

The IBM logo consists of the letters "IBM" in a bold, white, sans-serif font, set against a solid black square background.

Systems Reference Library

**IBM System/360 Operating System
Starter Operating System Guide**

Program Number 360S-CI-514

This publication provides the information necessary to prepare the starter operating system package for use. It describes the contents of the starter system, configuration requirements, initialization, preparation for system generation, and the sample programs to test the functioning of components of the system.



PREFACE

This publication provides systems programmers the information necessary for preparing the starter operating system package for use in system generation. The following publications are prerequisites for the use of this manual.

IBM System/360 Operating System: Concepts and Facilities, Form C28-6535.

IBM System/360 Operating System: Utilities, Form C28-6586.

IBM System/360 Operating System: Operator's Guide, Form C28-6540.

IBM System/360 Operating System: Job Control Language, Form C28-6539.

IBM System/360 Operating System: System Generation, Form C28-6554.

Section I of this publication provides a general description of the starter operating system package and detailed operating procedures for readying it for system generation. Section II describes the sample programs which may be used to test the functioning of components of the system; detailed operating procedures for executing these sample programs can be obtained from the operator's guide referred to above.

First Edition (September 1966)

Changes or additions to the specifications contained in this publication will be reported in subsequent revisions or technical newsletters.

This publication was prepared for production using an IBM computer to update the text and to control the page and line format. Page impressions for photo-offset printing were obtained from an IBM 1403 Printer using a special print chain.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form is provided at the back of this publication for reader's comments. If the form has been removed, comments may be addressed to the IBM Corporation, Programming Systems Publications, Department D58, PO Box 390, Poughkeepsie, N. Y. 12602

CONTENTS

INTRODUCTION	5	Description.	23
Starter Operating System	6	Operating Instructions	23
Libraries.	9	Output	23
Arrangement of Data Sets	10	FORTRAN E SAMPLE PROGRAM (IEJESP).	24
PROCESSING THE INITIAL PACKAGE	11	Description.	24
TAPE DISTRIBUTION.	12	Operating Instructions	24
DISK DISTRIBUTION (DISK BACKUP).	15	Output	24
DISK DISTRIBUTION (TAPE BACKUP).	17	COBOL E SAMPLE PROGRAM (IEPSAMP)	25
DELETING LIBRARIES	18	Description.	25
SECTION II: SAMPLE PROGRAMS	19	Operating Instructions	25
ASSEMBLER E&F SAMPLE PROGRAM (IETESP).	20	Output	25
Description	20	PL/1(F) SAMPLE PROGRAM (IEMSP)	26
Operating Instructions	20	Description.	26
Output	20	Operating Procedures	26
UPDATE ANALYSIS PROGRAM SAMPLE PROGRAM (IHGSAMP)	21	Output	26
Description.	21	EXPRESS GRAPHICS SAMPLE PROGRAM (SAMP2250, SAMP2260).	27
Operating Instructions	21	Description.	27
Output	21	Operating Instructions	27
Stage 1.	21	Output	27
Stage 2.	22	INDEX.	31
SORT SAMPLE PROGRAM (IERSP).	23		



The starter operating system package provides an operating system that can be used for system generation. However, before it can be used for system generation, certain initialization procedures must be performed. Performing these procedures gives

1. An operating system which may be used for system generation.
2. A backup copy of this operating system.
3. A set of independent utility programs in a usable format; these include
 Direct Access Device Initialization (DASDI)
 Dump/Restore (DUMPREST)
 Recover/Replace (RECOVREP)
 Initial Program Load (IEAIPL00)
4. Sample programs which will test the functioning of components of the generated system.

The starter operating system package consists of an operating system and a set of libraries. The operating system includes

1. Control program
2. Data set utilities
3. System utilities
4. Assembler F
5. Linkage Editor E & F

The initial package is provided on two disk packs or two tapes. (The volume serial numbers of the disk packs are DLIB01 and DLIB02.)

If the starter operating system is on tape, initialization consists of

1. Restoring the system to disks (the tapes then become a backup copy of the system).
2. Punching the independent utility programs from SYS1.SAMPLIB for later use.
3. Listing the system data.

The procedure for restoring the tape package to disks is detailed in "Processing the Initial Package."

If the starter operating system is on disks, it is ready to use; however, certain steps are necessary first:

1. Punch the independent utilities from SYS1.SAMPLIB on DLIB02.
2. Create backup copies of the system on disk or tape using the independent utilities.
3. List the data describing the system.

The procedure for performing these steps is detailed in the section "Processing the Initial Package."

Processing of the tape and disk package is shown schematically in Figure 1.

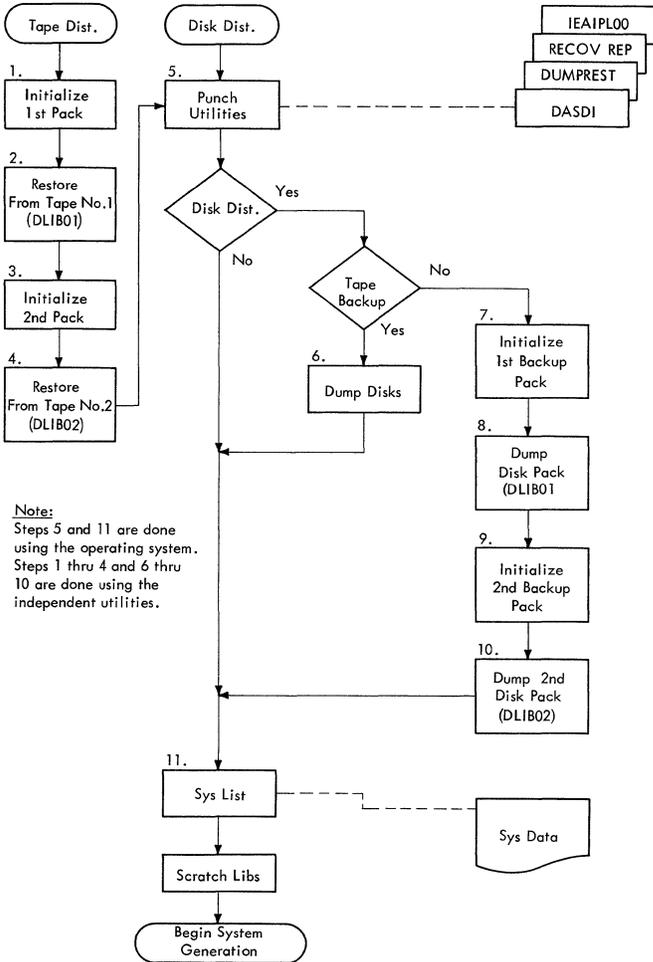


Figure 1. Processing the Initial Package

STARTER OPERATING SYSTEM

The starter operating system requires 64K bytes of main storage for operation. It contains a control program which can support the machine configuration shown in Figure 2 or any subset meeting the minimum requirements listed in Figure 3.

The control program permits the use of the generic unit names listed in Figure 4; additional unit names provided in the starter system are listed in Figure 5.

The starter operating system (DLIB01 and DLIB02) may be used for system generation on systems with greater than 64K storage and will use Assembler F and Linkage Editor F.

If system generation is performed with the starter operating system using a 1403 with the universal character set feature, the BPS UCS Utility program (360P-UT-048) must be executed prior to this in order to load the generator storage. When the UCS program is executed, NO-FOLDING and BLOCK-DATA-CHECK must be specified.

System generation on a system with only 64K storage must be performed using Linkage Editor E. To prepare DLIB01 for system generation with 64K storage, it is necessary to scratch the alias IEWL (IEWL is the alias for the largest linkage editor in a generated system--in the starter operating system, Linkage Editor F) from Linkage Editor F and change the Linkage Editor E module name IEWLE180 to IEWL. The following control statements must be used to accomplish this.

```
//RENAME JOB 123,PROGRAMMER, X
// MSGLEVEL=1
//RENAME EXEC PGM=IEHPROGM
//START DD DISP=OLD, X
// VOLUME=SER=DLIB01, X
// UNIT=2311
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
SCRATCH DSNAME=SYS1.LINKLIB, X
MEMBER=IEWL, X
VOL=2311=DLIB01
RENAME DSNAME=SYS1.LINKLIB, X
MEMBER=IEWLE180, X
NEWNAME=IEWL, X
VOL=2311=DLIB01
/*
```

System generation will then use Assembler F and Linkage Editor E.

