3100 3200 3300 3500

MSOS

CONTROL DATA CORPORATION

INSTALLATION HANDBOOK

3100
3200
3300
3500

MSOS

CONTROL DATA
CORPORATION

INSTALLATION HANDBOOK

REVISION RECORD

REVISION	DESCRIPTION
12/10/69	Original Printing
A	
2/24/70	Revision packet for USASI FORTRAN add on and miscellaneous corrections.
B	
5/15/70	Revision packet for MMTC driver and ASCII support add ons.
C	
12/17/70	This manual obsoletes previous editions.
D	
5/10/71	Revision packet for enhanced magnetic tape recovery update.
Publication No. 60280700D	

Address comments concerning this manual to:
Control Data Corporation
Technical Publications
4201 North Lexington Ave.
St. Paul, Minnesota 55112
or use Comment Sheet in the back of this manual.

CONTENTS

PART I

RELEASE SUMMARY

PART II

INSTALLATION PROCEDURES

PART III

RELEASE AIDS

PART I

RELEASE SUMMARY

SYSTEM SUMMARY 1

1.1 PRODUCT SET MEMBERS

Version 4.2 of the Mass Storage Operating System (MSOS) is accompanied by the following product set members:

COMPASS version 3.2

Mass Storage FORTRAN version 4.2

COSY version 3.2

USASI COBOL version 2.1

Mass Storage (BDP) COBOL version 4.2

Mass Storage (BCD) COBOL version 4.2

MSIO version 4.2

BSIPP version 2.5

SIPP version 2.5

Mass Storage SORT version 3.2

ALGOL version 1.4

Standard File Processor version 1.3

RESPOND version 1.3

ADAPT version 1.1 (32K variant)

ADAPT version 1.2 (16K variant)

PERT TIME version 2.1 (32K variant)

PERT TIME version 2.2 (16K variant)

PERT COST version 2.1

SAINT version 2.3

Tape SORT/MERGE version 2.2

Error Recovery version 1.3

Linked Index Sequential Access version 1.2

MSOS Utility version 1.3

MSOS USASI FORTRAN version 1.1

1.2 RELEASE MATERIALS

The release materials for MSOS 4.2 and its product set consist of the following packages:

1.2.1 PACKAGE A

Package A contains the binary source, the COSY source, and the list file for the MSOS 4.2 Operating System routines, the library generation routine, the utility routines and the following product set members.

COMPASS
MSOS UTILITY
COSY
ERROR RECOVERY
L-MSIO
MASS STORAGE FORTRAN
MASS STORAGE COBOL (BDP)
MASS STORAGE COBOL (BCD)
MASS STORAGE SORT
ALGOL
STANDARD FILE PROCESSOR
RESPOND
SAINT
SIPP
BSIPP
LISA
TAPE SORT/MERGE

1.2.2 PACKAGE B

Package B contains overlay preparation deck and the BCD source file for PERT TIME in a 16K MSOS environment.

1.2.3 PACKAGE C

Package C contains the overlay preparation deck and the BCD source file for PERT TIME in a 32K MSOS environment.

1.2.4 PACKAGE D

Package D contains the overlay preparation deck and the BCD source file for PERT COST.

1.2.5 PACKAGE E

Package E contains the overlay preparation deck for ADAPT in a 16K MSOS environment.

1.2.6 PACKAGE F

Package F contains the overlay preparation deck for ADAPT in a 32K MSOS environment.

1.2.7 PACKAGE G

Package G contains the binary source file, COSY source file, and list file for USASI FORTRAN.

1.2.8 PACKAGE H

Package H contains the binary source file, COSY source file, and list file for USASI COBOL.

1.3 RELEASE MATERIAL DESCRIPTION

1.3.1 PACKAGE A MSOS 4.2

Release package A contains the following tapes:

A1 - Binary release †

IUP (Installation Utility Package)
853, 854, 841, 863 INTERIM LIBRARY
BINARY SOURCE FILE

A2 - COSY release tapes ††

MSOS RESIDENT VARIABLE RESIDENT UTILITY ROUTINES	LISA ROUTINES
PRELIB ROUTINES	MSOS UTILITY ROUTINES
ERROR RECOVERY ROUTINES	PLIBEDIT
COSY ROUTINES	LOGICAL MSIO ROUTINES
COMPASS ROUTINES	BCD COBOL ROUTINES
FORTRAN ROUTINES	BDP COBOL ROUTINES
SAINT ROUTINES	MSSORT ROUTINES
RESPOND ROUTINES	TAPE SORT ROUTINES
	ALGOL ROUTINES

A3 - List Tapes

† IUP, the Interim Library, and the Binary source file are seprated by EOF marks.
†† Refer to Table II-3 for a complete list of the COSY decks on the COSY Release Tapes.

1.3.2 PACKAGE B PERT TIME - 16K

Release package B consists of the following tape:

OVERLAY PREPARATION DECKS
HOLLERITH SOURCE

1.3.3 PACKAGE C PERT TIME - 32K

Release package C consists of the following tape:

OVERLAY PREPARATION DECKS
HOLLERITH SOURCE

1.3.4 PACKAGE D PERT COST

Release package D consists of the following tape:

OVERLAY PREPARATION DECKS
HOLLERITH SOURCE

1.3.5 PACKAGE E ADAPT - 16K

Release package E consists of the following tape:

OVERLAY PREPARATION DECKS

1.3.6 PACKAGE F ADAPT - 32K

Release package F consists of the following tape:

OVERLAY PREPARATION DECK

1.3.7 PACKAGE G USASI FORTRAN

Release package G consists of the following tape:

BINARY SOURCE FILE
COSY FORMATTED SOURCE
LIST FILE

1.3.8 PACKAGE H USASI COBOL

Release package H consists of the following tape:

BINARY SOURCE FILE
COSY FORMATTED SOURCE
LIST FILE

1.4 NEW FEATURES

1.4.1 MSOS 4.2

- Added a driver for 841 disk packs
- Installation procedures for the 841 disk pack driver
- Use of all functions such as OPEN, CLOSE, ALLOCATE, RELEASE, MODIFY, EXPAND, ENTER, DELETE, LIST, and BAD TRACK for the 841 disk pack
- Multiple editions of the MSOS system on mass storage
- Auxiliary library capability
- Ability to dump all of core memory with RDUMP
- Printing of character address as well as word address on dump produced by RDUMP
- Scans control cards only to column 72
- Use of date on IDC cards of Binary Decks
- Assembly option to set the default value for the device type in an ALLOCATE request
- Use by variable resident of manual interrupt control
- Enhanced Mass Storage UTILITY routines to improve throughput and reliability
- Enhanced Installation Utility Package (IUP) which replaces BOOT
- AET macro
- POSITION control card under PLIBEDIT

1.4.2 COMPASS 3.2

- Relocatable error recovery
- Date punched on IDC card of binary decks

1.4.3 MS FORTRAN 4.2

- Scans only to column 72 on control cards
- Date punched on IDC card of binary decks
- Option to selectively terminate job if errors occur in BCD input
- Focus updates and corrections to MS FORTRAN object-time routines

1.4.4 COSY 3.2

- Scans only to column 72 on control cards
- Compute size of available common

1.4.5 USASI COBOL 2.1

- Date punched in column 51 on IDC card of binary decks
- USASI COBOL source card sequence checked

1.4.6 MS COBOL

- Detects zero or negative subscript values during object program execution
- Provides expanded error message for subscript errors (A GPIO 084)
- Changes source program listing to tie source code to memory dump through the following features:

  No compiler generated sequence numbers

  Relative addresses of DATA DIVISION entries are always given.

  Compiler generated sequence numbers for PROCEDURE DIVISION are replaced by relative address of object code generated by statement on a given line.

  DATA DIVISION COPY MACRO's are no longer restricted to the names COBOLFD and COBOLREC. This feature facilitates adding, deleting, and changing COPY MACRO's on the library and speeds COPY processing by eliminating the need for a sequential search of the library to find the requested MACRO.

1.4.7 L-MSIO 4.2

- Uses four overlays to reduce core requirements by 2400_{10}. A mass storage file is used for overlay.

1.4.8 TAPE SORT/MERGE 2.2

- Diagnoses an attempt by the operator to restart when a restart dump has been specified.

1.4.9 USASI FORTRAN 1.1

- Date punched on IDC card of binary decks

1.5 MODIFICATIONS AND CORRECTIONS

1.5.1 MSOS 4.2

This release contains corrections for the following PSR numbers:

Summary	PSR Number	Summary	PSR Number	Summary	PSR Number	Summary	PSR Number
147	4252	152	3745	167	4970	172	5189
167	4317	144	4058	158	3358	172	4684
154	4426	144	3864	159	4629	179	5178
162	4219	144	3546	156	4432	181	5138
144	3815	169	3857	156	3879	180	5030
146	3804	144	3535	156	4067	172	5163
156	4205	157	4419	146	3586	181	3546
144	4134	154	4287	166	4649	170	5039
149	3747	149	4324	169	5016	175	5307
168	3488	149	4229	144	4141	178	4758
145	4190	157	4277	149	3958	179	5456
159	4610	161	4506	162	4579	179	5178
144	4135	167	5027	147	4115	179	5138
151	4078	144	3549	169	5073	177	5030
144	3922	144	3692	171	4527	179	5456
171	4318	169	4914	175	5207		
144	3698	144	3776	178	4758		
144	3921	163	3691	179	5456		

This release contains modifications for the following RSM's:

1749	A032	A053	A056
A029	A039	A054	A067

1.5.2 COMPASS 3.2

This release contains corrections and modifications for the following PSR and RSM numbers:

Summary	PSR Number
173	4353
173	3942
180	5034

RSM: A053

1.5.3 MS FORTRAN 4.2

This release contains corrections and modifications for the following PSR and RSM numbers:

Summary	PSR Number	Summary	PSR Number
166	3640	149	4302
149	3940	149	4309
144	4125	152	4322
144	4126	155	4416
149	4166	155	4424
147	4193	169	4503
147	4214	159	4535
152	4267	158	4557
		178	5389

RSM:	A039	A149
	A053	A251
	A147	

1.5.4 COSY 3.2

This release contains corrections and modifications for the following PSR and RSM numbers:

Summary	PSR Number	Summary	PSR Number
171	4616	162	4699
171	4615	162	4638
173	4207	182	5376

RSM: A039

1.5.5 USASI COBOL 2.1

This release contains corrections for the following PSR and RSM numbers:

Summary	PSR Number	Summary	PSR Number	Summary	PSR Number	Summary	PSR Number
144	3749	145	3892	146	3960	170	5047
151	3961	144	3963	144	3965	171	4670
144	4016	144	4084	144	4085	172	5049
144	4094	144	4095	164	4096	172	5123
144	4159	144	4160	144	4161	173	4852
144	4162	147	4171	144	4175	173	5222
144	4176	144	4177	144	4178	173	5128
144	4179	144	4180	144	4181	174	5190
155	4200	149	4223	153	4300	174	5242
149	4303	152	4342	153	4362	175	5087
152	4363	152	4364	155	4382	175	4671
156	4383	169	4396	153	4413	179	4788
153	4414	172	4415	174	4458	180	5050
164	4531	163	4553	159	4588	180	5167
165	4645	168	4664	162	4689	180	4885
160	4691	161	4692	162	4694	180	5011
164	4716	163	4738	169	4793	180	5427
165	4798	163	4824	163	4828	180	4458
165	4840	168	4854	168	4856	181	5384
164	4877	164	4885	165	4893	182	4943
168	4908	165	4910	165	4913	183	5632
166	4944	168	5007	167	5013	185	4944
169	5062	169	5063	170	4855		

RSM: A053

1.5.6 MASS STORAGE COBOL 4.2

This release contains corrections and modifications for the following PSR and RSM numbers:

Summary	PSR Number	Summary	PSR Number	Summary	PSR Number	Summary	PSR Number
153	3775	145	4187	152	4390	178	5184
156	4174	151	4379	169	4470	181	5283
147	4364	156	4452	165	4821	178	5371
153	4392	163	4737	178	4174	178	5428
162	4655	169	5085	178	4178		
166	4880	147	4149	178	4880		
145	3954	146	4224	177	5183		

RSM: A041 A042
A043 A046

1.5.7 LOGICAL MSIO 4.2

This release contains corrections for the following PSR numbers:

Summary	PSR Number	Summary	PSR Number	Summary	PSR Number
173	4155	151	4327	177	5401
151	4251	163	4781	178	5432
162	4721	161	4231	186	5509
169	5057	161	4659	181	5596
146	4228	164	4865		

1.5.8 MS SORT 3.2

This release contains corrections for the following PSR numbers:

Summary	PSR Number	Summary	PSR Number	Summary	PSR Number
144	3890	156	4230	166	4585
144	4041	156	4472	166	4663
144	4108	158	4368		
144	4109	165	4795		

1.5.9 ALGOL 1.4

This release contains corrections for the following PSR number:

Summary	PSR Number
144	3445

1.5.10 RESPOND 1.3

This release contains corrections for the following PSR numbers:

Summary	PSR Number
144	3375
144	3853

1.5.11 TAPE SORT/MERGE 2.2

This release contains corrections and modifications for the following PSR and RSM numbers:

Summary	PSR Number
151	4278
164	4397
165	4268
165	4714

RSM: A080

1.5.12 ERROR RECOVERY 1.3

This release contains corrections for the following PSR numbers:

Summary	PSR Number	Summary	PSR Number
152	4227	154	4316
165	4643	164	4859
171	5097		

1.5.13 LISA 1.2

This release contains corrections for the following PSR number:

Summary	PSR Number
144	4157

1.5.14 MSOS UTILITY 1.3

This release contains corrections for the following PSR numbers:

Summary	PSR Number
147	3733
167	4912
151	4343

1.5.15 USASI FORTRAN 1.1

This release contains corrections and modifications for the following PSR and RSM numbers:

Summary	PSR Number	Summary	PSR Number	Summary	PSR Number	Summary	PSR Number
160	4581	165	4090	165	4469	169	4811
165	4660	165	4729	165	4635	169	4823
165	4662	164	4754	165	3882	177	5185
163	4748	171	4815	165	3353	182	5511
164	4754	171	5121	172	4679		
165	4314	165	3881	165	4711		
169	4295	165	3880	163	4748		

RSM: A053

1.5.16 BSIPP

This release contains corrections for the following PSR numbers:

Summary	PSR Number
177	5021

1.5.17 SIPP

This release contains corrections for the following PSR numbers:

Summary	PSR Number
177	4947
177	4948
177	5021

1.6 DEFICIENCIES

1.6.1 MASS STORAGE SORT 3.2

MS Sort does not recognize any records added by COBOL to a continuation file after the first call to MS Sort.

MS Sort cannot process LISA files that contain records in universal format.

1.6.2 RESPOND 1.3

If a COPY request for a nonexistent file is issued, the COPY will not execute and RESPOND delivers the message:

001 CONTINUE

A COPY with step size greater than 9 is rejected. The following diagnostic message appears:

007 PARAMETER 4 IN ERROR

1.6.3 L-MSIO 4.2

The invalid key on a PUT occurs when the write is attempted beyond allocated area, thus the records contained in that block are lost.

1.7 LIMITATIONS

1.7.1 MSOS

Support for 852 disk pack is available for MSOS 4.2 only by special request.

Character I/O is not allowed on 659 tape transports. The following MMTC status bits will be edited into the existing parity bit in the UST:

- Parity errors
- Phase encoded errors
- Cyclic redundancy errors
- Memory flag bit errors

MSOS converts the CDC 64 character subset of ASCII only. Therefore, opening and closing braces must be represented by 12-0 and 11-0 punches respectively.

A $\genfrac{}{}{0pt}{}{7}{8}$ punch in column 1 is interpreted by CDC card readers as an end-of-file card. Therefore a $\genfrac{}{}{0pt}{}{7}{8}$ punch (left brace in Hollerith, quotation mark in ASCII) in column 1 of card input causes an end-of-file indication.

1.7.2 COMPASS

When using nested conditionals or macros the number of lines tagged with IFZ or IFN may not be accurate. However, the assembly will be correct and the program will execute properly.

1.7.3 FORTRAN

The user can DIMENSION in all TYPE statements except TYPE other.

1.7.4 ADAPT (16K VARIANT)

The 16K variant does not contain the new error recovery package.

1.7.5 USASI COBOL

Compute statement involving exponentiation of BCD numeric fields that are greater than 11 digits will lose accuracy since exponentiation is performed in floating-point mode.

USASI COBOL is limited to the following minimum hardware configuration:

Minimum requirements for MSOS 4.2

3300 or 3500 machine type

32K memory

Business Data Processor Module 3304-2, 3304-3, 3504-1, 3514-2, or 3514-4

1.7.6 MSOS UTILITY

Performing an I/O function from a unit onto itself does not result in a diagnostic.

Setting SELECT JUMP 2 during operation terminates only COPYT, COPYS, and VERIFY.

MSOS Utility functions which use tape parity (even or odd) to determine the mode in which data is recorded (BCD or Binary) do not perform correctly with nine track tape transports (MMTC/659). In particular, COPYT, COPYS, and VERIFY are unable to process mixed-mode nine track input tapes correctly. COPYT, COPYS, and VERIFY may process a conversion mode input tape incorrectly if a spurious parity error occurs. COPYT, COPYS, and VERIFY will process a pack mode input tape correctly. COPYT and COPYS will write output tapes correctly, regardless of the modes to be used.

1.7.7 USASI FORTRAN

USASI FORTRAN is limited to the following minimum configuration:

Minimum requirements for MSOS 4.2.

32K memory

USASI FORTRAN utilizes an overlay structure which imposes the following limitations:

Operating environment must be established at the time overlays are created.

File ordinal 49 is used by the compiler for its overlay file. Therefore, the user may not open a file 49 before a compile run. If the user wishes to use a file ordinal 49 during execution, he may close 49 after the compile phase and open a new file 49 preceding the load. File ordinal 49 may be used if the USASI FORTRAN object program is run separately from the compile.

The overlays, once created, become absolute programs and are dependent on the system for which they were absolutized. Therefore, if the resident portion of the system is altered, the overlays must be recreated.

Compiler routines compiled with USASI FORTRAN must be compiled with the S option.†

Object-time routines compiled with USASI FORTRAN must be compiled without the S option.†

The MS FORTRAN and USASI FORTRAN object-time routines cannot reside on the same MSOS library because they have some of the same entry point names. Therefore, one set of the object-time routines must be placed on an auxiliary library. The release materials for USASI FORTRAN are set up to place the USASI FORTRAN object-time routines on an auxiliary library.

---

† The presence of the S parameter on the $^{7}_{9}$ UFORTRAN card specifies integers and logical variables compiled as 24-bit entities; the absence of the S parameter as 48-bit entities.

1.7.8 SIPP

SIPP functions which use tape parity (even or odd) to determine the mode in which data is recorded (BCD or binary) do not perform correctly with nine track tape transports (MMTC/659). In particular, TP, TC and TPC are unable to correctly process a mixed-mode (pack and conversion mode records on the same tape) nine track input tape correctly. TP, TC, and TPC may process a conversion mode input tape incorrectly if a spurious parity error occurs. TP, TC, and TPC will process pack mode input tape correctly. CT will write output tapes correctly, regardless of the modes to be used.

1.7.9 BSIPP

BSIPP functions which use tape parity (even or odd) to determine the mode in which data is recorded (BCD or binary) will not perform correctly with nine track tape transports (MMTC/659). In particular, TP, TC, and TPC are unable to correctly process a mixed-mode (pack and conversion mode records on the same tape) nine track input tape correctly. TP, TC, and TPC may process a conversion mode input tape incorrectly if a spurious parity error occurs. TP, TC, and TPC will process pack mode input tape correctly. CT will write output tapes correctly, regardless of the modes to be used.

1.7.10 ALGOL

A $\frac{7}{8}$ punch in column 1 is interpreted by CDC card readers as an end-of-file card. Therefore, a $\frac{7}{8}$ punch (left brace in Hollerith, quotation mark in ASCII) in column 1 of card input causes an end-of-file indication. Therefore, ALGOL source cards using ASCII card punch codes should not begin control statements (such as 'BEGIN' and 'END') in column 1.

1.8 PUBLICATIONS

The following publications are relevant to MSOS 4.1 installation:

Mass Storage Operating System Reference Manual	60223100
MSOS Operator's Guide	60176100
MSOS/RTS Diagnostic Handbook	60236700
COMPASS Reference Manual	60236800
FORTRAN Reference Manual	60057600
COSY/MSOS Reference Manual	60207300
COBOL/MSOS Reference Manual	60191100
MSIO/MSOS Reference Manual	60191300
SIPP Background Operator Reference Manual	60190200
SIPP Reference Manual	60130400
MSOS Sort Merge Reference Manual	60281500
ALGOL Reference Manual	60134800
SFP/MSOS Reference Manual	60207200
RESPOND EXPORT/IMPORT Reference Manual	60207500
ADAPT Reference Manual	60173400
PERT/COST Reference Manual	60132500
PERT/TIME Reference Manual	60131100
SAINT Reference Manual	60213700
LISA Reference Manual	60236900
MSOS Utility Reference Manual	60279400
USASI COBOL Reference Manual	60281100
USASI FORTRAN Reference Manual	60281400

PART II

INSTALLATION PROCEDURES

INSTALLATION PROCEDURES 1

The following general steps are required for MSOS installation. (See Figure II-1.)

System definition and configuration

MSOS option selection

Hardware initialization

Install interim library

Deck preparation

Binary source modification

Final library generation

Product set incorporation

1.1 SYSTEM DEFINITION AND CONFIGURATION

The chart in Figure II-2 is provided as an aid for determining system hardware configuration parameters. It is recommended that the chart be completed at this time. Once controllers, equipment, channels, and units are defined, this chart will prove an invaluable aid in completing machine operations and assembly option settings.

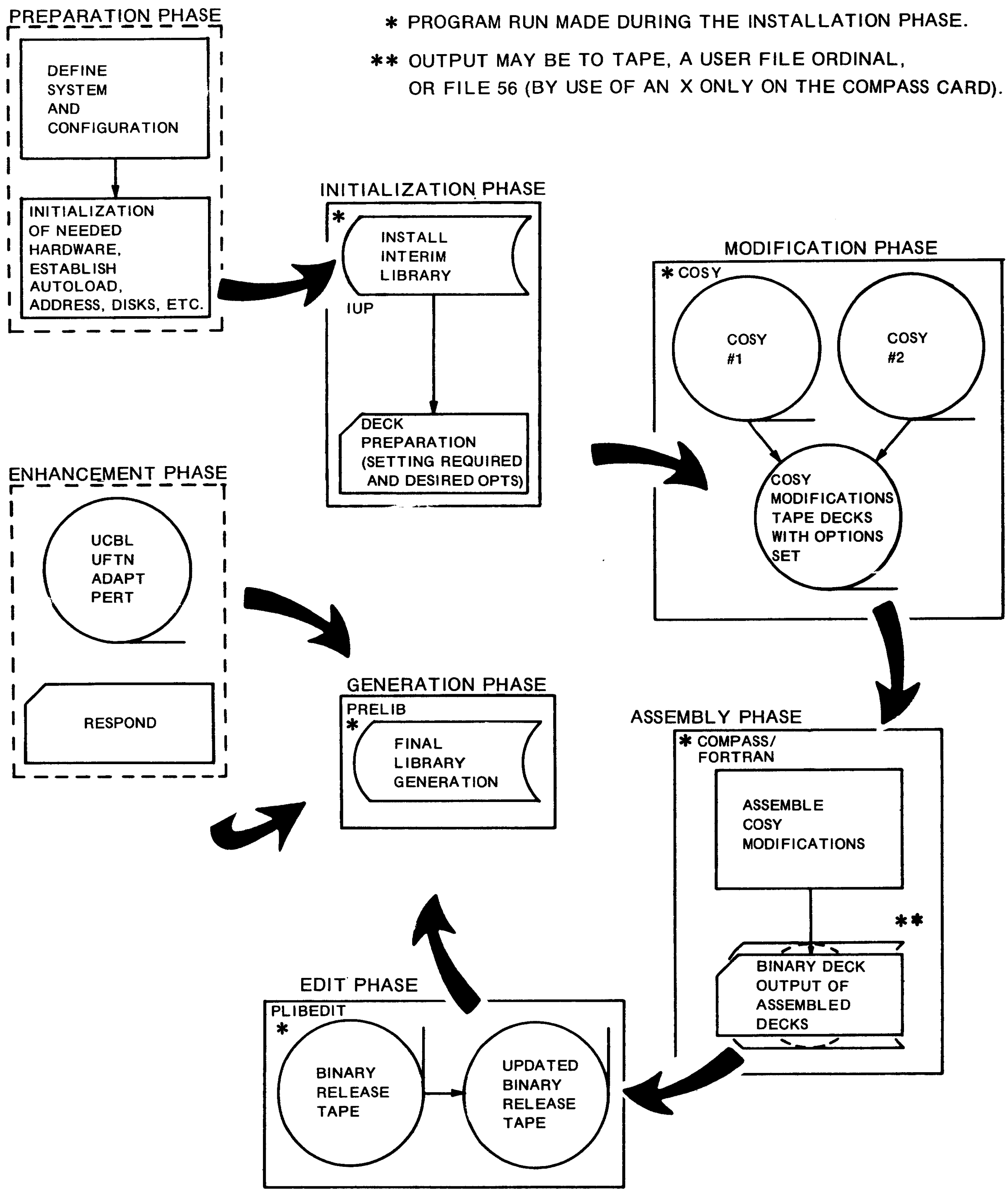


Figure II-1. Overview of Installation Procedures

COMPUTER (Circle One) 3500 3300 3200 3150 3100

MEMORY SIZE (Circle One) 32K 16K

SYSTEM TYPE (Circle One) STANDARD BATCH 3300/3500 MEMORY PROTECT 3100/3150/3200 MP

3500/3300 OPTIONAL HARDWARE			3200/3150/3100 OPTIONAL HARDWARE		
Floating Point Hardware	YES	NO	Floating Point Hardware	YES	NO
Business Data Processor (BDP)			BCD Option	YES	NO
Non-enhanced	YES	NO			
Enhanced	YES	NO			
Memory Protect Hardware	YES	NO	Memory Protect Hardware	YES	NO

TYPE	CONTROLLER	QUAN.	CHAN.	EQUIP.	UNIT	AET ORD.	HDWR. CODE	ENTRY POINT	SYSTEM UNIT
854	3234	1			10	01	13	MSIO3234	LIB
853	3234	1			10	01	13	MSIO3234	LIB
863	3436 3637	1			00	01	15	MSIO3436	LIB
841	3553	1			10	01	13	MSI03553	LIB
852	3232	1			00	01	13	MSIORADD	LIB
console		1	00	00	00	02	05	DRIVER05	CTO/CFO
405	3447 3649	1			00	03	02	DRIVER02	INP
405	3248	1			00	03	02	DRIVER02	INP
501	3256	1			00	04	03	DRIVER03	OUT
505	3659 3152				00	04	03	DRIVER03	OUT
512	3555	1			00	04	03	DRIVER03	OUT
415	3446	1			00	05	04	DRIVER04	PUN
415	3245	1			00	05	04	DRIVER04	PUN
			available magnetic tape units						
			available mass storage units						
			other available hardware units						

Figure II-2. System Definition and Configuration Chart

MSOS OPTION SELECTION 2

The MSOS binary source file requires preparation before a final MSOS system can be generated. The preparation procedures are divided into the following subsections:

Installation options

Installation dependent routines

2.1 MSOS INSTALLATION OPTIONS

2.1.1 BATCH

The BATCH option provides the ability to process batch jobs exclusively.

2.1.2 STANDARD

The STANDARD option provides the ability to process batch jobs and a priority program in a time-shared manner. 32K of memory is recommended.

2.1.3 MEMORY PROTECT

The Memory Protect option provides the ability to process batch jobs and a priority program in a time-shared manner and provides memory protection for the operating system and priority program through the use of hardware toggle switches. 3200/3150/3100 Memory Protect option requires the 10123 Memory Protect hardware. 3300/3500 Memory Protect option requires the 10099 Memory Protect hardware. 32K of memory is required.

2.2 INSTALLATION-DEPENDENT ROUTINES

Table II-1, Assembly Options, lists the routines that contain assembly options. Some of the options apply to all system types, while others apply only to Standard (STD), Memory Protect (MP), or Batch MSOS. Users with a core memory size of 16K please refer to Part III, Section 8.5.

Table II-1 shows the installation-dependent routines in the order they appear on the COSY tape. Table II-2 shows these routines in the order they appear on the MSOS binary source. The binary source is divided into seven sections including Standard, Memory Protect, Batch, Common Absolute, BCD, BDP, and Common Relocatable routines.

The Assembly Option table (Table II-1) is not inclusive. Options in MSIO and BMSIO for privacy code, system device, maximum tracks, maximum segment count, and maximum file count, plus all the options in RESPOND, SAINT, and ALGOL have been omitted.

TABLE II-1. MSOS ASSEMBLY OPTIONS

	MZERO	ZERO	CIO	BCIO	DRIV3528	DRIV3659	DRIV3555	MSIO	BSIO	MCICREC1	SCICREC1	BCICREC1	RDUMP	PRELOAD	MSIOCCP	LOADER	OVPRO	POSTLOAD	INITIAL	OC	AREM	SIPP
ACCT										X	X	X		X								
AET										X	X	X										
ATTMPT1													X	X	X	X		X				
ATTMPT2													X	X	X	X		X				
BDP										X	X	X										
BDP3312	X	X																				
BF														X								
BFDTL								X														
BKSPLIM2													X	X	X	X		X				
BRHT										X	X											
CARDPNCH													X									
CARDRDR													X									
CIT										X	X											
CKSMCNST													X	X	X	X		X				
CLOCK										X	X	X										
DP			X	X																		
FDP										X												
FFDTL								X	X													
INP										X	X	X										
IOT												X										
LERP																X		X				
MAXCORE										X	X	X										
MAXERASE													X	X	X	X		X				
MEMPTKT																						
MP	X								X				X	X		X		X	X			
MST								X	X													
MZEROSUP																						
NCHANS										X	X	X										
NLPS						X	X															
NOATMPTS													X	X	X	X		X				
NOPS																						X
NRMST																				X		
PRINTER													X									
PT										X	X	X										
PUN										X	X	X										
RATL								X	X													
READB														X	X			X				
REQ																				X		
RHT										X	X	X										
SER, OPT													X									
SETCLK											X	X										
SFDTL								X														
SPACE																					X	
SXTK																						
SYST. IOM										X	X	X										
T657					X																	
T659					X																	
TAPES													X	X	X	X		X				
TIMEOUT			X	X																		
TRAIN														X								
TRN595X														X								
USASI																X	X	X				
USARDT										X	X	X										
ZROSUPOC																						

	BSIPP	RAAR	SCAR	MTWPRR	MTRPRR	MTLDCPRR	WHATISIT	NRD	NRC	PASSONE	BCDOUT	MSIOGPRW	PCOBOLP2	PMULTIPL	PVARAN	MSSORT	MSSINTS	MSSMERG	SORT	SORTPHS1	SRTEQUAL	ALGOL
ACCT																						
AET																						
ATTMPT1					X																	
ATTMPT2					X																	
BDP																						
BDP3312										X		X	X		X	X	X	X	X	X	X	
BF																						
BFDTL																						
BKSPLIM2					X																	
BRHT																						
CARDPNCH			X																			
CARDRDR			X																			
CIT																						
CKSMCNST				X																		
CLOCK																						
DP																						
FDP														X								
FFDTL																						
INP																						
IOT																						
LERP																						
MAXCORE																						
MAXERASE				X																		
MEMPTKT																X			X			
MP																						
MST																						
MZEROSUP											X											
NCHANS																						
NLPS																						
NOATMPTS																						
NOPS	X																					
NRMST																						
PRINTER			X																			
PT																						
PUN																						
RATL																						
READB																						
REQ																						
RHT																						
SER, OPT																						
SETCLK																						
SFDTL																						
SPACE																						
SXTK																X						
SYST. IOM																						
T657																						
T659																						
TAPES		X	X				X		X													
TIMEOUT																						
TRAIN																						
TRN595X																						
USASI																						
USARDT																						
ZROSUPOC											X											

TABLE II-2. INSTALLATION-DEPENDENT ROUTINES ON THE BINARY SOURCE

STD ROUTINES	MP ROUTINES	BATCH ROUTINES	COMMON ABS ROUTINES	BCD ROUTINES	BDP ROUTINES	COMMON REL ROUTINES
ZERO	MZERO	ZERO	MSIOCCP	MSIOGPRW	MSIOGPRW	RAAR
CIO	CIO	BCIO	OC	MSSORT	COBOLP2	SCAR
MSIO††	MSIO††	BMSIO††	AREM	MSSINTS	MULTIPLY	MTWPRR
DRIV3528	DRIV3528	DRIV3528	PASSONE	MSSMERG	VARAN	MTRPRR
DRIV3659	DRIV3659	DRIV3659	MTWPRR	SORT	MSSORT	MTLDCPRR
DRIV3555	DRIV3555	DRIV3555	MTRPRR	SORTPHS1	MSSINTS	WHATISIT
SCICREC1††	MCICREC1††	BCICREC1††	MTLDCPRR	SRTEQUAL	MSSMERG	NRD
INITIAL	INITIAL†	INITIAL	NRC		SORT	NRC BCDOUT
RDUMP	RDUMP	RDUMP			SORTPHS1	ALGOL
PRELOAD	PRELOAD	PRELOAD			SRTEQUAL	BSIPP
LOADER	LOADER	LOADER				SIPP
OVPRO	OVPRO	OVPRO				OC
POSTLOAD	POSTLOAD	POSTLOAD				AREM

† Must be modified for 3100/3150/3200 Memory Protect

†† Must be modified by all installations

2.2.1 MSOS COSY TAPE CONTENTS

The MSOS COSY tape contains all MSOS routine (decks) used in generating either STD, MP, or BATCH MSOS libraries. Table II-3 lists the COSY routines in the order that they occur on the COSY tape. Each deck is coded to indicate its use as follows:

S	Routine is used for STD section of the MSOS binary source
M	Routine is used for MP section of the MSOS binary source
B	Routine is used for Batch section of the MSOS binary source
O	Routine is used for STD, MP, and Batch sections of the MSOS binary source containing assembly options
Y	Routine is used for 3100/3150/3200 Memory Protect (MP EQU 32)
A	Routine is used for STD, Batch, and MP
SM	Routine is used for both STD and MP MSOS
SB	Routine is used for both STD and Batch MSOS
†	Exists as macros only
Z	Routine has a special assembly option (refer to Section 2.3 for specified routine)
E	Routine has error recovery assembly options (refer to Section 2.4)
U	Routine is used for USASI COBOL

