume number in all DLBL and EXTENT cards.

Tape, Disk, and Data Cell Configurations

In addition to the minimum system configuration, three tape drives are required to run the Tape Sort/Merge sample problem, and five tape drives are required to run the Tape and Disk Sort/Merge sample problem (2400 application). A 2314/2319 direct access storage facility is required to run the Tape and Disk Sort/Merge sample problem (2314 application). For installations with 7-track drives, convert feature OFF and translate ON (X'B8' as third operand of ASSGN) must be specified. Where 9-track tape drives are used, the third operand (X'B8') can be omitted. The logical I/O assignments for tape are shown in Figure 1-17, and those for disk and data cell are shown in Figure 1-18.

In addition to checking the proper function of each sample problem, the programs also test job control and linkage editor functions. They are helpful in verifying the correct generation of the user's operational system. The main purposes of the sample problems, however, are demonstration and instruction.

A listing of the source program and job control cards is written on SYSLST for each problem. If LOG is keyed into SYSLOG at the beginning of the job, a listing of all job control cards and operator messages is written on SYSLOG. Detailed setup procedures, including job control cards, are given for each sample problem.

The SYSLOG output for the 1401/1440/1460 Emulator Programs can be found in the Emulator Program manual listed in the Preface.

Sample Problem	Disk Extent No. 1	Tape Drive No. 1[1]	Tape Drive No. 2[1]	Tape Drive No. 3[1]	Tape Drive No. 4	Tape Drive No. 5
Tape Sort/Merge Execute	SYSRES	SYS004	SYS001[2] SYS003[3]	SYS002[2] SYS005[3]		
Tape and Disk Sort/Merge (2400 Application)	SYSRES	SYS001	SYS002	SYS003	SYS004	SYS005
Tape-to-Printer Utility Execute	SYSRES	------	SYS004	------		

1. Tape drive number refers to X'cuu' operand in ASSGN card.

2. If a 7-track tape drive is used, the third operand (X'B8') is required to turn byte convert off.

3. For Tape Sort/Merge, SYS001 and SYS003 must be assigned to the same tape drive, as must SYS002 and SYS005.

Figure 1-17. SAMPLE PROBLEMS--Tape Logical I/O Assignments

Sample Problem	Disk Extent No. 1	Disk Extent No. 2	Disk Extent No. 3	Disk Extent No. 4	Disk Extent No. 5
FORTRAN or Basic FORTRAN:					
Compile	SYSLNK	SYS001	SYS002*	------	------
Link Edit	SYSLNK	SYS001	------	------	------
Execute	-------	------	------	------	------
COBOL, American National Standard COBOL, RPG, Assembler:					
Compile	SYSLNK	SYS001	SYS002	SYS003	SYS004
Link Edit	SYSLNK	SYS001	------	------	------
Execute	-------	------	------	------	------
COBOL LCP					
Execute	SYS001	SYS002	SYS003	------	------
Disk Sort/Merge Execute	SYS002	SYS004	------	------	------
Tape and Disk Sort/Merge (2311 Application)	SYS001	SYS002	SYS003	------	------
Tape and Disk Sort/Merge (2314 Application)	SYS001	SYS002	SYS003	------	------
Disk-to-Printer Utility Execute	SYS004	SYS005	------	------	------
Data Cell-to-Printer Utility Execute	SYS004	------	------	------	------
1287, 1288, 1419:					
Assemble	SYSLNK	SYS001	SYS002	SYS003	------
Link Edit	SYSLNK	SYS001	------	------	------
Execute	-------	------	------	------	------

* FORTRAN IV only

Figure 1-18. SAMPLE PROBLEMS--Disk/Data Cell Logical I/O Assignments

The SYSLST output reproduces much of the SYSLOG output. In addition, SYSLST displays the source programs, storage maps, sort/merge and utility program control cards, problem results, and other information. If SYSLST is a 1403 printer equipped with the Universal Character Set (UCS) feature, see UCS command in the DOS Operating Guide, GC24-5022. If this specification is not made, the issuance of a control command by job control causes a command reject, resulting in job cancellation.

System Generation Examples

Two IBM 2311 Disk Drives Example

Figure 1-19 illustrates the general procedure to be followed by users with two or more 2311 disk drives available.

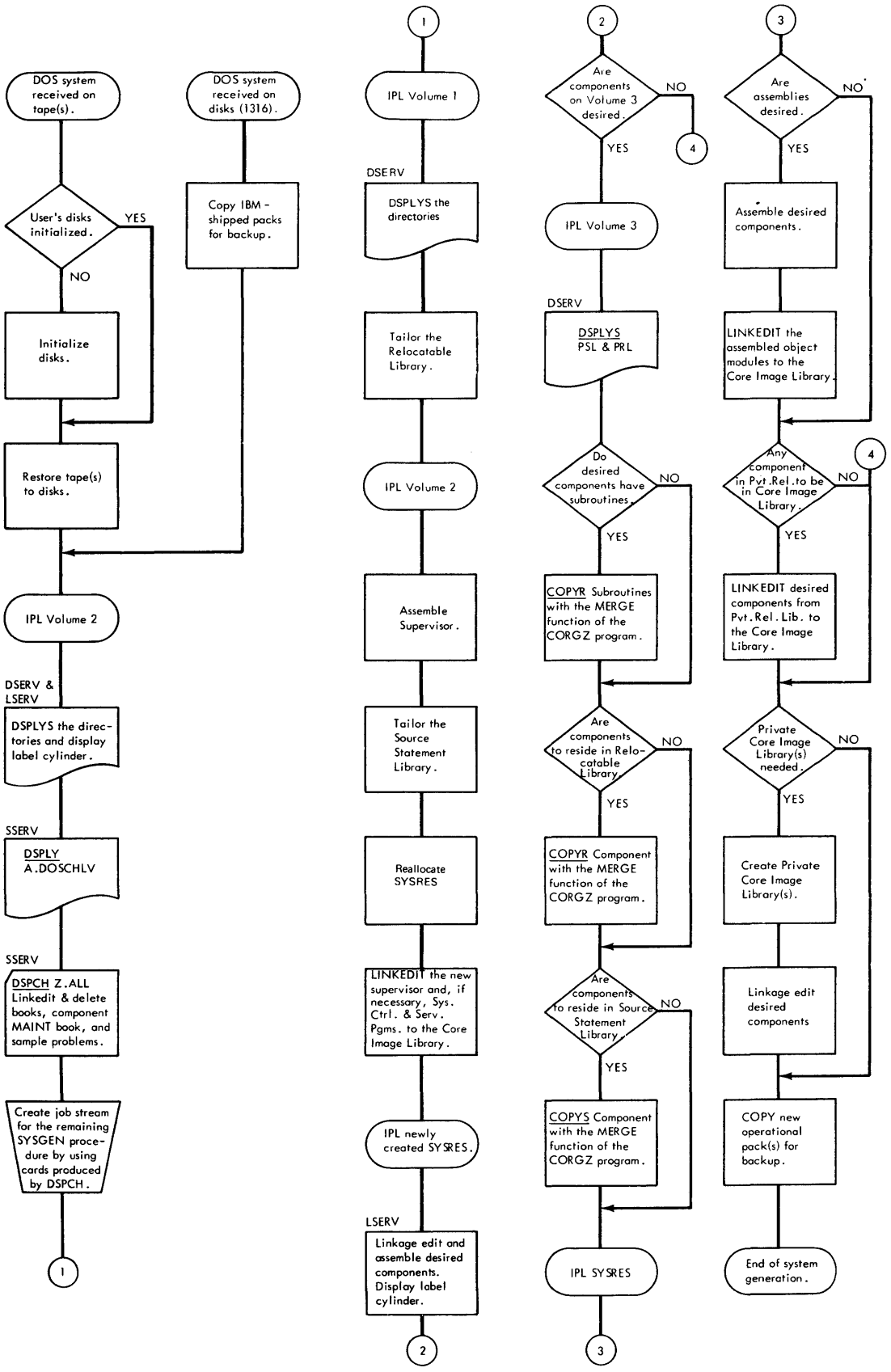


Figure 1-19. PROCEDURES--Overview of Multiple Disk System Generation

The following is only an example of one of the many possible methods for generating a system. Figure 1-20 illustrates the system configuration upon which the Example Two IBM 2311 Disk Drive is based.

Note: If you intend to follow this example for your system generation, then tailor it to meet the requirements of your installation.

INPUT/OUTPUT DEVICE CONFIGURATION FOR EXAMPLE ONE (FIGURE 1-20)

Device	Channel	Unit	Use
3505	0	0C	Card Reader (SYSRDR, SYSIPT)
3525P	0	0D	Card Punch (SYSPCH)
3211	0	0E	Printer (SYSLST)
3210/3215	0	1F	Console Printer-Keyboard (SYSLOG)
2311	1	91	Disk (SYSLNK, SYS001, SYS002, SYS003, SYS004, SYSRLB, SYSSLB, SYSCLB, SYSREC)
2311	1	90	Disk (SYSRES)
3420T7	1	80	Magnetic tape with the data conversion feature (switchable to channel 2)
3420T9	1	81	Magnetic tape (switchable to channel 2)

EXAMPLE ONE: RESULTS OF SYSTEM GENERATION

When system generation is completed for example one, the operational disk of the installation contains: user selected components and programs in its system core image library together with the installation's tailored supervisor, job control, linkage editor, and librarian programs. The tape(s) shipped by IBM are retained as a backup tape. They are self-loading tape(s) capable of being restored onto disk.

The sample problems are punched out during step 4.

The private relocatable and source statement libraries contain all modules and macro definitions shipped from IBM (except those deleted).

The system core image library of the operational pack is built to contain those IBM programs chosen.

The private core image library is created for the user who needs additional core image library space and for the user who desires to link-edit non-relocatable IBM-supplied and user programs for execution in a batched-job foreground partition, as well as the background partition.

The private libraries are condensed, and the IOCS modules are assembled and cataloged to the private relocatable library.

When system generation is completed, the sample problems should be run against the operational pack to ensure correct creation of all system programs.

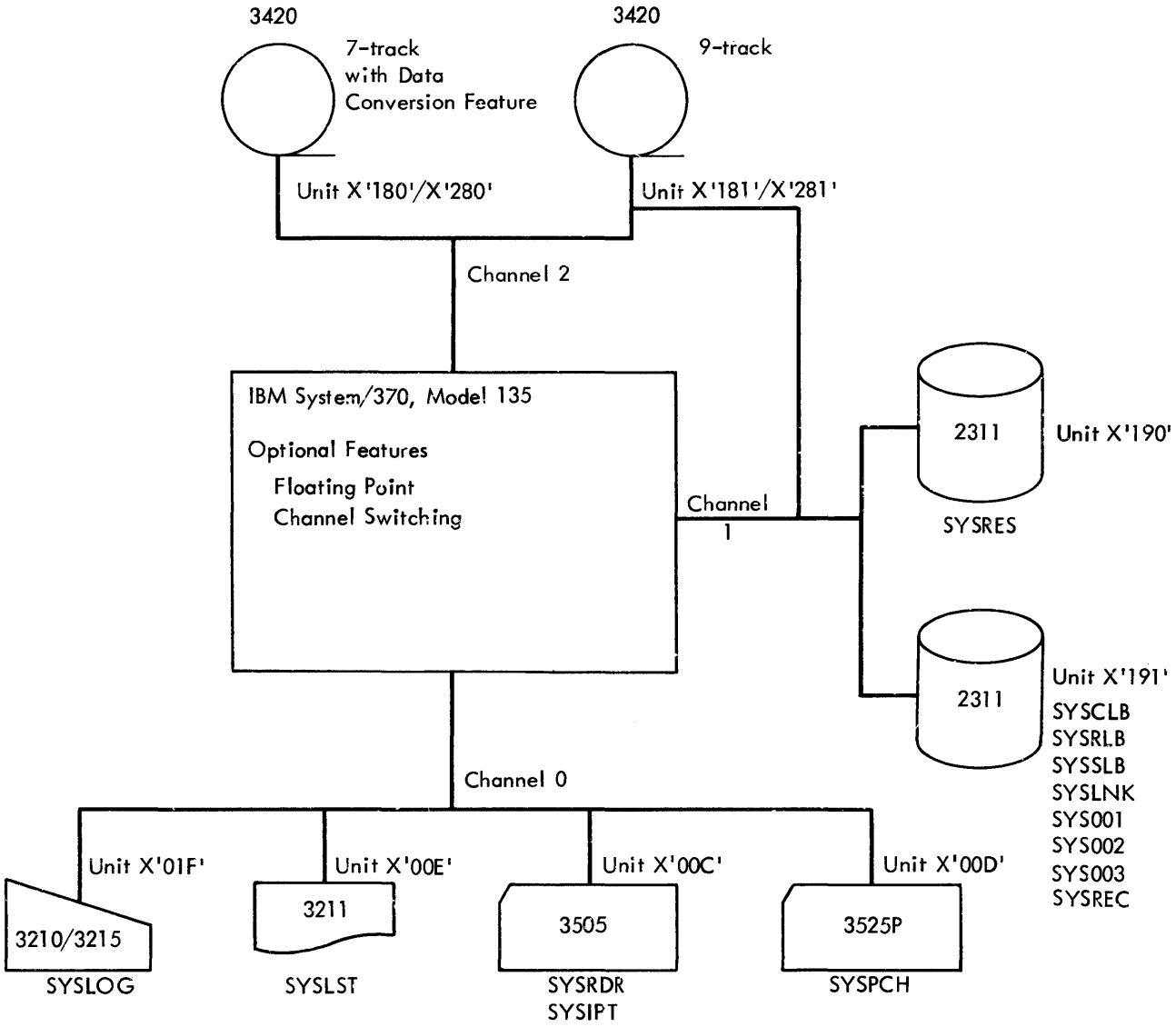


Figure 1-20. PROCEDURES--System Configuration for the Two IBM 2311 Disk Drive Example

The following steps are keyed to the two IBM 2311 disk drive example.

DISK-ONLY USERS

Disk-only users receive the 2311 system on 1316 disk packs. Another initialized disk pack (VTOC on cylinder 199) is required for private libraries and workfiles.

- Mount the IBM-supplied core image library, source statement library pack, and the workfile pack and proceed to Step 3.

TAPE USERS

Disk users with at least one tape unit available receive the 2311 system on one or two reels of magnetic tape. This IBM-supplied tape is capable of restoring the 2311 system on two 1316 disk packs. Another initialized disk

pack (VTOC on cylinder 199) is required for private libraries and workfiles.

- Mount the IBM-supplied tape and two disk packs and proceed as follows:

Step 1

Before restoring the tape, take the necessary action to perform or bypass the initialize disk routine.

a. If the disk packs have not been initialized, the IBM-supplied volume, a self-loading tape, is capable of initializing the packs. See Processing the Distribution Tape for the proper use of control cards.

   If initialization is required, mount the packs on units 190 and 191 and mount the IBM-supplied tape on unit 281 (9-track tape unit). Place the following cards in the card reader. Dial unit 281 in the load unit switches and press LOAD. When the system enters the wait state, press START and ECF on the card reader. Mount the third and fourth packs on units 190 and 191 after the previously mounted packs have been initialized, rewind the tape, and repeat the process. The multipart forms of the linkage editor maps and the supervisor listing are required by your IBM customer engineer and your system programmers for maintenance purposes.

```
// JOB INTDSK
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSOPT,X'190',D1
// ASSGN SYS002,X'191',D1
// EXEC
// UID IA
// VTOC STRTADR=(0199000),EXTENT=(10)
VOL1111111
// VTOC STRTADR=(0199000),EXTENT=(10)
VOL1111111
// END
```

b. If the disk packs were previously initialized, the initialize disk routine can be bypassed . Any volume serial number used in the EXTENT cards for the following steps must agree with the volume serial number used when the packs were initialized.

   To bypass the initialize disk program on the IBM-supplied tape, mount the IBM-supplied tape on unit 281 (9-track drive), mount the initialized disk on unit 190 (VTOC on cylinder, 199), mount the work pack on unit 191 and insert the following control cards in the reader. Dial unit 281 in the load unit switches and press LOAD . When the system enters wait state, press START and ECF on the card reader. The job is complete when the 00C and 4000A messages are logged.

```
// JOB INTDSK
// DATE 72182
// ASSGN SYSIPT,X'281',T2
// ASSGN SYSLOG,X'01F',C1
// FILES SYSIPT,1
```

Step 2

After step 1 is completed, you must restore the systems from the self-loading tape to the initialized packs. This step is omitted when a system is received on disk.

Do not rewind unit 281. To load the tape onto disk, place the following cards in the reader, dial unit 281 in the load switches, and press LOAD. When the system enters the wait state, press START and EOF on the card reader. See Restoring the IBM-Supplied System Tape-to-Disk for operating procedures.

Note: Volume 3 is restored to SYS000.

```
// JOB DISRST
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSLST,X'00E',L1
// ASSGN SYS000,X'190',D1
// ASSGN SYS001,X'191',D1
// EXEC
```

Step 3

Dial the address (unit 191) of Volume 2 (the core image and source statement system residence disk drive) into the load unit switches, and IPL (press LOAD) to pass control to the DOS supervisor. See IPL Control and ASSGN Statements for System Generation, later in this module, for the IPL control statements. When the system enters the wait state, press START on the card reader. The following cards are in the card reader (SYSRDR/SYSIPT):

```
ADD X'00C',3505
ADD X'00D',3525P
ADD X'00E',3211
ADD X'01F',1050A
ADD X'190',2311
ADD X'191',2311
SET DATE=06/30/72
ASSGN SYSLOG,X'01F'
LCG
ASSGN SYSRDR,X'00C'
ASSGN SYSIPT,X'00C'
ASSGN SYSPCH,X'00D'
ASSGN SYSLST,X'00E'
ASSGN SYSREC,X'191'
SET RF=CREATE
```

Step 4

Perform a sorted DSERV to display the directories. Then SSERV (display and punch) all the desired sample problems from the source statement library into cards. Punched output includes the sample problems with BKEND and CATALS cards. Four other books in the source statement library that should be punched out at this time (Z.LINKEDIT, Z.DELETECL, Z.DELETERL, and Z.DELETESL) contain the necessary control statements to selectively linkage edit and delete all IBM components. It is advisable to use these four books to assist you with the creation of the remaining system generation job stream. The sample problem program names, and the linkage edit and the delete book names can be chosen and punched into the DSPCH statement(s).

After these sample problems and books have been punched, they can be deleted from the source statement library, along with unwanted macros (e.g., those macros never to be used).

```
// JOB DSERVSLB
// EXEC DSERV
 DSPLYS ALL
/*
// EXEC SSERV
 DSPLY A.DOSCHLV (Display the change level.)
 DSPCH Z.ALL (Display and punch sample problems, delete, and link edit
  books.)
/*
// PAUSE REMOVE CARDS FROM SYSPCH.  PRESS END TO CONTINUE
/&
```

Step 5

Dial the address (unit 190) of Volume 1 (the core image and relocatable system residence disk drive) into the load unit switches, and IPL (press LOAD) to pass control to the DOS supervisor. See IPL Control and ASSGN Statements for System Generation, later in this module, for the IPL control statements. When the system enters the wait state, press START on the card reader. The following cards are in the card reader (SYSRDR/SYSIPT):

```
ADD X'00C',3505
ADD X'00D',3525P
ADD X'00E',3211
ADD X'01F',1050A
ADD X'190',2311
ADD X'191',2311
SET DATE=06/30/72
ASSGN SYSLOG,X'01F'
LCG
ASSGN SYSRDR,X'00C'
ASSGN SYSIPT,X'00C'
ASSGN SYSPCH,X'00D'
ASSGN SYSLST,X'00E'
ASSGN SYSREC,X'191'
```

Step 6

Perform a sorted DSERV to display the directories, and a DELETR to delete any relocatable components never to be used.

Note: The listing can be checked to determine the size of the remaining library.

```
// JOB DSERVRLB
// EXEC DSERV
 DSPLYS ALL
/*
// EXEC MAINT
 DELETR (Delete Desired Components from Relocatable Library)
/*
/&
```

Step 7

Copy the relocatable library to a third initialized disk, defining it as a private relocatable library. During this copying, you can allocate the private library to the desired size. When allocating the size of the

private library, consider the components in the private relocatable library of Volume 3 and the workfile requirements discussed in step 9. To compute the minimum size of a library, see Allocating Library Sizes. For techniques on copying libraries, see Maintenance Procedures earlier in this module.

If a small private relocatable library is built on the operational system, remove the pack from unit 191 and mount the initialized pack that the operational system is built on. Then, repeat this job with the proper EXTENT and NEWVOL statements.

```
// JOB PVTRLB
// ASSGN SYSRLB,X'191'
// DLBL IJSYSRL,'DOS PVT REL LIB',99/365,SD
// EXTENT SYSRLB,nnnnnn,1,1,nnnn,nnnn
// EXEC CORGZ
  NEWVOL RL=nnn(n)
  COPYR ALL
/*
/&
// PAUSE IPL VOLUME 2
```

Step 8

Remove Volume 1 (the core image and relocatable library volume), and mount Volume 2 (the core image and source statement library volume). Dial the address (unit 190) of the system residence disk drive into the load unit switches and IPL (press LOAD) to pass control to the DOS supervisor. When the system enters the wait state, press START on the card reader. These cards are in the card reader (SYSRDR/SYSIPT):

```
ADD X'00C',3505
ADD X'00D',3525P
ADD X'00E',3211
ADD X'01F',1050A
ADD X'190',2311
ADD X'191',2311
ADD X'181',3420T9
SET DATE=06/30/72
ASSGN SYSLOG,X'01F'
LOG
ASSGN SYSRDR,X'00C'
ASSGN SYSIPT,X'00C'
ASSGN SYSPCH,X'00D'
ASSGN SYSLST,X'00E'
ASSGN SYS001,X'191'
ASSGN SYS002,X'191'
ASSGN SYS003,X'191'
ASSGN SYSREC,X'191'
```

Step 9

Define assembler workfiles for SYS001, SYS002, and SYS003 to the second drive. The workfiles SYS001, SYS002, and SYS003 are defined by use of the DLBL and EXTENT cards. These cards must be preceded by the OPTION STDLABEL or OPTION PARSTD card. If these workfiles are on the same pack as the private relocatable library copied in Step 7, include the same DLBL and EXTENT cards for SYSRLB in this step (Step 9).

Note: Because it is difficult to determine the workfile allocations, it is best to allocate an entire pack.

```
// JOB ASSEMSUP
// OPTION STDLABEL
// DLBL IJSYS01,'SYSTEM WORK FILE NO. 1',99/365,SD
// EXTENT SYS001,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS02,'SYSTEM WORK FILE NO. 2',99/365,SD
// EXTENT SYS002,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS03,'SYSTEM WORK FILE NO. 3',99/365,SD
// EXTENT SYS003,nnnnnn,1,n,nnnn,nnnn
```

Step 10

Assemble the supervisor.

```
// EXEC ASSEMBLY
         SUPVR MPS=YES
         CONFG MODEL=135,DEC=YES,FP=YES
         STDJC LISTX=YES,LINES=46
         FOPT IT=BG,PC=YES,OC=YES,CCHAIN=YES,DASDFP=(1,1,2311),       X
              SYSFIL=YES,TEBV=CR,PCII=YES,TOD=YES,ZONE=(WEST,5)
         PIOCS CHANSW=RWTAU,TAPE=7
         ALLOC F1=14K,F2=14K
         IOTAB JIB=10,CHANQ=10,F1PGR=8,F2PGR=8,IODEV=10,BGPGR=10
         DVCGEN CHUN=X'00C',DVCTYP=3505
         DVCGEN CHUN=X'00D',DVCTYP=3525P
         DVCGEN CHUN=X'00E',DVCTYP=3211
         DVCGEN CHUN=X'01F',DVCTYP=1050A
         DVCGEN CHUN=X'190',DVCTYP=2311
         DVCGEN CHUN=X'191',DVCTYP=2311
         DVCGEN CHUN=X'180',DVCTYP=3420T7,CHANSW=YES
         DVCGEN CHUN=X'181',DVCTYP=3420T9,CHANSW=YES
         ASSGN SYSRDR,X'00C'
         ASSGN SYSIPT,X'00C'
         ASSGN SYSPCH,X'00D'
         ASSGN SYSLST,X'00E'
         ASSGN SYSLOG,X'01F'
         ASSGN SYSREC,X'191'
         ASSGN SYSLNK,X'191'
         ASSGN SYS001,X'191'
         ASSGN SYS002,X'191'
         ASSGN SYS003,X'191'
         ASSGN SYS004,X'191'
         SEND 16384
         END
/*
* CHECK ASSEMBLY LISTING FOR ERRORS.  IF CORRECT
* REMOVE ASSEMBLED SUPERVISOR FROM SYSPCH.  INSERT IN READER
* FOLLOWING THE INCLUDE CARD OF JOB CATALSUP (IN STEP 13).
// PAUSE TO CONTINUE PRESS END
/&
```

Note: The operator needs to know the address of label CNLSVE in order to obtain a "special purpose dump". As he normally does not have access to the supervisor assembly listing, you should locate the address in this listing and ask him to write it down under the heading "Special Purpose Dump" (Reference 10) in DOS Version 4 Messages, GC33-5009.

Step 11

Delete unwanted macros from the source statement library, and then create a private source statement library. If space allows, this copy could be made to a free area of the pack to which the relocatable library was copied. For techniques on creating a new library, see Maintenance Procedures.

```
// JOB DELETSL
// EXEC MAINT
   (include DELETS statements for those macro definitions that are not wanted on
   the private source statement library.All DELETS statements are included in the
   Z.DELETESL book that was displayed and punched in step 4.)
/*
/&
// PAUSE CHECK YOUR ALLOCATIONS AGAINST SYSTEM DIRECTORY.
// JOB PVTSLB
// ASSGN SYSSLB,X'191'
// DLBL IJSYSSL,'DOS PVT SRC LIB',99/365,SD
// EXTENT SYSSLB,nnnnnn,1,1,nnn,nnn
* CREATE PRIVATE SOURCE STATEMENT LIBRARY ON UNIT 191
// EXEC CORGZ
   NEWVOL SL=nnn(n)
   COPYS ALL
/*
/&
```

Step 12

This step defines the creation of the system core image library of your operational pack. You can either delete the source statement library and reallocate the system to create a large core image library, or reallocate to create a large core image and a small system source statement library on the operational pack. For techniques on creating a new library, see Maintenance Procedures. The number of cylinders allocated to CL must be at least two less than EXTENT for SYSRES.

```
// JOB SYSRES
// EXEC MAINT
 DELETS A.ALL,Z.ALL
/*
// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',99/365,SD
// EXTENT SYSRES,111111,1,,0001,nnn9
// EXEC MAINT
   ALLOC CL=nnn(n),SL=0(0)
/*
/&
// JOB LABELS
// OPTION STDLABEL
// DLBL IJSYSLN,'SYSTEM WORK FILE NO.  0',99/365,SD
// EXTENT SYSLNK,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS01,'SYSTEM WORK FILE NO.  1',99/365,SD
// EXTENT SYS001,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS02,'SYSTEM WORK FILE NO.  2',99/365,SD
// EXTENT SYS002,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS03,'SYSTEM WORK FILE NO.  3',99/365,SD
// EXTENT SYS003,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYSCL,'DOS PVT COR LIB',99/365,SD
// EXTENT SYSCLB,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYSRL,'DOS PVT REL LIB',99/365,SD
// EXTENT SYSRLB,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYSSL,'DOS PVT SRC LIB',99/365,SD
// EXTENT SYSSLB,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYSRC,'DOS RECORDER FILE',99/365,SD
// EXTENT SYSREC,nnnnnn,1,1,nnnn,nnnn
/*
/&
```

Step 13

During this step you must be IPLed from the operational pack that you are building and have your relocatable library on line. If linkage edit workfiles are not assigned, they must be assigned now.

Linkage edit and catalog the assembled supervisor (object module from step10) to the core image library. If the SEND address is larger than the one used by the supervisor being replaced, certain key programs must also be linkage edited and cataloged to the core image library in the same job step with the new supervisor. These key programs are the librarian programs CORGZ, RSERV, and SSERV. The LINKEDIT deck punched out in step 4 contains all of the necessary control statements to linkage edit all IBM components shipped on the system. If the SEND address is not exceeded, only the supervisor need be cataloged. The new supervisor cannot be retrieved until after the /& is read. For any program being cataloged into the core image library (CIL), the system places an entry in the CIL directory. In addition, for any program beginning with one of the special prefixes $, $$A, $$B, $$BO, or FGP, the system also places an entry in the appropriate subdirectory (if space is available) after /& is read. The supervisor entry is placed in the transient (sub)directory even if it must overlay the last entry on the track.

When retrieving one of these special prefix programs, the system first searches the appropriate subdirectory. If it does not find an entry for that program, the system then searches the CIL directory. Suppose in this job, the supervisor is cataloged, but the job terminates before /& is reached. The CIL directory contains an entry for the new supervisor (and entries for the other cataloged programs), but the transient directory still contains an entry for the old supervisor. If you IPL at this point, the system merely reloads the old supervisor. To recover from this situation, rerun the job with at least one special prefix program and all the programs in the steps that were not successfully completed. Including the special prefix program causes the subdirectories to be updated (rewritten). To re-link edit and catalog more than described here uses

additional core image library space unnecessarily until a condense is performed.

Do not attempt any other operations until the supervisor and the other programs in this job are cataloged, /& is reached, and the subsequent IPL is performed.

Step 13 contains the coding for including the IBM system service and control programs. The LINKEDIT book displayed and punched in step 4 contains all the necessary job control statements to linkage edit any of the IBM components shipped on the system. You are encouraged to use this book to tailor a job stream to include any IBM components desired.

```
// JOB CATALSUP
// ASSGN SYSRLB,X'191'
// ASSGN SYSLNK,X'191'
// OPTION CATAL
   ACTION CLEAR
 INCLUDE
    (Supervisor object deck here.)
/*
// EXEC LNKEDT
 INCLUDE IJBSL3          RSERV
// EXEC LNKEDT
 INCLUDE IJBSL4          SSERV
// EXEC LNKEDT
 INCLUDE IJBSL5          CORGZ
// EXEC LNKEDT
 PHASE LSERV,S or +0 (Note)
 INCLUDE IJBLSERV
// EXEC LNKEDT
/&
```

Note: S is required for a non-MPS system and +0 for an MPS system.

Step 14

At this point the system indicates that re-IPL is needed. IPL from SYSRES (190).

Re-IPL, issue the SET statement and create the recorder file.

Linkage edit and catalog any additional components desired to the core image library. See the section for each IBM component for its cataloging control statements. Before the next step is performed, perform an LSERV to display the label cylinder, check the linkage editor listings, and make all necessary corrections.

You may delete unwanted transients by executing selected jobs from the Z.DELETECL book, punched in step 4.

```
 SET
 Put your DELETEC job stream here, if desired.
/*
/&
ASSGN SYSRLB,X'191'
ASSGN SYSSLB,X'191'
// JOB CATALCLB
// OPTION CATAL
 (Linkage edit desired components and user programs to the core
 image library.)
// EXEC LNKEDT
/*
/&
```

Step 15

Perform all necessary assemblies. (If System/370 instructions are to be assembled, Assembler D (14K) must be present in the core image library.) The assemblies for components should be performed as separate jobs. For the information required to assemble Emulator Programs, refer to the Emulator Program manuals listed in the Preface. Be careful to keep all assemblies in order.

Assemble all your required IOCS modules. By assigning SYSPCH to a tape unit, the IOCS modules can be cataloged to the relocatable library without punching them on cards. The IOCS modules required by compilers as defined in the section Compiler IOCS modules are supplied in the relocatable library by IBM.

After you have assembled your programs, close the tape assigned to SYSPCH and reassign SYSPCH to its permanent assignment by using the close command. The assembly listings should be checked for errors before proceeding.

Now assemble the IOCS modules to be cataloged to the relocatable library. Unit 281 must be unassigned. Modules should be assembled on an as-required basis. The following is an example:

```
// JOB ASSEM
// OPTION DECK,LIST,LOG
// ASSGN SYSPCH,X'281'
// EXEC ASSEMBLY
         CDMOD RECFORM=FIXUNB,CTLCHR=ASA,TYPEFLE=CUTPUT,ICAREA2=YES,   X
               DEVICE=3525,SEPASMB=YES
 END
/*
// EXEC ASSEMBLY
         MTMOD RECFORM=FIXUNB,READ=FORWARD,CKPTREC=YES,SEPASMB=YES
 END
/*
CLOSE SYSPCH,X'00D'
/&
* CHECK ASSEMBLY LISTINGS FOR ERRORS.  RELOAD 281 WITH THE SAME TAPE.
// PAUSE IF CORRECT,PRESS END TO CONTINUE.
```

Step 16

Reload the tape that was assigned to SYSPCH in step 15 and assign it to SYSIPT. With this tape the MAINT program catalogs the IOCS modules to the relocatable library by the control card // EXEC MAINT. You may set new standard labels (OPTION STDLABEL), reallocate library sizes, and set automatic condense limits, if required. Backup for the operational disk can be obtained by copying the operational disk pack to tape by using the copy disk-to-tape utility program.

System generation for Volume 1 and 2 users is complete. You may set new standard label (OPTION STDLABEL), reallocate library sizes, and set automatic condense limits, if required. The recorder file (IJSYSRC) should then be created again (see Recorder File (SYSREC).

```
// JOB CATALRLB
// ASSGN SYSIPT,X'281'
* CATALOG MODULES TO RELOCATABLE LIBRARY
// EXEC MAINT
/*
* THE FOLLOWING DELETIONS ARE OPTIONAL.
// PAUSE PLACE REMAINING CARDS IN READER.  PRESS END TO CONTINUE.
// EXEC MAINT
    (Place any desired deletes here.)
 CONDS RL,SL
/*
/&
```

If you desire components from Volume 3, proceed to System Generation Procedures for Volume 3 Users.

Note: A copy of the new operational pack should be obtained for backup after the system generation process is completed.

Step 17

Create the desired private core image library or libraries. You may use a private core image library for expansion of the system core image library. In systems supporting the batched-job foreground option, create a private core image library for each batched-job foreground partition desired. You may set standard labels (OPTION STDLABEL or OPTION PARSTD) if required.

```
// JOB PVTCLB
// ASSGN SYS003,X'191'
// DLBL IJSYSPC,'DOS PVT CORE IMAGE LIB',99/365,SD
// EXTENT SYS003,nnnnnn,1,1,nnnn,nnnn
// EXEC CORGZ
  NEWVOL CL=nnn(nn)
/*
/&
```

Now the desired components may be link-edited and cataloged into the private core image library or libraries. If it is desired to link-edit a non-relocatable IBM-supplied program for execution in a foreground partition (if enough core storage is available in which to execute the link-edited program), start the desired partition and execute the linkage editor in that partition using the same input as for the background partition.

One IBM 2311 Disk Drive Example

Figure 1-21 illustrates the general procedure followed by users with a single 2311 disk drive available.

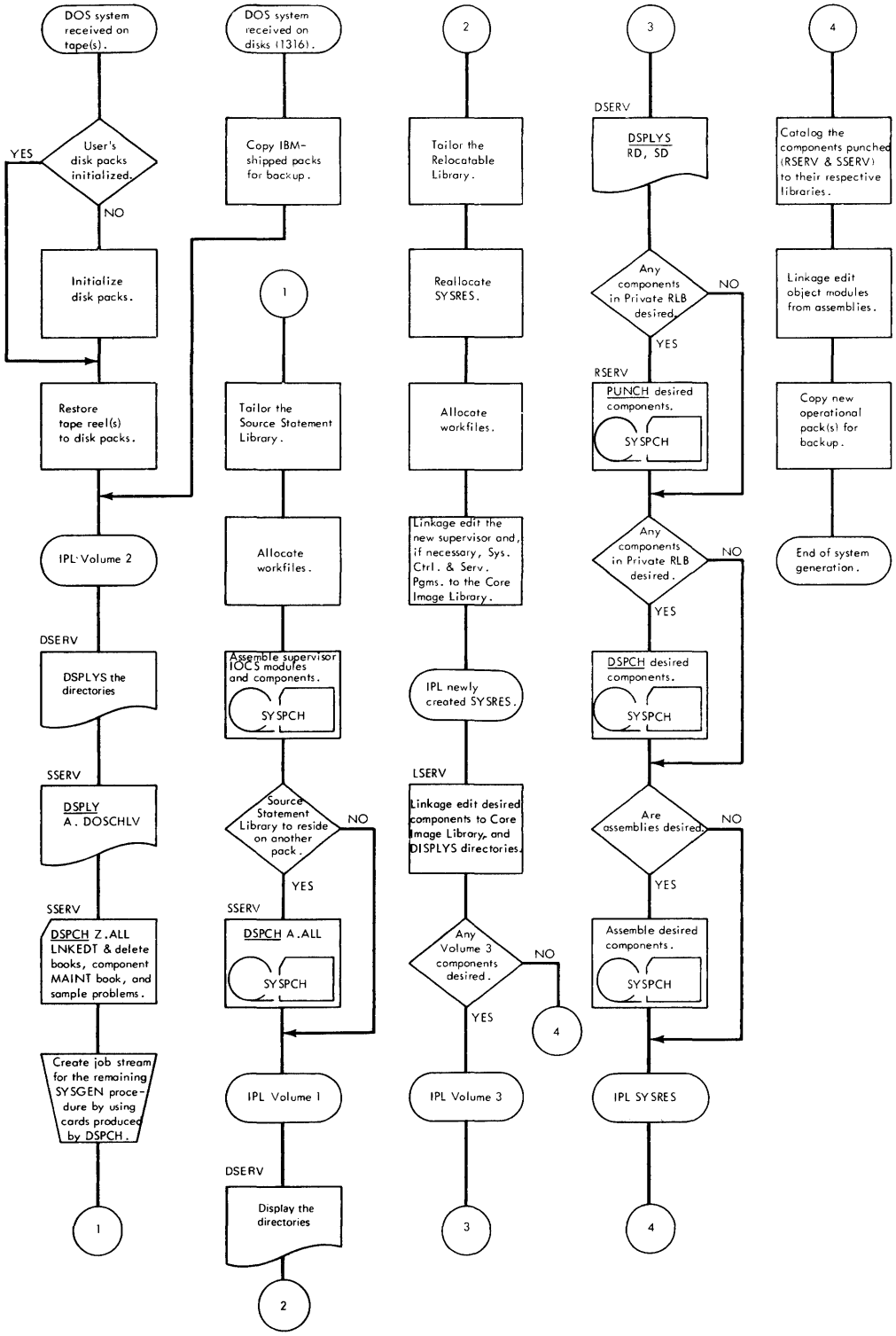


Figure 1-21. PROCEDURES--Overview of Single Disk System Generation

IBM supplies the 14K disk workfile assembler in the core image library of Volume 2. The following procedure employs this assembler variant for system generation. To use another variant:

- Linkage edit and catalog the desired assembler to the core image library of the relocatable library system.
- CSERV the assembler just cataloged.
- Delete the supplied assembler from the core image library of the source statement library system, and condense this library.
- Catalog the assembler obtained through the previous CSERV to the core image library of the source statement library volume.

Now system generation as described can be performed. Figure 1-22 illustrates the system configuration upon which the One IBM 2311 Disk Drive Example is based.

INPUT/OUTPUT DEVICE CONFIGURATION FOR SINGLE DISK EXAMPLE (FIGURE 1-22)

Device	Channel	Unit	Use
2540R	0	0C	Card Reader (SYSRDR/SYSIPT)
2540P	0	0D	Card Punch (SYSPCH)
1403	0	0E	Printer (SYSIST)
3210/3215	0	1F	Console Printer-Keyboard (SYSLOG)
2401T7	2	80	Magnetic Tape with data conversion feature (switchable to channel 1)
2311	1	90	Disk (SYSRES, SYSLNK, SYSREC, SYS001, SYS002, SYS003)

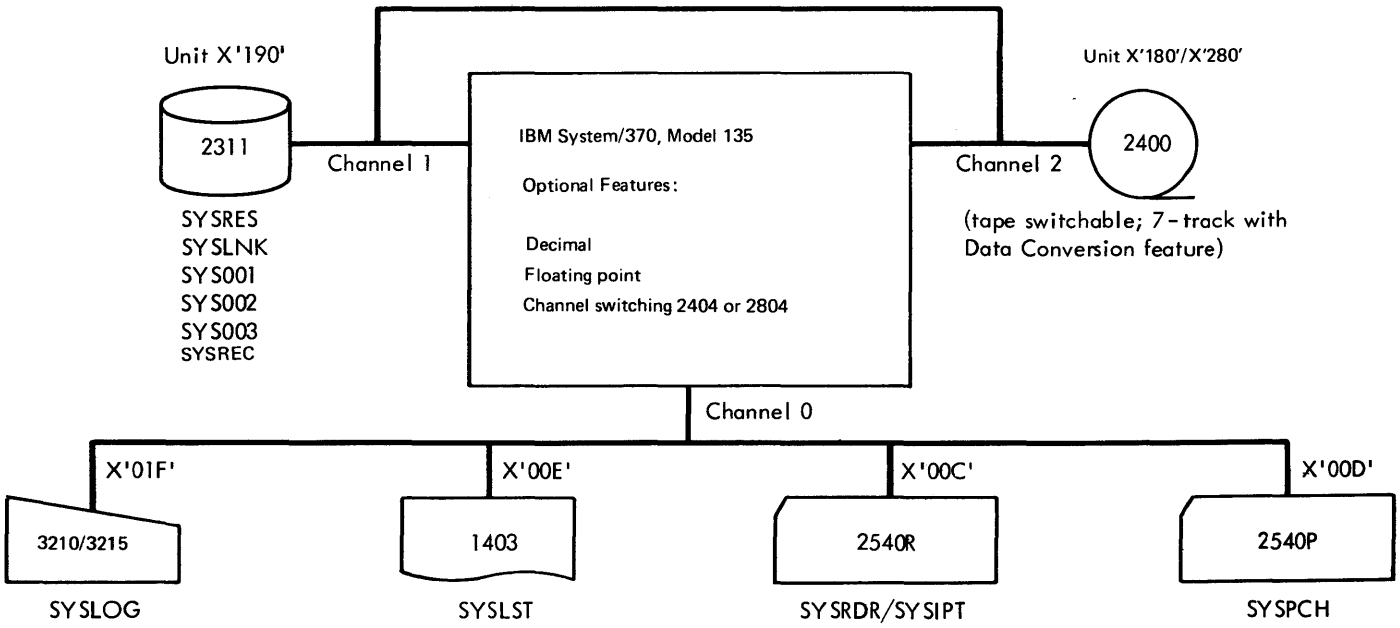


Figure 1-22. PROCEDURES--System Configuration for the One IBM 2311 Disk Drive Example

SINGLE DISK EXAMPLE: RESULTS OF SYSTEM GENERATION

When system generation is completed for single disk example, the operational pack contains user selected components and programs, together with the installation's tailored supervisor, job control, linkage editor, and librarian programs in the core image library. The tape shipped by IBM is retained as a backup tape. The relocatable library contains all those components shipped by IBM except those that were deleted in steps 10 and 14. There is no source statement library on this operational pack. This is optional, and was indicated as such in the example. You may either utilize the source statement library shipped by IBM on the source statement library volume, or build a source statement library on the operational pack by allocating space and cataloging the desired macro definitions.

DISK-ONLY USERS

Disk-only users receive the 2311 system on 1316 disk packs.

- Mount Volume 2 (the IBM-supplied core image library and source statement library pack) and proceed to step 3.

TAPE USERS

Disk users with at least one tape unit available receive the 2311 system on one or two reels of magnetic tape. This IBM-supplied tape is capable of restoring the 2311 system on 1316 disk packs.

- Mount the IBM-supplied tape and a disk pack and proceed as follows:

Step 1

Before restoring the tape, take the necessary action to perform or bypass the initialize routine.

a. If the disk pack has not been initialized, the IBM-supplied volume, a self-loading tape, is capable of initializing the pack. Any volume serial number used in the EXTENT cards for the following steps must agree with the volume serial number in the VOL card for the initialize disk control card.

If initialization is required, mount a pack on unit 190. Mount the IBM-supplied tape on unit 280 (7-track drive). Place the following cards in the card reader. Dial unit 280 in the load unit switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. To initialize additional packs rewind the IBM-supplied tape and repeat the procedure for each pack. The multipart forms of the linkage editor maps and the supervisor listing are required by your IBM customer engineer and your system programmers for maintenance purposes.

```
// LOG
// JOB INTDSK
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSOPT,X'190',D1
// EXEC
// UID IA
// VTOC STRTADR=(0199000),EXTENT=(10)
VOL1111111
// END
```

b. If the disk pack has been previously initialized, the initialize disk routine can be bypassed. Any volume serial number used in the EXTENT cards for the following steps must agree with the volume serial number used when the pack was initialized.

To bypass the initialize disk program on the IBM-supplied tape, mount the IBM-supplied tape on unit 280 (7-track drive). Mount an initialized disk on unit 190 (VTOC on CYL 199). Insert the following control cards into the reader. Dial 280 in load unit switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. The job is complete when the 00C and 4000A messages are logged.

```
// LOG
// JOB INTDSK
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSIPT,X'280',T1
// FILES SYSIPT,1
```

Step 2

After step 1 is completed, restore the systems from the self-loading tape to the initialized packs. This step is omitted when a system is received on disk.

Do not rewind unit 280. To load the tape on the disk, place the following cards in the reader. Dial 280 in the load switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. See Restoring the IBM-Supplied System Tape-to-Disk for operating procedures.

```
// JOB DISRST
// DATE 72182
// ASSGN SYS000,X'190',D1
// ASSGN SYS001,X'190',D1
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSLST,X'00E',L1
// EXEC
```

Step 3

Dial 190 (the address of the unit containing Volume 2) in the load unit switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader to pass control to the DOS supervisor). See IPL Control and ASSGN Statements for System Generation, later in this module, for the IPL control statements. The following cards are in the card reader.

```
ADD X'00C',2540R
ADD X'00D',2540P
ADD X'00E',1403
ADD X'01F',1050A
ADD X'190',2311
ADD X'180',2400T7
SET DATE=06/30/72
ASSGN SYSLOG,X'01F'
LOG
ASSGN SYSRDR,X'00C'
ASSGN SYSIPT,X'00C'
ASSGN SYSPCH,X'00D'
ASSGN SYSLST,X'00E'
ASSGN SYS001,X'190'
ASSGN SYS002,X'190'
ASSGN SYS003,X'190'
ASSGN SYSREC,X'190'
SET RF=CREATE
```

Step 4

Perform a sorted DSERV to display the directories. SSERV (display and punch) all the desired sample problems from the source statement library into cards. Punched output includes the sample problems with BKEND and CATALS cards. Four other books in the source statement library that should be punched out at this time (Z.LINKEDIT, Z.DELETECL, Z.DELETERL, and Z.DELETESL) contain the necessary control statements to selectively linkage edit and delete all IBM components. The sample problem program names, the linkage edit, and the delete book names can be chosen and punched using the DSPCH statements.

```
// JOB DSERVSLB
// EXEC DSERV
   DSPLYS ALL
/*
// EXEC SSERV
   DSPCH Z.ALL
/*
/&
* REMOVE SAMPLE PROBLEMS, DELETE BOOKS, AND THE LINK EDIT
// PAUSE  BOOK CARDS FROM SYSPCH, PRESS END TO CONTINUE.
```

Step 5

After these sample problems and books have been punched, they can be deleted from the source statement library along with unwanted macros (e.g., those components never to be used).

```
// JOB DELETE
// EXEC MAINT
   DELETS Z.ALL
   (Delete other unwanted components from the source statement library)
/*
// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',99/365,SD
// EXTENT SYSRES,111111,1,n,nnnn,nn9
// EXEC MAINT
   ALLOC CL=nn(n),SL=nn(n)
/*
/&
```

Step 6

To obtain space needed for workfiles to perform assemblies, (SYS001, SYS002 and SYS003) reallocate the system to take advantage of the space gained by the deletes from the previous step. The workfiles SYS001, SYS002, and SYS003 are defined by use of the DLBL and EXTENT cards. These cards must be preceded by the OPTION STDLABEL or OPTION PARSTD card. SYSRES now contains the original core image library and a tailored source statement library.

```
// JOB ASSEM
// OPTION STDLABEL
// DLBL IJSYS01,'SYSTEM WORK FILE NO. 1',99/365,SD
// EXTENT SYS001,111111,8,n,nnn0,nnn,2
// DLBL IJSYS02,'SYSTEM WORK FILE NO. 2',99/365,SD
// EXTENT SYS002,111111,8,n,nnn3,nnn,5
// DLBL IJSYS03,'SYSTEM WORK FILE NO. 3',99/365,SD
// EXTENT SYS003,111111,8,n,nnn6,nnn,9
```

Step 7

Assemble the supervisor.

```
// EXEC ASSEMBLY
         SUPVR ERRLOG=YES
         CONFG MODEL=135,DEC=YES,FP=YES
         STDJC LISTX=YES,LINES=46
         FOPT IT=BG,PC=YES,OC=YES,TEBV=IR,TOD=YES,ZONE=(WEST,5)
         PIOCS CHANSW=RWTAU,TAPE=7
         IOTAB IODEV=10,JIB=8,CHANQ=6,D2311=1,D2400=1
         DVCGEN CHUN=X'00C',DVCTYP=2540R
         DVCGEN CHUN=X'00D',DVCTYP=2540P
         DVCGEN CHUN=X'00E',DVCTYP=1403
         DVCGEN CHUN=X'01F',DVCTYP=1050A
         DVCGEN CHUN=X'190',DVCTYP=2311
         DVCGEN CHUN=X'180',DVCTYP=24007,CHANSW=YES,MODE=X'90'
         ASSGN SYSRDR,X'00C'
         ASSGN SYSIPT,X'00C'
         ASSGN SYSPCH,X'00D'
         ASSGN SYSLST,X'00E'
         ASSGN SYSLOG,X'01F'
         ASSGN SYSREC,X'190'
         ASSGN SYSLNK,X'190'
         ASSGN SYS001,X'190'
         ASSGN SYS002,X'190'
         ASSGN SYS003,X'190'
         SEND 14336
         END
/*
* CHECK ASSEMBLY LISTINGS FOR ERRORS.  IF CORRECT
* REMOVE ASSEMBLED SUPERVISOR FROM SYSPCH, INSERT IN READER
* FOLLOWING THE INCLUDE CARD IN STEP 13
/&
```

Note: The operator needs to know the address of label CNLSVE in order to obtain a "special purpose dump". As he normally does not have access to the supervisor assembly listing, you should locate the address in this listing and ask him to write it down under the heading "Special Purpose Dump" (Reference 10) in DOS Version 4 Messages, GC33-5009.

Step 8

Perform all necessary assemblies. The assemblies for components should be performed as separate job steps. Be careful to keep all assemblies in order.