Output from the linkage editor used in system generation will be in SYS1.LINKLIB and SYS1.FORTLIB and must be acceptable input to the linkage editor in the resulting system. This consideration and the chart below will assist in selecting linkage editors.

Linkage Editor Used in System Generation	SYS1.LINKLIB, SYS1.FORTLIB Contents Acceptable to Linkage Editor
E	E, F
F (with LBMAINT=E specified in GENERATE macro)	E, F
F (without LBMAINT=E parameter in GENERATE macro)	F

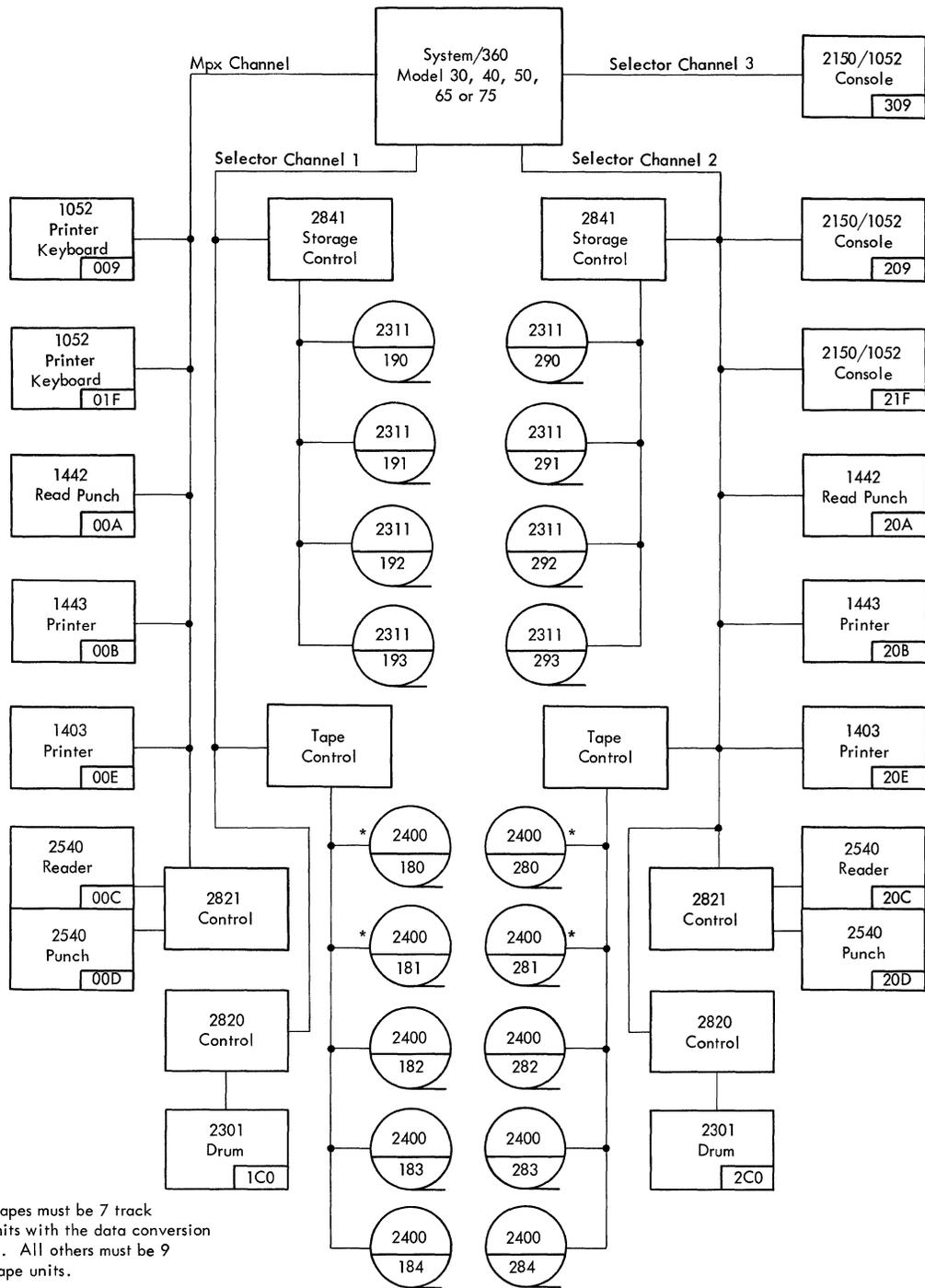


Figure 2. System Configuration

Minimum Requirement	Function	Choose from the following				
		Device	Device Address			
			MPX Channel	Selector Channel 1	Selector Channel 2	Selector Channel 3
1	System Console	1052	009,01F		209,21F	309
		2150/1052	009,01F		209,21F	309
2	Storage Units for System Residence and Data Sets	2311		190,191,192,193	290,291,292,293	
1	System Input	2540 Reader	00C		20C	
		1442 Read Punch *	00A		20A	
		2400 (7-Tr-DC)		180,181	280,281	
		2400 (9-Track)		182,183,184	282,283,284	
1	Punch Output	2540 Punch	00D		20D	
		1442 Read Punch *	00A		20A	
		2400 (7-Tr-DC)		180,181	280,281	
		2400 (9-Track)		182,183,184	282,283,284	
1	Print Output	1443	00B		20B	
		1403	00E		20E	
		2400 (7-Tr-DC)		180,181	280,281	
		2400 (9-Track)		182,183,184	282,283,284	
2**	Intermediate (Work) Data Sets	2311		190,191,192,193	290,291,292,293	
		2301		1C0	2C0	
		2400 (7Tr-DC)		180,181	280,281	
		2400 (9 Track)		182,183,184	282,283,284	
2**	Intermediate (Work) Data Sets	2311		190,191,192,193	290,291,292,293	
		2301		1C0	2C0	

* A single 1442 may serve as either system input or punch output, but not both simultaneously.

** The same direct access storage devices may serve for system residence and sequential and partitioned data sets if sufficient space is available.

Figure 3. Minimum I/O Requirements

<u>Magnetic Tape Drives</u>	
<u>Unit Name</u>	<u>Device Type</u>
2400	2400 Series 9-track Magnetic Tape Drive
2400-2	2400 Series Magnetic Tape Drive with Seven Track Compatibility and Data Conversion
<u>Direct-Access Devices</u>	
<u>Unit Name</u>	<u>Device Type</u>
2311	2311 Disk Storage Drive
2301	2301 Drum Storage
<u>Unit Record Equipment</u>	
<u>Unit Name</u>	<u>Device Type</u>
1052	1052 Printer Keyboard
1403	1403 Printer 1404 Printer (continuous form only)
1442	1442 Serial Reader Punch
1443	1443 Printer
2540	2540 Reader Punch (read feed)
2540-2	2540 Reader Punch (punch feed)

Figure 4. Generic Unit Names

<u>Name</u>	<u>Function</u>
SYSSQ	Sequential access on devices at any of the following addresses: 182, 183, 184, 282, 283, 284, 190, 191, 192, 193, 290, 291, 292, 293 (any 9-track tape or 2311 disk storage drive).
SYSDA	Direct access on devices at any of the following addresses: 190, 191, 192, 193, 290, 291, 292, 293, (any 2311 disk storage drive).
SYSCP	A 2540 card punch at address 00D or 20D.

Figure 5. Additional Unit Names Supporting IBM Supplied Cataloged Procedures

The starter operating system requires that all devices to be used by the system be ready prior to initial program loading. Any device not ready will automatically be varied off line at IPL time. If any device which was not ready at initial program loading time is required during a job step, the operator should enter a VARY ONLINE command for that device. When using the starter operating system, it is permissible to have devices other than those listed in Figure 2 attached to the system. However, an interrupt must not be issued from any of these additional devices while the starter

operating system is running. For example, the operator must not make ready any one of these devices.