The deck name of each routine in the following list is suffixed with a source language code The definitions of the codes are:

CMP	COMPASS
UFTN	USASI FORTRAN

TABLE II-3. MASS STORAGE COSY TAPES

COSY Tape Number 1

Deck Name	Code	Source Language	Product Name
MZERO	MY	CMP	MSOS SYSTEM ROUTINES
ZERO	SB	CMP	
CIO	SMZ	CMP	
BCIO	BZ	CMP	
DRIV607	AZ	CMP	
DRIV606	AZ	CMP	
DRIV3528	AZ	CMP	
DRIV3649	A	CMP	
DRIV3248	A	CMP	
DRIV3659	AZ	CMP	
DRIV3555	AZ	CMP	
DRIV3644	A	CMP	
DRIVTYWR	A	CMP	
DRIV3245	A	CMP	
DRIV3195	A	CMP	
DRIV3293	A	CMP	
OCRMACRO	*A	CMP	
DRIV3691	A	CMP	
MSIO3232	A	CMP	
MSIO3234	A	CMP	
MSIO3553	A	CMP	
MSIO3436	A	CMP	
MSIO	SMZ	CMP	
BMSIO	BZ	CMP	
MCICREC1	MYZ	CMP	
SCICREC1	SZ	CMP	
BCICREC1	BZ	CMP	
RDUMP	OZE	CMP	
PRELOAD	OZE	CMP	
MSIOCCP	ZE	CMP	
CKREC1	A	CMP	
LOADER	OZE	CMP	
OVPRO	AZ	CMP	
PROTECT	A	CMP	
POSTLOAD	OZE	CMP	
AUTOLOAD	A	CMP	
INITIAL	OY	CMP	
OC	AZ	CMP	OCAREM
AREM	AZ	CMP	
MISC	A	CMP	
EXECOVR	A	CMP	OBJECT ROUTINE

Deck Name	Code	Source Language	Product Name
MSUTIL	A	CMP	MSUTIL ROUTINES
UDUMP	A	CMP	
LOAD	A	CMP	
ENTER	A	CMP	
DMSD	A	CMP	
PSM	A	CMP	
PSM2	A	CMP	
FLL2	A	CMP	
FLL	A	CMP	
PFLD	A	CMP	
CLRDSK	A	CMP	
WTADR	A	CMP	
MSDLRW	A	CMP	
PRINT3	A	CMP	
BINTOBCD	A	CMP	
SPLTDATE	A	CMP	
DTB	A	CMP	
BDTRCK	A	CMP	
IOPACK	A	CMP	OBJECT ROUTINE
IODRAIN	A	CMP	
SNAPSHOT	A	CMP	
COMAC	*A	CMP	COMPASS MACROS
FDPBOXS	A	CMP	SIMULATION ROUTINES
BCDBOXS	A	CMP	
OPTBOXS	A	CMP	
BSIPP	SMZ	CMP	BSIPP
SIPP	AZ	CMP	SIPP
SUMS	SM	CMP	SUMS
SFP	A	CMP	STD. FILE PROC.
PRELIB	A	CMP	PRELIB ROUTINES
PLOVINT	A	CMP	
OV1	A	CMP	
PHASE1	A	CMP	
PHASE2	A	CMP	
PLOVINT1	A	CMP	
PLOV2	A	CMP	
PLOV3	A	CMP	

TABLE II-3. MASS STORAGE COSY TAPES (cont'd)

Deck Name	Code	Source Language	Product Name
RAAR	AZ	CMP	STD. ERROR RECOVERY
SCAR	AZ	CMP	
MTWPR	A	CMP	
MTRPR	A	CMP	
MTLDACPR	A		
MTDER	A	CMP	
PRCPR	A	CMP	
CRDER	A	CMP	
CPDER	A	CMP	
MTWPRR	AZ	CMP	
MTRPRR	AZ	CMP	
MTLDCPRR	AZ	CMP	
CMNRTNS	A	CMP	
WHATISIT	AZ	CMP	
NRD	AZ	CMP	
NRC	AZ	CMP	
NWR	A	CMP	
T.NOTRDY	A	CMP	
TYPEOUT	A	CMP	
WHATKIND	A	CMP	
COSYRDWT	A	CMP	COSY
COSY2.0	A	CMP	
COMPASSB	A	CMP	COMPASS LDR
COMPASS	AE	CMP	COMPASS ASSEMBLER
OVERLAY1	AE	CMP	
PASSONE	AEZ	CMP	
PASSTWO	AE	CMP	
SYMTBLE	A	CMP	
CRT	AE	CMP	
FT1	AE	CMP	MS FORTRAN COMPILER
FT2	AE	CMP	
FT3	AE	CMP	
FT4	AE	CMP	
FT5	AE	CMP	
FT6	AE	CMP	
FTE	AE	CMP	
FTN	A	CMP	FORTRAN LDR.
FLOVER	A	CMP	MS FORTRAN OBJECT TIME ROUTINES
BCDINP	A	CMP	
BCDOUT	A	CMP	
FORMAT	A	CMP	

Deck Name	Code	Source Language	Product Name
BINARY	A	CMP	MS FORTRAN OBJECT TIME ROUTINES
BUFFER	A	CMP	
UNIT	A	CMP	
IOCHK	A	CMP	
EOFCHK	A	CMP	
TAPEHAND	A	CMP	
PAUSE	A	CMP	
CONTROL	A	CMP	
CIO.MSIO	A	CMP	
LOCATE	A	CMP	
DOUBLE	A	CMP	
DFPRIME	A	CMP	
DFP	A	CMP	
Q1QADRI	A	CMP	
ITOJ	A	CMP	
ITOX	A	CMP	
XTOI	A	CMP	
POWRF	A	CMP	
SINCOS	A	CMP	
ATANF	A	CMP	
EXPF	A	CMP	
LOGF	A	CMP	
SIGNF	A	CMP	
SQRTF	A	CMP	
ABSF	A	CMP	
EXTREMA1	A	CMP	
EXTREMA2	A	CMP	
FLOATF	A	CMP	
FIXF	A	CMP	
MASKINGF	A	CMP	
FAULTS	A	CMP	
SENSLITE	A	CMP	
SENSWTCH	A	CMP	
Q8QERROR	A	CMP	
SAINT	AZ	CMP	SAINT
RESPOND	SMZ	CMP	RESPOND ROUTINES
ROVER1	SMZ	CMP	
ROVER2	SMZ	CMP	
ROVER3	SMZ	CMP	
FGCOPY	SMZ	CMP	

End of COSY Tape Number 1

TABLE II-3. MASS STORAGE COSY TAPES (cont'd)

COSY Tape Number 2

Deck Name	Code	Source Language	Product Name
EXPANDSF	A	CMP	
REPLACE	A	CMP	
DELETE	A	CMP	
INSERT	A	CMP	
UPDATE	A	CMP	
ISGET	A	CMP	
BUILD	A	CMP	
BLOCKER	A	CMP	LISA ROUTINES
FILEDEF	A	CMP	
RECDEF	A	CMP	
BUFDEF	A	CMP	
FILEDEFF	A	CMP	
RECDEFF	A	CMP	
BUFDEFF	A	CMP	
FD	A	CMP	
MSOSUTIL	A	CMP	
COPYS	A	CMP	
COPYT	A	CMP	MSOS UTILITY ROUTINES
VERIFY	A	CMP	
DUMP	A	CMP	
ERROR	A	CMP	
PLIBEDIT	A	CMP	PLIBEDIT
MSIOMAIN	AZ	CMP	
MSIOSTER	AZ	CMP	
MSIOGPRW	AZ	CMP	L-MSIO ROUTINES
MSIOTPOV	A	CMP	
MSIOMSOV	AZ	CMP	
MRESTART	A	CMP	
COBOL	A	CMP	
COBOLIE	A	CMP	
COBOLD1	A	CMP	
COBOLP1	A	CMP	MS COBOL BCD COMPILER
COBOLD2	A	CMP	
COBOLP2	A	CMP	
COBOLDP3	A	CMP	
COBOLIO	A	CMP	

Deck Name	Code	Source Language	Product Name
TRANSMIT	A	CMP	
FIGCON	A	CMP	
COMPARE	A	CMP	
EDIT	A	CMP	
EDITCOBL	A	CMP	
MULTPLY	A	CMP	
DIVIDE	A	CMP	
BMULTPLY	A	CMP	
BDIVIDE	A	CMP	
ERRSTOP	A	CMP	
TYPELOOP	A	CMP	
DPMULDIV	A	CMP	
STRIPPER	A	CMP	
BSTRIPPR	A	CMP	
SUBSCRP	A	CMP	
DEEDIT	A	CMP	
EXAMINE	A	CMP	
NUMERIC	A	CMP	
ALPHABET	A	CMP	MS COBOL BCD OBJECT-TIME ROUTINES
ACCEPT	A	CMP	
DISPLAY	A	CMP	
MVFIGCON	A	CMP	
VARC1	A	CMP	
VARN	A	CMP	
VARAN	A	CMP	
ROUNDER	A	CMP	
CONVERT	A	CMP	
ZIPPER	A	CMP	
LOGICAL	A	CMP	
PCOBOL	A	CMP	
PCOBOLIE	A	CMP	
PCOBOLD1	A	CMP	
PCOBOLP1	A	CMP	MS COBOL BDP COMPILER ROUTINES
PCOBOLD2	A	CMP	
PCOBOLP2	AZ	CMP	
PCOBOLDP	A	CMP	
PCOBOLO	A	CMP	
PERRSTOP	A	CMP	
PTYPELOO	A	CMP	MS COBOL BDP OBJECT-TIME ROUTINES
PFIGCON	A	CMP	

TABLE II-3. MASS STORAGE COSY TAPES (cont'd)

Deck Name	Code	Source Language	Product Name
PCOMPARE	A	CMP	MS COBOL BDP OBJECT TIME ROUTINES
PMULTIPL	A	CMP	
PDIVIDE	A	CMP	
PSTRIPPE	A	CMP	
PSUBSCRP	A	CMP	
PDEEDIT	A	CMP	
PEXAMINE	A	CMP	
PNUMERIC	A	CMP	
PALPHABE	A	CMP	
PACCEPT	A	CMP	
PDISPLAY	A	CMP	
PMVFIGCO	A	CMP	
PVARC1	A	CMP	
PVARN	A	CMP	
PVARAN	AZ	CMP	
PDPBINBC	A	CMP	
PDPBCDBI	A	CMP	
PZIPPER	A	CMP	
PLOGICAL	A	CMP	
MSSORT	AZ	CMP	MS SORT ROUTINES
MDYNALL	A	CMP	
MSSEDIT	A	CMP	
MSSIOP1	A	CMP	
MSSINTS	AZ	CMP	
MSSIOP2	A	CMP	
MSSMERG	AZ	CMP	
SORT	AZ	CMP	TAPE SORT ROUTINES
TDYNALL	A	CMP	
BINANDEC	A	CMP	
SDUMP	A	CMP	
SRTPRINT	A	CMP	
SORTPOLY	A	CMP	
SRTMBALF	A	CMP	
SRTMBALB	A	CMP	
SRTMPOLF	A	CMP	
SRTMPOLB	A	CMP	
SORTPDMY	A	CMP	
SORTEDIT	A	CMP	
SORTIOP1	A	CMP	
SORTPHS1	AZ	CMP	
SRTRBALF	A	CMP	
SRTRPOLF	A	CMP	
SRTRBALB	A	CMP	
SRTRPOLB	A	CMP	
SRESTART	A	CMP	

Deck Name	Code	Source Language	Product Name
RESTART1	A	CMP	TAPE SORT ROUTINES
RSTRTDUM	A	CMP	
SORTIOP2	A	CMP	
SRTEQUAL	AZ	CMP	
POLYFORW	A	CMP	
BALCFORW	A	CMP	
BALCBACK	A	CMP	
POLYBACK	A	CMP	
WAITBEEP	A	CMP	
ALGOL	AZ	CMP	ALGOL LDR
ALG0	A	CMP	ALGOL COMPILER ROUTINES
ALG1	A	CMP	
ALG2	A	CMP	
ALG3	A	CMP	
ALG4	A	CMP	
ALG5	A	CMP	
ALGLIB00	A	CMP	ALGOL OBJECT TIME ROUTINES
ALGLIB01	A	CMP	
ALGLIB02	A	CMP	
ALGLIB03	A	CMP	
ALGLIB04	A	CMP	
ALGLIB05	A	CMP	
ALGLIB06	A	CMP	
ALGOLRUN	A	CMP	
IUP. INIT		CMP	IUP
IUP		CMP	

End of COSY Tape Number 2

2.2.2 ASSEMBLY OPTION CODING SHEETS

It is recommended that the assembly option coding sheets (Table II-4) be used as an aid in selecting assembly options. Unused lines can be crossed out and parameters supplied for chosen lines. These coding sheets contain all options on the MSOS COSY tapes.

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	14	
MP	EQU		MEMORY PROTECT OPTION
	DELETE/	15	
BD3312	EQU		BDP OPTION
	DELETE/	76, 91	
	OCT		BDP
	OCT		COLLATING
	OCT		SEQUENCE
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
MZERO	DECK/		
	DELETE/	9	BDP OPTION
BDP3312	EQU		
	DELETE/	55, 70	
	OCT		BDP
	OCT		COLLATING
	OCT		SEQUENCE
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
ZERO	DECK/		
	DELETE/	5	
DP	EQU		DUAL CHANNEL ACCESS OPTION
	DELETE/	6	
TIMEOUT	EQU		LOST INTERRUPT TIMEOUT
CIO	DECK/		
	DELETE/	4	
DP	EQU		DUAL CHANNEL ACCESS OPTION
	DELETE/	5	
TIMEOUT	EQU		LOST INTERRUPT TIMEOUT
BCIO	DECK/		
WRITEB	EQU		CONTROLLED BACKSPACE FUNCTION INSTALLED

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
DRIV607	DECK/		
WRITEB	EQU		CONTROLLED BACKSPACE FUNCTION INSTALLED
DRIV606	DECK/		
WRITEB	EQU		CONTROLLED BACKSPACE FUNCTION INSTALLED
DRIV3528	DECK/		
	DELETE/	120	
T657	EQU		NUMBER OF 657 DRIVES
	DELETE/	122	
T659	EQU		NUMBER OF 659 DRIVES
DRIV3528	DECK/		
	DELETE/	11	
NLPS	EQU		NUMBER OF LINE PRINTERS
DRIV36591	DECK/		
	DELETE/	1	
NLPS	EQU		NUMBER OF LINE PRINTERS
DRIV3555	DECK/		
	DELETE/	23	
RATL	EQU		NUMBER OF MST ENTRIES
	DELETE/	25, 26	
FFDTL	EQU		BATCH FDT LENGTH
BFDTL	EQU		PRIORITY FDT LENGTH
	DELETE/	27	
SFDTL	EQU		STANDARD UNITS FDT
	DELETE/	41	
	OCT		MASTER SECURITY CODE
	DELETE/	42	
	DEC		AET ORDINAL
	DELETE/	45	
	DEC		MAXIMUM FILE SIZE
	DELETE/	46	
	DEC		MAXIMUM SEGMENT COUNT
	DELETE/	47	
	DEC		MAXIMUM FILE COUNT
	DELETE/	52, 55	
MST	OCT		MASS STORAGE TABLE
	OCT		
	OCT		
	OCT		
MSIO	DECK/		

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	21	
RATL	EQU		NUMBER OF MST ENTRIES
	DELETE/	23	
FFDTL	EQU		BATCH FDT LENGTH
	DELETE/	38	
	OCT		MASTER SECURITY CODE
	DELETE/	39	
	DEC		AET ORDINAL
	DELETE/	42	
	DEC		MAXIMUM FILE SIZE
	DELETE/	43	
	DEC		MAXIMUM SEGMENT COUNT
	DELETE/	44	
	DEC		MAXIMUM FILE COUNT
	DELETE/	46,49	
MST	OCT		MASS STORAGE TABLE
	OCT		
	OCT		
	OCT		
BMSIO	DECK/		
	DELETE/	109	
ACCT	EQU		ACCOUNTING OPTION
	DELETE/	110	
BDP	EQU		BDP OPTION
	DELETE/	111	
CLOCK	EQU		CLOCK INTERRUPT PROCESSOR
	DELETE/	112	
FDP	EQU		FLOATING POINT HARDWARE
	DELETE/	113	
MAXCORE	EQU		MEMORY SIZE
	DELETE/	114	
MP	EQU		TYPE OF MEMORY PROTECT
	DELETE/	115	
NCHANS	EQU		NUMBER OF CHANNELS
	DELETE/	116	
USARDT	EQU		USASI COBOL USERS
	DELETE/	121	
INP	EQU		ASCII INPUT OPTION
	DELETE/	123	
PUN	EQU		ASCII PUNCH OPTION
	DELETE/	124	
PT	EQU		ASCII PAPER TAPE OPTION
	DELETE/	3271,3287	
	MACRO	( )	AVAILABLE EQUIPMENT TABLE

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	3354,3360	
RHT	OCT		RUNNING HARDWARE TABLE
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
	DELETE/	3368,3371	
BRHT	OCT		PRIORITY RUNNING
	OCT		HARDWARE TABLE
	OCT		
	OCT		
	DELETE/	3438,3445	
	00		ch 0 CENTRAL
	00		ch 1 INTERRUPT
	00		ch 2 TABLE
	00		ch 3
	00		ch 4
	00		ch 5
	00		ch 6
	00		ch 7
	DELETE/	3481	
SYST.IOM	OCT		SYSTEM INTERRUPT MASK
MCICREC1	DECK/		
	DELETE/	81	
ACCT	EQU		ACCOUNTING OPTION
	DELETE/	82	
BDP	EQU		BDP OPTION
	DELETE/	83	
CLOCK	EQU		CLOCK INTERRUPT PROCESSOR
	DELETE/	84	
MAXCORE	EQU		MEMORY SIZE
	DELETE/	85	
NCHANS	EQU		NUMBER OF CHANNELS
	DELETE/	86	
SETCLK	EQU		SETCLOCK
	DELETE/	87	
USARDT	EQU		USASI COBOL USERS
	DELETE/	92	
INP	EQU		ASCII INPUT OPTION
	DELETE/	94	
PUN	EQ		ASCII PUNCH OPTION
	DELETE/	95	
PT	EQU		ASCII PAPER TAPE OPTION
	DELETE/	2207,2223	
	MACRO	( )	AVAILABLE EQUIPMENT TABLE

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	2288,2294	RUNNING HARDWARE TABLE
RHT	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
	DELETE/	2302,2305	
BRHT	OCT		PRIORITY RUNNING
	OCT		HARDWARE TABLE
	OCT		
	OCT		
	DELETE/	2372,2379	
	00		ch 0 CENTRAL
	00		ch 1 INTERRUPT
	00		ch 2 TABLE
	00		ch 3
	00		ch 4
	00		ch 5
	00		ch 6
	00		ch 7
	DELETE/	2412	
SYST. IOM	OCT		SYSTEM INTERRUPT MASK
SCICREC1	DECK/		
	DELETE/	81	
ACCT	EQU		ACCOUNTING OPTION
	DELETE/	82	
BDP	EQU		BDP OPTION
	DELETE/	83	
CLOCK	EQU		CLOCK INTERRUPT PROCESSOR
	DELETE/	84	
MAXCORE	EQU		MEMORY SIZE
	DELETE/	85	
NCHANS	EQU		NUMBER OF CHANNELS
	DELETE/	86	
SETCLK	EQU		SETCLOCK
	DELETE/	87	
USARDT	EQU		USASI COBOL USERS

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	92	
INP	EQU		ASCII INPUT OPTION
	DELETE/	94	
PUN	EQU		ASCII PUNCH OPTION
	DELETE/	95	
PT	EQ		ASCII PAPER TAPE OPTION
	DELETE/	805, 812	
IOT	00		ch 0 CENTRAL
	00		ch 1 INTERRUPT
	00		ch 2 TABLE
	00		ch 3
	00		ch 4
	00		ch 5
	00		ch 6
	00		ch 7
	DELETE/	1499,1515	
	AET	( )	AVAILABLE EQUIPMENT TABLE
	DELETE/	1580,1586	
RHT	OCT		RUNNING HARDWARE TABLE
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
	OCT		
	DELETE/	1625	
SYST. IOM	OCT		SYSTEM INTERRUPT MASK
BCICERC1	DECK/		
	DELETE/	3	TYPE OF SYSTEM
MP	EQU		
	DELETE/	4	
SER.OPT	EQU		STANDARD ERROR RECOVERY
READB	EQU		READ BACKWARD FUNCTION LEGAL
WRITEB	EQU		CONTROLLED BACKSPACE FUNCTION INSTALLED
RDUMP	DECK/		
	DELETE/	2	
MP	EQU		TYPE OF SYSTEM
	DELETE/	3	
BF	EQU		BATCH FLAG
	DELETE/	4	
ACCT	EQU		ACCOUNTING OPTION
READB	EQU		READ BACKWARD FUNCTION LEGAL
WRITEB	EQU		CONTROLLED BACKSPACE FUNCTION INSTALLED

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	5	
TRAIN	EQU		USE THE 512 TRAIN CARD
	DELETE/	239	
TRN5951	EQU		TYPE OF TRAIN
	DELETE/	240	
TRN5952	EQU		TYPE OF TRAIN
	DELETE/	232 241	
TRN5923	EQU		TYPE OF TRAIN
PRELOAD	DECK/		
	DELETE/	2	
MP	EQU		TYPE OF SYSTEM
	DELETE/	3	
LERP	EQU		STANDARD ERROR RECOVERY
	DELETE/	4	
USASI	EQU		USASI USER
LOADER	DECK/		
	DELETE/	2	
MP	EQU		TYPE OF SYSTEM
	DELETE/	3	
LERP	EQU		STANDARD ERROR RECOVERY
	DELETE/	4	
USASI	EQU		USASI USER
POSTLOAD	DECK/		
	DELETE/	3	
USASI	EQU		USASI USER
OVPRO	DECK/		
	DELETE/	2	
MP	EQU		TYPE OF SYSTEM
INITIAL	DECK/		
	DELETE/	77	
REQ	BSS		TWO TIMES RATL
	DELETE/	78	
NRMST	BSS		EQUALS MSTL
OC	DECK/		

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	78	
SPACE	BSS		FIVE TIMES RATL
AREM	DECK/		
	DELETE/	3	
BDP3312	EQU		BDP OPTION
MSIOGPRW	DECK/		
	DELETE/	98	
NOPS	EQU		NUMBER OF OPERATION OPTION
	DELETE/	2960,2962	
BLIST	00		
	DELETE/	2963,2965	
B2LIST	00		
	DELETE/	2967,2969	
RSVLIST	00		
	DELETE/	2970,2972	
FALIST	00		
BSIPP	DECK/		
	DELETE/	120	
NOPS	EQU		
SIPP	DECK/		SIMULTANEOUS PERIPHERAL PROCESSOR
	DELETE/	21	
BDP3312	EQU		
PASSONE	DECK/		
	DELETE/	4	
ZROSUPOC	EQU		SUPPRESS LEADING ZEROS
	DELETE/	5	
MZEROSUP	EQU		PRINT NEGATIVE ZERO
BCDOUT	DECK/		
	DELETE/	4	
BDP3312	EQU		BDP OPTION
PCOBOLP2	DECK/		

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	9	
FDP	EQU		
PMULTIPL			
	DELETE/	18	
BDP3312	EQU		BDP OPTION
PVARAN	DECK/		
	DELETE/	9	
BDP3312	EQU		BDP OPTION
	DELETE/	12	
MEMPTKT	EQU		TYPE OF SYSTEM
	DELETE/	21	
SXTK	EQU		MEMORY SIZE
MSSORT	DECK/		
	DELETE/	4	
BDP3312	EQU		BDP OPTION
MSSINTS	DECK/		
	DELETE/	8	
BDP3312	EQU		BDP OPTION
MSSMERG	DECK/		
	DELETE/	9	
BDP3312	EQU		BDP OPTION
	DELETE/	12	
MEMPTKT	EQU		TYPE OF SYSTEM
SORT	DECK/		
	DELETE/	6	
BDP3312	EQU		BDP OPTION
SORTPHS1	DECK/		
	DELETE/	5	
BDP3312	EQU		BDP OPTION
SRTEQUAL	DECK/		

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	89	
TAPES	EQU		TAPE OPTION
	DELETE/	90	
CARDRDR	EQU		CARD READER
	DELETE/	91	
CARDPNCH	EQU		CARD PUNCH
	DELETE/	92	
PRINTER	EQU		PRINTER OPTION
SCAR	DECK/		
	DELETE/	44	
READB	EQU		READB FUNCTION LEGAL
NRD	DECK/		
	DELETE/	89	
TAPES	EQU		TAPE OPTION
RAAR	DECK/		
	DELETE/	16	
TAPES	EQU		TAPE OPTION
WHATISIT	DECK/		
READB	EQU		READ BACKWARD FUNCTION LEGAL
WRITEB	EQU		CONTROLLED BACKSPACE FUNCTION INSTALLED
MTRTNS	DECK/		

TABLE II-4. MSOS ASSEMBLY OPTION CODING SHEETS (cont'd)

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	71	
TAPES	EQU		TAPE OPTION
NRC	DECK/		

2.3 ASSEMBLY OPTION DEFINITION

The following definitions explain the assembly options within each particular routine. The user is cautioned to correlate the particular assembly option setting with the major installation option on the binary source file to determine if the assembly option must be changed. Note there is a difference between the COSY option setting and the assembled value on the binary source in some cases. All corrections are in COSY tape order. The format of each definition is in the following form:

Routine (applicable section of the binary source file)†

Assembly Option (mnemonic)

Definition:

Option:	COSY modifier	
mnemonic	operator	address

Dependency:

Release Value:

Deck Card:

A mnemonic designates each section of the file as follows:

STD	Standard routines
MP	Memory protect routines
BATCH	Batch routines
COMMON ABS	Common absolute routines
BCD	BCD routines
BDP	BDP routines
COMMON REL	Common relocatable routines

2.3.1 MZERO (MP)

Memory Protect Option (MP)

Definition: The option distinguishes between 3300/3500 Memory Protect and 3100/3150/3200 Memory Protect systems.

Option:	DELETE/	14	
MP	EQU	0	3300/3500 MP
MP	EQU	32	3100/3150/3200 MP

† The notation ALL in this section refers to the Standard, Batch, and Memory Protect sections of the MSOS binary source.

Dependency: Applicable to MP systems only. Must have MP hardware. If a 3100/3150/3200 MP system is being created, the option MP must be equated to 32.

Release Value: MSOS COSY tape and MSOS binary source = 0.

BDP Option (BDP3312)

Definition: Allows the BDP Collating Sequence to be assembled at absolute locations 40 to 57 octal.

Option:		DELETE/	15	
	BDP3312	EQU	0 or 1	no BDP hardware or non-enhanced BDP hardware
	BDP3312	EQU	2	enhanced BDP hardware

Dependency: Must have enhanced BDP hardware, to set BDP3312 to 2.

Release Value: MSOS COSY tape and STD Batch sections of the MSOS binary source = 0.

BDP Collating Sequence

Definition: Installations having the enhanced BDP hardware can manually reset the desired priority of their collating sequence.

Option:	DELETE/	76,91
	OCT	65666770,71727374
	OCT	75766526,27010203
	OCT	17313233,34353637
	OCT	40413014,15121316
	OCT	22434445,46475051
	OCT	52534220,21071011
	OCT	00235556,57606162
	OCT	63645424,25040506

Dependency: The BDP collating sequence is only applicable when BDP3312 is equated to 2. The BDP collating sequence is utilized by the enhanced BDP hardware and the numeric and alphanumeric compare instructions.

Release Value: On the MSOS COSY tape the BDP collating sequence is as above. On MSOS binary sources, the BDP collating sequence is not assembled, because BDP3312 is equated to 0.

Deck Card: MZERO DECK/ P_1, P_2

2.3.2 ZERO (STD-BATCH)

BDP Option (BDP3312)

Definition: Allows the BDP collating sequence to be assembled at absolute locations 40 to 57 octal.

Option:		DELETE/	9	
	BDP3312	EQU	0 or 1	for no BDP hardware or non-enhanced BDP hardware
	BDP3312	EQU	2	for enhanced BDP hardware

Dependency: Must have enhanced BDP hardware, to set BDP3312 to 2.

Release Value: MSOS COSY tape and STD Batch sections of the MSOS binary source= 0.

BDP Collating Sequence

Definition: Installations having enhanced BDP hardware can manually reset the desired priority of their collating sequence.

Option:

DELETE/	55,71
OCT	65666770,71727374
OCT	75766526,27010203
OCT	17313233,34353637
OCT	40413014,15121316
OCT	22434445,46475051
OCT	52534220,21071011
OCT	00235556,57606162
OCT	63645424,25040506

Dependency: The BDP collating sequence is only applicable when BDP3312 is equated to 2. The BDP collating sequence is utilized by the enhanced BDP hardware and the numeric and alphanumeric compare instructions.

Release Value: On the MSOS COSY tape the BDP collating sequence is as above. In all sections of the MSOS sources, the BDP collating sequence is not assembled, because BDP3312 is equated to 0.

Deck Card: ZERO	DECK/	P_1, P_2	

2.3.3 CIO (STD-MP)

Dual-Channel Access Option (DP)

Definition: Allows the user to use dual-channel access on a CONTROL DATA® 3234/3232 Disk Storage Controller.

Option:		DELETE/	5	
	DP	EQU	0	dual-channel access not selected
	DP	EQU	1	dual-channel access selected

Dependency: Must have a 3232/3234 disk storage control, and both channels defined in the available equipment table (AET), if DP = 1 (selected).

Release Value: DP is 0 on the MSOS COSY tape and STD and MP sections of the MSOS binary source.

Lost Interrupt Timeout Option (TIMEOUT)

Definition: Permits the routine CIO to detect lost equipment interrupts when selected.

Option:		DELETE/	6	
	TIMEOUT	EQU	0	TIMEOUT not selected
	TIMEOUT	EQU	1	TIMEOUT selected

Dependency: The clock option must be selected in the routine SCICREC1 or MCICREC1, if TIMEOUT is selected.

Release Value: TIMEOUT is 0 on the MSOS COSY tape and STD and MP sections of the MSOS binary source.

Deck Card: CIO	DECK/	P_1, P_2	

2.3.4 BCIO (BATCH)

Dual-Channel Access Option (DP)

Definition: Allows the user to use dual-channel access on a 3232/3234 disk storage controller.

Option:		DELETE/	4	
	DP	EQU	0	dual-channel access not selected
	DP	EQU	1	dual-channel access selected

Dependency: Must have a 3232/3234 disk storage control, and both channels defined in the available equipment table AET.

Release Value: DP is 0 on the MSOS COSY tape and Batch section of the MSOS binary source.

Lost Interrupt Timeout Option (TIMEOUT)

Definition: Permits the routine BCIO to detect lost equipment interrupt when selected.

Option:		DELETE/	5	
	TIMEOUT	EQU	0	TIMEOUT not selected
	TIMEOUT	EQU	1	TIMEOUT selected

Dependency: The clock option must be selected in the routine BCICREC1, if TIMEOUT is selected.

Release Value: TIMEOUT is 0 on the MSOS COSY tape and Batch section of the MSOS binary source.

Deck Card:

BCIO	DECK/	P_1, P_2

2.3.5 MAGNETIC TAPE DRIVERS (DRIV607, DRIV606, DRIV3528-65X (ALL))

Assembly Option

Definition: The WRITEB option permits those installations with the controlled backspace function installed to utilize the function during write recovery.

Option:

WRITEB	EQU	0	Controlled backspace not available
WRITEB	EQU	1	Controlled backspace function installed

Release Value: WRITEB = 0 in the add on decks

Deck Card:

DRIV 607 / 606 / 3528	DECK/	P_1, P_2

Option:

	DELETE/	122
T659	EQU	m

m number of 659 Tape Transports

Release Value: T659=1 on the MSOS COSY tape and MSOS binary source.

Deck Card: DRIV3528 DECK/ P_1, P_2

2.3.6 DRIV3659 - PRINTER DRIVER (ALL)

Number of Line Printers Option (NLPS)

Definition: NLPS defines the number of printers online. For each printer online, a certain number of words must be reserved in the printer parameter storage area. The number of words is determined by NLPS.

Option:	DELETE/	11	
NLPS	EQU	2	two line printers online

Dependency: Number of 3254, 501, or 505 printers online. Note that the same option is contained within both printer drivers and they are independent of one another.

Release Value: NLPS = 2 on the MSOS COSY tape and MSOS binary source. Must be changed only if more than two line printers are online.

Deck Card: DRIV3659 DECK/ P_1, P_2

2.3.7 DRIV3555 - PRINTER DRIVER (ALL)

Number of Line Printers Option (NLPS)

Definition: NLPS defines the number of printers online. For each printer online a certain number of words must be reserved in the printer parameter storage area. The number of words is determined by NLPS.

Option:	DELETE/	7	
NLPS	EQU	2	two line printers online

Dependency: Number of 512 printers online. Note that the same option is contained within both printer drivers and they are independent of one another.

Release Value: NLPS = 2 on the MSOS COSY tape and MSOS binary source. Must be changed only if more then two 512 line printers are online.

Deck Card: DRIV3555 DECK/ P_1, P_2

2.3.8 MSIO (STD-MP)

The routine MSIO is hardware configuration dependent and is not included in the MSOS binary source. The assembled binary deck of MSIO must replace the unit card ($^7_9$UNIT, AA) on the STD or MP sections of MSOS binary source file.

Mass Storage Table Size Option (RATL)

Definition: The option RATL must equal the number of MST entries (n_1).

Option:		DELETE/	23
	RATL	EQU	n_1

Dependency: Number of MST entries.

Release Value: RATL = 4 on the MSOS COSY tape.

File Description Table Length Option (FFDTL/BFDTL)

Definition: The length of the FDT is set by the values of FFDTL for batch files and BFDTL for priority files. These respective tables (FFDT for batch and BFDT for priority) contain the necessary information to process an open file. Each open file has a seven word header plus three words for each segment. (The XFDT size entry = 7 + (3 x segment count of file).)

Option:		DELETE/	25,26	
	FFDTL	EQU	250	batch FDT length
	BFDTL	EQU	125	priority FDT length

Dependency: An installation may change these values if desired.

Release Value: FFDTL = 250_{10} and BFDTL = 125_{10} on the MSOS COSY tape.

Standard Units on Mass Storage File Description Table Length Option (SFDTL)

Definition: The length of the FDT for standard units on mass storage is set by the value SFDTL. SFDTL contains the information necessary to process the standard files on mass storage (INP, OUT, PUN).