If the operational system contains a minimum source statement library with selected macros, obtain these macros through an SSERV. The macros obtained in this step will later be cataloged to a system source statement

library. Remove Volume 2 (the core image and source statement library pack), and replace it with Volume 1 (the core image and relocatable library pack). The pack removed should be retained for backup if the system was received on disk packs; otherwise, the tape can be retained for backup.

```
// JOB ASSEM2
// OPTION DECK,LIST,LOG
// ASSGN SYSPCH,X'280'
// EXEC ASSEMBLY
         CDMOD RECFORM=FIXUNB,CONTROL=YES,TYPEFLE=INPUT,DEVICE=2540,   X
               SEPASMB=YES
         END
/*
// EXEC ASSEMBLY
         MTMOD RECFORM=VARUNB,CKPTREC=YES,WORKA=YES,SEPASMB=YES
         END
/*
CLOSE SYSPCH,X'00D'
/&
// PAUSE
```

Step 9

Dial the address (unit 190) of the system residence disk drive into the load unit switches, and IPL (press LOAD) to pass control to the DOS supervisor. When the system enters the wait state, press START and EOF on the card reader. The following cards are in the card reader.

```
ADD X'C0C',2540R
ADD X'00D',2540P
ADD X'C0E',1403
ADD X'01F',1050A
ADD X'190',2311
ADD X'180',2400T7
SET DATE=06/30/72
ASSGN SYSLOG,X'01F'
LOG
ASSGN SYSRDR,X'00C'
ASSGN SYSIPT,X'00C'
ASSGN SYSPCH,X'00D'
ASSGN SYSLST,X'00E'
ASSGN SYSLNK,X'190'
ASSGN SYS001,X'190'
```

Step 10

Perform a sorted DSERV to display the directories. Delete all relocatable library components not to be used.

```
// JOB DSERVRLB
// EXEC DSERV
 DSPLYS ALL
/*
// EXEC MAINT
 DELETR (Delete desired components from the relocatable library.)
/*
```

Step 11

Reallocate library sizes, assigning all free space with the exception of linkage editor workfiles, to the core image library. If room is available at this time, also allocate the small source statement library to contain those macros punched in step 8. Otherwise, this allocation can be done in step 15.

```
// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE 111111',99/365,SD
// EXTENT SYSRES,111111,1,n,0001,nn9
// EXEC MAINT
   ALLOC CL=nn(n),RL=nn(n)
/*
/&
// JOB LABELS
// OPTION STDLABEL
// DLBL IJSYSLN,'SYSTEM WORK FILE NO. 0',99/365,SD
// EXTENT SYSLNK,nnnnnn,1,n,nnn,nnn
// DLBL IJSYS01,'SYSTEM WORK FILE NO. 1',99/365,SD
// EXTENT SYS001,nnnnnn,1,n,nnn,nnn
// DLBL IJSYSRC,'DOS RECORDER FILE',99/365,SD
// EXTENT SYSREC,nnnnnn,1,1,nnnn,nnnn
/*
/&
```

Step 12

Linkage edit and catalog the assembled supervisor (from step 7) to the core image library. Certain key programs must also be linkage edited and cataloged in the same job with the new supervisor if the SEND address is larger than the one used by the supervisor being replaced. These key programs are the librarian programs CORGZ, RSERV, and SSERV. The control statements to linkage edit these programs are in the linkage edit deck punched out in step 4. The SSERV and assembler components should be included as part of this job only if the operational system is to contain a minimum source statement library.

The new supervisor cannot be retrieved until after the /& is read. For any program being cataloged into the core image library (CIL), the system places an entry in the CIL directory. In addition, for any program beginning with one of the special prefixes $, $$A, $$B, $$BO, or FGP, the system also places an entry in the appropriate subdirectory (if space is available) after /& is read. The supervisor entry is placed in the transient (sub)directory even if it must overlay the last entry on the track.

When retrieving one of these special prefix programs, the system first searches the appropriate subdirectory. If it does not find an entry for that program, the system then searches the CIL directory. Suppose, in this job, the supervisor is cataloged, but the job terminates before /& is reached. The CIL directory contains an entry for the new supervisor (and entries for the other cataloged programs), but the transient directory still contains an entry for the old supervisor. If you IPL at this point, the system merely reloads the old supervisor. To recover from this situation, rerun the job with at least one special prefix program and all the programs in the steps that were not successfully completed. Including the special prefix program causes the subdirectories to be updated (rewritten). To re-link edit and catalog more than described here uses additional core image library space unnecessarily until a condense is performed.

Do not attempt any other operations until the supervisor and the other programs in this job are cataloged, /& is reached, and the subsequent IPL is performed.

```
// JOB CATAL
// OPTION CATAL
  ACTION CLEAR
 INCLUDE
 Supervisor object deck here
/*
// EXEC LNKEDT
 INCLUDE IJBSL3          RSERV
// EXEC LNKEDT
 INCLUDE IJBSL4          SSERV
// EXEC LNKEDT
 INCLUDE IJBSL5          CORGZ
 // EXEC LNKEDT
 INCLUDE IJQD16W         DISK WCRKFILE ASSEMBLER
// EXEC LNKEDT
 PHASE LSERV, S or +0 (Note)
 INCLUDE IJBLSERV
// EXEC LNKEDT
/&

Note:  S is required for a non-MPS system
       and +0 for an MPS system.
```

Step 13

After these key programs have been cataloged, re-IPL, issue the SET statement and create the recorder file.

Linkage edit and catalog any additional components desired to the core image library. See the section for each component for a complete list of control cards for that component. The LINKEDIT deck punched out in step 4 contains all of the necessary control statements to linkage edit all components shipped on the system.

Before the next step is performed, perform an LSERV to display the label cylinder, check the linkage editor listings and make all necessary corrections before deleting the modules from the relocatable library.

```
SET
// JOB CATAL
// OPTION CATAL
* AT THIS POINT LINKAGE EDIT DESIRED COMPONENTS AND PROGRAMS
// EXEC LNKEDT
/*
// PAUSE CHECK LISTING FOR ERRORS,IF CORRECT PRESS END TO CONTINUE
```

Step 14

If the components deleted in step 10 did not provide adequate space for the final allocation of system libraries, a new allocation can be performed by deleting components previously catalged to the core image library. More then one deletion and allocation may be required during linkage edit jobs. If a small source statement library is desired, allocate space for one.

```
// EXEC MAINT
   DELETR Delete the unwanted component programs from the relocatable library:
   CONDS RL
/*
/&
```

Step 15

Catalog the modules assembled in step 8 to the system relocatable library. If your decision was to have a source statement library on this pack, catalog the source statement macros punched in step 8 to the system source statement library.

Set new standard labels, reallocate library sizes, and set automatic condense limits. Backup for the operational system should also be obtained. The recorder file (IJSYSRC) should then be created again. See Recorder File (SYSREC).

```
// JOB CATAL
* RELOAD 280 WITH THE TAPE CREATED IN STEP 8.
* CHECK ASSEMBLY LISTING FOR ERRORS.  IF CORRECT
// PAUSE TYPE END TO CONTINUE
// ASSGN SYSIPT,X'280'
* CATALOG MODULES TO RELOCATABLE LIBRARY
// EXEC MAINT
/*
/&
```

If you want components from Volume 3, go to System Generation Procedures for Volume 3 Users.

Note: After system generation is complete, obtain a copy of the operational pack(s) for backup.

System Generation Procedures for Volume 3 Users

This procedure provides instructions for including Volume 3 into the system generation procedure for both the single-disk user and the multiple-disk user.

The sequence of arrangement of these instructions is not necessarily the sequence that must be followed. Most users who desire components from Volume 3 will not have to execute each of the steps of this procedure. It is your responsibility to choose the steps that you require for your given procedure.

The following assumptions have been made:

- SYSRES is on any pack.
- IPL has been performed.
- SYSLOG, SYSRDR, SYSIPT, and SYSLST have been assigned.

Step 1

Perform a sorted DSERV of SYSSLB and SYSRLB on Volume 3. The DLBL and EXTENT statements are not required if the SYSRES pack is Volume 3, because the label cylinder for SYSRES on Volume 3 contains these statements. SYSSLB defines the private source statement library, and SYSRLB defines the private relocatable library.

```
// JOB DSERVPLB
// PAUSE ASSGN SYSSLB AND/OR SYSRLB TO VOL.3,PRESS END TO CONTINUE.
// DLBL IJSYSSL,'DOS.SYSSLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYSSLB,111111,1,1,nnn,nnn
// DLBL IJSYSRL,'DOS.SYSRLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYSRLB,111111,1,1,nnn,nnn
// EXEC DSERV
   DSPLYS RD,SD
/*
/&
```

Step 2

Merge from Volume 3 to another private library pack. SYS000 defines the private source statement library of Volume 3. SYSSLB defines the private source statement library of another pack. SYS001 defines the private relocatable library of Volume 3. SYSRLB defines the private relocatable library of another pack.

```
// JOB MERGEPRV
// PAUSE ASSIGN SYS000, SYSSLB AND/OR SYS001, SYSRLB
// DLBL IJSYSPS,'DOS.SYSSLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYS000,111111,1,1,nnn,nnn
// DLBL IJSYSSL,'DOS PVT SRC LIB',99/365,SD
// EXTENT SYSSLB,nnnnnn,n,n,nnnn,nnnn
// DLBL IJSYSPR,'DOS.SYSRLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYS001,111111,1,1,nnn,nnn
// DLBL IJSYSRL,'DOS PVT REL LIB',99/365,SD
// EXTENT SYSRLB,nnnnnn,n,n,nnnn,nnnn
// EXEC CORGZ
   MERGE PRV,PRV
   COPYR (Copy desired modules from private relocatable library)
   COPYS (Copy desired books from private source statement
          library)
/*
/&
```

Step 3

Merge from Volume 3 to another SYSRES. SYS000 defines the private source statement library of Volume 3. SYS001 defines the private relocatable library of Volume 3.

```
// JOB MERGERES
// PAUSE ASSIGN SYS000 AND/OR SYS001 TO VOLUME 3
// DLBL IJSYSPS,'DOS.SYSSIB.FILE.VOLUME.3',99/365,SD
// EXTENT SYS000,111111,1,1,nnn,nnn
// DLBL IJSYSPR,'DOS.SYSRLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYS001,111111,1,1,nnn,nnn
// EXEC CORGZ
   MERGE PRV,RES
   COPYR (Copy desired modules(components) from private relocatable library)
   COPYS (Copy desired books(components) from pvt. source statement library)
/*
/&
```

Step 4

RSERV and/or SSERV of Volume 3. Display and punch the desired components from the private libraries of Volume 3. The DLBL and EXTENT statements are not required if the SYSRES pack is Volume 3.

```
// JOB PUNCHPLB
// PAUSE ASSIGN SYSSLB AND/OR SYSRLB TO VOLUME 3 AND SYSPCH TO A TAPE.
// DLBL IJSYSSL,'DOS.SYSSIB.FILE.VOLUME.3',99/365,SD
// EXTENT SYSSLB,111111,1,1,nnn,nnn
// EXEC SSERV
   DSPCH (Punch desired components from private source statement library)
/*
// DLBL IJSYSRL,'DOS.SYSRLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYSRLB,111111,1,1,nnn,nnn
// EXEC RSERV
   PUNCH (Punch desired components from private relocatable library)
/*
// PAUSE CLOSE SYSPCH
/&
```

Step 5

Linkage Edit from Volume 3. Linkage edit the desired components from the private relocatable library of Volume 3 to a core image library of another pack. The DLBL and EXTENT statements are not required for SYSLNK and SYS001 if they have been previously defined as standard labels. The control cards to linkage edit the desired components are available in the Z.LINKEDIT book from the source statement library of Volume 2.

```
// JOB LINKRLB
// PAUSE ASSIGN SYSRLB TO VOLUME 3.  ASSIGN SYSLNK AND SYS001 WORKFILES.
// DLBL IJSYSRL,'DOS.SYSRLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYSRLB,111111,1,1,nnn,nnn
// DLBL IJSYSLN,'SYSTEM WORKFILE NO. 0',99/365,SD
// EXTENT SYSLNK,nnnnnn,n,n,nnnn,nnnn
// DLBL IJSYS01,'SYSTEM WORKFILE NO. 1',99/365,SD
// EXTENT SYS001,nnnnnn,n,n,nnnn,nnnn
// OPTION CATAL
 (Include desired components from private relocatable library)
// EXEC LNKEDT
/*
/&
```

Step 6

Assemble from Volume 3. Programs in the private source statement library of Volume 3 can be assembled by using Assembler D which is in the core image library of Volume 3. If workfiles have to be assigned to Volume 3, they must be defined and additional cylinders can be allocated by having the workfile extents overlay the private relocatable library.

```
// JOB ASSEMPRV
// PAUSE ASSIGN SYSSLB TO VOLUME 3.  ASSIGN WORKFILES SYS001, 2, and 3.
// DLBL IJSYSSL,'DOS.SYSSLB.FILE.VOLUME.3',99/365,SD
// EXTENT SYSSLB,111111,1,1,nnnn,nnnn
// DLBL IJSYS01,'SYSTEM WORKFILE NO. 1',99/365,SD
// EXTENT SYS001,nnnnnn,n,n,nnnn,nnnn
// DLBL IJSYS02,'SYSTEM WORKFILE NO. 2',99/365,SD
// EXTENT SYS002,nnnnnn,n,n,nnnn,nnnn
// DLBL IJSYS03,'SYSTEM WORKFILE NO. 3',99/365,SD
// EXTENT SYS003,nnnnnn,n,n,nnnn,nnnn
// EXEC ASSEMBLY
   (Source deck)
/*
/&
```

Step 7

Catalog the components punched out in Step 4. SYSRLB and SYSSLB must be assigned if the programs on SYSIPT are catalged to private libraries.

```
// JOB CATALOG
// PAUSE ASSIGN SYSRLB AND/OR SYSSLB.  ASSIGN SYSIPT TO TAPE OF PUNCHPLB
// EXEC MAINT
/*
/&
```

IBM 2314 Direct Access Storage Facility Example

Figure 1-23 illustrates the general procedure to be followed by users with a 2314 available.

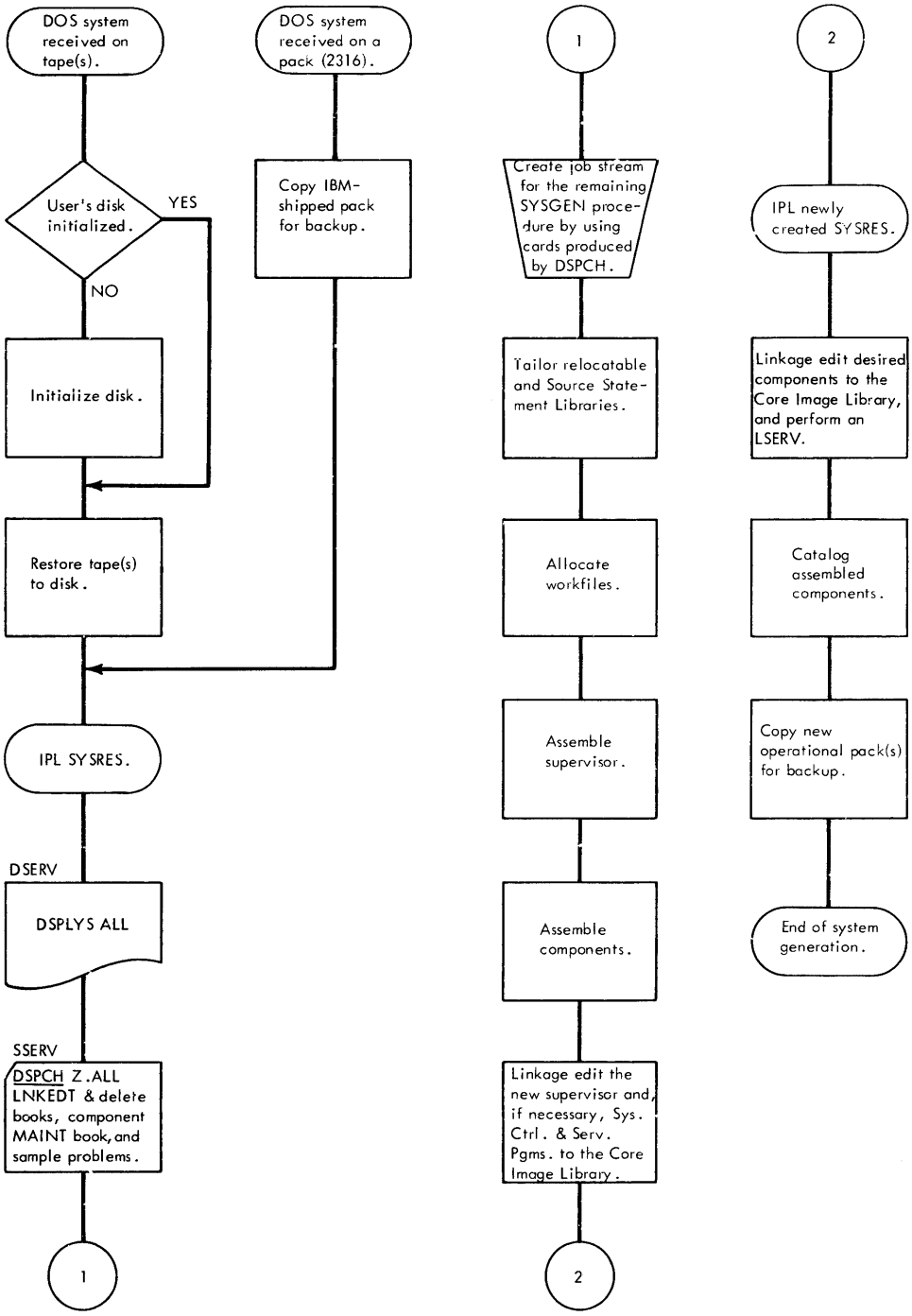


Figure 1-23. PROCEDURES--Overview of a 2314 System Generation

This example is one of the many possible methods for generating a system. Figure 1-24 illustrates the system configuration upon which the 2314 Example is based.

INPUT/OUTPUT DEVICE CONFIGURATION FOR 2314 EXAMPLE (SEE FIGURE 1-24)

Device	Channel	Unit	Use
3505	0	0C	Card Reader (SYSRDR, SYSIPT)
3525P	0	0D	Card Punch (SYSPCH)
3211	0	0E	Printer (SYSLST)
3210/3215	0	1F	Console Printer-Keyboard (SYSLOG)
2314	1	30	Disk (SYSRES, SYSLNK, SYSREC, SYS001, SYS002, SYS003, SYS004)
3420T9	2	80	Magnetic Tape

2314 EXAMPLE: RESULTS OF SYSTEM GENERATION

When system generation is completed, the operational disk of the installation contains user selected components and programs in its core image library, together with the installation's tailored supervisor, job control, linkage editor, and librarian programs. The tape shipped by IBM is retained as backup tape. It is a self-loading tape capable of being restored on disk. The sample problems are punched out during step 4. Libraries contain all modules and macro definitions shipped from IBM. The core image library is built to contain those IBM programs chosen.

The libraries are condensed, and IOCS modules are assembled and cataloged to the relocatable library. When system generation is completed, the sample problems should be run against the operational pack to ensure correct creation of all system programs.

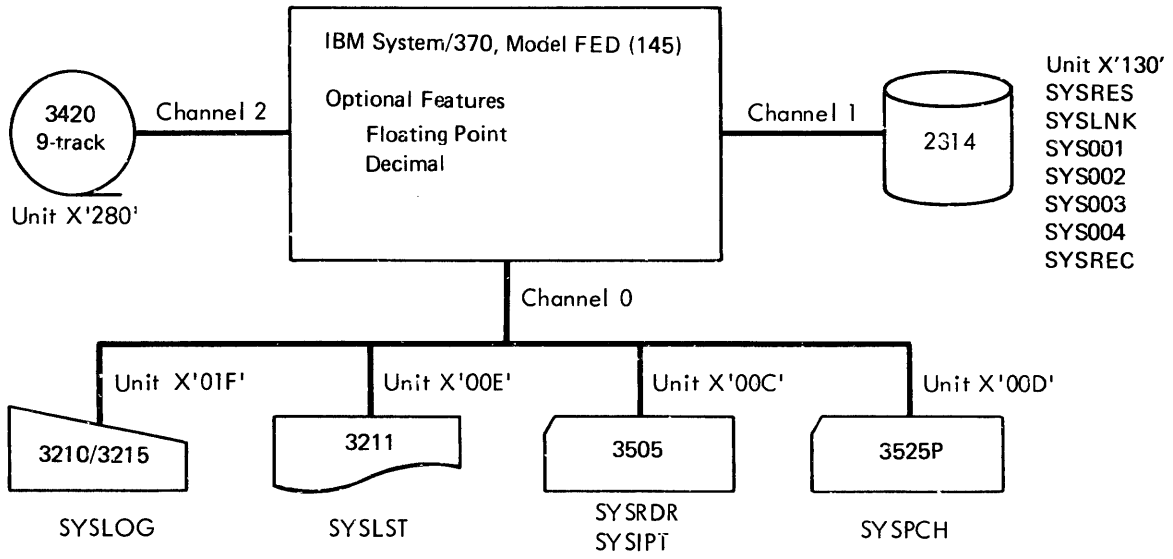


Figure 1-24. PROCEDURES--System Configuration for 2314 Example

IBM 2314 (AND, OPTIONALLY, 2319) ONLY USERS

IBM 2314-only users receive the 2314 system on a 2316 disk pack. This IBM-supplied pack consists of a core image library, a relocatable library, and a source statement library.

- Mount the IBM-supplied volume and proceed to Step 3.

IBM 2314 TAPE USERS

IBM 2314 users with at least one tape unit available receive the 2314 system on magnetic tape (see Figure 78 for distributions). This IBM-supplied tape is capable of restoring the 2314 system to a 2316 disk pack. After the tape has been restored to the 2316 disk pack, the pack contains a core image, a relocatable, and a source statement library.

Before generating a system, you should have at least one initialized disk pack (VTOC on cylinder 199).

Step 1

Mount the IBM-supplied magnetic tape and a disk pack to which the tape is to be restored.

Before restoring the tape, you must take the necessary action to perform or bypass the initialize disk routine.

a. If the disk pack has not been initialized, the IBM-supplied volume, a self-loading tape, is capable of initializing the packs. Any volume serial number used in the EXTENT cards for the following steps must agree with the volume serial number in the VOL card for the initialize disk control cards.

   If initialization is required, mount a pack on unit 130. Mount the IBM-supplied tape on unit 280 (9-track drive). Place the following cards in the card reader. Dial unit 280 in the load unit switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. The multipart forms of the linkage editor maps and the supervisor listing will be required by your IBM customer engineer for maintenance purposes.

```
// JOB INTDSK
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSOPT,X'130',D3
// EXEC
// UID IA
// VTOC STRTADR=(0199000),EXTENT=(20)
VCL1111111
// END
```

b. If the disk pack was previously initialized, the initialize disk routine can be bypassed. Any volume serial number used in the EXTENT cards for the following steps must agree with the volume serial number used when the packs were initialized.

   To bypass the initialize disk program on the IBM-supplied tape, mount the IBM-supplied tape on unit 280 (9-track drive), and mount an initialized disk on unit 130 (VTOC on CYL 199). Insert the following control cards in the reader. Dial unit 280 in load unit switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. The job is complete when the 00C and 4000A messages are logged.

```
// JOB INTDSK
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSIPT,X'280',T2
// FILES SYSIPT,1
```

Step 2

After step 1 is completed, you must restore the system from the self-loading tapes to the initialized pack.

Do not rewind unit 280. To load the tape on disk, place the following cards in the reader. Dial unit 280 in the load switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. See Restore the IBM-Supplied System Tape to Disk for operating procedures.

Note: If two tape units are available, assign the second tape unit for 2 tape distribution to an alternate unit (SYS002).

```
// JOB DISRST
// DATE 72182
// ASSGN SYS000,X'130',D3
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSLST,X'00E',L1
// EXEC
```

Step 3

When the tape is restored, dial the address (unit 130) of the system residence disk drive into the load unit switches, and IPL (press LOAD) to pass control to the DOS supervisor. See IPL Control and ASSGN Statements for System Generation, later in this module, for the IPL control statements. When the system enters the wait state, press START and ECF on the card reader. The following cards are in the card reader (SYSRDR/SYSIPT):

```
ADD X'00C',3505
ADD X'00D',3525
ADD X'00E',3211
ADD X'01F',1050A
ADD X'130',2314
ADD X'280',3420T9
SET DATE=06/30/72
ASSGN SYSLOG,X'01F'
LOG
ASSGN SYSRDR,X'00C'
ASSGN SYSIPT,X'00C'
ASSGN SYSPCH,X'00D'
ASSGN SYSLST,X'00E'
ASSGN SYS001,X'130'
ASSGN SYS002,X'130'
ASSGN SYS003,X'130'
ASSGN SYSLNK,X'130'
ASSGN SYSREC,X'130'
SET RF=CREATE
```

Step 4

Perform a sorted DSERV to display the directories. Then SSERV (display and punch) all the desired sample problems from the source statement library into cards. Punched output includes the sample problems with BKEND and CATALS cards. Four other books in the source statement library that should be punched out at this time (Z.LINKEDIT, Z.DELETECL, Z.DELETERL, and Z.DELETESL) contain the necessary control statements to selectively linkage edit and delete all IBM components. The sample problem program names, and the linkage edit and the delete book names can be chosen and punched into the DSPCH statement(s).

```
// JOB DSERV
// EXEC DSERV
 DSPLYS ALL
/*
// EXEC SSERV
 DSPCH Z.ALL
/*
/&
```

Step 5

Delete never-to-be-used components from the relocatable and source statement libraries.

```
// JOB DELETE
// EXEC MAINT
 DELETR (Unwanted Relocatable Library Components)
 DELETS (Unwanted Source Statement Library Components)
/*
/&
```

Step 6

Define workfiles for SYSLNK, SYS001, SYS002, SYS003, and SYSREC. The workfiles SYSLNK, SYS001, SYS002, SYS003, and SYSREC are defined by use of the DLBL and EXTENT cards. These cards must be preceded by the OPTION STDLABEL or OPTION PARSTD card.

```
// JOB LABEL
// OPTION STDLABEL
// DLBL IJSYSLN,'SYSTEM WORK FILE NO. 0',99/365,SD
// EXTENT SYSLNK,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS01,'SYSTEM WORK FILE NO. 1',99/365,SD
// EXTENT SYS001,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS02,'SYSTEM WORK FILE NO. 2',99/365,SD
// EXTENT SYS002,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYS03,'SYSTEM WORK FILE NO. 3',99/365,SD
// EXTENT SYS003,nnnnnn,1,n,nnnn,nnnn
// DLBL IJSYSRC,'SYSTEM RECORDER FILE',99/365,SD
// EXTENT SYSREC,nnnnnn,1,n,nnnn,nnnn
/&
```

Step 7

Assemble the installation tailored supervisor.

```
// JOB SUPVR
// EXEC ASSEMBLY
         SUPVR MPS=YES
         CONFG MODEL=145,DEC=YES,FP=YES
         STDJC LISTX=YES,LINES=46
         FOPT IT=BG,PC=YES,OC=YES,CCHAIN=YES,DASDFP=(1,1,2314),        X
               SYSFIL=YES,TEB=4,TOD=YES,ZONE=(WEST,5)
         PIOCS TAPE=9
         ALLOC F1=14K,F2=14K
         IOTAB JIB=10,F1PGR=8,F2PGR=8,IODEV=10,BGPGR=10,D2314=6,D3420=2
         DVCGEN CHUN=X'00C',DVCTYP=3505
         DVCGEN CHUN=X'00D',DVCTYP=3525P
         DVCGEN CHUN=X'00E',DVCTYP=3211
         DVCGEN CHUN=X'01F',DVCTYP=1050A
         DVCGEN CHUN=X'130',DVCTYP=2314
         DVCGEN CHUN=X'280',DVCTYP=3420T9
         ASSGN SYSRDR,X'00C'
         ASSGN SYSIPT,X'00C'
         ASSGN SYSPCH,X'00D'
         ASSGN SYSLST,X'00E'
         ASSGN SYSLOG,X'01F'
         ASSGN SYSLNK,X'130'
         ASSGN SYSREC,X'130'
         ASSGN SYS001,X'130'
         ASSGN SYS002,X'130'
         ASSGN SYS003,X'130'
         SEND 16384
 END
/*
* CHECK ASSEMBLY LISTING FOR ERRORS.  IF CORRECT
* REMOVE ASSEMBLED SUPERVISOR FROM SYSPCH.  INSERT IN READER
* FOLLOWING THE INCLUDE CARD OF JOB CATALSUP
// PAUSE TO CONTINUE PRESS END
/&
```

Note: The operator needs to know the address of label CNLSVE in order to obtain a "special purpose dump". As he normally does not have access to the supervisor assembly listing, you should locate the address in this listing and ask him to write it down under the heading "special Purpose Dump" (Reference 10) in DOS Version 4 Messages, GC33-5009.

Step 8

Perform all other necessary assemblies. The component assemblies should be performed as separate jobs. For the information required to assemble Emulator Programs, refer to the Emulator Program manual listed in the Preface. You must be careful to keep all assemblies in order. Assemble all your required IOCS modules. By assigning SYSPCH to a tape unit, the IOCS modules can be cataloged to the relocatable library without punching them on cards. The IOCS modules required by compilers, as defined in Module 4, Compiler IOCS Modules, are supplied in the relocatable library by IBM.

Close the tape assigned to SYSPCH and reassign SYSPCH to its permanent assignment by using the CLOSE command. The assembly listings should be checked for errors before proceeding.

```
// JOB ASSEM
// OPTION DECK,LIST,LOG
// ASSGN SYSPCH,X'280'
// EXEC ASSEMBLY
   CDMOD RECFORM=FIXUNB,CTLCHR=ASA,TYPEFLE=OUTPUT,IOAREA2=YES,     X
         DEVICE=3525,SEPASMB=YES
 END
/*
// EXEC ASSEMBLY
   MTMOD RECFORM=FIXUNB,READ=FORWARD,CKPTREC=YES,SEPASMB=YES
 END
/*
CLOSE SYSPCH,X'00D'
/&
* CHECK ASSEMBLY LISTINGS FOR ERRORS.  RELOAD 280 WITH SAME TAPE.
// PAUSE IF CORRECT PRESS END TO CONTINUE.
```

Step 9

Linkage edit and catalog the assembled supervisor (object module from step 7) to the core image library. If the SEND address is larger than the one used by the supervisor being replaced, certain key programs must also be linkage edited and cataloged to the core image library in the same job step with the new supervisor. These key programs are the librarian programs CORGZ, RSERV, and SSERV. The LINKEDIT deck punched out in step 4 contains all the necessary control statements to linkage edit all IBM components shipped on the system. If the SEND address is not exceeded, only the supervisor is cataloged.

The new supervisor cannot be retrieved until after the /& is read. For any program being cataloged into the core image library (CIL), the system places an entry in the CIL directory. In addition, for any program beginning with one of the special prefixes $, $$A, $$B, $$BO, or FGP, the system also places an entry in the appropriate subdirectory (if space is available) after /& is read. The supervisor entry is placed in the transient (sub)directory even if it must overlay the last entry on the track.

When retrieving one of these special prefix programs, the system first searches the appropriate subdirectory. If it does not find an entry for that program, the system then searches the CIL directory. Suppose, in this job, the supervisor is cataloged, but the job terminates before /& is reached. The CIL directory contains an entry for the new supervisor (and entries for the other cataloged programs), but the transient directory still contains an entry for the old supervisor. If you IPL at this point, the system merely reloads the old supervisor. To recover from this situation, rerun the job with at least one special prefix program and all the programs in the steps that were not successfully completed. Including the special prefix program causes the subdirectories to be updated (rewritten). To re-link edit and catalog more than described here uses additional core image library space unnecessarily until a condense is performed.

Do not attempt any other operations until the supervisor and the other programs in this job are cataloged, /& is reached, and the subsequent IPL is performed.

Note: It may be necessary to reallocate because of the size of the programs (components) desired in the various libraries. This may be accomplished by using the ALLOC statement of either the MAINT or CORGZ librarian programs.

```
// JOB CATALSUP
// OPTION CATAL
  ACTION CLEAR
 INCLUDE
 Supervisor object deck here.
/*
// EXEC LNKEDT
 INCLUDE IJBSL3          RSERV
// EXEC LNKEDT
 INCLUDE IJBSL4          SSERV
// EXEC LNKEDT
 INCLUDE IJBSL5          CORGZ
// EXEC LNKEDT
 PHASE LSERV, S or +0 (Note)
 INCLUDE IJBLSERV
// EXEC LNKEDT
/&

Note:  S is required for a non-MPS system
       and +0 for an MPS system.
```

Step 10

Re-IPL and issue the SET statement.

Linkage edit and catalog any additional components desired to the core image library. See the appropriate module for each component for a complete list of control cards for the components to be cataloged. Before the next step is performed, perform an LSERV to display the label cylinder, check the linkage editor listings, and make all necessary corrections.

```
 SET
// JOB CATALCLB
// OPTION CATAL
* AT THIS POINT LINKAGE EDIT DESIRED COMPONENTS
// EXEC LNKEDT
/*
/&
```

Step 11

Reload the tape that was assigned to SYSPCH in step 8 and assign it to SYSIPT. With this tape the MAINT program catalogs the IOCS modules to the relocatable library by the control card // EXEC MAINT.

```
// JOB CATALRLB
// ASSGN SYSIPT,X'280'
* CATALOG MODULES TO RELOCATABLE LIBRARY
// EXEC MAINT
/*
* THE FOLLOWING DELETIONS ARE OPTIONAL.
// PAUSE PLACE REMAINING CARDS IN READER.  PRESS EOB TO CONTINUE
// EXEC MAINT

Place any desired deletes here.
 CONDS RL,SL
/*
/&
// PAUSE EOJ SYSGEN
```

The basic SYSGEN is complete. You may set new standard labels, re-allocate library sizes by using the copy function (CORGZ) and set automatic condense limits if required. The recorder file (IJSYSRC) should then be created again. See Recorder File (SYSREC). Copy the operational pack for backup.

IBM 3330 Disk Storage Example

Figure 1-25 illustrates the general prccedure tc be fcllcwed by users with a 3330 which is to be used as the system residence volume.

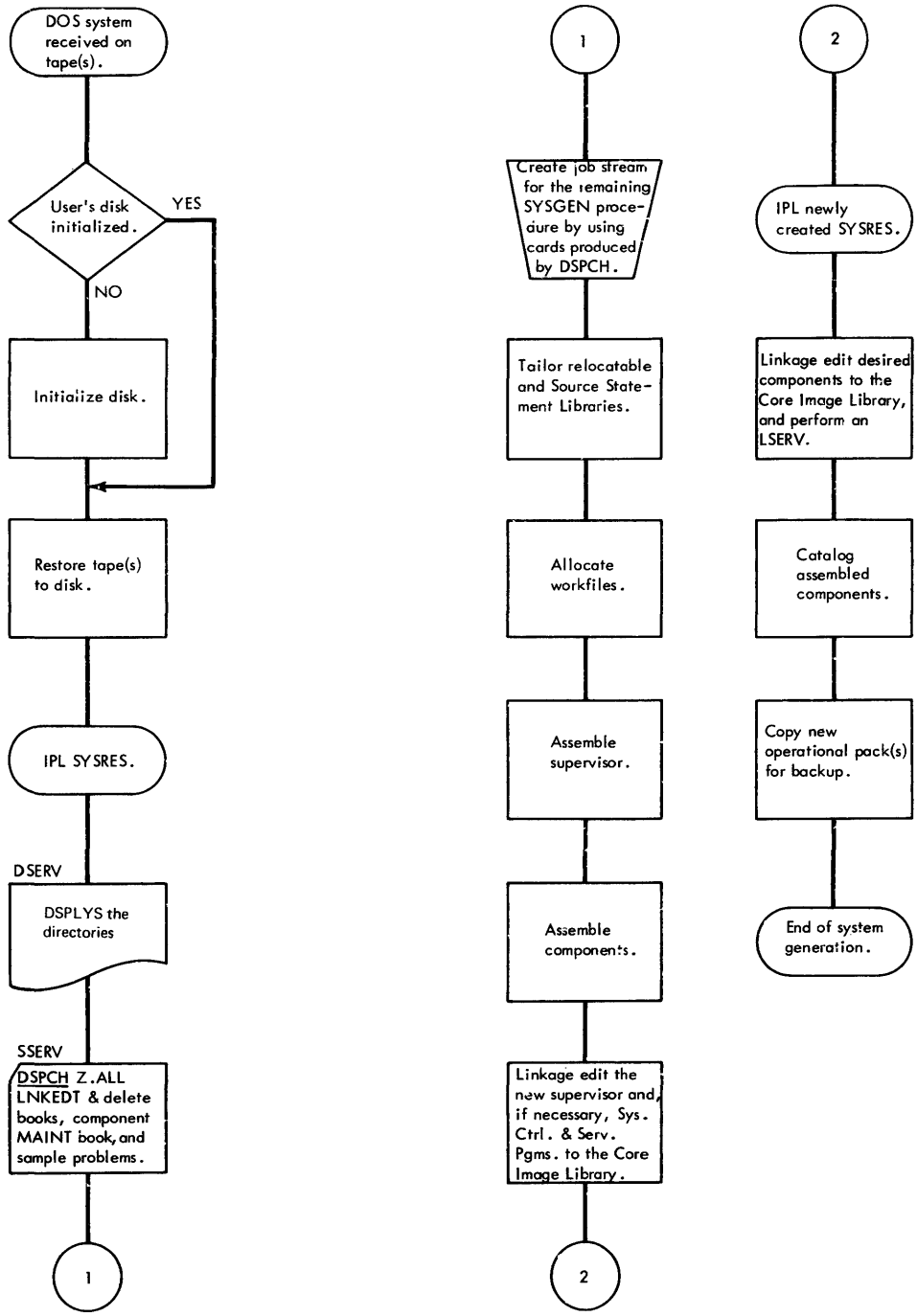


Figure 1-25 PROCEDURES--Overview of a 3330 System Generaticn

This example is one of the many possible methods fcr generating a system. Figure 1-26 illustrates the system ccnfiguraticn upcn which the 3330 example is based.

INPUT/OUTPUT DEVICE CONFIGURATION FOR 3330 EXAMPLE (FIGURE 1-26)

Device	Channel	Unit	Use
2540R	0	0C	Card Reader (SYSRDR, SYSIPT)
2540P	0	0D	Card Punch (SYSPCH)
3211	0	0E	Printer (SYSLST)
3210/3215	0	1F	Console Printer-Keyboard (SYSLOG)
3330	2	60	Disk (SYSRES, SYSLNK, SYSREC, SYS001, SYS002, SYS003, SYS004)
2400T9	1	80	Magnetic Tape

3330 EXAMPLE: RESULTS OF SYSTEM GENERATION

When system generation is completed, the operational disk of the installation contains user selected components and programs in its core image library, together with the installation's tailored supervisor, job control, linkage editor, and librarian programs. The tape shipped by IBM is retained as backup tape. It is a self-loading tape capable of being restored on disk. The sample problems are punched out during step 4. Libraries contain all modules and macro definitions shipped from IBM. The core image library is built to contain those IBM programs chosen.

The libraries are condensed, and IOCS modules are assembled and cataloged to the relocatable library. When system generation is completed, the sample problems should be run against the operational pack to ensure correct creation of all system programs.

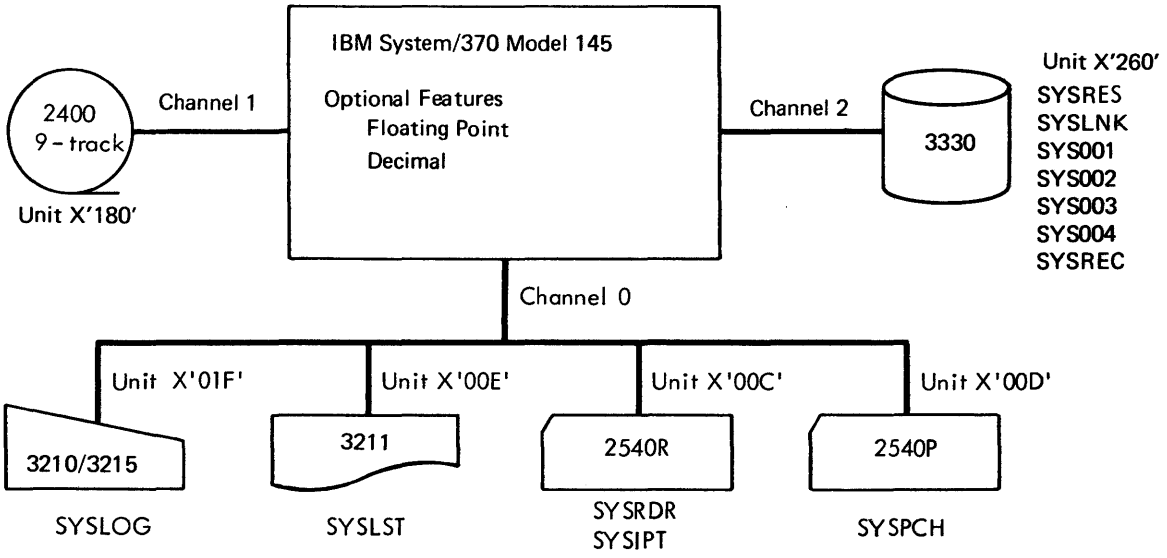


Figure 1-26 PROCEDURES--System configuration for 3330 example

3330 SYSTEM GENERATION PROCEDURES

IBM 3330 users receive the 3330 system on magnetic tape (see Figure 1-25 for distributions). This IBM-supplied tape is capable of restoring the 3330 system to a 3336 disk pack. After the tape has been restored to the 3336 disk pack, the pack contains a core image, a relocatable, and a source statement library.

Before generating a system, you should have at least one initialized disk pack (VTOC on cylinder 403).

Step 1

Mount the IBM-supplied magnetic tape and a disk pack to which the tape is to be restored.

Before restoring the tape, you must take the necessary action to perform or to bypass the initialize disk routine.

a. If the disk pack has not been initialized, the IBM-supplied volume, a self-loading tape, is capable of initializing the packs. The volume serial number used in the EXTENT cards for the following steps must agree with the volume serial number of the disk pack.

   If initialization is required, mount a pack on unit 260. Mount the IBM-supplied tape on unit 180 (9-track drive). Place the following cards in the card reader. Dial unit 180 in the load unit switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. The multipart forms of the linkage editor maps and the supervisor listing will be required by your IBM customer engineer for maintenance purposes.

```
// JOB INTDISK
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSOPT,X'260',D4
// EXEC
// UID IQ
// VTOC STRTADR=(0403000),EXTENT=(19)
VOL1111111
// END
```

b. If the disk pack was previously initialized, the initialize disk routine can be bypassed. Any volume serial number used in the EXTENT cards for the following steps must agree with the volume serial number used when the packs were initialized.

   To bypass the initialize disk program on the IBM-supplied tape, mount the IBM-supplied tape on unit 180 (9-track drive), and mount an initialized disk on unit 260 (VTOC on CYL 403). Insert the following control cards in the reader. Dial unit 180 in the load unit switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. The job is complete when the 00C and 4000A messages are logged.

```
// JOB INTDSK
// DATE 72182
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSIPT,X'180',T2
// FILES SYSIPT,1
```

Step 2

After step 1 is completed, you must restore the system from the self-loading tapes to the initialized pack.

Do not rewind unit 180. To load the tape on disk, place the following cards in the reader. Dial unit 180 in the load switches and press LOAD. When the system enters the wait state, press START and EOF on the card reader. See Restore the IBM-supplied System Tape to Disk for operating procedures.

Note: If two tape units are available, assign the second tape unit for 2 tape distribution to an alternate unit (SYS002).

```
// JOB DISRST
// DATE 72182
// ASSGN SYS000,X'260',D4
// ASSGN SYSLOG,X'01F',C1
// ASSGN SYSLST,X'00E',L1
// EXEC
```

Step 3

When the tape is restored, dial the address (unit 260) of the system residence disk drive into the load unit switches and IPL (press LOAD) to pass control to the DOS supervisor. See IPL Control and ASSGN Statements for System Generation, later in this module, for the IPL control statements. When the system enters the wait state, press START and EOF on the card reader. The following cards are in the card reader (SYSRDR/SYSIPT):

```
ADD X'00C',2540R
ADD X'00D',2540P
ADD X'00E',3211
ADD X'01F',1050A
ADD X'260',3330
ADD X'180',2400T9
SET DATE=06/30/72
ASSGN SYSLOG,X'01F'
LOG
ASSGN SYSRDR,X'00C'
ASSGN SYSIPT,X'00C'
ASSGN SYSPCH,X'00D'
ASSGN SYSLST,X'00E'
ASSGN SYS001,X'260'
ASSGN SYS002,X'260'
ASSGN SYS003,X'260'
ASSGN SYSLNK,X'260'
ASSGN SYSREC,X'260'
SET RF=CREATE
```

Step 4

Perform a sorted DSERV to display the directories. Then SSERV (display and punch) all the desired sample problems from the source statement library into cards. Punched output includes the sample problems with BKEND and CATALS cards. Four other books in the source statement library that should be punched out at this time (Z.LINKEDIT, Z.DELETECL, Z.DELETERL, and Z.DELETESL) contain the necessary control statements to selectively linkage edit and delete all IBM components. The sample problem program names, and the linkage edit and the delete book names can be chosen and punched into the DSPCH statement(s).

```
// JOB DSERV
// EXEC DSERV
DSPLYS ALL
/*
// EXEC SSERV
DSPCH Z.ALL
/*
/&
```

Step 5

Delete components you will never use from the relocatable and source statement libraries.

```
// JOB DELETE
// EXEC MAINT
DELETR (Unwanted Relocatable Library Components)
DELETS (Unwanted Source Statement Library Components)
/*
/&
```

Step 6

Define work files for SYSLNK, SYSREC, SYS001, SYS002, and SYS003. The workfiles SYSLNK, SYSREC, SYS001, SYS002, and SYS003 are defined by use of the DLBL and EXTENT cards. These cards must be preceded by the OPTION STDLABEL or OPTION PARSTD card.

```
// OPTION STDLABEL
// DLBL IJSYSLN,'SYSTEM WORK FILE NO. 0',99/365,SD
// EXTENT SYSLNK,nnnnnn,1,nnnnn,nnnn
// DLBL IJSYS01,'SYSTEM WORK FILE NO. 1',99/365,SD
// EXTENT SYS001,nnnnnn,1,nnnnn,nnnn
// DLBL IJSYS02,'SYSTEM WORK FILE NO. 2',99/365,SD
// EXTENT SYS002,nnnnnn,1,nnnnn,nnnn
// DLBL IJSYS03,'SYSTEM WORK FILE NO. 3',99/365,SD
// EXTENT SYS003,nnnnnn,1,nnnnn,nnnn
// DLBL IJSYSRC,'DOS RECORDER FILE',99/365,SD
// EXTENT SYSREC,nnnnnn,1,nnnnn,nnnn
```

Step 7

Assemble the installation tailored supervisor (see Note).

```
// JOB SUPVR
// EXEC ASSEMBLY
         SUPVR MPS=YES
         CONFG MODEL=145,DEC=YES,FP=YES
         STDJC LISTX=YES,LINES=46
         FOPT IT=BG,PC=YES,OC=YES,CCHAIN=YES,DASDFP=(2,2,3330),        X
              SYSFIL=YES,TOD=YES,ZONE=(WEST,5)
         PIOCS TAPE=9,DISK=3330
         ALLOC F1=14K,F2=14K
         IOTAB JIB=10,F1PGR=8,F2PGR=8,IODEV=10,BGPGR=10,D3330=2,D2400=1
         DVCGEN CHAN=X'00C',DVCTYP=2540R
         DVCGEN CHAN=X'00D',DVCTYP=2540P
         DVCGEN CHAN=X'00E',DVCTYP=3211
         DVCGEN CHAN=X'01F',DVCTYP=1050A
         DVCGEN CHAN=X'180',DVCTYP=2400T9
         DVCGEN CHAN=X'260',DVCTYP=3330
         DVCGEN CHAN=X'261',DVCTYP=3330
         ASSGN SYSRDR,X'00C'
         ASSGN SYSIPT,X'00C'
         ASSGN SYSPCH,X'00D'
         ASSGN SYSLST,X'00E'
         ASSGN SYSLOG,X'01F'
         ASSGN SYSLNK,X'260'
         ASSGN SYS001,X'260'
         ASSGN SYS002,X'260'
         ASSGN SYS003,X'260'
         ASSGN SYSREC,X'260'
         SEND 16384
END
/*
* CHECK ASSEMBLY LISTING FOR ERRORS.  IF CORRECT
* REMOVE ASSEMBLED SUPERVISOR FROM SYSPCH.  INSERT IN READER
* FOLLOWING THE INCLUDE CARD OF JOB CATALSUP
// PAUSE TO CONTINUE PRESS END
/&
```

Note: The operator needs to know the address of label CNLSVE in order to obtain a "special purpose dump". As he normally does not have access to the supervisor assembly listing, you should locate the address in this listing and ask him to write it down under the heading "special Purpose Dump" (Reference 10) in DOS Version 4 Messages, GC33-5009.

Step 8

Perform all other necessary assemblies. The component assemblies should be performed as separate jobs. For the information required to assemble Emulator Programs, refer to the Emulator Program manuals listed in the Preface. You must be careful to keep all assemblies in order. Assemble all your required IOCS modules. By assigning SYSPCH to a tape unit, the IOCS modules can be cataloged to the relocatable library without punching them in cards. The IOCS modules required by compilers, as defined in Module 4, Compiler IOCS Modules, are supplied in the relocatable library by IBM.

Close the tape assigned to SYSPCH and reassign SYSPCH to its permanent assignment by using the CLOSE command. The assembly listings should be checked for errors before proceeding.

```
// JOB ASSEM
// OPTION DECK,LIST,LOG
// ASSGN SYSPCH,X'180'
// EXEC ASSEMBLY
   CDMOD RECFORM=FIXUNB,CTLCHR=ASA,TYPEFLE=OUTPUT,IOAREA2=YES,      X
         DEVICE=2540,SEPASMB=YES
 END
/*
CLOSE SYSPCH,X'00D'
/&
* CHECK ASSEMBLY LISTINGS FOR ERRORS.  RELOAD 180 WITH SAME TAPE.
// PAUSE IF CORRECT PRESS END TO CONTINUE.
```

Step 9

Linkage edit and catalog the assembled supervisor (object module from step 7) to the core image library. If the SEND address is larger than the one used by the supervisor being replaced, certain key programs must also be linkage edited and cataloged to the core image library in the same job step with the new supervisor. These key programs are the librarian programs CORGZ, RSERV, and SSERV. The LINKEDIT deck punched out in step 4 contains all the necessary control statements to linkage edit all IBM components shipped on the system. If the SEND address is not exceeded, only the supervisor is cataloged. The new supervisor is not cataloged until a /& statement is read. Do not attempt any other operation from the time the supervisor and these preceding programs are cataloged until IPL time.

Note: It may be necessary to reallocate because of the size of the programs (components) desired in the various libraries. This may be accomplished by using the ALLOC statement of either the MAINT or CORGZ librarian programs.

```
// JOB CATALSUP
// OPTION CATAL
 ACTION CLEAR
 INCLUDE
 Supervisor object deck here.
/*
// EXEC LNKEDT
 INCLUDE IJBSL3      RSERV
// EXEC LNKEDT
 INCLUDE IJBSL4      SSERV
// EXEC LNKEDT
 INCLUDE IJBSL5      CORGZ
// EXEC LNKEDT
/&
```

Step 10

Re-IPL and issue the SET statement with the applicable parameters (see IPL control statements).

Linkage edit and catalog any additional components desired to the core image library. See the appropriate module for each component for a complete list of control cards for the components to be cataloged. Before the next step is performed, check the linkage editor listings, and make all necessary corrections.

```
 SET
// JOB CATALCLB
// OPTION CATAL
* AT THIS POINT LINKAGE EDIT DESIRED COMPONENTS
// EXEC LNKEDT
/*
/&
```

Step 11

Reload the tape that was assigned to SYSPCH in step 7 and assign it to SYSIPT. With this tape the MAINT program catalogs the IOCS modules to the relocatable library by the control card // EXEC MAINT.

```
// JOB CATALRLB
// ASSGN SYSIPT,X'180'
* CATALOG MODULES TO RELOCATABLE LIBRARY
// EXEC MAINT
/*
* THE FOLLOWING DELETIONS ARE OPTIONAL.
// PAUSE PLACE REMAINING CARDS IN READER.  PRESS END TO CONTINUE
// EXEC MAINT
Place any desired deletes here.
 CONDS RL,SL
/*
/&
// PAUSE EOJ SYSGEN
```

The basic SYSGEN is complete. You may set new standard labels, re-allocate library sizes by using the copy function (CORGZ) and set automatic condense limits if required. Copy the operational pack for backup.

IPL Control and ASSGN Statements for System Generation

The formats for the ADD and SET statements are given in this section. These statements are used at IPL time. If standard physical unit description and assignments are made when assembling the supervisor and if these correspond to the configuration used when the system was generated, no ADD or ASSGN statements are required during subsequent IPL procedures. The SET statement is always required.

ADD (ADD A DEVICE)

Operation	Operand
ADD	X'cuu'[(k)],devicetype[,X'ss']

The entries in the operand field represent the following:

X'cuu'

Channel and unit numbers.