LIBRARIES

In addition to the starter operating system, the starter system package includes the following libraries; each is a partitioned data set (PDS):

1. SYS1.MODLIB: This library contains the load modules from which the operating system programs are constructed or selected during system generation. It contains modules needed exclusively during system generation and maintenance updating.
2. SYS1.GENLIB: This library contains the macro definition modules that are unique to the system generation process. It is the COPY library of the assembler during stage I of system generation. It has a blocking factor of 42.
3. SYS1.PROCLIB: (Procedure Library) The members of the procedure library include IBM cataloged procedures used to perform certain system functions, e.g., compile-linkage edit-go. The records are unblocked.
4. SYS1.SAMPLIB: This library contains 80-character card images. Its principal use will be data decks for sample programs. This library also contains the DASDI, DUMPREST, RECOVERP, and IEAIPL00 programs.

5. **SYS1.MACLIB (Macro Library):** The members of the macro library include the macro-definitions for the system macro-instructions. It has a blocking factor of 42.
6. **SYS1.COBLIB (COBOL Library):** The members of the COBOL library are load modules (COBOL subroutines).
7. **SYS1.FORTLIB (FORTRAN Library):** This library is distributed without members. The members to be placed in it during system generation are contained in SYS1.MODLIB. All library members of SYS1.FORTLIB that have a non-IBM name are copied intact into the SYS1.FORTLIB of the new system.
8. **SYS1.PL1LIB (PL/I Library):** The members of the PL/I library are load modules (PL/I subroutines).
9. **SYS1.SORTLIB (Sort Library):** The members of the sort library are the load modules from which a sort/merge program is produced at execution time.

ARRANGEMENT OF DATA SETS

The starter operating system package on disk is operable (though a backup copy of it should be made); it contains the starter operating system and the libraries shown in Figure 6. (The data sets on each pack are not necessarily in the order shown.)

The starter operating system package on tape includes all the contents of the disk packs shown in Figure 6; however, since the tapes must be restored onto disk packs, two additional independent utility programs -- Direct Access Device Initialization (DASDI) and Dump/Restore (DUMPREST) -- are provided

at the beginning of the tapes for use in restoring onto the disk packs. Only those libraries shown in Figure 6 are restored onto the disks from the tapes. The tape layout is shown in Figure 7. The DASDI and Dump/Restore programs will always be at the beginning of the tapes, but the other data sets may not be in the order shown.

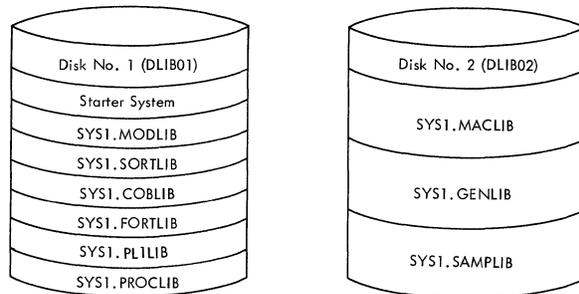


Figure 6. Disk Pack Contents

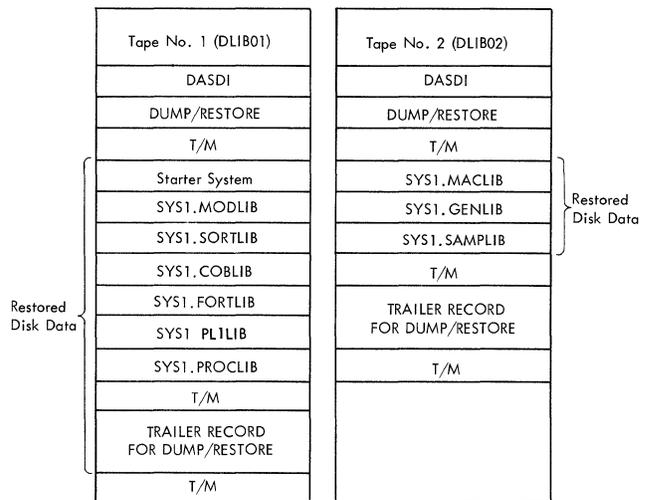


Figure 7. Tape Layout

For illustrative purposes the following procedures for processing the initial package assume the set of devices and device addresses listed in Figure 8.

The first procedure is for readying the tape package; the second procedure is for preparing the disk package with disk backup copy, and the third procedure is for preparing the disk package with tape backup. In the control decks shown the underlined fields are those that may require modification for different installations.

Device Function	Input/Output Device	Address
Printer Keyboard	1052	(01F)
System Residence and	2311 #1	(190)
System Data Sets	2311 #2	(191)
System Input	2540 Reader	(00C)
Punch Output	2540 Punch	(00D)
Print Output	1403 Printer	(00E)
Tape Drives ²	2400 Tape	(180) ¹
	2400 Tape	(181) ¹
	2400 Tape	(182) ¹

¹Tape units located at addresses 180 and 181 are 7-track tape units with data conversion feature. The tape unit at 182 is a 9-track tape unit.

²Tape units are required only if the initial distribution is received on tapes or if system backup will be dumped onto tape.

Figure 8. Sample Configuration

TAPE DISTRIBUTION

Initialize Disk
for
Operating System

1. Mount disk pack onto which starter operating system is to be placed.
2. Mount tape (DLIB01) of initial package.
3. Load DASDI from tape by setting the load selector switches and pressing the console LOAD key. When the program is loaded, the wait state is entered and the hexadecimal value FFFF is displayed in the console lights.
4. The control deck shown below should be placed in the input device to initialize the disk.

```
JOB
MSG      TODEV=1403,TOADDR=00E
DADEF    TODEV=2311,TOADDR=190,VOLID=SCRATCH
VLD      NEW VOLID=111111,OWNERID=DEPT38
VTOCD    STRTADR=50,EXTENT=10
END
```

Define the control statement input device by pressing the REQUEST key of the printer keyboard. The message DEFINE INPUT DEVICE will be printed. Enter the message INPUT=xxxx cuu where xxxx is the device type, c is the channel address and uu is the unit address. Device type can be 1442, 2400 or 2540.

When the disk initialization is complete, the message END OF JOB will be printed on the message output device, and the program will enter the wait state.

Restore Tape
to Disk

5. Load the Dump/Restore program by setting the load selector switches and pressing the control LOAD key. When FFFF is displayed on the console lights, the program is loaded.
6. The control deck shown below should be used to restore the contents of the tape to the disk.

```
JOB
MSG      TODEV=1403,TOADDR=00E
RESTORE  FROMDEV=2400,FROMADDR=180,TODEV=2311,TOADDR=190,VOLID=111111
END
```

Define the control statement input device by pressing the REQUEST key of the printer keyboard. The message DEFINE INPUT DEVICE will be printed. Enter the message INPUT=xxxx cuu where xxxx is the device type, c is the channel address and uu is the unit address. Device type can be 1442, 2400, or 2540.

When the restoring is complete, the message END OF JOB will be printed on the message output device, and the program will enter the wait state.

Initialize
Second Disk

7. Repeat steps 1-4 to initialize the second disk. Be sure that tape #2(DLIB02) is used.

Restore to
Second Disk

8. Repeat steps 5-6 to restore the second tape to disk.

You now have operable disk packs with backup tapes and may proceed to punch the utility programs, any sample programs you may want, and list the system data.