Option:		DELETE/	27
	SFDTL	EQU	165

Dependency: Must be equated to 165_{10} words for RESPOND/ MSOS and standard file processor (SFP) processing. If no priority job submission is going to be performed, SFDTL may be set to 0.

Release Value: SFDTL = 165_{10} on the MSOS COSY tape.

Master Security Code Option (MSEC)

Definition: MSEC contains an 8-character master security code which allows access to files described in the label directory.

Option:	DELETE/	41	
	OCT	0,0	master security code

Dependency: Should be changed and used for installation security.

Release Value: The location of MSEC is set to 0,0 on the MSOS COSY tape.

AET Ordinal of System Device (AETD)

Definition: The system device should be the first unit described in the AET. If it is not, the system ordinal must be changed to the decimal ordinal of the system device.

Option:	DELETE/	42	
	DEC	1	AET ordinal of system device

Dependency: System device must be decimal ordinal of library entry in AET.

Release Value: AETD = 1 on the MSOS COSY tape.

Maximum File Size (MFS)

Definition: The MFS is the maximum number of tracks that can be allocated for any one file.

Option:	DELETE/	45	
	DEC	3996	maximum file size

Dependency: May be changed, if desired.

Release Value: MFS = 3996_{10} on the MSOS COSY tape (equivalent to four 853 disks).

Maximum Segment Count (MSC)

Definition: The MSC is the maximum number of segments that can be generated for any one file. See Part III, Section 5.2.11.

Option:	DELETE/	46	
	DEC	16	maximum segment count

Dependency: MSC cannot exceed 63. It must equal the value used during INSTALL.

Release Value: MSC = 16_{10} on the MSOS COSY tape.

Maximum File Count (MFC)

Definition: The MFC limits the number of files that can be maintained by MSIO.

Option:	DELETE/	47	
	DEC	100	maximum file count

Dependency: MFC is limited by the size of IDFILE and LABELFILE which is determined during INSTALL. See Part III, Section 5.2.11.

Release Value: MFC = 100_{10} on the COSY tape.

Mass Storage Table (MST)

Definition: MST consists of a two-word entry for each online mass storage device.

Option:		DELETE/	52,55
	MST	OCT	ao000000, dt000000
		OCT	ao000000, dt000000
		OCT	ao000000, dt000000
		OCT	ao000000, dt000000

Dependency: Hardware configuration.

Release Value: Each MST must be in form that follows.

23   18 17 16   00

ao	s	0	0	0	0	0	0
dt	0	0	0	0	0	0	

ao   6-bit AET ordinal (in octal) of the drive correlated with device entry

s   Status bit is 0 if unit is operable; 1 if unit is down or inoperable.

dt   6-bit device code for this mass storage type

The device types are as follows:

code	meaning
40	1311 Disk Unit
41	CONTROL DATA® 852 Disk Drive
50	CONTROL DATA® 853 Disk Drive
51	CONTROL DATA® 854 Disk Drive
52	CONTROL DATA® 841 Multiple Disk Drive
60	CONTROL DATA® 813 Disk File and 814 Disk File
70	CONTROL DATA® 863 Drum

Deck Card: MSIO   DECK/ P_1, P_2

2.3.9 BMSIO (BATCH)

The routine BMSIO is hardware configuration dependent and is not included on the MSOS binary source. The assembled binary deck of BMSIO must replace the UNIT card ($^7_9$UNIT, AA) on the Batch section of the MSOS binary source.

Mass Storage Table Size Option (RATL)

Definition: The option RATL must equal the number of MST entries (n_1). MSTL must equal twice the number of MST entries (n_2).

Option:		DELETE/	21	
	RATL	EQU	n_1	number of MST entries
	MSTL	EQU	n_2	2 times MST entries

† MSOS does not support a library on 813 or 814 disk files. However, they may be used for file allocation (user files).

Dependency: Number of MST entries.

Release Value: RATL = 4 on the MSOS COSY tape.

File Description Table Length Option (FFDTL)

Definition: The length of the FDT is set by the values of FFDTL for Batch files. The table FFDT contains the necessary information to process an open file. Each open file has a seven word header plus three words for each segment. (The FFDT size entry = 7 + (3 x segment count of file).)

Option:		DELETE/	23	
	FFDTL	EQU	250	batch FDT length

Dependency: May be changed, if desired.

Release Value: FFDTL = 250_{10} on the MSOS COSY tape.

Master Security Code Option (MSEC)

Definition: MSEC contains an eight character master security code which allows access to files described in the label directory.

Option:	DELETE/	38	
	OCT	0,0	master security code

Dependency: Should be changed and used for installation security.

Release Value: The location of MSEC is set to 0,0 on the MSOS COSY tape.

AET Ordinal of System Device (AETD)

Definition: The system device should be the first unit described in the AET. If it is not, the system ordinal must be changed to the decimal ordinal of the system device.

Option:	DELETE/	39	
	DEC	1	AET ordinal of system device

Dependency: System device must be decimal ordinal of library entry in AET.

Release Value: AETD = 1 on the MSOS COSY tape.

Maximum File Size (MFS)

Definition: The MFS is the maximum number of tracks that can be allocated for any one file.

Option:	DELETE/	42	
	DEC	3996	maximum file size

Dependency: May be changed, if desired.

Release Value: MFS = 3996_{10} on the MSOS COSY tape (equivalent to four 853 disks).

Maximum Segment Count (MSC)

Definition: The MSC is the maximum number of segments that can be generated for any one file. See Part III, Section 5.2.11.

Option:	DELETE/	43	
	DEC	16	maximum segment count

Dependency: MSC cannot exceed 63. It must equal the value used during INSTALL.

Release Value: MSC = 16_{10} on the MSOS COSY tape.

Maximum File Count (MFC)

Definition: The MFC limits the number of files that can be maintained by BMSIO.

Option:	DELETE/	44	
	DEC	100	maximum file count

Dependency: MFC is limited by the size of IDFILE and LABELFILE which is determined during INSTALL (Part III, Section 5.2.11).

Mass Storage Table (MST)

Definition: MST consists of a two-word entry for each online drive.

Option:		DELETE/	46,49
	MST	OCT	ao000000,dt000000
		OCT	ao000000,dt000000
		OCT	ao000000,dt000000
		OCT	ao000000,dt000000

Dependency: Hardware configuration.

Release Value: Each entry must be in the form that follows.

23 18	17	16					00
ao	s	0	0	0	0	0	0
dt	0	0	0	0	0	0	

ao 6-bit AET ordinal (octal) of the drive correlated with a device entry

s Status bit is 0 if unit is operable; 1 if unit is down or inoperable.

dt 6-bit device type for this mass storage type

The types are as follows:

code	meaning
40	1311 Disk Unit
41	852 Disk Drive
50	853 Disk Drive
51	854 Disk Drive
52	841 Multiple Disk Drive
60	813 Disk File and 814 Disk File†
70	863 Drum

Deck Card: BMSIO DECK/ P_1, P_2

2.3.10 MCICREC1 (MP)

MCICREC1 is a resident program containing tables and various resident subroutines for a Memory Protect MSOS library. MCICREC1 contains the description of the hardware environment and must be updated to reflect the particular hardware configuration. The routine MCICREC1 is not on the MSOS binary source and must be reassembled and the binary deck put in place of the UNIT card ($^7_9$UNIT,BB) on the MP section of the MSOS binary source.

† MSOS does not support a library on 813 or 814 disk files. However, they may be used for file allocation (user files).

Accounting Option (ACCT)

Definition: The accounting option provides a basic job accounting of batch processing.

Option:				
		DELETE/	109	
	ACCT	EQU	0	accounting not selected
	ACCT	EQU	1	accounting selected

Dependency: If the basic job accounting of batch processing is desired, ACCT must be selected, and the clock interrupt processor (CIP) option must also be selected.

Release Value: ACCT = 0 on the MSOS COSY tape.

BDP Option (BDP)

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:				
		DELETE/	110	
	BDP	EQU	0	no BDP hardware present
	BDP	EQU	1	non-enhanced or enhanced BDP hardware present

Dependency: Must have CDC BDP hardware.

Release Value: BDP = 0 on the MSOS COSY tape.

Clock Interrupt Processor Option (CLOCK)

Definition: The CIP option permits the user to use the clock in a shared manner.

Option:				
		DELETE/	111	
	CLOCK	EQU	0	CIP not selected
	CLOCK	EQU	1	CIP selected

Dependency: If the options ACCT and TIMEOUT are selected, the clock option must be selected.

Release Value: CLOCK = 0 on the MSOS COSY tape.

Floating Point Hardware Option (FDP)

Definition: If the installation has floating point hardware, the option FDP must be selected to utilize it.

Option:				
		DELETE/	112	
	FDP	EQU	0	no FDP hardware present
	FDP	EQU	1	FDP hardware present

Dependency: Floating point hardware.

Release Value: FDP = 0 on the MSOS COSY tape.

Memory Size Option (MAXCORE)

Definition: The memory size of the computer must be properly recorded for system loading.

Option:		DELETE/	113	
	MAXCORE	EQU	77737B	32K for 32/33/3500

Dependency: Each installation must set the value of MAXCORE according to the table below.

Machine Type	32K Value
3100/3150	77637_8
3200/3300/3500	77737_8

Release Value: MAXCORE = 77737B (32/33/3500 computer with 32K of memory) on the MSOS COSY tape.

Memory Protect Option (MP)

Definition: The option MP distinguishes between 3300/3500 Memory Protect and 3100/3150/3200 Memory Protect systems.

Option:		DELETE/	114	
	MP	EQU	0	33/3500 MP systems
	MP	EQU	32	3100/3150/3200 MP systems

Dependency: Machine type (3300/3500 or 3100/3150/3200 computers) and availability of the optional memory protect hardware.

Release Value: MP = 0 on the MSOS COSY tape. For a 3100/3150/3200 Memory Protect system, MP must be equated to 32.

Number of Channels Option (NCHANS)

Definition: The option NCHANS is the controlling parameter to establish a CST and an EST for monitor control.

Option:		DELETE/	115	
	NCHANS	EQU	4	highest channel available is 3

Dependency: Highest available channel number.

Release Value: NCHANS = 4 on the MSOS COSY tape. It must be set to the highest channel available according to the table below.

Highest Available Channel Number	Value to Use
1	2
2 or 3	4
4 or 5	6
6 or 7	8

Resident Directory Table Option (USARDT)

Definition: The Resident Directory Table (RDT) provides the necessary linkage for the routine RDCKF1 to load subprograms from the absolute file.

The assembly option USARDT provides the additional space in the RDT to handle the USASI COBOL compiler when selected.

Option:		DELETE/	116	
	USARDT	EQU	0	NON-USASI COBOL user
	USARDT	EQU	1	USASI COBOL user

Dependency: Must be set to 1 if USASI COBOL is to be on the system.

Release Value: USARDT = 0 on the MSOS COSY tape.

ASCII Input Option (INP)

Definition: The MSOS system has the capability to support the ASCII (3447-2) or Hollerith (3447) card readers. The INP option provides the system with the proper conversion sequence.

Option:		DELETE/	121
	INP	EQU	ASCII
	INP	EQU	HOLL

Dependency: Must be set to ASCII if input punched in ASCII format.

Release Value: INP=HOLL on MSOS COSY tape.

ASCII Punch Option (PUN)

Definition: The MSOS system has the capability to support the ASCII (3446-2) or Hollerith (3446) card punches. The option PUN provides the system with the proper conversion sequence.

Option:		DELETE/	123
	PUN	EQU	ASCII
	PUN	EQU	HOLL

Dependency: Must be set to ASCII if output in ASCII format is required.

Release Value: PUN=HOLL on MSOS COSY tape.

ASCII Paper Tape Option (PT)

Definition: The MSOS system has the capability to punch paper tape in both ASCII and Flexowriter codes. The option PT provides the system with the proper conversion sequence.

Option:		DELETE/	124	
	PT	EQU	ASCII	
	PT	EQU	HOLL	(FLEXOWRITER)

Dependency: Must be set to ASCII if output in ASCII format is required.

Release Value: PT=HOLL on MSOS COSY tape.

Available Equipment Table Entries Option (AET)

Definition: The AET must define the equipment available to the system for each installation. This section describes only the option. See AET, AET format, macro description, and driver table in Part III, Section 6.

Option: DELETE/ 3271,3287

Dependency: Hardware installation dependent. The library should be defined as the first entry in the AET. If it is not, the location AETD in MSIO must be changed. It is recommended that the order of the system units be as follows to alleviate changing the RHT.

AET Ordinal	System Unit
1	Library
2	CTO/CFO
3	INP
4	OUT
5	PUN
6	Available units

Release Value: Hardware dependent.

Running Hardware Table Option (RHT)

Definition: The RHT defines the equipment used during a batch run and equipment assigned as system units (LIB, CTO/CFO, INP, OUT, PUN). The complete description is in part III, section 7.1.

Option:	DELETE/	3354,3360
RHT		modifications

Dependency: Available hardware, and order of the entries in the AET. The RHT need not be changed if the order of the system units are in the order as described under the AET options.

Release Value: See part III, section 7.1.

Priority Running Hardware Table Option (BRHT)

Definition: The BRHT is similar to the RHT, but has only logical units 1-49, 58, and 59 available for reference by a priority program. A complete description is given in part III, section 7.2.

Option:	DELETE/	3368,3371
BRHT		modification

Dependency: Available hardware for use by a priority program.

Release Value: See part III, section 7.2.

Central Interrupt Table Option (CIT)

Definition: The CIT reflects the channels available and not available to the input/output section.

Option:	DELETE/	3438,3445	
	00	x	ch 0
	00	x	ch 1
	00	x	ch 2
	00	x	ch 3
	00	x	ch 4
	00	x	ch 5
	00	x	ch 6
	00	x	ch 7

x = IOIP if channel is available as system channel defined in AET.
x = ABINRT if channel is not available.

Dependency: Hardware dependent.

Release Value: Channels 0, 1, 2, 3 = IOIP and channels 4, 5, 6, 7 = ABINRT on the MSOS COSY tape.

System Interrupt Mask Option (SYST.IOM)

Definition: The mask SYST.IOM reflects the channels available for system use.

Option:		DELETE/	3481
	SYST.IOM	OCT	17

Dependency: The format of the mask word is:

23	08 07	06	05	04	03	02	01	00
	c_7	c_6	c_5	c_4	c_3	c_2	c_1	c_0

c = bit corresponding to channel i

If the channel is available to the system, its corresponding bit is set to 1. If the channel is not available, the corresponding bit is set to 0. Dedicated channels (channels reserved for special applications such as RESPOND) are not represented; they are set to 0.

Release Value: SYST.IOM = 17_8 on the MSOS COSY tape.

Deck Card: MCICREC1 DECK/ P_1, P_2

2.3.11 SCICREC1 (STD)

SCICREC1 is a resident program containing tables and various resident subroutines for a STANDARD MSOS library. SCICREC1 contains the description of the hardware environment and must be updated to reflect the particular hardware configuration. The routine SCICREC1 is hardware configuration dependent and all installations must reassemble this routine according to its configuration and place the binary deck obtained in place of the UNIT card (${}^7_9$UNIT, BB) on the STD section of the MSOS binary source.

Accounting Option (ACCT)

Definition: The accounting option provides a basic job accounting of batch processing.

Option:		DELETE/	81	
	ACCT	EQU	0	accounting not selected
	ACCT	EQU	1	accounting selected

Dependency: If the basic job accounting of batch processing is desired, ACCT must be selected, and the clock interrupt processor (CIP) option must also be selected.

Release Value: ACCT = 0 on the MSOS COSY tape.

BDP Option (BDP)

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:	DELETE/	82	
BDP	EQU	0	no BDP hardware present
BDP	EQU	1	non-enhanced or enhanced BDP hardware present

Dependency: Must have CDC BDP hardware.

Release Value: BDP = 0 on the MSOS COSY tape.

Clock Interrupt Processor Option (CLOCK)

Definition: The clock option permits the user to use the clock in a shared manner.

Option:	DELETE/	83	
CLOCK	EQU	0	CIP not selected
CLOCK	EQU	1	CIP selected

Dependency: If the options ACCT and TIMEOUT are selected, the clock option must be selected.

Release Value: CLOCK = 0 on the MSOS COSY tape.

Memory Size Option (MAXCORE)

Definition: The memory size of the computer must be properly recorded for system loading.

Option:	DELETE/	84	
MAXCORE	EQU	77737B	32K for 32/33/3500

Dependency: Each installation must set the value of MAXCORE according to the table below.

Machine Type	32K Value	16K Value
3100/3150	77637_8	37637_8
3200/3300/3500	77737_8	37737_8

Release Value: MAXCORE = 77737B (32/33/3500 computer with 32K of memory) on the MSOS COSY tape.

Number of Channels Option (NCHANS)

Definition: The option NCHANS is the controlling parameter to establish a CST (Channel Status Table) and an EST (Equipment Status Table) for monitor control.

Option:	DELETE/	85	
NCHANS	EQU	4	highest channel available is 3

Dependency: Highest available channel number.

Release Value: NCHANS = 4 on the MSOS COSY tape. It must be set to the highest channel available according to the table below.

Highest Available Channel Number	Value to Use
1	2
2 or 3	4
4 or 5	6
6 or 7	8

SETCLOCK Option (SETCLK)

Definition: The SETCLOCK routine permits installations that used SETCLOCK under previous versions to continue to use the clock without changing their programs. Installations that have not used SETCLOCK previously, but wish to include the feature, should use CIP to save core.

Option:		DELETE/	86	
	SETCLK	EQU	0	not selected
	SETCLK	EQU	1	selected

Dependency: Installation option.

Release Value: SETCLK = 0 on the COSY tape.

Resident Directory Table Option (USARDT)

Definition: The Resident Directory Table (RDT) provides the necessary linkage for the routine RDCKF1 to load subprograms from the absolute file.

The assembly option USARDT provides the additional space in the RDT to handle the USASI COBOL compiler when selected.

Option:		DELETE/	87	
	USARDT	EQU	0	NON-USASI COBOL user
	USARDT	EQU	1	USASI COBOL user

Dependency: Must be set to 1, if USASI COBOL is to be on the system.

Release Value: USARDT = 0 on the MSOS COSY tape.

ASCII Input Option (INP)

Definition: The MSOS system has the capability to support the ASCII (3447-2) or Hollerith (3447) card readers. The option INP provides the system with the proper conversion sequence.

Option:		DELETE/	92
	INP	EQU	ASCII
	INP	EQU	HOLL

Dependency: Must be set to ASCII if input punched in ASCII format.

Release Value: INP=HOLL on MSOS COSY tape.

ASCII Punch Option (PUN)

Definition: The MSOS system has the capability to support the ASCII (3446-2) or Hollerith (3446) card punches. The option PUN provides the system with the proper conversion sequence.

Option:		DELETE/	94
	PUN	EQU	ASCII
	PUN	EQU	HOLL

Dependency: Must be set to ASCII if output in ASCII format is required.

Release Value: PUN=HOLL on MSOS COSY tape.

ASCII Paper Tape Option (PT)

Definition: The MSOS system has the capability to punch paper tape in both ASCII and Flexowriter codes. The option PT provides the system with the proper conversion sequence.

Option:		DELETE/	95
	PT	EQU	ASCII
	PT	EQU	HOLL (FLEXOWRITER)

Dependency: Must be set to ASCII if output in ASCII format is required.

Release Valuse: PT=HOLL on MSOS COSY tape.

Available Equipment Table Entries Option (AET)

Definition: The AET must define the equipment available to the system for each installation. See AET, AET format, macro description, and driver table in Part III, Section 6.

Option:		DELETE/	2207,2223

Dependency: Hardware installation dependent. The library should be defined as the first entry in the AET. If it is not, the location AETD in MSIO must be changed. It is recommended that the order of the system units be as follows to alleviate changing the RHT.

AET Ordinal	System Unit
1	Library
2	CTO/CFO
3	INP
4	OUT
5	PUN
6 etc.	Available units

Release Value: Hardware dependent.

Running Hardware Table Option (RHT)

Definition: The RHT defines the equipment used during a batch run and equipment assigned as system units (LIB, CTO/CFO, INP, OUT, PUN). The complete description is in part III, section 7.1.

Option:	DELETE/	2288,2294
RHT		modifications

Dependency: Available hardware and order of the entries in the AET. The RHT need not be changed if the order of the system units are in the order as described under the AET options.

Release Value: See part III, section 7.1.

Priority Running Hardware Table Option (BRHT)

Definition: The BRHT is similar to the RHT, but has only logical units 1-49, 58 and 59 available for reference by a priority program. A complete description is given in part III, section 7.2.

Option:	DELETE/	2302,2305
BRHT		modification

Dependency: Available hardware used by a priority program.

Release Value: See part III, section 7.2.

Central Interrupt Table Option (CIT)

Definition: The CIT reflects channels available and not available to the input/output section.

Option:	DELETE/	2372, 2379	
	00	x	ch 0
	00	x	ch 1
	00	x	ch 2
	00	x	ch 3
	00	x	ch 4
	00	x	ch 5
	00	x	ch 6
	00	x	ch 7

x = IOIP if channel is available as system channel defined in AET.
x = ABINRT if channel is not available.

Dependency: Hardware dependent.

Release Value: Channels 0, 1, 2, 3 = IOIP and channels 4, 5, 6, 7 = ABINRT on the MSOS COSY tape.

System Interrupt Mask Option (SYST. IOM)

Definition: The mask SYST. IOM reflects the channels available for system use.

Option:		DELETE/	2412
	SYST. IOM	OCT	17

Dependency: The format of the mask word is:

23 … 08	07	06	05	04	03	02	01	00
	c_7	c_6	c_5	c_4	c_3	c_2	c_1	c_0

c = bit corresponding to channel i

If the channel is available to the system, the corresponding bit is set to 1. If the channel is not available, the corresponding bit is set to 0. Dedicated channels (channels reserved for special applications such as RESPOND) are not represented. They are set to 0.

Release Value: SYST. IOM = 17_8 on the MSOS COSY tape.

Deck Card: SCICREC1 DECK/ P_1, P_2

2.3.12 BCICREC1 (BATCH)

BCICREC1 is a resident program containing tables and various resident subroutines for a Batch MSOS library. BCICREC1 contains the description of the hardware environment and must be updated to reflect the particular hardware configuration. The routine BCICREC1 must be reassembled according to the site configuration and the binary deck put in place of the UNIT card (${}^{7}_{9}$UNIT, BB) on the Batch section of the MSOS binary source.

<u>Accounting Option (ACCT)</u>

Definition: The accounting option provides a basic accounting of batch processing.

Option:		DELETE/	81	
	ACCT	EQU	0	accounting not selected
	ACCT	EQU	1	accounting selected

Dependency: If the basic job accounting of batch processing is desired, ACCT must be selected, and the clock interrupt processor (CIP) option must also be selected.

Release Value: ACCT = 0 on the MSOS COSY tape.

<u>BDP Option (BDP)</u>

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:		DELETE/	82	
	BDP	EQU	0	no BDP hardware present
	BDP	EQU	1	non-enhanced or enhanced BDP hardware present

Dependency: Must have CDC BDP hardware. Applicable to 3300/3500 computers only.

Release Value: BDP = 0 on the MSOS COSY tape.

<u>Clock Interrupt Processor Option (CLOCK)</u>

Definition: The clock option permits the user to use the clock in a shared manner.

Option:		DELETE/	83	
	CLOCK	EQU	0	CIP not selected
	CLOCK	EQU	1	CIP selected

Dependency: If the options ACCT and TIMEOUT are selected, the clock option must be selected.

Release Value: CLOCK = 0 on the MSOS COSY tape.

Memory Size Option (MAXCORE)

Definition: The memory size of the computer must be properly recorded for system loading.

Option:		DELETE/	84	
	MAXCORE	EQU	77737B	32K for 32/33/3500

Dependency: Each installation must set the value of MAXCORE according to the table below.

Machine Type	32K Value	16K Value
3100/3150	77637_8	37637_8
3200/3300/3500	77737_8	37737_8

Release Value: MAXCORE = 77737_8 (32/33/3500 computer with 32K of memory) on the MSOS COSY tape.

Number of Channels Option (NCHANS)

Definition: The option NCHANS is the controlling parameter to establish a Channel Status Table (CST) and an Equipment Status Table (EST) for monitor control.

Option:		DELETE/	85	
	NCHANS	EQU	4	highest channel available is 3

Dependency: Highest available channel number.

Release Value: NCHANS = 4 on the MSOS COSY tape. It must be set to the highest channel available according to the table below.

Highest Available Channel Number	Value to Use
1	2
2 or 3	4
4 or 5	6
6 or 7	8

SETCLOCK Option (SETCLK)

Definition: The SETCLOCK routine permits installations that used SETCLOCK under previous versions to continue to use the clock without changing their programs. Installations that have not used SETCLOCK previously, but wish to include the feature, should use CIP to save core.

Option:		DELETE/	86	
	SETCLK	EQU	0	not selected
	SETCLK	EQU	1	selected

Dependency: Installation option.

Release Value: SETCLK = 0 on the MSOS COSY tape.

Resident Directory Table (RDT) Option (USARDT)

Definition: RDT provides the necessary linkages for the routine RDCKF1 to load subprograms from the absolute file.

The assembly option USARDT provides the additional space in the RDT to handle the USASI COBOL compiler when selected.

Option:		DELETE/	87	
	USARDT	EQU	0	NON-USASI COBOL user
	USARDT	EQU	1	USASI COBOL user

Dependency: Must be set to 1 if USASI COBOL is to be on the system.

Release Value: USARDT = 0 on the COSY tape.

ASCII Input Option (INP)

Definition: The MSOS system has the capability to support the ASCII (3447-2) or Hollerith (3447) card readers. The option INP provides the sytem with the proper conversion sequence.

Option:		DELETE/	92
	INP	EQU	ASCII
	INP	EQU	HOLL

Dependency: Must be set to ASCII if input punched in ASCII format.

Release Value: INP=HOLL on MSOS COSY tape.

ASCII Punch Option (PUN)

Definition: The MSOS system has the capability to support the ASCII (3446-2) or Hollerith (3446) card punches. The option PUN provides the system with the proper conversion sequence.

Option:		DELETE/	94
	PUN	EQU	ASCII
	PUN	EQU	HOLL

Dependency: Must be set to ASCII if output in ASCII format is required.

Release Value: PUN = HOLL on MSOS COSY tape.

ASCII Paper Tape Option (PT)

Definition: The MSOS system has the capability to punch paper tape in both ASCII and Flexowriter codes. The option PT provides the system with the proper conversion sequence.

Option:		DELETE/	95	
	PT	EQU	ASCII	
	PT	EQU	HOLL	(FLEXOWRITER)

Dependency: Must be set to ASCII if output in ASCII format is required.

Release Value: PT=HOLL on MSOS COSY tape.

Input/Output Table Option (IOT)

Definition: The IOT reflects channels available and not available to the input/output section.

Opinion:		DELETE/	805,812	
	IOT	00	x	ch 0
		00	x	ch 1
		00	x	ch 2
		00	x	ch 3
		00	x	ch 4
		00	x	ch 5
		00	x	ch 6
		00	x	ch 7

x = IOIP if channel is available as system channel defined in AET
x = ABINRT if channel is not available

Dependency: Hardware dependent.

Release Value: Channels 0,1,2,3 = IOIP and channels 4,5,6,7 = ABINRT on the MSOS COSY tape.

Available Equipment Table Entries Option (AET)

Definition: The AET must define the equipment available to the system for each installation. This section only describes the option. See AET, AET format, macro description, and driver table in Part III, Section 6.

Option:	DELETE/	1499,1515
AET		modifications

Dependency: Hardware installation dependent. The library should be defined as the first entry in the AET. If it is not, the location AETD in MSIO must be changed. It is recommended that the order of the system units be as follows to alleviate changing the RHT:

AET Ordinal	System Unit
1	Library
2	CTO/CFO
3	INP
4	OUT
5	PUN
6 etc.	Available units

Release Value: Hardware dependent.

Running Hardware Table Option (RHT)

Definition: The RHT defines the equipment used during a batch run and equipment assigned as system units (LIB, CTO/CFO, INP, OUT, PUN). The complete description is in part III, section 7.1.

Option:	DELETE/	1580,1586
RHT		modifications

Dependency: Available hardware and order of the entries in the AET. The RHT need not be changed if the order of the system units are in the order as described under the AET options.

Release Value: part III, section 7.1.

System Interrupt Mask Option (SYST.IOM)

Definition: The mask SYST.IOM reflects the channels available for system use.

Option:	DELETE/	1625
SYST.IOM	OCT	17

Dependency: The format of the mask word is:

23 ...	07	06	05	04	03	02	01	00
	c_7	c_6	c_5	c_4	c_3	c_2	c_1	c_0

c = bit corresponding to channel i

If the channel is available to the system, the corresponding bit is set to 1. If the channel is not available, the corresponding bit is set to 0. Dedicated channels (channels reserved for special applications such as RESPOND) are not represented. They are set to 0.

Release Value: SYST.IOM = 17_8 on the MSOS COSY tape.

Deck Card: BCICREC1 DECK/ P_1, P_2

2.3.13 RDUMP (ALL)

Memory Protect Option (MP)

Definition: The option MP distinguishes between a STD/Batch system and a Memory Protect system.

Option:		DELETE/	3	
	MP	EQU	0	STD/Batch system
	MP	EQU	1	MP system

Dependency: MP hardware.

Release Value: MP = 0 on the COSY tape and STD/Batch section of the MSOS binary source. MP = 1 on MP section of the MSOS binary source.

Standard Error Recovery Option (SER.OPT)

Definition: The option SER.OPT when set, assembles the standard error recovery routines into RDUMP for magnetic tape and printer recovery.

Option:		DELETE/	4	
	SER.OPT	EQU	0	non-standard error recovery
	SER.OPT	EQU	1	standard error recovery

Dependency: If SER.OPT is set to 1, error recovery equates TAPES, CARDRDR, CARDPNCH, PRINTER, ATTMPT1, ATTMPT2, BKSPLIM2, NOATMPTS, MAXERASE, and CKSMCNST are valid. See Section 2.5 on error recovery assembly options.

Release Value: SER.OPT = 0 on the MSOS COSY tape and all (STD/MP/Batch) sections of the MSOS binary source.

Deck Card: RDUMP DECK/ P_1, P_2

2.3.14 PRELOAD (ALL)

To assemble the PRELOAD routine the system scratch file 55 must be 250 tracks or larger. File 55 on an established interim library is 150 tracks. See Sections 2.3 through 2.5 for error recovery assembly options in PRELOAD.

Memory Protect Option (MP)

Definition: The option MP distinguishes between a STD/Batch system and a Memory Protect system.

Option:

		DELETE/	2	
	MP	EQU	0	STD/Batch system
	MP	EQU	1	MP system

Dependency: MP hardware.

Release Value: MP = 0 on the COSY tape and on the STD/Batch sections of the MSOS binary source. MP = 1 on the MP section of the MSOS binary source.

Batch Flag Option (BF)

Definition: The option BF distinguishes between a Batch system and a STD/MP system.

Option:

		DELETE/	3	
	BF	EQU	0	STD/MP system
	BF	EQU	1	batch system

Dependency: User option.

Release Value: BF = 0 on the COSY tape and STD/MP sections of the MSOS binary source. BF = 1 on the Batch section of the binary source.

Accounting Option (ACCT)

Definition: The accounting option provides a basic job accounting of batch processing.

Option:

		DELETE/	4	
	ACCT	EQU	0	accounting not selected
	ACCT	EQU	1	accounting selected

Dependency: If ACCT is selected, the clock interrupt processor (CIP) option must also be selected. The setting of ACCT must match the value of ACCT in MCICREC1, SCICREC1, or BCICREC1.

Release Value: ACCT = 0 on the COSY tape and ALL (STD/MP/Batch) PRELIB sources.

For applicable standard error recovery assembly options, see the section on error recovery assembly options for the routine PRELOAD.

512 Line Printer Option (TRAIN) - With ASCII Support

Definition: If the hardware configuration contains a 512 line printer and the user desires the ability to change the trains, the option TRAIN must be selected.

Option:

	DELETE/	5	
TRAIN	EQU	n	highest number of train available

Dependency: Hardware availability

Release Value: TRAIN = 0 on the MSOS COSY tape and MSOS binary source.

Train Type Option (TRN595x)

Definition: The following train options are only applicable if the above option, TRAIN, is selected. The user must specify the trains which he wishes to use.

Option:

Users with a 595-1, 63-character, 501-compatible train cartridge must set the option as follows:

	DELETE/	241
TRN5951	EQU	1

Users with a 595-2, 48-character, AN-type train cartridge must set the option as follows:

	DELETE/	242
TRN5952	EQU	1

Users with a 595-3, 48-character, HN-type train cartridge must set the option as follows:

	DELETE/	243
TRN5953	EQU	1

Users with a 595-4, 64-character, ASCII support train cartridge must set the following:

	DELETE/	244	train 595-4 available
TRN5954	EQU	1	

Users with additional train cartridges (maximum total of 9) must add them in the following manner:

	INSERT/	260
	ENA	BUF595x

x train numer

Train numbers must be assigned in numerical order following the highest numbered standard train.

With each additional train defined in the above manner, the associated 144 word buffer (image memory) must also be supplied.

	INSERT/	843	
BUF595x	OCT	...	(x=train number)
	(144-word buffer)		

Release value: TRAIN=0, TRN5951=0, TRN5952=0, TRN5953=0, TRN5954=1 on the MSOS COSY tape and all sections of the MSOS binary source.

Deck Card:

PRELOAD	DECK/	P_1, P_2

2.3.15 LOADER (ALL)

Memory Protect Option (MP)

Definition: The option MP distinguishes between a STD/Batch system and a Memory Protect system.

Option:		DELETE/	2	
	MP	EQU	0	STD/Batch system
	MP	EQU	1	MP system

Dependency: MP hardware.

Release Value: MP = 0 on the MSOS COSY tape and STD/Batch sections of the MSOS binary source. MP = 1 on the MP section of the MSOS binary source.

Standard Error Recovery Option (LERP)

Definition: The option, LERP, when set, assembles the standard error recovery routines into the loader (LOADER) for magnetic tape, card reader, card punch, and printer.

Option:		DELETE/	3	
	LERP	EQU	0	non-standard error recovery
	LERP	EQU	1	standard error recovery

Dependency: When LERP is set to 1 (selected), see the sections on error recovery assembly options (Sections 2.3 through 2.5).

Release Value: LERP = 0 on the MSOS COSY tape and on ALL (STD/MP/Batch) sections of the MSOS binary source.

USASI Option (USASI)

Definition: The USASI option permits MSOS to accept multiple labeled data programs generated by the USASI compiler.