(k)

Specify k=S if the device is switchable (the device is physically attached to two adjacent channels). The designated channel is the lower of the two channels. If the device is not switchable, specify k=0-255. This indicates the priority on the channel of the device, with 0 the highest priority.

devicetype

Actual device (2311, 2400T9, 1443, etc). See Figure 1-5.

Note: Teleprocessing (TP) devices cannot be added if TP=NO in the SUPVR macro.

X'ss'

Device specifications: If absent for 7- or 9-track tapes, X'90' or X'C0' is assumed. (See ASSGN statement for the proper entries.)

The device specifications for an IBM 2702 Transmission Control Unit are:

X'00' for SAD0 X'02' for SAD2
X'01' for SAD1 X'03' for SAD3

The device specification for a 1053 attached to a 2848 is: X'01'.

If the device type is omitted for a 2702 Transmission Control Unit, X'00' is assumed.

The device specifications for a 1018 are:

X'00' for a 1018 without the error-correction feature.
X'01' for a 1018 with the error-correction feature.

The device specification designates the external interrupt bit associated with magnetic ink character readers or optical reader/sorters. The mode X'01' through X'20' corresponds to external interrupt PSWs bits 26 through 31, respectively. For dual address adapter 1419, this parameter is needed for both 1419P and 1419S.

X'01' Device attached to external line 7.
X'02' Device attached to external line 6.
X'04' Device attached to external line 5.
X'08' Device attached to external line 4.
X'10' Device attached to external line 3.
X'20' Device attached to external line 2.

SET (SET THE DATE AND TIME)

Operation	Operand
SET	[DATE=value1[,CLOCK=value2]][,ZONE={EAST|WEST}/hh/mm]

Depending on the message displayed on the console printer-keyboard, after the REQUEST button has been pressed, this statement can be specified without operands, with DATE operand only, with DATE and/or CLOCK operands only, with ZONE operand only, or with all operands.

value1 Specifies the year, month, and day of the month in one of the following formats:

mm/dd/yy

dd/mm/yy

value2 Specifies the time-of-day in the format hh/mm/ss, where hh is hours, mm is minutes, and ss is seconds.

EAST Specifies that the installation is located at a geographical position to the east of the meridian of Greenwich, Great Britain.

WEST Specifies that the installation is located at a geographical position to the west of the meridian of Greenwich, Great Britain.

hh A decimal value in the range 0-12, indicating the difference in hours between local time and Greenwich Mean Time.

mm A decimal value in the range 0-59. This value indicates the minutes portion of the difference between local time and Greenwich Mean Time.

The parameters that have to be specified with the SET statement depend upon the type of system and the type of communications device used. The following combinations can be distinguished:

1. 3210/3215 console printer-keyboard; system without support for either TOD clock, Job Accounting, or QTAM.

   The statement should be given in the form SET DATE=value1. CLOCK and ZONE may be specified but are ignored (message 0I31A is printed).

2. 3210/3215 console printer-keyboard; system without TOD clock support, but with support for either Job Accounting or QTAM.

   The statement must be given in the form SET DATE=value1,CLOCK=value2. ZONE may be specified but is ignored (message 0I31A is printed).

3. 3210/3215 console printer-keyboard; system with TOD clock support.

   If the TOD clock is in the set state (message 0I30I is printed), the statement may be given in any of the five possible forms, depending on whether the values are satisfactory or not.

   If the TOD clock is in the not-set state (message 0I31A is printed), the statement must be given in either of two forms

   1. SET DATE=value1,CLOCK=value2 , or

   2. SET with all three parameters.

   If the TOD clock is inoperative (message 0I32I is printed), perform the procedure as required for systems without TOD clock support (see items 1 and 2 above).

4. Card Reader; system without support for either TOD clock, Job Accounting, or QTAM.

   The statement must be given in the form SET DATE=value1.

5. Card Reader; system witout TOD clock support, but with support for either Job Accounting or QTAM.

   The statement must be given in the form SET DATE=value1,CLOCK=value2.

6. Card Reader; system with TOD clock support.

   The SET statement may, in principle, be given without parameters. However, if the clock is in the 'not set' or 'error' state, the system enters the wait state with a message of 0I31A in low main storage; the IPL procedure must be performed again and the SET statement must be given with DATE and CLOCK or with all three parameters.

   If the clock is not operational (the system enters a hard wait state and message code 0I32A is placed in low main storage) the IPL procedure cannot be performed from a card reader. In that case the system must be loaded using a 3210 or 3215 console printer-keyboard.

Note 1: The date and time-of-day supplied in the SET statement for systems with TOD clock support must be realistic values; the time-of-day clock must always contain the exact time (that is the time that has elapsed since January 1, 1900, 00.00 hours).

Note 2: If the SET statement contains DATE and CLOCK parameters and the system has TOD clock support, the operator must depress the TOD clock switch on the system control panel to the ENABLE SET position at the exact time he specified in the CLOCK parameter.

ASSGN (ASSIGN LOGICAL NAME)

The ASSGN command assigns a logical input/output unit to a physical device.

Operation	Operand
ASSGN	SYSnnn,X'cuu' [,X'ss'] [,ALT]

The entries in the operand field represent the following:

SYSnnn

The symbolic unit name. It may be one of the following:

SYSRDR	SYSLNK	SYSSLB
SYSIPT	SYSIN	SYSREC
SYSPCH	SYSOUT	SYS000-SYS221 (if MPS=BJF)
SYSLST	SYSCLB	SYS000-SYS243 (if MPS=NO or YES)
SYSLOG	SYSRLB	

X'cuu'

Indicates the hexadecimal channel (c) and unit (uu) number. C=0 for the multiplexor channel. C=1-6 for selector channels 1-6.

X'ss'

Device specifications for 7- and 9-track tape are:

ss	Bytes Per Inch	Parity	Translate Feature	Convert Feature
10	200	odd	off	on
20	200	even	off	off
28	200	even	on	off
30	200	odd	off	off
38	200	odd	on	off
50	556	odd	off	on
60	556	even	off	off
68	556	even	on	off
70	556	odd	off	off
78	556	odd	on	off
90	800	odd	off	on
A0	800	even	off	off
A8	800	even	on	off
B0	800	odd	off	off
B8	800	odd	on	off
C8	800	9-track single density		
C0	1600	9-track dual density		
C8	800	9-track dual density		
C0	1600	9-track single density		

Notes: There are two possible device specifications for 9-track tape units: X'C0' and X'C8'. C0 is the normal reset mode for the three kinds of 9-track tape units, namely:

1. A tape unit capable of reading from and writing on 800 b.p.i. tape only;

2. A tape unit capable of reading from and writing on 1600 b.p.i. tape only, and

3. A tape unit capable of reading from and writing on both 800 and 1600 b.p.i. tape.

C8 is an alternate mode setting for type 3 units only, when tape with a density of 800 b.p.i. is used.

When the system is generated, it is possible to make an explicit selection of mode setting for each magnetic tape unit, or let the system take a standard action. If the latter action is chosen, the system always assumes C0 for the device.

For a 9-track dual density output tape, the ASSGN density is compared with the VOL1 density of the mounted tape. If a discrepancy is found, and if the tape is at load point, the volume label(s) is written according to the ASSGN density.

ALT

Alternate tape unit is used when the capacity of the original assignment is reached.

DOS System/370 Distribution Program and BPS Messages

Supervisor Messages

When a response is typed in on all actions, press Interrupt to continue.

Message	Cause	Action
0901	Program Check	No restart; take a storage print. Examine the program old PSW for error information.
0701	Program Check	
0702A	The Console Printer Keyboard request key has been pressed.	Type in 4 blank and pre Interrupt to continue.
Ocuu	Device I/O errcr. The seccnd character gives the channel number. The others give the unit information.	Refer to Hardware Error Messages.

Job Control Messages

1050A	Missing DATE card.	Reload the program with corrected control cards
1040A	Missing JOB card.	
1110A	Duplicate JOB card.	
1200A	Invalid card, i.e., // not in columns 1 and 2	
1220A	Invalid or missing contrcl card.	
13xxA	Invalid field xx (field number).	
1703A	PAUSE card in deck. Shculd not occur because Utility prcgrams dc nct require or use a pause card.	To continue, enter two blanks cn SYSLOG or pre the Interrupt key when the message is displaye on the console.

Initialize Disk Messages

Message	Cause	Action
CONTROL CARD ANALYSIS AND LABEL CHECKING	Identifies beginning of initialize disk function.	None. Processing continues.
JOB CARD NOT INITDSK	Job name in job card is not INITDSK.	Reload program with corrected control cards.
SYSOPT NOT DEFINED	SYSOPT must be assigned.	
SYSOPT(002,003,etc.) NOT A VALID DISK DRIVE	Indicated symbolic unit is not a disk drive in JCL syntax.	
X'cuu' NOT A VALID DISK DRIVE [PACK IS DELETED FROM FURTHER PROCESSING]	Indicated disk drive not IBM supported, or wrong 'cuu'specification	Reload program with corrected control cards if ABNORMAL EOJ; otherwise no action required.
UTILITY MODIFIER CARD	Identifies the control card parameters that follow this heading.	None. Processing continues.
MISSING UTILITY MODIFIER CARD	Utility modifier card is missing.	Reload program with corrected control cards.
INVALID CARD	Utility modifier statement identified improperly.	
INVALID FORMAT	Format is incorrect (either a parameter is missing or it is out of order)	
INVALID I (or C or R) PARAMETER	Parameter value is incorrect.	
DISK TYPE SPECIFICATION FOR SYS002(003,004,or 005) IS NOT EQUAL SYSOPT. PACK IS DELETED FROM FURTHER PROCESSING	Indicated symbolic unit differs from SYSOPT type as specified in JCL syntax	None. Processing continues without this unit being used.
xxxx DISK TYPE ON X'cuu' WHILE yyyy IS EXPECTED (E.G. xxxx=2311 AND yyyy=3330) PACK IS DELETED FROM FURTHER PROCESSING	Indicated disk drive differs from specification in job control language (JCL)	
NO VOL1 LABEL (OR F4 LABEL OR VTOC AREA) FOUND ON X'cuu'. 'IS' OPTION INVALID	Pack on indicated drive is to be regarded as not being initialized, because no VOL1 label, F4 label or VTOC was found.	Reload program after 'IS' option in utility modifier card has been changed.
NO VTOC MATCH FOUND ON X'cuu'. 'IS' OPTION INVALID	Pack on indicated drive is to be regarded as not being initialized, because device constants do not match.	

Message	Cause	Action
SURFACE ANALYSIS AND HA-R0 GENERATION	Identifies the function of this phase (2311 and 2314/2319 only).	None. Processing continues.
NO SURFACE ANALYSIS, ONLY R0 GENERATION	Identifies the function of this phase for the 3330 only.	None. Processing continues.
SYSOPT(002,003,etc.)	Disk packs to be processed will be logged with their respective message, starting with SYSOPT.	
CYLxxx,TRKxx(DECIMAL),IS A DEFECTIVE ALT TRK, NO ALT TRK ASSIGNED	Cylinder and track are identified by decimal characters. The track on the alternate cylinder is defective. An alternate is not assigned.	
CYLxxx,TRKxx(DECIMAL), IS DEFECTIVE, AN ALT TRK IS ASSIGNED	Main area of the track identified by cylinder and track number is defective. An alternate is assigned.	
CYLxxx,TRKxx,(DECIMAL) IS DEFECTIVE, AN ASSIGNED ALT TRK WAS DETECTED	Main area of the track identified by cylinder and track number is defective. An alternate track had already been assigned (3330 only).	
CYLxx,TRKxx(DECIMAL), IS DEFECTIVE, ALT CYL IS FULL, NO ALT TRK ASSIGNED SYSxxx WILL BE DELETED AFTER PROCESSING IS FINISHED	This defective track cannot be assigned an alternate track. The alternate cylinders are full.	No correction procedure. Surface analysis (2311 and 2314 only) and R0 generation are performed on the rest of the pack; however, the pack will be deleted from further processing.
CYLxx,TRKxx(DECIMAL), HOME ADDRESS (HA) OR RECORD ZERO (R0) DEFECTIVE, SYSxxx WILL BE DELETED AFTER PROCESSING IS FINISHED	Portion of the track where the home address or record zero resides is defective.	
CYLxxx,TRKxx,,(DECIMAL) IS DEFECTIVE, NO ASSIGNED ALT TRK WAS DETECTED SYSxxx WILL BE DELETED AFTER PROCESSING IS FINISHED	Main area of the track identified by cylinder and track number is defective. No alternate track had been assigned (3330 only).	

Message	Cause	Action
VOL/VTOC LABELS CONTROL SET	Identifies sets of VOL1 and VTOC cards.	None. Processing continues.
SYSOPT (002,003,etc.)	Disk packs are identified with their respective messages, if any.	
SYSOPT (002,003,etc.) DELETED	Identified disk pack is deleted from further processing.	
VTOC CARD MISSING OR INCORRECT. 3LC1A	VTOC card is missing or is incorrect.	1. Correct the cards in the volume label set (beginning with // VTOC card followed by VOLn cards).
INVALID VTOC ADDRESS OR INVALID EXTENT PARAMETER 3LC2A	Invalid VTOC start address or invalid (or missing) EXTENT parameter.	
VTOC OVERFLOWS CYLINDER 3LC3A	Assigned VTOC area overflows the cylinder.	2. Place cards in the read hopper and ready the reader.
VOL1 CARD MISSING OR VOL1n OUT OF SEQUENCE 3LC4A	VOL1 card is missing, an incorrect volume card was supplied, or VOL1 through VOLn are out of sequence.	3. Space SYSLOG until program takes control. Press interrupt key to continue.
VOL1 SERIAL FIELD CONTAINS BLANKS 3LC5A	VOL1 card has blanks in volume serial field.	
VTOC CARD OR END CARD OR VOL CARD ERROR	Wrong VTOC or END control card, or END card is missing.	
VTOC CARD SETS UNEQUAL NUMBER OF ASSIGNED PACKS 3LC7A	VTOC card sets and the number of assigned packs is not equal.	
PARAMETER DELMITER 3LC8A	Comma or blank must be after tha parameter.	
4404A	No output-file format 4 label can be found.	Type in 4 blank to ignore the error and press Interrupt to continue processing. Type in 0 blank or 1 blank to terminate the job.
UNEXPIRED FILE ON xxxx AT X'cuu':"(filename from F1 label)". 4444A (xxxx=2311,2314, or 3330)	Identified file is not expired yet.	a) Type in 4 blank and press Interrupt to delete file by file. b) Type in 2 blank to delete this and all other files. c) Type in 1 or 0 to terminate the job (wait state). d) Type in any digit other than 0, 1, 2, or 4 to delete this pack, and continue processing.

Restore Messages

Message	Cause	Action
DISTRIBUTION PROGRAM	Identifies the program.	None. Processing continues.
RESTORE FUNCTION	Indicates that the restore mainline phase is in operation.	
TAPE READ ERROR 3003 ABNORMAL EOJ	Tape error.	None. Job is terminated.
JOB NAME NOT DISRST 3004 ABNORMAL EOJ	Job name in the jcb card is incorrect.	
ALTERNATE DISK (TAPE) SPECIFICATION REJECTED. PROCESSING CONTINUED	Assign card for SYS001 (SYS002) in JCL in error. SYS000(SYSIPT) will now function as an alternate disk (tape).	None. Processing continues.
'FILE NAME' XXXXXX RECORDS RESTORED FOR ABOVE FILE 3007	XXXXXXis the number of records restored from tape to disk for this file.	
END OF JOB 3008	End of job.	None. Job is terminated.
SYS000(OPT) AND SYS001 (ALT.OPT) NOT EQUAL DISK TYPE 3013 ABNORMAL EOJ	When processing multiple volumes, one type of disk storage is required for output and alternate output.	
SYS000(or SYS001) NOT ASSIGNED TO A VALID DISK 3016 ABNORMAL EOJ	SYS000 or SYS001 must be assigned to a 2311, 2314/2319, or 3330.	
A xxxx DISK FILE CANNOT BE RESTORED TO A yyyy 3026 ABNORMAL EOJ	The format of the tape file does not agree with the disk to be restored (e.g. xxxx=2314, yyyy=3330).	
END OF VOLUME SWITCHING TO ALTERNATE 3677	Occurs at end of tape volume when alternate drive has been assigned.	None. Processing continues.
END OF VOLUME ON SYSIPT READY NEXT REEL ON SYSIPT TAPE DRIVE 3777A	Occurs at end of tape volume when no alternate drive has been assigned.	Ready next reel on SYSIPT and type in 2 blank, then press Interrupt to continue. Reply 0 or 1 to terminate; any other reply results in message code repetition.

Message	Cause	Action
3999 ABNORMAL EOJ	SYSLST or SYSLOG not assigned, or SYSLST and SYSLOG assigned to the same device.	None. Job is terminated.
NO VOL1 LABEL ON SYS000 (or SYS001) 4306A	Disk pack has no VOL1 label.	Replace pack and type in 2 blank, then press Interrupt to continue; type in any digit other than 0 or 1, followed by a blank, to terminate. Any other reply results in message code repetition.
file name THIS FILE FOR SYS000 SYS001 4307A	Identifies the file to be restored.	Type in 2 blank and press Interrupt to restore this file. Type in 4 to bypass this file and continue processing with the next file. Type in 0 or 1 to terminate. Any other reply results in message code repetition. For the single drive user, SYS000, and SYS001, refer to the same drive.
SWITCHING TO SYS000 SYS001 4308A	Designates the drive, SYS000 or SYS001, to which the volume is to be restored.	Assure that an initialized pack is mounted on the designated drive. Type in 2 blank and press Interrupt to restore the volume. Type in 0 or 1 to terminate the job. Any other reply results in message code repetition. For the single-drive user, SYS000, and SYS001, refer to the same drive.
DISK WRITE ERROR 4311A	Disk error	Replace pack and type in 2 blank to continue; type in 0 or 1, followed by a blank, to terminate.

Message	Cause	Action
4400A ABNORMAL EOJ	No available space in VTOC .	None. Job is terminated.
4404A ABNORMAL EOJ	No output-file format 4 label can be found.	
4409A ABNORMAL EOJ	No output-file format 1 label can be found.	
4410A ABNORMAL EOJ	Data check occurred while reading output-file count field.	
4414A ABNORMAL EOJ	Error occurred while reading output-file format 4 label.	
4419A ABNORMAL EOJ	Read error occurred while searching the output-file VTOC.	
VTOC NOT ON CYL 199(OR 403) 4428 ABNORMAL EOJ	a) VTOC not on cylinder 199 for a 2311, or 2314/2319 b) VTOC not on cylinder 403 for a 3330.	None. Job is terminated.
UNEXPIRED FILE ON SYS000 (OR SYS001) AT X'cuu': "(filename)" 4444A	Extent limits of a file being restored overlap the identified unexpired file on the output disk pack.	a) Type in 4 blank and press Interrupt to delete file by file. b) Type in 2 blank to delete this and all other files. c) Type in 0 or 1 to terminate the job. Any other reply results in message code repetition.

Hardware Error Messages

No message is given, machine is in wait loop; registers A and B or instruction counter contains FFFF. (Note: All storage locations given are in hexadecimal.)

Program: Any.

Meaning: Byte 32 contains one of four error codes. These codes are shown with their respective operator actions in the following text. There is no restart option provided for these codes. IPL must be used.

00 - Machine check. Diagnostic information is in the diagnostic scan-out area, starting in byte 80. Use the storage print utility program to obtain a storage printout for the programmer.

0F - Channel error. Record the contents of the CSW, bytes 40-47; and the I/O Old PSW, bytes 38-3F.

1F - Unit check. The CSW has the original information. Sense information is in bytes 18-1D. Use the storage print utility program to obtain a storage printout, for the programmer.

3F - Device not operational. Program is addressing a nonexistent device. This device is identified in bytes 38-3F. If the device does exist, use the storage print utility program to obtain a storage printout for the programmer.

First Part of Three-Part PIOCS Message--Channel and Unit Address

Message	Meaning	Action
0cuu	Part 1 of 3 parts. This part tells the channel and unit address of a unit on which an I/C error has occurred. The second digit (c) of this part of the message identifies the channel, and the last two digits (uu) identify the unit.	None. The second part of the message is printed following the first part.

Second Part of Three-Part PIOCS Message - Status and Operation Code

Message	Meaning	Action	Operator reply; type in:
nnnn	Part 2 of 3 parts. This part tells the status and operation code information.	The first 2 digits of the status are supplied by either the device or the device control unit.	
	The meanings of the first digit are:		
	0nnn - None (examine the remaining characters).		
	1nnn - Device or control unit busy.		
	2nnn - Control unit end.		
	4nnn - Status modifier.		
	8nnn - Attention.		
	The meanings of the second digit are:		
	n0nn - None (examine the remaining characters).		
	n1nn - Unit exception.		
	n2nn - Unit check.		
	n4nn - Device end.		
	n8nn - Channel end.		
	n6nn - Unit check at device end, last item of output may be in error. There is a non-recoverable error present, regardless of the information in the third part (sense) of the message.	Ignore the error and continue processing.	4 blank or 5 blank
	The last two digits constitute the operation code of the last CCW executed. An exception to this is that on a DVC END error, the operation code digits are set to zero.		

Third Part of Three-Part PICCS Message-Sense Information

Message	Meaning	Action	Operator reply: type in:
nnnnA	Part 3 of 3 parts. This part tells the sense bit information. The characters in this part of the message indicate sense bits set by the device control unit. The meanings of these sense bits are found in the SRL publication for the particular device.	None. The third part of the message is printed following the second part.	
	The meanings of the first digit are:		
	0nnnA - None (examine the remaining characters).		
	1nnnA - Equipment check	Either ignore the error, and continue processing, or	4 blank
		Examine the device for trouble, then retry.	5 blank
	2nnnA - Parity error on channel.	Either ignore the error and continue processing, or	4 blank
		Reply to retry the operation.	5 blank
	4nnnA - Intervention required.	Either ignore the error and continue processing, or	4 blank
		Ready the device, then reply to retry.	5 blank
	8nnnA - Command reject (invalid, missing, or incorrect command).	Ignore the error and continue processing; (if the device was a printer, and a print or sync check occurred after channel end, a retry of the same line is not permitted); or	4 blank
		Reply to retry the operation.	5 blank
	8nn2A - Write inhibited on 3330 disk storage (switch is in Read Only position).	Reply to retry the operation.	5 blank

Third Part of Three-Part PIOCS Message-Sense Information (continued)

Message	Meaning	Action	Operator reply: type in:
	The meanings of the second digit of the third part are:		
	n0nnA - None (examine the remaining characters)		
	n1nnA - Unit exception.	Ignore the error and continue processing.	4 blank
	n2nnA - Nonrecovery.	Reply to retry the operation.	5 blank
	n4nnA - Overrun.	Ignore the error and continue processing.	4 blank
	n8nnA - Data check. If the device is a tape drive, the check is due to a read error.	Ignore the error, and continue processing, or Reply to retry the operation.	4 blank 5 blank
	The meanings of the third digit of the third part are:		
	nn0nA - None (examine the remaining characters).	Follow the action instructions specified by the first two digits.	
	nn1nA - If disk, invalid sequence of commands. If tape, tape unit has the 7-track feature installed.		
	nn2nA - If disk, end of the cylinder. If tape, tape drive not ready.		
	nn4nA - If disk, track overrun. If tape, tape drive ready, but not rewinding.		

Third Part of Three-Part PIOCS Message-Sense Information (continued)

Message	Meaning	Action	Operator reply: type in:
	nn6nA - If disk, end of cylinder and track overrun have occurred simultaneously. If tape, tape drive ready and rewinding. nn8nA - Data check in count field. (This message for disk only).		
	The meanings of the fourth digit are:	Follow the action instructions specified by the first two digits.	
	nnn0A - None (If the channel and unit address identified by the first part of the message is that of a tape or disk drive, and the third digit is also 0, a nonexistent tape or disk unit has been addressed.)		
	nnn1A - If disk, overflow incomplete. If tape, tape indicator is ON.		
	nnn2A - If disk, missing address mark. If tape, tape drive is in file-protect status.		
	nnn4A - If disk, file mask violated. If tape, tape drive is in write status.		
	nnn8A - If disk, no record found. If tape, the tape is at load point.		

DOS System Control -- 370N-CL-453

Figure 1-27 contains both the names of the programs included in this section and a summary of the library requirements for these programs.

Component	Core Image Library Blocks				Relocatable Library			Source Statement Library	
	Phase	2311	2314/ 2319	3330	Prefix	Module	Blocks	Books	Blocks
System Control:							**841		
IPL	3	6	6	8	IJB	1	**		
Job Control	10	29	29	30	IJB	10	**		
Linkage Editor	8	16	16	19	IJB	4	**		
Librarian:									
CORGZ	9	20	20	22	IJB	10	**		
MAINT	10	34	34	37	IJB	18	**		
CSERV	1	5	5	3	IJB	1	**		
DSERV (Sorted)	6	10	10	10	IJB	2	**		
RSERV	1	4	4	5	IJB	2	**		
SSERV	1	4	4	5	IJB	2	**		
Supervisor:	1	14	14	21					
Macros:									
Communications								26	180
Generation:									
Basic								23	3687
MICR								1	237
TP								7	181
IOCS: See Note.								***2	24
Card (Reader/Punch)					IJC	64	154	2	311
Console								1	73
Device Independent					IJJ	4	14	2	206
Printers					IJD	23	52	2	196
Serial Device								1	51
PIOCS (DTFPH)								1	55
Imperative Macros								27	306
Transient: (System and Common IOCS)	222	222	223	221					
MCAR/CCH	14	14	14	14					
Dumps:									
Standard System	6*	6	6	6	IJB	7	46		
Translating System	4*	4	4	4	IJB	5	31		
Aids:									
ESTVUT	1	3	3	3	IJB	1	14		
PDAID	15	18	18	17	IJB	2	38		
Transient	1	1	1	1					
DUMPGEN	2	8	9	9	IJB	1	51		
LSERV	1	5	5	6	IJB	1	18		
3211 Printer Support	2	2	2	2					
Transients	4	4	4	4					
UCSB					IJB	6	32		
System Generation								5	334

* Transients

** The number of System Control Relocatable Library blocks includes only the IPL, Job Control, Linkage Editor, and Librarian Programs.

***For BOS and BPS Compatibility

Note: See Compiler IOCS Modules for the preassembled module names and usage.

Figure 1-27. CONTROL--DOS System Control Summary

IOCS

SOURCE STATEMENT LIBRARY

The following macros are shipped in the source statement library.

Note: Module 4: Compiler IOCS Modules, contains a list of preassembled modules required by system programs.

File Definition Macros

A.CDMOD
A.DIMOD
A.DTFBG
A.DTFCD
A.DTFCN

A.DTFDI
A.DTFEN
A.DTFPH
A.DTFPR
A.DTFSR
A.PRMOD

Imperative Macros

	Main Storage Bytes		
	Basic	For Literals (1)	For Variables
A.CCB	16-24		
A.CHECK	8	+4	Note 2.
A.CLOSE	10	+8	Note 3.
A.CLOSER	14+10 if any Reg. is specified	+8	Note 4.
A.CNTRL	10-18	+4	
A.ERET			
A.EXCP	2-6	+0-4	
A.FEOV	8-12	+4	
A.FEOVD			
A.FREE	8-12	+0-4	
A.GET	8	+4 per sym. name	Note 2.
A.LBRET	2		+2
A.NOTE	12	+4	
A.OPEN	10	+8	Note 3.
A.OPENR	14+10 if any Reg. is specified	+8	Note 4.
A.POINTR	8	+4 per sym. name	Note 2.
A.POINTS	8-12	+4	
A.POINTW	8	+4 per sym. name	Note 2.
A.PRTOV	8	+4 per sym. name	Note 2.
A.PUT	8-12	+4	Note 2.
A.READ	8-36 Note 5.	+4 per sym. name	Note 6.
A.RELEASE			
A.RELSE	8	+4 per sym. name	
A.SEOV	10	+8	+3
A.TRUNC	8-12	+0-4	
A.WAITF	8-12	+0-4	Note 3.
A.WRITE	8-26 Note 5.	+0-4	

Notes:

1. Include the storage requirement for literals only once if the literal is used by more than one macro instruction.

2. +4 per symbolic name, +2 per ordinary register notation, or +0 per special register notation.

3. +4 per filename or +8 per register.

4. +14 per filename or +8 per register.

5. 20 bytes are required for BTAM.

6. +38 No operands coded in register notation.

IOCS Macro Storage Requirements

The storage required by the imperative macros may be determined by using the preceding figures with Figures 1-28 through 1-32.

CDMOD (CARD MODULE)

			TYPEFLE					
			INPUT	OUTPUT			CMBND	
RECFORM	IOAREA2	WORKA	1442 2520 2540 2501 2596 3505 3525	1442 2596	2520	2540 3525	1442	2520 2540
FIXUNB	-	-	104	74	80	40	126	198
FIXUNB	-	YES	106	116	122	84	154	226
FIXUNB	YES	-	136	118	124	82	126	198
FIXUNB	YES	YES	138	132	138	96	154	226
UNDEF	-	-	-	112	110	70	-	-
UNDEF	-	YES	-	124	132	92	-	-
UNDEF	YES	-	-	128	130	90	-	-
UNDEF	YES	YES	-	140	146	104	-	-
VARUNB	-	-	-	126	132	94	-	-
VARUNB	-	YES	-	140	146	108	-	-
VARUNB	YES	-	-	154	160	118	-	-
VARUNB	YES	YES	-	156	162	120	-	-

Figure 1-28. CONTROL--CDMOD Main Storage Requirements (Part 1 of 3)

Notes:

1. CTLCHR=YES or ASA; depending upon record format, number of I/O areas, and/or work area specifications:

   for YES, an minimum of 8 to a maximum of 36 additional bytes are required.

   for ASA, a minimum of 28 to a maximum of 65 additional bytes are required.

2. RDONLY=YES changes the size of the modules -50 to +50 bytes. In addition, the user's program must provide a 72-byte save area each time the module is reentered.

3. Parts 2 and 3 of Figure 1-28 shows the requirements for the other CDMOD options. The values selected must be added to the TYPEFLE value to determine the amount of storage needed.

CRDERR=RETRY	2520	2540
without ICAREA2 or WORKA	+73	+123
with either or both	+69	+119

Figure 1-28. CONTROL--CDMOD Main Storage Requirements (Part 2 of 3)

CONTROL=YES	1442	2540	2520	3505/3525*
INPUT	+26	+82	+30	+94
OUTPUT without WORKA	+24	+20	+12	+26
OUTPUT with WORKA	+24	+32	+12	+38
CMBND without WORKA	+24	+20	+12	--
CMBND with WORKA	+32	+32	+12	--

Figure 1-28. CONTROL--CDMOD Main Storage Requirements (Part 3 of 3)
*Control may not be specified for input files used as associated files on the 3525.

DTFCD (DEFINE THE FILE CARD)

TYPEFLE=INPUT requires 50 bytes. (56 bytes if ERROPT is specified for a 3505 or 3525, 60 bytes if ASOCFLE is specified for a 3525.)

TYPEFLE=CMBND requires 84 bytes.

TYPEFLE=OUTPUT requires 48 bytes. (52 bytes if ASOCFIE is specified for a 3525, 128 bytes if FUNC=I is specified for a 3525.)

CRDERR=RETRY (2540 only) requires an additional 88 bytes.

DEVICE=2520 requires 8 additional bytes.

DIMOD (DEVICE INDEPENDENT SYSTEM UNITS MODULES)

TYPEFLE=	Basic Module	IOAREA2=YES
INPUT	316	+58
OUTPUT	667	+84

Note: RDONLY=YES changes the size of the module -50 to +50 bytes.In addition, the user's program must provide a 72-byte save area each time the module is reentered.

Figure 1-29. DIMOD Main Storage Requirements

DTFDI (DEFINE THE FILE DEVICE INDEPENDENT SYSTEM UNITS)

The table requirement is 240 bytes.

PRMOD (PRINTER MODULE)

RECFORM=	BASIC MODULE (Note 3)	WORKA=YES	ICAREA2=YES	PRINTOV (Note 4)	CTLCHR= ASA	CTLCHR= YES	CONTROL=YES (Note 5)	ERROPT=YES (Note 2)
FIXUNB	72	+6	+24	+34	+128		+40	+68
UNDEF	102	+22	+12	+34	+132	+4	+40	+68
VARUNB	136	+14	+12	+34	+124	-4	+40	+68

Note 1: RDONLY=YES changes the size of the module -50 to +50 bytes. In addition, the user's program must provide a 72-byte save area each time the module is reentered.

Note 2: If ERROPT=YES is specified, then DEVICE=3211 must also be specified. For a 3525 add 22 bytes, instead of 68, for ERROPT.

Note 3: For the 3525, add from 52 to 72 bytes to the size of the basic module, depending on the type of associated file specified.

Note 4: For the 3525 add from 34 to 58 bytes.

Note 5: For the 3525 add from 40 to 160 bytes.

Figure 1-30. CONTROL--PRMOD Main Storage Requirements

DTFPR (DEFINE THE FILE PRINTER)

The table requirement is 48 bytes or 52 bytes if ERROPT is specified for the 3211 printer or, if ASOCFLE is specified, for the 3525 card punch.

DTFPH (DEFINE THE PHYSICAL IOCS)

Device	Size
Tape	104
DASD MOUNTED=ALL	40
DASD MOUNTED=SINGLE	84

Note: No module is required for this macro instruction.

Figure 1-31. CONTROL--DTFPH Main Storage Requirements

DTFCN (DEFINE THE FILE CONSOLE)

		TYPEFLE=	
RECFORM=	WORKA=	INPUT/OUTPUT	OUTPUT only
FIXUNB	-	94	60
FIXUNB	YES	150	90
UNDEF	-	156	114
UNDEF	YES	262	168

Note: No module is required for this macro instruction.

Figure 1-32. CONTROL--DTFCN Main Storage Requirements

DTFSR (DEFINE THE FILE SERIAL DEVICE)

When a DTFSR is assembled, it generates both a table and a module. For example, if DTFSR is used for a printer, a table and a module are generated just as though DTFPR and PRMOD were used. To determine the main storage requirements for DTFSR, add the table and the module requirements for the appropriate device type,such as 48 bytes (DTFPR) and n bytes (PRMOD) for a printer.

IPL

CORE IMAGE LIBRARY

Phases

$IPLRT2
$IPLRT3
$IPLRT4

RELOCATABLE LIBRARY

Module

IJBIPL

Linkage Edit Statements

```
   INCLUDE IJBIPL
// EXEC LNKEDT
```

Job Control

CORE IMAGE LIBRARY

Phases

$JCBACCT *
$JOBCTLA
$JOBCTLD
$JOBCTLF
$JOBCTLG

```
$JOBCTLJ
$JOBCTLK
$JOBCTLM
$JOBCTLN *
IJBRSTRT (restart)

* Required for Job Accounting Interface
```

RELOCATABLE LIBRARY

Modules

```
$JOBACCT
IJBJC
IJBJC1
IJBJC2
IJBJC3
IJBJC4
IJBJC5
IJBJC6
IJBJC7
IJBJC8 (Job Accounting Interface)
```

Linkage Edit Statements

```
   INCLUDE IJBJC
// EXEC LNKEDT
```

Linkage Editor

CORE IMAGE LIBRARY

Phases

$LNKEDT	Initialize/overhead (Phase 1)
$LNKEDTA	Pass 2 processor (Phase 7)
$LNKEDTC	Catalog processor (Phase 8)
$LNKEDT0	12-2-9 processor, ESD only (Phase 2)
$LNKEDT2	12-2-9 processor, non-ESD (Phase 3)
$LNKEDT4	Control card processor (Phase 4)
$LNKEDT6	Control card processor (Phase 5)
$LNKEDT8	MAP processor (Phase 6)

RELOCATABLE LIBRARY

Modules

```
IJBLBC
IJBLBI
IJBLE
IJBLE1
```

Linkage Edit Statements

```
   INCLUDE IJBLE
// EXEC LNKEDT
   INCLUDE JOBACCT
// EXEC LNKEDT
```

WORKFILE REQUIREMENTS FOR LINKAGE EDITOR

This section contains information for determining the workfile requirements for the Linkage Editor and Assembler when the workfile is on an IBM 2311, 2314/2319, or 3330.

Two workfiles are used by the Linkage Editor: SYSLNK for input, and SYS001 for a workfile.

The best overall performance for linkage editing results from using two disks and one tape: SYSRES and SYSLNK each assigned to a separate disk drive, and SYS001 assigned to tape. When possible, SYSLNK should be assigned to the faster of the two disks because more I/O is performed on SYSLNK than on SYSRES. Because linkage editing time is relatively small compared to assembler or compiler times, optimum assignments for assembling and linkage editing (or compiling and linkage editing) should be based on assembler or compiler conditions, not linkage edit time. When making optimum workfile assignments, the major consideration should be compiler time. A savings in linkage editing time is generally at the expense of compiler performance.

When built by a language processor, SYSLNK contains 25 card images per track on a 2311, 39 on a 2314/2319, and 60 on a 3330. When an object deck is used as input to the Linkage Editor, job control formats SYSLNK. In this case, SYSLNK contains 9 records per track on a 2311, 16 on a 2314/2319, and 28 on a 3330; ESD, TXT, and RLD cards are packed 4 per record while all other input cards are not packed (1 per record).

In a compile and link-edit situation, any allocation made for SYS001 for the compilation is more than sufficient as a workfile allocation for the Linkage Editor. However, when you must allocate SYS001, you can use the following information. The Linkage Editor workfile (SYS001) contains 11 records per track on a 2311, 21 on a 2314/2319, and 35 on a 3330. The total number of records (R) required for link-editing a program is equal to the following:

$$R = 1 + \left\lceil \frac{X_1}{4} \right\rceil + \left\lceil \frac{X_2}{4} \right\rceil + \cdots \left\lceil \frac{X_n}{4} \right\rceil$$

where $\lceil\ \rceil$ = rounded high

$x_1, x_2, \ldots, x_n$ = number of RLD cards in each module to be processed by the linkage editor.

Librarian

The following are the librarian programs.

Note: Module IJBLBC (error message routine) is a common module used by more than one librarian program and also by the linkage editor.

CORGZ

CORE IMAGE LIBRARY

Phases

```
CORGZ
CORGZ1
CORGZ2
CORGZ3
CORGZ4
CORGZ5
CORGZ6
CORGZ7
CORGZ8
```

RELOCATABLE LIBRARY

Modules

```
IJBLBJ
IJBLBK
IJBLBS
IJBLBT
IJBLBU
IJBLBV
IJBLBW
IJBLBX
IJBLBY
IJBSL5
```

Linkage Edit Statements

```
   INCLUDE IJBSL5
// EXEC LNKEDT
```

MAINT

CORE IMAGE LIBRARY

Phases

```
MAINT
MAINTA
MAINTCI
MAINTCN
MAINTDR
MAINTEJP
MAINTR2
MAINTS2
MAINTUP
$MAINECJ     Update program routine
```

RELOCATABLE LIBRARY

Modules

```
IJBLBA
IJBLBC
IJBLBD
IJBLBE
IJBLBF

IJBLBG
IJBLBH
IJBLBI
IJBLBM
IJBLBQ

IJBLBZ
IJBMCS
IJBMDS
IJBMDU
IJBMIN

IJBMIO
IJBMUP
IJBSL2
```

Linkage Edit Statements

```
   INCLUDE IJBSL2
// EXEC LNKEDT
```

CSERV

CORE IMAGE LIBRARY

Phase

```
CSERV
```

RELOCATABLE LIBRARY

Module

IJBLBP

Linkage Edit Statements

```
   INCLUDE IJBLBP
// EXEC LNKEDT
```

DSERV

CORE IMAGE LIBRARY

Phases

DSERV
DSERV1
DSERV2
DSERV3
DSERV4
DSERV5

RELOCATABLE LIBRARY

Modules

IJBLBC
IJBSL1

Linkage Edit Statements

```
   INCLUDE IJBSL1
// EXEC LNKEDT
```

RSERV

CORE IMAGE LIBRARY

Phase

RSERV

RELOCATABLE LIBRARY

Modules

IJBLBC
IJBSL3

Linkage Edit Statements

```
   INCLUDE IJBSL3
// EXEC LNKEDT
```

SSERV

CORE IMAGE LIBRARY

Phase

SSERV

RELOCATABLE LIBRARY

Modules

IJBSL4
IJBLBC

Linkage Edit Statements

```
   INCLUDE IJBSL4
// EXEC LNKEDT
```

Supervisor

CORE IMAGE LIBRARY

Phase

$$A$SUP1

SOURCE STATEMENT LIBRARY

Macros

SUPERVISOR COMMUNICATION:

	Main Storage Bytes		
	Basic	For Literals (1)	
A.ATTACH	18-32	+0-4	(Note 2)
A.CALL	2-16	+0-4	(Note 2)
A.CANCEL	4-6	ALL+2	

A.CHKPT	38-42	
A.CHNG		
A.CCMRG	6	
A.DEQ	4-8	+0-4
A.DETACH	2-6	+0-5
A.DUMP	6	8
A.ENQ	4-8	+0-4
A.EOJ	2	
A.EXIT	2	
A.FETCH	2-10	+8-12
A.GETIME	10-94	+0-8
A.LOAD	2-10	+8-12
A.MVCOM	12-16	+0-4
A.PDUMP	10-28	+0-16
A.POST	4-8	+0-8
A.RCB	8-10	
A.RETURN	2-6	
A.SAVE	4	
A.SETIME	6-14	+0-8
A.STXIT	2-14	+4-12
A.TECB	4	
A.WAIT	10-14	+0-4
A.WAITM	8-12	+0-4

Notes:

1. Include the storage requirement for literals only once if the literal is used by more than one macro instruction.

2. +4 bytes per operand for variables.

SUPERVISOR GENERATION:

*Basic:

A.ALLOC
A.ASSGN
A.CCMMN
A.COMMNEX
A.CONFG
A.DVCGEN

A.FOPT
A.IOTAB
A.LUBGEN
A.MCRAS
A.MAPLOWC
A.MAPPUB2
A.PIOCS
A.SEND

A.SGDFCH
A.SGDSK
A.SGSVC

A.SGTCHS
A.SGTCON
A.SGTHAP
A.SGUNCK

A.STDJC
A.SUPVR
A.TRTAB

*Optional

A.SMICR

Note: This macro is part of the MICR program but is required to generate a supervisor with MICR support.

* For a detailed description of storage requirements, refer to Supervisor Main Storage Requirements.

Teleprocessing:

A.IJLQBFRD
A.IJLQDSCT
A.IJLQIP1D
A.IJLQQCBD
A.IJLQSTBD
A.IJLQTSVC
A.IJLQVECD

Note: These macros are part of the QTAM program but are required to generate a QTAM supervisor.

System Generation Macros

A.DOSCHLV
Z.DELETECI
Z.DELETERL
Z.DELETESI
Z.LINKEDIT

MCAR/CCH

CORE IMAGE LIBRARY

Transients

$$BCCHHR	
$$RAST00	CCH and Initial MCAR analysis
$$RAST01	Channel and machine check record builder and recorder
$$RAST02	Channel check analysis
$$RAST03	EFL and MCAR analysis
$$RAST04	Unit record channel check ERP
$$RAST05	Unit record channel check ERP
$$RAST06	3505 and 3525 channel check handler ERP
$$RAST07	Magnetic tape ERP (2400 series)
$$RAST08	Machine check and channel check record writer
$$RAST09	Partition reallocation
$$RAST10	ERP message writer
$$RAST11	Message writer
$$RAST12	Magnetic tape ERP (3400 series)

Transients

Transient routines. Type A transient routines (device error routines) have the prefix $$A. ($ is an alphabetic character in System/370.) Type B transient routines (OPEN, CLOSE, CHKPT, etc) have the prefix $$B.

Device Error Routines and OPEN/CLOSE Phases for Disk Operating System

The following is a list of required system control and common IOCS transients.

$$ANERAA	Unit check/MDR record writer
$$A$SUP1	Supervisor
$$ANERAB	Message writer
$$ANERAC	Statistical counter update for disk and unit record
$$ANERAD	Statistical counter update for tape
$$ANERAE	Message writer
$$ANERAF	RMSR message writer
$$ANERAG	SVC44 record writer
$$ANERAM	Tape-error recovery
$$ANERAN	Tape-error recovery
$$ANERAP	Tape-error recovery
$$ANERAT	Unit check record builder
$$ANERAU	BTAM, MDR, and 3330 Unit check and MDR record builder
$$ANERAV	RMSR device code modifier
$$ANERR	Physical attention routine-cancel delay phase
$$ANERRA	Error recovery monitor
$$ANERRB	Error recovery monitor
$$ANERRC	Error recovery monitor
$$ANERRD	Tape-error recovery
$$ANERRE	Tape-error recovery
$$ANERRF	Tape-error recovery
$$ANERRG	Data cell (2321)-error recovery
$$ANERRH	Data cell (2321)-error recovery
$$ANERRI	Data cell (2321)-error recovery
$$ANERRJ	Data cell (2321)-error recovery
$$ANERRK	Data cell (2321)-error recovery
$$ANERRL	Tape-error recovery
$$ANERRM	Unit check record builder
$$ANERRN	Message writer (Phase 2)
$$ANERRO	Message Writer (Phase 3)
$$ANERRP	Message writer (Phase 4)
$$ANERRQ	Message writer (Phase 5)
$$ANERRR	Message writer (Phase 6)
$$ANERRS	Message Writer (Phase 7)
$$ANERRU	Unit record error recovery
$$ANERRV	Unit record error recovery
$$ANERRY	Physical attention routine
$$ANERRZ	Physical attention routine
$$ANERR0	Physical attention routine
$$ANERR1	Modify communication region
$$ANERR6	2495 Tape Cartridge Reader Error Recovery
$$ANERR7	2495 Tape Cartridge Reader Error Recovery
$$ANERR8	2495 Tape Cartridge Reader Error Recovery
$$ANERSA	3211 Error Recovery
$$ANERSB	MDR record builder (Phase 1)
$$ANERSC	MDR record builder (Phase 2)
$$ANERSD	3505 Card reader and 3525 card punch error recovery
$$ANERSE	2245 error recovery

$$BATTNA	Supervisor, program terminator
$$BATTNB	Supervisor, program terminator
$$BATTNC	Supervisor, initiator
$$BATTND	Supervisor, nonresident attention routine
$$BATTNE	Supervisor, nonresident attention routine
$$BATTNF	Supervisor, nonresident attention routine
$$BATTNG	Supervisor, nonresident attention routine
$$BATTNH	Supervisor, nonresident attention routine
$$BATTNI	Supervisor, foreground initiator
$$BATTNJ	Supervisor, foreground initiator
$$BATTNK	Supervisor, foreground initiator
$$BATTNL	Supervisor, foreground initiator
$$BATTNM	Supervisor, foreground initiator
$$BATTNN	Supervisor, nonresident attention routine
$$BATTNO	Supervisor, nonresident attention routine
$$BATTNP	Supervisor, nonresident attention routine
$$BATTNQ	Supervisor, nonresident attention routine
$$BATTNR	Supervisor, nonresident attention routine
$$BATTNS	Supervisor, nonresident attention routine
$$BATTNT	Process ALTER statement
$$BATTNU	Process DSPLY statement
$$BATTNV	DUMP command processor
$$BATTNW	DUMP command processor
$$BATTNX	DUMP command print routine
$$BATTNY	Supervisor, nonresident attention routine
$$BATTNZ	Supervisor, nonresident attention routine
$$BATTN4	ATTN-LOG-NOLOG processor
$$BCCPT1	System required tape open
$$BCEOV1	Monitor-EOV/EOF
$$BCHKPD	Disk-checkpoint
$$BCHKPE	Disk-checkpoint
$$BCHKPF	Disk-checkpoint
$$BCHKPT	Tape-checkpoint
$$BCHKP2	Tape-checkpoint
$$BCHK3G	Erase gap for logical files
$$BCIS0A	ISFMS - CLOSE
$$BCLOSE	Close monitor
$$BCLOSP	Punch file close
$$BCLOS2	Close
$$BCMT01	Tape EOF/EOV input-forward
$$BCMT02	Tape CLOSE-alternate switching
$$BCMT03	Tape CLOSE input-backward
$$BCMT04	Tape EOV output-forward
$$BCMT05	Tape CLOSE
$$BCMT06	Tape CLOSE-workfiles
$$BCMT07	Tape-alternate switching
$$BDRSTR	Disk restart phase
$$BDUMP	Supervisor, program terminator
$$BDUMPB	Supervisor, program terminator
$$BDUMPD	Supervisor, program terminator
$$BDUMPF	Supervisor, program terminator
$$BENDFF	ISFMS-ENDFL (Load phase 2)
$$BENDFL	ISFMS-ENDFL (Load phase 1)
$$BEOJ	Supervisor, program terminator
$$BEOJ1	Supervisor, program terminator
$$BEOJ2	Supervisor, program terminator
$$BEOJ2A	Supervisor, program terminator
$$BEOJ3	Supervisor, program terminator
$$BEOJ3A	Supervisor, program terminator
$$BEOJ4	Supervisor, program terminator

$$BEOJ5	Supervisor, program terminator
$$BEOJ7	Supervisor, program terminator
$$BERRTN	System required error recovery
$$BFCB	3211 FCB standard load
$$BILSVC	Supervisor, program terminator
$$BINDEX	Cylinder index in core (ISFMS)
$$BJCOPT	Job Control, OPEN Tape routine
$$BJCOP1	Job Control, OPEN Tape routine
$$BLISTV	List VTOC (2311 and 2314/2319 only)
$$BLVTOC	List VTOC (2311, 2314/19 and 3330)
$$BISTIO	System control
$$BMSGWR	Tape open/close
$$BOCPM1	System required message writer
$$BOCPM2	System required message writer
$$BOCPT1	System required OPEN for input tape (Phase 1)
$$BOCPT2	System required OPEN for unlabeled output tape
$$BOCPT3	System required OPEN for labeled output tape
$$BOCPT4	System required OPEN for input tape (Phase 2)
$$BOCP01	System required table, disk DTFs, OPEN Version II
$$BOCP02	System required table, nondisk DTFs, OPEN Version II
$$BOCP11	System required table, tape or disk system DTFs, OPEN Version I
$$BOCP12	System required table, tape system DTFs
$$BODACL	Close Routine-DA
$$BODAIN	Direct access input-OPEN (Input)
$$BODAI1	Open input-DA
$$BODAO1	Direct access-OPEN output Phase 1
$$BODAO2	Direct access-OPEN output Phase 2
$$BODAO3	Direct access OPEN-output Phase 3
$$BODAO4	Direct access OPEN-output Phase 4
$$BODAU1	Direct access-OPEN
$$BODQUE	Dequeue JIB's
$$BODSMW	Message writer
$$BODSPV	Disk VTOC display routine
$$BODSPW	Disk VTOC display routine
$$BOFLPT	DASD file protect
$$BOIS01	ISFMS - OPEN I/O Phase 1
$$BOIS02	ISFMS - OPEN I/O Phase 2
$$BOIS03	ISFMS - OPEN I/O Phase 3
$$BOIS04	ISFMS - OPEN I/O Phase 4
$$BOIS05	ISFMS - OPEN I/O Phase 5
$$BOIS06	ISFMS - OPEN I/O Phase 6
$$BOIS07	ISFMS - OPEN I/O Phase 7
$$BOIS08	ISFMS - OPEN I/O Phase 8
$$BOIS09	Index sequential independent overflow area integrity
$$BOIS10	ISFMS - OPEN
$$BOMRCE	OPEN for OMR and RCE
$$BOMSG1	Message non-abort types
$$BOMSG2	Message writer - abort types
$$BOMSG3	Message writer
$$BOMSG4	Message writer
$$BOMSG5	Message writer
$$BOMSG6	Message writer
$$BOMSG7	Message Writer (ASCII)
$$BOMTOM	Tape OPEN message
$$BOMTOW	Tape OPEN message

$$BOMT01	Tape OPEN input-forward-standard labels (Phase 1)
$$BOMT02	Tape OPEN input-backward-standard labels
$$BOMT03	Tape OPEN output forward-standard label
$$BOMT04	Tape OPEN output-standard labels
$$BOMT05	Tape OPEN I/O-forward/backward nonstandard/unlabeled
$$BOMT06	Tape OPEN workfiles
$$BOMT07	Tape OPEN input-forward-standard labels (Phase 2)
$$BCNVOL	Tape rewriting
$$BOPEN	RMSR tape processor
$$BOPEN1	Open monitor
$$BOPENC	OPENC
$$BOPEND	Disk OPEN-volume ID handler
$$BCPENR	Relocation Phase 1
$$BOPEN2	Open monitor
$$BCPIGN	Open monitor
$$BOPNLB	Locates source statement library. Directory also locates source statement library (for Assembler and COBOL)
$$BCPNR2	Relocation Phase 2
$$BORTV1	ISFMS Open
$$BCRTV2	ISFMS Open
$$BOSDC1	Sequential disk I/O - CLOSE
$$BOSDC2	Sequential disk-close
$$BCSDEV	DTFDA and DTFSD Forced End of Volume
$$BOSDI1	Sequential disk input - OPEN
$$BCSDI2	Sequential disk input - OPEN
$$BOSDI3	Sequential disk input - OPEN
$$BCSDI4	Sequential disk input - OPEN
$$BOSDO1	Sequential disk output Phase 1 - OPEN
$$BCSDO2	Sequential disk output Phase 2 - OPEN
$$BOSDO3	Sequential Disk output Phase 3 - OPEN
$$BCSDO4	Sequential disk output Phase 4 - OPEN
$$BOSDO5	Sequential disk output Phase 5 - OPEN
$$BOSDO6	Sequential disk open - Phase 6
$$BCSDO7	Sequential disk output - OPEN
$$BOSDO8	Sequential disk output - OPEN
$$BCSDC9	Sequential disk output - OPEN
$$BCSDW1	Sequential disk workfiles Phase 1 - OPEN
$$BOSDW2	Sequential disk workfiles Phase 2 - OPEN
$$BOSDW3	Sequential workfile Phase 3 - OPEN
$$BOSD00	Sequential disk OPEN - output Phase 0
$$BOSD01	Sequential disk OPEN - output Phase 1
$$BCSIGN	System open
$$BOUR01	Unit record-OPEN routine
$$BCVDMP	Disk VTOC dump
$$BOWDMP	Disk VTOC dump
$$BO2321	OPEN data cell
$$BPCHK	Supervisor - program terminator
$$BPDAID	Problem Determination
$$BPDUMP	Standard PDUMP monitor
$$BPDUM1	Standard PDUMP
$$BPSW	Supervisor - program terminator
$$BRELSE	Device release
$$BRMSG1	Message writer CHKPT-RSTRT
$$BRMSG2	CHECKPOINT-RESTART message writer
$$BRSTRB	Tape Restart
$$BRSTRT	RESTART message writer - Phase 2
$$BRSTR2	Tape and DASD verify for restart
$$BRSTR3	DASD verify for restart

```
$$BRSTR4        Tape Reposition
$$BSDRUP        Suppress writing on recorder file
$$BSETFF        ISFMS load Phase 2 of SETFL

$$BSETFG        ISFMS load Phase 3 of SETFL
$$BSETFH        ISFMS load Phase 4 of SETFL
$$BSETFL        ISFMS load Phase 1 of SETFL
$$BSETL         ISMOD - SETL
$$BSETL1        ISMOD - SETL
$$BSYSWR        Supervisor, nonresident attention
$$BTERM         Supervisor, program terminator
$$BUCB          3211 UCSB standard load
$$BUFLDR        3211 FCB and UCSB IPL loader
$$BUFLD2        3211 FCB and UCSB IPL loader error routine
```

Standard System Dump

CORE IMAGE LIBRARY

Transients

```
$$BDUMP
$$BDUMPB
$$BDUMPD
$$BDUMPF
$$BPDUMP
$$BPDUM1
```

RELOCATABLE LIBRARY

Modules

```
IJBDMPBS
IJBDMPDS
IJBDMPFS
IJBDMPS
IJBDUMPS
IJBPDMPS
IJBPDUMS
```

Linkage Edit Statements

```
   INCLUDE IJBDUMPS
// EXEC LNKEDT
```

Translating System Dump

CORE IMAGE LIBRARY

Transients

```
$$BDUMP
$$BDUMPB
$$BDUMPD
$$BPDUMP
```

RELOCATABLE LIBRARY

Modules

```
IJBDMPBT
IJBDMPDT
IJBDMPT
IJBDUMPT
IJBPDMPT
```

Linkage Edit Statements

```
   INCLUDE IJBDUMPT
// EXEC LNKEDT
```

ESTVUT

CORE IMAGE LIBRARY

Phase

```
ESTVUT
```

RELOCATABLE LIBRARY

Module

```
IJBTESUT
```

Linkage Edit Statements

For a multiprogramming system:

```
   PHASE ESTVUT,+0
   INCLUDE IJBTESUT
// EXEC LNKEDT
```

For a non-multiprogramming system:

```
   PHASE ESTVUT,S
   INCLUDE IJBTESUT
// EXEC LNKEDT
```

PDAID

CORE IMAGE LIBRARY

Phases

```
PDAID
PDAIDFTT*
PDAIDFTP*
PDAIDFTW*
PDAIDGTT*
PDAIDGTP*
PDAIDGTW*
PDAIDITT*
PDAIDITP*
PDAIDITW*
PDAIDQTT*
PDAIDQTW*
PDAIDTDP*
PDAIDTDT*
PDLIST
```

*Only present in core-image library.