9. Be sure all necessary volumes are mounted and make ready all devices to be used. The control deck shown below should be in the input device.

```
//JOB1      JOB      ACCT123,PROGRAMMER,MSGLEVEL=1
//          EXEC     PGM=IEBPTPCH
//SYSUT1    DD       DSNAME=SYS1.SAMPLIB,DISP=(OLD,KEEP),UNIT=2311, X
//          DD       VOLUME=SER=DLIB02
//SYSUT2    DD       UNIT=2540-2
//SYSPRINT  DD       SYSOUT=A
//SYSIN     DD       *
              PUNCH TYPORG=PO,MAXNAME=4
              MEMBER NAME=DUMPREST
              MEMBER NAME=DASDI
              MEMBER NAME=IEAIPLO0
              MEMBER NAME=RECOVREP
/*
```

Punch Utilities
and Sample
Programs

A MEMBER card may be added to the control deck above for any sample program desired for later use, provided the MAXNAME field in the PUNCH control card is adjusted. The member name card (first card) should be removed from each member deck punched. (Refer to the section, "Sample Programs" for member names of the sample programs.)

10. Set the LOAD UNIT switches on the control panel to the channel, control unit, and device of the system residence volume.
11. Press the LOAD key.
12. Wait for a READY message and the WAIT light to be turned on.
13. After receiving the READY message, enter a SET command specifying the date as follows:
- Press REQUEST key.
 - Wait for READ light to go on.
 - Type SET DATE=yy.ddd
 - Press alternate coding key and the numeric 5 key (EOB).
14. The starter system has built-in START RDR and START WTR commands which will be issued automatically; they are

```
START RDR, 00C
START WTR, 00E
```

If these are not the addresses of the reader and writer to be used, override these commands by entering new ones for the proper devices.

15. Enter a START command with no parameters. When the job is complete a READER CLOSED message will be printed, followed by a READY message, and the system will enter the wait state.

List Data

16. To list the data describing the system, place the control deck below in the input device and enter a START READER command followed by a START command.

Note 1

```
//JOB2      JOB      ACCT123,PROGRAMMER,MSGLEVEL=1
//STEP1     EXEC     PGM=IEHLIST
//SYSPRINT  DD       SYSOUT=A
//DD1       DD       UNIT=2311,VOLUME=SER=DLIB01,DISP=OLD,
```

Note 2

```
//DD2       DD       UNIT=(192,,DEFER),VOLUME=(PRIVATE,RETAIN), X
//          DD       DISP=(NEW,KEEP)
```

Note 3

```
//SYSIN     DD       *
              LISTCTLG
              LISTVTOC DUMP
              LISTVTOC DUMP,VOL=2311=DLIB02
              LISTPDS DSNAME=(SYS1.PROCLIB,SYS1.MODLIB
                          [,Optional PDS's on System Residence])
```

LISTPDS VOL=2311=DLIB02,DSNAME=(SYS1.GENLIB,SYS1.SAMPLIB,
[Optional PDS's on DLIB02])

/*
//

- Note 1: No MOUNT statement should be entered for the drive on which DLIB02 is mounted between IPL time and JOB2. If JOB2 is run immediately after JOB1 (in Step 9) eliminate the JOB2 card.
- Note 2: The physical address in this statement should be replaced with the address of the device containing DLIB02.
- Note 3: In STEP1 any of the PDS's in Figure 6 may be listed in the "LISTPDS."

The steps below detail the procedure for creating a backup copy of the starter operating system on disks.

Punch Utilities 1. To punch the utility programs from the disks (which you will need to prepare a backup copy of the disks), perform steps 9-15 exactly as they are described under "Tape Distribution."

Initialize First Disk Pack 2. Place the DASDI program in the input device with the control deck shown below behind it, and mount the disk pack to be processed.

```
JOB
MSG      TODEV=1403,TOADDR=00E
DADEF    TODEV=2311,TOADDR=190,VOLID=SCRATCH
VLD      NEWVOLID=111111,OWNERID=DEPT38
VTOCD    STRTADR=50,EXTENT=10
END
```

3. Load the DASDI program by setting the load selector switches and pressing the console LOAD key. When the program is loaded, the wait state is entered and the hexadecimal value FFFF is displayed in the console lights.

4. Define the control statement input device by pressing the REQUEST key of the printer keyboard. The message DEFINE INPUT DEVICE will be printed. Enter the following message from the printer keyboard.

```
INPUT=xxxx cuu
```

where xxxx is the device type, c is the channel address and uu is the unit address. The device type can be 1442, 2400, or 2540.

5. At the end of the job, the message END OF JOB will be printed on the message output device, and the program will enter the wait state.

Restore First Disk 6. Place the Dump/Restore program in the input device with the control deck shown below behind it.

```
JOB
MSG      TODEV=1403,TOADDR=00E
DUMP     FROMDEV=2311,FROMADDR=190,TODEV=2311,TOADDR=191,VOLID=111111
END
```

7. Load the program by setting the load selector switches and depressing the console LOAD key. When the program is loaded, the wait state is entered and the hexadecimal value FFFF is displayed in the console lights.

8. Define the control statement input device by pressing the REQUEST key of the printer keyboard. The message DEFINE INPUT DEVICE will be printed. Enter the following message from the printer keyboard.

```
INPUT=xxxx cuu
```

where xxxx is the device type, c is the channel address and uu is the unit address. The device type can be 1442, 2400, or 2540.

9. At the end of the job, the message END OF JOB will be printed on the message output device and the program will enter the wait state.

Initialize Second Disk 10. Repeat steps 2-5 to initialize the second disk.

Restore
Second Disk

11. Repeat steps 6-9 to restore DLIB02.

List Data
Describing
System

12. To list the data describing your system, place the control deck shown below in the input device.

Note 1

```
//JOB2          JOB          ACCT123,PROGRAMMER,MSGLEVEL=1
//STEP1        EXEC         PGM=IEHLIST
//SYSRINT      DD           SYSOUT=A
//DD1          DD           UNIT=2311,VOLUME=SER=DLIB01,DISP=OLD
//DD2          DD           UNIT=(192,,DEFER),VOLUME=(PRIVATE,RETAIN),      X
//             DD           DISP=(NEW,KEEP)
//SYSIN        DD           *
```

Note 2

```
LISTCTLG
LISTVTOC DUMP
LISTVTOC DUMP,VOL=2311=DLIB02
LISTPDS DSNAME=(SYS1.PROCLIB,SYS1.MODLIB
                [,Optional PDS's on System Residence])
LISTPDS VOL=2311=DLIB02,DSNAME=(SYS1.GENLIB,SYS1.SAMPLIB
                [,Optional PDS's on DLIB02])
```

```
/*
//
```

Note 1

The physical address underlined in this statement should be replaced with the address of the device containing DLIB02.

Note 2

In STEP1 any of the PDS's in Figure 6 may be listed in the "LISTPDS".

13. Set the LOAD UNIT switches on the control panel to the channel, control unit, and device of the system residence volume.

14. Press the LOAD key.

15. Wait for a READY message.

16. After receiving the READY message, enter a SET command specifying the date as follows:

- a. Press REQUEST key.
- b. Wait for READ light to go on.
- c. Type SET DATE=yy.ddd
- d. Press alternate coding key and the numeric 5 key (EOB).

17. The starter system has built-in START RDR and START WTR commands which will be issued automatically; they are

```
START RDR, 00C
START WTR, 00E
```

If these are not the addresses of the reader and writer to be used, the operator must override these commands by entering new ones for the proper devices.

18. Enter a START command with no parameters.

The steps below detail the procedure for creating a backup copy of the starter operating system package on tape.

Punch Utilities [1. To punch the utility programs from the disks (which you will need to prepare a backup copy of the disks), perform steps 9-15 exactly as they are described under "Tape Distribution."

Dump Disk to Tape [2. Place the Dump/Restore program in the input device with the control deck shown below behind it.

```
JOB
MSG  TODEV=1403,TOADDR=00E
DUMP FROMDEV=2311,FROMADDR=190,TODEV=2400,POADDR=181,VOLID=111111
END
```

3. Load the disk to be dumped and the tape to receive the backup copy.

4. Load the Dump/Restore program by setting the load selector switches and depressing the console LOAD key. When the program is loaded, the wait state is entered and the hexadecimal value FFFF is displayed in the console lights.

5. Define the control statement input device by pressing the REQUEST key of the printer keyboard. The message DEFINE INPUT DEVICE will be printed. Enter the following message from the console typewriter:

```
INPUT=xxxx cuu
```

where is the device type, c is the channel address and uu is the unit address. The device type can be 1442, 2400, or 2540.