Option:	DELETE/	4	
USASI	EQU	0	not selected
USASI	EQU	1	selected

Dependency: The USASI option must be selected if the final library is to contain a USASI compiler.

If selected, the same option must also be selected in the routines OVPRO and POSTLOAD.

Release Value: USASI = 0 on the MSOS COSY tape and all sections of the MSOS binary source.

Deck Card: LOADER DECK/ P_1, P_2

2.3.16 OVPRO (ALL)

USASI Option (USASI)

Definition: The USASI option permits MSOS to accept multiple labeled data blocks generated by the USASI compiler.

Option:	DELETE/	3	
USASI	EQU	0	not selected
USASI	EQU	1	selected

Dependency: The USASI option must be selected if the final library is to contain a USASI compiler.

Release Value: USASI = 0 on the MSOS COSY tape and all sections of the MSOS binary source.

Deck Card: OVPRO DECK/ P_1, P_2

2.3.17 POSTLOAD (ALL)

Memory Protect Option (MP)

Definition: The option MP distinguishes between a MP system and a STD/Batch system.

Option:	DELETE/	2	
MP	EQU	0	STD/Batch system
MP	EQU	1	MP system

Dependency: MP hardware available.

Release Value: MP = 0 on the MSOS COSY tape and STD/Batch sections of the MSOS binary source. MP = 1 on the MP section of the MSOS binary source.

Standard Error Recovery Option (LERP)

Definition: The option LERP, when selected, assembles the standard error recovery routines into POSTLOAD for magnetic tape, card reader, card punch, and printer.

Option:		DELETE/	3	
	LERP	EQU	0	non-standard error recovery (not selected)
	LERP	EQU	1	standard error recovery (selected)

Dependency: When LERP is set to 1 (selected), see the section error recovery assembly options (Sections 2.3 through 2.5) for valid error recovery equates. When standard error recovery is selected in POSTLOAD, it must be selected in LOADER.

Release Value: LERP = 0 on the MSOS COSY tape and on ALL (STD/MP/Batch) Sections of the MSOS binary source.

USASI Option (USASI)

Definition: The USASI option permits MSOS to accept multiple labeled data blocks generated by the USASI compiler.

Option:		DELETE/	4	
	USASI	EQU	0	not selected
	USASI	EQU	1	selected

Dependency: The USASI option must be selected if the final library is to contain a USASI compiler.

If selected, the same option must also be selected in the routines OVPRO and LOADER.

Release Value: USASI = 0 on the MSOS COSY tape and all sections of the MSOS binary source.

Deck Card: POSTLOAD DECK/ P_1, P_2

2.3.18 INITIAL (ALL)

Memory Protect Option (MP)

Definition: The option MP allows the routine INITIAL to be assembled for a STD/Batch, 3300/3500 Memory Protect, or 3100/3150/3200 Memory Protect systems.

Option:	DELETE/	2	
MP	EQU	0	STD/Batch systems
MP	EQU	1	3300/3500 MP system
MP	EQU	32	3100/3150/3200 MP system

Dependency: Must be modified for a 3100/3150/3200 Memory Protect system.

Release Value: MP = 0 on the MSOS COSY tape and STD/Batch sections of the MSOS binary source. MP = 1 on the MP section of the MSOS binary source.

Deck Card: INITIAL DECK/ P_1, P_2

2.3.19 OC (COMMON ABS and COMMON REL)

There are two copies of this routine on MSOS binary source. If the routine OC is reassembled because of an installation option, both copies must be replaced.

Resident Allocation Table Length Option (REQ)

Definition: The option REQ equals two times the Resident Allocation Table length (RATL in MSIO or BMSIO). REQ = 2 x RATL.

Option:	DELETE/	77	
REQ	BSS	20	RATL in MSIO or BMSIO equals 10

Dependency: Size of RATL in MSIO or BMSIO. Only if RATL in MSIO or BMSIO exceeds 10, must the value of REQ be changed to equal 2 times RATL.

Release Value: REQ equals 20 on the MSOS COSY tape and COMMON ABS and COMMON REL sections of the MSOS binary source.

Mass Storage Table Length Option (NRMST)

Definition: The option NRMST represents the value of MSTL in MSIO or BMSIO.

Option:	DELETE/	78	
NRMST	BSS	18	MSTL equals 18

Dependency: Only if MSTL in MSIO or BMSIO exceeds 18, must the value of NRMST be changed such that NRMST equals MSTL.

Release Value: NRMST equals 18 on the COSY tape and ALL PRELIB sources. If the routine OC is reassembled because of an installation option, both copies on the PRELIB source must be replaced.

Deck Card: OC DECK/ P_1, P_2

2.3.20 AREM (COMMON ABS and COMMON REL)

There are two copies of this routine on the MSOS binary source. If the routine AREM is reassembled because of an installation option, both copies must be replaced.

Resident Allocation Table Length Option (SPACE)

Definition: The option SPACE is five times the Resident Allocation Table length in MSIO or BMSIO.

Option:		DELETE/	78	
	SPACE	BSS	50	RATL equals 10

Dependency: Only if RATL in MSIO or BMSIO exceeds 10, must the value of SPACE be changed so that SPACE equals five times RATL.

Release Value: SPACE equals 50 on the MSOS COSY tape and COMMON ABS and COMMON REL sections of the MSOS binary source.

Deck Card: AREM DECK/ P_1, P_2

2.3.21 BACKGROUND SIMULTANEOUS PERIPHERAL PROCESSOR (BSIPP) (COMMON REL)

Number of Operations Option (NOPS)

Definition: The maximum number of operations that may be executed simultaneously is the value assigned to the nonrelocatable symbol NOPS. To change the limit of allowable simultaneous operations, change the value of NOPS and reassemble the routine. Also change various address lists and multidimensioned array sizes as indicated below.

Options:		DELETE/	98	
	NOPS	EQU	3	three operations permitted

Expand the following tables so they contain the number of entries NOPS is equated to. The example shown is for NOPS = 3.

	DELETE/	2960, 2962
BLIST	00	BUF
	00	BUF+BSIZ2
	00	BUF+BSIZ2+BSIZ2
	DELETE/	2963, 2965
B2LIST	00	B2BUF, 3
	00	B2BUF+B2SIZ, 3
	00	B2BUF+B2SIZ+B2SIZ, 3
	DELETE/	2967, 2969
RSVLIST	00	RSVBUF, 3
	00	RSVBUF+RSVSIZ, 3
	00	RSVBUF+RSVSIZ+RSVSIZ, 3
	DELETE/	2970, 2972
FALIST	00	FABUF
	00	FABUF+FASIZ
	00	FABUF+FASIZ+FASIZ

Change the buffer storage area. The example below is the same as the release (NOPS equal to 3).

	DELETE/	2975, 2978	
BUF	BSS	BSIZ2+BSIZ2+BSIZ2	(BSIZ2) (NOPS)
B2BUF	BSS	B2SIZ+B2SIZ+B2SIZ	(B2SIZ) (NOPS)
RSVBUF	BSS	RSVSIZ+RSVSIZ+RSVSIZ	(RSVSIZ) (NOPS)
FABUF	BSS	FASIZ+FASIZ+FASIZ	(FASIZ) (NOPS)

Dependency: Changing NOPS changes the number of storage buffers reserved. Consequently, the limit depends entirely on the amount of storage available.

Release Value: NOPS = 3 on the MSOS COSY tape and MSOS binary source.

Deck Card: BSIPP DECK/ P_1, P_2

2.3.22 SIMULTANEOUS PERIPHERAL PROCESSOR (SIPP) MODIFICATION (COMMON REL)

Number of Operation Options (NOPS)

Definition: The maximum number of operations that may be executed simultaneously is the value assigned to the nonrelocatable symbol NOPS.

Option:		DELETE/	120
	NOPS	EQU	3

Dependency: On the amount of storage available.

Release Value: NOPS = 3 on the MSOS COSY tape and MSOS binary source.

Deck Card: SIPP DECK/ P_1, P_2

2.3.23 COMPASS (COMMON ABS)

BDP3312 Option (BDP3312)

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:		DELETE/	21	
	BDP3312	EQU	0 or 1	no BDP hardware present or non-enhanced BDP hardware present
	BDP3312	EQU	2	enhanced BDP hardware present

Dependency: BDP hardware type.

Release Value: BDP3312 = 1 on the MSOS COSY tape and the COMMON ABS section of the MSOS binary source.

Deck Card: PASSONE DECK/ P_1, P_2

2.3.24 MS FORTRAN - BCDOUT (COMMON REL)

ZROSUPOC Option

Definition: The option ZROSUPOC permits the user to suppress leading zeros in BCD output.

Option:		DELETE/	4	
	ZROSUPOC	EQU	0	No octal leading zero suppress
	ZROSUPOC	EQU	$\neq 0$	Suppress leading zeros

Release Value: ZROSUPOC equals 0 on the MSOS COSY tape and the COMMON REL section of the MSOS binary source.

MZEROSUP Option

Definition: The option MZEROSUP allows the user to print negative zero as zero in BCD output.

Option:		DELETE/	5	
	MZEROSUP	EQU	0	Allows -0 to be printed as -0
	MZEROSUP	EQU	$\neq 0$	Prints -0 as 0

Release Value: The option MZEROSUP = 0 on the MSOS COSY tape and the COMMON REL section of the MSOS binary source.

Deck Card: BCDOUT DECK/ P_1, P_2

2.3.25 LOGICAL MSIO (BCD and BDP)

BDP Option (BDP3312)

Definition: The option BDP3312 permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:		DELETE/	3	
	BDP3312	EQU	0	no BDP hardware present
	BDP3312	EQU	1	non-enhanced or enhanced BDP hardware present

Dependency: Must have CDC BDP hardware.

Release Value: BDP3312 equals 0 on the MSOS COSY tape and BCD section of the MSOS binary source.

BDP3312 equals 1 on the BDP section of the binary source.

Deck Card: MSIOGPRW DECK/ P_1, P_2

2.3.26 MASS STORAGE COBOL (BDP)

The COSY deck names for BCD and BDP variants of Mass Storage COBOL are different on the MSOS COSY Tape, but the IDC names on binary decks are the same. All options described below are for BDP COBOL. Any new binary decks generated should replace decks in the BDP section of the MSOS binary source.

<u>BDP3312 Option (BDP3312)</u>

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:		DELETE/	4	
	BDP3312	EQU	1	non-enhanced BDP hardware present
	BDP3312	EQU	2	enhanced BDP hardware present
Deck Card:	PCOBOLP2	DECK/	P_1 , P_2	

Dependency: BDP hardware type

Release Value: BDP3312 = 1 on the MSOS COSY tape and BDP section of the MSOS binary source.

NOTE

The new binary deck will replace COBOLP2 in BDP section of the composite PRELIB source.

<u>FDP Option</u>

Definition: The FDP option permits those installations with CDC FDP hardware to utilize the hardware through software modifications.

NOTE

The new binary deck will replace MULTIPLY in BDP section of the composite PRELIB source.

Option:		DELETE/	9	
	FDP	EQU	0	FDP hardware not present
	FDP	EQU	1	FDP hardware present
Deck Card:	PMULTIPL	DECK/	P_1 , P_2	

Dependency: FDP hardware

Release Value: FDP = 1 on the MSOS COSY tape and BDP section of the MSOS binary source.

BDP Option (BDP3312)

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:		DELETE/	18	
	BDP3312	EQU	1	non-enhanced BDP hardware present
	BDP3312	EQU	2	enhanced BDP hardware present
Deck Card:	PVARAN	DECK/	P_1, P_2	

Dependency: BDP hardware type.

Release Value: BDP3312 = 1 on the MSOS COSY tape and BDP section of the MSOS binary source.

NOTE

The new binary deck will replace VARAN in BDP section of the composite PRELIB source.

2.3.27 MS SORT (BCD and BDP)

BDP Option (BDP3312)

Definition: The option BDP3312 allows the user optimum efficiency on moves.

Option		DELETE/	9	
	BDP3312	EQU	0	no BDP hardware present
	BDP3312	EQU	1	non-enhanced BDP hardware present
	BDP3312	EQU	2	enhanced BDP hardware present

Dependency: BDP hardware type.

Release Value: BDP3312 equals 0 on the MSOS COSY tape and BCD sections of the MSOS binary source.

BDP3312 equals 1 on the BDP section of the MSOS binary source.

Memory Protect Option (MEMPTKT)

Definition: The option MEMPTKT assembles the necessary code in MSSORT according to BDP hardware availability and the type of system being created.

Option:		DELETE/	12	
	MEMPTKT	EQU	0	for a STD or Batch system or under MP system with BDP hardware
	MEMPTKT	EQU	2	for a MP system without BDP hardware

Dependency: System type and BDP hardware availability.

Release Value: MEMPTKT equals 0 on the MSOS COSY tape and BCD and BDP sections of the MSOS binary source.

NOTE

This option must be modified for a memory protect system with BDP hardware.

Memory Size Option (SXTK)

Definition: The option SXTK assembles the proper code according to the computer memory size.

Option:		DELETE/	21	
	SXTK	EQU	0	32K memory size
	SXTK	EQU	1	16K memory size

Dependency: Computer memory size.

Release Value: SXTK equals 0 on the MSOS COSY tape and BCD and BDP sections of the MSOS binary source.

Deck Card: MSSORT	DECK/	P_1, P_2

2.3.28 MSSINTS (BCD/BDP)

BDP Option (BDP3312)

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:	DELETE/	4	
BDP3312	EQU	0	no BDP hardware present
BDP3312	EQU	1	non-enhanced BDP hardware present
BDP3312	EQU	2	enhanced BDP hardware present

Dependency: Must have CDC BDP hardware.

Release Value: BDP3312 equals 0 on the MSOS COSY tape, and BCD section of the MSOS binary source. BDP3312 equals 1 on the BDP section of the MSOS binary source.

Deck Card: MSSINTS	DECK/	P_1, P_2

2.3.29 MSSMERG (BCD/BDP)

BDP Option (BDP3312)

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:	DELETE/	8	
BDP3312	EQU	0	no BDP hardware present
BDP3312	EQU	1	non-enhanced BDP hardware present
BDP3312	EQU	2	enhanced BDP hardware present

Dependency: Must have CDC BDP hardware.

Release Value: BDP3312 equals 0 on the MSOS COSY tape, and BCD section of the MSOS binary source.

BDP3312 equals 1 on the BDP section of the MSOS binary source.

Deck Card: MSSMERG	DECK/	P_1, P_2

2.3.30 SORT (BCD/BDP)

BDP Option (BDP3312)

Definition: The option BDP3312 allows the user optimum efficiency on moves in Tape SORT.

Option:	DELETE/	9	
BDP3312	EQU	0	no BDP hardware present
BDP3312	EQU	1	non-enhanced BDP hardware
BDP3312	EQU	2	enhanced BDP hardware present

Dependency: BDP hardware type.

Release Value: BDP3312 equals 0 on the MSOS COSY tape and BCD section of the MSOS binary source.

BDP3312 equals 1 on the BDP section of the MSOS binary source.

Memory Protect Option (MEMPTKT)

Definition: The option MEMPTKT assembles the necessary code in MSSORT according to BDP hardware availability and the type of system being created.

Option:	DELETE/	12	
MEMPTKT	EQU	0	for a STD or Batch system or MP system with BDP hardware
MEMPTKT	EQU	2	for a MP system without BDP hardware

Dependency: System type and BDP hardware availability.

Release Value: MEMPTKT equals 0 on the MSOS COSY tape and BCD and BDP sections of the MSOS binary source.

NOTE

This option must be modified for a memory protect system with BDP hardware.

Deck Card: SORT DECK/ P_1, P_2

2.3.31 SORTPHS1 (BCD/BDP)

BDP Option (BDP3312)

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:	DELETE/	6	
BDP3312	EQU	0	no BDP hardware present
BDP3312	EQU	1	non-enhanced BDP hardware present
BDP3312	EQU	2	enhanced BDP hardware present

Dependency: Must have CDC BDP hardware.

Release Value: BDP3312 equals 0 on the MSOS COSY tape and BCD section of the MSOS binary source. BDP3312 equals 1 on the BDP section of the MSOS binary source.

Deck Card: SORTPHS1 DECK/ P_1, P_2

2.3.32 SRTEQUAL (BCD/BDP)

BDP Option (BDP3312

Definition: The BDP option permits those installations with CDC BDP hardware to utilize the hardware through software modifications.

Option:		DELETE/	5	
	BDP3312	EQU	0	no BDP hardware present
	BDP3312	EQU	1	non-enhanced BDP hardware present
	BDP3312	EQU	2	enhanced BDP hardware present

Dependency: Must have CDC BDP hardware.

Release Value: BDP3312 equals 0 on the MSOS COSY tape and BCD section of the MSOS binary source.

BDP3312 equals 1 on the BDP section of the MSOS binary source.

Deck Card: SRTEQUAL DECK/ P_1, P_2

2.3.33 ALGOL (COMMON REL)

Memory Protect Option

Definition: If the routine ALGOL is reassembled from the COSY tape for a memory protect variant, an extra 40 words of common must be deleted.

Option: DELETE/ 11

Dependency: Type of system.

Release Value: On the MSOS COSY tape and MSOS binary source the extra 40 words of common are present.

NOTE

This option must be modified for a Memory Protect system.

Deck Card: ALGOL DECK/ P_1, P_2

2.4 ERROR RECOVERY ASSEMBLY OPTIONS

The modularity of error recovery routines permits easy modification to reflect hardware changes and customer needs and to implement efficient recovery from detected errors. The routines are assembled into PRELOAD and MSIOCCP and can optionally be assembled into POSTLOAD (LERP = 1), LOADER (LERP = 1), and RDUMP (SER.OPT = 1). If error recovery is assembled into POSTLOAD or RDUMP, it must be assembled into LOADER. Any changes made to the error recovery routines should also be made in PRELOAD and MSIOCCP and in POSTLOAD, LOADER, and RDUMP when applicable.

The library contains two groups of error recovery routines, one in the relocatable file (file 2) and one in the absolute file (file 1). The two sets of routines are identical except file 2 contains RAAR, SCAR, NRD, and WHATISIT.

For most efficient loading, the recommended order of the error recovery routines on the library is:

RAAR

SCAR

MTWPR

MTRPR

MTLDACPR

MTDER

PRCPR

CRDER

CPDER

MTWPRR

MTRPRR

MTLDCPRR

MTRTNS

CMNRTNS

WHATISIT

NRD

NRC

NWR

T.NOTRDY

TYPEOUT.
WHATKIND

The following assembly options are available in error recovery:

Routine	Assembly Option
SCAR	TAPES, CARDRDR, CARDPNCH, PRINTER
NRD	READB
RAAR	TAPES
WHATISIT	TAPES
MTRTNS	READB, WRITEB
NRC	TAPES

2.4.1 SCAR (COMMON REL)

CARDRDR Option (CARDRDR)

Definition: The option CARDRDR controls references to card reader recovery within the SCAR routine.

Option:				
		DELETE/	90	
	CARDRDR	EQU	0	no card reader recovery
	CARDRDR	EQU	7	card reader recovery present

Dependency: If CARDRDR is equated to zero, all calls to card reader recovery are eliminated and CRDER is not called. If CARDRDR is equated to nonzero, card reader recovery is present.

Release Value: CARDRDR = 7 on the MSOS COSY tape and MSOS binary source.

CARDPNCH Option (CARDPNCH)

Definition: The option CARDPNCH controls references to card punch recovery.

Option:				
		DELETE/	91	
	CARDPNCH	EQU	0	no card punch recovery
	CARDPNCH	EQU	7	card punch recovery present

Dependency: If CARDPNCH is equated to zero, all calls to CARDPNCH routine are eliminated. If CARDPNCH is equated to nonzero, card punch recovery is present.

Release Value: CARDPNCH = 7 on the MSOS COSY tape and MSOS binary source.

PRINTER Option (PRINTER)

Definition: The option PRINTER controls references to printer recovery.

Option:		DELETE/	92	
	PRINTER	EQU	0	no printer recovery
	PRINTER	EQU	7	printer recovery present

Dependency: If PRINTER is equated to zero all calls to PRCPR are eliminated. If PRINTER is equated to nonzero, printer recovery is present.

Release Value: PRINTER = 7 on the MSOS COSY tape and MSOS binary source.

Deck Card: SCAR DECK/ P_1, P_2 †

2.4.2 MTLDCPRR (COMMON ABS AND COMMON REL)

NOATMPTS Option (NOATMPTS)

Definition: The option NOATMPTS determines the total number of attempts at recovery from a lost data or a channel parity error before asking for an operator decision.

Option:		DELETE/	42	
	NOATMPTS	EQU	4	four attempts

Dependency: Installation option.

Release Value: NOATMPTS = 4 on the MSOS COSY tape and MSOS binary source.

Deck Card: MTLDCPRR DECK/ P_1, P_2

2.4.3 NRD (COMMON REL)

READB Option (READB)

Definition: The option READB determines whether read backward is a legal function.

Option:		DELETE/	44	
	READB	EQU	0	read backward is not a legal function
	READB	EQU	77	read backward is legal

† See SCAR tape option (section 2.4.6)

Dependency: Installation option.

Release Value: READB = 77 on the MSOS COSY tape and MSOS binary source.

Deck Card:	NRD	DECK/	P_1, P_2

2.4.4 TAPES ASSEMBLY OPTIONS

Definition: The option TAPES controls references to magnetic tapes for the error recovery routines. The option TAPES is located in the routines SCAR, RAAR, WHATISIT, and NRC.

Options:		DELETE/	89	
	TAPES	EQU	7	maximum number of tapes used is 7
Deck Card:	RAAR	DECK/	P_1, P_2	
		DELETE/	89	
	TAPES	EQU	8	maximum number of tapes used is 8
Deck Card:	SCAR	DECK/	P_1, P_2	
		DELETE/	16	
	TAPES	EQU	8	maximum number of tapes used is 8
Deck Card:	WHATISIT	DECK/	P_1, P_2	
		DELETE/	71	
	TAPES	EQU	8	maximum number of tapes used is 8
Deck Card:	NRC	DECK/	P_1, P_2	

Dependency: If TAPES is equated to zero in the SCAR routine, all references to magnetic tape routines are eliminated. If TAPES is equated to nonzero the value must reflect the maximum number of tape drives used in any program for the SCAR and WHATISIT routines. If TAPES is nonzero, its value must be the same for all routines in which it occurs.

Release Value: TAPES = 8 on the MSOS COSY tape and MSOS binary source. (TAPES = 7 for RAAR.)

2.4.5 WRITEB ASSEMBLY OPTION

Definition: The option WRITEB ASSEMBLY provides for execution for a controlled backspace following a write during write recovery.

Options:		DELETE/	18	
	WRITEB	EQU	0	controlled backspace was not installed. A backspace will be used instead.
	WRITEB	EQU	77B	Controlled backspace is installed.
Deck Card:	MTRTNS	DECK/	P_1, P_2	

Dependency: The controlled backspace FCO must be installed on all drives using the magnetic tape error recovery update package.

Release Value: WRITEB = 0 on ITR release tape.

2.5 ERROR RECOVERY ROUTINE MACROS

Local macros are provided in the RAAR, SCAR, and NRC error recovery routines to facilitate generation of special error recovery decks. These special decks can be used effectively for alternate copies of the error recovery routines which have different values for the assembly options. For example, a user may maintain one special set of error recovery routines which do not include tape recovery for use by programs that do not use magnetic tapes. The advantage of such a special set of decks is that the magnetic tape recovery routines will not be loaded at object time (with a program calling the special routines) resulting in a corresponding saving in core size. The macros themselves only provide different symbols for the entry points and thus allow the loader to differentiate between standard and nonstandard copies of the error recovery routines. It is recommended that the IDENT cards be changed to reflect the different name given to the main entry point and to facilitate identifying these nonstandard decks on libraries.

2.5.1 SCAR

For SCAR, the ASSEMBLE macro gives the following entry point names for the given parameter in the macro call.

Parameter	Entry Points		
0	SCAR	SCARNM	SCARUST1
1	SCARSP	SCARNMSP	SPSCUST1
2	SCARMT	SCARNMMT	MTSCUST1

Make the following changes to the macro call:

```
                DELETE/     1
                IDENT       name
                DELETE/     94
                ASSEMBLE    (n)
SCAR            DECK/
```

The assembly options are not changed by the use of this macro. Therefore, the user can define these according to his needs.

2.5.2 RAAR

The ASSEMBLE macro in RAAR predefines types of hardware to be handled by the particular version of RAAR.

Parameter	Entry Point	Type of Equipment
0	RAAR	Magnetic tapes and card and visual recording equipment
1	RAAR	Card and visual recording equipment
2	RAARSP	Card and visual recording equipment
3	RAARMT	Magnetic tapes

Make the following changes to the macro call:

```
              DELETE/     1
              IDENT       name
              DELETE/     81
              ASSEMBLE    (n)
              DELETE/     82
TAPES         EQU         t
RAAR          DECK/
```

When a parameter value of 2 or 3 is used, the IDENT card should be changed to reflect the different name of the entry point. The definition of TAPES in RAAR must be consistent with the predefined meaning stated above. Further, for decks using a parameter value of 3, the following change must also be inserted:

```
              INSERT/     118
NRC           EQU         NRC.MT
```

2.5.3 NRC

The use of the ASSEMBLE macro in RAAR requires the use of the ASSEMBLE macro in NRC. The value of the calling parameter must be:

0 If the ASSEMBLE parameter in RAAR is 0

1 If the ASSEMBLE parameter in RAAR is 1 or 2

2 If the ASSEMBLE parameter in RAAR is 3

Make the following changes to the macro call:

```
              DELETE/     1
              IDENT       name
              DELETE/     70
              ASSEMBLE    (n)
              DELETE/     71
TAPES         EQU         t
NRC           DECK/
```

In NRC, the TAPES definition must be the same as the value assembled in the special RAAR deck.

2.6 ERROR RECOVERY EQUATES (VARIABLE RESIDENT)

Error recovery equates vary for the variable resident routines and can be tailored to each installation. In the routines PRELOAD and MSIOCCP, the standard error recovery routines have been permanently incorporated. In the variable resident routines POSTLOAD, LOADER, and RDUMP, the applicable standard error recovery routines have been made an assembly option. The following error recovery equates are only applicable when the assembly option for standard error recovery is selected. If the error recovery equate is selected in POSTLOAD, it must be selected in the routine LOADER.

2.6.1 PRELOAD

The following error recovery equates are always valid. They provide error recovery for magnetic tape, card punch, card reader, and printer.

		DELETE/	6,13
	ATTMPT1	EQU	4
	ATTMPT2	EQU	8
	BKSPLIM2	EQU	3
	CKSMCNST	EQU	4
	MAXERASE	EQU	11
	NOATMPTS	EQU	4
	READB	EQU	77
	TAPES	EQU	8
Deck Card:	PRELOAD	DECK/	P_1, P_2

2.6.2 POSTLOAD

If the equate LERP is set to 1, error recovery is present for magnetic tape, card reader, card punch, and printer, and the following error recovery equates are valid.

	DELETE/	5,12
ATTMPT1	EQU	4
ATTMPT2	EQU	8
BKSPLIM2	EQU	3
CKSMCNST	EQU	4
MAXERASE	EQU	11
NOATMPTS	EQU	4
READB	EQU	77
TAPES	EQU	8

Deck Card: POSTLOAD DECK/ P_1, P_2

2.6.3 LOADER

If the equate LERP is set to 1, error recovery is present for magnetic tape, card reader, card punch, and printer, and the following error recovery equates are valid.

	DELETE/	6,12
ATTMPT1	EQU	4
ATTMPT2	EQU	8
BKSPLIM2	EQU	3
CKSMCNST	EQU	4
MAXERASE	EQU	11
NOATMPTS	EQU	4
TAPES	EQU	8

Deck Card: LOADER DECK/ P_1, P_2

2.6.4 RDUMP

If the equate SER.OPT is set to 1, error recovery is present for magnetic tape and printer, and the following error recovery equates are valid.

	DELETE/	5,14
ATTMPT1	EQU	4
ATTMPT2	EQU	8
BKSPLIM2	EQU	3
CARDPNCH	EQU	0
CARDRDR	EQU	0
CKSMCNST	EQU	4
MAXERASE	EQU	11
NOATMPTS	EQU	4
PRINTER	EQU	7
TAPES	EQU	8

Deck Card: RDUMP DECK/ P_1, P_2

2.6.5 MSIOCCP

The following error recovery equates are always valid. They provide error recovery for magnetic tape, card punch, card reader, and printer.

	DELETE/	2,9
ATTMPT1	EQU	4
ATTMPT2	EQU	8
BKSPLIM2	EQU	3
CKSMCNST	EQU	4
MAXERASE	EQU	11
NOATMPTS	EQU	4
READB	EQU	77
TAPES	EQU	8

Deck Card: MSIOCCP DECK/ P_1, P_2

2.7 ASSEMBLY OPTION DEFAULT VALUES

Table II-5 lists the release value of each assembly option according to the routine and applicable section of the binary source.

TABLE II-5. ASSEMBLY OPTION DEFAULT VALUES

Routine	Assembly Option	OPER	Section of MSOS Binary Source						
			STD	MP	BATCH	COMMON ABS	BCD	BDP	COMMON REL
MZERO	MP	EQU		0					
	BDP3312	EQU		0					
ZERO	BDP3312	EQU	0		0				
CIO	DP	EQU	0	0					
	TIMEOUT	EQU	0	0					
BCIO	DP	EQU			0				
	TIMEOUT	EQU			0				
DRIV3528	T657	EQU	1	1	1				
	T659	EQU	1	1	1				
DRIV3659	NLPS	EQU	2	2	2				
DRIV3555	NLPS	EQU	2	2	2				
INITIAL	MP	EQU	0	1	0				
RDUMP	MP	EQU	0	1	0				
	SER.OPT	EQU	0	0	0				
PRELOAD	MP	EQU	0	1	0				
	BF	EQU	0	0	1				
	ACCT	EQU	0	0	0				
	TRAIN	EQU	0	0	0				
	TRN595x	EQU	0	0	0				
	NOATMPTS	EQU	4	4	4				
	MAXERASE	EQU	11	11	11				
	CKSMCNST	EQU	4	4	4				
	ATTMPT1	EQU	4	4	4				
	ATTMPT2	EQU	8	8	8				
	BKSPLIM2	EQU	3	3	3				
	TAPES	EQU	8	8	8				
	READB	EQU	77	77	77				
LOADER	MP	EQU	0	1	0				
	LERP	EQU	0	0	0				
	USASI	EQU	0	0	0				
OVPRO	USASI	EQU	0	0	0				
POSTLOAD	MP	EQU	0	1	0				
	LERP	EQU	0	0	0				
	USASI	EQU	0	0	0				

TABLE II-5. ASSEMBLY OPTION DEFAULT VALUES (cont'd)

Routine	Assembly Option	OPER	Section of MSOS Binary Source: STD	MP	BATCH	COMMON ABS	BCD	BDP	COMMON REL
MSIOCCP	NOATMPTS	EQU				4			
	MAXERASE	EQU				11			
	CKSMCNST	EQU				4			
	ATTMPT1	EQU				4			
	ATTMPT2	EQU				8			
	BKSPLIM2	EQU				3			
	READB	EQU				77			
	TAPES	EQU				8			
OC	REQ	BSS				20			20
	NRMST	BSS				18			18
AREM	SPACE	BSS				50			50
PASSONE	BDP3312	EQU				1			1
MTWPRR	MAXERASE	EQU				11			11
	CKSMCNST	EQU				4			4
MTRPRR	ATTMPT1	EQU				4			4
	ATTMPT2	EQU				8			8
	BKSPLIM2	EQU				3			3
MTLDCPRR	NOATMPTS	EQU				4			4
NRC	TAPES	EQU				8			8
MSIOGPRW	BDP3312	EQU					0	1	
MSSORT	BDP3312	EQU					0	1	
	MEMPTKT	EQU					0	0	
	SXTK	EQU					0	0	
MSSINTS	BDP3312	EQU					0	1	
MSSMERG	BDP3312	EQU					0	1	
SORT	BDP3312	EQU					0	1	
	MEMPTKT	EQU					0	0	
SORTPHS1	BDP3312	EQU					0	1	
SRTEQUAL	BDP3312	EQU					0	1	
COBOLP2	BDP3312	EQU						1	
MULTIPLY	FDP	EQU						1	
VARAN	BDP3312	EQU						1	

TABLE II-5. ASSEMBLY OPTION DEFAULT VALUES (cont'd)

Routine	Assembly Option	OPER	STD	MP	Section of MSOS Binary Source BATCH	COMMON ABS	BCD	BDP	COMMON REL
RAAR	TAPES	EQU							7
SCAR	TAPES	EQU							8
	CARDRDR	EQU							7
	CARDPNCH	EQU							7
	PRINTER	EQU							7
WHATISIT	TAPES	EQU							4
NRD	READB	EQU							77
ALGOL		BSS							40
BSIPP	NOPS	EQU							3
SIPP	NOPS	EQU							3

INTERIM LIBRARY INSTALLATION, HARDWARE INITIALIZATION AND DECK PREPARATION 3

The construction of an operable MSOS system requires the use of the Installation Utility Package (IUP). IUP establishes the interim library on the system mass storage device which is then used for hardware initialization, deck preparation, binary source modification, and final library generation. The user should be familiar with the IUP functions (Part III, Section 5).

3.1 INTERIM LIBRARY INSTALLATION AND HARDWARE INITIALIZATION

The following steps must be used to install the MSOS interim library:

1. Ready the system device. If it is an 852, 853, 854, or 841, mount the system pack.

2. Write address (853, 854, or 841) and clear the system pack (Part III, Section 3).

3. Establish an autoload routine in memory (Part III, Section 1).

4. Mount the MSOS binary release tape on unit 0 or, if the installation is from cards, place the deck of IUP in the card reader, followed by an EOF card and the generalized interim library deck.

5. Autoload from the input unit to initiate IUP (Part III, Section 5).
   JUMP 5 AUTO JUMP

6. IUP types:

   I IUP 003 INSTALLATION UTILITY PROGRAM LOADED.

   INP = LUN

7. Respond by typing:

   TY

8. IUP types:

   A IUP 006 FOR INPUT

9. Respond by typing:

   DEV

   and press FINISH

IUP now requests the DEV statement parameters. These statements define the equipment available to IUP and assign a logical unit number. When IUP is accepting a parameter and repeat is pressed, IUP will:

Repeat the present request if some characters have already been entered for the request, or

Repeat the previous request, if no characters have been entered.

10. Define the system device. The DEV statement parameters are explained in Part III, Section 5.2.1. In Chapters 3 and 4 one example is used throughout the installation procedures. The user, however, must supply his own parameters according to the equipment available.