Transient

```
$$BPDAID
```

RELOCATABLE LIBRARY

Modules

```
IJBPDAID
IJBPDLST
```

Linkage Edit Statements

For a non-multiprogramming system:

```
   PHASE PDAID,S
   INCLUDE IJBPDAID
/*
// EXEC LNKEDT
   PHASE PDLIST,S
   INCLUDE IJBPDLST
/*
// EXEC LNKEDT
```

For a multiprogramming system:

```
   PHASE PDAID,+0
   INCLUDE IJBPDAID
/*
// EXEC LNKEDT
   PHASE PDLIST,+0
   INCLUDE IJBPDLST
/*
// EXEC LNKEDT
```

DUMPGEN

CORE IMAGE LIBRARY

Phases

```
DUMPGEN
DUMPGEN1
```

RELOCATABLE LIBRARY

Module

```
IJBDMPGN
```

Linkage Edit Statements

For a non-multiprogramming system:

```
   PHASE DUMPGEN,S
   INCLUDE IJBDMPGN
/*
// EXEC LNKEDT
```

For a multiprogramming system:

```
   PHASE DUMPGEN,+0
   INCLUDE IJBDMPGN
/*
// EXEC LNKEDT
```

LSERV

CORE IMAGE LIBRARY

Phase

```
LSERV
```

RELOCATABLE LIBRARY

Module

```
IJBLSERV
```

Linkage Edit Statements

For a non-multiprogramming system:

```
   PHASE LSERV,S
   INCLUDE IJBLSERV
/*
// EXEC LNKEDT
```

For a multiprogramming system:

```
   PHASE LSERV,+0
   INCLUDE IJBLSERV
/*
// EXEC LNKEDT
```

IJBTRA11
IJBTRG11
IJBTRH11
IJBTRP11
IJBTRT11

3211 Printer Support

CORE IMAGE LIBRARY

Phases

SYSBUFLD
SYSBUFF1

Transients

$$BFCB*
$$BUCB*
$$BUFLDR*
$$BUFLD2*

*Required for IPL if 3211 is present.

RELOCATABLE LIBRARY

Modules

IJBSBUFF
IJBSBUF1

Linkage Edit Statements

```
   PHASE $$BUCB, +0
   INCLUDE IJBTRxnn
// EXEC LNKEDT
```

Where xnn is the train type on your 3211 printer other than the A11 supplied in the core image library.

For a non-multiprogramming system:

```
   PHASE SYSBUFLD,S
   INCLUDE IJBSBUFF
   PHASE SYSBUFF1,S
   INCLUDE IJBSBUF1
// EXEC LNKEDT
```

For a multiprogramming system:

```
   PHASE SYSBUFLD,+0
   INCLUDE IJBSBUFF
   PHASE SYSBUFF1,+0
   INCLUDE IJBSBUF1
// EXEC LNKEDT
```

Number of Phases	Number of Library Blocks		
	2311	2314/2319	3330
1	18	19	21

CONFIGURATION

Figure 1-31 represents the parameters and their specifications of the IBM-supplied supervisor.

Operation	Operand
SUPVR	
CONFG	MODEL=155
SIDJC	
FOPT	PC=YES,OLTEP=NO
PIOCS	BMPX=YES,TAPE=7,DISK=3330
ICTAB	JIB=10,D3420=4,D3330=2
SEND	14336
END	

Figure 1-31. SUPERVISOR (14K)--IBM-Supplied Supervisor for Disk Operating System

CORE IMAGE LIBRARY

Phase

$$A$SUP1

Module 2: Emulating the 1401, 1440, and 1460 on Models 135, 145, and 155

Modular Outline

The 1401/1440/1460 Emulator Program for the IBM System/370 consists of a group of macro instructions and object and load modules that are shipped in the source statement, relocatable, and core image libraries. Two sample programs that test the working of a 1401/1460 program and of a 1440 program under the emulator are contained in the source statement library under the name Z. EMSPL01. The source cards of this program are cataloged under the name Z. EMSPL01S.

PARTITION SIZE REQUIRED

This emulator program requires at least a 20,480 byte partition. For further information, refer to the emulator publication listed in the Preface.

SOURCE STATEMENT LIBRARY

Number of Macros	Number of Library Blocks (Physical Records)
32	4556

Macros

A.ANUM
A.BDIL
A.BIFLAG
A.COMP
A.DBIB

A.DDTF
A.DDUB
A.DIIQCR
A.DIL
A.DSUB

A.EMCNSL
A.EMDISK
A.EMEND
A.EMPNCH
A.EMPTR

A.EMRDR
A.EMSPL01
A.EMSPL01S
A.EMSUP
A.EMTAPE
A.EMULATOR
A.EMVERIFY

A.IIQBR
A.IIQCR
A.IIQDS
A.IIQEI
A.IIQID
A.IIQIU
A.IIQOJ
A.IIQUR
A.MCPU
A.MIC

Delete Statements

```
DELETS A.ANUM
DELETS A.BDIL
DELETS A.BIFLAG
DELETS A.COMP
.
.
.
DELETS A.MIC
```

RELOCATABLE LIBRARY

Number of Modules	Number of Library Blocks (Physical Records)
27	235

Module Name	Bytes of Main Storage
IIQAP	345
IIQBF	250
IIQBY	449
IIQCC	1675
IIQCF	2404
IIQCN	727
IIQCP	422
IIQCS	5216
IIQDB	764
IIQDI	786
IIQDK	2045
IIQEJ	290
IIQEP	755
IIQMC	486
IIQMD	814
IIQMT	629
IIQMTP	741
IIQMW	592
IIQNT	2106
IIQOA	1674
IIQOB	1538
IIQPOST	6005
IIQPRE	7015
IIQSD	390
IIQTP	1829
IIQUR1	1281
IIQVT	3376

Linkage Edit Statements

Tape Preprccessor Program

```
   PHASE phasename,*,NCAUTC
   INCLUDE IIQMTP
   INCLUDE IIQPRE
   ENTRY IIQPRE
/*
// LBLTYP TAPE
// EXEC LNKEDT
```

Tape Postprccessor Program

```
   PHASE phasename,*,NOAUTO
   INCLUDE IIQMTP
   INCLUDE IIQPOST
   ENTRY IIQPOST
/*
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statement

```
DELETR IIQ.ALL
```

CORE IMAGE LIBRARY

	Phases	Blccks
Transients	11	11
Phases	8	9

Transients

```
$$BIIQBD
$$BIIQBS
$$BIIQMW
$$BIIQSD
$$BIIQSS

$$BIIQT1
$$BIIQT2
$$BIIQT3
$$BIIQT4
$$BIIQT5
$$BIIQT6
```

Phases

```
IIQOC
IIQOD
IIQOE
IIQOF

IIQCG
IIQOH
IIQOI
IIQOK
```

Delete Statements

```
DELETC IIQC.ALL
DELETC $$BIIQBD
DELETC $$BIIQBS
DELETC $$BIIQMW
.
.
.
DELETC $$BIIQT6
```

Sample Prcgram

```
Z.EMSPL01,Z.EMSPL01S
```

SAMPLE PROGRAM SUMMARY: Figure 2-1 illustrates the card deck that is punched when the sample programs are retrieved from the source statement library.

```
Cards Retrieved
CATALS Z.EMSPL01,4.0
BKEND Z.EMSPL01
 1400 ckject decks and source decks
BKEND
CATALS Z.EMSPL01S,4.0
BKEND Z.EMSPL01S
 1400 scurce deck
BKEND
```

Figure 2-1. Sample Program Card Deck, 1401/1440/1460 Emulatcr

The sample program lists 20 records on the printer, and it writes out the reccrds cn tape if tape is available and assigned. For a complete descriptiocn of the sample prcgrams, refer to the 1401/1440/1460 Emulator manual listed in the Preface.

Module 3: Emulating the 1410 and 7010 on Models 145 and 155

Modular Outline

The 1410/7010 Emulator Program consists of a group of macro instructions and object and load modules that are shipped in the source statement, relocatable, and core image libraries. A sample program that tests the working of a 1410 and 7010 program under the emulator is contained in the source statement library under the name Z.EMSPL10. The source cards of this program are cataloged under the name Z.EMSPL10S.

PARTITION SIZE

28,672 bytes. For further information, refer to the emulator publication listed in the Preface.

Source Statement Library

Number of Macros	Number of Library Blocks (Physical Records)
31	2639

Macros

A.ANUM
A.BDIL
A.BIFLAG
A.COMP
A.DBIB
A.DDTF
A.DDUB
A.DECCB
A.DIIRCR
A.DIL
A.DSUB
A.EMCNSL
A.EMDISK
A.EMEND
A.EMPNCH
A.EMPTR
A.EMRDR
A.EMSPL10
A.EMSPL10S
A.EMSUP
A.EMTAPE
A.EMULATOR
A.EMVERIFY
A.IIRCR
A.IIRDCC
A.IIREI
A.IIRID
A.IIRIU
A.IIROJ
A.MCPU
A.MIO

Delete Statements

```
DELETS A.ANUM
DELETS A.BDIL
DELETS A.BIFLAG
.
.
.
A.MIO
```

RELOCATABLE LIBRARY

Number of Modules	Number of Library Blocks (Physical Records)
25	238

Module Name	Bytes of Main Storage
IIRBF	250
IIRCC	1675
IIRCP	1555
IIRDB	772
IIRDK	4114
IIRDI	2236
IIRDS	656
IIREJ	290
IIRFP	2718
IIRIS	976
IIRMI	1160
IIRMT	629
IIRMTP	741
IIRMW	596
IIRNT	2106
IIRCA	1706
IIROB	1538
IIRPCST	6013
IIRPR	442
IIRPRE	7023
IIRSD	366
IIRST	366
IIRTP	1931
IIRUR	1842
IIRVT	3375

Linkage Edit Statements

Tape Preprocessor Program

```
   PHASE phasename,*,NCAUTC
   INCLUDE IIRMTP
   INCLUDE IIRPRE
   ENTRY IIRPRE
/*
// LBLTYP TAPE
// EXEC LNKEDT
```

Tape Postprocessor Program

```
   PHASE phasename,*,NOAUTO
   INCLUDE IIRMTP
   INCLUDE IIRPOST
   ENTRY IIRPOST
/*
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statement

```
DELETR IIR.ALL
```

CORE IMAGE LIBRARY

	Phases	Blocks
Transients	10	10
Phases	8	9

Transients

```
$$BIIRBD
$$BIIRBS
$$BIIRMW
$$BIIRSD
$$BIIRT1

$$BIIRT2
$$BIIRT3
$$BIIRT4
$$BIIRT5
$$BIIRT6
```

Phases

```
IIROC
IIRCD
IIRCE
IIROF

IIROG
IIROH
IIRCI
IIROK
```

Delete Statements

```
DELETC IIRC.ALL
DELETC $$BIIRBD
DELETC $$BIIRBS
DELETC $$BIIRMW
.
.
.
DELETC $$BIIRT6
```

Sample Program

Z.EMSPI10, Z.EMSPL10S

SAMPLE PROGRAM SUMMARY: Figure 3-1 illustrates the card deck is punched when the sample program is retrieved from the source statement library.

```
Cards Retrieved
CATALS Z.EMSPI10,4.0
BKEND Z.EMSPL10
 1400 object decks and source decks
BKEND
CATALS Z.EMSPL10S,4.0
BKEND Z.EMSPL10S
 1400 source deck
BKEND
```

Figure 3-1. Sample Program Card Deck, 1410/7010 Emulator

The sample program lists 20 records on the printer. If a tape is available and assigned, the sample program writes out the records on tape. For a complete description of the sample programs, refer to the 1410/7010 Emulator manual listed in the Preface.

Module 4: Compiler IOCS Modules--370N-IO-476

Modular Outline

Modular Index

Module 4 indicates the names of the preassembled modules shipped by IBM in the relocatable library, their sizes in bytes, their external storage requirements in library blocks, and the compilers that create linkage for their use. No IOCS modules are required for FORTRAN or Basic FORTRAN. An x in a column headed by a compiler indicates that the particular module may be linked by that compiler to the problem program. For example, an x in the columns headed by COBOL and PL/I (D) indicates that COBOL and PL/I (D) link the module so identified. For assembler language programs, the user can preassemble IOCS modules as described in the DOS Supervisor and I/O Macros publication. Note that separate modules are not assembled for DTFCN or DTFSR.

The preassembled modules named in this module can be used by any other problem program whenever applicable.

	Modules	Blocks
	223	1247
Am. Nat. St. COBOL	132	864
Am. Nat. St. COBOL & COBOL	158	1053
Am. Nat. St. COBOL & RPG	159	1032
Am. Nat. St. COBOL & PL/I (D)	163	970
Am. Nat. St. COBOL, COBOL & PL/I (D)	341	1134
Am. Nat. St. COBOL, COBOL & RPG	171	1081
Am. Nat. St. COBOL, PL/I (D) & RPG	186	1127
Am. Nat. St. COBOL, PL/I (D), RPG & COBOL	194	1161
COBOL	77	548
COBOL & PL/I (D)	107	662
COBOL & RPG	93	595
COBOL, PL/I (D) & RPG	116	675
PL/I (D)	70	400
RPG	43	298
RPG & PL/I (D)	98	612
Required IOCS Modules	27	84

Note: Certain preassembled IOCS modules are required when cataloging IBM components to the core image library. Do not delete these modules from the relocatable library until after all the IBM components you desire have been catalcged to the core image library. The modules are found in Required IOCS Modules.

Relocatable Module Naming Conventions

Each module name begins with a 3-character prefix and consists of a 5-character field corresponding to the option permitted in generation of the module. The following 3-character prefixes identify the preassembled IOCS modules shipped by IBM:

IJC I/O Card (CDMOD)

IJD I/O Printer (PRMOD)

IJF I/O Magnetic Tape (MTMOD)

IJG Sequential Direct Access (SDMOD)

IJH Index Sequential Direct Access (ISMOD)

IJI Direct Access Method (DAMOD)

IJJ Device Independent Access Method (DIMOD)

CDMOD

CDMOD name = IJCabcde

a = F RECFORM=FIXUNB (always for INPUT and CMBND files)
  = U RECFORM=UNDEF
  = V RECFORM=VARUNB

b = A CTLCHR=ASA (not specified CMBND)
  = C CONTROL=YES
  = Y CTLCHR=YES
  = Z neither CTLCHR nor CCNTRCL is specified

c = B RDONLY=YES and TYPEFLE=CMBND
  = C TYPEFLE=CMBND
  = H RDONLY=YES and TYPEFLE=INPUT
  = I TYPEFLE=INPUT
  = N RDONLY=YES and TYPEFLE=OUTPUT
  = O TYPEFLE=OUTPUT

d = B WORKA=YES and IOAREA2=YES
  = I IOAREA2=YES
  = W WORKA=YES
  = Z neither WORKA nor IOAREA2 is specified

For CMBND Files

d = W WORKA=YES
  = Z WORKA is not specified

e = 0 DEVICE=2540
  = 1 DEVICE=1442
  = 2 DEVICE=2520 or DEVICE=2596
  = 3 DEVICE=2501
  = 4 DEVCIE=2540 and CRDERR is specified
  = 5 DEVICE=2520 and CRDERR is specified
  = 6 DEVICE=3505
  = 7 DEVICE=3525 and FUNC cmitted, FUNC=R cr FUNC=P
  = A DEVICE=3525 and FUNC=RP
  = B DEVICE=3525 and FUNC=RW
  = C DEVICE=3525 and FUNC=PW
  = D DEVICE=3525 and FUNC=I
  = E DEVICE=3525 and FUNC=RPW

CDMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes cf Main Stcrage
IJCFAOI0					116
IJCFAOI1	x	x		x	152
IJCFAOI2	x	x			156
IJCFAOI4	x	x		x	264
IJCFAOI5	x			x	248
IJCFAOZ0					92
IJCFAOZ1	x	x		x	128
IJCFAOZ2		x			132
IJCFAOZ4	x	x		x	248
IJCFAOZ5	x			x	224
IJCFCCZ0			x		218
IJCFCCZ1			x		132

CDMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes of Main Storage
IJCFCCZ2			x		218
IJCFCIZ0			x		158
IJCFCIZ1			x		132
IJCFCIZ2			x		158
IJCFYOI0			x		96
IJCFYOI1		x	x		132
IJCFYOI2		x	x		136
IJCFYOI4		x			248
IJCFYOZ0			x		72
IJCFYOZ1		x	x		108
IJCFYOZ2		x	x		112
IJCFYOZ4		x			224
IJCFZII0	x	x	x	x	136
IJCFZII1	x	x	x	x	140
IJCFZII2	x	x	x	x	136
IJCFZII3	x	x	x	x	136
IJCFZIZ0	x	x		x	104
IJCFZIZ1	x	x		x	108
IJCFZIZ2	x	x		x	104
IJCFZIZ3	x	x	x	x	104
IJCFZOI1	x	x		x	116
IJCFZOI2		x			124
IJCFZOI4	x	x		x	232
IJCFZOI5	x			x	216
IJCFZOZ1	x	x		x	74
IJCFZOZ2		x			80
IJCFZOZ4	x	x		x	208
IJCFZOZ5	x			x	192
IJCUAOI1				x	164
IJCUAOI4				x	280
IJCUAOI5				x	264
IJCUAOZ1				x	140
IJCUAOZ4				x	256
IJCUAOZ5				x	240
IJCUZOI1				x	124
IJCUZOI4				x	240
IJCUZOI5				x	224
IJCUZOZ1				x	104
IJCUZOZ4				x	224
IJCUZOZ5				x	200
IJCVAOI1				x	180
IJCVAOI4				x	296
IJCVAOI5				x	280
IJCVAOZ1				x	156
IJCVAOZ4				x	272
IJCVAOZ5				x	256
IJCVZOI1				x	152
IJCVZOI4				x	272
IJCVZOI5				x	248
IJCVZOZ1				x	128
IJCVZOZ4				x	248
IJCVZOZ5				x	224

DAMOD

DAMOD name = IJIabcde

a = B RECFORM=UNDEF (handles both UNDEF and FIXUNB)
  = F RECFORM=FIXUNB
  = S RECFORM=SPNUNB
  = V RECFORM=VARUNB

b = A AFTER=YES
  = Z AFTER is not specified

c = E IDLOC=YES and FEOVD=YES
  = I IDLOC=YES
  = R FEOVD=YES
  = Z neither is specified

d = H ERREXT=YES and RELTRK=YES
  = P ERREXT=YES
  = R RELTRK=YES
  = Z neither is specified

e = W HOLD=YES and RDONLY=YES
  = X HOLD=YES
  = Y RDONLY=YES
  = Z neither is specified

DAMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes of Main Storage
IJIBAIRZ				x	1632
IJIBAIZZ	x			x	1220
IJIBAZRZ				x	1220
IJIBAZZZ	x			x	1012
IJIBZIRZ				x	1416
IJIBZIZZ	x			x	1004
IJIBZZRZ				x	1004
IJIBZZZZ	x		x	x	796
IJIFAIRZ				x	1532
IJIFAIZZ	x			x	1120
IJIFAZRZ				x	1148
IJIFAZZZ	x	x		x	940
IJIFZIRZ				x	1228
IJIFZIZZ	x			x	816
IJIFZZRZ				x	848
IJIFZZZZ	x	x		x	636
IJISAIRZ				x	3684
IJISAIZZ				x	3640
IJISAZRZ				x	3324
IJISAZZZ				x	3324
IJISZIRZ				x	2680
IJISZIZZ				x	2636
IJISZZRZ				x	2320
IJISZZZZ				x	2320

DIMOD

```
DIMOD name = IJJabcde

a = F    RECFORM=FIXUNB

b = C    always

c = B    TYPEFLE=OUTPUT (processes both input and output)
  = I    TYPEFLE=INPUT

d = I    IOAREA2=YES
  = Z    IOAREA2 is not specified

e = C    RDONLY=YES
  = D    RDONLY is not specified
```

DIMOD Names	PL/I (D)	Bytes of Main Storage
IJJFCBID	x	775
IJJFCBZD	x	691
IJJFCIID	x	386
IJJFCIZD	x	326

Note: See Required IOCS Modules before deleting modules with the IJJ prefix.

ISMOD

```
ISMOD name = IJHabcde

a = A  RECFORM=BOTH and IOROUT=ADD or ADDRTR
  = B  RECFORM=FIXBLK and IOROUT=ADD or ADDRTR
  = U  RECFORM=FIXUNB and IOROUT=ADD or ADDRTR
  = Z  RECFORM is not specified and IOROUT=LOAD or RETRVE

b = A  IOROUT=ADDRTR
  = I  IOROUT=ADD
  = L  IOROUT=LOAD
  = R  IOROUT=RETRVE

c = B  TYPEFLE=RANSEQ
  = G  IOAREA2=YES and TYPEFLE=SEQNTL or IOROUT=LOAD
  = R  TYPEFLE=RANDOM
  = S  TYPEFLE=SEQNTL
  = Z  neither is specified and IOROUT=LOAD or ADD

d = B  CORINDX=YES and HOLD=YES
  = C  CORINDX=YES
  = O  HOLD=YES
  = Z  neither specified

e = F  CORDATA=YES, ERREXT=YES, and RDONLY=YES
  = G  CORDATA=YES and ERREXT=YES
  = O  CORDATA=YES and RDONLY=YES
  = P  CORDATA=YES
  = S  ERREXT=YES and RDONLY=YES
  = T  ERREXT=YES
  = Y  RDONLY=YES
  = Z  nothing is specified
```

ISMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes of Main Storage
IJHAABCP				x	
IJHAABCZ				x	
IJHAABZP				x	
IJHAABZZ	x		x		4802
IJHAARCP	x	x		x	4570
IJHAARCZ	x	x		x	4198
IJHAARZP	x	x		x	4288
IJHAARZZ	x	x	x	x	3916
IJHAASZZ	x		x		4090
IJHAIZZZ	x		x		3196
IJHBABCP				x	5208
IJHBABCZ				x	4836
IJHBABZP				x	4930
IJHBABZZ	x		x	x	4558
IJHBARCP	x	x		x	4326
IJHBARCZ	x	x		x	3950
IJHBARZP		x		x	4048
IJHBARZZ	x	x	x	x	3672
IJHBASZZ	x		x		3844
IJHBIZZZ	x		x		2950
IJHUABCP				x	4940
IJHUABCZ				x	4624
IJHUABZP				x	4658
IJHUABZZ	x		x	x	4342
IJHUARCP	x	x		x	4058
IJHUARCZ	x	x		x	3738
IJHUARZP	x	x		x	3776
IJHUARZZ	x	x	x	x	3456
IJHUASZZ	x		x		3630
IJHUIZZZ	x		x		2736
IJHZLZZ	x	x	x	x	823
IJHZRBCZ				x	2550
IJHZRBZZ	x		x	x	2374
IJHZRRCZ	x	x		x	1668
IJHZRRZZ	x	x	x	x	1492
IJHZRSZZ	x	x	x	x	1234

MTMOD

MTMOD name = IJFabcde

a = F RECFORM=FIXUNB or FIXBLK
  = S RECFORM=SPNUNB or SPNBLK
  = U RECFORM=UNDEF
  = V RECFORM=VARUNB or VARBLK

  = N RECFORM=UNDEF and ASCII=YES
  = R RECFORM=VARUNB or VARBLK, and ASCII=YES
  = X RECFORM=FIXUNB or FIXBLK, and ASCII=YES

b = B READ=BACK
  = Z READ=FORWARD, or READ is not specified

c = C CKPTREC=YES
  = Z CKPTREC is not specified

d = W WORKA=YES is specified
  = Z WORKA is not specified

e = M ERREXT=YES and RDONLY=YES
  = N ERREXT=YES
  = Y RDONLY=YES
  = Z neither is specified

MTMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes of Main Storage
IJFFBCZZ	x				904
IJFFBZZN				x	880
IJFFBZZZ		x			784
IJFFZCZZ			x	x	784
IJFFZZZZ		x		x	688
IJFSZZWN				x	2104
IJFUBCZZ	x				696
IJFUBZZZ		x			576
IJFUZZZN				x	656
IJFUZZZZ		x		x	560
IJFVBCWZ	x				1128
IJFVBCZZ	x				1000
IJFVZCWZ			x		1064
IJFVZZZN				x	920
IJFVZZZZ		x		x	824

Name list for workfile type modules (TYPEFLE=WORK):

MTMOD name = IJFabcde

a = W always

b = E ERROPT=YES
  = Z ERROPT is not specified

c = N NOTEPNT=YES
  = S NOTEPNT=POINTS
  = Z NOTEPNT is not specified

d = Z always

e = M ERREXT=YES and RDONLY=YES
  = N ERREXT=YES
  = Y RDONLY=YES
  = Z neither is specified

System I/O Modules

IJFWEZZZ
IJFWZNZZ
IJFWZZZZ

Note: See Required IOCS Modules before deleting modules with the IJF prefix.

PRMOD

PRMOD name = IJDabcde

a = F RECFORM=FIXUNB
  = V RECFORM=VARUNB
  = U RECFORM=UNDEF

b = A CTLCHR=ASA
  = Y CTLCHR=YES
  = C CONTROL=YES
  = S STLIST=YES
  = Z neither CTLCHR nor CONTROL nor STLIST is specified

c = B ERROPT=YES (ERROPT=name in DTFPR) and PRINTOV=YES
  = P PRINTOV=YES, DEVICE is not a 3525, and ERROPT is not specified
       (ERROPT=RETRY or omitted in DTFPR)
  = I PRINTOV=YES, DEVICE=3525, and FUNC=W T or omitted
  = F PRINTOV=YES, DEVICE=3525, and FUNC=RW T
  = C PRINTOV=YES, DEVICE=3525, and FUNC=PW T
  = D PRINTOV=YES, DEVICE=3525, and FUNC=RPW T
  = Z neither ERROPT (ERROPT=RETRY or omitted in DTFPR) nor PRINTOV is
       specified, and DEVICE is not a 3525
  = O PRINTOV=YES is not specified, DEVICE=3525 and FUNC=W T or omitted
  = R PRINTOV=YES is not specified, DEVICE=3525 and FUNC=RW T
  = S PRINTOV=YES is not specified, DEVICE=3525 and FUNC=PW T
  = T PRINTOV=YES is not specified, DEVICE=3525 and FUNC=RPW T
  = E ERROPT=YES (ERROPT=name in DTFPR) and PRINTOV=YES is not specified

d = I IOAREA2=YES
  = Z IOAREA2 is not specified

e = V RDONLY=YES and WORKA=YES
  = W WORKA=YES
  = Y RDONLY=YES
  = Z neither is specified

PRMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes of Main Storage
IJDFAPIZ	x			x	280
IJDFAPZZ	x			x	268
IJDFAZIZ		x			220
IJDFAZZZ		x			196
IJDFYPIZ				x	152
IJDFYPZW	x		x		146
IJDFYPZZ	x			x	140
IJDFYZIZ		x			96
IJDFYZZZ		x			72
IJDFZPIZ	x	x		x	152
IJDFZPZZ	x	x		x	118
IJDUAPIZ				x	292
IJDUAPZZ				x	280
IJDUYPIZ				x	164
IJDUYPZZ				x	152
IJDUZPIZ				x	160
IJDUZPZZ				x	148
IJDVAPIZ				x	312
IJDVAPZZ				x	296
IJDVYPIZ				x	184
IJDVYPZZ				x	168
IJDVZPIZ				x	188
IJDVZPZZ				x	172

SDMOD

SDMODxx name = IJGabcde

a = C RECFORM=FIXUNB or FIXBLK and HOLD=YES
  = F RECFORM=FIXUNB or FIXBLK and HOLD is nct specified
  = P RECFORM=SPNUNB or SPNBLK and HOLD=YES
  = Q RECFORM=SPNUNB or SPNBLK and HOLD is nct specified
  = R RECFORM=UNDEF and HOLD=YES
  = S RECFORM=VARUNB or VARBLK and HOLD=YES
  = U RECFORM=UNDEF and HOLD is nct specified
  = V RECFORM=VARUNB or VARBLK and HOLD is nct specified

b = I SDMODxI
  = O SDMODxO
  = U SDMODxU

c = C ERROPT=YES and ERREXT=YES
  = E ERROPT=YES
  = Z neither is specified

d = M TRUNCS=YES and FEOVD=YES
  = T TRUNCS=YES
  = W FEOVD=YES
  = Z neither is specified

e = B CONTROL=YES and RDONLY=YES
  = C CONTROL=YES
  = Y RDONLY=YES
  = Z neither is specified

SDMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes of Main Storage
IJGFIETZ	x	x			746
IJGFIEWZ				x	670
IJGFIEZZ	x	x			614
IJGFIZZZ	x		x		470
IJGFOEWZ				x	718
IJGFOEZZ		x			630
IJGFOZZZ	x		x		566
IJGFUETZ		x			1102
IJGFUEWZ				x	1054
IJGFUEZZ		x			998
IJGFUZZZ	x		x		834
IJGQIEWZ				x	1157
IJGQIEZZ				x	1113
IJGQOEWZ				x	2409
IJGQOEZZ				x	2317
IJGQUEWZ				x	2662
IJGQUEZZ				x	2598
IJGUIEWZ				x	685
IJGUIEZZ	x	x			641
IJGUIZZZ	x				541
IJGUOEWZ				x	801
IJGUOEZZ		x			721
IJGUOZZZ	x				653
IJGUUEWZ				x	1153
IJGUUEZZ		x			1097

SDMOD Names	COBOL	PL/I (D)	RPG	Am. Nat. St. COBOL	Bytes of Main Storage
IJGUUZZZ	x				949
IJGVIEWZ				x	785
IJGVIEZZ	x	x			741
IJGVIZZZ	x		x		637
IJGVOEWZ				x	1229
IJGVOEZZ		x			1137
IJGVOZZZ	x		x		1065
IJGVUEWZ				x	1346
IJGVUEZZ		x			1282
IJGVUZZZ	x		x		1106

NAME LIST FOR WORKFILE TYPE MODULES (TYPEFLE=WORK)

SDMODxx name = IJGabcde

a = T SDMODW specifies HOLD=YES
  = W SDMODW does not specify HOLD=YES

b = C ERROPT=YES and ERREXT=YES
  = E ERROPT=YES
  = Z neither is specified

c = N NOTEPNT=YES
  = R NOTEPNT=POINTRW
  = Z NOTEPNT is not specified

d = C CONTROL=YES
  = Z CONTROL is not specified

e = T RDONLY=YES and UPDATE=YES
  = U UPDATE=YES
  = Y RDONLY=YES
  = Z neither is specified

System I/O Modules

IJGWEZZU
IJGWEZZZ
IJGWZNZZ
IJGWZRZZ

Note: See Required IOCS Modules before deleting modules with the IJG prefix.

Required IOCS Modules

The following preassembled IOCS modules are required when cataloging and/or linkage editing IBM components to the core image library.

Module Names	Bytes of Main Storage
IJFWEZZZ	280
IJFWZNZZ	424
IJFWZZZZ	232
IJGFIETZ	746
IJGWEZZU	868
IJGWEZZZ	796
IJGWZNZZ	902
IJGWZRZZ	840
IJJCPA1N	621
IJJCPDV	533
IJJCPDV1	477
IJJCPDV2	214
IJJCPD0	569
IJJCPD0N	513
IJJCPD1	501
IJJCPD1N	445
IJJCPD2	282
IJJCPD3	202
IJJCPV	343
IJJCPV1	271
IJJCPV2	68
IJJCP0	389
IJJCP0N	317
IJJCP1	311
IJJCP1N	239
IJJCP2	128
IJJCP3	58

Module 5: Direct Access--370N-IO-454

Modular Outline

Number of Macros	Source Statement Library Blocks
3	883

SOURCE STATEMENT LIBRARY

File Definition Macros

A.DAMOD
A.DAMODV
A.DTFDA

STORAGE REQUIREMENTS:

Figure 5-1 defines the storage requirements for DAMOD. The following are the storage requirements for DTFDA.

DTFDA (Define The File: Direct Access Device) Table Requirements

RECFORM=FIXUNB requires 205-225 bytes, depending upon imperative macros used in the DTF.

VERIFY=YES requires 40-80 bytes, depending upon imperative macros used in the DTF.

AFTER=YES requires 80 additional bytes.

RECFORM=SPNUNB requires 282-378 bytes, depending upon the imperative macros used in the DTF.

VERIFY=YES requires 64-72 additional bytes, depending upon the imperative macros used in the DTF.

AFTER=YES requires 88 additional bytes.

RECFORM=UNDEF requires 265-285 bytes, depending upon the imperative macro used in the DTF.

VERIFY=YES requires 40-80 additional bytes, depending upon the imperative macros used in the DTF.

AFTER=YES requires 16 additional bytes.

RECFORM=VARUNB requires 216-330 bytes, depending upon the imperative macros used in the DTF.

VERIFY=YES requires 64-72 additional bytes, depending upon the imperative macros used in the DTF.

AFTER=YES requires 88 additional bytes.

Relative addressing increases the size of the DTF 60-80 bytes plus 8 bytes per extent. (See DOS Supervisor and I/O Macros .)

DAMOD (Direct Access Device Module)

RECFORM=	Basic Module	IDLOC	Formatting Module: AFTER	Formatting Module: AFTER and IDLCC	RELTRK	HOLD	ERREXT
FIXUNB	636	+180	+304	+484	+212	+84	+28
UNDEF	796	+208	+216	+426	+208	+88	+28
VARUNB	686	+210	+298	+508	+226	+84	+34
SPNUNB	2320	+316	+1004	+1320	---	+104	+32

Notes:

1. Basic Module includes coding to handle either FIXUNB or UNDEF records and the WRITEKY, READKEY, READID, WRITEID, SRCHM, VERIFY, and CCNTRCL functions.

2. AFTER includes coding to create the file and to handle the RZERO option.

3. IDLOC includes coding to return the record identifier to the user in a location he specifies.

4. Specification of trailer label processing in the DTF increases the size of each module by 50 ± 20 bytes.

5. RDONLY=YES changes the size of the module -50 to +50 bytes. In addition, the user's program must provide a 72-byte save area each time the module is reentered.

6. RELTRK may be specified for SPNUNB records. This specification requires no additional bytes.

Figure 5-1. DAMOD--Main Storage Requirements

Module 6: Index Sequential--370N-IO-457

Modular Outline

Number of Macros	Source Statement Library Blocks
19	1772

SOURCE STATEMENT LIBRARY

Macros

FILE DEFINITION

A.DTFIS
A.DTFIS1
A.DTFIS2
A.DTFIS3
A.ISMOD

A.ISMOD0
A.ISMOD1
A.ISMOD2
A.ISMOD3
A.ISMOD4

A.ISMOD5
A.ISMOD6
A.ISMOD7
A.ISMOD8
A.ISMOD9

IMPERATIVE

A.ENDFL
A.ESETL
A.SETFL
A.SETL

Storage Requirements

Figure 6-1 defines the storage requirements for DIMOD. The storage requirements for DTFIS are:

DTFIS (Define The File Indexed Sequential) Table Requirements:

1. IOROUT=LOAD requires 248 bytes plus 4 bytes per disk extent specified, plus 8 bytes for IOAREA2.

2. IOROUT=ADD requires 530 bytes plus 4 bytes per disk extent specified plus KL (the length of the key).

3. IOROUT=RETRVE requires 276 bytes plus 4 bytes per disk extent specified, when TYPEFLE=SEQNTL.

4. IOROUT=RETRVE requires 292 bytes plus 4 bytes per disk extent specified when TYPEFLE=RANDOM or RANSEQ.

5. IOROUT=ADDRTR requires 548 bytes plus 4 bytes per disk extent specified plus KL (the length of the key).

6. IOROUT=ADDRTR, TYPEFLE=RANDOM, INDAREA=name, and INDSIZE=n require 572 bytes plus 4 bytes per disk extent.

ISMOD (Indexed Sequential Module)

	IOROUT=									
	LOAD				ADD					
		ERREXT	IOAREA2	ERREXT ICAREA2		CORDATA	ERREXT	HOLD	CORDATA ERREXT	HOLD CORDATA ERREXT
RECFORM=										
FIXUNB					2608	+184	+476	+194	+66C	+854
FIXBLK					2822	+272	+498	+186	+770	+956
BOTH	823	+224	+212	+436	3068	+246	+428	+210	+674	+884

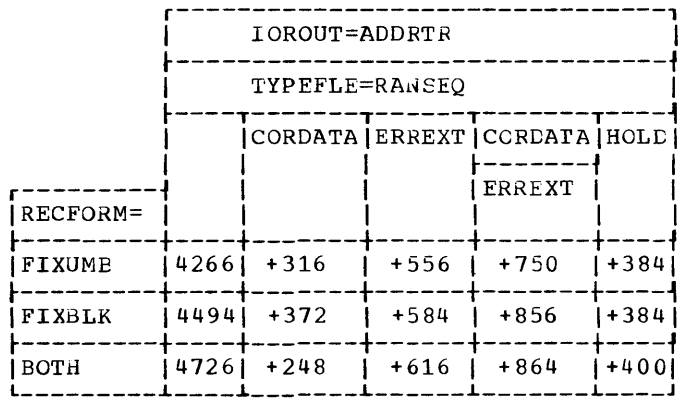
Figure 6-1. ISMOD Main Storage Requirements (Part 1 of 5)

	IOROUT=RETRVE									
	TYPEFLE=									
	RANDOM			SEQNTL				RANSEQ		
		ERREXT	HOLD		ERREXT	IOAREA2	HOLD		ERREXT	HOLD
RECFORM										
FIXUNB										
FIXBLK										
BOTH	1304	+302	+156	1326	+246	+836	+104	2186	+332	+236

Figure 6-1. ISMOD Main Storage Requirements (Part 2 of 5)

	IORCUT=ADDRTR									
	TYPEFLE=									
	RANDOM					SEQNTL				
		CORDATA	ERREXT	CORDATA ERREXT	HOLD		CCRDATA	IOAREA2	HOLD	ERREXT
RECFORM										
FIXUNB	3252	+320	+534	+718	+304	3502	+94	+386	+278	+510
FIXBLK	3468	+376	+554	+826	+304	3716	+272	+386	+278	+528
BOTH	3712	+372	+588	+834	+320	3962	+274	+386	+294	+558

Figure 6-1. ISMOD Main Storage Requirements (Part 3 of 5)



	IOROUT=ADDRTR			
	TYPEFLE=RANSEQ			
RECFORM=	CORDATA	ERREXT	CORDATA ERREXT	HOLD
FIXUNB	4266 +316	+556	+750	+384
FIXBLK	4494 +372	+584	+856	+384
BOTH	4726 +248	+616	+864	+400

Figure 6-1. ISMOD Main Storage Requirements (Part 4 of 5)

Note 1: When RECFORM=BOTH is specified, the module processes FIXUNB and FIXBLK records.

Note 2: For CORINDX = YES, add 212 bytes.

Note 3: RDONLY=YES changes the module size by ±50 bytes with the following exceptions. When IORCUT=ADD or IOROUT=ADDRTR, the module changes in size +60 to +100 bytes. In addition, the user's program must provide a 72-byte save area each time the module is reentered, regardless of function.

Figure 6-1. ISMOD Main Storage Requirements (Part 5 of 5)

Module 7: Magnetic Reader--370N-IO-477

Modular Outline

Libraries					
Core Image				Source Statement	
	Blocks				
Transients	2311	2314/2319	3330	Macros	Blocks
5	5	5	5	7	479

CORE IMAGE LIBRARY

Transients

$$ANERRT Error Recovery Procedure
$$ANERRW Error Recovery Procedure
$$BCMR01 CLOSE
$$BMMR20 Message Writer
$$BCMR01 OPEN

SOURCE STATEMENT LIBRARY

Macros

FILE DEFINITION

A.DTFMR
A.MRMOD

IMPERATIVE

A.DISEN
A.LITE

SYSTEM GENERATION

A.SMICR

Refer to system generation macros in the DOS System Control section.

SAMPLE PROBLEMS

Z.MCR1
Z.MCR2

STORAGE REQUIREMENTS

DTFMR (Define The File Magnetic Character Reader) Table Requirements:

If ADDRESS=DUAL is specified, the table requires 264 bytes.

If ADDRESS=DUAL is not specified, the table requires 250 bytes.

MRMOD (Magnetic Character Reader Module)

If ADDRESS=DUAL is specified, the module requires 1,050 bytes.

If ADDRESS=DUAL is not specified, the module requires 946 bytes.

Sample Problems

	Disk Extent Number			
Phase	1	2	3	4
ASSEMBLE	SYSLNK	SYS001	SYSCC2	SYS003
LINKEDIT	SYSLNK	SYS001	-	-
EXECUTE	-	-	-	-

Figure 7-1. MICR--Sample Problem File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.MCR1 BKEND Z.MCR1 MCR1 Source Deck (71 Cards) BKEND	$477	CC01-0071
CATALS Z.MCR2 BKEND Z.MCR2 MCR2 Source Deck (115 Cards) BKEND	$477	CC01-0115

Figure 7-2. MICR--Sample Problem Card Decks

SAMPLE PROBLEM SUMMARY:

Figure 7-1 defines the files required for the MICR sample problems. Figure 7-2 shows the sample problem card decks retrieved from the source statement library.

The first 1419 Magnetic Character Reader sample problem processes 500 documents from one magnetic character reader using GET logic. The documents are read into pocket 3 in groups of approximately 50. After each group is read, the 1419 Magnetic Character Reader is disengaged, the batch number is updated, and the pocket light is turned on. All documents are listed on the printer. If the Selective Tape List

feature is present on the printer, the documents are listed on the leftmost tape. The controls on the magnetic character reader that must be pressed, if present, are: BATCH NUMBER ON, PROG SORT, and at least one field for the VALIDITY CHECK & READ OUT control.

The second 1419 Magnetic Character Reader sample problem processes 250 documents from each of two magnetic character readers, (both of the same type: both with a single address adapter or both with a dual address adapter), using READ, CHECK, and WAITF logic. The only controls that must be pressed are PROG SORT and at least one field for the VALIDITY CHECK, & READ OUT control. Documents from one reader are selected into pockets one or zero depending on whether or not the selected field is present. Documents read in error are rejected, and all data is listed on a printer. If the Selective Tape List feature is present on the printer, data is printed on the leftmost tape.

Documents from the second reader are selected according to a digit in the field read and printed on SYSLST. If manual intervention is required on the second reader, a message is printed on SYSLOG. The message is:

INTERVENTION REQUIRED ON FILE2.

The program names are Z.MCR1 and Z.MCR2. The 1419 sample problems support the 1255/1259.

Instructions show how to remove the following optional features:

1. Dual addressing adapter feature
2. Batch numbering
3. Pocket lights
4. Selective Tape List Feature (Printer)

SYSLOG Output Is:

```
BG // JOB MCR SAMPLE PROBLEMS
BG // PAUSE   END OF MCR1 SAMPLE PROBLEM
BG
EG  INTERVENTION REQUIRED ON FILE2
BG EOJ MCR
EG // PAUSE   END OF MCR SAMPLE PROBLEMS
```

SYSLST Output Summary:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor input diagnostics
- Linkage editor storage map
- Document data listings

Module 8: Magnetic Tape--370N-IO-456

Modular Outline

Source Statement Library

Source Statement Library	
Macros	Blocks
2	794

File Definition Macros

A.DTFMT
A.MTMOD

STORAGE REQUIREMENTS

Figures 8-1 and 8-2 define the storage requirements for DTFMT and MTMOD.

DTFMT (Define The File: Magnetic Tape) Table Requirements

TYPEFLE=WORK requires 48 bytes per workfile.

The table requirements for INPUT and OUTPUT files are:

TYPEFLE=	RECFORM =	Basic Size Without STDLABELS	ERROPT, ERREXT (Without STDLABELS)	Basic Size With STDLABELS	ERROPT, ERREXT (With STDLABELS)
INPUT	FIXUNB or FIXBLK	96	*	112	*
	VARUNB or VARBLK	109	*	128	*
	UNDEF	92	*	108	*
	SPNUNB	132	*	132	*
OUTPUT	FIXUNB or FIXBLK	86	+10	104	+4
	VARUNB or VARBLK	98	+10	116	+4
	UNDEF	84	+4	100	+4
	SPNUNB	132	*	132	*

* Included in basic Size of Module.

Figure 8-1. DTFMT--Table Requirements

MTMOD (Magnetic Tape Module)

EBCDIC MAGNETIC TAPE MODULE						
		INDEPENDENT OPTIONS				
RECFORM=	BASIC MODULE	WORKA= YES	CKPRTEC= YES	READ= BACK	ERREXT	
FIXUNB/FIXBLK	688	+80	+112	+96	+96	
VARUNB/VARBLK	824	+112	+108	+84	+96	
UNDEF	560	+80	+120	+16	+96	
SPNUNB/SPNBLK	1808	-	+260	+300	+252	

Notes:

1. Only one module is required for processing all files having a common RECFORM. This module can be generated with the options charted above. To determine the size of the module with the options, the number of option bytes specified in the chart must be added to the basic module.

2. RDONLY=YES, changes the size of the module -50 to +50 bytes. In addition, the user's program must provide a 72-byte save area each time the module is reentered.

Figure 8-2. MTMOD--Main Storage Table Requirements (Part 1 of 3)

WORKFILE MODULE			
		NOTEPNT=	
TYPEFLE=WORK	without NOTEPNT	YES	POINTS
without ERROPT	232	424	286
with ERROPT	280	540	380
with ERROPT & ERREXT	436	654	494

Figure 8-2. MTMOD--Main Storage Table Requirements (Part 2 of 3)

ASCII MAGNETIC TAPE MODULE					
		INDEPENDENT OPTICNS			
RECFORM=	BASIC MODULE	WORKA= YES	CKPRTEC= YES	READ= BACK	ERREXT
FIXUNB/FIXBLK	824	+92		+128	+120
VARUNB/VARBLK	968	+112		+56	+96
UNDEF	656	+80		+8	+96

Figure 8-2. MTMOD--Main Storage Requirements (Part 3 of 3)

Module 9: Optical Reader--370N-IO-478

Modular Outline

Libraries					
Core Image				Source Statement	
	Blocks			Macros	Blocks
Transients	2311	2314/2319	3330		
2	2	2	2	7	609

CORE IMAGE LIBRARY

Transients

$$ANERR9 Error Recovery Procedure
$$BCOR01 OPEN

SOURCE STATEMENT LIBRARY

ORMOD (Optical Reader Module)

Macros

FILE DEFINITION

A.DTFOR
A.ORMOD

IMPERATIVE

A.DSPLY
A.RDLNE
A.RESCN

SAMPLE PROBLEMS:

Z.ORDC
Z.ORJT

STORAGE REQUIREMENTS

Figure 9-1 defines the storage requirements for ORMOD.