6. At the end of the job, the message END OF JOB will be printed on the message output device, and the program will enter the wait state.

7. To dump the contents of the second disk, perform steps 2-6 above after mounting a second tape and the second disk.

List Data [8. To list the data describing the system, load the operating system, and place the following control deck in the input device. Enter a START command with no parameters.

```
//JOB2      JOB      ACCT123,PROGRAMMER,MSGLEVEL=1
//STEP1     EXEC     PGM=IEHLIST
//SYSPRINT  DD       SYSOUT=A
//DD1       DD       UNIT=2311,VOLUME=SER=DLIB01,DISP=OLD
//DD2       DD       UNIT=(192,,DEFER),VOLUME=(PRIVATE,RETAIN),    X
//          DD       DISP=(NEW,KEEP)
//SYSIN     DD       *
             LISTCTLG
             LISTVTOC DUMP
             LISTVTOC DUMP,VOL=2311,DLIB02
             LISTPDS DSNAME=(SYS1.PROCLIB,SYS1.MODLIB
                        [Optional PDS's on System Residence])
             LISTPDS VOL=2311=DLIB02,DSNAME=(SYS1.GENLIB,SYS1.SAMPLIB,
                        [Optional PDS's on DLIB02])
/*
//
```

DELETING LIBRARIES

In order to make additional direct access storage available for the system generation process, when only two disk drives are attached to the system, you may elect to delete SYS1.SAMPLIB from your operable pack (DLIB02) before starting the system generation process. This may be accomplished by loading the operating system and using the control deck below in the input device.

```
//JOB2      JOB          ACCT123,PROGRAMMER,MSGLEVEL=1
//STEP2     EXEC        PGM=IEHPROGM
//SYSPRINT  DD          SYSOUT=A
//DD1       DD          UNIT=(191,,DEFER),VOLUME=(PRIVATE,RETAIN),DISP=(NEW,KEEP)
//SYSIN     DD          *
            SCRATCH     DSNAME=SYS1.SAMPLIB,VOL=2311=DLIB02,PURGE
/*
```

In order to generate a primary control program with the sequential scheduler, sequential and partitioned access methods, utilities and a selection of the "E" level processors using two 2311 Disk Storage Drives and three volumes, it is essential to scratch SYS1.PL1LIB from DLIB01. This may be accomplished by including the following statements in the above control deck when that job is run.

```
//DD2       DD          UNIT=2311,VOLUME=SER=DLIB01,DISP=OLD
            SCRATCH     DSNAME=SYS1.PL1LIB,PURGE,VOL=2311=DLIB01
```

SECTION II: SAMPLE PROGRAMS

This section contains descriptions of the sample programs provided by IBM to test the functioning of various components of the generated system after system generation. The programs should be punched prior to system generation; the procedure for punching them is detailed in Section I under "Processing the Initial Package." Each sample program writeup contains a description of the program, operating instructions, and a description of the program execution results. More detailed operating procedures can be found in the publication IBM System/360 Operating System: Operator's Guide referred to in the

Preface. Sample programs are contained in SYS1.SAMPLIB on DLIB02 of the starter operating system package; the following sample programs are provided:

<u>Program</u>	<u>Member Name</u>
Assembler E & F Program	IETESP
Update Analysis Program	IHGSAMP
Sort Program	IERSP
FORTRAN E Program	IEJESP
COBOL E Program	IEPSAMP
PL/I F Program	IEMSP
Express Graphics Program	SAMP2250, SAMP2260

ASSEMBLER E&F SAMPLE PROGRAM (IETESP)

DESCRIPTION

The IETESP card deck (punched from SYS1.SAMPLIB) is the sample program source deck.

The sample problem demonstrates the use of the assembler and serves as a minimal test of the functioning of the assembler. In addition, it provides sample coding demonstrating the definition and use of user written macro-instructions, the calling of system macro-instructions, and the proper method of saving and restoring registers upon entry and exit from a problem program.

The input is assembled into the program in the form of a TABLE and a LIST of entries which are to be passed against the table. Each item in the table contains an argument name such as ALPHA and space in which information concerning that name is to be placed. Each entry in the LIST contains an argument name and function values. The formats of the TABLE entries and the LIST entries are different, and both formats are described by means of DSECTS. The program searches the TABLE for an argument name in the list. If a match is found, the function values are reformatted and moved to the appropriate TABLE entry. If an argument name in the LIST cannot be found in the TABLE, a switch is set in the LIST entry. After all LIST entries have been processed, both the LIST and TABLE areas are compared with TESTTABL which contains the predefined results. If the comparison is equal, the routine executed properly and a message is written to indicate this.

Use of the program for Assembler E may be achieved by calling the IBM supplied cataloged procedure ASMECLG in the following manner:

```
//jobname      JOB
//stepname     EXEC      PROC=ASMECLG
//SYSIN       DD        *
Sample Program Source Deck
/*
```

The program for Assembler F may be called by substituting the following execute card in the above deck:

```
//stepname     EXEC      PROC=ASMFCGLG
```

This procedure calls for an assembly, link edit, and execution of the sample program. A more detailed explanation of the contents of the procedure and use of the assembly program is given in IBM System/360 Operating System: Assembler (E) Programmers Guide, Form C28-6595 and in IBM System/360 Operating System: Assembler (F) Programmers Guide, Form C26-3756.

OPERATING INSTRUCTIONS

1. Mount the operating system on a 2311 and an initialized scratch pack.
2. Set the load address switches and press the Initial Program Load key to load the operating system.
3. Place the sample program source deck in card reader, ready reader, and press the End of File key.
4. Ready the printer.
5. Execute the job.

OUTPUT

The check for the successful execution of the sample program is a message on the operator's console. The message says either, "ASSEMBLER SAMPLE PROGRAM SUCCESSFUL" or "ASSEMBLER SAMPLE PROGRAM UNSUCCESSFUL." The program gives a normal return to the operating system with a return code of zero.

DESCRIPTION

The IHGSAMP card deck (punched from SYS1.SAMPLIB) consists of

1. Job control language statements and data to accomplish Stage 1, steps 1, 1A, 2, 3, 4, 5, 6 and 7.
2. Job control language statements to accomplish Stage 2, steps 3, 4, 5 and 6.

This sample program is a two-stage operation requiring two input card decks. The Stage 1 input deck is complete. However, the output created by Stage 1 must be placed in front of the Stage 2 input deck before Stage 2 can be executed (The output of Stage 1 is the job control language statements to accomplish Stage 2, steps 1 and 2.)

In order to demonstrate the functions of Update Analysis, it is necessary to create some data sets on a disk pack of the operating system. However, the sample program has been constructed so that at its completion all data sets created by it are deleted. Therefore, the program can be run without leaving residue from its operations on the system disk packs. The printer output acts as a graphic demonstration of the changing programs.

Stage 1 builds two sample symbolic libraries and one sample change library. It then executes the Update Analysis Program which creates the job stream required to update the sample libraries. Stage 2 utilizes the output of the Update Analysis to update the two sample libraries, then proceeds to eliminate the data sets from the system. This is described in the section "Output."

OPERATING INSTRUCTIONS

1. Mount the operating system on a 2311 and initialized scratch pack.
2. Set the load address switches and press the Initial Program Load key to load the operating system.
3. Place Stage 1 card deck in reader and ready the reader, printer, and punch.

4. Execute the Stage 1 job stream.
5. At the end of the job, take the output of Stage 1 from the punch stacker and place in front of Stage 2 deck.
6. Place Stage 2 deck in reader and ready reader.
7. Issue commands:
START RDR, 00C
START
to execute the Stage 2 job stream.