    LUN 01 HT DP C 2 E O UU 10 DT 853 CLR N

11. IUP types:

    A IUP 006 READY FOR INPUT

12. Respond by typing:

    DEV

    and press FINISH

13. Define the logical unit containing the MSOS binary source.

    LUN 02 HT MT C O E O UU OO DT 657

14. IUP types:

    A IUP 006 READY FOR INPUT

15. Respond by typing:

    DEV

    and press FINISH

16. Define the logical unit for the card punch.

    LUN 03 HT CP C O E 1 UU OO CDT 3446

17. IUP types:

    A IUP 006 READY FOR INPUT

18. Respond by typing:

    DEV

    and press FINISH

19. Define the logical unit for the line printer.

    LUN 04 HT PR C O E 2 UU OO DT 512

20. IUP types:

    A IUP 006 READY FOR INPUT

21. Instruct IUP to install the interim library on the system device defined in step 10 (Part III, Section 5.2.11).

    8541 500

    INST LIB 01 DUMP UNIT 02 EDITION IL DN 8531 MFC ~~100~~ MSC 16 EXID MSOS 42

This example installs the interim library from unit 02 onto the system device defined as unit 01 setting the edition number to 42. It sets the device number field to 8531 and external identifier field to MSOS 42 in the device label on the system pack. It also creates a FLD and ID file for a maximum segment count of 16 and large enough to contain 100 files.

Interim library installation is now complete. The following steps are for hardware initialization.

22. IUP types the following message when the interim library has been established:

    A IUP 006 READY FOR INPUT

23. Generate an autoload card for the library device type to autoload the interim and final libraries (Part III, Section 5.2.8).

    AGEN LIB 01 CP 03 NUMBER 25

    This example generates 25 autoload cards for library device 01 on the card punch 03.

24. Load image memory of the 512/3555 before executing print operations (Part III, Section 5.2.5).

    INIT LUN 04 IMAGE 1

    This example loads the image memory of the 512/3555 defined as device 04 with train image 1 (501 compatible).

25. Load image memory of the 65x/MMTC (3518 or 3528) before executing character mode operations (BCD). See Part III, Section 5.2.5.

    INIT LUN 02 IMAGE 1

    This example loads the image memory of the 65x/MMTC defined as device 02 with image 1 (ASCII).

NOTE

For completion of the MSOS installation, the MMTC image memory must contain image 1.

26. Interim library installation and hardware initialization should be complete.

    Type:

    END

27. IUP responds:

    I IUP 039 INSTALLATION UTILITY PROGRAM TERMINATED

The user must now define the interim library environment according to the installation configuration (Section 8, Part III).

1. Press MC

2. Set SELECT JUMP 6

3. Place autoload card in card reader and ready the reader.

4. Press AUTOLOAD

   The CPU halts in the autoload region of core memory.

5. Press GO

   The autoload card routine reads the autoload record of the interim library into memory.

6. When autoload is complete, the interim library routine types:

   EDITION

7. Respond by typing:

   IL

8. The interim library routine types the following. The user supplies the underlined parameters (Part III, Section 8).

```
        *INTERIM SYSTEM*
        DEFINE ENVIRONMENT
XCTO 1  LIB = DPC2EOU10
        DT = 853
     3  INP = CRC1E1U00
        CDT = 3248
     4  OUT= PRC0E2U00
        DT= 512
        PUN = CPC0E1U00
        CDT = 3446
     5  SCR= DPC0EOU11
        DT = 853
     6  SCR = MTC0E0U00
        DT = 657
        SCR = MTC0E0U01
        DT = 657
        SCR =
```

9. Upon completion of the interim library environment:

   Press FINISH
   Message: DATE
   Type date: mmddyy
   Press FINISH

   Message: TIME
   Type time: hhmm
   Press FINISH

   Message: PUN
   If a punchless system is desired, type 00
   Press FINISH

   Message: OUT
   Press FINISH

   Message: INP
   Press FINISH

   Message: CFO
   Type OPER

10. The compter responds
    I SYS 048 (OPER. CONTROL)

11. Type: EQUIP, 01=DPC2EOU11
    Press: FINISH

    Type: CALL, MSUTIL, 58
    Press: FINISH
    Message: 01=DP, C2, EO, U11
    READY
    Press: MANUAL INTERRUPT

    Type: GO
    Press: MANUAL INTERRUPT

    Type: ENTER, 1, 853/1, S
    Press: FINISH

    Type: SCOPE
    Press: FINISH

12. System scratch files 54, 55, and 56 are allocated on the interim library at the minimum size of 1 track. Therefore, the user must release and reallocate these files according to his device type. Type the following statements:

RAT, 8541

```
FET,MSOS,FILE54,480,00,0000,0000
RELEASE,ALL
ALLOCATE,x,991231          (see below for value of x)
FET,MSOS,FILE55,480,00,0000,0000
RELEASE,ALL
ALLOCATE,y,991231          (see below for value of y)
FET,MSOS,FILE56,960,00,0000,0000
RELEASE,ALL
ALLOCATE,z,991231          (see below for value of z)
```

FET,MSØS,L-MSIØ,1024,00,0000,0000
ALLØCATE,10,991231

13. TYPE: .

PRESS FINISH

14. SYSTEM ED = neue Tape generieren
und PEELIB-INPUT

15. Nach. SEQU.

RAT. 854/1

The following table shows the recommended sizes for x, y, and z.

Device type	813	841	852	853	854	863
x (54)	50	50	220	100	100	100
y (55)	125	125	550	250	250	250
z (56)	75	75	320	150	150	150

The interim library is now available to process batch jobs. Proceed to Section 3.2.

3.2 DECK PREPARATION

The interim library is now ready to prepare the MSOS binary decks necessary to generate the final MSOS PRELIB source. To prepare the binary decks, the user must:

1. Run a COSY job to incorporate the site dependent assembly options described in Part II, Section 2.

2. Call COMPASS to assemble the new binary decks for the PRELIB source.

3.2.1 EXAMPLE OF COSY AND COMPASS JOBS

The following site dependent hardware configuration is used.

Standard MSOS
Floating-point hardware available; no BCD hardware
32K 3100 Computer

Equipment type	Quantity	Controller	Channel	Equipment	Unit
Console Type-writer	1	none	none	none	none
853 disk drive	2	3234	2	0	10,11
405 card reader	1	3248	1	1	0
512 printer	1	3555	0	2	0
415 card punch	1	3446	0	1	0
657 tape units	4	3518	0	0	0,1,2,3

Step 1.

$^{7}_{9}$SEQUENCE, 416

$^{7}_{9}$JOB, , , ,

$^{7}_{9}$CTO, , LUN01 IS COSY TAPE NO. 1

$^{7}_{9}$FET, USER, RESDECK, 960

$^{7}_{9}$ALLOCATE, 50

$^{7}_{9}$OPEN, 10 — fo10 is for binary decks output from the COMPASS assembly

$^{7}_{9}$EQUIP, 01=MT — LUN01 is MSOS COSY tape 1.

$^{7}_{9}$COSY

	DELETE/	22, 23	
RATL	EQU	2	Maximum of 2 mass storage table entries
	DELETE/	46, 49	
MST	OCT	01000000, 50000000	AET ordinal 01, 853
	OCT	12000000, 50000000	AET ordinal 10, 853
MSIO	DECK/	I=01, H, R	Hollerith output to file 54; Revisions listed
	DELETE/	7, 8	
MAXCORE	EQU	77637B	3100, 3150 or 3200 32K environment
NCHANS	EQU	4	Channels 0, 1, 2, 3 available
	DELETE/	2100, 2152	

```
            AET[†]       (DP,40,0,0,MSIO3234,RES)        Library device
                                                         Ch 2 Eq 0, Unit 0
            AET          (TY,,,,DRIVER05,RES)            Console Typewriter
            AET          (CR,100,1,0,DRIVER02,RES)       Card reader
                                                         Ch 1, Eq 1, Unit 0
            AET          (PR,200,2,0,DRIVER03,RES)       Printer
                                                         Ch 0, Eq 2, Unit 0
            AET          (CP,200,1,0,DRIVER04,RES)       Card Punch
                                                         Ch 0, Eq 1, Unit 0
            AET          (MT,200,0,3,DRIVER01)           Magnetic Tape
                                                         Ch 0, Eq 0, Unit 3
            AET          (MT,200,0,2,DRIVER01)           Magnetic Tape
                                                         Ch 0, Eq 0, Unit 2
            AET          (MT,200,0,1,DRIVER01)           Magnetic Tape
                                                         Ch 0, Eq 0, Unit 1
            AET          (MT,200,0,0,DRIVER01)           Magnetic Tape
                                                         Ch 0, Eq 0, Unit 0
            AET          (DP,40,0,1,MSIO3234)            Available disk
                                                         Ch 2, Eq 0, Unit 1
            DELETE/      2301,2308
            00           IOIP      CH 0                  Central Interrupt Table (CIT)
            00           IOIP      CH 1                  Channels 0,1, and 2 available
            00           IOIP      CH 2                  Channels 3-7 not available
            00           ABINRT    CH 3
            00           ABINRT    CH 4
            00           ABINRT    CH 5
            00           ABINRT    CH 6
            00           ABINRT    CH 7
            DELETE/      2341
SYST.IOM    OCT          7         CH 0,1,2
SCICREC1    DECK/        I=01,H,R                        Hollerith output to file 54.
                                                         Revisions listed.
            ENDCOSY/               END OF PART I
```

Step 2.

```
7
9COMPASS,I=54,L,R,X=10      PART II          Input from file 54. Punched output
                                             is on file 10.
77
88                          END OF PART II
```

† See Part III, Section 6.2 for format of the AET macro.

MSOS BINARY SOURCE MODIFICATION AND FINAL LIBRARY GENERATION 4

The MSOS binary source file requires modification to generate a final MSOS PRELIB source. The modification procedures include the following steps:

1. UNIT card replacement
2. BCD, BDP, and FDP options
3. Driver selection
4. PLIBEDIT run

When the PRELIB source is generated, the user may then create the final library.

4.1 UNIT CARD REPLACEMENT

UNIT cards are included in the MSOS binary source structure as a convenience for modification of the source decks. They must be replaced by other UNIT cards or binary decks.

When assembling binary files to mass storage, observe the following precautions.

1. File ordinals must range from 20-53 for binary files to avoid conflict with PRELIB file ordinals.
2. Prior to PRELIB execution, the binary files must be opened from the typewriter or by cards. PRELIB does not have access to the FET statements of these files.

The first UNIT card is:

$^{7}_{9}$UNIT,AA

Either replace this card with the binary deck of MSIO (STD/MP) or BMSIO (Batch) or with one of the following:

$^{7}_{9}$UNIT,lu

lu Logical unit number of nonmass storage device containing the binary of MSIO or BMSIO

$^{7}_{9}$UNIT,fo,D

fo File ordinal of mass storage file containing the binary of MSIO or BMSIO

The second UNIT card is:

$^{7}_{9}$UNIT,BB

Either replace this card with the binary deck of MCICREC1, BCICREC1, SCICREC1, or with one of the following:

$^{7}_{9}$UNIT,lu

lu Logical unit number of the nonmass storage device containing the binary of MCICREC1, SCICREC1, or BCICREC1.†

$^{7}_{9}$UNIT,fo,D

fo File ordinal of mass storage file containing the binary of MCICREC1,SCICREC1, or BCICREC1.

4.2 BCD, BDP, AND FDP OPTIONS

4.2.1 STD/BATCH MSOS FOR 3100/3150/3200 AND MP MSOS FOR 3100/3150/3200

Remove the appropriate SEPOINT cards as indicated below:

BCD Hardware	FDP Hardware	Action††
Absent	Absent	Remove SEPOINT,BCDBOXS and SEPOINT,FDPBOXS
Absent	Present	Remove SEPOINT,BCDBOXS
Present	Absent	Remove SEPOINT,FDPBOXS

4.2.2 STD/BATCH MSOS FOR 3300/3500

Where BDP hardware exists and is defined, BDP COBOL must be used. Otherwise BCD COBOL is required. If BDP is equated to 1 in BCICREC1, MCICREC1, or SCICREC1, the library must always be run with the console BDP switch turned on.

† The nonmass storage device must be defined during interim environment operations through a DEVICE statement. It must also be equipped prior to the PRELIB run according to the logical unit specified on the unit card.

†† In all cases, only BCD COBOL can be used.

BDP Hardware	FDP Hardware	Action
Absent	Absent	Remove SEPOINT, BCDBOXS and SEPOINT, FDPBOXS
Absent	Present	Remove SEPOINT, BCDBOXS
Present	Absent	Remove SEPOINT, FDPBOXS

4.2.3 MP MSOS FOR 3300/3500

Where BDP hardware exists and is defined in MCICREC1 (BDP EQU 1) BDP COBOL must be used. Otherwise, BCD COBOL is required. If BDP is equated to 1 in MCICREC1, the library must always be run with the console BDP switch turned on.

BDP Hardware	FDP Hardware	Action
Absent	Absent	Remove SEPOINT, BCDBOXS and SEPOINT, FDPBOXS
Absent	Present	Remove SEPOINT, FDPBOXS
Present	Absent	Remove SEPOINT, FDPBOXS
Present	Present	Remove SEPOINT,OPTBOXS

4.3 UNDEFINED SEPOINT

A SEPOINT card must be inserted directly after the LOADER record.

A SEPOINT card is used to prevent the loading of a subroutine or program. If the name specified on the control card is undefined, the symbol is equated to ABNORMAL, and the job terminates. This technique is particularly useful for installations with optional floating-point and BCD hardware. For example, the following cards in file 1 would prevent loading of hardware simulation routines FDPBOXS and BCDBOXS.

```
SEPOINT, FDPBOXS
SEPOINT,BCDBOXS
```

Other undefined system entry points such as CIP, PERRADD, and SETCLOCK do not invalidate the PRELIB as long as no routines reference the undefined SEPOINT.

4.4 DRIVER SELECTION

4.4.1 RESIDENT DRIVERS

The drivers in Table II-6 are contained on the STD, MP, and Batch sections of the MSOS binary source. Drivers that are nonapplicable to the installation hardware availability must be deleted.

The drivers are identified by unique program lengths on their IDC cards and may be removed by cross-referencing the binary source listing and Table II-6.

All drivers placed in resident must have unique entry point names specified.

TABLE II-6. RESIDENT DRIVERS

Hardware	Control Data Controllers	Control Data Equipment (in combination)	COSY Name	Entry Point Name	Deck Length
MT	3X2X	604 607 608	DRIV607	DRIVER01	354
MT	3X2X	601 603 606	DRIV606	DRIVER01	330
MT	35X8	657 659	DRIV3528	DRIVER01	457
CR	3447 3649	405	DRIV3649	DRIVER02	244
CR	3248 3142	405	DRIV3248	DRIVER02	234
PR	3256 3659 3254	501 505	DRIV3659	DRIVER03	467
PR	3555	512	DRIV3555	DRIVER03	536
CP	3644 3446	415	DRIV3644	DRIVER04	254
CP	3245	415	DRIV3245	DRIVER04	345
TY	Console Typewriter		DRIVTYWR	DRIVER05	115
PL	3293	Plotter	DRIV3293	DRIVER11	60
DP	3232	852	MSIO3232	MSIO3232	257
DP	3234	853, 854 813, 814	MSIO3234	MSIO3234	124
DP	3553	841	MSIO3553	MSIO3553	124
DR	3436 3637	863	MSIO3436	MSIO3436	77
OR	3195	915	DRIV3195	DRIVER16	330
TR TP	3691	Paper Tape Station	DRIV3691	DRIVER06 DRIVER07	1140

4.4.2 NONRESIDENT

If an MSOS MP library is to be installed, all drivers must be included within the resident record. If an MSOS STD or Batch library is to be installed, drivers not required for system units may be placed in nonresident (relocatable - file 2) environment to decrease the amount of core required for the resident record. The below procedures and rules apply:

1. Do not incorporate the driver (drivers) in the resident file.
2. Place the binary deck of the driver in the relocatable file
3. Add the following SEPOINTS

DRIVER	SEPOINT
DRIV606 or DRIV607	${}^{7}_{9}$SEPOINT, SEL
DRIV3644	${}^{7}_{9}$SEPOINT, CIO5.7
	${}^{7}_{9}$SEPOINT, IOCON
	${}^{7}_{9}$SEPOINT, SEL
DRIV3691	${}^{7}_{9}$SEPOINT, SEL
DRIV3528	${}^{7}_{9}$SEPOINT, SEL
	${}^{7}_{9}$SEPOINT, CIO5.7

See AET for nonresident driver address (part III, section 6).

Equipment with nonresident drivers are only available through EQUIP statements and therefore cannot be used by autoload or ENDSCOPE procedures.

4.5 PLIBEDIT

Before the user can run a PLIBEDIT job to generate the MSOS PRELIB source, he must have completed the following operations:

Generated an operable interim library

Obtained the appropriate autoload card

Completed COSY and COMPASS jobs to incorporate site dependent assembly options

In addition, the user should be familiar with PLIBEDIT (Part III, Section 10).

4.5.1 BINARY SOURCE TAPE DESCRIPTION

The binary source tape has seven sections. Each installation requires at least four of the sections. The sections are designated as follows:

STD	Standard MSOS resident routines
MP	Memory Protect MSOS resident routines
BATCH	Batch MSOS resident routines
COMMON ABS	Common absolute routines
BCD	BCD routines
BDP	BDP routines
COMMON REL	Common relocatable routines

SAINT, RESPOND, ADAPT, PERT TIME, PERT COST, USASI FORTRAN, and USASI COBOL are contained in separate release packages. See Part III, Section 5.

The following table indicates the usage of these sections:

	STD	MP	BATCH	COMMON ABS	BCD	BDP	COMMON REL
STD/BCD	X			X	X		X
STD/BDP	X			X		X	X
MP/BCD		X		X	X		X
MP/BDP		X		X		X	X
BATCH/BCD			X	X	X		X
BATCH/BDP			X	X		X	X

4.5.2 PLIBEDIT EXAMPLE

The following procedure illustrates the use of the binary source for the system defined in Section 3. This procedure is for a standard MSOS system utilizing BCD hardware simulation.

1. Autoload the interim library disk. Clear SELECT JUMP 6.

2. Define the interim system. Upon completion of autoload the following is typed:

    Message: EDITION
    Type: IL
    Message: *INTERIM SYSTEM*
    DEFINE ENVIRONMENT

    In the following console typewriter inputs, underlined characters are site dependent:

    LIB = DPC2E0U10

    DT = 853

    INP = CRC1E1U00

    CDT = 3248

    OUT = PRC0E2U00

    DT = 512

    PUN = press FINISH

    SCR = MTC0E0U00

    DT = 657

    SCR = MTC0E0U01

    DT = 657

    SCR =

3. The interim library is defined. Press FINISH

    Message: DATE
    Type: mmddyy
    Press: FINISH

    Message: TIME
    Press: FINISH

4. Use PLIBEDIT to modify the MSOS binary source on binary release tape 1, LUN 01, and place the output on LUN 02. This output is called the PRELIB source tape. LUN 01 and 02 are equipped through a LED card within the library program PLIBEDIT. The created PRELIB source tape is assigned as standard input for a PRELIB run to generate a MSOS system. Note that the decks edited must be in the order they appear on the MSOS binary source.

			COMMENTS
$^7_9$SEQUENCE, 416			
$^7_9$JOB,,,,			
$^7_9$CTO,,LUN01 IS BINARY RELEASE TAPE NO. 1			
$^7_9$CTO,,LUN02 IS NEW PLIBEDIT (SOURCE) TAPE			
$^7_9$FET, USER, RESDECK, 960			Open the file containing binary card images of MSIO and SCICREC1.
$^7_9$OPEN,10			
$^7_9$PLIBEDIT			
	LOCATE/	EOF=2	Skip over IUP and Interim
	REPLACE/	PRELIB PARAMETERS	Edit in control card
$^7_9$PRELIB,,,,IL,DH			
	POSITION/STD		Locate to first card following STD control card w/o copying
	REPLACE/	UNIT, AA	Edit in MSIO deck from fo 10 to the output tape
MSIO	FILE/	10	
DRIV607	DELETE/	DRIV606	
DRIV3649	DELETE/		
DRIV3659	DELETE/		Delete those drivers not required by the installation
DRIV3245	DELETE/		
DRIV3293	DELETE/	MSIO3232	
MSIO3553	DELETE/	DRIV3691	
	REPLACE/	UNIT, BB	Edit in SCICREC1 binary deck
SCICREC1	FILE/	10	
	LOCATE/	SEPOINT, FDPBOXS	
	SKIP/	1	Skip over SEPOINT, BCDBOXS control card

POSTLOAD	LOCATE/	Copy the remainder of standard source
	POSITION/ COMMON ABS	Skip to first record after COMMON ABS control card
WHATKIND	LOCATE/	Copy all of common absolute records to output tape
	POSITION/BCD	
WAITBEEP	LOCATE/	Skip to BCD routines and copy to output tape
	POSITION COMMON REL	
	ENDEDIT/	Skip to first card after COMMON REL control card and copy remainder to output tape.

77
88

NOTE

If user files exist on the interim system after completion of these runs, it is recommended that they be released before final library preparation. System files 54, 55, and 56 need not be released. However, user files existing on the system pack during a PRELIB run will remain resulting in a non-contiguous final MSOS system.