			INDEPENDENT OPTIONS			
DEVICE=	RECFORM=	Basic Module	CONTROL= YES	IOAREA2= YES	WORKA= YES	IOAREA2= YES and WORKA=YES
1287T/	FIXUNB	892	+220	+76	+60	+96
	FIXBLK	1256	+264	+56	+28	+84
	UNDEF	848	+224	+56	+64	+88
	UNDEF					
	BLKFAC=YES	1180	+268	+56	+24	+80
1287D or 1288	FIXUNB	1360	+184	---	---	---
	UNDEF	1256	+188	---	---	---

Figure 9-1. OPTICAL READER--ORMOD Main Storage Requirements

DTFOR (Define The File: Optical Reader) Table Requirements

1. RECFORM=FIXUNB requires 136 bytes.

2. RECFORM=FIXBLK varies as a function of blocking.
   Size = 136 + (16 x blocking factor x number of I/O areas)

3. RECFORM=UNDEF requires 136 bytes.

Sample Problem Summary

The 1288 sample problem is obtained by retrieving the sample problem for the 1287 journal tape mode of operation (Z.CRJT) and changing the cards with the sequence numbers 1058 and 1071 as follows: For both cards change the parameter DEVICE=1287T to DEVICE=1287D. (Refer to the listing containing the source statements for the sample test program for journal tape processing.) Figure 9-2 illustrates the files used by both problems, and Figure 9-3 illustrates the card decks that are punched when the sample problem is retrieved.

The sample problem illustrating document processing for the 1287 or 1288 optical reader (Z.ORDC) reads documents for data input into an input area, and then prints this data on SYSLST. Whenever the 1287 or 1288 document hopper empties, press end-of-file or replenish the stack.

Any approved document type may be used because the reference mark and data field coordinates are entered at program execution time. However, the data field chosen to be read may not exceed six characters in length.

Figure 9-4 is a sample input document. The listing contains examples of keyed-in error corrections identified by comments.

Note: When using this program on a 1288, remove the RESCN and DSPLY macro statements within sequence numbers 0091 and 0109.

The sample problem illustrating journal tape mode processing for the 1287 optical reader reads undefined records from the 1287 optical reader into a work area from two I/O areas, and then prints these records on SYSLST.

Any journal tape with a maximum record size of 38 characters is suitable for this sample problem.

Figure 9-5 is a portion of a sample input journal tape.

	Disk Extent Number			
Phase	1	2	3	4
Assemble	SYSLNK	SYS001	SYS002	SYS003
Link Edit	SYSLNK	SYS001	-	-
Execute	-	-	-	-

Figure 9-2. OPTICAL READER--Sample Problem

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.CRDC BKEND Z.CRDC Document Mode Source Deck (179 Cards) BKEND	$478	C001-0179
CATALS Z.CRJT BKEND Z.ORJT Journal Tape Source Deck (81 Cards) BKEND	$478	1001-1081

Figure 9-3. OPTICAL READER--Sample Problem Card Deck

SYSLOG Output:

```
// JOB OPTICAL READER SAMPLE PROBLEM
EOJ OPTICAL
// PAUSE END OF OPTICAL READER SAMPLE
PROBLEM
```

SYSLST Output Summary:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor input diagnostics
- Linkage editor storage map
- List of fields read from:
  for document sample problem, or
  for journal tape sample problem

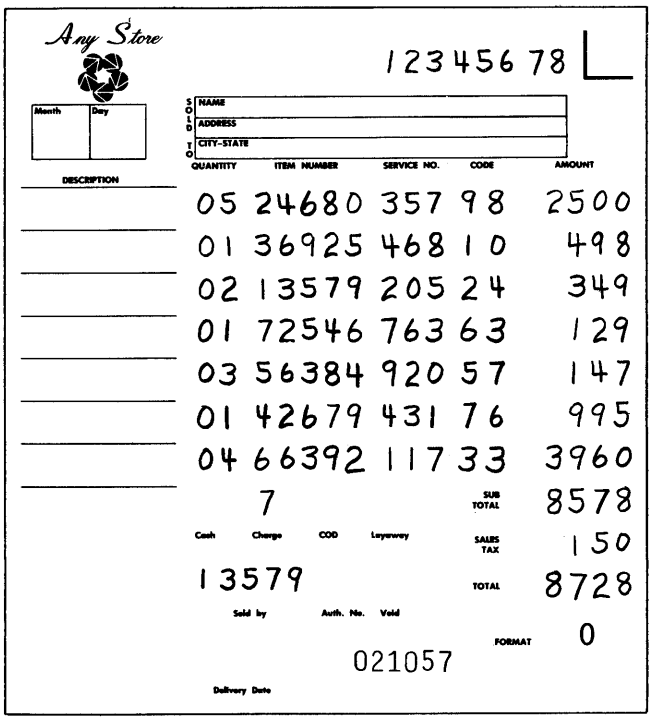


Figure 9-4. OPTICAL READER--Sample Input Document for Document Mode Processing

```
012 3456 789C
123 4567 890S
234 5678 901T
345 6789 012N
456 7890 123S
567 8901 234X
678 9012 345C
789 0123 456T
890 1234 567Z
901 2345 678/
012 3456 789C
123 4567 890S
234 5678 901T
345 6789 012N
456 7890 123S
567 8901 234X
678 9012 345C
789 0123 456T
890 1234 567Z
901 2345 678/
012 3456 789C
123 4567 890S
234 5678 901T
345 6789 012N
456 7890 123S
567 8901 234X
678 9012 345C
789 0123 456T
890 1234 567Z
901 2345 678/
```

Figure 9-5. OPTICAL READER--Sample Input for Journal Tape Mode Processing

Module 10: Paper Tape--370N-IO-458

Modular Outline

Libraries						
Core Image					Source Library	
	Blocks					
Transients	2311	2319 2314/2319	3330		Macros	Blocks
4	4	4	4		2	459

CORE IMAGE LIBRARY

Transients

$$ANERAI	Error Recovery Procedure (ERP)
$$ANERAJ	ERP
$$ANERRX	ERP
$$BERPTP	ERP - 1018 with error correction feature

SOURCE STATEMENT LIBRARY

File Definition Macros

A.DTFPT
A.PTMOD

STORAGE REQUIREMENTS

DTFPT (Define The File: Paper Tape) Table Requirements

INPUT FILE

The possible table specifications and sizes are:

1. No translations, no shifts, and no deletes require 72 bytes.
2. TRANS=name with no shifts and no deletes requires 76 bytes.
3. TRANS=name, SCAN=name, RECFORM=FIXUNB require 110 bytes.
4. TRANS=name, SCAN=name, RECFORM=UNDEF require 94 bytes.

OUTPUT FILE

The possible table specifications and sizes are:

1. No shifts require 69 bytes.
2. Shifts require 83 bytes.

PTMOD (Paper Tape Module)

The module specifications and sizes are specified in Figure 10-1.

	INPUT		OUTPUT
	2671	1017	1018
1. No parameters specified (no translation, no shifts and no deletes)	244	288	
2. TRANS=YES with no shifts and no deletes	310	354	
3. TRANS=YES, SCAN=YES, RECFORM=FIXUNB	536	570	
4. TRANS=YES, SCAN=YES, RECFORM=UNDEF	436	474	
5. No shifts			352
6. Shifts			570

Note: If module 2 is used, all records require translation.

Figure 10-1. PTMOD--Main Storage Requirements

Module 11: Sequential Disk--370N-IO-455

Modular Outline

Source Statement Library	
Macro	Blocks
12	2735

SOURCE STATEMENT LIBRARY

File Definition Macros

A.DTFSD
A.SDMOD
A.SDMODFI
A.SDMODFO
A.SDMODFU
A.SDMODVI

A.SDMODVO
A.SDMODVU
A.SDMODUI
A.SDMODUO
A.SDMODUU
A.SDMODW

STORAGE REQUIREMENTS

Figures 11-1 and 11-2 specify the storage required by DTFSD and SDMOD.

DTFSD (DEFINE THE FILE: SEQUENTIAL DASD)

	TYPEFLE					
	INPUT		OUTPUT	WORK		
RECFORM=	with UPDATE	without UPDATE		with UPDATE	without UPDATE	CONTROL = YES
FIXBLK or FIXUNB	176	152	160	152	152	+24
VARBLK or VARUNB	192	152	170	---	---	+24
SPNBLK or SPNUNB	240	188	244	---	---	+24
UNDEF	192	152	162	152	152	+24

Figure 11-1. DTFSD--Main Storage Requirements

SDMOD (SEQUENTIAL DASD MODULE)

Module Name	Basic Module	TRUNCS	CONTROL	ERROPT	HOLD	ERRCPT ERREXT	RECFORM=SPNELK RECFORM=SPNUNB
SDMODFI	462	+80	+28	+144	*	+228	--
SDMODFO	546	+136	+28	+64	*	+200	--
SDMODFU	798	+88	+28	+164	+96	+252	--
SDMODVI	729		+28	+104	*	+188	300-400
SDMODVO	1045		+28	+68	*	+120	1050-1150
SDMODVU	1086		+28	+176	+76	+296	1500
SDMODUI	533		+28	+100	*	+171	--
SDMODUO	653		+28	+68	*	+116	--
SDMODUU	941		+28	+148	+40	+248	--
SDMODW	572		+22	+148	+10	+246	--

*The HOLD function does not apply to these modules.

Notes:

1. For SDMODW, NOTEPNT=YES requires 206 additional bytes: NOTEPNT=PCINTRW requires 144 additional bytes, UPDATE=YES requires 40 additional bytes.

2. RDONLY=YES changes the size of the module -50 to +50 bytes. In addition, the user's program must provide a 72-byte save area each time the module is reentered.

Figure 11-2. SDMOD--Storage Requirements

Module 12: Assembler D--370N-AS-465

Modular Outline

IBM-SUPPLIED ASSEMBLERS

If System/370 instructions are to be assembled, Assembler D must be used. The relocatable library of the IBM-supplied system residence contains assembler modules suitable for building an Assembler using both tape and disk workfiles.

Note: See also Module 13, Assembler F --360N-AS-466, of this manual.

Assembler D requires a minimum of 14K bytes of contiguous problem storage for use by the assembler.

The names of the 14K modules begin with IJQ, thus modules can be copied, punched, or deleted by a COPYR, PUNCH, or DELETR statement specifying IJQ.ALL as an operand.

Assembler D can reside with Assembler F if the name of the first phase of one of them is changed from ASSEMBLY to some other name by means of the RENAMC function of the MAINT program. The renaming must be performed before the second assembler is linkage edited into the core image library. The first one is then invoked under its new name; the second under the name ASSEMBLY.

It is expedient, during system generation, to use the largest assembler the machine can support because the performance improvement is most significant.

WARNING DIAGNOSTICS

The following warning diagnostics appear in the linkage editor maps during system generation, but they do not indicate errors.

```
 *UNREFERENCED SYMBOLS

 EXTRN          IJQD0$45
 EXTRN          IJQD0$57
 EXTRN          IJQD0$58
 EXTRN          IJQD0A10
 EXTRN          IJQRTA30
 EXTRN          IJQRTB39
 EXTRN          IJQRTB54
 EXTRN          IJQD2$30

       POSSIBLE INVALID ENTRY POINT
DUPLICATION IN INPUT
```

CORE IMAGE LIBRARY

	Blocks		
Phases	2311	2314/2319	3330
32	85	88	102

Phases

ASSEMBLY
ASSEM02
ASSEM02A

ASSEM03
ASSEM03A
ASSEM04
ASSEM04A
ASSEM04B

ASSEM05
ASSEM05A
ASSEM05B
ASSEM06
ASSEN07

ASSEN07A
ASSEN07C
ASSEN07I
ASSEN08
ASSEN08A

ASSEN08C
ASSEN088
ASSEN09
ASSEN09I
ASSEN10

ASSEN10B
ASSEN11A
ASSEN11B
ASSEN11C
ASSEN11D

ASSEN11E
ASSEN12
ASSEN13
ASSEN14

RELOCATABLE LIBRARY

Modules	Blocks
26	567

Modules

IJQABT
IJQDIA$
IJQDPP
IJQD0$
IJQD2$$

IJQD2A
IJQD3$$
IJQD3A$
IJQD32
IJQD4A$
IJQD4M$

IJQD4P
IJQD5A$
IJQD5M$
IJQD5P$
IJQD7$$

IJQD7I
IJQD8$$
IJQD9$$
IJQD9I$
IJQRTA

IJQRTB
IJQ10$$
IJQ10B$
IJQ21A$
IJQ21B$

Linkage Edit Statements

```
   INCLUDE IJQD32
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJQ.ALL
```

SOURCE STATEMENT LIBRARY

Sample Problems

Z.AS1
Z.AS2
Z.AS3
Z.AS4
Z.AS5
Z.AS6

See Assembler Sample Problems.

WORKFILES

The Assembler D workfile requirements can be determined by adding the appropriate track values from Figure 12-2 to the appropriate track values determined from Figure 12-1. Note that Figure 12-1 is expressed in terms of number of bytes. The approximate number of tracks can be calculated by dividing the number of bytes by 3000 for a 2311 file, by 6000 for a 2314/2319 file, or by 12000 for a 3330 file. These numbers represent the approximate number of text bytes per track for a 2311, a 2314/2319, and a 3330 respectively. In Figure 12-2 SYSLNK requirements are expressed in terms of tracks per macro instruction; the other three areas are expressed in tracks per macro definition.

In determining the total number of tracks required for Assembler workfiles, the SYSLNK value must be included each time the macro instruction is used. The SYS001, SYS002, and SYS003 values need be included only once per macro definition, regardless of the number of times the macro instruction is used.

For example, if three tape files are defined, three DTFMT macros are used. The track values for SYS001, SYS002, SYS003 are included once because the Assembler uses macro definition only once; but the SYSLNK value is multiplied by three because the macro instruction is expanded three times.

		Number of Bytes per Statement			
		SYSLNK	SYS001	SYS002	SYS003
IJQD32	1 for 1 Statements	15	130	130	36

Figure 12-1. ASSEMBLER D --Nonmacro Assembly Workfile Requirements for 2311, 2314/2319 and 3330

	Number of Tracks											
	SYSLNK Tracks per Macro Inst.			SYS001 Tracks per Macro Def.			SYS002 Tracks per Macro Def.			SYS003 Tracks per Macro Def.		
	2311	2314/ 2319	3330	2311	2314/ 2319	3330	2311	2314/ 2319	3330	2311	2314/ 2319	3330
Short Macros like IOCS Imperatives	0.02	0.01	0.01	1	1	1	2	1	1	1	1	1
CDMOD	0.15	0.09	0.05	11	6	3	3	7	4	17	8	5
DAMOD	0.38	0.24	0.17	10	5	3	13	7	4	9	4	3
DIMOD	0.55	0.35	0.21	8	4	2	11	6	3	11	5	3
DTFCD	0.05	0.03	0.02	6	3	2	7	4	2	9	4	3
DTFCN	0.08	0.05	0.03	4	2	1	5	3	2	6	3	2
DTFDA	0.08	0.05	0.03	9	5	3	10	6	3	13	6	4
DTFDI	0.36	0.23	0.12	4	2	1	4	2	1	5	2	2
*DTFEN	0	0	0	127	67	34	143	75	39	162	84	42
DTFIS	0.12	0.08	0.05	9	5	3	11	6	3	13	6	4
DTFMR	0.34	0.17	0.12	4	2	1	5	3	2	5	3	2
DTFMT	0.05	0.03	0.02	9	5	3	10	5	3	12	6	3
DTFOR	0.08	0.05	0.03	4	2	1	4	2	1	5	2	2
DTFPH	0.08	0.05	0.03	3	2	1	4	2	1	5	2	2
DTFPR	0.05	0.03	0.02	3	2	1	4	2	1	5	2	2
DTFPT	0.06	0.04	0.02	8	5	3	9	5	3	10	5	3
DTFSD	0.08	0.05	0.03	10	5	3	11	6	3	13	7	4
DTFSR	0.08	0.05	0.03	79	41	21	90	47	24	105	54	27
ISMOD	0.74	0.47	0.32	53	28	15	64	33	17	58	30	15

Figure 12-2. ASSEMBLER D --Macro Instruction Workfile Requirements for the IBM 2311, IBM 2314/2319, and IBM 3330 (Part 1 of 2)

	Number of Tracks											
	SYSLNK Tracks per Macro Inst.			SYS001 Tracks per Macro Def.			SYS002 Tracks per Macro Def.			SYS003 Tracks per Macro Def.		
	2311	2314/2319	3330	2311	2314/2319	3330	2311	2314/2319	3330	2311	2314/2319	3330
MRMOD	0.67	0.34	0.18	10	5	3	12	6	3	8	4	2
MTMOD	0.23	0.15	0.08	22	12	6	23	12	6	27	14	7
ORMOD	0.20	0.13	0.07	11	6	3	16	8	4	15	7	4
PRMOD	0.12	0.08	0.04	5	3	2	6	4	2	8	4	2
PTMOD	0.21	0.12	0.06	18	10	5	18	10	5	14	10	5
SDMODFI	0.80	0.50	0.28	12	6	3	14	8	4	16	8	4
SDMODFO	0.80	0.50	0.28	12	7	3	15	8	4	16	8	4
SDMODFU	0.80	0.50	0.28	15	8	4	18	10	5	19	9	5
SDMODVI	0.80	0.50	0.28	11	6	3	12	6	3	14	7	4
SDMODVO	0.80	0.50	0.28	13	7	4	16	9	5	16	8	4
SDMODVU	0.80	0.50	0.28	13	7	4	17	9	5	16	8	4
SDMODUI	0.80	0.50	0.28	10	5	3	11	6	3	13	7	4
SDMODUO	0.80	0.50	0.28	11	6	3	12	6	3	14	7	4
SDMODUU	0.80	0.50	0.28	13	7	4	15	8	4	16	8	4
SDMODW	0.80	0.50	0.28	16	9	5	17	9	5	22	11	6

* The measurements for DTFEN, unlike the other macro instruction measurements, represent minimum workfile requirements. The measurements were made assembling DTFEN by itself, which resulted in minimum expansion. The use of DTFEN should be avoided wherever possible because of its large workfile requirements.

Figure 12-2. ASSEMBLER D --Macro Instruction Workfile Requirements for the IBM 2311, IBM 2314/2319, and IBM 3330 (Part 2 of 2)

Module 13: Assembler F--360N-AS-466

Modular Outline

IBM-SUPPLIED ASSEMBLERS

If System/370 instructions are to be assembled, Assembler D must be used. Assembler F requires a minimum of 44K bytes of contiguous storage.

The names of the 44K modules begin with IJY; therefore, modules can be copied, punched, or deleted by a CCPYR, PUNCH, or DELETR statement specifying IJY.ALL as an operand.

Assembler D can reside with Assembler F if the name of the first phase of one of them is changed from ASSEMBLY to some other name by means of the RENAMC function of the MAINT program. The renaming must be performed before the second assembler is linkage edited into the core image library. The first one is then invoked under its new name; the second under the name ASSEMBLY.

It is expedient, during system generation, to use the largest assembler the machine can support because the performance improvement is most significant.

CORE IMAGE LIBRARY

Phases

	Blocks	
Phases	2311	2314/2319
8	70	71

ASSEMAPT
ASSEMBLY
ASSEMFI
ASSEMFPP
ASSEMF8
ASSEM3
ASSEM3E
ASSEM7

RELOCATABLE LIBRARY

Modules	Blocks
34	477

Modules

IJYAPT
IJYASM
IJYCM
IJYFD
IJYFI0

IJYFPP
IJYF0
IJYF1
IJYF2
IJYF3

IJYF3E
IJYF7C
IJYF7D
IJYF7E
IJYF7G
IJYF7I
IJYF7L
IJYF7N
IJYF7S
IJYF7V

IJYF7X
IJYF8A
IJYF8C
IJYF8D
IJYF8I

IJYF8L
IJYF8M
IJYF8N
IJYF8P
IJYF8S

IJYF8V
IJYIN
IJYRTA
IJYRTB

Linkage Edit Statements

```
   INCLUDE IJYASM
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJY.ALL
```

SOURCE STATEMENT LIBRARY

Sample Problems

Z.AS1
Z.AS2
Z.AS3

Z.AS4
Z.AS5
Z.AS6

See Module 14, Assembler Sample Problems.

WORKFILES

The Assembler F workfile requirements can be determined by adding the appropriate track values from Figure 13-2 to the appropriate track values determined from Figure 13-1. Note that Figure 13-1 is expressed in terms of number of bytes. The approximate number of tracks can be calculated by dividing the number of bytes by 3000 for a 2311 file or by 6000 for a 2314/2319 file. These numbers represent the approximate number of text bytes per track for a 2311 and a 2314/2319, respectively. In Figure 13-2 SYSLNK requirements are expressed in terms of tracks per macro instruction; the other three areas are expressed in tracks per macro definition.

In determining the total number of tracks required for assembler workfiles, the SYSLNK value must be included each time the macro instruction is used. The SYS001, SYS002, and SYS003 values need be included only once per macro definition, regardless of the number of times the macro instruction is used.

For example, if three tape files are defined, three DTFMT macros are used. The track values for SYS001, SYS002, SYS003 are included once, because the assembler uses macro definition only once; but the SYSLNK value is multiplied by three because the macro instruction is expanded three times.

		Number of Bytes per Statement			
		SYSLNK	SYS001	SYS002	SYS003
IJYASM	1 for 1 Statements	15	130	130	36

Figure 13-1. ASSEMBLER F --Nonmacro Assembly Workfile Requirements for 2311 and 2314/2319

	Number of Tracks						
	SYSLNK Tracks per Macro Inst.	SYS001 Tracks per Macro Def.		SYS002 Tracks per Macro Def.		SYS003 Tracks per Macro Def.	
	2311 and 2314/2319	2311	2314/ 2319	2311	2314/ 2319	2311	2314/ 2319
Short Macros like IOCS Imperatives	0.02	1	1	1	1	2	1
CDMOD	0.15	2	2	2	1	10	5
DAMOD	0.38	5	3	5	3	6	4
DIMOD	0.55	5	3	5	3	7	4
DTFCD	0.05	2	1	2	1	5	3
DTFCN	0.08	2	1	2	1	4	3
DTFDA	0.08	3	2	3	2	7	4
DTFDI	0.36	2	1	2	1	4	3
*DTFEN	0	2	1	2	1	121	62
DTFIS	0.12	3	2	3	2	7	4
DTFMR	0.34	2	1	2	1	4	3
DTFMT	0.05	2	1	3	2	7	4
DTFOR	0.08	2	1	1	1	3	2
DTFPH	0.08	1	1	1	1	3	2
DTFPR	0.05	2	1	1	1	3	2
DTFPT	0.06	2	1	2	1	7	3
DTFSD	0.08	2	1	3	2	8	4
DTFSR	0.08	2	1	3	2	70	38
ISMOD	0.74	18	17	15	15	26	25
MRMOD	0.67	5	3	4	2	7	4

Figure 13-2. ASSEMBLER F --Macro Instruction Workfile Requirements for the IBM 2311 and IBM 2314/2319 (Part 1 of 2)

	Number of Tracks							
	SYSLNK Tracks per Macro Inst.	SYS001 Tracks per Macro Def.		SYS002 Tracks per Macro Def.		SYS003 Tracks per Macro Def.	
	2311 and 2314/2319	2311	2314/ 2319	2311	2314/ 2319	2311	2314/ 2319
MTMOD	0.23	6	3	5	3	20	11
ORMOD	0.20	3	4	3	4	7	6
PRMOD	0.12	3	2	2	1	5	3
PTMOD	0.16	4	2	3	2	13	7
READ, WRITE	0.02	1	1	1	1	2	1
SDMODFI	0.80	4	3	4	2	12	6
SDMODFO	0.80	5	3	4	2	12	7
SDMODFU	0.80	6	3	5	3	15	8
SDMODVI	0.80	5	3	5	3	10	5
SDMODVO	0.80	7	4	6	3	12	6
SDMODVU	0.80	7	4	6	4	12	7
SDMODUI	0.80	5	3	4	2	9	5
SDMODUO	0.80	5	3	4	3	10	5
SDMODUU	0.80	7	4	6	3	12	6
SDMODW	0.80	6	4	6	3	13	7

* The measurements for DTFEN, unlike the other macro instruction measurements, represent minimum workfile requirements. The measurements were made assembling DTFEN by itself, which resulted in minimum expansion. The use of DTFEN should be avoided wherever possible because of its large workfile requirements.

Figure 13-2. ASSEMBLER F --Macro Instruction Workfile Requirements for the IBM 2311 and IBM 2314/2319 (Part 2 of 2)

Module 14: Assembler Sample Problems

Modular Outline

The files required by all the assembler sample problems are listed in Figure 14-1. Figure 14-2 illustrates the sample problem card decks retrieved from the source statement library.

Phase	Disk Extent Number 1	2	3	4
Assemble	SYSLNK	SYS001	SYS002	SYS003
Link Edit	SYSLNK	SYS001	--	--
Execute	-	-	-	-

Figure 14-1. ASSEMBLERS--Sample Problem File Requirements

Cards Retrieved	Card Columns 73-76	Card Columns 77-80	Assembler Sample Problem	Generates Input for
CATALS Z.AS1 BKEND Z.AS1 Assembler Source Deck (23 Cards) BKEND	$465	0001-0023	1	Tape Sort/Merge Z.SM1
CATALS Z.AS2 BKEND Z.AS2 Assembler Source Deck (23 Cards) BKEND	$465	0101-0123	2	Disk Sort/Merge Z.SM2
CATALS Z.AS3 BKEND Z.AS3 Assembler Source Deck (23 Cards) BKEND	$465	0201-0223	3	Data Cell to Printer Utility Z.UTDCPR1
CATALS Z.AS4 BKEND Z.AS4 Assembler Source Deck (23 Cards) BKEND	$465	0402-0423	4	Tape and Disk Sort/Merge Z.SM4
CATALS Z.AS5 BKEND Z.AS5 Assembler Source Deck (23 Cards)	$465	0502-0523	5	Tape and Disk Sort/Merge Z.SM5
CATALS Z.AS6 BKEND Z.AS6 Assembler Source Deck (23 Cards) BKEND	$465	0602-0623	6	Tape and Disk Sort/Merge Z.SM6

Figure 14-2. ASSEMBLERS--Sample Problem Card Decks

Problem 1: Assembler 1

Program Name is Z.AS1. The Assembler 1 sample problem generates numbers from 2000 to 0001, in descending order and writes them on tape as 15-character unblocked records (11 blanks followed by 4 zone digits, unsigned).

To execute the Assembler 1 sample problem, the following job control cards are needed:

```
// JOB ASSEMBLE SAMPLE 1.
// OPTION LINK,LIST,LOG,NOXREF
// EXEC ASSEMBLY
   Assembler Source Deck (23 cards)
/*
// EXEC LNKEDT
// ASSGN SYS002,X'cuu'
// PAUSE ASSGN SYS002 TO A 9 TRACK TAPE DRIVE
// MTC REW,SYS002
// EXEC
/&
```

SYSLOG Output

```
// JOB ASSEMBLE
// PAUSE  ASSGN SYS002 TO A 9 TRACK TAPE DRIVE
EOJ ASSEMBLE
```

SYSLST Output Summary:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor storage map

Problem 2: Assembler 2

Program Name is Z.AS2. The Assembler 2 sample problem generates numbers from 2000 to 0001, in descending order, and writes them on disk.

To execute the Assembler 2 sample problem, the following job control cards are needed:

```
              Col. 16                                  Col. 54      Col. 72
              ↓                                        ↓            ↓
// JOB ASSEMBLE SAMPLE 2
// OPTION LINK,LIST,LOG,NOXREF
// EXEC ASSEMBLY
   Assembler Source Deck (23 cards)
/*
// EXEC LNKEDT
// ASSGN SYS002,X'cuu'
// PAUSE ASSGN SYS002 to SCRATCH PACK
// DLBL OUTFILE,'SAMPLE PROBLEM FILE OF 2000 RECORDS'[,yy/ddd]
// EXTENT SYS002,hhhhhh,1,0,1800,43
// EXEC
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number of the disk pack.

SYSLOG Output

```
// JOB ASSEMBLE
// PAUSE  ASSIGN SYS002 TO SCRATCH PACK
EOJ ASSEMBLE
```

SYSLST Output Summary:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor storage map

Problem 3: Assembler 3

Program Name is Z.AS3. The Assembler 3 sample problem generates numbers from 100 to 001, in descending order, and writes them on a data cell.

To execute the Assembler 3 sample problem, the following job control cards are needed:

```
                Col. 16                                  Col. 54      Col. 72
                  ↓                                        ↓            ↓
// JOB ASSEMBLE SAMPLE 3
// OPTION LINK,LIST,LOG,NOXREF
// EXEC ASSEMBLY
   Assembler Source Deck (23 cards)
/*
// EXEC LNKEDT
// DLBL OUTFILE,'SAMPLE PROBLEM FILE OF 100 RECORDS'[,yy/ddd]
// EXTENT SYS004,hhhhhh,1,0,10200,20,,B=3
// ASSGN SYS004,X'cuu'
*  PAUSE ASSGN SYS004 TO DATA CELL
// PAUSE MOUNT CELL hhhhhh ON STATION 3
// EXEC
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number of the data cell.

SYSLOG Output

```
// JOB ASSEMBLE
*  PAUSE  ASSGN SYS004 TO DATA CELL
// PAUSE  MOUNT CELL nnnnnn ON STATION 3
EOJ ASSEMBLE
```

SYSLST Output Summary:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor storage map

Problem 4: Assembler 4

Program Name is Z.AS4. The Assembler 4 sample problem generates numbers from 2000 to 0001, in descending order, and writes them on tape as 15 character unblocked records (11 blanks followed by 4 zoned digits, unsigned).

To execute the Assembler 4 sample problem, the following job control cards are needed:

```
// JOB ASSEMBLY SAMPLE 4
// OPTION LINK,LIST,LOG,NOXREF
// EXEC ASSEMBLY
   Assembler Source Deck (23 cards)
/*
// EXEC LNKEDT
// ASSGN SYS002,X'cuu'
// PAUSE ASSGN SYS002 TO A 9 TRACK TAPE DRIVE
// MTC REW,SYS002
// EXEC
/&
```

SYSLOG Output

```
BG // JOB ASSEMBLY SAMPLE 4
BG // PAUSE ASSGN SYS002 TO A 9 TRACK TAPE DRIVE
BG
BG EOJ ASSEMBLY
```

SYSLST Output Summary:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor storage map

Problem 5: Assembler 5

Program Name is Z.AS5. The Assembler 5 sample problem generates numbers from 2000 to 0001, in descending order, and writes them to a 2311 disk storage device.

To execute the Assembler 5 sample problem, the following job control cards are needed:

```
              Col. 16                                  Col. 54      Col. 72
                 ↓                                        ↓           ↓
// JOB ASSEMBLY SAMPLE 5
// OPTION LINK,LIST,LOG,NOXREF
// EXEC ASSEMBLY
   Assembler Source Deck (23 cards)
/*
// EXEC LNKEDT
// ASSGN SYS002,X'cuu'
// PAUSE ASSGN SYS002 TO SCRATCH PACK
// DLBL OUTFILE,'SAMPLE PROBLEM FILE OF 2000 RECORDS'[,yy/ddd]
// EXTENT SYS002,hhhhhh,1,0,1800,43
// EXEC
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number of the disk pack.

SYSLOG Output

```
BG // JOB ASSEMBLY SAMPLE 5
BG // PAUSE ASSGN SYS002 TO SCRATCH PACK
BG
BG EOJ ASSEMBLY
```

SYSLST Output Summary

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor storage map

Problem 6: Assembler 6

Program Name is Z.AS6. The Assembler 6 sample problem generates numbers from 3000 to 0001, in descending order, and writes them to a 2314 direct access storage facility or 2319 disk storage facility.

To execute the Assembler 6 sample problem, the following job control cards are needed:

```
              Col. 16                                  Col. 54      Col. 72
                ↓                                        ↓            ↓
// JOB ASSEMBLY SAMPLE 6
// OPTION LINK,LIST,LOG,NOXREF
// EXEC ASSEMBLY
   Assembler Source Deck (23 cards)
/*
// EXEC LNKEDT
// ASSGN SYS002,OUTFILE
// PAUSE ASSGN SYS002 TO SCRATCH PACK
// DLBL OUTFILE.'SAMPLE PROBLEM FILE OF 2000 RECORDS'[,yy/ddd]
// EXTENT SYS002,hhhhhh,1,0,3600,83
// EXEC
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number of the disk pack.

SYSLOG Output

```
BG // JOB ASSEMBLY SAMPLE 6
BG // PAUSE ASSGN SYS002 TO SCRATCH PACK
BG
BG EOJ ASSEMBLY
```

SYSLST Output Summary:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Linkage editor storage map

Module 15: American National Standard COBOL -- 360N-CB-482

Modular Outline

PARTITION SIZE

54,272 bytes

CORE IMAGE LIBRARY

Number of Phases	Number of Library Blocks (Physical Records)	
	2311	2314/2319
13	220	223
3	2	2

*Transients

Phases

FCOBOL
FCOBOL11
FCOBOL12
FCOBOL20
FCOBOL21

FCOBOL22
FCOBOL30
FCOBOL40
FCOBOL50
FCOBOL51
FCOBOL60
FCOBOL61
FCOBOL70

Transients

$$BCOBER
$$BCOBR1
$$BFCMUL

RELOCATABLE LIBRARY

Modules	Number of Library Blocks (Physical Records)
16	1584
67*	256

*Subroutines

Modules

ILACBD
ILACBL00
ILACBL01
ILACBL10
ILACBL11

ILACBL12
ILACBL20
ILACBL21
ILACBL22
ILACBL30

ILACBL40
ILACBL50
ILACBL51
ILACBL60
ILACBL61
ILACBL70

Linkage Edit Statements

```
   INCLUDE ILACBD
// EXEC LNKEDT
```

Delete Statements

```
DELETR ILA.ALL
```

SUBROUTINES

Module Name	Bytes of Main Storage
IILBDABX0	242
ILBDACP0	764
IILBDANE0	322
IILBDANF0	110
IILBDASY0	88
ILBDATB0	256
ILBDBID0	112
ILBDBIE0	116
ILBDBII0	464
IILBDCKP0	736
ILBDCLK0	56
IILBDCLS0	150
ILBDCRD0	150
IILBDDAE0	296
IILBDDCI0	177
ILBDDIO0	592
ILBDDSP0	2008
IILBDDSR0	342
ILBDDUM0	2
ILBDEFL0	524
ILBDETB0	256
ILBDFMT0	182

ILBDFPW0	808
ILBDGPW0	88
ILBDIDA0	218
ILBDIDB0	116
ILBDIDR0	1663
ILBDIDT0	692
ILBDIFB0	296
ILBDIFD0	156
ILBDIML0	88
ILBDISE0	404
ILBDISM0	366
ILBDITB0	256
ILBDIVL0	72
ILBDMFT0	148
ILBDMNS0	1
ILBDMOV0	66
ILBDMVE0	224
ILBDNSL0	616
ILBDOSY0	132
ILBDRCR0	148
ILBDRDI0	440
ILBDRDS0	246
ILBDRFM0	132
ILBDSAE0	236
ILBDSCH0	744
ILBDSEM0	279
ILBDSET0	40
ILBDSPA0	1244
ILBDSRT0	2380
ILBDSTI0	596
ILBDTEF0	600
ILBDTCD0	200
ILBDTRN0	256
ILBDUPS0	108
ILBDUSL0	372
ILBDUTB0	256
ILBDVBL0	346
ILBDVCO0	512
ILBDVMC0	468
ILBDVTR0	138
ILBDWTB0	256
ILBDXDI0	273
ILBDXMU0	184
ILBDXPR0	608
ILBDXTN0	264

SOURCE STATEMENT LIBRARY

Source Statement Library	
Macro	Block
1	24

Sample Problem

Z.CB2

SAMPLE PROGRAM SUMMARY

Program Name is Z.CB2. Figure 15-1 illustrates the files required by the American National Standard CCBCL sample problem. Figure 15-2 represents the American National Standard COBOL sample problem card deck that is retrieved from the source statement library. The American National Standard COBOL sample program generates and prints a table of weekly, monthly, and annual salaries based on ten-dollar increments in monthly salary from $500 to $1000. To execute the American National Standard CCBCL sample problem, the following job control cards are needed:

```
// JOB SAMPLE
// OPTION NODECK,LINK,LIST,LOG,LISTX,DUMP,SYM,ERRS
// EXEC FCOBOL
          CBL QUOTE
     (American National Standard COBOL Source Deck -- 80 cards)
/*
// EXEC LNKEDT
// ASSGN SYS004,X'cuu' (SYS004 must be assigned to printer designated as
    SYSLST during system generation)
// EXEC
/&
```

SYSLOG Output:

```
// JOB SAMPLE
EOJ SAMPLE
```

SYSLST Output Summary:

- Job control cards
- Source program listing
- Data division storage map
- Procedure division storage map
- Diagnostics
- Linkage editor storage map
- Table of salaries generated by the program

Note: Output is on SYS004 (same as printer).

	Disk Extent Number				
Phase	1	2	3	4	5
Compile	SYSLNK	SYS001	SYS002	SYS003	SYS004
Link Edit	SYSLNK	SYS001	-	-	-
Execute	-	-	-	-	-

Figure 15-1. American National Standard CCBCL--Sample Problem File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.CB2		
BKEND Z.CB2	$482	0001 - 0080
American National Standard COBCL source deck (80 cards)		
BKEND		

Figure 15-2. American National Standard COBOL--Sample Problem Card Deck

WORKFILES

The workfile requirements for American National Standard COBOL are as follows:

IBM 2311

The amount of work space for American National Standard COBOL depends mainly on the size of the object program. Figure 15-3 shows the approximate percentage of tracks that should be assigned to each programmer logical unit, for programs of approximately 1000 and 2100 source records. Both programs are assumed to request a cross-reference listing.

Number of Source Records	Total Tracks	SYS001 %	SYS002 %	SYS003 %	SYS004 %
1000	60	33	37	22	8
2100	84	28	33	24	15

Figure 15-3. American National Standard COBOL--IBM 2311 Approximate Allocation of Workfile Space

IBM 2314/2319

The amount of work space for American National Standard COBOL depends mainly on the size of the object program. Figure 15-4 shows the approximate percentage of tracks that should be assigned to each programmer logical unit, for programs of approximately 1000 and 2100 source records. Both programs are assumed to request a cross-reference listing.

Number of Source Records	Total Tracks	SYS001 %	SYS002 %	SYS003 %	SYS004 %
1000	26	35	31	23	11
2100	39	31	31	23	15

Figure 15-4. American National Standard COBOL--IBM 2314/2319 Approximate Allocation of Workfile Space

Module 16: COBOL--360N-CB-452

Modular Outline

PARTITION SIZE

14,336 bytes

CORE IMAGE LIBRARY

Component	Phases	Blocks 2311	Blocks 2314/2319
COBOL:	47	172	174
Transients	5	5	5
COBOL Debug	1	4	4

Phases

COBCL
COBCL000
COBOL001
COBCL002

COBOL003
COBCL004
COBOL005
COBCL006
COBCL007

COBOL008
COBCL009
COBOL010
COBCL011
COBOL012

COBOL013
COBCL014
COBOL015
COBCL016
COBOL017

COBCL018
COBOL019
COBCL020
COBOL021
COBCL022

COBCL023
COBOL024
COBCL025
COBOL027
COBCL028

COBOL029
COBCL030
COBOL031
COBCL032
COBOL033

COBOL034
COBCL035
COBOL036
COBCL037
COBOL038

COBCL039
COBCL040
COBOL041
COBCL042
COBOL043

COBCL044
COBOL050
COBCL055

Transients

$$BCBLIS
$$BCBICP
$$BCBODA
$$BCBUSR
$$BCBUSW

COBOL Debug Phase

DEBUG

RELOCATABLE LIBRARY

	Modules	Blocks
COBOL:	51	1327
Subroutines	42	162
COBOL Debug	2	23

Modules

IJSCBD
IJSCBL01
IJSCBL02
IJSCBL03
IJSCBL04

IJSCBL05
IJSCBL06
IJSCBL07
IJSCBL08
IJSCBL09

IJSCBL10
IJSCBL11
IJSCBL12
IJSCBL13
IJSCBL14

IJSCBL15
IJSCBL16
IJSCBL17
IJSCBL18
IJSCBL19

```
IJSCBL20
IJSCBL21
IJSCBL22
IJSCBL23
IJSCBL24


IJSCBL25
IJSCBL26
IJSCBL27
IJSCBL28
IJSCBL29


IJSCBL31
IJSCBL32
IJSCBL33
IJSCBL34
IJSCBL35


IJSCBL36
IJSCBL37
IJSCBL38
IJSCBL39
IJSCBL40

IJSCBL41
IJSCBL42
IJSCBL43
IJSCBL44
IJSCBL45

IJSCBL46
IJSCBL47
IJSCBL48
IJSCBL49
IJSCBL50
IJSCBL55
```

Linkage Edit Statements

```
   INCLUDE IJSCBD
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJS.ALL
```

Note: This statement also deletes the Debug program.

COBOL Debug Modules

```
IJSDDB
IJSCBL60
```

COBOL Debug Linkage Edit Statements

```
   INCLUDE IJSDDB
// EXEC LNKEDT
```

Note: The COBOL Debug program is deleted when the COBOL Compiler is deleted.

SUBROUTINES

For additional information about these subroutines, see the COBOL Programmer's Guide listed in the Preface of this publication.

Module Name	Bytes of Main Storage
IHD00000	426
IHD00100	428
IHD00200	142
IHD00300	170
IHD00400	804
IHD00500	436
IHD00600	348
IHD00700	280
IHD00800	68
IHD00900	56
IHD01000	53
IHD01100	216
IHD01200	36
IHD01300	56
IHD01400	52
IHD01500	330
IHD01600	80
IHD01700	217
IHD01800	57
IHD01900	812
IHD02000	358
IHD02100	178
IHD02200	20
IHD02300	188
IHD02400	182
IHD02500	196
IHD02600	64
IHD02700	420
IHD02800	372
IHD02900	316
IHD03000	580
IHD03100	880
IHD03200	611
IHD03300	228
IHD03400	638
IHD03500	796
IHD03600	338
IHD03700	691
IHD03800	1162
IHD03900	781
IHD04000	622
IHD04100	1948

SOURCE STATEMENT LIBRARY

Macro	Blocks
1	25

Sample Problem

Z.CB1

SAMPLE PROBLEM SUMMARY

Program Name is Z.CB1. The COBCL sample problem generates and prints a table of weekly, monthly, and annual salaries based on ten- dollar increments in monthly salary from $500 to $1000. Figure 16-1 defines the files required. Figure 16-2 illustrates the card deck for the COBOL sample problem that is retrieved from the source statement library.

	Disk Extent Number				
Phase	1	2	3	4	5
Compile	SYSLNK	SYS001	SYS002	SYS003	SYS004
Link Edit	SYSLNK	SYS001	-	-	-
Execute	-	-	-	-	-

Figure 16-1. COBOL--Sample Problem File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.CB1		
BKEND Z.CE1		
COBOL Source Deck (81 cards)	$452	0001-0081
BKEND		

Figure 16-2. COBOL--Sample Problem Card Deck

To execute the COBOL sample problem, the following job control cards are needed:

```
// JOB COBOL SAMPLE
// OPTION LINK,LIST,LOG,LISTX,DUMP,SYM,ERRS
// EXEC COBCL
   COBOL Source Deck (81 cards)
/*
// EXEC LNKEDT
// ASSGN SYS004,X'cuu' (SYS004 must be assigned to printer designated as
   SYSLST during system generation)
// EXEC
/&
```

SYSLOG Output

```
// JOB COBOL
EOJ COBOL
```

SYSLST Output Summary:

- Job control cards
- Source program listing
- Data division storage map
- Prccedure division storage map
- Diagnostics
- Linkage editor storage map
- Table of salaries generated by the program

Note: Output is on SYS004 (same as printer).

WORKFILES

COBOL Workfiles (IBM 2311)

Although the amount of COBOL work space depends mainly on the size of the object program, these general guidelines can be given: For a COBOL program of about 400 source statements, 10 tracks should be assigned to SYSLNK, SYS001, SYS002, and SYS003. For a COBOL program of about 800 source statements, 20 tracks should be assigned to SYSLNK, SYS001, SYS002, and SYS003.

COBOL Workfiles (IBM 2314/2319)

Although the amount of COBOL work space depends mainly on the size of the object program, these general guidelines can be given: For a COBOL program of about 400 source statements, 5 tracks should be assigned to SYSLNK, SYS001, SYS002, and SYS003. For a COBOL program of about 800 source statements, 10 tracks should be assigned to SYSLNK, SYS001, SYS002, and SYS003.

Module 17: COBOL and PL/I (D) DASD--360N-CB-468

Modular Outline

SOURCE STATEMENT LIBRARY

Macro	Blocks
8	270

Macros

A.LOADA
A.LODIS *

A.RANDA
A.RRUIS *
A.RUADA
A.RUAIS *
A.SEQDA
A.SRUIS *

*Used by PL/I (D).

Module 18: COBOL LCP--360N-CV-489

Modular Outline

PARTITION SIZE

18,176 bytes

CORE IMAGE LIBRARY

Phases	Blocks	
	2311	2314/2319
25	62	62

Phases

LCPC
LCPCOBDK
LCPCOBTP
LCPCOB11

LCPCOB12
LCPCOB13
LCPCOB21
LCPCOB22
LCPCOB23

LCPCOB24
LCPCOB31
LCPCOB32
LCPCOB33
LCPCOB34

LCPCOB35
LCPCOB36
LCPCOB37
LCPCOB4C
LCPCOB4D

LCPCOB4E
LCPCOB4F
LCPCOB4G
LCPCOB41
LCPCOB42

LCPCOB43

RELOCATABLE LIBRARY

Modules	Blocks
33	461

Modules

IKLB0C
IKLB0D

IKLB1B
IKLB11
IKLB12

IKLB13
IKLB21
IKLB22
IKLB23
IKLB24

IKLB31
IKLB32
IKLB33
IKLB34
IKLB35
IKLB36

IKLB37
IKLB4B
IKLB4C
IKLB4D

IKLB4E
IKLB4F
IKLB4G
IKLB41
IKLB42

IKLDAA
IKLDAB
IKLDAC
IKLD0A
IKLD0B
IKLD1A
IKLD4A
IKLLCP

Linkage Edit Statements

```
   INCLUDE IKLLCP
// EXEC LNKEDT
```

Delete Statements

```
DELETR IKL.ALL
```

SOURCE STATEMENT LIBRARY

Macro	Blocks
1	38

Sample Problem

Z.DLCP12

SAMPLE PROBLEM SUMMARY

Program Name is Z.DLCP12. The sample program is a COBOL D source program which is converted to American National Standard COBOL by the COBOL LCP program.

Figure 18-1 defines the files required by this sample problem. Figure 18-2 shows the sample problem card deck retrieved from the source statement library.

Disk Extent Number		
1	2	3
SYS001	SYS002	SYS003

Figure 18-1. COBOL LCP--Sample Problem File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.DLCP12		
BKEND Z.DLCP12	$489	0001-0130
COBOL LCP Source Deck (130 cards)		
BKEND		

Figure 18-2. COBOL LCP--Sample Problem Card Deck

To execute the sample program, the following two job control cards are needed:

```
// JOB DLCP12
// EXEC LCPC
 LCP COB=D,RESEQ,START=000000,INCR=010
 Source Deck (130 cards)
/*
/&
```

SYSLST output summary:

- Job control cards
- LCP option card (line number 00001). Options specified are for COBOL D, resequencing of the ID field of the source program, starting with 000000, and incrementing by 010.
- Converted source program. When a line is changed, the original line is followed by the changed line. At the extreme right of the changed line is a word indicating the status of the line (CHANGED, INSERTED, DELETED), followed by a message number corresponding to the action taken.
- Text of all messages issued by the program, in numerical order.
- Line indicating the highest severity code issued by the program.

Note: The sample program also produces a converted source deck on SYSPCH. If a deck is not desired, change the LCP option card to read as follows:

```
LCP COB=D,RESEQ,START=000000,INCR=010,NODECK
```

WORKFILES

COBOL LCP (IBM 2311)

LCP work space depends on the size of the scurce program to be converted, and, to a lesser extent, on the amount of main storage available for processing. Assuming a minimum stcrage area cf 18K and a source program of 400 cards (20% of which must be converted), 4 tracks should be assigned to SYS001 and 25 tracks each tc SYS002 and SYS003. Fcr a scurce program cf 800 cards, the allocation to SYS002 and SYS003 should be increased to 50 tracks.

COBOL LCP (IBM 2314/2319)

LCP work space depends on the size of the source program to be converted, and, to a lesser extent, on the amcunt of main stcrage available fcr processing. Assuming a minimum storage area of 18K and a source program of 400 cards (20% of which must be converted), 2 tracks shculd be assigned to SYS001 and 13 tracks each to SYS002 and SYS003. For a source program cf 800 cards, the allocation tc SYS002 and SYS003 shculd be increased tc 25 tracks.

Module 19: FORTRAN (BASIC)--360N-FO-451

Modular Outline

PARTITION SIZE

10,240 bytes

FORTRAN COMPATIBILITY

The FORTRAN IV library contains a module, ILFACOM, to provide compatibility with Basic FORTRAN subprograms. Because of differences in design objectives, object modules produced by the FORTRAN and Basic FORTRAN compilers are not compatible without this interface. With it, subprograms compiled under Basic FORTRAN can be incorporated into a FORTRAN program.

The compatibility module can be invoked on a job-by-job basis, or the system can be altered at system generation time so that it is brought in with all Basic FORTRAN linkage edits. The procedure for using it for a single job is discussed in the FORTRAN IV Programmer's Guide, listed in the Preface.

The system can be altered permanently by executing the jobstream book named Z.ILFMERGE which is in the source statement library of the DOS system residence volume supplied by IBM. This action deletes all Basic FORTRAN object time library routines from the relocatable library. It replaces them with references to the corresponding FORTRAN object time library routines and to ILFACOM. Subsequently, any Basic FORTRAN object module that refers to a library routine, such as the square root subprogram, IJTSSQT, actually uses the corresponding FORTRAN routine, in this case, ILFSSQRT.

This procedure is recommended only for installations that intend to convert entirely from Basic FORTRAN to FORTRAN, but do not want to recompile existing programs and subprograms. The action provides additional space in the relocatable library by eliminating the Basic FORTRAN routines, but it also increases the execution time of all Basic FORTRAN modules.

CORE IMAGE LIBRARY

	Blocks	
Phases	2311	2314/2319
4	20	20

Phases

FORTRAN
FORTREL
FORTRGE
FORTRPU

RELOCATABLE LIBRARY

	Modules	Blocks
FORTRAN, Basic:	5	143
Subroutine	37	149

Modules

IJTFO
IJTFO1
IJTFC2
IJTFO3
IJTFC4

Linkage Edit Statements

```
   INCLUDE IJTFO
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJTFO
DELETR IJTFO1
DELETR IJTFO2   Compiler Only
DELETR IJTFO3
DELETR IJTFO4
     or
DELETR IJT.ALL
```

SUBROUTINES

Module Name	Bytes of Main Storage
IJTAAFR	153
IJTACOM	2492
IJTACON	2464
IJTADIR	1120
IJTADXD	200
IJTADXI	140
IJTAIXI	156
IJTAPST	144
IJTARBE	1002
IJTARXI	136

IJTARXR	204
IJTDVCK	60
IJTEXPN	280
IJTFDMP	644
IJTFIOS	3762
IJTFXIT	24
IJTHEXC	140
IJTIFIX	112
IJTLEXP	476
IJTLLOG	365
IJTLSCN	388
IJTLSQT	164
IJTLTAN	312
IJTLTNH	332
IJTMAXD	101
IJTMODI	50
IJTMODR	104
IJTOVRF	76
IJTSINT	68
IJTSLIT	178
IJTSLOG	257
IJTSMX0	185
IJTSMX1	185
IJTSSCN	276
IJTSSQT	188
IJTSTAN	192
IJTSTNH	260

SOURCE STATEMENT LIBRARY

Macros	Blocks
1	5

Sample Problem

Z.F01

SAMPLE PROBLEM SUMMARY:

Program Name is Z.F01.

Problem Summary: Program Name is Z.F01. The Basic FORTRAN sample problem generates all prime numbers between 0 and 1000. Figure 19-1 defines the files required by the sample problem. Figure 19-2 illustrates the card deck retrieved from the source statement library.