OUTPUT

Stage 1

- Step 1 EXEC IEBUPDAT
This step creates a temporary data set that contains the SYSIN control cards for step 1A.
- Step 1A EXEC IEBGENER
This step creates the PDS SAMPLCHG which is the change PDS containing three change members to be applied to the two sample libraries.
- Step 2 EXEC IEBUPDAT
This step creates the PDS SAMPLIB1 which is a sample system library containing two members: MEMBER1 and MEMBER2.
- Step 3 EXEC IEBUPDAT
This step creates the PDS SAMPLIB2 which is a sample system library containing one member: MEMBER3.
- Step 4 EXEC IEHLIST
This step lists the directories of the two sample libraries. The list will show the SSI bytes of the members prior to updating.
- Step 5 EXEC IHGUAP
This step causes the SAMPLCHG PDS to be analyzed and creates the job stream which will update the sample system libraries. For the purpose of the sample problem, this output is temporarily stored on disk to be available for steps 6 and 7.

Step 6	EXEC IEBTPCH This will print the output of step 5.		of the two sample libraries. The list will show the SSI bytes of the members after they have been updated.
Step 7	EXEC IEBTPCH This will punch the output of step 5.	Step 4	EXEC IEBUPDAT This reproduces SAMPLIB1 in its updated form. This operation creates a printed list which shows the effect of the change.
<u>Stage 2</u>		Step 5	EXEC IEBUPDAT This reproduces SAMPLIB2 in its updated form. It creates a printed list which shows the effect of the change.
Step 1 and 2	(Using punched output from step 7 of Stage 1) EXEC IEBUPDAT Update SAMPLIB1 EXEC IEBUPDAT Update SAMPLIB2	Step 6	EXEC IEHPRGM This causes the deletion of data sets SAMPLCHG, SAMPLIB1 and SAMPLIB2.
Step 3	EXEC IEHLIST This step lists the directories		

DESCRIPTION

The IERSP card deck (punched from SYS1.SAMPLIB) consists of:

1. Job control language statements for TAPESORT.
2. Job control language statements for DISKSORT.
3. Data to be sorted.

The data to be sorted consists of 500 80-character records, each containing a 6-digit sequence number and a 10-character control field. As provided, the data deck is in the 6-digit sequence number order. The SORT control card specifies that the data is to be sorted on the 10-character control field. The output of either a TAPESORT or a DISKSORT will be in 10-character control field sequence.

The TAPESORT job control language statements provided in IERSP call for four tape units (three for work and one for SORTOUT). To use more than three work tapes, additional cards may be added behind the SORTWK03 DD card. They should be identical to other SORTWK DD cards except for DD name which must be consecutive, i.e., SORTWK04, SORTWK05....SORTWK32.

The DISKSORT job control language statements provided in IERSP call for six work areas of 30 contiguous tracks each on a 2311.

OPERATING INSTRUCTIONS

1. Mount the operating system on a 2311 and an initialized scratch pack.

2. Set the load address switches and press the Initial Program Load key to load the operating system.

3. For a TAPESORT, the data deck should be placed on tape as one unblocked sequential data set. An operator message (during step #7 below) will give mounting instructions.

For a DISKSORT, the data deck should be placed on the scratch pack (mounted in step #1 above) as one unblocked sequential data set and cataloged under the name TEST.

4. For a TAPESORT, place the TAPESORT job control language statements in the card reader.

For a DISKSORT, place the DISKSORT job control language statements in the card reader.

5. Ready the reader and press the End of File key.
6. Ready the printer.
7. Execute the job.

OUTPUT

A message to the operator will indicate the unit assigned to SORTOUT (unit containing output). This output can be printed and reviewed. The records will be in 10-character control field sequence.

Output of TAPESORT will be on tape.

Output of DISKSORT will be on disk.

FORTRAN E SAMPLE PROGRAM (IEJESP)

DESCRIPTION

The IEJESP card deck (punched from SYS1.SAMPLIB) consists of:

1. Job control language statements for a FORTRAN compilation.
2. FORTRAN sample program source statements.
3. Job control language statements for a link edit and execution of the sample program.
4. Data deck for FORTRAN sample program execution.

The sample program is a simultaneous equations routine which consists of 104 FORTRAN source statements:

1	Specification
14	Format
17	Read/Write
27	Control
16	Arithmetic
29	Comments

The program processes eight data cards as its input. The comments cards show a complete list of the expected output from the execution of the program. This list may be used for checking output.

OPERATING INSTRUCTIONS

1. Mount the operating system on a 2311 and an initialized scratch pack.
2. Set the load address switches and press the Initial Program Load key to load the operating system.

3. Place the IEJESP card deck in the card reader, ready reader, and press the End of File key.
4. Ready the printer.
5. Execute the job.

OUTPUT

1. The scheduler will read, process, and print (on the device specified as SYSOUT by the operator) all job control statements.
2. The compiler will read and list the source program on the device specified as SYSOUT by the operator.
3. The compiler will prepare and list (on SYSOUT) the following:
 - a. Message showing compiler options in effect
 - b. Heading including date and level
 - c. The source program
 - d. Storage map including size of COMMON and size of program
 - e. Message showing "END OF COMPILATION program-name"
4. The linkage editor will prepare and list (on SYSOUT) a storage map including the relative address of each external reference.
5. The FORTRAN load module will list (on SYSOUT) the results of execution. The results should correspond to the comments in the source program.

DESCRIPTION

The IEPSAMP card deck (punched from SYS1.SAMPLIB) consists of:

1. Job control language statements to execute the COBOL E cataloged procedure COBECLG to compile, link edit, and execute.
2. COBOL sample program source statements.
3. DD statements for tape data sets required at execution time.

The sample program tests the COBOL compiler's ability to WRITE to and READ from tape, checking proper data alignment where data is a mixture of DISPLAY COMPUTATIONAL and COMPUTATIONAL-3.

The verbs exercised in this program are OPEN, CLOSE, READ, WRITE, PERFORM, IF, MOVE, GO TO, DISPLAY, ADD. There are 575 source statements.

OPERATING INSTRUCTIONS

1. Mount the operating system on a 2311 and an initialized scratch pack.
2. Mount unlabeled tapes on 182, 183, and 282.
3. Set the load address switches and press the Initial Program Load key to load the operating system.

4. Place the IEPSAMP card deck in the card reader, ready reader, and press End of File key.
5. Ready the printer.
6. Execute the job.

OUTPUT

1. The scheduler will read, process, and print (on the device specified as SYSOUT by the operator) all job control language statements.
2. The compiler will read and list the source program on the device specified as SYSOUT by the operator.
3. The compiler will prepare and list (on SYSOUT)
 - a. Heading including date and level
 - b. The source program
 - c. A data map of the Data Division
 - d. An object code listing (PMAP) of the Procedure Division
4. The linkage editor will prepare and list (on SYSOUT).
 - a. A module map
 - b. A cross-reference list
5. The COBOL load module will list (on SYSOUT) the following:

GROUP B LEVEL P TEST CASE 1
END OF PROGRAM

PL/1(F) SAMPLE PROGRAM (IEMSP)

DESCRIPTION

The IEMSP card deck consists of

- 1. Job control language statements for a PL/1 compilation, link edit, and execution of the compiled program. The cataloged procedure PL1LFCLG is used.
- 2. PL/1 sample program source statements.

The purpose of the sample program is to illustrate the use of list, edit and data directed output. There are 20 source statements. They include PROCEDURE, DECLARE, DO, END, PUT LIST, PUT EDIT, and PUT DATA.

No data input is required.

The sample program is described in detail in IBM System/360 Operating System PL/1(F) Programmers Guide, Form C28-6594.

OPERATING PROCEDURES

- 1. Mount the operating system pack on a 2311.

- 2. Set the load address switches and press the Initial Program Load key to load the operating system.
- 3. Place the IEMSP card deck in the card reader, ready the reader, and press End of File.
- 4. Ready printer.
- 5. Execute job.

OUTPUT

- 1. Normal compilation output.
 - a. All job control language statements in the catalogued procedure.
 - b. All source statements.
 - c. An attribute and cross-reference listing.
 - d. A listing of generated object code.
 - e. A list of errors found during the compilation.
- 2. Normal link edit step output.
- 3. The correct generated output is shown in Figure 9.