4.5.3 BINARY SOURCE TAPE LISTING

```
CN=00001    C.C.  SEQUENCE,100,MSOS V4.2 PRELIB SOURCE
CN=00002    C.C.  JOB,,,,
CN=00003    C.C.  PRELIB PARAMETERS.
CN=00004    C.C.  STD                    * * * * * STANDARD PRELIB SOURCE * * * * *
CN=00005    C.C.  RECORD
CN=00006    C.C.  ORIGIN,0
CN=00007    IDC  ZERO       PROG.LENGTH=00040          07/10/70
CN=00015    IDC  CIO        PROG.LENGTH=01667          08/31/70
CN=00058    C.C.  UNIT,AA                WHERE AA IS FO/LU OF MSIO BINARY.
CN=00059    IDC  DRIV607    PROG.LENGTH=00354          08/31/70
CN=00071    IDC  DRIV606    PROG.LENGTH=00340          08/31/70
CN=00083    IDC  DRIV3528   PROG.LENGTH=00457          07/10/70
CN=00098    IDC  DRIV3649   PROG.LENGTH=00244          07/10/70
CN=00108    IDC  DRIV3248   PROG.LENGTH=00234          07/10/70
CN=00117    IDC  DRIV3659   PROG.LENGTH=00467          07/10/70
CN=00133    IDC  DRIV3555   PROG.LENGTH=00536          08/31/70
CN=00150    IDC  DRIV3644   PROG.LENGTH=00254          07/10/70
CN=00160    IDC  DRIV3245   PROG.LENGTH=00345          07/10/70
CN=00170    IDC  DRIVTYWR   PROG.LENGTH=00112          08/31/70
CN=00177    IDC  DRIV3293   PROG.LENGTH=00060          07/10/70
CN=00183    IDC  MSIO3232   PROG.LENGTH=00257          07/10/70
CN=00194    IDC  MSIO3234   PROG.LENGTH=00124          07/10/70
CN=00201    IDC  MSIO3553   PROG.LENGTH=00124          07/10/70
CN=00208    IDC  MSIO3436   PROG.LENGTH=00077          07/10/70
CN=00214    IDC  DRIV3195   PROG.LENGTH=00330          07/10/70
CN=00226    IDC  DRIV3691   PROG.LENGTH=01140          07/10/70
CN=00251    C.C.  UNIT,BB                WHERE BB IS FO/LU OF SCICREC1 BINARY.
CN=00252    IDC  INITIAL    PROG.LENGTH=01134          07/11/70
CN=00282    EXS   ENDPROT=POSTLOAD
CN=00283    EXS   LBUF=START2
CN=00284    EXS   LOADMSIO=ENDPROT
CN=00285    EXS   PROMEM=START2
CN=00286    C.C.  RECORD
CN=00287    C.C.  ORIGIN,BINBUF+50
CN=00288    IDC  RDUMP      PROG.LENGTH=00631          08/31/70
CN=00306    C.C.  RECORD
CN=00307    C.C.  ORIGIN,BINBUF+50
CN=00308    IDC  PRELOAD    PROG.LENGTH=13076          10/25/70
CN=00509    C.C.  RECORD
CN=00510    C.C.  ORIGIN,PROMEM
CN=00511    IDC  CKREC      PROG.LENGTH=00000          07/11/70
CN=00513    IDC  LOADER     PROG.LENGTH=05442          10/25/70
CN=00611    C.C.  SEPOINT,ABNORMAL
CN=00612    C.C.  SEPOINT,ACCOUNTS
CN=00613    C.C.  SEPOINT,AET
CN=00614    C.C.  SEPOINT,ALGOPT
CN=00615    C.C.  SEPOINT,BERRADD
CN=00616    C.C.  SEPOINT,BKEXIT
CN=00617    C.C.  SEPOINT,BKRUNFLG
CN=00618    C.C.  SEPOINT,BNJ.
CN=00619    C.C.  SEPOINT,BRHT
```

```
CN=00620     C.C.  SEPOINT,CIO
CN=00621     C.C.  SEPOINT,CIC3.2
CN=00622     C.C.  SEPOINT,CIC3.01
CN=00623     C.C.  SEPOINT,CIP
CN=00624     C.C.  SEPOINT,CIT
CN=00625     C.C.  SEPOINT,CIT.RTM
CN=00626     C.C.  SEPOINT,CST
CN=00627     C.C.  SEPOINT,DINT.
CN=00628     C.C.  SEPOINT,EINT.
CN=00629     C.C.  SEPOINT,EST
CN=00630     C.C.  SEPOINT,IOP
CN=00631     C.C.  SEPOINT,LENGRDT
CN=00632     C.C.  SEPOINT,LOADER
CN=00633     C.C.  SEPOINT,LOADMSIO
CN=00634     C.C.  SEPOINT,LOC5
CN=00635     C.C.  SEPOINT,MEMORY
CN=00636     C.C.  SEPOINT,MIBKADD
CN=00637     C.C.  SEPOINT,MIBUF
CN=00638     C.C.  SEPOINT,MIFORADD
CN=00639     C.C.  SEPOINT,MSIO
CN=00640     C.C.  SEPOINT,MSIOFLG
CN=00641     C.C.  SEPOINT,MSIO.SP
CN=00642     C.C.  SEPOINT,NDEXIT
CN=00643     C.C.  SEPOINT,PERRADD
CN=00644     C.C.  SEPOINT,RDCKF1
CN=00645     C.C.  SEPOINT,RHT
CN=00646     C.C.  SEPOINT,RIO
CN=00647     C.C.  SEPOINT,RPT
CN=00648     C.C.  SEPOINT,RSTORE
CN=00649     C.C.  SEPOINT,RSTOREQ
CN=00650     C.C.  SEPOINT,SETCLOCK
CN=00651     C.C.  SEPOINT,START2
CN=00652     C.C.  SEPOINT,UST
CN=00653     C.C.  SEPOINT,FDPBOXS
CN=00654     C.C.  SEPOINT,BCDBOXS
CN=00655     C.C.  RECORD
CN=00656     C.C.  ORIGIN,LSTLGTH
CN=00657     IDC  OVPRO      PROG.LENGTH=02056         08/31/70
CN=00704     C.C.  RECORD
CN=00705     C.C.  ORIGIN,BINBUF
CN=00706     IDC  PROTECT    PROG.LENGTH=00310         07/11/70
CN=00717     IDC  POSTLOAD   PROG.LENGTH=02772         08/31/70
CN=00777     C.C.  MP                    * * * * MEMORY PROTECT PRELIB SOURCE * * *
CN=00778     C.C.  RECORD
CN=00779     C.C.  ORIGIN,0
CN=00780     IDC  MZERO      PROG.LENGTH=00040         07/10/70
CN=00787     IDC  CIO        PROG.LENGTH=01667         08/31/70
CN=00830     C.C.  UNIT,AA               WHERE BB IS THE FO/LU OF MSIO BINARY.
CN=00831     IDC  DRIV607    PROG.LENGTH=00354         08/31/70
CN=00843     IDC  DRIV606    PROG.LENGTH=00340         08/31/70
CN=00855     IDC  DRIV3528   PROG.LENGTH=00457         07/10/70
CN=00870     IDC  DRIV3649   PROG.LENGTH=00244         07/10/70
CN=00880     IDC  DRIV3248   PROG.LENGTH=00234         07/10/70
CN=00889     IDC  DRIV3659   PROG.LENGTH=00467         07/10/70
CN=00905     IDC  DRIV3555   PROG.LENGTH=00536         08/31/70
```

```
CN=00922     IDC   DRIV3644   PROG.LENGTH=00254          07/10/70
CN=00932     IDC   DRIV3245   PROG.LENGTH=00345          07/10/70
CN=00942     IDC   DRIVTYWR   PROG.LENGTH=00112          08/31/70
CN=00949     IDC   DRIV3293   PROG.LENGTH=00060          07/10/70
CN=00955     IDC   MSIO3232   PROG.LENGTH=00257          07/10/70
CN=00966     IDC   MSIO3234   PROG.LENGTH=00124          07/10/70
CN=00973     IDC   MSIO3553   PROG.LENGTH=00124          07/10/70
CN=00980     IDC   MSIO3436   PROG.LENGTH=00077          07/10/70
CN=00986     IDC   DRIV3195   PROG.LENGTH=00330          07/10/70
CN=00998     IDC   DRIV3691   PROG.LENGTH=01140          07/10/70
CN=01023     IDC   PROTECT    PROG.LENGTH=00310          07/11/70
CN=01034     C.C.   UNIT,BB              WHERE BB IS THE FO/LU OF MCICREC1 BINARY.
CN=01035     IDC   INITIAL    PROG.LENGTH=01217          07/12/70
CN=01066     EXS    ENDPROT=POSTLOAD
CN=01067     EXS    LBUF=START2
CN=01068     EXS    LOADMSIO=PROMEM
CN=01069     EXS    PROMEM=START2
CN=01070     C.C.   RECORD
CN=01071     C.C.   ORIGIN,BINBUF+50
CN=01072     IDC   RDUMP      PROG.LENGTH=00660          08/31/70
CN=01091     C.C.   RECORD
CN=01092     C.C.   ORIGIN,BINBUF+50
CN=01093     IDC   PRELOAD    PROG.LENGTH=13125          10/25/70
CN=01295     C.C.   RECORD
CN=01296     C.C.   ORIGIN,PROMEM
CN=01297     IDC   CKREC      PROG.LENGTH=00000          07/11/70
CN=01299     IDC   LOADER     PROG.LENGTH=05522          10/25/70
CN=01397     C.C.   SEPOINT,ABNORMAL
CN=01398     C.C.   SEPOINT,ACCOUNTS
CN=01399     C.C.   SEPOINT,AET
CN=01400     C.C.   SEPOINT,ALGOPT
CN=01401     C.C.   SEPOINT,BERRADD
CN=01402     C.C.   SEPOINT,BKEXIT
CN=01403     C.C.   SEPOINT,BKRUNFLG
CN=01404     C.C.   SEPOINT,BNJ.
CN=01405     C.C.   SEPOINT,BRHT
CN=01406     C.C.   SEPOINT,CIO
CN=01407     C.C.   SEPOINT,CIC3.2
CN=01408     C.C.   SEPOINT,CIP
CN=01409     C.C.   SEPOINT,CIT
CN=01410     C.C.   SEPOINT,CIT.RTM
CN=01411     C.C.   SEPOINT,CST
CN=01412     C.C.   SEPOINT,DINT.
CN=01413     C.C.   SEPOINT,EINT.
CN=01414     C.C.   SEPOINT,EST
CN=01415     C.C.   SEPOINT,IOP
CN=01416     C.C.   SEPOINT,LENGRDT
CN=01417     C.C.   SEPOINT,LOADER
CN=01418     C.C.   SEPOINT,LOADMSIO
CN=01419     C.C.   SEPOINT,LOC5
CN=01420     C.C.   SEPOINT,MEMORY
CN=01421     C.C.   SEPOINT,MIBKADD
CN=01422     C.C.   SEPOINT,MIBUF
CN=01423     C.C.   SEPOINT,MIFORADD
CN=01424     C.C.   SEPOINT,MSIO
```

```
CN=01425     C.C.  SEPOINT,MSIOFLG
CN=01426     C.C.  SEPOINT,MSIO.SP
CN=01427     C.C.  SEPOINT,NDEXIT
CN=01428     C.C.  SEPOINT,PERRADD
CN=01429     C.C.  SEPOINT,RDCKF1
CN=01430     C.C.  SEPOINT,RHT
CN=01431     C.C.  SEPOINT,RIO
CN=01432     C.C.  SEPOINT,RPT
CN=01433     C.C.  SEPOINT,RSTORE
CN=01434     C.C.  SEPOINT,RSTOREQ
CN=01435     C.C.  SEPOINT,SETCLOCK
CN=01436     C.C.  SEPOINT,START2
CN=01437     C.C.  SEPOINT,UST
CN=01438     C.C.  SEPOINT,FDPBOXS
CN=01439     C.C.  SEPOINT,OPTBOXS
CN=01440     C.C.  SEPOINT,BCDBOXS
CN=01441     C.C.  RECORD
CN=01442     C.C.  ORIGIN,LSTLGTH
CN=01443     IDC  OVPRO     PROG.LENGTH=02056        08/31/70
CN=01490     C.C.  RECORD
CN=01491     C.C.  ORIGIN,BINBUF
CN=01492     IDC  POSTLOAD  PROG.LENGTH=03017        08/31/70
CN=01553     C.C.  BATCH                   * * * * * * BATCH PRELIB SOURCE * * * * * *
CN=01554     C.C.  RECORD
CN=01555     C.C.  ORIGIN,0
CN=01556     IDC  ZERO      PROG.LENGTH=00040        07/10/70
CN=01564     IDC  BCIO      PROG.LENGTH=01577        08/31/70
CN=01616     C.C.  UNIT,AA                 WHERE AA IS THE FO/LU OF BMSIO BINARY.
CN=01617     IDC  DRIV607   PROG.LENGTH=00354        08/31/70
CN=01629     IDC  DRIV606   PROG.LENGTH=00340        08/31/70
CN=01641     IDC  DRIV3528  PROG.LENGTH=00457        07/10/70
CN=01656     IDC  DRIV3649  PROG.LENGTH=00244        07/10/70
CN=01666     IDC  DRIV3248  PROG.LENGTH=00234        07/10/70
CN=01675     IDC  DRIV3659  PROG.LENGTH=00467        07/10/70
CN=01691     IDC  DRIV3555  PROG.LENGTH=00536        08/31/70
CN=01708     IDC  DRIV3644  PROG.LENGTH=00254        07/10/70
CN=01718     IDC  DRIV3245  PROG.LENGTH=00345        07/10/70
CN=01728     IDC  DRIVTYWR  PROG.LENGTH=00112        08/31/70
CN=01735     IDC  DRIV3293  PROG.LENGTH=00060        07/10/70
CN=01741     IDC  MSIO3232  PROG.LENGTH=00257        07/10/70
CN=01752     IDC  MSIO3234  PROG.LENGTH=00124        07/10/70
CN=01759     IDC  MSIO3553  PROG.LENGTH=00124        07/10/70
CN=01766     IDC  MSIO3436  PROG.LENGTH=00077        07/10/70
CN=01772     IDC  DRIV3195  PROG.LENGTH=00330        07/10/70
CN=01784     IDC  DRIV3691  PROG.LENGTH=01140        07/10/70
CN=01809     C.C.  UNIT,BB                 WHERE BB IS THE FO/LU OF BCICREC1 BINARY.
CN=01810     IDC  INITIAL   PROG.LENGTH=01134        07/11/70
CN=01840     EXS   ENDPROT=POSTLOAD
CN=01841     EXS   LBUF=START2
CN=01842     EXS   LOADMSIO=ENDPROT
CN=01843     EXS   PROMEM=START2
CN=01844     C.C.  RECORD
CN=01845     C.C.  ORIGIN,BINBUF+50
CN=01846     IDC  RDUMP     PROG.LENGTH=00631        08/31/70
CN=01864     C.C.  RECORD
```

```
CN=01865    C.C.  ORIGIN,BINBUF+50
CN=01866    IDC   PRELOAD    PROG.LENGTH=13076          10/25/70
CN=02067    C.C.  RECORD
CN=02068    C.C.  ORIGIN,PROMEM
CN=02069    IDC   CKREC      PROG.LENGTH=00000          07/11/70
CN=02071    IDC   LOADER     PROG.LENGTH=05442          10/25/70
CN=02169    C.C.  SEPOINT,ABNORMAL
CN=02170    C.C.  SEPOINT,ACCOUNTS
CN=02171    C.C.  SEPOINT,AET
CN=02172    C.C.  SEPOINT,ALGOPT
CN=02173    C.C.  SEPOINT,BERRADD
CN=02174    C.C.  SEPOINT,BKEXIT
CN=02175    C.C.  SEPOINT,BKRUNFLG
CN=02176    C.C.  SEPOINT,BNJ.
CN=02177    C.C.  SEPOINT,BRHT
CN=02178    C.C.  SEPOINT,CIO
CN=02179    C.C.  SEPOINT,CIC3.2
CN=02180    C.C.  SEPOINT,CIC3.01
CN=02181    C.C.  SEPOINT,CIP
CN=02182    C.C.  SEPOINT,CIT
CN=02183    C.C.  SEPOINT,CIT.RTM
CN=02184    C.C.  SEPOINT,CST
CN=02185    C.C.  SEPOINT,DINT.
CN=02186    C.C.  SEPOINT,EINT.
CN=02187    C.C.  SEPOINT,EST
CN=02188    C.C.  SEPOINT,IOP
CN=02189    C.C.  SEPOINT,LENGRDT
CN=02190    C.C.  SEPOINT,LOADER
CN=02191    C.C.  SEPOINT,LOADMSIO
CN=02192    C.C.  SEPOINT,LOC5
CN=02193    C.C.  SEPOINT,MEMORY
CN=02194    C.C.  SEPOINT,MIBKADD
CN=02195    C.C.  SEPOINT,MIBUF
CN=02196    C.C.  SEPOINT,MIFORADD
CN=02197    C.C.  SEPOINT,MSIO
CN=02198    C.C.  SEPOINT,MSIOFLG
CN=02199    C.C.  SEPOINT,MSIO.SP
CN=02200    C.C.  SEPOINT,NDEXIT
CN=02201    C.C.  SEPOINT,PERRADD
CN=02202    C.C.  SEPOINT,RDCKF1
CN=02203    C.C.  SEPOINT,RHT
CN=02204    C.C.  SEPOINT,RIO
CN=02205    C.C.  SEPOINT,RPT
CN=02206    C.C.  SEPOINT,RSTORE
CN=02207    C.C.  SEPOINT,RSTOREQ
CN=02208    C.C.  SEPOINT,SETCLOCK
CN=02209    C.C.  SEPOINT,START2
CN=02210    C.C.  SEPOINT,UST
CN=02211    C.C.  SEPOINT,FDPBOXS
CN=02212    C.C.  SEPOINT,BCDBOXS
CN=02213    C.C.  RECORD
CN=02214    C.C.  ORIGIN,LSTLGTH
CN=02215    IDC   OVPRO      PROG.LENGTH=02056          08/31/70
CN=02262    C.C.  RECORD
CN=02263    C.C.  ORIGIN,BINBUF
```

```
CN=02264     IDC   PROTECT    PROG.LENGTH=00310          07/11/70
CN=02275     IDC   POSTLOAD   PROG.LENGTH=02772          08/31/70
CN=02335     C.C.  COMMON ABS           * * * * * COMMON ABSOLUTE ROUTINES * * * *
CN=02336     C.C.  RECORD
CN=02337     C.C.  ORIGIN,BINBUF+50
CN=02338     IDC   MSIOCCP    PROG.LENGTH=06312          09/18/70
CN=02464     IDC   OC         PROG.LENGTH=01347          07/11/70
CN=02494     IDC   AREM       PROG.LENGTH=04264          08/31/70
CN=02549     IDC   MISC       PROG.LENGTH=01365          07/11/70
CN=02581     C.C.  RECORD
CN=02582     C.C.  ORIGIN,PROMEM
CN=02583     IDC   COMPASS    PROG.LENGTH=04746
CN=02688     C.C.  RECORD
CN=02689     C.C.  ORIGIN,PROMEM,4746
CN=02690     IDC   OVERLAY1   PROG.LENGTH=10370
CN=02744     C.C.  RECORD
CN=02745     C.C.  ORIGIN,PROMEM,4746
CN=02746     IDC   PASSONE    PROG.LENGTH=13216          08/31/70
CN=02856     C.C.  RECORD
CN=02857     C.C.  ORIGIN,PROMEM,4746
CN=02858     IDC   PASSTWO    PROG.LENGTH=13403
CN=02974     IDC   SYMTBLE    PROG.LENGTH=00001
CN=02978     C.C.  RECORD
CN=02979     C.C.  ORIGIN,PROMEM,4746
CN=02980     IDC   CRT        PROG.LENGTH=06645
CN=03013     C.C.  RECORD
CN=03014     C.C.  ORIGIN,PROMEM,74
CN=03015     IDC   ALG0       PROG.LENGTH=01752
CN=03050     C.C.  RECORD
CN=03051     C.C.  ORIGIN,PROMEM,74
CN=03052     IDC   ALG1       PROG.LENGTH=10731
CN=03193     C.C.  RECORD
CN=03194     C.C.  ORIGIN,PROMEM,74
CN=03195     IDC   ALG2       PROG.LENGTH=03053
CN=03248     C.C.  RECORD
CN=03249     C.C.  ORIGIN,PROMEM,74
CN=03250     IDC   ALG3       PROG.LENGTH=12744
CN=03437     C.C.  RECORD
CN=03438     C.C.  ORIGIN,PROMEM,74
CN=03439     IDC   ALG4       PROG.LENGTH=14664
CN=03643     C.C.  RECORD
CN=03644     C.C.  ORIGIN,PROMEM,74
CN=03645     IDC   ALG5       PROG.LENGTH=04317
CN=03701     C.C.  RECORD
CN=03702     C.C.  ORIGIN,PROMEM,240
CN=03703     IDC   FTN1       PROG.LENGTH=15120
CN=03874     C.C.  RECORD
CN=03875     C.C.  ORIGIN,PROMEM,6763
CN=03876     IDC   FTN2       PROG.LENGTH=15020
CN=03947     C.C.  RECORD
CN=03948     C.C.  ORIGIN,PROMEM,1621
CN=03949     IDC   FTN3       PROG.LENGTH=10230          09/04/70
CN=04028     C.C.  RECORD
CN=04029     C.C.  ORIGIN,PROMEM,1621
CN=04030     IDC   FTN4       PROG.LENGTH=14537
```

```
CN=04165    C.C.  RECORD
CN=04166    C.C.  ORIGIN,PROMEM,1703
CN=04167    IDC   FTN5      PROG.LENGTH=11070
CN=04257    C.C.  RECORD
CN=04258    C.C.  ORIGIN,PROMEM,1703
CN=04259    IDC   FTN6      PROG.LENGTH=04011           08/30/70
CN=04296    C.C.  RECORD
CN=04297    C.C.  ORIGIN,PROMEM,1613
CN=04298    IDC   FTNE      PROG.LENGTH=12473
CN=04443    C.C.  RECORD,CLST
CN=04444    C.C.  ORIGIN,LOADMSIO
CN=04445    IDC   MSIOMSOV  PROG.LENGTH=01044
CN=04470    IDC   OC        PROG.LENGTH=01347           07/11/70
CN=04500    IDC   MISC      PROG.LENGTH=01365           07/11/70
CN=04532    C.C.  RECORD
CN=04533    C.C.  ORIGIN,LOADMSIO
CN=04534    IDC   MSIOTPOV  PROG.LENGTH=04210           09/04/70
CN=04604    C.C.  RECORD
CN=04605    C.C.  ORIGIN,LOADMSIO
CN=04606    IDC   MSIOSTER  PROG.LENGTH=00201
CN=04626    IDC   MTWPR     PROG.LENGTH=00010
CN=04631    IDC   MTRPR     PROG.LENGTH=00020
CN=04637    IDC   MTLDACPR  PROG.LENGTH=00010
CN=04642    IDC   MTDER     PROG.LENGTH=00020
CN=04648    IDC   PRCPR     PROG.LENGTH=00016
CN=04653    IDC   CRDER     PROG.LENGTH=00041
CN=04660    IDC   CPDER     PROG.LENGTH=00113
CN=04668    IDC   MTWPRR    PROG.LENGTH=00274
CN=04679    IDC   MTRPRR    PROG.LENGTH=00210
CN=04690    IDC   MTLDCPRR  PROG.LENGTH=00123
CN=04698    IDC   CMNRTNS   PROG.LENGTH=00270
CN=04719    IDC   NRC       PROG.LENGTH=00112
CN=04729    IDC   NWR       PROG.LENGTH=00054
CN=04735    IDC   T.NOTRDY  PROG.LENGTH=00032
CN=04741    IDC   TYPEOUT.  PROG.LENGTH=00205
CN=04751    IDC   WHATKIND  PROG.LENGTH=00037
CN=04756    C.C.  BCD                 * * * * * * * BCD ROUTINES * * * * * * *
CN=04757    C.C.  RECORD
CN=04758    C.C.  ORIGIN,LOADMSIO
CN=04759    IDC   MSIOGPRW  PROG.LENGTH=04101
CN=04831    C.C.  FILE
CN=04832    C.C.  REPLACE
CN=04833    IDC   MSIOMAIN  PROG.LENGTH=03165
CN=04908    IDC   MRESTART  PROG.LENGTH=01710
CN=04946    IDC   COBOL     PROG.LENGTH=01214
CN=04984    IDC   COBOLIE   PROG.LENGTH=10046
CN=05119    IDC   COBOLD1   PROG.LENGTH=10053
CN=05252    IDC   COBOLP1   PROG.LENGTH=06601
CN=05377    IDC   COBOLD2   PROG.LENGTH=04346
CN=05462    IDC   COBOLP2   PROG.LENGTH=06504
CN=05577    IDC   COBOLDP3  PROG.LENGTH=05725
CN=05678    IDC   COBOLIO   PROG.LENGTH=00206
CN=05687    IDC   FIGCON    PROG.LENGTH=00033
CN=05692    IDC   TRANSMIT  PROG.LENGTH=00103
CN=05699    IDC   COMPARE   PROG.LENGTH=00106
```

```
CN=05707     IDC   EDIT       PROG.LENGTH=00105
CN=05714     IDC   EDITCOBL   PROG.LENGTH=00243
CN=05732     IDC   MULTIPLY   PROG.LENGTH=00062
CN=05739     IDC   DIVIDE     PROG.LENGTH=00067
CN=05746     IDC   BMULTPLY   PROG.LENGTH=00055
CN=05753     IDC   BDIVIDE    PROG.LENGTH=00053
CN=05760     IDC   ERRSTOP    PROG.LENGTH=00027
CN=05765     IDC   TYPELOOP   PROG.LENGTH=00051
CN=05772     IDC   DPMULDIV   PROG.LENGTH=00570
CN=05790     IDC   STRIPPER   PROG.LENGTH=00235
CN=05801     IDC   BSTRIPPR   PROG.LENGTH=00321
CN=05813     IDC   SUBSCRP    PROG.LENGTH=00146
CN=05824     IDC   DEEDIT     PROG.LENGTH=00120
CN=05831     IDC   EXAMINE    PROG.LENGTH=00111
CN=05838     IDC   NUMERIC    PROG.LENGTH=00030
CN=05843     IDC   ALPHABET   PROG.LENGTH=00031
CN=05848     IDC   ACCEPT     PROG.LENGTH=00210
CN=05857     IDC   DISPLAY    PROG.LENGTH=00425
CN=05872     IDC   MVFIGCON   PROG.LENGTH=00025
CN=05877     IDC   VARC1      PROG.LENGTH=00054
CN=05883     IDC   VARN       PROG.LENGTH=00050
CN=05889     IDC   VARAN      PROG.LENGTH=00104
CN=05896     IDC   ROUNDER    PROG.LENGTH=00006
CN=05901     IDC   CONVERT    PROG.LENGTH=00151
CN=05909     IDC   ZIPPER     PROG.LENGTH=00103
CN=05918     IDC   LOGICAL    PROG.LENGTH=00244
CN=05929     IDC   MSSORT     PROG.LENGTH=02721
CN=06001     IDC   MDYNALL    PROG.LENGTH=00557
CN=06010     IDC   MSSEDIT    PROG.LENGTH=07772
CN=06121     IDC   MSSIOP1    PROG.LENGTH=02625          10/25/70
CN=06185     IDC   MSSINTS    PROG.LENGTH=01724
CN=06223     IDC   MSSIOP2    PROG.LENGTH=04446
CN=06319     IDC   MSSMERG    PROG.LENGTH=02754
CN=06375     IDC   SORT       PROG.LENGTH=00477
CN=06398     IDC   TDYNALL    PROG.LENGTH=00573
CN=06408     IDC   BINANDEC   PROG.LENGTH=00070
CN=06415     IDC   SDUMP      PROG.LENGTH=00135
CN=06422     IDC   SRTPRINT   PROG.LENGTH=00243
CN=06432     IDC   SORTPOLY   PROG.LENGTH=00251
CN=06453     IDC   SRTMBALF   PROG.LENGTH=00007
CN=06462     IDC   SRTMBALB   PROG.LENGTH=00007
CN=06471     IDC   SRTMPOLF   PROG.LENGTH=00007
CN=06480     IDC   SRTMPOLB   PROG.LENGTH=00006
CN=06489     IDC   SORTPDMY   PROG.LENGTH=00076
CN=06505     IDC   SORTEDIT   PROG.LENGTH=10042
CN=06630     IDC   SORTIOP1   PROG.LENGTH=02622
CN=06690     IDC   SORTPHS1   PROG.LENGTH=02170
CN=06747     IDC   SRTRBALF   PROG.LENGTH=00003
CN=06752     IDC   SRTRPOLF   PROG.LENGTH=00003
CN=06757     IDC   SRTRBALB   PROG.LENGTH=00003
CN=06762     IDC   SRTRPOLB   PROG.LENGTH=00003
CN=06767     IDC   SRESTART   PROG.LENGTH=01121
CN=06794     IDC   RESTART1   PROG.LENGTH=00105
CN=06801     IDC   RSTRTDUM   PROG.LENGTH=00002
CN=06805     IDC   SORTIOP2   PROG.LENGTH=04313
```

```
CN=06899    IDC   SRTEQUAL  PROG.LENGTH=00245
CN=06909    IDC   POLYFORW  PROG.LENGTH=02160
CN=06950    IDC   BALCFORW  PROG.LENGTH=02277
CN=06993    IDC   BALCBACK  PROG.LENGTH=02470
CN=07040    IDC   POLYBACK  PROG.LENGTH=02300
CN=07083    IDC   WAITBEEP  PROG.LENGTH=00055
CN=07090    C.C.   BDP                  * * * * * * * BDP ROUTINES * * * * * * *
CN=07091    C.C.   RECORD
CN=07092    C.C.   ORIGIN,LOADMSIO
CN=07093    IDC   MSIOGPRW  PROG.LENGTH=04101          08/29/70
CN=07165    C.C.   FILE
CN=07166    C.C.   REPLACE
CN=07167    IDC   MSIOMAIN  PROG.LENGTH=03165
CN=07242    IDC   MRESTART  PROG.LENGTH=01710
CN=07280    IDC   COBOL     PROG.LENGTH=01214
CN=07318    IDC   COBOLIE   PROG.LENGTH=10052
CN=07453    IDC   COBOLD1   PROG.LENGTH=10055
CN=07586    IDC   COBOLP1   PROG.LENGTH=06633
CN=07713    IDC   COBOLD2   PROG.LENGTH=04347
CN=07798    IDC   COBOLP2   PROG.LENGTH=07075
CN=07923    IDC   COBOLDP3  PROG.LENGTH=05727
CN=08024    IDC   COBOLIO   PROG.LENGTH=00206
CN=08033    IDC   ERRSTOP   PROG.LENGTH=00027
CN=08038    IDC   TYPELOOP  PROG.LENGTH=00051
CN=08045    IDC   FIGCON    PROG.LENGTH=00044
CN=08050    IDC   COMPARE   PROG.LENGTH=00062
CN=08055    IDC   MULTIPLY  PROG.LENGTH=00151
CN=08064    IDC   DIVIDE    PROG.LENGTH=00125
CN=08072    IDC   STRIPPER  PROG.LENGTH=00307
CN=08084    IDC   SUBSCRP   PROG.LENGTH=00154
CN=08093    IDC   DEEDIT    PROG.LENGTH=00120
CN=08100    IDC   EXAMINE   PROG.LENGTH=00111
CN=08107    IDC   NUMERIC   PROG.LENGTH=00030
CN=08112    IDC   ALPHABET  PROG.LENGTH=00031
CN=08117    IDC   ACCEPT    PROG.LENGTH=00210
CN=08126    IDC   DISPLAY   PROG.LENGTH=00425
CN=08141    IDC   MVFIGCON  PROG.LENGTH=00025
CN=08146    IDC   VARC1     PROG.LENGTH=00054
CN=08152    IDC   VARN      PROG.LENGTH=00051
CN=08158    IDC   VARAN     PROG.LENGTH=00121
CN=08165    IDC   DPBINBCD  PROG.LENGTH=00041
CN=08171    IDC   DPBCDBIN  PROG.LENGTH=00051
CN=08177    IDC   ZIPPER    PROG.LENGTH=00104
CN=08186    IDC   LOGICAL   PROG.LENGTH=00250
CN=08197    IDC   MSSORT    PROG.LENGTH=02671
CN=08268    IDC   MDYNALL   PROG.LENGTH=00557
CN=08277    IDC   MSSEDIT   PROG.LENGTH=07772
CN=08388    IDC   MSSIOP1   PROG.LENGTH=02625          10/25/70
CN=08452    IDC   MSSINTS   PROG.LENGTH=01724
CN=08490    IDC   MSSIOP2   PROG.LENGTH=04446
CN=08586    IDC   MSSMERG   PROG.LENGTH=02754
CN=08642    IDC   SORT      PROG.LENGTH=00447
CN=08664    IDC   TDYNALL   PROG.LENGTH=00573
CN=08674    IDC   BINANDEC  PROG.LENGTH=00070
CN=08681    IDC   SDUMP     PROG.LENGTH=00135
```

```
CN=08688     IDC   SRTPRINT   PROG.LENGTH=00243
CN=08698     IDC   SORTPOLY   PROG.LENGTH=00251
CN=08719     IDC   SRTMBALF   PROG.LENGTH=00007
CN=08728     IDC   SRTMBALB   PROG.LENGTH=00007
CN=08737     IDC   SRTMPOLF   PROG.LENGTH=00007
CN=08746     IDC   SRTMPOLB   PROG.LENGTH=00006
CN=08755     IDC   SORTPDMY   PROG.LENGTH=00076
CN=08771     IDC   SORTEDIT   PROG.LENGTH=10042
CN=08896     IDC   SORTIOP1   PROG.LENGTH=02622
CN=08956     IDC   SORTPHS1   PROG.LENGTH=02170
CN=09013     IDC   SRTRBALF   PROG.LENGTH=00003
CN=09018     IDC   SRTRPOLF   PROG.LENGTH=00003
CN=09023     IDC   SRTRBALB   PROG.LENGTH=00003
CN=09028     IDC   SRTRPOLB   PROG.LENGTH=00003
CN=09033     IDC   SRESTART   PROG.LENGTH=01121
CN=09060     IDC   RESTART1   PROG.LENGTH=00105
CN=09067     IDC   RSTRTDUM   PROG.LENGTH=00002
CN=09071     IDC   SORTIOP2   PROG.LENGTH=04313
CN=09165     IDC   SRTEQUAL   PROG.LENGTH=00245
CN=09175     IDC   POLYFORW   PROG.LENGTH=02160
CN=09216     IDC   BALCFORW   PROG.LENGTH=02277
CN=09259     IDC   BALCBACK   PROG.LENGTH=02470
CN=09306     IDC   POLYBACK   PROG.LENGTH=02300
CN=09349     IDC   WAITBEEP   PROG.LENGTH=00055
CN=09356     C.C.   COMMON REL          * * * * COMMON RELOCATABLE ROUTINES * * * *
CN=09357     IDC   COMPASSB   PROG.LENGTH=00051
CN=09366     IDC   RAAR       PROG.LENGTH=00071
CN=09372     IDC   SCAR       PROG.LENGTH=00337
CN=09384     IDC   MTWPR      PROG.LENGTH=00010
CN=09389     IDC   MTRPR      PROG.LENGTH=00020
CN=09395     IDC   MTLDACPR   PROG.LENGTH=00010
CN=09400     IDC   MTDER      PROG.LENGTH=00020
CN=09406     IDC   PRCPR      PROG.LENGTH=00016
CN=09411     IDC   CRDER      PROG.LENGTH=00041
CN=09418     IDC   CPDER      PROG.LENGTH=00113
CN=09426     IDC   MTWPRR     PROG.LENGTH=00274
CN=09437     IDC   MTRPRR     PROG.LENGTH=00210
CN=09448     IDC   MTLDCPRR   PROG.LENGTH=00123
CN=09456     IDC   CMNRTNS    PROG.LENGTH=00270
CN=09477     IDC   WHATISIT   PROG.LENGTH=00336
CN=09510     IDC   NRD        PROG.LENGTH=00122
CN=09518     IDC   NRC        PROG.LENGTH=00112
CN=09528     IDC   NWR        PROG.LENGTH=00054
CN=09534     IDC   T.NOTRDY   PROG.LENGTH=00032
CN=09540     IDC   TYPEOUT.   PROG.LENGTH=00205
CN=09550     IDC   WHATKIND   PROG.LENGTH=00037
CN=09555     IDC   COSYRDWT   PROG.LENGTH=02010
CN=09603     IDC   COSY       PROG.LENGTH=05775
CN=09709     IDC   FORTRAN    PROG.LENGTH=00163
CN=09721     IDC   FLOVER     PROG.LENGTH=00252
CN=09731     IDC   BCDINP     PROG.LENGTH=01137
CN=09755     IDC   BCDOUT     PROG.LENGTH=01635
CN=09788     IDC   FORMAT     PROG.LENGTH=00413
CN=09801     IDC   BINARY     PROG.LENGTH=01010
CN=09815     IDC   BUFFER     PROG.LENGTH=00510
```

```
CN=09831     IDC   UNIT        PROG.LENGTH=00102
CN=09837     IDC   IOCHK       PROG.LENGTH=00043
CN=09843     IDC   EOFCHK      PROG.LENGTH=00050
CN=09849     IDC   TAPEHAND    PROG.LENGTH=00222
CN=09858     IDC   PAUSE       PROG.LENGTH=00065
CN=09864     IDC   CONTROL     PROG.LENGTH=00776
CN=09887     IDC   CIO.MSIO    PROG.LENGTH=00422
CN=09900     IDC   LOCATE      PROG.LENGTH=00074
CN=09906     IDC   DOUBLE      PROG.LENGTH=01250
CN=09937     IDC   DFPRIME     PROG.LENGTH=01236
CN=09963     IDC   DFP         PROG.LENGTH=01226
CN=09993     IDC   Q1QADRI     PROG.LENGTH=00165
CN=10003     IDC   ITOJ        PROG.LENGTH=00143
CN=10010     IDC   ITOX        PROG.LENGTH=00022
CN=10015     IDC   XTOI        PROG.LENGTH=00223
CN=10023     IDC   POWRF       PROG.LENGTH=00103
CN=10029     IDC   SINCOS      PROG.LENGTH=00305
CN=10039     IDC   ATANF       PROG.LENGTH=00141
CN=10046     IDC   EXPF        PROG.LENGTH=00123
CN=10053     IDC   LOGF        PROG.LENGTH=00200
CN=10061     IDC   SIGNF       PROG.LENGTH=00015
CN=10065     IDC   SQRTF       PROG.LENGTH=00101
CN=10071     IDC   ABSF        PROG.LENGTH=00011
CN=10075     IDC   EXTREMA1    PROG.LENGTH=00105
CN=10082     IDC   EXTREMA2    PROG.LENGTH=00053
CN=10088     IDC   FLOATF      PROG.LENGTH=00014
CN=10093     IDC   FIXF        PROG.LENGTH=00040
CN=10098     IDC   MASKINGF    PROG.LENGTH=00040
CN=10102     IDC   FAULTS      PROG.LENGTH=00034
CN=10106     IDC   SENSLITE    PROG.LENGTH=00113
CN=10113     IDC   SENSWTCH    PROG.LENGTH=00044
CN=10119     IDC   Q8QERROR    PROG.LENGTH=00245
CN=10129     IDC   ALGOL       PROG.LENGTH=00454
CN=10140     IDC   ALGLIB00    PROG.LENGTH=02671
CN=10194     IDC   ALGLIB01    PROG.LENGTH=00773
CN=10214     IDC   ALGLIB02    PROG.LENGTH=05732
CN=10323     IDC   ALGLIB03    PROG.LENGTH=00771
CN=10346     IDC   ALGLIB04    PROG.LENGTH=01455
CN=10379     IDC   ALGLIB05    PROG.LENGTH=00703
CN=10401     IDC   ALGLIB06    PROG.LENGTH=00763
CN=10424     IDC   ALGOLRUN    PROG.LENGTH=01675
CN=10469     C.C.    MACRO,COMAC
               MACRO   OPEN
               MACRO   CLOSE
               MACRO   ALLOCATE
               MACRO   RELEASE
               MACRO   EXPAND
               MACRO   MODIFY
               MACRO   FILEID
               MACRO   READS
               MACRO   WRITES
               MACRO   READB
               MACRO   REWIND
               MACRO   UNLOAD
               MACRO   BKSP
```

```
MACRO   SEFF
MACRO   SEFB
MACRO   WEOF
MACRO   ERASE
MACRO   STATUS
MACRO   FORMAT
MACRO   TRANSMIT
MACRO   EDIT.
MACRO   COMPARE
MACRO   MULTIPLY
MACRO   DIVIDE
MACRO   OPENF
MACRO   CLOSEF
MACRO   READF
MACRO   WRITEF
MACRO   PAUSEF
MACRO   GET
MACRO   PUT
MACRO   RELSE
MACRO   LOCATE
MACRO   CHECKPT
MACRO   FILEDESC
MACRO   LABELING
MACRO   VARIABLE
MACRO   STOPOPEN
MACRO   RERUN
MACRO   INPUT
MACRO   READOCR
MACRO   POSITION
MACRO   ZMIRR
MACRO   LINELOC
MACRO   ADVANCC
MACRO   CLEARC
MACRO   MIRSTAT
MACRO   STPREAD
MACRO   SORT
MACRO   ALARM
MACRO   MARK
MACRO   FRMTOCR
MACRO   WHATKIND
MACRO   LOWREJ
MACRO   SCAR
MACRO   SCARNM
MACRO   LOWDATA
MACRO   NRD
MACRO   REPLACEF
MACRO   UPDATEF
MACRO   ISCLOSEF
MACRO   EXPANDSF
MACRO   FINDF
MACRO   INSERTF
MACRO   DELETEF
MACRO   FILEDEFF
MACRO   RECDEFF
MACRO   BUFDEFF
```

```
                 MACRO   ISOPENF
                 MACRO   BUILDF
                 MACRO   GETF
                    END
CN=11459      IDC   EXECOVR    PROG.LENGTH=00166          07/11/70
CN=11467      IDC   SUMS       PROG.LENGTH=02222          07/11/70
CN=11508      IDC   MSUTIL     PROG.LENGTH=03067          10/25/70
CN=11563      IDC   UDUMP      PROG.LENGTH=00516          08/31/70
CN=11581      IDC   LOAD       PROG.LENGTH=00604          07/11/70
CN=11599      IDC   ENTER      PROG.LENGTH=02100          08/31/70
CN=11618      IDC   DMSD       PROG.LENGTH=02263          07/11/70
CN=11633      IDC   PSM        PROG.LENGTH=00231          07/11/70
CN=11643      IDC   PSM2       PROG.LENGTH=00570          07/11/70
CN=11667      IDC   FLL2       PROG.LENGTH=00562          08/31/70
CN=11697      IDC   FLL        PROG.LENGTH=00522          07/11/70
CN=11706      IDC   PFLD       PROG.LENGTH=00532          07/11/70
CN=11716      IDC   CLRDSK     PROG.LENGTH=00204          07/11/70
CN=11726      IDC   WTADR      PROG.LENGTH=00076          07/11/70
CN=11733      IDC   MSDLRW     PROG.LENGTH=00464          07/11/70
CN=11741      IDC   PRINT3     PROG.LENGTH=00175          07/11/70
CN=11750      IDC   BINTOBCD   PROG.LENGTH=00022          07/11/70
CN=11755      IDC   SPLTDATE   PROG.LENGTH=00045          07/11/70
CN=11761      IDC   DTB        PROG.LENGTH=00027          07/11/70
CN=11766      IDC   BDTRCK     PROG.LENGTH=10217          07/11/70
CN=11838      MAIN   55                                       PRELIB
CN=11839      IDC   PRELIB     PROG.LENGTH=01153          07/11/70
CN=11879      OV     55,1                                     PLOVINT
CN=11880      IDC   PLOVINT    PROG.LENGTH=01643          09/01/70
CN=11917      OV     55,2                                     OV1
CN=11918      IDC   OV1        PROG.LENGTH=02230          07/11/70
CN=11978      SEG    55,1                                     PHASE1
CN=11979      IDC   PHASE1     PROG.LENGTH=02267          09/01/70
CN=12024      SEG    55,2                                     PHASE2
CN=12025      IDC   PHASE2     PROG.LENGTH=01541          08/31/70
CN=12059      OV     55,3                                     PLOVINT1
CN=12060      IDC   PLOVINT1   PROG.LENGTH=00241          08/31/70
CN=12074      OV     55,4                                     PLOV2
CN=12075      IDC   PLOV2      PROG.LENGTH=04550          10/25/70
CN=12134      OV     55,5                                     PLOV3
CN=12135      IDC   PLOV3      PROG.LENGTH=00723
CN=12164      IDC   SNAPSHOT   PROG.LENGTH=01450          07/11/70
CN=12192      IDC   IOPACK     PROG.LENGTH=00501          07/11/70
CN=12210      IDC   IODRAIN    PROG.LENGTH=00024          07/11/70
CN=12215      IDC   FDPBOXS    PROG.LENGTH=01221          07/11/70
CN=12242      IDC   BCDBOXS    PROG.LENGTH=00443          06/21/70
CN=12257      IDC   OPTBOXS    PROG.LENGTH=00073          07/11/70
CN=12263      IDC   EXPANDSF   PROG.LENGTH=00137
CN=12272      IDC   REPLACE    PROG.LENGTH=00060          08/29/70
CN=12279      IDC   DELETE     PROG.LENGTH=00114          08/29/70
CN=12289      IDC   INSERT     PROG.LENGTH=00713          08/29/70
CN=12313      IDC   UPDATE     PROG.LENGTH=00573          08/29/70
CN=12338      IDC   ISGET      PROG.LENGTH=00445          08/29/70
CN=12361      IDC   BUILD      PROG.LENGTH=00427          08/29/70
CN=12376      IDC   BLOCKER    PROG.LENGTH=00274          08/29/70
CN=12393      IDC   FILEDEF    PROG.LENGTH=00037          08/29/70
```

```
CN=12398     IDC   RECDEF     PROG.LENGTH=00041          08/29/70
CN=12404     IDC   BUFDEF     PROG.LENGTH=00065          08/29/70
CN=12412     IDC   FILEDEFF   PROG.LENGTH=00030          08/29/70
CN=12417     IDC   RECDEFF    PROG.LENGTH=00142          08/29/70
CN=12425     IDC   BUFDEFF    PROG.LENGTH=00075          08/29/70
CN=12433     IDC   FD         PROG.LENGTH=02107          08/29/70
CN=12514     IDC   PLIBEDIT   PROG.LENGTH=04035          08/31/70
CN=12587     LED    0101,0201
CN=12589     IDC   MSOSUTIL   PROG.LENGTH=01063
CN=12618     IDC   COPYS      PROG.LENGTH=00472
CN=12633     IDC   COPYT      PROG.LENGTH=00741
CN=12653     IDC   VERIFY     PROG.LENGTH=01174
CN=12679     IDC   DUMP       PROG.LENGTH=00330
CN=12691     IDC   ERROR      PROG.LENGTH=01114
CN=12716     IDC   BSIPP      PROG.LENGTH=11354
CN=12824     IDC   SIPP       PROG.LENGTH=06262
CN=12936     IDC   SFP        PROG.LENGTH=06414          07/11/70
CN=13013     IDC   OC         PROG.LENGTH=01347          07/11/70
CN=13043     IDC   AREM       PROG.LENGTH=04264          08/31/70
CN=13098     IDC   MISC       PROG.LENGTH=01365          07/11/70
CN=13130      C.C.  FILE
            EOF
            ENDSCOPE,R
            EOF

I PLIBEDIT 099 PLIBEDIT COMPLETED.
```

4.6 FINAL LIBRARY GENERATION

To generate the final MSOS library, installation must have the following:

An operable interim library

The appropriate autoload card

A completed PRELIB source (output of Section 4.5)

The following procedure is an example of the steps which may be used for final library generation.