	Disk Extent No.	
Phase	1	2
Compile	SYSLNK	SYS001
Link Edit	SYSLNK	SYS001
Execute	-	-

Figure 19-1. FORTRAN (Basic)--Sample Problem File Requirements

Cards Retrieved	Card Columns 73 - 76	Card Columns 77 - 80
CATALS Z.FC1		
BKEND Z.FC1		
FORTRAN Source Deck (22 cards) (Basic)	$451	0001 - 0022
BKEND		

Figure 19-2. FORTRAN (Basic)--Sample Problem Card Deck

To execute the Basic FORTRAN sample problem, the following job control cards are needed:

```
// JOB FORTRAN SAMPLE
// OPTION LINK,LIST,LOG
// EXEC FORTRAN
   FORTRAN (Basic) Source Deck (22 cards)
/*
// EXEC LNKEDT
// EXEC
/&
```

SYSLOG Output

```
// JOB FORTRAN SAMPLE
EOJ FORTRAN
```

SYSLST Output Summary:

- Job control cards
- Source program listing
- Object program storage map
- Linkage editor storage map
- List of prime numbers generated by the program

WORKFILES

IBM 2311

For a Basic FORTRAN program, 4 tracks should be assigned to SYSLNK and SYS001 for every 100 source statements.

IBM 2314/2319

For a Basic FORTRAN program, 2 tracks should be assigned to SYSLNK and to SYS001 for every 100 source statements.

Module 20: FORTRAN IV--360N-FO-479

Modular Outline

PARTITION SIZE

40,960 bytes

FORTRAN COMPATIBILITY

The FORTRAN library contains a module, ILFACOM, to provide compatibility with Basic FORTRAN subprograms. Because of differences in design objectives, object modules produced by the FORTRAN and Basic FORTRAN compilers are not compatible without this interface. With it, subprograms compiled under Basic FORTRAN can be incorporated into a FORTRAN program.

The compatibility module can be invoked on a job-by-job basis, or the system can be altered at system generation time so that it is brought in with all Basic FORTRAN linkage edits. The procedure for using it for a single job is discussed in the FORTRAN IV Programmer's Guide, listed in the Preface.

The system can be altered permanently by executing the jobstream book named Z.ILFMERGE which is in the source statement library of the DOS system residence volume supplied by IBM. This action deletes all Basic FORTRAN object time library routines from the relocatable library. It replaces them with references to the corresponding FORTRAN object time library routines and to ILFACOM. Subsequently, any Basic FORTRAN object module that refers to a library routine, such as the square root subprogram, IJTSSQT, actually uses the corresponding FORTRAN routine, in this case, ILFSSQRT.

This procedure is recommended only for installations that intend to convert entirely from Basic FORTRAN to FORTRAN, but do not want to recompile existing programs and subprograms. The action provides additional space in the relocatable library by eliminating the Basic FORTRAN routines, but it also increases the execution time of all Basic FORTRAN modules.

CORE IMAGE LIBRARY

	Blocks	
Phases	2311	2314/ 2319
8	43	43

Phases

FFORTRAN
FFORTRBK
FFORT1
FFORT2

FFORT3
FFORT4
FFORT5
FFORT9

RELOCATABLE LIBRARY

Modules	Blocks
9	316

Modules

ILFALL
ILFEXT
ILFFO
ILFFORT
ILFGEN

ILFPAR
ILFRCL
ILFTRBK
ILFUNF

Linkage Edit Statements

```
   INCLUDE ILFFO
// EXEC LNKEDT
```

Delete Statements

```
DELETR ILFFO
DELETR ILFFORT
DELETR ILFPAR
DELETR ILFALL
DELETR ILFUNF
DELETR ILFGEN
DELETR ILFEXT
DELETR ILFRCL
DELETS Z.ILFMERGE
(Compatibility
Merge Book)
```

SOURCE STATEMENT LIBRARY

Macros	Blocks
2	15

Macro

Z.ILFMERGE

Sample Problem

Z.ILFSAMPL

SAMPLE PROGRAM SUMMARY

Program Name is Z.ILFSAMPL. This sample problem generates all prime numbers between 2 and 1000. Figure 20-1 defines the files required by the sample problem. Figure 20-2 illustrates the cards punched when the sample program is retrieved from the source statement library.

	Disk Extent Number		
Phase	1	2	3
Compile	SYSLNK	SYS001	SYS002
Link Edit	SYSLNK	SYS001	-
Execute	-	-	-

Figure 20-1. FORTRAN IV--Sample Problem File Requirement

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-8
CATALS Z.ILFSAMPL		
BKEND Z.ILFSAMPL		
FORTRAN IV Source Deck (22 cards)	$479	0001-0022
BKEND		

Figure 20-2. FORTRAN IV--Sample Problem Card Deck

To execute the Fortran IV sample problem, the following job control statements are needed:

```
// JCB FCRTRAN4 SAMPLE
// OPTION LINK,LIST,LOG
// EXEC FFORTRAN
   FORTRAN IV Source Deck (22 cards)
/*
// EXEC LNKEDT
// EXEC
/*
/&
```

SYSLOG Output

```
// JCB FORTRAN4 SAMPLE
EOJ FORTRAN4
```

SYSLST Output Summary

- Job control cards
- Source program listing
- Object program storage map
- Linkage editor storage map
- List of prime numbers generated by the program

WORKFILES

FORTRAN Workfiles (IBM 2311)

For a FORTRAN program, six tracks should be assigned to SYSLNK and four tracks each to SYS001 and SYS002 for every 100 source statements.

FORTRAN Workfiles (IBM 2314/2319)

For FORTRAN program, three tracks should be assigned to SYSLNK and two tracks each to SYS001 and SYS002 for every 100 source statements.

Module 21: FORTRAN IV Subroutines -- 360N-LM-480

Modular Outline

FORTRAN COMPATIBILITY

The FORTRAN IV library contains a module, ILFACOM, to provide compatibility with Basic FORTRAN subprograms. Because of differences in design objectives, object modules produced by the FORTRAN and Basic FORTRAN compilers are not compatible without this interface. With it, subprograms compiled under Basic FORTRAN can be incorporated into a FORTRAN program.

The compatibility module can be invoked on a job-by-job basis, or the system can be altered at system generation time so that it is brought in with all Basic FORTRAN linkage edits. The procedure for using it for a single job is discussed in the FORTRAN IV Programmer's Guide, listed in the Preface.

The system can be altered permanently by executing the jobstream book named Z.ILFMERGE which is in the source statement library of the DOS system residence volume supplied by IBM. This action deletes all Basic FORTRAN object time library routines from the relocatable library. It replaces them with references to the corresponding FORTRAN object time library routines and to ILFACOM. Subsequently, any Basic FORTRAN object module that refers to a library routine, such as the square root subprogram, IJTSSQT, actually uses the corresponding FORTRAN routine, in this case, ILFSSQRT.

This procedure is recommended only for installations that intend to convert entirely from Basic FORTRAN to FORTRAN, but do not want to recompile existing programs and subprograms. The action provides additional space in the relocatable library by eliminating the Basic FORTRAN routines, but it also increases the execution time of all Basic FORTRAN modules.

RELOCATABLE LIBRARY

Modules	Blocks
61	286

FORTRAN IV Subroutines

Module Name	Bytes of Main Storage
ILFACOM	1472
ILFADCON	4333
ILFCLABS	172
ILFCLAS	216
ILFCLEXP	244
ILFCLLOG	260
ILFCLSCN	448
ILFCLSQT	200
ILFCSABS	156
ILFCSAS	196
ILFCSEXP	244
ILFCSLOG	248
ILFCSSCN	385
ILFCSSQT	188
ILFDEBUG	1885

ILFDIOCS	658
ILFFCDXI	308
ILFFCXPI	280
ILFFDUMP	465
ILFFDVCH	68
ILFFDXPD	208
ILFFDXPI	156
ILFFEXIT	28
ILFFINT	1482
ILFFIOCS	3589
ILFFIXPI	168
ILFFMAXD	109
ILFFMAXI	201
ILFFMAXR	201
ILFFOVER	80
ILFFRXPI	148
ILFFRXPR	212
ILFFSLIT	186
ILFGHTAB	256
ILFIBCOM	4063
ILFIBERR	206
ILFIASCN	400
ILFLATN2	500
ILFLERF	808
ILFLEXP	460
ILFLGAMA	728
ILFLLOG	376
ILFLSCN	400
ILFLSCNH	332
ILFLSQRT	150
ILFLTANH	340
ILFLTNCT	392
ILFNAMEL	2222
ILFSASCN	308
ILFSATN2	365
ILFSERF	456
ILFSEXP	286
ILFSGAMA	504
ILFSLOG	268
ILFSSCN	280
ILFSSCNH	280
ILFSSQRT	172
ILFSTANH	264
ILFSTNCT	296
ILFTRBK	816
ILFUNTAB	256

Module 22: PL/I (D) --360N-PL-464

Modular Outline

Prior to system generation, decide which PL/I (D) variant you desire and in which library the chosen variant resides.

VARIANTS OF PL/I (D)

Two variants of the PL/I (D) compiler can be built. One of the PL/I (D) variants requires 10K bytes (of problem program storage), while the other requires 12K. The 12K variant allows the system input and output files to be assigned to a 2311 disk drive or a 2314/2319, if the supervisor supports SYSFIL. PL/I (D) is capable of using either disk or tape workfiles, as are other compilers. Thus, PL/I (D) also furnishes compile-time device independence for workfiles.

Variants of PL/I (D) Library

Two variants of the regional transmitter are distributed: IJKTRGZ for overlapped, and IJKTRGM for non-overlapped. However, only one standard call is used at object time, which is for IJKTRGM. If the overlap version is required, the RENAMR function must be used to rename IJKTRGZ to IJKTRGM. When IJKTRGZ is used, special programming logic must be employed which compensates for the overlap mode (e.g., the last record is only written during CLOSE; therefore, any conditions arising are not signaled to your program).

PL/I (D) and System Input/Output on Disk

The two variants of the PL/I (D) compiler differ in their treatment of SYSIPT, SYSLST, and SYSPCH. If one or more of these logical units are assigned to a 2311 during compilation, the variant of the PL/I (D) compiler requiring 24K bytes of main storage (12K for the compiler itself), must be built.

If either SYSIPT or SYSLST is assigned to a 2311 or 2314/2319 for execution of PL/I (D) object programs, the I/O modules invoked to support these assignments must be retained. During system generation, a relocatable library maintenance run should be performed to rename these modules before deleting the PL/I (D) compiler from the relocatable library. The following control statements delete the I/O modules that do not support the disk facility from the relocatable library (because they are no longer needed), rename the PL/I (D) object time I/O modules that do support the disk facility, and delete the PL/I (D) compiler from the relocatable library:

```
// EXEC MAINT
  DELETR IJKSYSA,IJKSYSI
  RENAMR IJXSYSA,IJKSYSA,IJXSYSI,IJKSYSI
  DELETR IJX.ALL
```

You may want to retain the card modules that do not support the disk facility (if there is a possibility that card or tape input only and printer or tape output only are desired at some future time). The following control statements permit punching these modules. Punch these modules before they are deleted from the relocatable library as shown in the preceding example.

```
// EXEC RSERV
   PUNCH IJKSYSA,IJKSYSI
/*
```

If neither SYSIPT nor SYSLST is assigned to a 2311 or 2314/2319 during execution of PL/I (D) object programs, only the DELETR IJX.ALL statement is required for the maintenance run.

During system generation, utilities and workfile variants that are not needed can be deleted. If the system volume received is a disk pack, it is capable of operating as a system. If the system volume is a tape reel, it is self-loading tape that must be restored onto a disk pack before it is operable.

CORE IMAGE LIBRARY

	Phases	Blocks 2311	Blocks 2314/2319
PL/I (D)	110	278	282
Transients	1	1	1

Phases

PL/I
PL/IA10
PL/IA20
PL/IA25
PL/IA27

PL/IA30
PL/IA35
PL/IA45
PL/IA50
PL/IA60

PL/IA65
PL/IB10
PL/IB15
PL/IB20
PL/IB25

PL/IB25A
PL/IB30
PL/IB40
PL/IB70
PL/IB75

PL/IB80
PL/IB85
PL/IB85A
PL/IB87
PL/IB90

PL/IB92
PL/IB95
PL/IB97
PL/IC00
PL/IC03

PL/IC25
PL/IC30
PL/IC31
PL/IC32
PL/IC33

PL/IC34
PL/IC34A
PL/IC34B
PL/IC35
PL/IC37

PL/IC40
PL/IC50
PL/IC55
PL/IC60
PL/IC65

PL/IC85
PL/IC86
PL/IC95
PL/ID00
PL/ID03

PL/ID05
PL/ID10
PL/ID11
PL/ID12
PL/ID15

PL/ID17
PL/ID20
PL/ID20A
PL/ID40
PL/ID70

PL/ID75
PL/ID80
PL/IE25
PL/IE25A
PL/IE25B

PL/IE25C
PL/IE25D
PL/IE25E
PL/IE25F
PL/IE25G

PL/IE25H
PL/IE25I
PL/IE25J
PL/IE25K
PL/IE50

PL/IE55
PL/IE60
PL/IE60A
PL/IF25
PL/IF35

PL/IF50
PL/IF75
PL/IF90
PL/IF95
PL/IG00

PL/IG01
PL/IG15
PL/IG16
PL/IG17
PL/IG17B

PL/IG17D
PL/IG17E
PL/IG17R
PL/IG17S
PL/IG17X

PL/IG17Y
PL/IG20
PL/IG25
PL/IG30
PL/IG31

PL/IG40
PL/IG55
$IJKS00
$IJKS10
$IJKS20

$IJKS30
$IJKS40
$IJKS50
$IJKS60
$IJKS70

Note: The $$ and $ phases are cataloged into the core image library along with the compiler.

Transients

$$BPLOSE

RELOCATABLE LIBRARY

	Modules	Blocks
PL/I (D)	115	2335
Subroutines	95	360

Modules

IJXA00 PL/I (D) (SYSIPT, SYSPCH, SYSLST, never on a disk device

IJXA00D PL/I (D) (SYSIPT, SYSPCH, SYSLST, may be on a disk device

Note: The two variants differ in the first phase (PL/I (D) only. The following list applies to either variant.

IJXA10
IJXA20
IJXA25
IJXA27
IJXA30

IJXA35
IJXA45
IJXA50
IJXA60
IJXA65

IJXB10
IJXB15
IJXB20
IJXB25
IJXB30

IJXB40
IJXB70
IJXB75
IJXB80
IJXB85

IJXB87
IJXB90
IJXB92
IJXB95
IJXB97

IJXC00
IJXC03
IJXC25
IJXC30
IJXC31

IJXC32
IJXC33
IJXC34
IJXC34A
IJXC34B

IJXC35
IJXC37
IJXC40
IJXC50
IJXC55

IJXC60
IJXC65
IJXC85
IJXC86
IJXC95

IJXD00
IJXD03
IJXD05
IJXD10
IJXD11

IJXD12
IJXD15
IJXD17
IJXD20
IJXD20A

IJXD40
IJXD70
IJXD75
IJXD80
IJXE25

IJXE25A
IJXE25B
IJXE25C
IJXE25D
IJXE25E

IJXE25F
IJXE25G
IJXE25H
IJXE25I
IJXE25J

IJXE25K
IJXE50

IJXE55
IJXE60
IJXE61

IJXF25
IJXF35
IJXF50
IJXF75
IJXF90

IJXF95
IJXG00
IJXG01
IJXG15
IJXG16

IJXG17
IJXG17B
IJXG17D
IJXG17E
IJXG17R

IJXG17S
IJXG17X
IJXG17Y
IJXG20
IJXG25

IJXG30
IJXG31
IJXG40
IJXG55

IJXPLID
IJXPLI1
IJXPLI2
* IJXPLOSE

* IJXS00
* IJXS10
* IJXS20
* IJXS30
* IJXS40

* IJXS50
* IJXS60
* IJXS70
** IJXSYSA
** IJXSYSI

* At system generation time, these phases are cataloged into the core image library along with the PL/I (D) compiler.

**Subroutines for 2311: IJXSYSA and IJXSYSI should be either renamed to replace IJKSYSA or IJKSYSI, respectively, or deleted from the operational volume.

Linkage Edit Statements

For the IJXA00 (10K Variant)

```
   INCLUDE IJXPLI1
// EXEC LNKEDT
   INCLUDE IJXPLI2
// EXEC LNKEDT
```

For the IJXA00D (12K Variant)

```
   INCLUDE IJXPLID
// EXEC LNKEDT
```

Delete Statement

```
DELETR IJX.ALL,IJK.ALL
```

PL/I (D) SUBROUTINES

Module Name	Bytes of Main Storage
IJKEXHC	1264
IJKQAIM	456
IJKQASM	256
IJKQBIA	280
IJKQBSA	208
IJKQCLA	288
IJKQCSA	208
IJKQDLA	288
IJKQDSA	208
IJKQLLA	384
IJKQLSA	272
IJKQNID	608
IJKQNSD	480
IJKQQIM	160
IJKQQSM	176
IJKQRIB	768
IJKQRSB	408
IJKQSID	416
IJKQSSD	304
IJKQTIB	360
IJKQTSB	280
IJKRBBM	424
IJKRBIM	292
IJKRBKA	292
IJKREBM	92
IJKRELM	152
IJKREPM	140
IJKRESM	144
IJKRGIM	108
IJKRGKM	84
IJKRMBX	278
IJKRMLX	172
IJKRMPX	386
IJKRMSX	132
IJKRSBM	196
IJKRSLM	208
IJKRSPM	265
IJKRSSM	200
IJKRUBM	148
IJKRWBM	356
IJKRWLM	244
IJKRWPM	577
IJKRWSM	236
IJKRXLM	168
IJKRXSA	152

```
IJKSDMP              52
IJKSDTM              58
IJKSTMM             104
IJKSYSA             201
IJKSYSI             152

IJKSZBA            1804
IJKSZCA            1716
IJKSZLM              60
IJKTCBM             586
IJKTCUM             264

IJKTDIM             540
IJKTDPD             184
IJKTFDM             480
IJKTFMM             196
IJKTGDI             438

IJKTLCM             876
IJKTLIM            1158
IJKTLOM            1076
IJKTLTB               7
IJKTPSM              72

IJKTRGM             390
IJKTRGZ             398
IJKTRON            1416
IJKTSIM             668
IJKTSTM             674

IJKTXCF             702
IJKTXRM             420
IJKTXRN             410
IJKVBCM              60
IJKVBTM             132

IJKVCBM             238
IJKVCEM            1024
IJKVCFM             680
IJKVCPM             214
IJKVCTM             392

IJKVECM             404
IJKVFCM             536
IJKVGIM             254
IJKVIGM             148
IJKVIIM             236

IJKVNPM             376
IJKVPCM              68
IJKVPNM             316
IJKVPRM            1252
IJKVRPM             796

IJKVTBM             228
IJKVTCM             320
IJKXTBM             128
IJXSYSA             201
IJXSYSI             224
```

SOURCE STATEMENT LIBRARY

Macros	Blocks
2	17

Macro

A.IJKZL

Sample Problem

Z.PL1

SAMPLE PROBLEM SUMMARY

Program Name is Z.PL1. The PL/I (D) sample problem produces a table of mathematical functions:

SQRT(x^2+1),x^2,x^3,SIN(x),COS(x),SQRT(x), and SQRT(x^3).

Figure 22-1 illustrates the card deck punched when the sample problem is retrieved from the source statement library.

Card Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.PL1		
ENDD Z.PL1		
PL/I (D) Source Deck (38 cards)	$464	0001-0038
*END SOURCE DECK/BEGIN INPUT DATA		
PL/I (D) data (1 card)		
ENDD		

Figure 22-1. PL/I (D) --Sample Problem Card Deck

To execute the PL/I (D) sample problem, the following job control cards are needed (if the user has the 48-character chain, he should change the 60C parameter in the option card to 48C):

```
// JOB PL/I SAMPLE
// OPTION LINK,NODECK,SYM,LISTX,60C,XREF
// EXEC PL/I
*  PROCESS STMT
   PL/I (D) Source Deck (38 Cards)
/*
// EXEC LNKEDT
// EXEC
   PL/I (D) Data (1 Card)
/*
/&
```

SYSLOG Output:

```
// JOB PL/I
EOJ PL/I
```

SYSLST Output Summary

- Job control cards
- Source program listing
- Symbol table listing
- Cross-reference listing
- Diagnostics
- Offset table listing
- Object program listing
- External symbol table
- Block table (automatic storage requirements)
- Linkage editor storage map
- Table produced by object program

WORKFILES

PL/I (D) Workfiles (IBM 2311)

For each 100 PL/I (D) source statements, three tracks should be reserved for SYSLNK; five tracks for SYS001; and seven tracks each for SYS002 and SYS003. This estimate is based on the following assumptions:

1. There are about 30 variable names per 100 statements in the external procedure.
2. The number of PL/I (D) syntactical elements per source statement (except DECLARE statements and format lists) is about 10. For example, the statement
```
A=B+C;
```
has 6 syntactical elements, and the statement
```
READ FILE (F) INTO (AREA7);
```
has 10 syntactical elements.

PL/I (D) Workfiles (IBM 2314/2319)

For each 200 PL/I (D) statements, three tracks should be reserved for SYSLNK; five tracks for SYS001; and seven tracks each for SYS002 and SYS003. This estimate is based upon the following assumptions:

1. There are about 30 variable names per 100 statements in the external procedure.
2. The number of PL/I (D) syntactical elements per source statement (except DECLARE statements and format lists) is about 10. For example, the statement
```
A=B+C;
```
has 6 syntactical elements; and the statement
```
READ FILE (F) INTO (AREA7);
```
has 10 syntactical elements.

Module 23: RPG--360N-RG-460

Modular Outline

PARTITION SIZE

10,240 bytes

CORE IMAGE LIBRARY

Phases	Blocks	
	2311	2314/2319
28	98	101

Phases

RPG1
RPG10010
RPG10020
RPG10025
RPG10030

RPG10040
RPG10050
RPG10060
RPG10070
RPG10080

RPG10090
RPG10100
RPG10110
RPG10120
RPG10130

RPG10140
RPG10150
RPG10160
RPG10170
RPG1018A

RPG1018G
RPG10180
RPG10190
RPG10200
RPG10210

RPG10220
RPG10230
RPG10240

RELOCATABLE LIBRARY

Modules	Blocks
60	785

Modules

IJRRG
IJR000
IJR010
IJR020
IJR025

IJR030
IJR039
IJR040
IJR049
IJR050

IJR059
IJR060
IJR069
IJR070
IJR079

IJR080
IJR089
IJR090
IJR099
IJR100

IJR109
IJR110
IJR119
IJR120
IJR129

IJR130
IJR139
IJR140
IJR149
IJR150

IJR159
IJR160
IJR169
IJR170
IJR179

IJR18A
IJR18F
IJR18G
IJR18H
IJR180

IJR189
IJR190
IJR199
IJR200
IJR209

IJR210
IJR219
IJR220
IJR229
IJR230

IJR239
IJR240
IJR241
IJR242
IJR243

IJR244
IJR245
IJR246
IJR247
IJR249

SOURCE STATEMENT LIBRARY

Macro	Blocks
1	13

Linkage Edit Statements

```
   INCLUDE IJRRG
// EXEC LNKEDT
```

Sample Problem

Z.RG1

Delete Statement

```
DELETR IJR.ALL
```

SAMPLE PROBLEM SUMMARY

Program Name is Z.RG1. The RPG sample problem produces an accounts receivable register. Both the RPG source deck and data deck are retrieved from the maintenance volume under Retrieving the Sample Problems in Module 1. Figure 23-1 defines the files required by this problem. Figure 23-2 illustrates the cards punched when the sample program is retrieved from the source statement library.

	Disk Extent Number				
Phase	1	2	3	4	5
Compile	SYSLNK	SYS001	SYS002	SYS003	SYS004
Link Edit	SYSLNK	SYS001	-	-	-
Execute	-	-	-	-	-

Figure 23-1. RPG--Sample Problem File Requirements

Card Retrieved	Card Cols. 73-76
CATALS Z.RG1	
BKEND Z.RG1	
RPG Source Deck (45 cards)	$460
*END SOURCE DECK/ BEGIN INPUT DATA	
RPG Data (13 cards)	
BKEND	

Figure 23-2. RPG--Sample Problem Card Deck

To execute the sample problem, the following job control cards are needed:

```
// JOB RGP SAMPLE
// OPTION LINK,DUMP
// EXEC RPG
   RPG Source Deck (45 cards)
/*
// EXEC LNKEDT
// EXEC
   RPG Data (13 cards)
/*
/&
```

SYSLOG Output

```
// JOB RPG
EOJ RPG
```

SYSLST Output Summary:

- Job control cards
- Source program listing
- Symbol tables
- Memory map
- Linkage editor storage map
- Accounts receivable register generated by program

WORKFILES

RPG Workfiles (IBM 2311)

For an RPG program of about 150 source statements, eight tracks should be assigned to SYSLNK and to SYS003, and three tracks to SYS001 and to SYS002.

For an RPG program of about 500 source statements, 22 tracks should be assigned to SYSLNK; 10 tracks to SYS001 and to SYS002; and 26 to SYS003.

For an PRG program of about 1,000 source statements, 44 tracks should be assigned to SYSLNK; 20 tracks each to SYS001 and to SYS002; and 52 tracks to SYS003.

RPG Workfiles (IBM 2314/2319)

For an RPG program of about 150 source statements, five tracks should be assigned to SYSLNK and to SYS003, and two tracks to SYS001 and to SYS002.

For a RPG program of about 500 source statements, 11 tracks should be assigned to SYSLNK, 5 tracks to SYS001 and to SYS002, and 13 tracks to SYS003.

For an RPG program of about 1,000 source statements, 22 tracks should be assigned to SYSLNK, 10 tracks to SYS001 and to SYS002, and 26 tracks to SYS003.

Module 24: Disk Sort/Merge--360N-SM-450

Modular Outline

PARTITION SIZE

10,240 bytes

CORE IMAGE LIBRARY

	Blocks	
Phases	2311	2314/2319
26	74	76

Phases

DSORT
DSORT002
DSORT003
DSORT004
DSORT005

DSORT006
DSORT007
DSORT008
DSORT009
DSORT010

DSORT101
DSORT102
DSORT103
DSORT104
DSORT105

DSORT201
DSORT202
DSORT203
DSORT204
DSORT301

DSORT302
DSORT303
DSORT304
DSORT401
DSORT402
DSORT501

RELOCATABLE LIBRARY

Modules	Blocks
38	521

Modules
The program numbers given in the program numbers column identify the modules required to support the Disk Sort/Merge program(s) selected. For identification and descriptions of the referenced programs see Linkage Edit Statements. Thus, the number 1 next to a module indicates that it is required by Program Number 1, which is the entire sort/merge program. Where more than one program number appears next to a module, that module is required by all these programs.

	Program Number
IJOSM	1
IJOSMERG	7
IJOSMFOM	2
IJOSMFOS	5
IJOSMFVS	4
IJOSMF23	1,2,4,5
IJOSMPH0	1 through 7
IJOSMPH1	1 through 6
IJOSMPH4	1,2,3,7
IJOSMVOM	3
IJOSMVOS	6
IJOSMV23	1,3,4,6
IJOSM001	1 through 7
IJOSM002	1 through 7
IJOSM003	1 through 7
IJOSM004	1 through 7
IJOSM005	1 through 7
IJOSM006	1 through 7
IJOSM007	1 through 7
IJOSM008	1 through 7
IJOSM009	1 through 7
IJOSM010	1 through 7
IJOSM101	1 through 6
IJOSM102	1 through 6
IJOSM103	1 through 6
IJOSM104	1 through 6
IJOSM105	1 through 6
IJOSM201	1,2,4,5
IJOSM202	1,2,4,5
IJOSM203	1,3,4,6
IJOSM204	1,3,4,6
IJOSM301	1,2,4,5
IJOSM302	1,2,4,5
IJOSM303	1,3,4,6
IJOSM304	1,3,4,6
IJOSM401	1,2,3,7
IJOSM402	1,2,3,7
IJOSM501	1,2,4,5

Linkage Edit Statements

The information given enables a user to linkage edit into the core image library only those modules required to tailor a generalized sort/merge program to specific job applications. At system generation time, the user can linkage edit any one of seven distinct sort/merge object programs into the core image library.

The following programs can be generated at linkage edit time:

1. The entire sort/merge program.
2. A program that
   a. sorts fixed-length records;
   b. performs the ADDROUT option for fixed or variable-length record; and
   c. merges fixed or variable-length records.
3. A program that
   a. sorts variable-length records;
   b. performs the ADDROUT option for fixed or variable-length records; and
   c. merges fixed or variable-length records.
4. A program that
   a. sorts fixed or variable-length records; and
   b. performs the ADDROUT option for fixed or variable-length records.
5. A program that
   a. sorts fixed-length records; and
   b. performs the ADDROUT option for fixed or variable-length records.
6. A program that
   a. sorts variable-length records; and
   b. performs the ADDROUT option for fixed or variable-length records.
7. A program that merges fixed or variable-length records.

The user should linkage edit only the sort/merge program that satisfies his particular requirements. To generate:

Program 1. specify: INCLUDE IJOSM
// EXEC LNKEDT

Program 2. specify: INCLUDE IJOSMFOM
// EXEC LNKEDT

Program 3. specify: INCLUDE IJOSMVOM
// EXEC LNKEDT

Program 4. specify: INCLUDE IJOSMFVS
// EXEC LNKEDT

Program 5. specify: INCLUDE IJOSMFOS
// EXEC LNKEDT

Program 6. specify: INCLUDE IJOSMVOS
// EXEC LNKEDT

Program 7. specify: INCLUDE IJOSMERG
// EXEC LNKEDT

Delete Statement

```
DELETR IJC.ALL
```

Macros	Blocks
1	2

Sample Problem

```
Z.SM2
```

SAMPLE PROBLEM SUMMARY

Program Name is Z.SM2. The Disk Sort/merge sample problem rearranges, in ascending sequence, the numbers generated by the Assembler 2 problem. Figure 24-1 defines the file requirements. Figure 24-2 illustrates the card deck that is retrieved from the source statement library.

Disk Extent Number	
1	2
SYS002	SYS004

Figure 24-1. DISK SORT/MERGE--Sample Problem File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80	Generates Input for
CATALS Z.SM2 BKEND Z.SM2 S/M Control Cards (6 cards) BKEND	$450	0001-0006	Disk-to-Printer Utility, Z.UTDKPR1

Figure 24-2. DISK SORT/MERGE--Sample Problem Card Deck

To execute the disk sort/merge sample problem, the following job control cards are needed:

```
                Col. 16                                  Col. 54      Col. 72
                  ↓                                        ↓            ↓
// JOB DSORT
// PAUSE ASSGN SYS002 AND SYS004 to SCRATCH PACK
// ASSGN SYS002,X'cuu'
// ASSGN SYS004,X'cuu'
// DLBL FILEA,'SAMPLE PROBLEM FILE OF 2000 RECORDS'[,yy/ddd]
// EXTENT SYS002,hhhhhh,1,0,1800,43
// DLBL FILEW,'SAMPLE PROBLEM WORK AREA FOR SORT RUN'[,yy/ddd]
// EXTENT SYS002,hhhhhh,1,0,1843,23
// DLBL FILEC,'SORTED FILE OF 2000 RECORDS FOR UTILITIES'[,yy/ddd]
// EXTENT SYS004,hhhhhh,1,0,1800,43
// EXEC DSORT
   S/M Control Cards (6 cards)
/*
/&
```

Note: Replace hhhhhh in the EXTENT cards with the volume serial number of the disk pack.

SYSLOG Output

```
// JOB DSORT
// PAUSE  ASSGN SYS002 AND SYS004 TO SCRATCH PACK
EOJ DSORT
```

SYSLST Output Summary:

- Job control cards
- Sort control cards
- Sort parameters
- Computed constants

Module 25: Tape and Disk Sort/Merge--360N-SM-483

Modular Outline

PARTITION SIZE

- Tape 2311 Variant 10,240 bytes
- 2314 Variant 22,528 bytes

CORE IMAGE LIBRARY

	Blocks	
Phases	2311	2314/2319
40	69	71

Phases

SORT
SORTASA
SORTRAA
SORTRAB
SORTRAC

SORTRAD
SORTRBA
SORTRBB
SORTRBC
SORTRBD

SORTRCB
SORTRCC
SORTRCD
SORTRCE
SORTRCF

SORTRCH
SORTRCJ
SORTRCK
SORTRCL
SORTRCN

SORTRDA
SORTRDB
SORTRDC
SORTRGA
SORTRGB

SORTRGG
SORTRGH
SORTRGF
SORTROA
SORTROB

SORTROC
SORTRPA
SORTRPB
SORTRSD
SORTRSE

SORTRSG
SORTRSH
SORTRSJ
SORTRSM
SORTRSN

RELOCATABLE LIBRARY

Modules	Blocks
101	987

Modules

The following keys identify the modules required to support the sort program selected; see Linkage Edit Statements.

Keys: A - All programs
D - Direct Access (2311,2314/2319) Sort Programs
M - Merge Program
S - Entire Sort Program
T - 2400 Sort Program

	Key
ILHSAAC	T
ILHSAAD	D
ILHSABF	S
ILHSABG	A
ILHSAGD	T
ILHSAGE	D
ILHSAGF	S
ILHSAGG	D
ILHSAGH	M
ILHSALL	S,M
ILHSAOB	S
ILHSACC	A
ILHSAPC	T
ILHSAPD	D
ILHSAPF	S
ILHSAPG	D
ILHSAPH	M
ILHSASA	S
ILHSASF	S
ILHSASG	S
ILHSASK	A
ILHSASL	A
ILHSD1	D
ILHSD4	D
ILHSMRG	M
ILHSNMPS*	D
ILHSCRT	S
ILHSPHN*	D
ILHSPH0	A
ILHSPH0M	M
ILHSPH0S	S

ILHSPH0T	T
ILHSPH01	D
ILHSPH04	D
ILHSPH1	S
ILHSPH1T	T
ILHSPH11	D
ILHSPH14	D
ILHSPH2	S
ILHSPH2T	T
ILHSPH21	D
ILHSPH24	D
ILHSPH3	S,M
ILHSPH3M	M
ILHSPH3S	S
ILHSPH3T	T
ILHSPH31	D
ILHSPH34	D
ILHSPPI	A
ILHSRAA	T
ILHSRAB	D
ILHSRAC	T
ILHSRAD	D
ILHSRBA	S
ILHSRBB	T
ILHSRBC	D
ILHSRBD	S
ILHSRBF	S
ILHSRBG	A
ILHSRCA	A
ILHSRCB	A
ILHSRCC	A
ILHSRCD	A
ILHSRCE	T
ILHSRCF	D
ILHSRCH	M
ILHSRCI	A
ILHSRCJ	A
ILHSRCK	A
ILHSRCL	A
ILHSRCM	A
ILHSRCN	A
ILHSRDA	S
ILHSRDB	S
ILHSRDC	D
ILHSRGA	S
ILHSRGB	S
ILHSRGD	T
ILHSRGE	D
ILHSRGF	S
ILHSRGG	D
ILHSRGH	M
ILHSRMC	A,S
ILHSROA	S
ILHSROB	S
ILHSROC	A
ILHSRPA	T
ILHSRPB	D
ILHSRPC	T
ILHSRPD	D
ILHSRPE	A
ILHSRSD	S
ILHSRSE	S
ILHSRSG	S
ILHSRSH	S
ILHSRSI	S
ILHSRSJ	A
ILHSRSM	M,S
ILHSRSN	A
ILHSRTMG	A
ILHST	T

*Required for systems without multiprogramming support.

Linkage Edit Statements

The information given enables a user to linkage edit into the core image library only those modules required to tailor a generalized sort/merge program to specific job applications. At system generation time, the user can linkage edit any one of six distinct sort/merge object programs into the core image library. Associated with each sort program is a key. The key identifies the module(s) required to support the corresponding program. For example, Program Number 1 is the entire sort/merge program and is assigned the keys S, M. Thus, all modules identified by keys S or M are required to support Program Number 1. See Relocatable Library for a complete description of the keys and a list of the modules required to support the disk sort/merge programs described.

1. The entire sort/merge program (key = S, M).

2. The entire sort program (key = S).

3. The merge program (key = M).

4. A sort program that uses 2400 tape units as intermediate storage (key = T).

5. A sort program that uses 2311 direct access devices as intermediate storage (key = D).

6. A sort program that uses 2314/2319 direct access devices as intermediate storage (key = D).

To conserve library space, the user should linkage edit only the sort/merge program that satisfies his particular requirements. Note that an attempt to execute an option not selected results in

abnormal termination of the sort/merge program. To generate:

Program 1. specify: INCLUDE ILHSALL
// EXEC LNKEDT

Program 2. specify: INCLUDE ILHSCRT
// EXEC LNKEDT

Program 3. specify: INCLUDE ILHSMRG
// EXEC LNKEDT

Program 4. specify: INCLUDE ILHST
// EXEC LNKEDT

Program 5. specify: INCLUDE ILHSD1
// EXEC LNKEDT

Program 6. specify: INCLUDE ILHSD4
// EXEC LNKEDT

If the sort/merge program is generated in a system without multiprogramming support, the module ILHSNMPS must be linkage edited immediately after the selected program, such as:

```
   INCLUDE ILHSALL
// EXEC LNKEDT
   INCLUDE ILHSNMPS
// EXEC LNKEDT
```

Note: If standard label tapes are processed or a user's exit processes nonsequential disk labels, include a // LBLTYP in the linkage edit statements of the appropriate program.

Delete Statement

DELETR ILH.ALL

SOURCE STATEMENT LIBRARY

Macros	Blocks
3	6

Sample Problems

Z.SM4
Z.SM5
Z.SM6

SAMPLE PROBLEM SUMMARY

Figure 25-1 defines the files required by the various sample problems. Figure 25-2 illustrates the card decks punched when the problems are retrieved from the source statement library.

Sample	Disk	Tape Drive Number				
Problem	Ext. #1	1	2	3	4	5
Z.SM4	SYSRES	SYS001	SYS002	SYS003	SYS004	SYS005

Note: Tape drive number refers to X'cuu' in the ASSGN card.

Figure 25-1. TAPE and DISK SORT/MERGE--Sample Problems, File Requirements (Part 1 of 2)

Sample	Disk Extent Number		
Problem	1	2	3
Z.SM5 and Z.SM6	SYS001	SYS002	SYS003

Figure 25-1. TAPE and DISK SORT/MERGE--Sample Problems, File Requirements (Part 2 of 2)

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80	Secondary Storage Device	Generates Input for
CATALS Z.SM4 BKEND Z.SM4 S/M Control Cards (6 cards) BKEND	$483	0001-0006	2400	Tape-to-Printer Utility, Z.UTTPPR2
CATALS Z.SM5 BKEND Z.SM5 S/M Control Cards (6 cards) BKEND	$483	0101-0106	2311	Disk-to-Printer Utility, Z.UTDKPR2
CATALS Z.SM6 BKEND Z.SM6 S/M Control Cards (6 cards) BKEND	$483	0201-0206	2314	Disk-to-Printer Utility, Z.UTDKPR3

Figure 25-2. TAPE and DISK SORT/MERGE--Sample Problem Card Decks

TAPE AND DISK SORT/MERGE -- 2400 APPLICATION

Program Name is Z.SM4. The Tape and Disk Sort/Merge sample problem rearranges, in ascending sequence, the records produced by the Assembler 4 program and writes them on another tape, five records per block.

To execute the Tape and Disk Sort/Merge sample problem, the following job control cards are needed:

```
// JOB SORT 2400
// PAUSE ASSGN SYS002 TO SAME DRIVE AS PREVIOUS JCB
// PAUSE ASSGN SYS001,SYS003,SYS004,SYS005 TO DIFF TAPES
// ASSGN SYS001,X'cuu'
// ASSGN SYS002,X'cuu'
// ASSGN SYS003,X'cuu'
// ASSGN SYS004,X'cuu'
// ASSGN SYS005,X'cuu'
// TLBL SORTOUT,,64/001
// TLBL SORTIN1,,64/001
// TLBL SORTWK1,,64/001
// TLBL SORTWK2,,64/001
// TLBL SORTWK3,,64/001
// LBLTYP TAPE
// EXEC SORT
   Sort/merge control cards (6 cards)
/&
```

SYSLOG Output

```
BG // JOB SORT 2400
BG // PAUSE ASSGN SYS002 TO SAME DRIVE AS PREVIOUS JCB
BG
BG // PAUSE ASSGN SYS001,SYS003,SYS004,SYS005 TO DIFF TAPES
BG
BG 7905I RCD IN0002000, OUT 002000, ESTIMATED 0000000
BG 7101I END SORT PH
BG 7905I RCD IN 0002000, OUT 0002000, ESTIMATED 0000000
BG 7201I END MERGE PH
BG 7905I RCD IN 0002000, OUT 0002000, ESTIMATED 0000000
BG 7302I EOJ
BG EOJ SORT
```

SYSLST Output Summary:

- Job control cards
- Sort control cards
- Sort parameters
- Computed constants

TAPE AND DISK SORT/MERGE--2311 APPLICATION

Program Name is Z.SM5. The Tape and Disk Sort/Merge sample problem rearranges, in ascending sequence, the records produced by the Assembler 5 program and writes them to a 2311 disk storage device.

To execute the tape and disk sort/merge sample problem, the following control cards are needed:

```
             Col. 16                                  Col. 54      Col. 72
               ↓                                        ↓            ↓
// JOB SORT 2311
// PAUSE ASSGN SYS001,SYS002,SYS003 TO SCRATCH PACK
// ASSGN SYS001,X'cuu'
// ASSGN SYS002,X'cuu'
// ASSGN SYS003,X'cuu'
// DLBL SORTCUT,'SORTED FILE OF 2000 RECORDS FOR UTILITIES'[,yy/ddd]
// EXTENT SYS001,hhhhhh,1,0,1710,33
// SORTIN1,'SAMPLE PROBLEM FILE OF 2000 RECORDS'[,yy/ddd]
// EXTENT SYS002,hhhhhh,1,0,1800,43
// DLBL SORTWK1,'SAMPLE PROGRAM WORK AREA FOR SORT RUN'[,yy/ddd]
// EXTENT SYS003,hhhhhh,1,0,1843,33
// EXEC SORT
   Sort/merge control cards (6 cards)
/&
```

Note: Replace hhhhhh in the EXTENT cards with the volume serial number of the disk pack.

SYSLOG Output

```
BG // JOB SORT 2311
BG // PAUSE ASSGN SYS001,SYS002,SYS003 TO SCRATCH PACK
BG
BG 7905I RCD IN 0002000, OUT 0002000, ESTIMATED 0002000
BG 7101I END SORT PH
BG 7905I RCD IN 0002000, OUT 0002000, ESTIMATED 0002000
BG 7201I END MERGE PH
BG 7905I RCD IN 0002000, OUT 0002000, ESTIMATED 0002000
BG 7302I EOJ
BG EOJ SORT
```

SYSLST Output Summary:

- Job control cards
- Sort control cards
- Sort parameters
- Computed constants

TAPE AND DISK SORT/MERGE--2314/2319 APPLICATION

Program Name is Z.SM6. The Tape and Disk Sort/Merge sample problem rearranges the records produced by the Assembler 6 program in ascending sequence and writes them back to a 2314/2319 direct access storage facility.

To execute the tape and disk sort/merge sample problem, the following job control cards are needed:

```
              Col. 16                                  Col. 54      Col. 72
                 ↓                                        ↓           ↓
// JOB SORT 2314/2319
// PAUSE ASSGN SYS001,SYS002,SYS003 TO SCRATCH PACK
// ASSGN SYS001,S'cuu'
// ASSGN SYS002,X'cuu'
// ASSGN SYS003,X'cuu'
// DLBL SORTOUT,'SORTED FILE OF 2000 RECORDS FOR UTILITIES'[,yy/ddd]
// EXTENT SYS001,hhhhhh,1,0,3420,63
// DLBL SORTIN1,'SAMPLE PROBLEM FILE OF 2000 RECORDS'[,yy/ddd]
// EXTENT SYS002,hhhhhh,1,0,3600,83
// DLBL SORTWK1,'SAMPLE PROGRAM WORK AREA FOR SORT RUN'[,yy/ddd]
// EXTENT SYS003,hhhhhh,1,0,3686,63
// EXEC SORT
   Sort/merge control cards (6 cards)
/&
```

Note: Replace hhhhhh in the EXTENT cards with the volume serial number of the disk pack.

SYSLOG Output

```
BG // JOB SORT 2314/2319
BG // PAUSE ASSGN SYS001,SYS002,SYS003 TO SCRATCH PACK
BG
BG 7905I RCD IN 0003000, OUT 0003000, ESTIMATED 0003000
BG 7101I END SORT PH
BG 7905I RCD IN 0003000, OUT 0003000, ESTIMATED 0003000
BG 7201I END MERGE PH
BG 7905I RCD IN 0003000, OUT 0003000, ESTIMATED 0003000
BG 7302I EOJ
BG EOJ SORT
```

SYSLST Output Summary:

- Job control cards
- Sort control cards
- Sort parameters
- Computed constants

Module 26: Tape Sort/Merge -- 360N-SM-400

Modular Outline

PARTITION SIZE

10,240 bytes

Modules	Blocks
5	206

CORE IMAGE LIBRARY

Phases	Blocks	
	2311	2314/2319
20	36	36

Phases

TSRT
TSRTP002
TSRTP003
TSRTP004
TSRTP005

TSRTP006
TSRTP007
TSRTP008
TSRTP101
TSRTP102

TSRTP103
TSRTP104
TSRTP105
TSRTP201
TSRTP202

TSRTP203
TSRTP204
TSRTP301
TSRTP302
TSRTP303

RELOCATABLE LIBRARY

Modules

IJPSM
IJPSM001
IJPSM002
IJPSM003
IJPSM004

Linkage Edit Statement

```
   INCLUDE IJPSM
// LBLTYP TAPE(nn)
// EXEC LNKEDT
```

Delete Statement

```
DELETR IJP.ALL
```

SOURCE STATEMENT LIBRARY

Macros	Blocks
1	2

Sample Problem

Z.SM1

SAMPLE PROBLEM SUMMARY

Program Name is Z.SM1. The Tape Sort/Merge sample problem rearranges the records produced by the Assembler 1 program in ascending sequence and writes them on another tape, five records per block. Figure 26-1 defines the sample problem file requirements. Figure 26-2 illustrates the card deck retrieved from the source statement library.

Disk Extent #1	Tape Drive Number 1	2	3
SYSRES	SYS004	SYS001 SYS003	SYS002 SYS005

Note:
- Tape drive number refers to x'cuu' operand in the ASSGN card.
- If a 7-track drive is used, the third operand (X'B8') is required tc turn byte convert off.

Figure 26-1. TAPE SORT/MERGE--Sample Prcblem, File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80	Generates Input fcr
CATALS Z.SM1 BKEND Z.SM1 S/M Control Cards (6 cards) BKEND	$400	0001-0006	Tape-to-Printer Utility Z.UTTPPR1

Figure 26-2. TAPE SORT/MERGE--Sample Prcblem Card Deck

To execute the Tape Sort/Merge sample prcblem, the fcllowing jcb control cards are needed:

```
// JOB SORT
// PAUSE ASSGN SYS002 TO SAME TAPE DRIVE AS IN PREVIOUS JOB
// ASSGN SYS002,X'cuu'
// PAUSE ASSGN SYS001,SYS003,SYS004,SYS005 TO DIFF TAPES
// ASSGN SYS001,X'cuu'
// ASSGN SYS003,X'cuu'
// ASSGN SYS004,X'cuu'
// ASSGN SYS005,X'cuu'
// EXEC TSRT
   S/M Control Cards (6 cards)
/*
/&
```

SYSLOG Output

```
// JOB SORT
// PAUSE  ASSIGN SYS002 TO SAME TAPE DRIVE AS IN PREVICUS JCB
// PAUSE  ASSGN SYS001,SYS003,SYS004,SYS005 TO DIFF TAPES
EOJ SORT
```

SYSLST Output Summary:

- Job control cards
- Sort control cards
- Sort parameters
- Computed constants

Module 27: OLTEP--370N-DN-481

Modular Outline

PARTITION SIZE

14,336 bytes
18,432 bytes with RETAIN active

CORE IMAGE LIBRARY

		Blocks		
	Phases	2311	2314/2319	3330
OLTEP	54	63	64	67
B-Transients	4	4	4	4

Phases

IJZACEOM
IJZACKTP
IJZACMNT
IJZACOMP
IJZACONV

IJZADO00
IJZADO02
IJZADO03
IJZADO07
IJZADO10

IJZADO11
IJZADO16
IJZADO21
IJZADO22
IJZADO24

IJZADO26
IJZADO28
IJZADO31
IJZADO32
IJZADO33

IJZADO35
IJZADO36
IJZADO37
IJZADO39
IJZADO40

IJZADO41
IJZADO42
IJZADO43
IJZADO44
IJZADO45

IJZADO47
IJZADO52
IJZADO56
IJZADO57
IJZADO62

IJZADO64
IJZADO70
IJZADO71
IJZADO76
IJZADO77

IJZADO85
IJZADO98
IJZADO99
IJZADOAA
IJZADOAB
IJZADOAC

IJZADOAJ
IJZADOLT
IJZADPRT
IJZAEXIO

IJZAHEAD
IJZAOPUT
IJZARATA
IJZARSIT
IJZAWAIT

B-Transients

$$BOLTEP
$$BSCPEN
$$BZTIME
$$BTCLTP

Note: The B-Transients are cataloged into the core image library along with the OLTEP component.

CONFIGURATION DATA SET

A Configuration Data Set (CDS) is required in the core image library for each device on the system and one for the CPU. This is the formula for calculating the block requirements:

$$\frac{\text{number of devices on system} + 1}{6} = \text{number of core image library blocks}$$

ON-LINE TESTS (OLTS)

The number of core image library blocks required for the On-Line Tests (OLTs) is dependent on the number of devices types on the system, and the number and size of the OLTs needed to test these devices. This information can be found in the writeup that accompanies the OLTs.

GENERATING THE OLT/CDS LIBRARY

The Program Information Department (PID) supplies the on-line tests and configuration data sets on tape or on disk if there are no tape devices on the system. This tape is generated by PID from

information supplied by the customer engineer.

The OLTs and CDSs must be link-edited into the core image library before OLTEP can execute. To do this, use the Editor function of the Stand Alone Service Program (SOSP), which is included on the tape. Also use this program to add new tests as they are made available and to generate additional CDSs when you add devices to the system.