EXAMPLES OF LIST/DATA/EDIT OUTPUT

LIST DIRECTED EXAMPLES:

12345 1.23450E+04 1.23450E+04+1.23450E+04I '1100111000'B ABC'DEFG

DATA DIRECTED EXAMPLES:

A= 12345 B= 1.23450E+04 C= 1.23450E+04+1.23450E+04I D='1100111000'B E='ABC'DEFG ';
ARRAY(1,1)= 2.00000E+00 ARRAY(1,2)= 3.00000E+00 ARRAY(2,1)= 3.00000E+00 ARRAY(2,2)= 4.00000E+00;

EDIT DIRECTED EXAMPLES:

12345.00 12345E+00 12345E+00 12345E+00 1100111000ABC'DEFG

Figure 9. PL/I Generated Output

DESCRIPTION

The SAMP2250 card (sample program to exercise the 2250 Display Unit) and the SAMP2260 card deck (sample Program to exercise the 2260 Display Station, local) can be punched from SYS1.SAMPLIB.

Each card deck consists of:

1. Appropriate job control language statements which will call a cataloged procedure (ASMECLG) to assemble, linkage edit, and execute the compiled program. Before compiling either deck, the four over-ride cards below must be added to the end of the deck after the assembler language END card.

/*

```
//LKED.SYSLIB DD DSNAME=SYS1.LINKLIB,X  
                DISP=(OLD,KEEP),    X  
                UNIT=2311
```

```
//GO.GRAPHIC DD UNIT=2250
```

/*

2. Sample program input symbolic deck.

OPERATING INSTRUCTIONS

1. Insure that the graphic device is ON.
2. Place the sample program deck in the card reader.
3. Mount the operating system on a 2311.
4. Set the load address switches and press the Initial Program Load key to load the system.
5. Perform assemble-link-edit-go.
6. Follow instructions which appear on your display screen.

OUTPUT

The displays which appear on the 2250 Display Unit are shown on Figure 10. The sample program for the 2260 Display Station is shown in Figure 11.

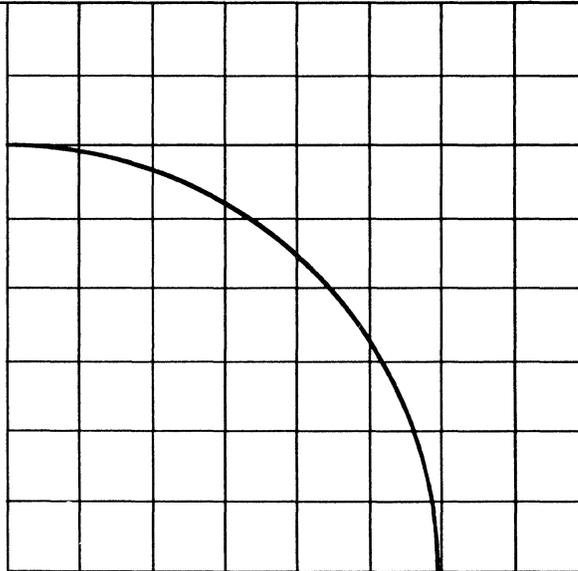
2250 SAMPLE PROGRAM - END ATTENTION TO BEGIN

a. First Display

THIS PROGRAM WILL PLOT A GRID WITH CORNERS AT 600,600-600,1000-1000,1000-1000,600. IT WILL PLOT 90 DEGREES OF ARC WHOSE RADIUS IS 300 RASTER UNITS. IT ILLUSTRATES THE USE OF ORDER MACROS TO CREATE THIS DISPLAY AND PORS TO CREATE THE NEXT DISPLAY.

DEPRESS KEY D AND THEN THE END KEY ON THE ALPHAMERIC KEYBOARD TO INITIATE THE NEXT DISPLAY.

b. Second Display



THIS DISPLAY ILLUSTRATES THE OUTPUT OF THE GSTOR, GCGRID, GARC, AND GCPRNT PROBLEM-ORIENTED ROUTINES.

DEPRESS KEY E AND THEN THE END KEY ON THE ALPHAMERIC KEYBOARD TO TERMINATE THIS SAMPLE PROGRAM.

c. Third Display

Figure 10. 2250 Displays

IBM

IBM System/360 Assembler Coding Form

X28-6509-2
Printed in U.S.A.

PROGRAM		2260 SAMPLE PROBLEM		PUNCHING INSTRUCTIONS		GRAPHIC		PAGE 1 OF 4	
PROGRAMMER		DATE		PUNCH				CARD ELECTRO NUMBER	
STATEMENT									
Name	Operation	Operand	Comments	Identification-Sequence					
SAMPLE	CSECT	5,0	INITIALIZE BASE REGISTER						
	BALR	*5	DEFINE BASE REGISTER						
	USING	*5							
	ST	13,SAVE	SAVE REGISTER 13						
	ST	14,SAVE+4	SAVE RETURN ADDR TO SUPERVISOR						
	LA	13,SAVEREA	LOAD ADDRESS OF REGISTER SAVE						
*			AREA IN PROBLEM PROGRAM						
	OPEN	(DCB1)	OPEN 2260						
	GWRITE	DECB1,EBW,DCB1,43,OPNMSG	WRITE OPEN MSG TO OPERATOR						
	WAIT	ECB=DECB1							
TIMING	ANALYZ	POLST,POINTR	POLL FOR ATTENTION						
*									
*	DETERMINE IF AN ATTENTION OCCURRED								
*									
	B	TIMING							
*									
*	KEYBOARD ROUTINE TO SERVICE KEYBOARD ATTENTIONS								
*									
KYBDRTN	SAVE	(14,12)	SAVE MAIN P-P REGISTERS						
	ST	13,SAVAR2+8							
	LA	13,SAVAR2							
	GREAD	DECB2,DSM,DCB1,,INPUT	READ DS MANUAL INPUT						
	WAIT	ECB=DECB2	WAIT FOR I-O OPERATION						
*			I-O COMPLETE						

IBM

IBM System/360 Assembler Coding Form

X28-6509-2
Printed in U.S.A.

PROGRAM		2260 SAMPLE PROBLEM		PUNCHING INSTRUCTIONS		GRAPHIC		PAGE 2 OF 4	
PROGRAMMER		DATE		PUNCH				CARD ELECTRO NUMBER	
STATEMENT									
Name	Operation	Operand	Comments	Identification-Sequence					
*	CLC	INPUT(4),PRC	COMPARE FOR CONTINUE CODE						
*			IN FIRST 4 BYTES OF MSG						
	BC	8,CONTIN	IF YES - CONTINUE						
*	GWRITE	DECB3,EBW,DCB1,10,TERMSG	IF NO - ERASE AND WRITE						
			TERMSG						
	WAIT	ECB=DECB3	WAIT FOR I-O OPERATION						
	CLOSE	(DCB1)	CLOSE 2260						
	L	13,SAVE	RESTORE REGISTER 13						
	L	14,SAVE+4	RESTORE REGISTER 14						
	RETURN	RC=0	TERMINATE TASK AND RETURN TO						
*			SUPERVISOR						
*									
*	PROCESSING MANUAL INPUT REQUEST IS DONE IN CONTIN ROUTINE BELOW								
*									
CONTIN	BCR	0,0	DUMMY INSTRUCTION - USER						
*			PROCESSING ROUTINE WOULD						
*			GO HERE						
	GWRITE	DECB4,DSB,DCB1,44,PRCMSG	WRITE PROCESSING COMPLETE MSG						
	WAIT	ECB=DECB4	WAIT FOR I-O OPERATION						
	L	13,SAVAR2+8	RESTORE REGISTER 13						
	RETURN	(14,12),RC=0	RETURN TO POLLING						