1. Place the PRELIB source deck tape on magnetic tape 1.[†]

2. Autoload interim library.

3. Define the interim library system:

   Upon completion of autoload, the following is typed:

Message:	EDITION
Type:	IL
Message:	*INTERIM SYSTEM*
	DEFINE ENVIRONMENT

   In the following console typewriter inputs, underlined characters are site dependent:

   LIB = DPC2EQU10

   DT = 853

   INP = MTCOEQU01 (PRELIB source deck tape is standard input)[††]

   CDT = 657

   OUT = PRCOE2U00

   DT = 512

   PUN = CPCOE1U00

   CDT = 3446

   SCR = DPCOEOU11

   DT = 853

   SCR =

[†] This step assumes previous PLIBEDIT run to create the PRELIB source deck tape.
[††] PRELIB source tape is the new PLIBEDIT file (NPF) defined in Part III, Section 10.

4. The interim system is defined; press FINISH. The following is typed:

Message:	DATE
Type:	mmddyy

To execute a PRELIB on an 853 or 852 mass storage pack, it is necessary to EQUIP and ENTER another mass storage device into the system for PRELIB's files. This procedure is shown in step 5. It is also recommended that the same procedure be used for 854 or 841 mass storage devices. With the use of an 854 or 841 scratch pack, the efficiency of PRELIB is increased.

Set:	SELECT JUMP 6
Press:	FINISH
Message:	TIME
Type:	hhmm
Press:	FINISH

With SELECT JUMP 6 on, the following output occurs:

Message:	PUN
Press:	FINISH
Message:	OUT
Press:	FINISH
Message:	INP
Press:	FINISH
Message:	CFO
Press:	FINISH

5. After SEQUENCE is typed:

Type:	EQUIP, 02=DPC2EOU11
Press:	FINISH
Type:	CALL, MSUTIL, 58
Press:	FINISH
Type:	ENTER, 2, 853/2, S
Press:	FINISH
Type:	SCOPE
Press:	FINISH
Type:	RAT, 853/2
Press:	FINISH
Clear:	SELECT JUMP 6
Type:	. (a period)
Press:	FINISH

When ENDSCOPE is typed, the PRELIB run is complete. Check the listing carefully to verify that it is correct.

6. The final library has been generated. If it is a Memory Protect system with the 3311 multi-programming option, refer to Part III, Section 2 for page index file setting.

7. Autoload final library. To examine system files:

   Type: CALL, MSUTIL, 58
   Press: FINISH

   Type: LIST, 61, MSD
   Press: FINISH

   Type: LIST, 61, FLD
   Press: FINISH

8. Step 12 on page II-3-5 must be complete at this time.

9. Prepare the Logical MSIO overlay file if necessary.

   FET, MSOS, L-MSIO, 10240, 00, 0000, 0000

   ALLOCATE, X, 991231, , NOSEG

	813	841	852	853	854	863	
x=	5	6	23	10	10	10	tracks

10. Execute verification programs on the MSOS product set contained in package A (Part II, Section 6).

11. The final MSOS library is complete. Proceed to Part II, Section 5, for incorporating any of the following product set members.

   SAINT
   RESPOND
   ADAPT
   PERT COST
   PERT TIME
   USASI FORTRAN
   USASI COBOL

4.7 ENHANCED MAGNETIC TAPE ERROR RECOVERY

The enhanced magnetic tape error recovery release is provided to upgrade the MSOS magnetic tape drivers and the standard error recovery routines under MSOS 4.2. A summary of these enhancements involves:

- Tape mark integrity that ensures all tape marks are written in the correct mode.
- An improved algorithm to facilitate positioning during write recovery. A controlled backspace feature is also provided as an installation option for this enhancement.
- Rewrite verification provided to ensure the readability of a rewritten record. The rewrite verification is performed in both directions within the area between the last good block and the rewritten block.
- Intentional system noise records written before and after erased areas to improve tape positioning from both directions.
- Noise recovery performed on all forward reads, backward reads, and backspaces checked by the recovery package. Records of less than 18 characters are discarded as noise records during read recovery. Records read as noise in one direction must also be noise length in the opposite direction before the recovery routine discards them.

4.7.1 RELEASE MATERIALS

The release materials provided for this update consist of a tape comprised of thirty files.

File one consists of binary decks for the following routines. Where applicable, those assembly options that are turned on are indicated in parentheses.

RAAR	
SCAR	
MTRPR	
MTDER	
CRDER	
CPDER	
MTWPRR	
MTRPRR	
MTLDCPRR	
MRTNS	(READB)
CMNRTNS	
WHATISIT	
NRD	(READB)
NRC	
NWR	
T.NOTRDY	
TYPEOUT.	
CIO	

BCIO	
DRIV607	(WRITEB=0)
DRIV606	(WRITEB=0)
DRIV3528	(WRITEB=0)
RDUMP	(SER. OPT, READB)
PRELOAD	(READB)
LOADER	(LERP, READB)
POSTLOAD	(LERP, READB)
MSIOCCP	(READB)
MSIOSTER	
MSIOMAIN	

Files 2 through 29 consist of the enhanced tape recovery corrections written in COSY format designed for input to COSY*. Each deck that has changes has a corresponding file of changes on the release tape.

File 30 consists of the COSY file for the MTRTNS routine. The file is used as input to COSY when MTRTNS must be reassembled.

4.7.2 INSTALLATION CONSIDERATIONS

If the controlled backspace capability is present, the MTRTNS routine should be assembled from the release tape with the WRITEB option turned on. The WRITEB option is also applicable to the variable resident routines and magnetic tape driver(s).

If several types of tape drives (606's, 607's and 65X's) are available, separate editions are recommended for each type. If separate editions are not used, possible conflicts can arise due to duplicated entry points if both DRIV606 and DRIV607 are in resident, or conflicting recovery procedures are being attempted when drives with and without controlled backspace are on the system, and WRITEB is turned on in MTRTNS. There is, however, complete compatibility for mixed drive types that are functionally the same in terms of all being specified with or without controlled backspace.

Installations having only MMTC drives must edit an EXS card into their PRELIB source deck to equate the new sepoint BK. SP to ABNORMAL (section 4.7.4).

4.7.3 INSTALLATION PROCEDURES

There are two possible installation procedures for the enhanced magnetic tape recovery release depending on whether the PSR summary 201 tape is being merged with the release tape or not.

Installation with PSR Summary 201

The procedures required for instllation of the enhanced magnetic tape recovery release combined with the PSR Summary 201 tape involve:

1. Those routines which are to be changed according to the user's site requirements are read from the release tape beginning with the second file onto an intermediate merge tape.

   Assembly option changes which apply to MSOS 4.2 COSY sequence numbers must be inserted at this time according to the sequence numbers specified in section 2.7 using the appropriate COSY* control cards (section III-14).

   The assembly options that are new with this release are specified relative to the release tape sequence numbers as indicated in section 4.7.5.

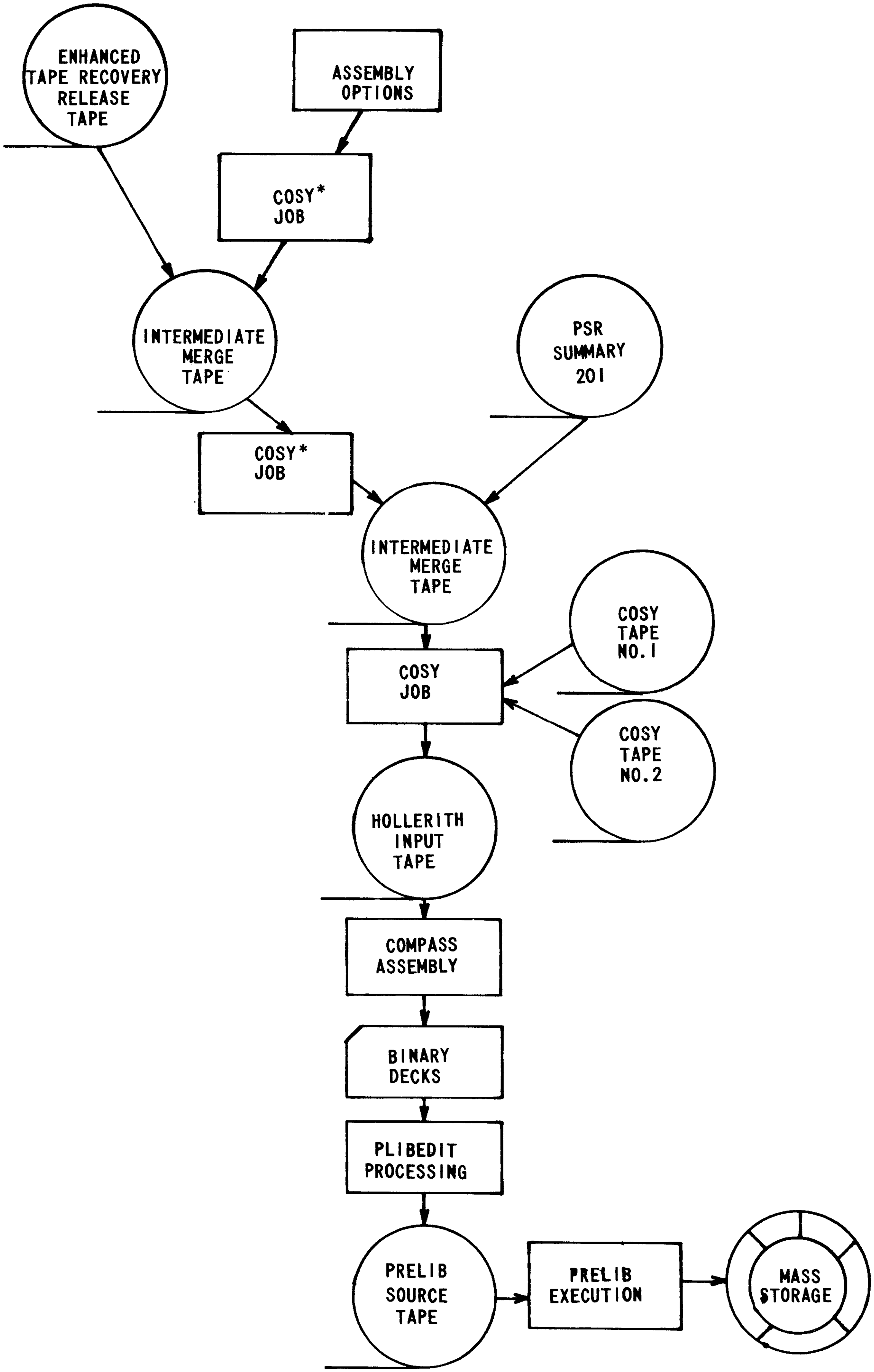
2. With the exception of the last deck read from the release tape, all other decks selected from the release tape must have the ENDCOSY* and END* control cards deleted. The sequence numbers for these cards are specified in section 4.7.5.

3. The intermediate tape constructed from the release tape is merged with the PSR Summary 201 tape resulting in a new intermediate tape. To reduce the impact of merge conflicts during this merging operation, COSY* should be instructed to suppress further job execution when conflicts occur.

4. The intermediate tape containing both release modifications and PSR Summary 201 corrections is used as input to COSY for the creation of the Hollerith input tape. This Hollerith tape is in turn used for COMPASS assembly of the binary source decks.

5. The assembled binary decks are used to create the PRELIB source tape (section 4.7.3).

The following is a graphic representation of these installation procedures.


ENHANCED TAPE RECOVERY RELEASE TAPE
ASSEMBLY OPTIONS
COSY* JOB
INTERMEDIATE MERGE TAPE
PSR SUMMARY 201
COSY* JOB
INTERMEDIATE MERGE TAPE
COSY TAPE NO. 1
COSY JOB
COSY TAPE NO. 2
HOLLERITH INPUT TAPE
COMPASS ASSEMBLY
BINARY DECKS
PLIBEDIT PROCESSING
PRELIB SOURCE TAPE
PRELIB EXECUTION
MASS STORAGE

Example:

```
[SEQUENCE,200
[JOB,,,,
[CTO,THIS JOB WORKS WITH SUMMARY LEVEL 201 CODE
[EQUIP,10=MT
[CTO,LUN 10 IS ITR RELEASE TAPE
[UTILITY,60,61.
SKFF,10.
SCOPE.
[EQUIP,30=MT
[CTO,LUN 30 IS A SCRATCH TAPE
[COSY,*
          INSERT*   1
          DELETE/   5
DP        EQU       1
          DELETE*   28,29
CIO       DECK*     I=10,H=30
MSIOMAIN DECK*      I=10,H=30
          END*
[[

[SEQUENCE,210
[JOB,,,,
[EQUIP,30=MT
[CTO,LUN 30 IS OUTPUT FROM PREVIOUS JOB
[EQUIP,20=MT
[CTO,LUN 20 IS SUMMARY 201 COSY* TAPE
[EQUIP,40=MT
[CTO,LUN 40 MERGE TAPE FOR ITR AND SUMM. 201 CORRECTIONS
[COSY,*,I=30,S
[[

[SEQUENCE,220
[JOB,,,,
[EQUIP,40=MT
[CTO,LUN 40 IS MERGED CORRECTIONS TAPE FROM LAST JOB
[EQUIP,01=MT
[CTO,LUN 1 IS MSOS V4.2 COSY 1
[EQUIP,02=MT
[CTO,LUN 2 IS MSOS V4.2 COSY 2
[EQUIP,05=MT
[CTO,LUN 5 IS SCRATCH FOR COMPASS INPUT
[COSY,I=40,S
          ENDCOSY/
[[

[SEQUENCE,230
[JOB,,,,
[EQUIP,05=MT
[CTO,LUN 5 IS THE COMPASS INPUT TAPE CREATED IN THE LAST JOB
[COMPASS,I=5,L,R,P
[[
```

Installation Without PSR Summary 201

The procedures required to install the enhanced magnetic tape recovery release without combining the release with the PSR Summary tape are as follows:

1. Those routines which are to be changed according to needs of the user's site are read from the release tape beginning with the second file via COSY* onto an intermediate tape.

   Assembly option changes requiring MSOS 4.2 COSY sequence numbers must be inserted at this time according to the sequence numbers specified in section 2.7 using the appropriate COSY* control cards (section III-14).

2. The COSY* control cards contained in the corrections must be deleted. The sequence numbers of the COSY* cards within each routine are specified in section 4.7.5.

3. All selected decks on the release tape must have the ENDCOSY* and END* control cards deleted. The sequence numbers for these cards are specified in section 4.7.5.

4. For each routine of the release tape being modified, the DECK* control card for the files must be replaced with a COSY control card of the form:

   routinename DECK/ I=n, H=5

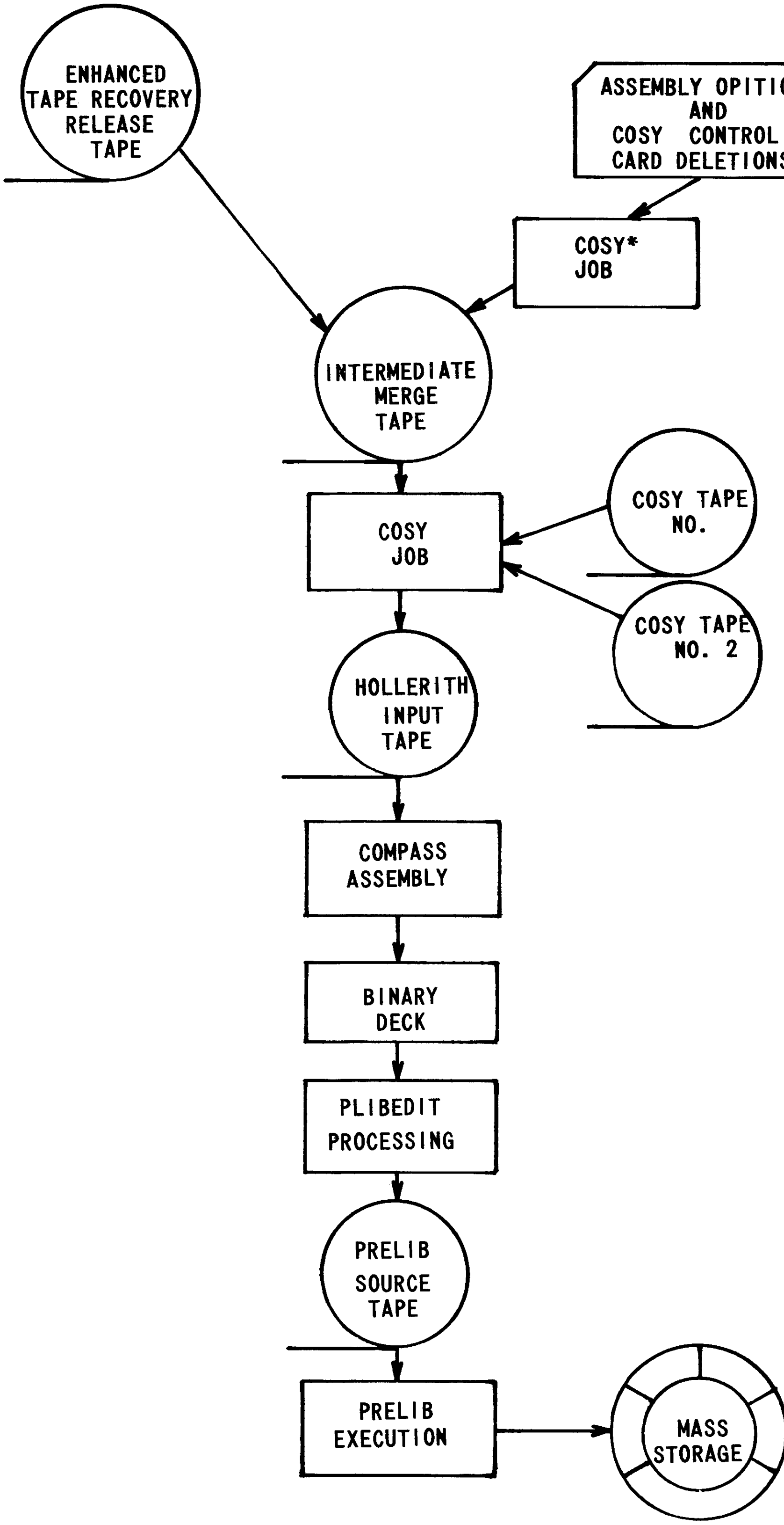
   n - indicates the number of the COSY tape (1 or 2) on which the routine to be changed is located.

5. The intermediate tape containing the modified routines is used as input to COSY. To reduce the impact of merge conflicts during the COSY job, COSY should be instructed to suppress further execution when conflicts occur.

6. The output tape constructed from the COSY job is used as input to COMPASS for assembly of binary decks.

7. The assembled binary decks are used to create the PRELIB source tape (section 4.7.3).

The following is a graphic representation of these installation procedures:


ENHANCED TAPE RECOVERY RELEASE TAPE
ASSEMBLY OPITIONS AND COSY CONTROL CARD DELETIONS
COSY* JOB
INTERMEDIATE MERGE TAPE
COSY JOB
COSY TAPE NO.
COSY TAPE NO. 2
HOLLERITH INPUT TAPE
COMPASS ASSEMBLY
BINARY DECK
PLIBEDIT PROCESSING
PRELIB SOURCE TAPE
PRELIB EXECUTION
MASS STORAGE

Example:

```
[SEQUENCE,100
[JOB,,,,
[CTO,THIS JOB WORKS WITH V4.2 RELEASE LEVEL CODE
[EQUIP,10=MT
[CTO,LUN 10 IS ITR RELEASE TAPE
[UTILITY,60,61.
SKFF,10.
SCOPE.
[EQUIP,30=MT
[CTO,LUN 30 IS A SCRATCH TAPE
[FET,MSOS,AUXFILE,500,XX
[OPEN,17
[AUX,17
[COSY,*,S
         DELETE*   1
         DELETE/   3,4
LERP     EQU       1
USASI    EQU       1
         DELETE*   1180
LOADER   DECK/     I=1,H=5
         DELETE*   1181,1182
LOADER   DECK*     I=10,H=30
         DELETE*   1
         DELETE/   3,4
LERP     EQU       1
USASI    EQU       1
         DELETE*   11
         DELETE*   1171
POSTLOAD DECK/     I=1,H=5
         DELETE*   1172,1173
         ENDCOSY/
POSTLOAD DECK*     I=10,H=30
ENDCOSY/ DECK*     I=10,H=30
         END*
[[

[SEQUENCE,110
[JOB,,,,
[EQUIP,30=MT
[CTO,LUN 30 IS OUTPUT TAPE FROM PREVIOUS JOB
[EQUIP,5=MT
[CTO,LUN 5 IS A SCRATCH TAPE FOR COMPASS INPUT
[EQUIP,1=MT
[CTO,LUN 1 IS MSOS V4.2 COSY 1
[COSY,I=30
         ENDCOSY/
[[

[SEQUENCE,120
[JOB,,,,
[EQUIP,5=MT
[COMPASS,I=5,L,R,P
[[
```

4.7.4 CONSTRUCTION OF THE PRELIB SOURCE

File one of the enhanced magnetic tape error recovery release tape has been constructed to allow routines that were not modified to be retrieved via UNIT/ control cards (section III-9.2.9) during execution of PLIBEDIT.

For additional information concerning construction of the PRELIB source, refer to section III-10.

Examples:

In the first example, MTRTNS was reassembled with the WRITEB assembly option specified as non-zero. At PLIBEDIT time, MTRTNS resides on file 56 due to X being specified on the COMPASS card.

```
[SEQUENCE,130
[JOB,,,,
[CTO,GENERATE MTRTNS WITH WRITEB ≠ 0
[EQUIP,37=MT
[CTO,LUN 37 IS THE ITR RELEASE TAPE
[UTILITY,60,61.
SKFF,37,29.
SCOPE.
[EQUIP,02=MT
[COSY
         DELETE/   19
WRITEB   EQU       1
MTRTNS   DECK/     I=37,H=2
         ENDCOSY/
[COMPASS,I=2,L,R,X
[[
[[

[SEQUENCE,100
[JOB,,,,
[EQUIP,10=MT
[CTO,,LUN 10 IS ITR/MSOS RELEASE TAPE
[PLIBEDIT
         REPLACE/  PRELIB PARAMETERS.
[PRELIB,,,,,ER
         POSITION/ BATCH
```

```
BCIO      REPLACE/
BCIO      UNIT/      10
          REPLACE/   UNIT,AA
            * * *   (BMSIO BINARY)   * * *
DRIV607   DELETE/    DRIV3528
            * * *   (DRIV607 BINARY)   * * *
DRIV3248 DELETE/
DRIV3555 DELETE/
DRIV3245 DELETE/
DRIV3293 DELETE/    MSI03232
MSI03553 DELETE/    DRIV3691
          REPLACE/   UNIT,BB
            * * *   (BCICREC1 BINARY)   * * *
RDUMP     REPLACE/
RDUMP     UNIT/      10
PRELOAD   REPLACE/
PRELOAD   UNIT/      10
LOADER    REPLACE/
            * * *   (LOADER BINARY)   * * *
          INSERT/    SEPOINT,BKEXIT
[SEPOINT,BK.SP
OVPRO     REPLACE/
            * * *   (OVPRO BINARY)   * * *
POSTLOAD REPLACE/
            * * *   (POSTLOAD BINARY)   * * *
          POSITION/ COMMON ABS
MSIOCCP   REPLACE/
MSIOCCP   UNIT/      10
CRT       LOCATE/
          SKIP/      2
ALGO      DELETE/    ALG5
MSIOSTER REPLACE/
MSIOSTER UNIT/       10
MTWPR     DELETE/    MTLDACPR
MTDER     REPLACE/
MTDER     UNIT/      10
CRDER     REPLACE/
CRDER     UNIT/      10
CPDER     REPLACE/
CPDER     UNIT/      10
MTWPRR    DELETE/    NWR
MTWPRR    UNIT/      10
MTRPRR    UNIT/      10
MTLDCPRR UNIT/       10
MTRTNS    FILE/      56
CMNRTNS   UNIT/      10
NRD       UNIT/      10
NRC       UNIT/      10
NWR       UNIT/      10
T.NOTRDY DELETE/    TYPEOUT.
T.NOTRDY UNIT/       10
TYPEOUT. UNIT/       10
          DELETE/    BCD
          POSITION/ BDP
MSIOMAIN REPLACE/
MSIOMAIN UNIT/       10
          DELETE/    COMMON REL
```

```
RAAR      DELETE/   SCAR
RAAR      UNIT/     10
SCAR      UNIT/     10
MTRPR     REPLACE/
MTRPR     UNIT/     10
MTDER     REPLACE/
MTDER     UNIT/     10
CRDER     DELETE/   NWR
CRDER     UNIT/     10
CPDER     UNIT/     10
MTWPRR    UNIT/     10
MTRPRR    UNIT/     10
MTLDCPRR  UNIT/     10
MTRTNS    FILE/     56
CMNRTNS   UNIT/     10
WHATISIT  UNIT/     10
NRD       UNIT/     10
NRC       UNIT/     10
NWR       UNIT/     10
T.NOTRDY  DELETE/   TYPEOUT.
T.NOTRDY  UNIT/     10
TYPEOUT.  UNIT/     10
ALGOL     DELETE/   ALGOLRUN
          ENDEDIT/
[[END-OF-FILE
```

The second example involves 65X PLIBEDIT using the input tape constructed in the preceeding PLIBEDIT example:

```
[SEQUENCE,200
[JOB,,,,
[EQUIP,10=MT
[CTO,,LUN 10 IS ITR/MSOS RELEASE TAPE
[PLIBEDIT
          REPLACE/  PRELIB,,,,,ER
[PRELIB,,,,ER,MM
DRIV607   REPLACE/
DRIV3528  UNIT/     10
BCICREC1  REPLACE/
            * * *   (BCICREC1 BINARY)   * * *
          INSERT/   RECORD
*** EXS BK.SP=ABNORMAL
MTRTNS    DELETE/
MTRTNS    UNIT/     10
MTRTNS    DELETE/
MTRTNS    UNIT/     10
          ENDEDIT/
[[END-OF-FILE
```

4.7.5 SEQUENCE NUMBERS TABLE

	COSY* CONTROL CARD SEQUENCE NUMBERS	DECK*	ENDCOSY*	END*	WRITEB	READB
CIO	1	27	28	29	--	--
BCIO	1	27	28	29	--	--
DRIV607	1 30	117	118	119	3	--
DRIV606	1 30	117	118	119	3	--
DRIV3528	1	40	41	42	7	--
RDUMP	1	1176	1177	1178	8	4
PRELOAD	1 10	1175	1176	1177	6	--
LOADER	1	1180	1181	1182	8	4
POSTLOAD	1 11	1171	1172	1173	6	--
MSIOCCP	1	1175	1176	1177	7	--
RAAR	1	9	10	11	--	--
SCAR	1	141	142	143	--	--
MTRPR	1	7	8	9	--	--
MTDER	1	7	8	9	--	--
CRDER	1	9	10	11	--	--
CPDER	1	11	12	13	--	--
MTWPRR	1	273	274	278	--	--
MTRPRR	1	178	179	180	--	--
MTLDCPRR	1	64	65	66	--	--
CMNRTNS	1	33	34	35	--	--
WHATISIT	1	78	79	80	--	--
NRD	1	168	169	170	--	9
NRC	1	40	41	42	--	--
NWR	1	26	27	28	--	--
T.NOTRDY	1	12	13	14	--	--
TYPEOUT.	1	33	34	35	--	--
MSIOMAIN	1 10	158	159	160	--	--
MSIOSTER	1	30	3p	32	--	--
MTRTNS	--	--	--	--	18	15

INCORPORATING PRODUCT SET MEMBERS 5

Some members of the MSOS product set are incorporated using separate procedures for one of the following reasons.

1. The products involved are special applications programs or use special applications equipment that some users may or may not have or need.

2. The procedures depend on the existence of a fully operable MSOS final library.

The procedures that follow involve installation and generation of:

SAINT

RESPOND

ADAPT

PERT TIME

PERT COST

USASI FORTRAN

USASI COBOL

5.1 SAINT

SAINT is released on MSOS COSY tape 1. To add SAINT to a library using MSOS 4.2, the following sample procedures should be followed.

1. Mount MSOS pack on unit 0 and scratch pack on unit 1

2. Mount MSOS COSY tape 1 on logical unit 1, scratch tape on logical unit 2

3. Create and run a card deck of the following form:

   Job 1 is an assembly in which several options (16K simulator with multiply and divide) are selected from available SAINT options. COMPASS assembles Hollerith tape 2. Job 1 produces a listing of SAINT and a binary deck containing the specified options.

$^{7}_{9}$SEQUENCE, 001

$^{7}_{9}$JOB, SAINT-COSY,

$^{7}_{9}$EQUIP, 01=MT, 02=MT

$^{7}_{9}$COSY

	DELETE/	39,41	
DIVIDE	EQU	TRUE	
MEMSIZE	EQU	16000	Option Examples
MULTIPLY	EQU	TRUE	
SAINT	DECK/	I=01, H=02	
	ENDCOSY/		

$^{7}_{9}$COMPASS, I=02, L, R, P

$^{77}_{88}$end-of-file

4. Create and run a PRELIB job of the following form:

   Job 2 inserts a binary image of the SAINT binary deck generated in Job 1 at the beginning of file 2 on the system scratch pack.

$^{7}_{9}$SEQUENCE, 002

$^{7}_{9}$JOB, SAINT-LIB,

$^{7}_{9}$PRELIB, , , , IE, OE †

$^{7}_{9}$FILE

$^{7}_{9}$INSERT

binary SAINT deck from job 1

$^{7}_{9}$FILE

$^{77}_{88}$end-of-file

5. Autoload the new edition.

---

† See PRELIB, Part III, Section 9.

5.1.1 ASSEMBLY OPTIONS

The following table provides the COSY sequence number, SAINT core requirements, and the name of the assembly parameters that must be equated to TRUE to select the various options in SAINT. For a complete description of assembly options, refer to Appendix A of the 3100/3200/3300/3500 SAINT Reference Manual, Pub. No. 60213700.

Sequence Number	Name	Core Required (decimal)
30	ADDRTABL	288
31	BRANBIT	20
32	COLBIN	306
33	COMPRESS	88
39	DIVIDE	123
40	MEMSIZE	4K - 2000 †
		8K - 4000†
		12K - 6000†
		16K - 8000†
41	MULTIPLY	90
43	SIM1403	-7
44	SIM1407	236
46	STERLING	1386
47	TRACEDMP	293
48	TRANSLAT	101
49	WMIO	195
50	WMBL	151

For WMBL, WMIO must be equated to TRUE.

If WMIO is TRUE and WMBL is nonzero, tape word mark I/O is performed in a non-time-dependent manner. This can be equated to the minimum number of characters desired in the input/output buffer. When using word mark I/O, if WMBL is equated to the maximum number of input or output characters, no density (tape) restrictions apply. However, if WMBL is equated to FALSE, the following restrictions apply.

3200/3300	601, 603, 604 tape transports, no density restrictions. 606, 607 tape transports, restricted to 200 bpi.
3100	Restricted to 200 bpi for all tape units.

† The simulation of SAINT memory requires half the nonsimulated core size.

5.1.2 CORE REQUIREMENTS

Use the following equation to find total core size of SAINT.

$$st = 2510 + \frac{memsize}{2} + so + \frac{wmbl}{4}$$

st	Total core requirement for SAINT
so	Total of option core requirements
wmbl	Size of word mark buffer length (variable)
memsize	Size of simulated memory

5.2 RESPOND

When a fully operational MSOS final library with memory protection (toggle switches), standard units on mass storage, and a dedicated (nonsystem) channel is generated, users with RESPOND equipment can obtain the RESPOND product. Before installation is begun, the decks must be assembled.

5.2.1 RESPOND DECK MODIFICATIONS

The number of stations active at any given time depends upon the installation. This value is not necessarily the total number of local and remote stations available to RESPOND, but only the greatest number that can be active (logged in) at any one time. In order to preserve a reasonable response time, do not exceed 10 to 12 active stations.

As released, RESPOND is set to accept a maximum of five active stations. If this is acceptable for the installation, proceed to definition of mass storage devices. If more than five active stations are necessary, perform the following modification.

```
              DELETE/    61,65
ASSIGNTB      OCT        0          STATION 1
              OCT        0          STATION 2
              .
              .
              OCT        0          STATION k
```

The parameter k represents the greatest number of active stations.

ASSIGNTB defines active stations to RESPOND. In association with it, ASATABLE is used to record values and parameters unique to servicing the active station. It exists in a one-to-one relation with ASSIGNTB so that the first entry of ASSIGNTB is assigned the first 48-word entry in ASATABLE; therefore, the length in entries of each table must be identical.

If the length of ASSIGNTB was altered, the ASATABLE must be modified accordingly.

```
            DELETE/     71,105
ASATABLE    BCD         12,                         )
            OCT         0,0,0,0,0,0,0,0             |
            BCD         2,                          |
            OCT         0,0                         |
            BCD         10,                         |  First Entry
            OCT         0,0                         |
            OCT         1†                          |
            OCT         53                          |
            OCT         0,0,0,0,0,0,0,0,0,0         )
            BCD         12,                         )
            OCT         0,0,0,0,0,0,0,0             |
            BCD         2,                          |
            OCT         0,0                         |
            BCD         10,                         |  Second Entry
            OCT         0,0                         |
            OCT         2†                          |
            OCT         53                          |
            OCT         0,0,0,0,0,0,0,0,0,0         )
            .
            .
            .
            BCD         12,                         )
            OCT         0,0,0,0,0,0,0,0             |
            BCD         10,                         |
            OCT         0,0                         |  Last Entry
            OCT         k†                          |
            OCT         53                          |
            OCT         0,0,0,0,0,0,0,0,0,0         )
```

5.2.2 DEFINE MASS STORAGE DEVICES

RESPOND requires internal definition of mass storage devices to allocate file and to allow the installation the opportunity to dedicate a device type to RESPOND. Only the device defined in RESPOND will be considered for allocating files.

```
            DELETE/     131,133
DEV.TBL     VFD         A9/spt,A9/wps,O6/dt
            .
            .
            .
```

spt   Sectors per track of device

wps   Words per sector of device

dt   Device type as defined by MSIO

---

† The parameter k is an octal integer set to 1 in the first entry and incremented by 1 for each succeeding ASATABLE entry. In no case is the value of k to be duplicated in ASATABLE.