RELOCATABLE LIBRARY

Modules	Blocks
59	463

Modules

IJZABOOK
IJZACEOM
IJZACKTP
IJZACMNT
IJZACOMP

IJZACONV
IJZADO00
IJZADO02
IJZADO03

IJZADO07
IJZADO10
IJZADO11
IJZADO16
IJZADO21

IJZADO22
IJZADO24
IJZADO26
IJZADO28
IJZADO31

IJZADO32
IJZADO33
IJZADO35
IJZADO36
IJZADO37

IJZADO39
IJZADO40

IJZADO41
IJZADO42
IJZADO43

IJZADO44
IJZADO45
IJZADO47
IJZADO52
IJZADO56

IJZADO57
IJZADO62
IJZADO64
IJZADO70
IJZADO71

IJZADO76
IJZADO77
IJZADO78
IJZADOAA
IJZADOAB

IJZADOAC
IJZADOAJ
IJZADOLT
IJZADPRT
IJZAEXIC

IJZAHEAD
IJZAOLTP
IJZACPEN
IJZAOPUT
IJZARATA

IJZARSLT
IJZATIME
IJZATOLT
IJZAWAIT

Linkage Edit Statements

```
   INCLUDE IJZABOOK
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJZ.ALL
```

Module 28: BTAM--370N-CQ-469

Modular Outline

Modular Index

CORE IMAGE LIBRARY

		Blocks		
	Phases	2311	2314/2319	3330
BTAM On-Line Terminal Test Patterns	52	52	52	52
Transients	7	7	7	7
BTAM Transient Routines (other than test patterns)	19	19	19	19

BTAM On-Line Terminal Test Pattern Phase Identification

Phase Names	Test Patterns for On-Line Terminal Tests
IJLBOT02	Transparent EBCDIC Message
IJLBOT03	USASCII Transparency Message
IJLBOT04	Normal EBCDIC Message
IJLBOT05	Normal USASCII Message
IJLBOT06	Alphameric USASCII Message
IJLBOT07	USASCII Printer Message
IJLBOT08	USASCII Punch Message
IJLBOT09	TRANSCODE Printer Message
IJLBOT10	TRANSCODE Punch Message
IJLBOT11	TRANSCODE Multipcint Message
IJLBOT12	EBCDIC Printer Message
IJLBOT13	EBCDIC Punch Message
IJLBOT14	EBCDIC Alphameric Message
IJLBOT15	EBCDIC Weak Pattern Message fcr Switched Line
IJLBOT16	EBCDIC Weak Pattern Message for Leased Line
IJLBOT17	TRANSCODE Weak Pattern Message for Switched Line
IJLBOT18	TRANSCODE Weak Pattern Message fcr Leased Line
IJLBOT19	EBCDIC Weak Pattern for OLE SYN Insertion
IJLBOT20	EBCDIC 80-Character Transparent Message
IJLBOT21	EBCDIC 120-Character Transparent Message
IJLBOT22	EBCDIC 144-Character Transparent Message
IJLBOT23	3270 Basic EBCDIC Test Pattern
IJLBOT24	3270 Model 1 Align EBCDIC Test Pattern
IJLBOT25	3270 Model 2 Align EBCDIC Test Pattern
IJLBOT26	3270 Orders EBCDIC Test Pattern
IJLBOT27	3270 Universal Character Set Test Pattern
IJLBOT28	3270 NL/EOM Printer EBCDIC Test Pattern
IJLBOT29	3270 Basic ASCII Test Pattern
IJLBOT30	3270 Model 1 Align ASCII Test Pattern
IJLBOT31	3270 Model 2 Align ASCII Test Pattern
IJLBOT32	3270 Orders ASCII Test Pattern
IJLBOT33	3270 Universal Character Set Test Pattern
IJLBOT34	3270 NL/EOM Printer ASCII Test Pattern
IJLT2ALC	All Character Test for IBM 2848
IJLT2ROT	Rotate Test for IBM 2848
IJLT2TLT	Tilt Test for IBM 2848
IJLT2TWS	Twist Test for IBM 2848
IJLT3ALC	All Character Test for IBM 1030

IJLT3ROT	Rotate Test for IBM 1030
IJLT3SLA	Analyzer Test for IBM 1030 (SELECTRIC®)
IJLT3TLT	Tilt Test for IBM 1030
IJLT3TWS	Twist Test for IBM 1030
IJLT5ALC	All Character Test for IBM 1050 or 2740
IJLT5ROT	Rotate Test for IBM 1050 or 2740
IJLT5SLA	SELECTRIC Analyzer Test for IBM 1050 or 2740
IJLT5TLT	Tilt Test for IBM 1050 or 2740
IJLT5TWS	Twist Test for IBM 1050 or 2740
IJLT6ALC	All Character Test for IBM 1060
IJLT6ROT	Rotate Test for IBM 1060
IJLT6SLA	SELECTRIC Analyzer Test for IBM 1060
IJLT6TLT	Tilt Test for IBM 1060
IJLT6TWS	Twist Test for IBM 1060

BTAM Transients

$$ABERP1	Control handler for BTAM special message writer and error statistics recorder
$$ABERP2	BSC online test message writer
$$ABERP3	Console special message writer
$$ABERP4	Remote 3270 status/sense error recorder
$$ABERP5	Error recovery message writer, phase 1
$$ABERP6	Error recovery message writer, phase 2
$$ABERP7	Error recovery message writer, phase 3
$$ABERP8	RMSR record builder
$$ABERP9	Free areas from real storage
$$BCTC01	CLOSE routine
$$BETPRT	Error threshold message
$$BHDRCK	Terminal test request validation and comparison
$$BLEPRT	Line error print routine
$$BLOPEN	OPEN routine
$$BOTC01	OPEN routine
$$BRESPL	Reset Polling Lines
$$BOTC02	OPEN for local devices routine
$$BTCNCL	Cancel routine
$$BTMEBG	Terminal test request - IBM 1030 Manual Entry and Badge Reader
$$BT1030	Terminal test module - IBM 1030
$$BT1050	Terminal test module - IBM 1050
$$BT1060	Terminal test module - IBM 1060
$$BT2260	Terminal test module - IBM 2260
$$BT2740	Terminal test module - IBM 2740
$$BT2848	Terminal test module - IBM 2848
$$BBT3SC	OLTEP device assignment completion
$$BBT3SCI	OLTEP device assignment initiation

RELOCATABLE LIBRARY

Modules	Blocks
65	161

Modules

Module Name	Description	Bytes Of Main Storage
CONCURRENT OLTEP		
BT3270SC	Concurrent OLTEP Set-up	2500
CHANNEL PROGRAM		
(For start-stop, the modules include a table of special characters.)		
IJL0EZ	WTTA	124
IJL00Y	IBM 7770	76
IJL01J	IBM 1030 (Auto Poll)	187
IJL01Z	IBM 1030	139
IJL02J	IBM 1060 (Auto Poll)	175
IJL02Z	IBM 1060	127
IJL03Z	IBM 2848 Remote	209
IJL04Z	AT&T 83B3	78
IJL05Z	Western Union 115A	72
IJL06Z	Table Generation for 2260 Lccal Channel Program	16
IJL07J	IBM 1050 Nonswitched (Auto Poll)	132
IJL07Y	IBM 1050 Switched	202
IJL07Z	IBM 1050 Nonswitched	124
IJL08H	IBM 2740 with Station Control (Auto Poll)	97
IJL08M	IBM 2740 with Dial, Transmit Control, and Checking	176
IJL08P	IBM 2740 with Station Ccntrcl and Checking	132
IJL08Q	IBM 2740 with Dial and Checking	172
IJL08R	IBM 2740 with Checking	141
IJL08U	IBM 2740 with Dial and Transmit Control	122
IJL08X	IBM 2740 with Station Control	92
IJL08Y	IBM 2740 with Dial	114
IJL08Z	IBM 2740 Basic	93
IJL081	IBM 2740 and OIU with Dial and Checking	246
IJL082	IBM 2740 and OIU with Checking	181
IJL089	IBM 2740 with Station Control and Checking (Auto Poll)	145
IJL09Y	TWX 33	110
IJL10Y	IBM 7772	76
WTTA SUBROUTINE		
IJLWTZ	WTTA Subroutines	1223
LOCAL 3270 TABLE OF OFFSETS		
IJL0HZ	3277L table of offsets with command codes	30

CHANNEL PROGRAM MODULES FOR BSC

The following relocatable modules are BSC channel program modules for switched lines using ID verification (CPU-to-CPU). The correspondence between relocatable module name and the codes that appear in the FEATURE operand sublist in the DTFBT macro instruction is indicated.

IJL0BY	For switched lines	415
IJL0BZ	For nonswitched lines	281
IJL0DY	IBM 2780 switched point to point	212
IJL0DZ	IBM 2780 nonswitched point to point	148
IJL1BZ	BSC Multipoint Lines	253
IJL2DZ	IBM 2780 Multipoint for TRANSCODE	136
IJLEDIT3	Logic, TP EDIT Macro	2090

BSC TABLES OF SPECIAL CHARACTERS

IJLASC	In ASCII for BSC	85
IJLEBD	In EBCDIC for BSC	85
IJLTCD	In 6-bit transcode for BSC	69

TRANSLATION TABLES

IJLRASA	For USASCII to EBCDIC for BCS	256
IJLRCTW	For ITA2 to EBCDIC (World Trade Teletype)	256
IJLRCT1	From BAUDOT to EBCDIC	256
IJLRCT2	From TWX TO EBCDIC	256
IJLRCT3	From ZSC3 to EBCDIC (World Trade Teletype)	256
IJLRC30	From 1080 to EBCDIC	256
IJLRC40	From 2740 to EBCDIC (lowercase)	256
IJLRC50	From 1050 to EBCDIC (lowercase)	256
IJLRC60	From 1060 to EBCDIC	256
IJLRC80	From 6-Bit TRANSCODE to EBCDIC	256
IJLRF40	From 2740 to EBCDIC (uppercase)	256
IJLRF50	From 1050 to EBCDIC (uppercase)	256
IJLRSCI	From USASCII to EBCDIC with 2848 Attached to 2701 via IBM Terminal Control Type III	256
IJLSASA	From EBCDIC to USASCII for BSC	256
IJLSCTW	From EBCDIC to ITA2 (World Trade Teletype)	256
IJLSCT1	From EBCDIC to BAUDOT	256
IJLSCT2	From EBCDIC to TWX	256
IJLSCT3	From EBCDIC to ZSC3 (World Trade Teletype)	256
IJLSD30	From EBCDIC to 1030	256

IJLSD40	From EBCDIC to 2740	256
IJLSD50	From EBCDIC to 1050	256
IJLSD60	From EBCDIC to 1060	256
IJLSD80	From EBCDIC to 6-bit TRANSCODE	256
IJLSSCI	From EBCDIC to USASCII with 2848 Attached to 2701 via IBM Terminal Control Type III	256

Delete Statements

To delete the BTAM phases from the core image library, the BTAM modules from the relocatable library, and the BTAM books from the source statement library, you must supply the DELET card for the appropriate library and the name of the phase, module, or book to be deleted. For example:

```
// JOB DELETC
// EXEC MAINT
   DELETC $$BCT01,$$BETPRT,etc.
   DELETC IJLT2ALC,IJLT2ROT,etc.
   DELETR IJL00Y,IJL01Z,IJL02Z,etc.
   DELETS A.CONTROL,A.LERB,etc.
/&
```

To delete both BTAM and QTAM from the core image, relocatable and source statement libraries, the DELET card for the appropriate library and the following entries must be made: For the core image transients, you must supply the DELETC card with a separate entry for each transient (QTAM and BTAM) to be deleted. For the core image phase names, enter the delete card as follows:

```
   DELETC $$BCTC01,$$BCOOO3,$$BETPRT,etc.
   DELETC IJLB.ALL
   DELETC IJLT.ALL
```

To delete both BTAM and QTAM from the relocatable library enter the delete card as follows:

```
   DELETR IJL.ALL
```

To delete the BTAM and QTAM books from the source statement library, enter the DELETS card with a separate entry for each book to be deleted. The following example shows the required delete cards and type of entries to be made in those cards:

```
// JOB DELETE
// EXEC MAINT
   DELETS A.CONTROL,A.CANCEIM,etc.
/&
```

SOURCE STATEMENT LIBRARY

Macro	Blocks
46	7452

Macros

	Main Storage Bytes	
	Basic	For Variables
A.AS		
A.ASCTR		
A.ASLIST		
A.ASMTRTAB	256n (Note 1)	
A.BTBONLI	(inner macro in BTMOD)	
A.BTBTEIH	(inner macro in BTMCD)	
A.BTBTROD	(inner macro in BTMOD)	
A.BTCKID	(inner macro in BTMCD)	
A.BTCSE	(inner macro in BTMOD)	
A.BTCTUPT	(inner macro in BTMCD)	
A.BTMOD		
A.BTNCKID	(inner macro in BTMOD)	
A.BTOLTIH	(inner macro in BTMCD)	
A.BTONLAP	(inner macro in BTMOD)	
A.BTONLOA	(inner macro in BTMCD)	
A.BTRD		
A.BTRLBF	(inner macro in BTMOD)	
A.BTRQBF	(inner macro in BTMCD)	
A.BTTRANS	(inner macro in BTMOD)	
A.BTRWC		
A.BTWAIT		
A.CHGNTRY	20 (Start-stop)	+6 (Note 2)
	72 (BCS)	
	20 (Start-stop Auto Poll)	+72 (Note 2)
	22 (2260 Local)	+4 (Note 2)
A.CONFIGUR		
A.CONTROL	20	+38 (Note 2)
A.CTRGROUP		
A.CTRLIST		
A.CTRSCHED		
A.DEULIST		
A.DISPGUID		
A.DFTRMLST		
A.DTFBT		
A.DTFBTND		
A.GDUAS		
A.GDULIST		
A.GDUTRANS		
A.IJLBTMDS		
A.LERB	20	
A.LERPRT	14	+8 (Note 2)
A.LOPEN	8	+2 (Note 2)
A.ONLTST		
A.PARAMNUM		
A.PARMLIST		
A.RELBUF	14	+4 (Note 3)
A.REQBUF	12	+8 (Note 4)
A.RESETPL	12	+4 (Note 2)
A.RMSRTAB		
A.SCANREQ		
A.SDRTAB		

A.STEND		
A.TGROUP		
A.TPEDIT		
A.TRLIST		
A.TRANSLAT		
A.TRNSLATE	26	+16 (Note 2)
A.TRSRCTW	256	
A.TRSRCT3	256	
A.TRSSCTW	256	
A.TRSSCT3	256	
A.TWAIT	24	+20 (Note 4)

Notes:

1. n=number of different operands coded.

2. No operands coded in register notation.

3. First operand not coded in register notation.

4. All possible operands not coded in register notation.

Storage Requirements

DTFBT (DEFINE THE FILE: BTAM)

TABLE REQUIREMENTS

Size = 64 + N (40 + 8x) + BUFCB + BUFNO (BUFL + f) + y + z
(+ 32N for BSC only)

Where:

- N = number of lines in the line group (or, for a local 2260 and local 3270**, the number of devices attached to the control unit)
- x = number of CCWs in the largest channel program available for the device, given in Figure 28-1
- *BUFCB = 8 if a buffer pool is used
  = 0 if a buffer is not used
- *BUFNO = number of buffers in the pool
- *BUFL = length of each buffer
- f = number of bytes required to extend each buffer to a multiple of 8
- y = size of the model channel program table for the line group, given in Figure 28-1. If two or more DTFBTs use the same model channel program and are linkage edited together, include the value only once.
- Z = Size of the table of special characters given in Figure 28-1. For BSC only, if two or more DTFBTs use the same transmission code, include the value only once.

* If the buffer pool is shared by two or more DTFBTs, include the value only once.

**For a local 3270, at least one DTFBT is required for each control unit.

Device	m
1030	1
1050NS	2
1050S	2
1060	2
2260L	Not applicable
2260R	2
2740	Not applicable
2740C	Not applicable
2740D	Not applicable
2740DC	Not applicable
2740DT	Not applicable
2740DTC	Not applicable
2740S	1
2740SC	1
2740DC0	Not applicable
2740C0	Not applicable
3277R	5
115A	2
83B3	2
1130	2
2780	3
WTTA	Not applicable

Figure 28-2. BTAM--Number of Polling/Addressing Characters

Table Requirements for DIALST
Size = n(m + 1) + 1 + q + p + i

where:
n = number of list entries
m = values specified in Figure 28-2
q = 2 if the list includes entries for polling or addressing
= 0 otherwise
p = number of dial digits
i = 3 if the inlist operand is coded (ESC cnly)
= 0 otherwise

Table Requirements for IDLST
Size of TWX calling list = d + 5 + 2b
Size of TWX answering list = 5 + b
Size of BSC IDLST only = 3 + d + 2r + s + i

where:
d = number of dial digits
b = number of TWX ID characters
= 0 otherwise
r = number of ID characters expected tc be received (BSC)
s = number of ID characters to be sent (BSC)
i = 3 if the inlist operand is coded (ESC)
= 0 otherwise

Table Requirements for SSLAST/SSAWLST
Size = n(m + 1) + 6

where: n = number of list entries
m = values specified in Figure 28-2

Table Requirements for AUTOLST/AUTOWLST
Size = n(m + 2) + 8

where: n = number of list entries
m = values specified in Figure 28-2

Table Requirements for WTTALST

1. When the WRU feature is present in DTFBT:

   Size = 3 + 2r + s

2. When the IAM feature is present, and WRU is not present in DTFBT:

   Size = 2 + s

where: r = number of ID characters expected to be received
s = number of ID characters to be sent

BTMOD (BTAM Logic Module)

The size of the BTAM module varies with the options selected as shown in Figure 28-3. When all operands are omitted or the standard (default) options are coded, the resulting basic module requires 5515 bytes of main storage.

BTAM DATA EVENT CONTROL BLOCK TABLE REQUIREMENTS

Size = 40 bytes

When the MF operand of a READ or WRITE macro is coded MF=L, or when the MF operand is omitted, a Data Event Control Block (DECB) is reserved. One DECB should be reserved per line.

SCANREQ DISPLAY ALPHAMERIC CONTROL AREA (DACA) REQUIREMENTS

Size of each DACA = 24 bytes

When the MF operand of a SCANREQ macro is coded MF=L, or when the MF operand is omitted, a Display Alphameric Control Area (DACA) is reserved. One DACA should be reserved per device.

Operand	Option	Number of bytes added to basic module
ERLOGIC= *	N	-2090
	C	+400
	NC	-1800
SWITCH=	YES	+1700 (IF BUFFER=NO, BSCTEST=NO)
		+1750 (IF BUFFER=YES, BSCTEST=NO)
		+1800 (IF BUFFER=YES, BSCTEST=YES
	NEWID	+2900 (IF BUFFER=NO, BSCTEST=NO
		+2950 (IF BUFFER=YES, BSCTEST=NO)
		+3000 (IF BUFFER=YES, BSCTEST=YES)
	= YES	+ 290
AUDIO=	YES	+ 410 (SWITCH must equal YES)
BUFFER=	YES	+1950 (If L2260=NO and L3277=NO)
		+2075 (IF L2260=YES OR L3277=YES)
	REQREL	+ 710 (If L2260=NO and L3277=NO)
		+ 825 (If L2260=YES or L3277=YES)
TERMTST=	YES	+1044
L2260= L3277=	YES	- 980 (If ERLOGIC=N)
		+1440 (If ERLOGIC=E)
	NO	+1750 (If ERLOGIC=C)
		- 730 (If ERLOGIC=NC)

*When estimating the size of BTAM modules, use the ERLOGIC storage estimate only if L2260 and/or L3270 and/or SSAPL are not specified.

Figure 28-3. BTAM--Main Storage Requirements (Part 1 of 2)

Operand	Option	Number of bytes added tc basic mcdule
L3277=	YES	- 15 (If ERLOGIC=N)
L2260=	NC	+2680 (If ERLOGIC=E)
		+3000 (If ERLOGIC=C)
		- 270 (If ERLOGIC=NC)
L2260=	YES	+ 770 (If ERLOGIC=N)
L3277=	YES	+3725 (If ERLOGIC=E)
		+4050 (If ERLOGIC=C)
		+1160 (If ERLOGIC=NC)
TST3277=	YES	+1160
TRANSL=	YES	+ 200
BSCS=	YES	+11000 (If SWITCH=YES and BUFFER=YES)
		+5900 (If SWITCH=YES and BUFFER=NO)
		+6235 (If SWITCH=NO and BUFFER=NC)
		+9240 (If SWITCH=NO and BUFFER=YES)
		+9125 (If SWITCH=NEWID and BUFFER=NO)
		+12180 (If SWITCH=NEWID and BUFFER=YES)
SSAPL=	YES	-1170 (If ERLOGIC=N)
		+1520 (If ERLOGIC=E)
		- 910 (If ERLOGIC=NC)
		+1830 (If ERLOGIC=C)
BSCMPT=	YES	+1010 (BSCS must be specified YES)
WITA=	YES	+ 320
BSCTEST=	YES	+3550 (If BUFFER=NO)
		+3700 (If BUFFER=YES)
DECBEXT=	YES	- 100 (BSCS must be specified YES)
RMSR=	YES	+568 (If L2260=NO and L3277=NO)
		+694 (If L2260=YES and L3277=NO)
		+726 (If L2260=NO and L3277=YES)
		+852 (If L2260=YES and L3277=YES)
RESETPL	NO	- 120

Figure 28-3. BTAM--Main Storage Requirements (Part 2 of 2)

Module 29: QTAM--370N-CQ-470

Modular Outline

Modular Index

CORE IMAGE LIBRARY

	Blocks		
Transients	2311	2314/2319	3330
22	22	22	22

Transients

$$BCQC01	Close Phase 1
$$BCQC02	Close Phase 2
$$BCQC03	Close Phase 3
$$BOQO01	Open Monitor/DASD Message Queues, Phase 1
$$BOQO02	Open Line Group and Main Storage Process/Destination Queues
$$BOQO03	Open Checkpoint/Restart, Phase 1
$$BOQO04	Open Checkpoint/Restart, Phase 2
$$BOQO06	Open Main Storage Process/Destination Queues
$$BOQO07	Open IBM 7772 Vocabulary DASD File
$$BOQO08	Open Audio Line Group and Output Queue
$$BQCNCL	Cancel
$$BQCNCM	Cancel
$$BQHDCK	Terminal Test Header Analysis
$$BQWTRA	ARU Message Writer
$$BQWTR1	Message Writer, Phase 1
$$BQWTR2	Message Writer, Phase 2
$$BQWTR3	Message Writer, Phase 3
$$BQ1030	Terminal Test Module for IBM 1030
$$BQ1050	Terminal Test Module for IBM 1050
$$BQ1060	Terminal Test Module for IBM 1060
$$BQ2260	Terminal Test Module for IBM 2260
$$BQ2740	Terminal Test Module for IBM 2740

RELOCATABLE LIBRARY

Modules	Blocks
109	392

Modules

BASIC QTAM LOGIC MODULES: Basic QTAM Logic modules are identified by one asterisk (*) preceding the listed module.

QTAM DEVICE I/O MODULES: QTAM device I/O modules are identified by two asterisks (**) preceding the listed module. Include the storage requirements for a particular device I/O module only once, and only if the terminal type is present in the system.

Module Name	Description	Bytes of Main Storage
IJLQAA	Audio Line Appendage	2386
Note:	Required if the QTAM audio support is selected.	
*IJLQAD	IBM 7772 Vocabulary Disk Appendage	1604
Note:	Required if the IBM 7772 Vocabulary File is used.	
IJLQBO	Breakoff (BREAKOFF)	212
*IJLQCK	Checkpoint	1208
Note:	Required if the Checkpoint/Restart Facility is selected in the DTFQT.	
IJLQCL	Change Line (STARTLN AND STOPLN)	557
IJLQCM	Cancel Message (CNACELM)	160
IJLQCP	Change Polling List Entry (CHNGP)	146
IJLQCR	Checkpoint Request (CKREQ)	98
IJLQCT	Change Terminal Table Entry (CHNGT)	250
*IJLQDA	Disk I/O Module	1372
Note:	Not required when the system contains only audio devices.	
IJLQDC	Copy Counters (COPYC)	280
IJLQDE	Copy Terminal Table Entry (COPYT)	120
IJLQDL	Distribution List	156
IJLQDP	Copy Polling List Entry (COPYP)	104
IJLQDQ	Copy Queue Control Block Status (COPYQ)	98
IJLQDT	Insert Date in Message Header (DATESTMP)	60
IJLQEA	End-of-Address (EOA)	100
IJLQEB	End-of-Block (EOB)	132
IJLQEC	End-of-Block and Line Correction (EOBLC)	396
IJLQER	Error Message (ERRMSG)	292
IJLQEX	Expand Message header	70
IJLQFL	DTF Locator	78
IJLQGA	Get Audio Message	324
IJLQGB	Get Audio or Nonaudio Message	732

Module Name	Description	Bytes of Main Storage
IJLQGC	Get audio message or nonaudio message record (GET)	760
IJLQGD	Get audio message or nonaudio message segment (GET)	714
IJLQGM	Get complete message (GET)	472
IJLQGR	Get message record (GET)	500
IJLQGS	Get message segment (GET)	454
*IJLQIP	QTAM implementation	3692
IJLQIT	Intercept message (INTERCPT, permanent queuing)	140
*IJLQLA	Line appendage and ERP module PCI and program	4152
Note:	Not required when the system contains only audio devices.	
*IJLQLC	Line appendage for PCI and program check module	630
Note:	Not required when the system contains only audio devices.	
IJLQLG	Audio input message logging (LOGSEG)	304
IJLQLK	Lookup terminal table entry (DIRECT)	104
*IJLQLO	IBM 2260 local appendage	1288
Note:	Required if the IBM 2260 local device is used.	
IJLQMC	Conversational mode (MODE)	384
IJLQMI	Initiate mode (MODE)	18
IJLQMM	Message-mode interface (MODE)	44
IJLQMP	Priority mode (MODE)	36
IJLQMT	Compare Message Type (MSGTYPE)	60
*IJLQMW	Error recovery procedures message writer subtask	1359
**IJLQM0	Model channel program for IBM 1030 terminals	194
**IJLQM1	Model channel program for IBM 1060 terminals	168
**IJLQM2	Model channel program for IBM 2260 terminals	206
**IJLQM3	Model channel program for AT&T 83B3 terminals	102
**IJLQM4	Model channel program for Western Union Plan 115A terminals	91

Module Name	Description	Bytes of Main Storage
**IJLQM5	Model channel program for IBM 1050 switched and nonswitched terminals	207
**IJLQM6	Model channel program for IBM 1050 nonswitched terminals	194
**IJLQM8	Model channel program for AT&T TWX terminals (Models 33 and 35)	113
**IJLQM9	Model channel program for IBM 2260 local terminals	75
**IJLQN0	Model channel program for IBM 2740 Basic terminals	99
**IJLQN1	Model channel program for IBM 2740 Basic Dial terminals	166
**IJLQN2	Model channel program for IBM 2740 terminals with station control	172
**IJLQN3	Model channel program for IBM 2740 terminals with station control checking	221
**IJLQN4	Model channel program for IBM 2740 dial terminals with transient control and checking	198
**IJLQN5	Model channel program for IBM 2740 terminals with checking	128
**IJLQN6	Model channel program for IBM 2740 dial terminals with checking	192
**IJLQN7	Model channel program for IBM 2740 dial terminals with transmit control	156
**IJLQN8	Model channel program for World Trade Telegraph Terminals (WTTA)	356
IJLQOA	Operator awareness	1380
IJLQOC	Operator Control (OPCTL)	3530
IJLQPA	Put audio message (PUT)	370

Module Name	Description	Bytes of Main Storage
IJLQPL	Polling limit control (POLLIMIT)	120
IJLQPM	Put complete message (PUT)	480
IJLQPR	Put message record (PUT)	544
IJLQPS	Put message segment (PUT)	520
IJLQPZ	Pause-transmit idle characters (PAUSE)	360
IJLQQT	Close message control (CLOSEMC)	474
IJLQRA	Translate table RCVARU: ARU code to EBCDIC	266
IJLQRB	Translation table RCVITA2 (EBCDIC to International Telegraph Alphabet No. 2)	266
IJLQRC	Translation table RCVZ-RCVZC3 (EBCDIC to Figure-Protected Code ZSC3)	266
IJLQRD	Retrieve message segment by DASD address (RETRIEVE)	124
IJLQRG	Route message (ROUTE)	48
IJLQRM	Release message (RELEASEM)	234
IJLQRR	Reroute message (REROUTE)	72
IJLQRS	Retrieve message header by sequence number (RETRIEVE)	380
*IJLQRW	Physical input/output control	1569
Note:	Not required when the system contains only audio devices.	
IJLQR1	Translate table RCV1030: 1030 to EBCDIC	266
IJLQR2	Translate table RCV1050: 1050 to EBCDIC	266
IJLQR3	Translate table RCV1050F: 1050 to monocase EBCDIC	266

Module Name	Description	Bytes of Main Storage
IJLQR4	Translate table RCV1060: 1060 to EBCDIC	266
IJLQR5	Translate table RCV2260: 2260 to EBCDIC	266
IJLQR6	Translate tables RCV83B3 or RCV115A: AT&T 83B3 or WU Plan 115A to EBCDIC	266
IJLQR7	Translate table RCVTWX: AT&T Models 33/35 (TWX) to EBCDIC	266
IJLQR8	Translate table RCV2740: 2740 to EBCDIC	266
IJLQR9	Translate table RCV2740F: 2740 to EBCDIC	266
IJLQSB	Translation table SNDITA2 (International Telegraph Alphabet No. 2 to EBCDIC)	266
IJLQSC	Translation table SNDZSC3 (Figure-Protected Code ZSC3 to EBCDIC	266
IJLQSH	Scan message header	104
IJLQSI	Sequence-in number verification (SEQIN)	140
IJLQSK	Skip-through-character (SKIP)	76
IJLQSO	Insert sequence-cut number in message header (SEQOUT)	68
IJLQSR	Source terminal name verification (SOURCE)	128
IJLQSS	Change audio line (STARTARU and STOPARU)	300
IJLQST	Skip-on-count (SKIP)	48
IJLQS1	Translate table SND1030: EBCDIC to 1030	266
IJLQS2	Translate table SND1050: EBCDIC to 1050	266

Module Name	Description	Bytes of Main Storage
IJLQS4	Translate table SND1060: EBCDIC to 1060	266
IJLQS5	Translate table SND2260: EBCDIC to 2260	266
IJLQS6	Translate tables SND83B3 or SND115A EBCDIC to AT&T 83B3 or WU Plan 115A	266
IJLQS7	Translate table SNDTWXE: EBCDIC to AT&T Models 33/35--TWX (even parity)	266
IJLQS8	Translate table SND2740: EBCDIC TO 2740	266
IJLQS9	Translate table SNDTWXO: EBCDIC to AT&T Models 33/35--TWX (nonparity)	266
*IJLQTA	World Trade Telegraph Terminals (WTTA) Line Appendage Note: Required if World Trade Telegraph support is selected.	1406
IJLQTR	Code translation used in conjunction with QTAM or user provided translate table (TRANS). Note: T = number of translation tables.	114+226T
IJLQTS	Insert time-of-day in message header (TIMESTMP)	198
*IJLQTT	Terminal test recognition (LPSTART) Note: Not required when the system contains only audio devices.	1394

Delete Statements

To delete the QTAM phases from the core image library, the QTAM modules from the relocatable library, and the QTAM books from the source statement library, the DELET card for the appropriate library and the name of the phase, module or book to be deleted must be supplied. For example:

```
// JOB DELETE
// EXEC MAINT
   DELETC $$BCQC01,$$BQWTR1,etc.
   DELETR IJLQBO,IJLQCI,IJLQCM,etc.
   DELETS A.BREAKOFF,A.BUFFER,A.CANCELM,etc.
/&
```

To delete both BTAM and QTAM from the core image, relocatable and source statement libraries, the DELET card for the appropriate library and the following entries must be made: For the core image transients, you must supply the DELETC, card with a separate entry for each transient (QTAM and BTAM) to be deleted. For the core image phase names, enter the delete card as follows:

```
DELETC $$BCTC01,$$BCO003,$$BETPRT, etc.
DELETC IJLB.ALL
DELETC IJLT.ALL
```

To delete both BTAM and QTAM from the relocatable library enter the delete cards as follows:

```
DELETR IJL.ALL
```

To delete the BTAM and QTAM books from the source statement library, enter the DELETS card with a separate entry for each book to be deleted. The following example shows the required delete cards and type of entries to be made in those cards.

```
// JOB DELETE
// EXEC MAINT
   DELETS A.CONTROL,A.CANCELM, etc.
/&
```

SOURCE STATEMENT LIBRARY

Macro	Blocks
95	1808

Macros

Basic

A.ARUMGTYP
A.BREAKOFF
A.BUFARU
A.BUFFER
A.CANCELM

A.CHECKARU
A.CHNGP
A.CHNGT
A.CKREQ
A.CLOSEMC

A.COPYC
A.COPYP
A.COPYQ
A.COPYT
A.COUNTER

A.CTLTBL
A.DATESTMP
A.DIRECT
A.DTFQT
A.ENDRCV

A.ENDREADY
A.ENDSEND
A.EOA
A.EOB

A.EOBLC
A.ERRMSG
A.IJLQBABD
A.IJLQBFRD
A.IJLQBRBD

A.IJLQCKPD
A.IJLQCTLD
A.IJLQDEQU
A.IJLQDTFD

A.IJLQLABD
A.IJLQLCBD
A.IJLQMCBD
A.IJLQOBRD

A.IJLQSVCD
A.IJLQTBLD
A.INTERCPT
A.LCBD
A.LINE

A.LINETBL
A.LIST
A.LOGSEG
A.LPSTART
A.MODE
A.MSGTYPE
A.OPCTL

A.OPTION
A.PAUSE
A.POLL
A.POLLIMIT
A.POSTARU

A.POSTRCV
A.POSTSEND
A.PREFIXD
A.PROCESS
A.QCBD

A.RCVHDR
A.RCVITA2
A.RCVSEG
A.RCVZSC3
A.RELEASEM

A.REPEAT
A.REROUTE
A.RETRIEVE
A.ROUTE
A.SENDHDR

A.SENDSEG
A.SEQIN
A.SEQOUT
A.SKIP
A.SNDITA2

A.SNDZSC3
A.SOURCE
A.STARTARU
A.STARTLN
A.STCBD

A.STOPARU
A.STCPLN
A.TERM
A.TERMTBL
A.TERMTBID

A.TIMESTMP
A.TRANS
A.WORD
A.WORDTBL
A.WRU

System Generation

A,IJLQDSCT
A.IJLQIP1D
A.IJLQQCBD
A.IJLQSTBD
A.IJLQTSVC
A.IJLQVECD

QTAM STORAGE REQUIREMENTS

The main storage requirements for QTAM depend to a great extent on the configuration of your teleprocessing installation and the nature of your applications. Storage requirements increase proportionately as the number of communication lines, terminals, and QTAM-provided processing functions increase. These requirements can be estimated from formulas and tables presented in this module.

To determine the main storage requirements for the Basic QTAM Logic modules and the QTAM Device I/O modules, refer to Relocatable Library.

DTFQT (Define the File: QTAM)

Seven types of DTF tables may be generated by a DTFQT macro instruction. The storage estimates for each follow.

DASD Message Queues Files
Size = 315 bytes

Communication Line Group File
For Nonaudio Line Group

Size = 48 + (128 + 8x)N

For Audio Line Group

Size = 64 + P + (161 + L_1 + L_2 + G + x-z)N

where: N = number of lines in the line group
x = a function of the device, given in Figure 29-1
P = 22 for IBM 7772 only
L_1 = length of input buffers
L_2 = length of address chain buffers
G = 9 when using time stamping option, otherwise = 0
z = 17 when information mode is used

Device	x
1030	8
1050NS	8
1050S	10
1060	7
2260 Remote	8
2260 Local	5
274A	4
274B	6
274C	6
274D	7
274E	7
274F	4
274G	6
274H	7
115A	8
83B3	7
TWX33/35	6
7770	0
7772	60
WTTA	8

Figure 29-1. QTAM--Values for Communication Line Group Table

Main Storage Process Queue
Size = 84 + 12x

where: x = 0 for nonmixed application, and
1 for mixed application

Main Storage Destination Queue
Size = 80 bytes

Checkpoint Records File
Size = 220+L

where: L = length of the Checkpoint record specified by the SOWA keyword operand.

IBM 7772 Vocabulary File
Size = 40+4p

where: P = number of BUFARU macro instructions

Audio Output Queue
Size = 76 bytes

CONTROL INFORMATION
The storage estimates for required control information are shown in Figure 29-2.

Control Blocks and Information	Storage Requirements (in bytes)
Terminal table TERMTBL macro instruction	12
OPTION macro instruction	No storage is reserved for this macro; it defines user areas that are included in the expansion of the TERM macro instruction (U parameter)
TERM macro instruction[1]	9 + I + U + D + 44F where: (I + U + D)≤243
LIST macro instruction[1]	12 + L + 2N + 140[2] where: (3 + L +2N)≤243
PROCESS macro instruction[1]	13 + A for audio process program entry 9 + Y for nonaudio process program entry
Polling list POLL macro instruction[3]	4 + 3N for autopolled terminals except IBM 1030 4 + 2N for nonswitched terminals or autopolled IBM 1030 5 for switched IBM terminals 3 + I for TWX 4 + T for WTTA terminals
Queue Control Block for Process Queues and Destination Queues	32X
Audio Line Table LINETBL macro instruction[1]	4
LINE macro instruction[1]	5 + Z
Audio Word Table WORDTBL macro instruction[1]	4
WORD macro instruction[1]	8 + W

Figure 29-2. QTAM--Storage Requirements for Control Information (Part 1 of 2)

Buffers

For Nonaudio Applications:
  Buffer Pool Size = 8+(x+16)N + 24M

For Audio Applications with IBM 7772:

Audio Buffer Pool Size = 24 + (X + 88)N

where: N = number of buffers specified.
X = size of each buffer
M = number of CCWs QTAM generates for data insertion by the PAUSE macro

where: N = number of terminals
I = number of bytes in terminal ID
U = number of bytes in optional area
D = number of bytes in device address area; size depends on contents:

For nonswitched terminal--addressing and polling characters (1 byte/character)
For IBM switched terminal--field telling the number of dial digits (1 byte) + dial digits (1 byte/digit) + addressing characters (1 byte/character)
For TWX--field telling the number of dial characters (1 byte) + dial digits (1 byte/digit) + field telling the number of ID characters (1 byte) + ID characters (2 bytes/character)
For WTTA terminals--1 byte + field telling the number of ID characters (1 byte) + ID characters (2 bytes/character)
For IBM 2260 Local--a field of 6 fullwords consisting of a CCB and other control information

L = number of bytes in name of the distribution list entry in terminal table (1-8)
A = number of bytes in name of the audio process entry in the terminal table. Because the following field in the entry must be aligned to a fullword boundary, this field must be either three, seven or eleven bytes long.
X = number of lines or terminals (depending on queueing techniques) and the number of process queues
Y = number of bytes in name of the process entry in terminal table (1-8)
Z = number of bytes in name of the line entry in line table (1-8)
W = number of bytes of the selected word
F = 1 for IBM 2740 Model 2 terminals with the Buffer Receiver Option; 0 for other terminals
T = number of bytes in the CPU identification (WTTA terminals)

Notes:

1. Add the number of bytes necessary for fullword boundary.

2. This number (140) is the number of bytes in the Distribution List module (IJLQDL). This number is included in the storage requirements only once if the LIST macro is used more than once.

3. Add the number of bytes necessary for halfword boundary.

Figure 29-2. QTAM--Storage Requirements for Control Information (Part 2 of 2)

QTAM Macro Instructions

Figure 29-3 gives the storage requirements for all other QTAM macro instructions.

Much QTAM logic consists of modules introduced by the use of certain QTAM macro instructions in the user's program. These macro instructions expand into in-line coding that establishes the linkage to, and parameters for, the QTAM modules. Often a module so introduced into the system itself introduces another module, a process termed a second level routine.

Column two of Figure 29-3 shows the extent of the coding produced by the expansion of the macro instruction in column one. If the coding links to QTAM module, the module is presented in column three; if that module links to other modules, they are presented in column four.

Storage requirements for a sharable module, or for a second level routine that is linked to more than once in the same partition are included only once. For example, the macro instructions DIRECT, EOA, and ROUTE all link to the same module, IJLQLK. If two or more of these macro instructions are used in the same message control program, the module is included only once. Similarly, if the same macro instruction is used more than once in the same program, storage is required for only the additional linkage because the module is included just once.

Macro Instruction	In-line linkage or code Note 5	Sharable modules		Second level routine	
		Name	Size	Name	Size
ARUMGTYP	16				
BREAKOFF	8	IJLQBC	212		
CANCELM	8	IJLQCM	160		
CHECKARU	70+ message text				
CHNGP	42	IJLQCP	146	IJLQFL	78
CHNGT	26	IJLQCT	250		
CKREQ	10	IJLQCR	98		
CLOSEMC	6	IJLQQT	474	IJIQCL	549
COPYC	24	IJLQDC	504		
COPYP	34	IJLQDP	104	IJIQFL	78
COPYQ	34	IJLQDQ	98		
COPYT	30	IJLQDE	120		
COUNTER	12				
DATESTMP	8	IJLQDT	60	IJIQEX	70
DIRECT	12	IJLQLK	104		
ENDRCV	10	Note 1			
ENDRCV (WTTA)	16	IJLQEB	132		
ENDREADY	80	Note 1			
ENDSEND	10	Note 1			
EOA	28	IJLQEA	100	IJIQSH IJLQSK IJIQRG IJLQMT IJIQIK	104 76 48 60 104 Note 2
EOB	6	IJLQEB	132	Note 1	
EOBLC	6	IJLQEC	380	Note 1	

Figure 29-3. QTAM Storage Requirements for Other Macro Instructions (Part 1 of 5)

Macro Instruction	In-line linkage or code Note 5	Sharable modules		Second level routine	
		Name	Size	Name	Size
ERRMSG	32+ message text	IJIQER Note 1	292	IJLQIK	104
GET					
Segment		IJIQGS	454		
Message		IJLQGM	472		
Record		IJIQGR	500		
GET (AUDIO)					
Audio message		IJIQGA	324		
Audio and non-audio messages		IJLQGB	732		
Audio message & nonaudio record		IJLQGC	760		
Audio message & nonaudio segment		IJIQGD	714		
INTERCPT	12	IJLQIT	140		
LOGSEG	32		Note 3		
LOGSEG (ARU)	26	IJLQLG	304		
LPSTART	28	Note 1			
MODE (C)	14	MODE (U) module & IJLQMM	44	IJLQSH	104
MODE (U)					
INITIATE	10	IJLQMI	18		
PRIORITY	10	IJLQMP	36	IJLQSH	104
CONVERSE	10	IJLQMC Note 1	384		
MOD2260	4				
MSGTYPE (C)	19	IJLQMT	60	IJLQSH	104
MSGTYPE (U)	4				
OPCTL	56	IJLQOC	3610	IJLQIK IJLQSH IJLQCL Note 1	104 104 549

Figure 29-3. QTAM--Storage Requirements for Other Macro Instructions (Part 2 of 5)

Macro Instruction	In-line Linkage or code Note 5	Sharable Modules		Second Level routine	
		Name	Size	Name	Size
PAUSE	13 + insert chars.	IJLQPZ	360	Note 1	
POLIIMIT	12	IJLQPL	120		
POSTARU	6		Note 1		
POSTRCV	6	Note 1			
POSTSEND	12	Note 1			
PUT					
Segment		IJLQPS	520		
Message)		IJLQPM	480		
Record		IJLQPR	544		
PUT (AUDIO)		IJLQPA	370		
RCVHDR	8				
RCVSEG Note 4	0				
RELEASEM	12	IJLQRM	230		
REPEAT	34		Note 1		
REROUTE	26	IJLQRR Note 1	72	IJLQLK	104
RETRIEVE					
DASD address	14	IJLQRD	124		
By sequence number	26	IJLQRS	380	IJLQRD	124
ROUTE	8	IJLQRG	48	IJLQLK IJLQSH	104 104
SENDHDR	16				
SENDSEG Note 4	4				
SEQIN	8	IJLQSI	140	IJLQSH	104
SEQOUT	8	IJLQSO	68	IJLQEX	70
SKIP (CT)	8	IJLQST	48	IJLQSH	104

Figure 29-3. QTAM--Storage Requirements for Other Macro Instructions (Part 3 of 5)

Macro Instruction	In-line linkage or code Note 5	Sharable modules		Second level routine	
		Name	Size	Name	Size
SKIP (S)	8 + no. to be skipped	IJLQSK	76	IJLQSH	104
SOURCE	8	IJLQSR	128	IJLQSH	104
STARTARU	42	IJLQSS	300	IJLQFL	78
STARTIN	12	IJLQCL	512	IJLQFL	78
STOPARU	42	IJLQSS	300	IJLQFL	78
STOPLN	12	IJLQCL	512	IJLQFL	78
TIMESTMP	8	IJLQTS	198	IJLQEX	70
TRANS	10	IJLQTR	114+ 266*T		
WRU	0				

Figure 29-3. QTAM--Storage Requirements for Other Macro Instructions (Part 4 of 5)

where:
C = character operand specified (conditional)
U = character operand null (unconditional)
S = skip to and include designated character configuration
CT = skip designated count of nonblank characters
T = number of translation tables

Translation tables are: RCV1030, RCV1050, RCV1050F, RCV1060, RCV2260, RCV2740, RCV2740F, RCVARU, RCV83B3, RCV115A, RCVTWX, RCVITA2, RCVZSC3, SND1030, SND1050, SND1060, SND2260, SNDITA2, SNDZSC3, SND2740, SND83B3, SND115A, SNDTWXE, and SNDTWXO

Notes:

1. These delimiters or modules cause linkages to QTAM routines included in Basic QTAM Logic Modules listed earlier in this module under Relocatable Library..

2. If the macro instruction MSGTYPE, ROUTE, or SKIP (S) is used in the program, the storage estimate for IJLQMT, IJLQRG, or IJLQSK, respectively, is not included in the storage estimate for EOA.

3. Because the user defines his own DTFxx and xxMOD macros for his message log file, the size requirements cannot be specified here. Information to determine the storage requirements for the specific logging medium is given under the pertinent DTF and Module, i.e., DTFMT and MTMOD for Tape. Modules 4-11 contain storage requirements for declarative macros.

4. Identifies entry point for RCVSEG and SENDSEG subgroups of LPS.

5. The linkage requirements for OPEN, CLOSE, GET, and PUT are given in Module 1.

Figure 29-3. QTAM--Storage Requirements for Other Macro Instructions (Part 5 of 5)

Module 30: MPS Utility Macros--360N-UT-471

Modular Outline

CORE IMAGE LIBRARY

	Blocks	
Transients	2311	2314/2319
3	3	3

Transients

$$BMU100
$$BMU200
$$BMU300

SOURCE STATEMENT LIBRARY

Macro	Blocks
10	1187

Macros

A.INCARD
A.INDISK
A.INLOG
A.INTAPE
A.INTCR

A.OUTCARD
A.OUTDISK
A.OUTLOG
A.OUTPRT
A.OUTAPE

MPS Utility Macro Storage Requirements

Figure 30-1 gives the main storage requirements for the MPS utility macro instructions. These macros are designed so that a file-to-file utility program can fit in a 2K foreground program area (4K if the INTCR macro is used). To determine the size of the macro when expanded, add the variable requirements to, or subtract them from, the basic expansion requirement.

	INTAPE	OUTAPE
Basic Macro Expansion	776	718
BUFSIZ=		
80	+0	+0
n	+2(n-80)	+2(n-80)
RECSIZ=		
n	226	
name	260	
FILE=		
name	-24	140
(r)	-12	120
LBL=		
name	324	+276
ERROR=		
SKIP	+8	
IGNORE	0	
name	+72	
CHKPT=		
NO	-26	
name	+28	
RETURN=		
NO		-6
YES		+0
BLK=		
n		+150
name		+182

Figure 30-1. MPS Utility Macro Storage Requirements (Part 1 of 3)

	INDISK	OUTDISK
Basic Macro Expansion	816	1140
BUFSIZ=		
80	+0	+0
n	+2(n-80)	+2(n-80)
FILE=		
name		+0
(r)	+24	+12
LBL=		
name	+36	+36
RECSIZ=		
n	260	
name	268	
ERROR=		
SKIP	+24	
IGNORE	+0	0
name	+100	148
RETURN=		
NO		0
YES		+0
FORMAT=		
name		+68
FULL		+0
n		44
BLK=		
n		150
name		182

	INLOG	OUTLOG
Basic Macro Expansion	168	170
BUFFER=		
name	+0	+0
(r)	+4	0
COUNT=		
n	+0	0
(r)	-28	+4
RETURN=		
NO		-2
YES		+0

	OUTPRT
Basic Macro Expansion	622
BUFSIZ=	
n	+2(n-144)
RETURN=	
NO	+10
YES	+0
FORMS=	
A	+28
B	+144
C	+114
D	+96

Figure 30-1. MPS Utility Macro Storage Requirements (Part 2 of 3)

	INCARD	OUTCARD
Basic Macro Expansion	418	444
STCTL= YES		+60
name		+112
DEVICE= 2540		+186
2520		+48
RETURN= NO		+4
YES		+0

	INCTR			
	INPUT= MTST,STDUC or MST,STDLC	INPUT= MTST,name1	INPUT= MTST,NOTRAN or MTDI,NOEDIT	INPUT= MTDI,EDIT or MTDI,EDITR
Basic Macro Expansion	1158	1023	570	1720
BUFSIZ= (n)	+3n	+3n	+3n	+3n
(n,m1)	+2n+m1	+2n+m1	+2n+m1	+2n+m1
RECFORM= VAR	+0	+0	+0	+0
UNDEF	-16	-16	-16	-16
ERROPT= IGNORE	+0	+0	+0	+0
name2	+106	+106	+106	+134

Figure 30-1. MPS Utility Macrc Stcrage Requirements (Part 3 of 3)

Multiprogramming Examples

Multiprogramming capabilities are demonstrated by the multiprogramming example. A control program supporting multiprogramming and the MPS utilities is required. A card-to-printer utility macrc lists fifteen test cards ccntaining the instructions and necessary cperator commands to initiate the job in the foreground-one (F1) area. The card-tc-printer utility macrc is first catalcged and then assembled and executed as a background job. The fifteen test cards are listed on the printer. When the last card in the reader has been read and processed, the following message will be printed on SYSLOG:

```
0P08A INTERV REQ SYSRDR ...
```

To clear the intervention required condition on the reader, enter the following two cards in the reader and ready the device:

```
/&
// PAUSE REFER TO PRINTER FOR ADDITIONAL INSTRUCTIONS
```

You can now initiate this same program in the foreground area by following the instructions previously listed on SYSLST when the job was run in the background problem area.

The same data previously used for the background execution (fifteen test cards) is used as input data for the foreground execution. The data cards must, therefore, be removed from the card reader output stacker and they must be replaced in the input hopper. Then the card reader is made ready. Now follow the instructions contained in the test data to initiate the job in the foreground area.

Several assumptions have been made in the multiprogramming examples. The first assumption is that a foreground-one area has previously been allocated. If this has not been done, refer to the DOS Operating Guide, GC24-5022.

The second assumption is that SYSRDR or SYSIPT is assigned to 00C, and that SYSLST is assigned to 00E and that there are no other logical units assigned to these devices. If this is not the case, message

```
1A13D CONFLICTING I/O ASSIGNMENTS
```

is issued on SYSLOG when you try to initiate the program in the foreground area. To recover from this condition, reply CANCEL to the preceding message. The message

```
F1 0S02I JOB FG INIT.  CANCELED DUE TO PROGRAM REQUEST
```

is then issued. Then press the Request key on SYSLOG. The message

```
1I60A READY FOR COMMUNICATIONS
```

is issued. Next, enter the START command followed by

```
ASSGN SYSxxx,UA
```

where xxx is the unit(s) retaining a background assignment for the indicated device. You can now initiate the job in the foreground-one area as previously described. To execute this example, enter the following cards:

```
// JOB CATALOG SAMPLE FOREGRCUND/BACKGROUND PROGRAM
// OPTION CATAL
   PHASE CARDPRNT,+0
// EXEC ASSEMBLY
         Column 10
         PRINT NOGEN
         START 0
         INCARD
         OUTPRT BUFSIZ=80
         END
/*
// EXEC LNKEDT
/&
// JOB EXECUTE SAMPLE FOREGROUND/BACKGROUND PROGRAM
// ASSGN SYS001,X'00C'                    (Ncte 1)
// ASSGN SYS002,X'00E'                    (Note 2)
// PAUSE CARD READER END-OF-FILE SIGNALS END OF CARD INPUT
// EXEC CARDPRNT
* THESE COMMENTS AND THE FCLLOWING STATEMENTS ARE TEST INPUT
* NOW IN A PAUSE STATUS, UNASSIGN THE PRINTER (X'00E') AND THE CARD
* READER (X'00C') TO PERMIT SUBSEQUENT ASSIGNMENT TO FOREGROUND
* AREA.  TO DO THIS, TYPE THE FOLLOWING
ASSGN SYSRDR,UA
ASSGN SYSIPT,UA                           (Note 3)
ASSGN SYSLST,UA
* TO SUSPEND BACKGROUND (BG) AREA PROCESSING TYPE
STOP
* TO INITIATE PROGRAM NAMED CRDPRNT IN FOREGROUND1 (F1) AREA,
* THE FOLLOWING FOUR STATEMENTS ARE TO BE TYPED ON SYSLOG (1052)
START F1                                  (Note 4)
ASSGN SYS001,X'00C'                       (Ncte 1)
ASSGN SYS002,X'00E'                       (Note 2)
// EXEC CARDPRNT
```

Note 1: SYS001 must be assigned to a card reader.

Note 2: SYS002 must be assigned to a printer.