Figure 11. 2260 Sample Program

PROGRAM		2260 SAMPLE PROBLEM		PUNCHING INSTRUCTIONS		GRAPHIC		PAGE 3 OF 4	
PROGRAMMER		DATE		PUNCH				CARD ELECTRO NUMBER	
Name	Operation	Operand		Comments		Identification-Sequence			
*CONSTANT	AREA								
DCB1	DCB	DSORG=GS,MACRF=(RC,WC),DDNAME=SAMPLEPG,GNCP=4,							*
		POLST=POLST							
	DS	0F							
INPUT	DC	240F'0'		DEFINE INPUT AREA					
PRC	DC	CL4'PROC'		PROCESS ACTION CODE					
POLST	DC	2F'0'		POLST CONSTRUCTED BY OPEN					
*				RESERVE NUMBER OF OPENED					
*				DEVICES+1 FULL WORDS					
POINTR	DC	1F'0'		POINTER TABLE FOR ANALYZ					
	DC	A(KYBDRTN)		ADDR OF KEYBOARD ROUTINE					
	DC	1F'0'		ZERO					
	DC	1F'0'		ZERO					
	DC	A(OUTPUT)		ADDR OF OUTPUT AREA					
OUTPUT	DC	2F'0'		RESERVE TWO WORDS FOR					
*				OUTPUT AREA					
SAVE	DC	2F'0'		SAVE AREA FOR SUPV REGISTERS					
SAVEAREA	DC	18F'0'		P-P REGISTER SAVE AREA					
SAVAR2	DC	18F'0'							

PROGRAM		2260 SAMPLE PROBLEM		PUNCHING INSTRUCTIONS		GRAPHIC		PAGE 4 OF 4	
PROGRAMMER		DATE		PUNCH				CARD ELECTRO NUMBER	
Name	Operation	Operand		Comments		Identification-Sequence			
*MESSAGE	AREA								
OPNMSG	DC	CL43'DISPLAY STATION OPEN - INSERT DS MI REQUEST'							
TERMSG	DC	CL10'END OF JOB'							
PRCMSG	DC	CL44'PROCESSING COMPLETE - INSERT NEXT MI REQUEST'							
*									
*SECONDARY PROGRAMS									
*									
*GRAPHIC	ATTENTION ANALYSIS ROUTINE								
*									
*GRAPHIC	I-O CONTROL ROUTINE								
*									
	END								

Figure 11. 2260 Sample Program (Continued)

REPORT PROGRAM GENERATOR SAMPLE PROGRAM (RPGSMPL)

DESCRIPTION

The RPGSMPL card deck (punched from SYS1.SAMPLIB) consists of

1. Job control language statements to call a cataloged procedure for a compilation, link-edit, and execution of the sample program.
2. Report Program Generator source statements; there are 45 of these.
3. Data cards for the program to process; there are 13 of these, and no other input is necessary.

The program processes the 13 data cards which contain details of customer transactions and prepares the report shown in Figure 12. A detailed description of the source program appears in IBM Operating System/360: Report Program Generator Language, Form C24-3337.

OPERATING INSTRUCTIONS

1. Mount the operating system on a 2311 and an initialized scratch pack.
2. Set the load address switches and press the Initial Program Load key to load the operating system.
3. Place the RPGSMPL card deck in the card reader, ready the reader, and press the End of File key.
4. Ready the printer.
5. Execute the job.

OUTPUT

Program output will consist of a printed listing of the source program and the printed report shown in Figure 12.

A C C O U N T S R E C E I V A B L E R E G I S T E R							
CUSTOMER NUMBER	CUSTOMER NAME	LOCATION STATE	CITY	INVOICE NUMBER	INVOICE NO.	DATE DAY	INVOICE AMOUNT
10712	AMALGAMATED CORP	33	61	11603	11	10	\$ 389.25 \$ 389.25*
11315	BROWN WHOLESALE	30	231	12324	12	28	\$ 802.08
11315	BROWN WHOLESALE	30	231	99588	12	14	\$ 261.17 \$ 1,063.25*
11897	FARM IMPLEMENTS	47	77	10901	10	18	\$ 27.63 \$ 27.63*
18530	BLACK OIL	16	67	11509	11	8	\$ 592.95
18530	BLACK OIL	16	67	12292	12	23	\$ 950.97 \$ 1,543.92*
20716	LEATHER BELT CO	36	471	11511	11	8	\$ 335.63
20716	LEATHER BELT CO	36	471	12263	12	17	\$ 121.75 \$ 457.38*
29017	GENERAL MFG CO	6	63	11615	11	14	\$ 440.12
29017	GENERAL MFG CO	6	63	11676	11	23	\$ 722.22 \$ 1,162.34*
29054	A-B-C DIST CO	25	39	9689	9	11	\$ 645.40
29054	A-B-C DIST CO	25	39	11605	11	11	\$ 271.69
29054	A-B-C DIST CO	25	39	12234	12	14	\$ 559.33 \$ 1,476.42* \$ 6,120.19**

Figure 12. RPG Sample Program Printed Report

Arrangement of data sets on tape and disk	9	Initial Program Load (IEAIPL00)	5
Assembler sample program	19	Initialization, description of	5
Backup copy of Starter System	5	Initialization, disk with disk backup	14
COBOL sample program	24	Initialization, disk with tape backup	16
Configuration requirements	6	Initialization, results of	5
Deleting libraries	17	Initialization, tape	11
Device readiness at IPL time	8	I/O requirements	3
Direct Access Device Initialization (DASDI)	5	Libraries	8,9
Disk contents	9	Libraries, deletion of	17
Dump/Restore (DUMPREST)	5	Operating System contents	5
Express Graphics sample program	26	PL/I F sample program	25
FORTRAN sample program	23	Recover/Replace (RECOVREP)	5
Graphics (express) sample programs	26	SAMP2250	26
IEJESP	23	SAMP2260	26
IEMSP	25	Sample configuration	8
IEPSAMP	24	Sample programs, introduction	18
IERSP	20	Sort sample program	22
IETESP	19	Starter Operating System package, description	5
IHGSAMP	20	System generation	6
Independent utility programs	5	Tape contents	9
Initial package, description of	5-9	Update Analysis sample program	20
Initial package, processing	10		

File Number S360-20
Re: Form No. C28-6630-0
This Newsletter No. N28-2192
Date October 24, 1966
Previous Newsletter Nos. None

IBM SYSTEM/360 OPERATING SYSTEM
STARTER OPERATING SYSTEM GUIDE

This technical newsletter amends the publication IBM System/360 Operating System: Starter Operating System Guide, Form C28-6630-0. The attached pages, 30A and 30B should be inserted in the publication.

Summary of Amendments

The amendment to this publication consists of a new sample program: Report Program Generator Sample Program.

Note: Please file this cover letter at the back of the publication. Cover letters provide a quick reference to changes and a means of checking receipt of all amendments.

READER'S COMMENTS

Title: IBM System/360 Operating System
Starter Operating System Guide

Form: C28-6630-0

Is the material:	Yes	No
Easy to Read?	___	___
Well organized?	___	___
Complete?	___	___
Well illustrated?	___	___
Accurate?	___	___
Suitable for its intended audience?	___	___

How did you use this publication?

___ As an introduction to the subject

___ For additional knowledge

Other _____

fold

Please check the items that describe your position:

___ Customer personnel

___ Operator

___ Sales Representative

___ IBM personnel

___ Programmer

___ Systems Engineer

___ Manager

___ Customer Engineer

___ Trainee

___ Systems Analyst

___ Instructor

Other _____

Please check specific criticism(s), give page number(s), and explain below:

___ Clarification on page(s)

___ Addition on page(s)

___ Deletion on page(s)

___ Error on page(s)

Explanation:

fold

CUT ALONG LINE

staple

staple

fold

fold

<p>FIRST CLASS PERMIT NO. 81 POUGHKEEPSIE, N.Y.</p>

<p>BUSINESS REPLY MAIL NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.</p>
--

POSTAGE WILL BE PAID BY
 IBM CORPORATION
 P.O. BOX 390
 POUGHKEEPSIE, N. Y. 12602

ATTN: PROGRAMMING SYSTEMS PUBLICATIONS
 DEPT. D58

fold

fold



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

IBM World Trade Corporation
 821 United Nations Plaza, New York, New York 10017
 [International]

Printed in U.S.A.
 C28-6630-0

staple



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

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]