Note 3: These cards are required cnly if SYS001 cr SYS002 retain a background assignment. A device used by a foreground program cannot be assigned to a background area at the same time.

Note 4: This command is processed by the ATTN routine (Refer to DOS Operating Guide, GC24-5022.)

The test cards for the multiprogramming example are output on SYS002, which was previously assigned tc the system printer. LOG must be keyed on SYSLOG at the beginning of the job to obtain a complete listing of all job control cards and messages

SYSLOG output is shown in Figure 30-2.

SYSLST Output Contains:

- Job control cards
- External symbol dictionary
- Source program listing
- Relocation dictionary
- Diagnostics
- Linkage editor storage map
- List of test cards processed by card-to-printer utility program

```
BG // JOB CATALOG SAMPLE FOREGROUND/BACKGROUND PROGRAM
BG EOJ CATALOG

BG // JOB EXECUTE SAMPLE FOREGROUND/BACKGROUND PROGRAM
BG // PAUSE CARD READER END-OF-FILE SIGNALS END OF CARD INPUT
BG
BG 0P08A     INTERV REQ SYSRDR=00C
        CCSW=021000224002000000 SNS=400000000000 CCB=002220
BG EOJ EXECUTE
BG // PAUSE REFER TO PRINTER FOR ADDITIONAL INSTRUCTIONS
BG assgn sysrdr,ua
BG assgn sysipt,ua
BG assgn syslst,ua
BG stop
AR 1160A  READY FOR COMMUNICATIONS.
AR start f1
F1 assgn sys001,x'00c'
F1 assgn sys002,x'00e'
F1 exec cardprnt
F1 0S10I PROGRAM CARDPRNT COMPLETED
```

Figure 30-2. MPS--SYSLOG Cutput for Multiprogramming Examples

Module 31: Utilities-Group 1--360N-UT-461

Modular Outline

Modular Index

The library requirements for the programs of the Group 1 Utilities are the following:

Component	Libraries: Core Image Blocks Phases	2311	2314/ 2319	Relocatable Unique Modules	Blocks
Utilities Group 1 (Note 1):	44	122	125		657
Alternate Track Assignment Disk	5	26	27	6	
Card to Disk	5	10	10	4	
Card to Printer and/or Punch	5	12	12	4	
Clear Disk	3	4	4	2	
Copy Disk to Card	2	7	7	3	
Copy Disk to Disk	2	7	8	3	
Disk to Card	5	11	11	4	
Disk to Disk	5	11	11	2	
Disk to Print (Note 2)	5	12	12	2	
Initialize Disk	4	14	15	5	
Restore Card to Disk	1	4	4	2	
VTOC Display:	1	3	3	2	
Transient	1	1	1		

Note 1: The following are the various combinations of utility modules common to Group 1:

- Common modules in Group 1 are: IJWIVT and IJWLV1, which require 22 blocks.
- Common modules for Groups 1 and 3 are: IJWCID2, IJWCLD3, IJWDD3, IJWDD4, IJWDP3, and IJWDP4, which require 97 blocks.
- Common modules for all 3 groups are: IJWGEN and IJWLAB, which require 19 blocks.
- The block count for all common modules is included in the total for the group.

Note 2: The disk-to-printer utility program has three sample problems in the source statement library that requires three blocks.

PARTITION SIZE

10,240 bytes

Note: To delete all utilities, Group 1, Group 2 and Group 3 use the following statement:

```
DELETR IJW.ALL
```

Alternate Track Assignment Disk

CORE IMAGE LIBRARY

Phases

ATAD
ATAD2
ATAD3
ATAD4
ATAD5

RELOCATABLE LIBRARY

Modules

IJWAD
IJWAD1
IJWAD2
IJWAD3
IJWAD4
IJWAD5

Linkage Edit Statements

```
   INCLUDE IJWAD
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWAD
DELETR IJWAD1
DELETR IJWAD2
```

```
DELETR IJWAD3
DELETR IJWAD4
DELETR IJWAD5
```

Card to Disk

CORE IMAGE LIBRARY

Phases

```
CDDK
CDDK2
CDDK3
CDDK4
CDDK5
```

RELOCATABLE LIBRARY

Modules

```
IJWCD
IJWCD1
IJWCD3
IJWCD4
IJWGEN
IJWLAB
```

Linkage Edit Statements

```
   INCLUDE IJWCD
   PHASE CDDK5,IJWCDCS2,NOAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCD
DELETR IJWCD1
DELETR IJWCD3
DELETR IJWCD4
```

Card to Printer/Punch

CORE IMAGE LIBRARY

Phases

```
CDPP
CDPP2
CDPP3
CDPP4
CDPP5
```

RELOCATABLE LIBRARY

Modules

```
IJWCP
IJWCP1
IJWCP3
IJWCP4
IJWGEN
IJWLAB
```

Linkage Edit Statements

```
   INCLUDE IJWCP
   PHASE CDPP5,IJWCPCS2,NOAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCP
DELETR IJWCP1
DELETR IJWCP3
DELETR IJWCP4
```

Clear Disk

CORE IMAGE LIBRARY

Phases

```
CLRDSK
CLRD2
CLRD3
```

RELOCATABLE LIBRARY

Modules

```
IJWCLD
IJWCLD1
IJWCLD2
IJWCLD3
```

Linkage Edit Statements

```
   INCLUDE IJWCLD
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCLD
DELETR IJWCLD1
```

Copy Disk to Card

CORE IMAGE LIBRARY

Phases

```
CRDC
CRDC2
```

RELOCATABLE LIBRARY

Modules

```
IJWKC
IJWKC1
IJWKC2
```

Linkage Edit Statements

```
   INCLUDE IJWKC
// LBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWKC
DELETR IJWKC1
DELETR IJWKC2
```

Copy Disk to Disk

CORE IMAGE LIBRARY

Phases

```
CRDD
CRDD2
```

RELOCATABLE LIBRARY

Modules

```
IJWRD
IJWRD1
IJWRD2
```

Linkage Edit Statements

```
   INCLUDE IJWRD
// LBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWRD
DELETR IJWRD1
DELETR IJWRD2
```

Disk to Card

CORE IMAGE LIBRARY

Phases

```
DKCD
DKCD2
DKCD3
DKCD4
DKCD5
```

RELOCATABLE LIBRARY

Modules

```
IJWDC
IJWDC1
IJWDC3
IJWDC4
IJWGEN
IJWLAB
```

Linkage Edit Statements

```
   INCLUDE IJWDC
   PHASE DKCD5,IJWDCCS2,NOAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWDC
DELETR IJWDC1
DELETR IJWDC3
DELETR IJWDC4
```

Disk to Disk

CORE_IMAGE_LIBRARY

Phases

DKDK
DKDK2
DKDK3
DKDK4
DKDK5

RELOCATABLE_LIBRARY

Modules

IJWDD
IJWDD1
IJWDD3
IJWDD4
IJWGEN
IJWLAB

Linkage_Edit_Statements

```
   INCLUDE IJWDD
   PHASE DKDK5,IJWDDCS2,NOAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete_Statements

```
DELETR IJWDD
DELETR IJWDD1
```

Disk to Printer

CORE_IMAGE_LIBRARY

Phases

DKPR
DKPR2
DKPR3
DKPR4
DKPR5

RELOCATABLE_LIBRARY

Modules

IJWDP
IJWDP1
IJWDP3
IJWDP4
IJWGEN
IJWLAB

Linkage_Edit_Statements

```
   INCLUDE IJWDP
   PHASE DKPR5,IJWDPCS2,NCAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete_Statements

```
DELETR IJWDP
DELETR IJWDP1
```

SOURCE_STATEMENT_LIBRARY

Sample_Problems

Z.UTDKPR1
Z.UTDKPR2
Z.UTDKPR3

SAMPLE_PROBLEM_SUMMARY

Figure 31-1 defines the file required by the disk-to-printer sample problems. Figure 31-2 illustrates the cards punched when the sample problems are retrieved from the source statement library.

Disk Extent Number	
1	2
SYS004	SYS005

Figure 31-1. DKPR--Sample Problem File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.UTDKPR1 BKEND Z.UTDKPR1 D-P Control Cards (2 Cards) BKEND	$461	0001-0002
CATALS Z.UTDKPR2 BKEND Z.UTDKPR2 D-P Control Cards (2 Cards) BKEND	$461	0101-0102
CATALS Z.UTDKPR3 BKEND Z.UTDKPR3 D-P Control Cards (2 Cards) BKEND	$461	0201-0202

Figure 31-2. DKPR--Sample Problem Card Decks

Disk-to-Printer Utility 1

Program Name is Z.UTDKPR1. The disk-to-printer utility sample problem lists the numbers handled by the Assembler 2 and disk scrt/merge sample problems. To execute the disk-to-printer utility sample problem, you need the following job control cards:

```
              Col. 16                                  Ccl. 54      Ccl. 72
                 ↓                                        ↓            ↓
// JOB DKPR
// PAUSE ASSGN SYS004 TO SCRATCH PACK
// ASSGN SYS004,X'cuu'
// PAUSE ASSGN SYS005 TO THE PRINTER
// ASSGN SYS005,X'cuu'
// UPSI 0000
// DLBL UIN,'SORTED FILE OF 2000 RECORDS FOR UTILITIES'[,yy/ddd]
// EXTENT SYS004,hhhhhh,1,0,1800,43
// EXEC DKPR
   D-P Control Cards (2 cards)
/*
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number of the disk pack.

SYSLOG Output Is:

```
// JOB DKPR
// PAUSE ASSGN SYS004 TO SCRATCH PACK
// PAUSE ASSGN SYS005 TO THE PRINTER
EOJ DKPR
```

SYSLST Output Summary:

- Job control cards
- Disk-to-printer control cards
- Utility parameter statements (10 lines)
- Numbers 0001-2000 (400 lines)
- NUMBER OF INPUT BLCCKS PROCESSED 000400
- NUMBER OF OUTPUT BLOCKS PROCESSED 000400
- END OF JCB

Disk-to-Printer Utility 2

Program Name is Z.UTDKPR2. The disk-tc-printer utility sample problem lists the numbers handled by the Assembler 5 and tape and disk sort/merge (2311 application) sample problems. Tc execute the disk-to-printer utility sample problem, you need the following job control cards are needed:

```
              Col. 16                                   Col. 54      Col. 72
                 ↓                                         ↓            ↓
// JOB DKPR 2311 TO PRINTER
// PAUSE ASSGN SYS004 TO SCRATCH PACK
// ASSGN SYS004,X'cuu'
// PAUSE ASSGN SYS005 TO THE PRINTER
// ASSGN SYS005,X'cuu'
// UPSI 000
// DLBL UIN,'SORTED FILE OF 2000 RECORDS FOR UTILITIES'[,yy/ddd]
// EXTENT SYS004,hhhhhh,1,0,1710,33
// EXEC DKPR
   Disk-to-printer utility control cards (2 cards)
/*
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number of the disk pack.

SYSLOG Output Is:

```
// JOB DKPR 2311 TO PRINTER
// PAUSE ASSGN ON SYS004 TO SCRATCH PACK
// PAUSE ASSGN SYS005 TO THE PRINTER
EOJ DKPR
```

SYSLST Output Summary:

- Job control cards
- Utility parameter statements (10 lines)
- Numbers 0001-2000 (400 lines)
- NUMBER OF INPUT BLOCKS PROCESSED 000400
- NUMBER OF OUTPUT BLOCKS PROCESSED 000400
- END OF JOB

Disk-to-Printer Utility 3

Program Name is Z.UTDKPR3. The disk-tc-printer utility sample problem lists the numbers handled by the Assembler 6 and tape and disk sort/merge (2314/2319 application) sample problems. Tc execute the disk-to-printer utility sample problem, you need the following job contrcl cards::

```
              Col. 16                                  Col. 54      Col. 72
              ↓                                        ↓            ↓
// JOB DKPR 2314/2319 TO PRINTER
// PAUSE ASSGN SYS004 TO SCRATCH PACK
// ASSGN SYS004,X'cuu'
// PAUSE ASSGN SYS005 TO THE PRINTER
// ASSGN SYS005,X'cuu'
// UPSI 000
// DLBL UIN,'SORTED FILE OF 2000 RECORDS FOR UTILITIES'[,yy/ddd]
// EXTENT SYS004,hhhhhh,1,0,3420,63
// EXEC DKPR
              Disk-to-printer utility contrcl cards (2 cards)
/*
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number cf the disk pack.

SYSLOG Output Is:

```
// JOB DKPR 2314/2319 TO PRINTER
// PAUSE ASSGN SYS004 TO SCRATCH PACK
// PAUSE ASSGN SYS005 TO THE PRINTER
EOJ DKPR
```

SYSLST Output Summary:

- Job control cards
- Disk-to-printer control cards
- Utility parameter statements (10 lines)
- Numbers 0001-3000 (600 lines)
- NUMBER OF INPUT BLCCKS PROCESSED 000600
- NUMBER OF OUTPUT BLOCKS PROCESSED 000600
- END OF JOB

Initialize Disk

CORE IMAGE LIBRARY

Phases

```
INTD
INTD2
INTD3
INTD4
```

RELOCATABLE LIBRARY

Modules

```
IJWID
IJWID1
IJWID2
IJWID3
IJWID4
```

Linkage Edit Statements

```
   INCLUDE IJWID
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWID
DELETR IJWID1
DELETR IJWID2
DELETR IJWID3
DELETR IJWID4
```

Restore Card to Disk

CORE IMAGE LIBRARY

Phases

```
CRCD
```

RELOCATABLE LIBRARY

Modules

```
IJWRC
IJWRC1
```

Linkage Edit Statements

```
   INCLUDE IJWRC
// LBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWRC
DELETR IJWRC1
```

VTOC Display

CORE IMAGE LIBRARY

Phases

```
LVTOC
```

Transient

```
$$BLISTV List VTOC
```

Note: $$BLISTV is cataloged into the core image library along with the VTOC program.

RELOCATABLE LIBRARY

Modules (BJS)

```
IJWLVB
IJWLVT
IJWLV1
```

Linkage Edit Statements, Non-multiprogramming System

```
   INCLUDE IJWLVB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements, Non-multiprogramming System

```
DELETR IJWLVB
```

Modules (MPS)

```
IJWIVM
IJWLVT
IJWIV1
```

Delete Statements, Multiprogramming System

```
DELETR IJWLVM
```

Linkage Edit Statements, Multiprogramming System

```
   INCLUDE IJWLVM
// EXEC LNKEDT
```

Module 32: Utilities-Group 2--360N-UT-462

Modular Outline

The library requirements for the programs of the Group 2 Utilities are the following:

Component	Libraries: Core Image Blocks Phases	2311	2314/ 2319	Relocatable Unique Modules	Blocks
Utilities Group 2 (Note 1):	47	103	104		472
Card to Tape	5	10	10	4	
Copy Disk or Data Cell to Tape	2	7	7	3	
Data Cell to Tape	5	10	10	2	
Disk to Tape	5	10	10	2	
Initialize Tape	1	3	3	1	
Restore Tape to Disk or Data Cell	1	4	4	2	
Tape Compare	3	5	5	5	
Tape to Card	5	11	11	4	
Tape to Data Cell	5	11	11	2	
Tape to Disk	5	11	11	2	
Tape to Printer (Note 2)	5	11	12	4	
Tape to Tape	5	10	10	4	

Note 1: Following are the various combinations of utility modules common to Group 2:

- Common modules within Group 2 are: IJWDT3, IJWDT4, IJWTD3, and IJWTD4, which require 67 blocks.
- Common modules for all 3 groups are: IJWGEN and IJWIAB, which require 19 blocks.
- The block count for common modules within Group 2 is included in the total for the group.

Note 2: The tape-to-printer utility program has two sample problems in the source statement library that requires two blocks.

PARTITION SIZE

10,240 bytes

Note: To delete all utilities, Group 1, Group 2 and Group 3 use the following statement:

```
DELETR IJW.ALL
```

Card to Tape

CORE IMAGE LIBRARY

Phases

CDTP
CDTP2
CDTP3
CDTP4
CDTP5

RELOCATABLE LIBRARY

Modules

IJWCT
IJWCT1
IJWCT3
IJWCT4
IJWGEN
IJWLAB

Linkage Edit Statements

```
   INCLUDE IJWCT
   PHASE CDTP5,IJWCTCS2,NCAUTO
   INCLUDE IJWLAB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCT
DELETR IJWCT1
DELETR IJWCT3
DELETR IJWCT4
```

Copy Disk or Data Cell to Tape

CORE IMAGE LIBRARY

Phases

```
CRDT
CRDT2
```

RELOCATABLE LIBRARY

Modules

```
IJWKT
IJWKT1
IJWKT2
```

Linkage Edit Statements

```
   INCLUDE IJWKT
// LBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWKT
DELETR IJWKT1
DELETR IJWKT2
```

Data Cell to Tape

CORE IMAGE LIBRARY

Phases

```
DCTP
DCTP2
DCTP3
DCTP4
DCTP5
```

RELOCATABLE LIBRARY

Modules

```
IJWDT3
IJWDT4
IJWGEN
IJWLAB
IJWMT
IJWMT1
```

Linkage Edit Statements

```
   INCLUDE IJWMT
   PHASE DCTP5,IJWMTCS2,NOAUTO
   INCLUDE IJWLAB
// IBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
   DELETR IJWMT
   DELETR IJWMT1
```

Disk to Tape

CORE IMAGE LIBRARY

Phases

```
DKTP
DKTP2
DKTP3
DKTP4
DKTP5
```

RELOCATABLE LIBRARY

Modules

```
IJWDT
IJWDT1
IJWDT3
IJWDT4
IJWGEN
IJWLAB
```

Linkage Edit Statements

```
   INCLUDE IJWDT
   PHASE DKTP5,IJWDTCS2,NOAUTO
   INCLUDE IJWLAB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWDT
DELETR IJWDT1
```

Initialize Tape

CORE IMAGE LIBRARY

Phase

INTT

RELOCATABLE LIBRARY

Module

IJWIT

Linkage Edit Statements

```
   PHASE INTT,*,NOAUTO
   INCLUDE IJWIT
// EXEC LNKEDT
```

Delete Statements

DELETR IJWIT

Restore Tape-to-Disk or Data Cell

CORE IMAGE LIBRARY

Phases

CRTD

RELOCATABLE LIBRARY

Modules

IJWRT
IJWRT1

Linkage Edit Statements

```
   INCLUDE IJWRT
// LBLTYP NSD(10)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWRT
DELETR IJWRT1
```

Tape Compare

CORE IMAGE LIBRARY

Phases

TPCP
TPCP2
TPCP3

RELOCATABLE LIBRARY

Modules

IJWTCP
IJWTCP2
IJWTCP3
IJWTPCP
IJWXIT

Linkage Edit Statements

```
   PHASE TPCP,*,NOAUTO
   INCLUDE IJWTCP
   INCLUDE IJJCPD0
   INCLUDE IJWXIT
   INCLUDE IJWTPCP
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWTPCP
DELETR IJWTCP
DELETR IJWTCP2
DELETR IJWTCP3
DELETR IJWXIT
```

Tape to Card

CORE IMAGE LIBRARY

Phases

TPCD
TPCD2

```
TPCD3
TPCD4
TPCD5
```

RELOCATABLE LIBRARY

Modules

```
IJWGEN
IJWIAB
IJWTC
IJWTC1
IJWTC3
IJWTC4
```

Linkage Edit Statements

```
   INCLUDE IJWTC
   PHASE TPCD5,IJWTCCS2,NOAUTO
   INCLUDE IJWLAB
// IBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWTC
DELETR IJWTC1
DELETR IJWTC3
DELETR IJWTC4
```

Tape to Data Cell

CORE IMAGE LIBRARY

Phases

```
TPDC
TPDC2
TPDC3
TPDC4
TPDC5
```

RELOCATABLE LIBRARY

Modules

```
IJWGEN
IJWIAB
IJWTD3
IJWTD4
IJWTM
IJWTM1
```

Linkage Edit Statements

```
   INCLUDE IJWTM
   PHASE TPCD5,IJWTMCS2,NOAUTO
   INCLUDE IJWIAB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWTM
DELETR IJWTM1
```

Tape to Disk

CORE IMAGE LIBRARY

Phases

```
TPDK
TPDK2
TPDK3
TPDK4
TPDK5
```

RELOCATABLE LIBRARY

Modules

```
IJWGEN
IJWIAB
IJWTD
IJWTD1
IJWTD3
IJWTD4
```

Linkage Edit Statements

```
   INCLUDE IJWTD
   PHASE TPDK5,IJWTDCS2,NOAUTO
   INCLUDE IJWLAB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWTD
DELETR IJWTD1
```

Tape to Printer

CORE IMAGE LIBRARY

Phases

TPPR
TPPR2
TPPR3
TPPR4
TPPR5

RELOCATABLE LIBRARY

Modules

IJWGEN
IJWLAB
IJWTP
IJWTP1
IJWTP3
IJWTP4

Linkage Edit Statements

```
   INCLUDE IJWTP
   PHASE TPPR5,IJWTPCS2,NOAUTO
   INCLUDE IJWLAB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

DELETR IJWTP
DELETR IJWTP1
DELETR IJWTP3
DELETR IJWTP4

SOURCE STATEMENT LIBRARY

Sample Problems

Z.UTTPPR1
Z.UTTPPR2

SAMPLE PROBLEM SUMMARY

Figure 32-1 defines the files required by these sample problems. Figure 32-2 illustrates the card decks punched out when the sample problems are retrieved from the source statement library.

Disk Extent No. 1	Tape Drive No. 2
SYSRES	SYS004

Figure 32-1. TPPR--Sample Problem File Requirements

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.UTTPPR1 BKEND Z.UTTPPR1 T-P Control Cards (2 Cards) BKEND	$462	C001-0002
CATALS Z.UTTPPR2 BKEND Z.UTTPPR2 T-P Control Cards (2 Cards) BKEND	$462	C101-0102

Figure 32-2. TPPR--Sample Problem Card Decks

Tape to Printer Utility, Sample Problem 1

Program Name is Z.UTTPPR1. The Tape-to-Printer Utility sample problem lists the numbers handled by the assembler 1 and tape scrt/merge sample problems, allowing the user to check for proper functioning of all three programs.

To execute the tape to printer utility sample problem, you need the following job control cards:

```
// JOB TPPR
// PAUSE ASSGN SYS004 TO SYS001 OF PREVIOUS JOB
// ASSGN SYS004,X'cuu'
// PAUSE ASSGN SYS005 TO THE PRINTER
// ASSGN SYS005,X'cuu'
// UPSI 1000
// EXEC TPPR
   T-P Control Cards (2 cards)
/*
/&
```

The operator must reply IT IS EOF to the logged message by entering Y and EOB/END from SYSLOG..

SYSLOG Output

```
// JOB TPPR
// PAUSE ASSGN SYS004 to SYS001 OF PREVIOUS JOB
// PAUSE ASSGN SYS005 TO THE PRINTER
8001D IS IT EOF
Y
EOJ TPPR
```

SYSLST Output Summary:

- Job control cards
- Tape-to-printer control cards
- Utility parameter statements (11 lines)
- Numbers 0001-2000 (400 lines)
- IS IT EOF
- REPLY Y
- END OF DATA
- NUMBER OF INPUT BLOCKS PROCESSED 000400
- NUMBER OF OUTPUT BLOCKS PROCCESSED 000400
- END OF JOB

Tape to Printer Utility, Sample Problem 2

Program Name is Z.UTTPPR2. The Tape-to-Printer Utility sample problem lists the numbers handled by the Assembler 4 and tape and disk sort/merge (2400 application) sample programs.

To execute the tape to printer utility sample problem, you need the following job control cards:

```
// JOB TPPR 2400 TO PRINTER
// PAUSE ASSGN SYS004 TO SYS001 OF PREVIOUS JOB
// ASSGN SYS004,X'cuu'
// PAUSE ASSGN SYS005 TO THE PRINTER
// ASSGN SYS005,X'cuu'
// UPSI 1000
// EXEC TPPR
   Tape-to-printer utility control cards (2 cards)
/*
/&
```

The operator must reply to the logged message by entering Y and ECB/END from SYSLOG.

SYSLOG Output

```
BG // JOB TPPR 2400 TO PRINTER
BG // PAUSE ASSGN SYS004 TO SYS001 OF PREVIOUS JOB
BG
BG // PAUSE ASSGN SYS005 TO THE PRINTER
BG
BG 8001D IS IT EOF
Y
BG EOJ TPPR
```

SYSLST Output Summary:

- Job control cards
- Tape-to-printer control cards
- Utility parameter statements (11 lines)
- Numbers 0001-2000 (400 lines)
- IS IT EOF
- REPLY Y
- END OF DATA
- NUMBER OF INPUT BLOCKS PROCESSED 000400
- NUMBER OF OUTPUT BLOCKS PROCESSED 000400
- END OF JOB

Tape to Tape

CORE IMAGE LIBRARY

Phases

```
TPTP
TPTP2
TPTP3
TPTP4
TPTP5
```

RELOCATABLE LIBRARY

Modules

```
IJWGEN
IJWLAB
IJWTT
IJWTT1
IJWTT3
IJWTT4
```

Linkage Edit Statements

```
   INCLUDE IJWTT
   PHASE TPTP5,IJWTTCS2,NOAUTO
   INCLUDE IJWLAB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWTT
DELETR IJWTT1
DELETR IJWTT3
DELETR IJWTT4
```

Module 33: Utilities-Group 3--360N-UT-463

Modular Outline

The following are the library requirements for the Group 3 Utility programs:

Component	Libraries: Core Image Blocks Phases	2311	2314/ 2319	Relocatable Unique Modules	Blocks
Utilities Group 3 (Note 1):	32	74	76		326
Alternate Track Assignment Data Cell	5	17	17	6	
Clear Data Cell	3	4	4	2	
Data Cell to Data Cell	5	11	11	2	
Data Cell to Disk	5	11	11	2	
Data Cell to Printer (Note 2)	5	12	12	2	
Disk to Data Cell	5	11	11	2	
Initialize Data Cell	4	8	10	5	

Note 1: Following are the various combinations of utility modules common to Group 3:

- Common modules for Groups 1 and 3 are: IJWCLD2, IJWCLD3, IJWDD3, IJWDD4, IJWDP3, and IJWDP4.
- Common modules for all 3 groups are: IJWGEN and IJWIAB.
- The block count for common modules is not included in the total for the group.

Note 2: The data cell-to-printer utility program has one sample problem in the source statement library that requires one block.

PARTITION SIZE

10,240 bytes

Note: To delete all Group 1, Group 2, and Group 3 utilities, use the following statement:

```
DELETR IJW.ALL
```

Alternate Track Assignment Data Cell

CORE IMAGE LIBRARY

Phases

```
ATAM
ATAM2
ATAM3
ATAM4
ATAM5
```

RELOCATABLE LIBRARY

Modules

```
IJWALTM
IJWAM1
IJWAM2
IJWAM3
IJWAM4
IJWAM5
```

Linkage Edit Statements

```
   INCLUDE IJWALTM
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWALTM
DELETR IJWAM1
DELETR IJWAM2
DELETR IJWAM3
DELETR IJWAM4
DELETR IJWAM5
```

Clear Data Cell

CORE IMAGE LIBRARY

Phases

CLDC
CLDC2
CLDC3

RELOCATABLE LIBRARY

Modules

IJWCLM
IJWCLM1
IJWCLD2
IJWCLD3

Linkage Edit Statements

```
   INCLUDE IJWCLM
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCLM
DELETR IJWCLM1
```

Data Cell to Data Cell

CORE IMAGE LIBRARY

Phases

DCDC
DCDC2
DCDC3
DCDC4
DCDC5

RELOCATABLE LIBRARY

Modules

IJWDD3
IJWDD4
IJWGEN
IJWIAB
IJWMM
IJWMM1

Linkage Statements

```
   INCLUDE IJWMM
   PHASE DCDC5,IJWMMCS2,NCAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWMM
DELETR IJWMM1
```

Data Cell to Disk

CORE IMAGE LIBRARY

Phases

DCDK
DCDK2
DCDK3
DCDK4
DCDK5

RELOCATABLE LIBRARY

Modules

IJWDD3
IJWDD4
IJWGEN
IJWIAB
IJWMD
IJWMD1

Linkage Edit Statements

```
   INCLUDE IJWMD
   PHASE DCDK5,IJWMDCS2,NOAUTO
   INCLUDE IJWIAB
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWMD
DELETR IJWMD1
```

Data Cell to Printer

CORE IMAGE LIBRARY

Phases

DCPR
DCPR2
DCPR3
DCPR4
DCPR5

RELOCATABLE LIBRARY

Modules

IJWDP3
IJWDP4
IJWGEN
IJWLAB
IJWMP
IJWMP1

Linkage Edit Statements

```
   INCLUDE IJWMP
   PHASE DCPR5,IJWMPCS2,NOAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWMP
DELETR IJWMP1
```

SOURCE STATEMENT LIBRARY

Sample Problem

Z.UTDCPR1

DATA CELL-TO-PRINTER SAMPLE PROBLEM SUMMARY:

Program Name is Z.UTDCPR1. The data cell-to-printer sample problem lists the numbers that the Assembler 3 sample problem wrote on the data cell. Figure 33-1 illustrates the card deck produced when the sample problem is retrieved from the source statement library. Disk extent number 1 must be assigned to SYS004.

Cards Retrieved	Card Cols. 73-76	Card Cols. 77-80
CATALS Z.UTDCPR1		
BKEND Z.UTDCPR1		
DC-P Control Cards (2 Cards)	$463	0001-0002
BKEND		

Figure 33-1. UTILITIES--Sample Problem Card Deck

To execute the data cell-to-printer sample problem, the following job control cards are needed:

```
                Col. 16                                  Col. 54      Col. 72
                  ↓                                        ↓            ↓
// JOB DCPR
// PAUSE ASSGN SYS004 TO DATA CELL
// ASSGN SYS004,X'cuu'
// PAUSE ASSGN SYS005 TO THE PRINTER
// ASSGN SYS005,X'cuu'
// UPSI 0000
// DLBL UIN,'SAMPLE PROBLEM FILE OF 100 RECORDS'[,yy/ddd]
// EXTENT SYS004,hhhhhh,1,0,10200,20,,B=3
// EXEC DCPR
   DC-P Control Cards (2 cards)
/*
/&
```

Note: Replace hhhhhh in the EXTENT card with the volume serial number of the data cell.

SYSLOG Output Is:

```
// JOB DCPR
// PAUSE ASSGN SYS004 TO THE DATA CELL
// PAUSE ASSGN SYS005 TO THE PRINTER
EOJ DCPR
```

SYSLST Output Summary:

- Job control cards
- Data cell-to-printer control cards
- Utility parameter statements (10 lines)
- Number 0100-0001 (100 lines)
- NUMBER OF INPUT BLOCKS PROCESSED 000100
- NUMBER OF OUTPUT BLOCKS PROCESSED 000100
- END OF JOB

Disk to Data Cell

CORE IMAGE LIBRARY

Phases

```
DKDC
DKDC2
DKDC3
DKDC4
DKDC5
```

RELOCATABLE LIBRARY

Module

```
IJWDD3
IJWDD4
IJWDM
IJWDM1
IJWGEN
IJWIAB
```

Linkage Edit Statements

```
   INCLUDE IJWDM
   PHASE DKDC5,IJWDMCS2,NOAUTO
   INCLUDE IJWLAB
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWDM
DELETR IJWDM1
```

Initialize Data Cell

CORE IMAGE LIBRARY

Phases

```
INTM
INTM2
INTM3
INTM4
```

RELOCATABLE LIBRARY

Modules

```
IJWIM
IJWIM1
IJWIM2
IJWIM3
IJWIM4
```

Linkage Edit Statements

```
   INCLUDE IJWIM
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWIM
DELETR IJWIM1
DELETR IJWIM2
DELETR IJWIM3
DELETR IJWIM4
```

Module 34: System Utilities -- 370N-UT-491

Modular Outline

Modular Index

The library requirements for the Utility programs are the following:

Component	Phases	Libraries: Core Image Blocks 2311	2314/ 2319	3330	Relocatable Modules
Assign Alternate Track Disk	5	26	27	29	6
Assign Alternate Track Data Cell	5	17	17	19	6
Clear Disk	3	5	5	5	4
Clear Data Cell	3	5	5	5	4
Copy Disk to Card	2	7	7	7	3
Copy Disk to Disk	2	7	8	8	3
Copy Disk or Data Cell to Tape	2	7	7	7	3
Initialize Disk	4	16	16	17	5
Initialize Data Cell	4	8	10	11	5
Initialize Tape	1	3	3	3	2
Restore Card to Disk	1	4	4	5	2
Restore Tape to Disk or Data Cell	1	4	5	5	2
VTOC Display	1	3	3	4	4
Transient	1	1	1	1	

PARTITION SIZE

10,240 bytes

Note: To delete all utilities, use the following statement:

```
DELETR IJW.ALL
```

Assign Alternate Track Disk

CORE IMAGE LIBRARY

Phases

ALTDK
ALTDK2
ALTDK3
ALTDK4
ALTDK5

RELOCATABLE LIBRARY

Modules

IJWATD
IJWATD1
IJWATD2
IJWATD3
IJWATD4
IJWATD5

Linkage Edit Statements

```
   INCLUDE IJWATD
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWATD
DELETR IJWATD1
DELETR IJWATD2
DELETR IJWATD3
DELETR IJWATD4
DELETR IJWATD5
```

Assign Alternate Track Data Cell

CORE IMAGE LIBRARY

Phases

ALTDC
ALTDC2
ALTDC3
ALTDC4
ALTDC5

RELOCATABLE LIBRARY

Modules

IJWATM
IJWATM1

IJWATM2
IJWATM3
IJWATM4
IJWATM5

Linkage Edit Statements

```
   INCLUDE IJWATM
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWATM
DELETR IJWATM1
DELETR IJWATM2
DELETR IJWATM3
DELETR IJWATM4
DELETR IJWATM5
```

Clear Disk

CORE IMAGE LIBRARY

Phases

CLRDK
CLRDK2
CLRDK3

RELOCATABLE LIBRARY

Modules

IJWCCD
IJWCCD1
IJWCCD2
IJWCCD3

Linkage Edit Statements

```
   INCLUDE IJWCCD
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCCD
DELETR IJWCCD1
DELETR IJWCCD2 (See note)
DELETR IJWCCD3 (See note)
```

Note: Only if CLEAR DATA CELL not performed.

Clear Data Cell

CORE IMAGE LIBRARY

Phases

CLRDC
CLRDC2
CLRDC3

RELOCATABLE LIBRARY

Modules

IJWCCM
IJWCCM1
IJWCCD2
IJWCCD3

Linkage Edit Statements

```
   INCLUDE IJWCCM
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCCM
DELETR IJWCCM1
DELETR IJWCCD2 (See note)
DELETR IJWCCD3 (See note)
```

Note: Cannot delete if already deleted by CLEAR DISK.

Copy Disk to Card

CORE IMAGE LIBRARY

Phases

CDKCD
CDKCD2

RELOCATABLE LIBRARY

Modules

```
IJWCDC
IJWCDC1
IJWCDC2
```

Linkage Edit Statements

```
   INCLUDE IJWCDC
// IBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCDC
DELETR IJWCDC1
DELETR IJWCDC2
```

Copy Disk to Disk

CORE IMAGE LIBRARY

Phases

```
CDKDK
CDKDK2
```

RELOCATABLE LIBRARY

Modules

```
IJWCDD
IJWCDD1
IJWCDD2
```

Linkage Edit Statements

```
   INCLUDE IJWCDD
// IBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCDD
DELETR IJWCDD1
DELETR IJWCDD2
```

Copy Disk or Data Cell to Tape

CORE IMAGE LIBRARY

Phases

```
CDKTP
CDKTP2
```

RELOCATABLE LIBRARY

Modules

```
IJWCDT
IJWCDT1
IJWCDT2
```

Linkage Edit Statements

```
   INCLUDE IJWCDT
// IBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWCDT
DELETR IJWCDT1
DELETR IJWCDT2
```

Initialize Disk

CORE IMAGE LIBRARY

Phases

```
INTDK
INTDK2
INTDK3
INTDK4
```

RELOCATABLE LIBRARY

Modules

```
IJWIND
IJWIND1
IJWIND2
IJWIND3
IJWIND4
```

Linkage Edit Statements

```
   INCLUDE IJWIND
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWIND
DELETR IJWIND1
DELETR IJWIND2
DELETR IJWIND3
DELETR IJWIND4
```

Initialize Data Cell

CORE IMAGE LIBRARY

Phases

```
INTDC
INTDC2
INTDC3
INTDC4
```

RELOCATABLE LIBRARY

Modules

```
IJWINM
IJWINM1
IJWINM2
IJWINM3
IJWINM4
```

Linkage Edit Statements

```
   INCLUDE IJWINM
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWINM
DELETR IJWINM1
DELETR IJWINM2
DELETR IJWINM3
DELETR IJWINM4
```

Initialize Tape

CORE IMAGE LIBRARY

Phase

```
INTTP
```

RELOCATABLE LIBRARY

Modules

```
IJWINT
IJWINT1
```

Linkage Edit Statements

```
   INCLUDE IJWINT
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWINT
DELETR IJWINT1
```

Restore Card to Disk

CORE IMAGE LIBRARY

Phase

```
RCDDK
```

RELOCATABLE LIBRARY

Modules

```
IJWRCD
IJWRCD1
```

Linkage Edit Statements

```
   INCLUDE IJWRCD
// LBLTYP NSD(nn)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWRCD
DELETR IJWRCD1
```

Restore Tape to Disk or Data Cell

CORE IMAGE LIBRARY

Phase

```
RTPDK
```

RELOCATABLE LIBRARY

Modules

```
IJWRTD
IJWRTD1
```

Linkage Edit Statements

```
   INCLUDE IJWRTD
// LBLTYP NSD(10)
// EXEC LNKEDT
```

Delete Statements

```
DELETR IJWRTD
DELETR IJWRTD1
```

VTOC Display

CORE IMAGE LIBRARY

Phase

```
LVTOC
```

Transient

```
$$BLVTOC
```

Note: $$BLVTOC is catalcged into the core image library along with the VTOC program.

RELOCATABLE LIBRARY

Mcdules (BJS)

```
IJWLTVB
IJWLTVT
IJWLTV1
```

Linkage Edit Statements, Non-Multiprogramming System

```
   INCLUDE IJWLTVB
// LBLTYP TAPE
// EXEC LNKEDT
```

Delete StateMents, Non-multiprogramming System

```
DELETR  IJWLTVB
DELETR  IJWLTVT DELETR  IJWLTV1
```

Mcdules (MPS)

```
IJWLTVM
IJWLTVT
IJWLTV1
```

Linkage Edit Statements, Multiprogramming System

```
   INCLUDE IJWLTVM
// EXEC LNKEDT
```

Delete Statements, Multiprogramming System

```
DELETR  IJWLTVM
DELETR  IJWLTVT
DELETR  IJWLTV1
```

CLEAR DISK UTILITY, SAMPLE PROBLEM

The Clear Disk utility sample problem preformats cylinders 1 - 10 of a 3330 disk with fixed-length blocks containing count, key, and data areas. Key areas are 9 bytes in length each; each data area is 80 bytes long. Both key and data areas are filled with hexadecimal zeros (X'00', defined by the X'xx' parameter of the utility modifier statement).

To execute the Clear Disk sample problem, you need the following job control cards:

```
// JOB CLRDK
// PAUSE ASSGN SYS000 TO THE 3330 DISK TO BE CLEARED
// DLBL UOUT,'DISK LABEL'
// EXTENT SYS000,,1,0,00019,00190
// EXEC CLRDK
// UCL B=(K=9,D=80),X'00',OY,E=(3330)
// END
/&
```

SYSLOG Output Is:

```
// JOB CLRDK
// PAUSE ASSGN SYS000 TO THE 3330 DISK TO BE CLEARED
ASSGN SYS000,X'cuu'
EOJ CLRDK
```

SYSLST Output Summary:

- Job Control Cards
- Clear Disk control card
- Utility parameter statements (10 lines)
- END OF JOB

INITIALIZE DISK UTILITY, SAMPLE PROBLEM

The initialize disk utility sample problem initializes a 3330 disk pack. The VTOC is written on cylinder 0.

To execute the initialize disk sample problem, you need the following job control cards:

```
// JOB INIT
// PAUSE ASSGN SYS000 TO THE 3330 DISK TO BE INITIALIZED
// EXEC INTDK
UID IQ
// VTOC STANDARD
VOL1111111
// END
/&
```

If any unexpired files are encountered, the message

```
8118D UNEXPIRED FILE
```

is displayed. Type in 2 and press END to delete the file. Any other reply results in termination of the job.

SYSLOG Output Is:

```
// JOB INIT
// PAUSE ASSGN SYS000 TO THE 3330 DISK TO BE INITIALIZED
ASSGN SYS000,X'cuu'
8120I END OF INIT.DISK
EOJ INIT
```

SYSLST Output Summary:

- Job control cards
- Initialize disk control card
- 8120I END OF INITIALIZE DISK
- END OF JOB

Module 35: EREP--370N-UT-492

Modular Outline

PARTITION SIZE

10.240 bytes.

CORE IMAGE LIBRARY

	Blocks		
Phases	2311	2314/ 2319	3330
68	112	114	123

Modular Outline

Phases

EREP
EREPADTP
EREPASTA
EREPCCH
EREPCCHC
EREPC135
EREPC145
IJBERDE
EREPDOLD
EREPEDCP
EREPEDCR
EREPEDCU
EREPEDDA
EREPEDES
EREPEDD1
EREPEDIT
EREPEDTD
EREPEDTP
EREPEDTR
EREPEDT1
EREPEDUR
EREPEDU1
EREPEDX1
EREPEOD
EREPESPT
EREPESTR
EREPESWK
EREPHIST
EREPIPL
EREPMCAR
EREPMCRC
EREPMNTR
EREPM145
EREPRETV
EREPSHRT
EREPSHRU
EREPSMCP
EREPSMCR
EREPSMCU
EREPSMDA
EREPSMD1
EREPSMES
EREPSMTD
EREPSMTP
EREPSMTR
EREPSMT1
EREPSMUR
EREPSMU1
EREPSPCL
EREPTES
EREPTPE1
EREPUCCM
EREPUNIT
EREPUOLD
EREP145A
EREP145B
EREP145C
EREP145D
EREP2400
EREP2715
EREP3211
EREP333X
EREP333Y
EREP333Z
EREP3330
EREP3410
EREP3420
EREP3500

RELOCATABLE LIBRARY

Modules	Blocks
70	509

Modules

IJBEREP
IJBEADTP
IJBEASTA
IJBECCH
IJBECCHC
IJBEC135
IJBEC145
IJBEDOLD
IJBEEDCP
IJBEEDCR
IJBEEDCU
IJBEEDDA
IJBEEDD1
IJBEEDES
IJBEEDIT
IJBEEDTD
IJBEEDTP
IJBEEDTR

```
IJBEEDT1
IJBEEDUR
IJBEEDU1
IJBEEDX1
IJBEEOD
IJBEESPT
IJBEESTR
IJBEESWK
IJBEHIST
IJBEIPL
IJBEMCAR
IJBEMCRC
IJBEMNTR
IJBEM145
IJBERDE
IJBERETV
IJBESHRT
IJBESHRU
IJBESMCP
IJBESMCR
IJBESMCU
IJBESMDA
IJBESMD1
IJBESMES
IJBESMTD
IJBESMTP
IJBESMTR
IJBESMT1
IJBESMUR
IJBESMU1
IJBESPCL
IJBETES
IJBETPE1
IJBEUCCM
IJBEUNIT
IJBEUOLD
IJBE145A
IJBE145B
IJBE145C
IJBE145D
IJBE2400
IJBE2715
IJBE3211
IJBE333Z
IJBE3330
IJBE3410
IJBE3420
IJBE3500
IJBECALA
IJBECALB
IJBECALC
IJBECALD
```

Linkage Edit Statements

```
   PHASE EREP,S+80*    (For a non-multiprogramming system)    Choose
   PHASE EREP,+0       (For a multiprogramming system)          one
   INCLUDE IJBECALA    (Monitor)
// EXEC LNKEDT
   INCLUDE IJBECALB    (Edit/Select Retrieval)
// EXEC LNKEDT
   INCLUDE IJBECALC    (Summary)
// EXEC LNKEDT
   INCLUDE IJBECALD    (Tape Error Statistics)
// EXEC LNKEDT
   INCLUDE IJBERDE     (RDE Summary**)
// EXEC LNKEDT
```

* 80 byte area reserved for tape label processing. This procedure does away with the necessity of including a // LBLTYP card.

**The summary function will supply meaningful information only if ERRLOG=RDE is specified during system generation.

Delete Statements

```
DELETR IJBEREP,IJBEADPT,IJBEASTA,etc.
```

Glossary

For a more complete list of data processing terms, refer to IBM Data Processing Techniques, A Data Processing Glossary, GC20-1699.

ANSI (American National Standards Institute, Inc.) Label Format: The tape file format used when the label is written in the ASCII mode.

ASCII (American National Standard Code for Information Interchange: A 128-character, 7-bit code. The high-order bit in the System/360 8-bit environment is zero.

CCH (Channel Check Handler): A feature that assesses channel errors to determine if the system can continue operations.

channel inbcard error: An error that occurs between one I/O device and the central processing unit.

chronological area of the recorder file: The area of the recorder (IJSYSRC) where error records are printed as they occur. The record types included are MCAR, CCH, unit check, IPL/EOD, counter overflow records for temporary read/write errors, magnetic tape statistics by volume, and miscellaneous records.

DOS Tape Error Statistics (TES): A facility provided by RMSR that monitors and records read and write errors per volume for 2400 and 3400 series tapes.

data set security: A feature that provides protection for disk files. A data secured file cannot be accidentally accessed by a problem program.

EREP (Environmental Recording, Editing, and Printing): A program that edits and prints the data contained on the system recorder file.

EVA (Error Volume Analysis): An option that causes the system to issue a message to the operator when a number of temporary read or write errors (specified by the user at system generation time) has been exceeded on a currently accessed tape volume.

IDRA (Independent Directory Read-in Area): A resident area created by a supervisor option that the system uses to read core image library directories for fetch and load operations. Using IDRA frees the physical transient area to perform error recovery procedures.

I/O (input/output) error logging: The process of recording I/O error records on the system recorder file.

job accounting interface: A program that accumulates accounting information for each job step to charge usage of the system, help plan new applications, and help supervise system operation more efficiently.

LSERV (label cylinder display): A program that formats a listing of the label cylinder located on SYSRES. LSERV can run in any partition and outputs the list on SYSLST, which may be assigned to disk, tape, or printer.

MCAR (machine check analysis and recording): A feature that records System/370 machine check interrupt error information on the system recorder file and then attempts to recover from the interrupt.

MCI (machine check interrupt): The interrupt that occurs if the central processing unit fails to operate.

PCIL (private core image library): A file referenced in the same manner and for the same purposes as the system core image library, but distinct from the system core image library. PCIL increases available core image library space to enable compiling, linkage editing, and executing in the foreground partition when a private core image library is assigned to that partition.

PDAID (Problem Determination Aids): Programs that trace a specified event when it occurs during the operation of a program.

problem determination: A procedure or process (provided by IBM) to assist users in determining the cause of a system error.

SORTED DSERV: A program that gives you an alphamerically sorted listing of any or all of the library directories.

stand-alone dump: A program that displays the contents of main storage from a minimum of 8K bytes to a maximum of 1,6384K bytes. It helps to determine the cause of an error.

system recorder file: The system logical unit (SYSREC) that contains RMSR records.

transients, IBM-supplied (CONT.)

$$BCOBER	15
$$BCOBR1	15
$$BCQC01	29
$$BCQC02	29
$$BCQC03	29
$$BCTC01	28
$$BDRSTR	1
$$BDUMP	1
$$BDUMPB	1
$$BDUMPD	1
$$BDUMPF	1
$$BENDFF	1
$$BENDFL	1
$$BEOJ	1
$$BEOJ1	1
$$BEOJ2	1
$$BEOJ2A	1
$$BEOJ3	1
$$BEOJ3A	1
$$BEOJ4	1
$$BEOJ5	1
$$BEOJ7	1
$$BERPTP	10
$$BERRTN	1
$$BETPRT	28
$$BFCB	1
$$BFCMUL	15
$$BHDRCK	28
$$BIIQBD	2
$$BIIQBS	2
$$BIIQMW	2
$$BIIQSD	2
$$BIIQSS	2
$$BIIQT1	2
$$BIIQT2	2
$$BIIQT3	2
$$BIIQT4	2
$$BIIQT5	2
$$BIIQT6	2
$$BIIRBD	3
$$BIIRBS	3
$$BIIRMW	3
$$BIIRSD	3
$$BIIRT1	3
$$BIIRT2	3
$$BIIRT3	3
$$BIIRT4	3
$$BIIRT5	3
$$BIIRT6	3
$$BILSVC	1
$$BINDEX	1
$$BJCOPT	1
$$BJCOP1	1
$$BLEPRT	28
$$BLISTV	1,31
$$BLOPEN	28
$$BLSTIO	1
$$BLVTOC	1
$$BMMR20	7
$$BMSGWR	1
$$BMU100	30
$$BMU200	30
$$BMU300	30
$$BOCPM1	1
$$BOCPM2	1
$$BOCPT1	1
$$BOCPT2	1
$$BOCPT3	1
$$BOCP01	1
$$BOCP02	1
$$BOCP11	1
$$BOCP12	1
$$BODACL	1
$$BODAIN	1
$$BODAI1	1
$$BODAO1	1
$$BODAO2	1
$$BODAO3	1
$$BODAO4	1
$$BODAU1	1
$$BODQUE	1
$$BODSMW	1
$$BODSPV	1
$$BODSPW	1
$$BOFLPT	1
$$BOIS01	1
$$BOIS02	1
$$BOIS03	1
$$BOIS04	1
$$BOIS05	1
$$BOIS06	1
$$BOIS07	1
$$BOIS08	1
$$BOIS09	1
$$BOIS10	1
$$BOLTEP	27
$$BOMRCE	1
$$BOMR01	7
$$BOMSG1	1
$$BOMSG2	1
$$BOMSG3	1
$$BOMSG4	1
$$BOMSG5	1
$$BOMSG6	1
$$BOMSG7	1
$$BOMT0M	1
$$BOMT0W	1
$$BOMT01	1
$$BOMT02	1
$$BOMT03	1
$$BOMT04	1
$$BOMT05	1
$$BOMT06	1
$$BOMT07	1
$$BONVOL	1
$$BOOR01	9
$$BOPEN	1
$$BOPENC	1
$$BOPEND	1
$$BOPENR	1
$$BOPEN1	1
$$BOPEN2	1
$$BOPIGN	1
$$BOPNLB	1
$$BOPNR2	1
$$BOQO01	29
$$BOQO02	29
$$BOQO03	29
$$BOQO04	29
$$BOQO06	29
$$BOQO07	29
$$BOQO08	29
$$BORTV1	1
$$BORTV2	1
$$BOSDC1	1
$$BOSDC2	1
$$BOSDEV	1

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

READER'S COMMENT FORM

GC33-5008-1

This sheet is for comments and suggestions about this manual. We would appreciate *your* views, favorable or unfavorable, in order to aid us in improving *this* publication. This form will be sent directly to the author's department. Please include your name and address if you wish a reply. Contact your IBM branch office for answers to technical questions about the system or when requesting additional publications. Thank you.

Your comments* and suggestions:

*** We would especially appreciate your comments on any of the following topics:**

Clarity of the text	Accuracy	Index	Illustrations	Appearance	Paper
Organization of the text	Cross-references	Tables	Examples	Printing	Binding

YOUR COMMENTS, PLEASE . . .

This manual is part of a library that serves as a reference source for systems analysts, programmers and operators of IBM systems. Your answers to the questions on the back of this form, together with your comments, will help us produce better publications for your use. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Please note: Requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

Fold

Fold

FIRST CLASS
PERMIT NO. 1359
WHITE PLAINS, N. Y.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM Corporation
1133 Westchester Avenue
White Plains, N.Y. 10604

Attention: Department 813 U

Fold

Fold

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

CUT ALONG THIS LINE

DOS Version 4 System Generation Printed in U.S.A. GC33-5008-1