Device	spt	wps	dt
852 Disk	20	25	41
853 Disk	16	64	50
854 Disk	16	64	51
841 Disk	16	160	52
813 Disk	32	64	60
814 Disk	32	64	61
863 Drum	16	64	70

5.2.3 ESTABLISH ACCOUNT IDENTIFIERS

RESPOND possesses an internal user-formed table (ACCOUNT) that contains a list of legal account identifiers. Only those users who specify an account identification that exists in the ACCOUNT table will be allowed access to RESPOND.

Each entry in the ACCOUNT table consists of an 8-character, left-justified (2 word) alphanumeric identifier followed by a word of zeroes.

```
          BCD         2, identifier
          OCT         0
```

Establishment of the ACCOUNT table requires the following correction. This correction deletes account numbers 4JL55 and 4EL24.

```
          DELETE/     139,142
ACCOUNT   BCD         2,identifier1   }  Entry 1
          OCT         0               }

          BCD         2,identifier2   }  Entry 2
          OCT         0               }
           .
           .
           .
```

5.2.4 ESTABLISH CORRECT HARDWARE ENVIRONMENT

```
          DELETE/     375,378
DDCHAN    EQU         c                Display data channel (must be dedicated)
DDPOLLRS  EQU         p                Number of pollers
DDEQUIP   EQU         e                Equipment number
DDMODEL   EQU         m                Controller type
```

c   Number of the data channel connected to the CONTROL DATA 3290 Inquiry/ Retrieval Controller

p   Number of pollers
- 0 No pollers
- 1 One poller (the poller must be hardware unit 14)
- 2 Two pollers (they must be units 13 and 14)

e   Equipment number of the 3290 controller in bits 9-11 (i.e., if the equipment number is 4, then e = 4000 octal)

m   Controller type
- 0 3290C Controller
- 1 3290D Controller

5.2.5 SELECT CORRECT DISPLAY FORMAT

Two display format options are available (13 lines of 80 characters per line or 20 lines of 50 characters per line). RESPOND requires one of the two formats and does not allow mixed formats.

RESPOND is set up to process the 20x50 format. If the 13x80 format is used, make the following change.

```
            DELETE/     379,381
DDLINES     EQU         13
DDCHARS     EQU         80
DDSCREEN    EQU         260
```

The deck card for these changes is:

```
RESPOND     DECK/       p1,p2,...
```

5.2.6 ROVER DECK MODIFICATIONS

To properly prepare input for the assembly phase of RESPOND installation procedures, make the following modifications to decks named ROVER (providing the 13x80 display format is selected).

Modify ROVER1 as follows:

```
            DELETE/     90,93
DDLINES     EQU         13
DDCHARS     EQU         80
DDSCREEN    EQU         260
DDCHPSCN    EQU         1040
```

Insert a COSY deck card regardless of whether the preceding change was made.

```
ROVER1      DECK/       p1,p2,...
```

Modify ROVER2 as follows:

	DELETE/	84, 87
DDLINES	EQU	13
DDCHARS	EQU	80
DDSCREEN	EQU	260
DDCHPSCN	EQU	1040

Insert a COSY deck card regardless of whether the preceding change was made.

ROVER2	DECK/	$p_1, p_2, \ldots$

Unmodified Routines

Assemble and insert decks for ROVER3 and FGCOPY.

ROVER3	DECK/	$p_1, p_2, \ldots$
FGCOPY	DECK/	$p_1, p_2, \ldots$

5.2.7 RESPOND INSTALLATION PROCEDURES

Assemble the COSY correction decks established during RESPOND environment preparation using the following procedures.

1. Mount MSOS COSY tape 1 on logical unit 01.

2. Mount a scratch tape on logical unit 02 to receive the Hollerith output.

3. Add MSOS control cards to the COSY correction decks to create an input deck of the following form, and assemble.

   ${}^{7}_{9}$SEQUENCE, 001

   ${}^{7}_{9}$JOB, , , ,

   ${}^{7}_{9}$EQUIP, 1=MT, 2=MT

   ${}^{7}_{9}$COSY

   RESPOND corrections

RESPOND	DECK/	I=01, H=02

   ROVER1 corrections (if any)

ROVER1	DECK/	I=01, H=02

   ROVER2 corrections (if any)

```
ROVER2          DECK/       I=01, H=02

ROVER3          DECK/       I=01, H=02
FGCOPY          DECK/       I=01, H=02

                ENDCOSY/
```

$^7_9$COMPASS, I=02, L, P, R

$^{77}_{88}$end-of-file

Since RESPOND is an overlay program, the RESPOND user must now add overlay control cards to the assembled RESPOND binary decks to create an overlay deck of the following form. The column of numbers represents row punches in column 1 of each overlay control card. Other numbers represent punches starting in column 2.

```
12
 0 49                                    Overlay control card, main element
 7
 9
```

binary deck of RESPOND (IDC to TRA)

```
12
 0
 3 49,1                                  Overlay control card, overlay element
 7
 9
```

binary deck of ROVER1 (IDC to TRA)

```
12
 0
 3 49,2                                  Overlay control card, overlay element
 7
 9
```

binary deck of ROVER2 (IDC to TRA)

```
12
 0
 3 49,3                                  Overlay control card, overlay element
 7
 9
```

binary deck of ROVER3 (IDC to TRA)

binary deck of FGCOPY (IDC to TRA)

The final step in installing RESPOND is a special PRELIB run that inserts RESPOND into a final MSOS library ready for use. A sample deck structure for the PRELIB run is:

$^{7}_{9}$SEQUENCE, 001

$^{7}_{9}$JOB, , , ,

$^{7}_{9}$PRELIB, , , , , OE

$^{7}_{9}$FILE

$^{7}_{9}$INSERT

RESPOND/MSOS overlay deck

$^{7}_{9}$FILE

$^{77}_{88}$end-of-file

5.3 ADAPT

The ADAPT Version 1.1 release consists of two packages, one for 16K and one for 32K.

Each package consists of:

A binary (load-and-go) tape from which to make ADAPT overlay tape

A set of operating procedures

Also available by request are:

A BCD tape of the ADAPT source decks

A COSY tape containing FORTRAN subroutines for the 16K ADAPT variant

ADAPT Internal Maintenance Specifications

ADAPT operates from an overlay tape made from the binary tape furnished as a part of the ADAPT system release, or from decks punched from the tape.

5.3.1 16K ADAPT

To create the 16K overlay structure from ADAPT 16K source tape:

1. Mount source tape on logical unit 01 and scratch tapes on logical units 02 and 03.

2. Compile and assemble ADAPT routines using the following cards.

${}^{7}_{9}$SEQUENCE,001

${}^{7}_{9}$JOB,,,,NP,ND

${}^{7}_{9}$EQUIP,01=MT, 02=MT, 03=MT

${}^{7}_{9}$FORTRAN, I=01,P=02,L=03

${}^{7}_{9}$COMPASS, I=01,P=02,L=03

${}^{7}_{9}$FORTRAN, I=01,P=02,L=03

${}^{7}_{9}$COMPASS, I=01,P=02,L=03

${}^{7}_{9}$FORTRAN, I=01,P=02,L=03

${}^{77}_{88}$end-of-file

3. Punch the object decks from logical unit 02 and list the source from logical unit 03.

4. Reproduce the following decks.

	Deck	Sequence number	Copies
	STDPACK	0110	1
	STDUNPK	0111	1
	PTRIT	0125	1
GROUP A	GENPLN	0223	1
GROUP A	ZVECT	0224	1
GROUP A	ZVALUE	0225	1
GROUP A	DOT	0227	1
GROUP A	CROSS	0226	2
GROUP A	LENGTH	0228	1
GROUP A	NORM	0229	2

Place routines STDPACK and STDUNPK in main parts of overlay 1 and overlay 2.

Place PTRIT in segments 2 and 3 of overlay 1. Place copies of the GROUP A decks in segments 2 and 3 of overlay 2.

Place copies of CROSS and NORM in overlay 4.

5. Mount the second 16K source tape on logical unit 01 and assemble the FORTRAN object-time routines using the following cards.

$^{7}_{9}$SEQUENCE,002

$^{7}_{9}$JOB,,,,NP,ND

$^{7}_{9}$EQUIP,01=MT,02=MT,03=MT,04=MT

$^{7}_{9}$COSY

BCDINP	DECK/	I=01,H=02
BCDOUT	DECK/	I=01,H=02
FORMAT	DECK/	I=01,H=02
BINARY	DECK/	I=01,H=02
IOCHK	DECK/	I=01,H=02
EOFCHK	DECK/	I=01,H=02
TAPEHAND	DECK/	I=01,H=02
PAUSE	DECK/	I=01,H=02
CONTROL	DECK/	I=01,H=02
Q8QOUTTB	DECK/	I=01,H=02
Q1QADRI	DECK/	I=01,H=02
XTOI	DECK/	I=01,H=02
POWRF	DECK/	I=01,H=02
SINCOS	DECK/	I=01,H=02
ATANF	DECK/	I=01,H=02
EXPF	DECK/	I=01,H=02
LOGF	DECK/	I=01,H=02
SIGNF	DECK/	I=01,H=02
SQRTF	DECK/	I=01,H=02
ABSF	DECK/	I=01,H=02
FLOATF	DECK/	I=01,H=02
FIXF	DECK/	I=01,H=02
MASKINGF	DECK/	I=01,H=02
FAULTS	DECK/	I=01,H=02
SENSLITE	DECK/	I=01,H=02

Q8QERROR	DECK/	I=01, H=02
CIO. MSIO	DECK/	I=01, H=02
	ENDCOSY/	

$^{7}_{9}$COMPASS, I=02, L=03, P=04

$^{77}_{88}$end-of-file

6. List the source from logical unit 02 and punch binary decks from logical unit 03.

7. Reproduce the following binary decks.

Deck	Copies
BCDINP	1
BCDOUT	1
FORMAT	3
Q1QADRI	4
XTOI	4
POWRF	4
SINCOS	5
ATANF	3
SIGNF	1
SQRTF	4
ABSF	6
FLOATF	5
FIXF	4
MASKINGF	2

8. Set up the following jobs:

Job 1 enters an overlay pack and allocates files:

$^{7}_{9}$SEQUENCE, 001

$^{7}_{9}$JOB, , , , NP, ND

$^{7}_{9}$EQUIP, 01=DP

$^{7}_{9}$MSUTIL, 60

ENTER, 01, 853/103, S, , , 001, 999

SCOPE

${}^{7}_{9}$RAT,853/103

${}^{7}_{9}$FET,ADAPT,OVLY49,200

${}^{7}_{9}$ALLOCATE,249,991231

${}^{7}_{9}$FET,ADAPT,ADAPT5,512

${}^{7}_{9}$ALLOCATE,350,991231

${}^{7}_{9}$FET,ADAPT,ADAPT7,512

${}^{7}_{9}$ALLOCATE,350,991231

${}^{77}_{88}$end-of-file

Job 2 creates an overlay file:

${}^{7}_{9}$SEQUENCE,002

${}^{7}_{9}$JOB,,,,NP,ND

${}^{7}_{9}$ FET,ADAPT,OVLY49,200

${}^{7}_{9}$OPEN,49

${}^{7}_{9}$ FET,ADAPT,ADAPT5,512

${}^{7}_{9}$ OPEN,05

${}^{7}_{9}$ FET,ADAPT,ADAPT7,512

${}^{7}_{9}$ OPEN,07

$\begin{matrix}12\\0\\7\\9\end{matrix}$ 49

binary decks for MAIN

CDCADAPT
INITIAL
PRCNTL
LBSRCH
TAPERD
TAPEWT
FLOVER
CIO.MSIO
IOCHK
EOFCHK
TAPEHAND
PAUSE
CONTROL
Q8QOUTTS
BINARY
Q8QERROR

12
0
3 49,1
7
9

binary decks for overlay 1

PASS1A
ITYPE
DOMAC
PTIDENT
ERRMSG
BCDFETCH
JUGGLE
NUMSTOR
STDPACK
STDUNPK
Q1QADRI
SIGNF
ABSF
FIXF
MASKINGF

12
0
2 49,1
7
9

binary decks for overlay 1, segment 1

P1ASEG1
CARDBKUP
PTGEN
SERCHV
TABLCK
BCDINP
FORMAT

12
0
2 49,2
7
9

binary decks for overlay 1, segment 2

P1ASEG2
POSMACH
MACDEF
CAWL
TERMAC
RESERV
PTRIT

12
0
2 49,3
7
9

binary decks for overlay 1, segment 3

P1ASEG3
PTRIT
GEOM1A
FINI
BCDINP
FORMAT

12
0
2 49,4
7
9

binary decks for overlay 1, segment 4

P1ASEG4
TABLSET
XABLE
YABLE

12
0
2 49,5
7
9

binary decks for overlay 1, segment 5

P1ASEG5
COMPUT
IFJUMP
POWRF
XTOI
ATANF
LOGF
EXPF
SINCOS
SQRTF
FLOATF

12
0
3 49,2
7
9

binary decks for overlay 2

PASS1B
PTID
TABLS1B
ERRMSG
STDPACK
STDUNPK
CANPUT
CANGET
DEFPRE
MASKINGF
Q1QADRI
FIXF

12
0
2 49,1
7
9

binary decks for overlay 2, segment 1

SEND
BCDINF
FINIB
MISC
PSIS
POKE
RITAPE
MOTION
PREPRO
TABCO
QUAD
RFCTF
SSQRF
PPARAM
ATANF
XTOI
POWRF
SINCOS
ABSF
FLOATF

12
0
2 49,2
7
9

binary decks for overlay 2, segment 2

JDSEG2
GENPLN
ZVECT
ZVALUE
CROSS
DOT
LENGTH
NORM
LINE01
LINE02
LINE03
LINE04
LINE05
LINE06
LINE07
LINE08
LINE09

PLAN01
PLAN02
PLAN03
ELLP01
ELLHYP
GCON01
MATX01
MATX02
MATX03
MATX04
SINCOS
XTOI
POWRF
SQRTF
ABSF
FLOATF

12
0
2 49,3
7
9

binary decks for overlay 2, segment 3

JDSEG3
GENPLN
ZVECT
ZVALUE
CROSS
DOT
LENGTH
NORM
PONT01
PONT02
PONT03
PONT04
PONT05
PONT07
CIRL01
CIRL02
CIRL03
CIRL05
CIRL08
CIRL09
CIRL10
CIRL11
CIRL12

XTOI
POWRF
SQRTF
ABSF
FLOATF

12
0
3 49,3
7
9

binary decks for overlay 3

CALLSEC2
SUPER
AMINDX
DDSTX
UNRMALX
AERRX
ASTOSX
CENTRX
RADARX
CPLANX
CCURVX
DELTAX
ARLMG
TLNORMX
CHECKX
QUADX
ATAPEX
VNOMRX
STRTUPX
AJUNDDX
AREPREX
FAULTS
FLOATF
Q1QADRI
SQRTF
SIGNF
ABSF
FIXF

12
0
2 49,1
7
9

binary decks for overlay 3, segment 1

CALLSEG1
SECTN2X

12
0
2 49,2
7
9

binary decks for overlay 3, segment 2

ARLMCL
SINCOS
ATANF

12
0
2 49,3
7
9

binary decks for overlay 3, segment 3

CALLSEG3
UNTABC
DDTABC
SINCOS
ATANF

12
0
3 49,4
7
9

binary decks for overlay 4

PICKPOCK
POCKET
ATAPEY
ASTOSY
AERRY
CROSS
NORM
XTOI
POWRF
Q1QADRI
ABSF
SQRTF
FIXF
FLOATF
SENSLITE

12
0
3 49,5
7
9

binary decks for overlay 5

PASS3
CLPRNT
SEARCH
IFIXED
MULTM
TSFMPT
TSFMVC
MATMOV
TABFCT
MASKINGF
BCDOUT
FORMAT
ABSF
Q1QADRI
SINCOS
FIXF

12
0
3 49,6
7
9

binary decks for overlay 6

SECTN4
DISPAT
BCDOUT
FORMAT

(Post processors, if any, must be inserted here. Each post processor constitutes a separate segment of overlay 6.)

${}^{7}_{9}$RUN

test part programs

${}^{77}_{88}$end-of-file

9. Once the overlay file is created, part programs are run as follows.

${}^{7}_{9}$SEQUENCE,001

${}^{7}_{9}$JOB,,,,NP,ND

${}^{7}_{9}$FET,ADAPT,OVLY49,200

${}^{7}_{9}$OPEN,49

${}^{7}_{9}$FET,ADAPT,ADAPT5,512

${}^{7}_{9}$OPEN,05

${}^{7}_{9}$FET,ADAPT,ADAPT7,512

${}^{7}_{9}$OPEN,07

binary decks for MAIN

${}^{7}_{9}$RUN

part programs

${}^{77}_{88}$end-of-file

5.3.2 INSTALL 16K ADAPT USING THE BINARY RELEASE TAPE

1. Enter the overlay pack and allocate files in the manner described for the source tape.

2. Reautoload the system and assign standard input to the tape drive on which the ADAPT binary tape is mounted.

3. Run the jobs on tape.

4. To run ADAPT part programs after creating the overlay file, punch MAIN binary decks from the tape and proceed as described above for source tape.

5.3.3 32K ADAPT

Create 32K overlay structure from ADAPT source tape.

1. Mount source tape on logical unit 01 and scratch tapes on logical units 02 and 03.

2. Compile and assemble ADAPT routines using the following cards.

$^7_9$SEQUENCE, 001

$^7_9$JOB, , , ,

$^7_9$EQUIP, 01=MT, 02=MT, 03=MT

$^7_9$FORTRAN, I=01, P=02, L=03

$^7_9$COMPASS, I=01, P=02, L=03

$^7_9$FORTRAN, I=01, P=02, L=03

$^7_9$COMPASS, I=01, P=02, L=03

$^7_9$FORTRAN, I=01, P=02, L=03
$^{77}_{88}$end-of-file

3. Punch the object decks from logical unit 02 and list the source from logical unit 03.

4. Duplicate the following decks:

STDPACK (used in overlays 1 and 2)
STDUNPK (used in overlays 1 and 2)
PTRIT (used in overlay 1, segments 2 and 3)
CROSS (used in overlays 2 and 4)
NORM (used in overlays 2 and 4)

5. Set up the following jobs.

Job 1 enters the overlay pack and allocates files.

${}^{7}_{9}$SEQUENCE, 001

${}^{7}_{9}$JOB, , , ,

${}^{7}_{9}$EQUIP, 01=DP

${}^{7}_{9}$MSUTIL, 60
ENTER , 01, 854/103, S, , , 0001, 1999
SCOPE

${}^{7}_{9}$RAT, 854/103

${}^{7}_{9}$FET, ADAPT, OVLY49, 200

${}^{7}_{9}$ALLOCATE, 249, 991231, S, , 854

${}^{7}_{9}$FET, ADAPT, ADAPT5, 512

${}^{7}_{9}$ALLOCATE, 375, 991231, S, , 854

${}^{7}_{9}$FET, ADAPT, ADAPT7, 512

${}^{7}_{9}$ALLOCATE, 375, 991231, S, , 854

${}^{77}_{88}$end-of-file

Job 2 creates the overlay file.

```
7
9SEQUENCE,002

7
9JOB,,,,

7
9FET,ADAPT,OVLY49,200

7
9OPEN,49

7
9FET,ADAPT,ADAPT5,512

7
9OPEN,05

7
9FET,ADAPT,ADAPT7,512

7
9OPEN,07

12
 0
 7 49
 9
```

binary decks for MAIN overlay

```
CDCADAPT
INITIAL
PRCNTL
LBSRCH
TAPERD
TAPEWT
FLOVER

12
 0
 3 49,1
 7
 9
```

binary decks for overlay 1

PASS1A
ITYPE
DOMAC
PTIDENT
ERRMSG
BCDFETCH
JUGGLE
TABLSFT
NUSTOR
STDPACK
STDUNPK
XABLE
YABLE

12
0
2 49,1
7
9

binary decks for overlay 1, segment 1

P1ASEG1
CARDBKUP
PTGEN
SERCHV
TABLCK

12
0
2 49,2
7
9

binary decks for overlay 1, segment 2

P1ASEG2
POSMACH
MACDEF
CAWL
TERMAC
RESERV
PTRIT

12
0
2 49,3
7
9

binary decks for overlay 1, segment 3

P1ASEG3
COMPUT
IFJUMP
PTRIT
GEOM1A
FINI

12
0
3 49,2
7
9

binary decks for overlay 2

PASS1B
PTID
TABLS1B
ERRMSG
STDPACK
STDUNPK
CANPUT
CANGET

12
0
2 49,1
7
9

binary decks for overlay 2, segment 1

SEND
BCDINF
FINIB
MISC
PSIS
POKE
RITAPE
MOTION
PREPRO
TABCO
QUAD
RFCTF
SSQRF
PPARAM

12
0
2 49,2
7
9

binary decks for overlay 2, segment 2

DEFPRE
JDSPAT
GENPLN
ZVECT
ZVALUE
CROSS
DOT
LENGTH
NORM
PONT01
PONT02
PONT03
PONT04
PONT05
PONT07
LINE01
LINE02
LINE03
LINE04
LINE05
LINE06
LINE07
LINE08
LINE09
PLAN01
PLAN02
PLAN03
CIRL01
CIRL02
CIRL03
CIRL05
CIRL08
CIRL09
CIRL10
CIRL11
CIRL12
ELLP01
ELLHYP
GCON01
MATX01
MATX02

MATX03
MATX04

12
0
3 49,3
7
9

binary decks for overlay 3

CALLSEG2
SUPER
AMINDX
DDSTX
UNRMALX
AERRX
ASTOSX
CENTRX
RADARX
CPLANX
CCURVX
DELTAX
ARLMG
SNAP
TLNORMX
CHECKX
QUADX
ATAPEX
VNORMX
STRTUPX
AJUNDDX
AREPREX

12
0
2 49,1
7
9

binary decks for overlay 3, segment 1

CALLSEG1
SECTN2X

12
0
2 49,2
7
9

binary decks for overlay 3, segment 2

```
ARLMCL

12
 0
 2 49,3
 7
 9
```

binary decks for overlay 3, segment 3

```
CALLSEG3
UNTABC
DDTABC

12
 0
 3 49,4
 7
 9
```

binary decks for overlay 4

```
PICKPOCK
POCKET
ATAPEY
ASTOSY
AERRY
NORM
CROSS

12
 0
 3 49,5
 7
 9
```

binary decks for overlay 5

```
PASS3
CLPRNT
SEARCH
IFIXED
MULTM
TSFMPT
TSFMVC
MATMOV
TABFCT
```

12
0
3 49,6
7
9

binary decks for overlay 6

SECTN4
DISPAT

(Post processors, if any, must be inserted here; each post processor constitutes a separate segment of overlay 6.)

${}^{7}_{9}$RUN

test part programs

${}^{77}_{88}$end-of-file

6. Once the overlay file has been created, part programs may be run as follows.

${}^{7}_{9}$SEQUENCE, 001

${}^{7}_{9}$JOB, , , ,

${}^{7}_{9}$FET, ADAPT, OVLY49, 200

${}^{7}_{9}$OPEN, 49

${}^{7}_{9}$FET, ADAPT, ADAPT5, 512

${}^{7}_{9}$OPEN, 05

${}^{7}_{9}$FET, ADAPT, ADAPT7, 512

${}^{7}_{9}$OPEN, 07

binary decks for MAIN overlay

${}^{7}_{9}$RUN

part programs

${}^{77}_{88}$end-of-file

5.3.4 INSTALL ADAPT USING THE BINARY RELEASE TAPE

1. Enter the overlay pack and allocate files in the manner described for the source tape.

2. Reautoload the system and assign standard input to the tape drive on which the ADAPT binary tape is mounted.

3. Run the job on the tape.

4. To run ADAPT part programs after creating the overlay file, punch the MAIN overlay decks from the binary tape and proceed as described for the ADAPT source tape.

5.4 PERT TIME

To generate a PERT TIME system, use the following short-form procedure.

> Punch the first file (PERT TIME binary deck with control cards) from the release tape† and incorporate necessary MSOS control cards to create an overlay file. If a listing is desired, mount the release tape as standard input, skip to the second file (source cards for the PERT TIME variant with control cards for compiling and listing on standard output), and compile.

If re-creation of the overlay preparation deck is necessary, use the following procedure:

1. Punch the second file on cards.

2. Replace the FORTRAN and COMPASS control cards with cards that will produce punched and listed output.

3. Compile the program.

† If tape units are not available, the release consists of a binary card deck of the PERT TIME program with necessary overlay control cards. Use this deck with the appropriate MSOS control cards to create the overlay file.

5.4.1 ESTABLISH MEMORY AND MASS STORAGE REQUIREMENTS

PERT TIME operates in either 11200_{10}† or 21000_{10} words of core storage and requires a minimum of two 852 Disk Storage Drives, or their equivalent, in addition to those required by MSOS. PERT TIME files for the 21K variant using two 852 disk drives are:

File Name	Logical Unit	Mode	Block Size	File Size (tracks)
EVEN1	none	Sector	2000	1
SCRATCH1	01	Sector	4000	300
SCRATCH2	02	Sector	4000	300
SCRATCH3	03	Sector	4000	300
COSTLINK	07	Sector	4000	98
EVEN2	none	Sector	2000	1
OLDMASTER	06	Sector	4000	300
NEWMASTER	04	Sector	4000	300
OVERLAY	05	Sector	100	90

Maximum capacities are obtainable with 853 or 854 Disk Storage Drives or additional 852 drives. File sizes can then be increased to use the added disk storage capacity. The following illustrates added capacity with four 852 Disk Storage Drives.

† The 11200_{10} variant of PERT TIME allows a maximum of 1200 activities and 900 events. The 11200_{10} variant will run on a 16K system only if it is a 16K BATCH system with no options. This means the NP and ND options on the job card will also have to be used.

File Name	Logical Unit	Mode	Block Size	File Size (tracks)
EVEN1	none	Sector	2000	1
OVERLAY	05	Sector	100	90
SCRATCH1	01	Sector	4000	500
EVEN2	none	Sector	2000	1
SCRATCH2	02	Sector	4000	500
EVEN3	none	Sector	2000	1
SCRATCH3	03	Sector	4000	500
COSTLINK	07	Sector	4000	98
EVEN4	none	Sector	2000	1
OLDMASTER	06	Sector	4000	498
NEWMASTER	04	Sector	4000	498

The PERT TIME system operates most efficiently when the files are allocated with the mode and block size previously specified. File EVEN1 must be the first file allocated on the first 852 Disk Storage Drive. File EVEN2 must be the first file allocated on the second 852 Disk Storage Drive. When used, files EVEN3 and EVEN4 must be the first files allocated on the third and fourth 852 Disk Storage Drives. Although PERT TIME does not use these four files, they ensure that input/output requests do not have to cross cylinders on 852 Disk Storage Drives. These files are not necessary when 853 or 854 Disk Storage Drives are used. With 853 or 854 Disk Storage Drives the block size for file OVERLAY should be changed to 256.

5.4.2 ALLOCATE PERT TIME FILES

The control cards shown below illustrate the allocation of PERT TIME files to two previously labeled 852 Disk Storage Drives. Control cards must be in the order shown. See the MSOS Reference Manual for complete format and discussion of control cards.

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , , , NP, ND

$^{7}_{9}$RAT, 852/25

$^{7}_{9}$FET, PERT, EVEN1, 2000

$^{7}_{9}$ALLOCATE, 1, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, SCRATCH1, 4000

$^{7}_{9}$ALLOCATE, 300, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, SCRATCH2, 4000

$^{7}_{9}$ALLOCATE, 300, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, SCRATCH3, 4000

$^{7}_{9}$ALLOCATE, 300, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, COSTLINK, 4000

$^{7}_{9}$ALLOCATE, 98, , S, NOSEG, 852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , , , NP, ND

$^{7}_{9}$RAT, 852/26

$^{7}_{9}$FET, PERT, EVEN2, 2000

$^{7}_{9}$ALLOCATE, 1, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, NEWMASTER, 4000

$^{7}_{9}$ALLOCATE, 300, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, OLDMASTER, 4000

$^{7}_{9}$ALLOCATE, 300, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, OVERLAY, 100

$^{7}_{9}$ALLOCATE, 90, , S, NOSEG, 852

$^{77}_{88}$end-of-file

The following control cards illustrate the allocation of PERT TIME files to four previously labeled 852 Disk Storage Drives. This allocation allows the maximum capacities of 8000 activities and 4095 events to be processed with the 21K program variant.

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , , ,NP,ND

$^{7}_{9}$RAT,852/25

$_{9}$FET,PERT,EVEN1,2000

$^{7}_{9}$ALLOCATE,1, ,S,NOSEG,852

$^{7}_{9}$FET,PERT,OVERLAY,100

$^{7}_{9}$ALLOCATE,90, ,S,NOSEG,852

$^{7}_{9}$FET,PERT,SCRATCH1,4000

$^{7}_{9}$ALLOCATE,500, ,S,NOSEG,852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , , ,NP,ND

$^{7}_{9}$RAT,852/26

$^{7}_{9}$FET,PERT,EVEN2,2000

$^{7}_{9}$ALLOCATE,1, ,S,NOSEG,852

$^{7}_{9}$FET,PERT,SCRATCH2,4000

$^{7}_{9}$ALLOCATE,500, ,S,NOSEG,852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , , ,NP,ND

$^{7}_{9}$RAT,852/27

$^{7}_{9}$FET,PERT,EVEN3,2000

$^{7}_{9}$ALLOCATE,1, ,S,NOSEG,852

$^{7}_{9}$FET,PERT,SCRATCH3,4000

$^{7}_{9}$ALLOCATE,500, ,S,NOSEG,852

$^{7}_{9}$FET,PERT,COSTLINK,4000

${}^{7}_{9}$ALLOCATE, 98, , S, NOSEG, 852

${}^{77}_{88}$end-of-file

${}^{7}_{9}$SEQUENCE

${}^{7}_{9}$JOB, , , , NP, ND

${}^{7}_{9}$RAT, 852/28

${}^{7}_{9}$FET, PERT, EVEN4, 2000

${}^{7}_{9}$ALLOCATE, 1, , S, NOSEG, 852

${}^{7}_{9}$FET, PERT, NEWMASTER, 4000

${}^{7}_{9}$ALLOCATE, 498, , S, NOSEG, 852

${}^{7}_{9}$FET, PERT, OLDMASTER, 4000

${}^{7}_{9}$ALLOCATE, 498, , S, NOSEG, 852

${}^{77}_{88}$end-of-file

5.4.3 CREATE OVERLAY FILE

Create a PERT TIME overlay file by adding MSOS control cards to PERT TIME binary decks. The resultant program is as follows.

${}^{7}_{9}$SEQUENCE

${}^{7}_{9}$JOB, , , , NP, ND

${}^{7}_{9}$FET, PERT, SCRATCH1, 4000

${}^{7}_{9}$OPEN, 01

${}^{7}_{9}$FET, PERT, SCRATCH2, 4000

${}^{7}_{9}$OPEN, 02

${}^{7}_{9}$FET, PERT, SCRATCH3, 4000

${}^{7}_{9}$OPEN, 03

${}^{7}_{9}$FET, PERT, NEWMASTER, 4000

${}^{7}_{9}$OPEN, 04

${}^{7}_{9}$FET, PERT, OVERLAY, 100

${}^{7}_{9}$OPEN, 05

${}^{7}_{9}$FET, PERT, OLDMASTER, 4000

$^{7}_{9}$OPEN, 06

$^{7}_{9}$FET, PERT, COSTLINK, 4000

$^{7}_{9}$OPEN, 07

+
0
7 05
9

binary cards for main program

+
0
305, 1
7
9

binary cards for first overlay

+
0
305, 2
7
9

binary cards for second overlay

.
.
.

+
0
305, n
7
9

binary cards for nth overlay

$^{7}_{9}$RUN

K control card
L control card

calendar modification cards, if any

M control card
X control card
Y control card
W control card

activity and/or event name cards sorted by predecessor and successor event number

```
Z  control card
K
.
.
.
Z
STOP
7
9CLOSE,04
7
9CLOSE,05
7
9CLOSE,06
7
9CLOSE,07
77
88end-of-file
7
9SEQUENCE
7
9JOB,,,,NP,ND
7
9FET,PERT,EVEN1,2000
7
9RELEASE,ALL
7
9FET,PERT,SCRATCH1,4000
7
9RELEASE,ALL
7
9FET,PERT,SCRATCH2,4000
7
9RELEASE,ALL
7
9FET,PERT,SCRATCH3,4000
7
9RELEASE,ALL
77
88end-of-file
```

If either master file COSTLINK or OVERLAY is not being saved for future executions, the CLOSE card with the appropriate file ordinal should be followed by FET and RELEASE cards to remove the label from the file label directory.

To run successive PERT TIME networks, cards K through Z are followed by a STOP control card after the last Z card. If an error is encountered in a network, a diagnostic is printed, the current network is abandoned, and the next network is processed.

5.4.4 EXECUTION RUN

To run the final PERT TIME system installed on an MSOS final library, use an established overlay file, control cards, binary cards, and PERT TIME input cards as follows.

${}^{7}_{9}$SEQUENCE

${}^{7}_{9}$JOB,,,,NP,ND

${}^{7}_{9}$FET,PERT,SCRATCH1,4000

${}^{7}_{9}$OPEN,01

${}^{7}_{9}$FET,PERT,SCRATCH2,4000

${}^{7}_{9}$OPEN,02

${}^{7}_{9}$FET,PERT,SCRATCH3,4000

${}^{7}_{9}$OPEN,03

${}^{7}_{9}$FET,PERT,NEWMASTER,4000

${}^{7}_{9}$OPEN,04

${}^{7}_{9}$FET,PERT,OVERLAY,100

${}^{7}_{9}$OPEN,05

${}^{7}_{9}$FET,PERT,OLDMASTER,4000

${}^{7}_{9}$OPEN,06

${}^{7}_{9}$FET,PERT,COSTLINK,4000

${}^{7}_{9}$OPEN,07

binary cards for main program

${}^{7}_{9}$RUN

K control card
L control card

calendar modification cards, if any

M control card
X control card
Y control card
W control card

activity and/or event name cards sorted by predecessor and successor event number

Z control card
K
.
.
Z
STOP

$^{7}_{9}$CLOSE,04

$^{7}_{9}$CLOSE,05

$^{7}_{9}$CLOSE,06

$^{7}_{9}$CLOSE,07

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB,,,,NP,ND

$^{7}_{9}$FET,PERT,EVEN1,2000

$^{7}_{9}$RELEASE,ALL

$^{7}_{9}$FET,PERT,SCRATCH1,4000

$^{7}_{9}$RELEASE,ALL

$^{7}_{9}$FET,PERT,SCRATCH2,4000

$^{7}_{9}$RELEASE,ALL

$^{7}_{9}$FET,PERT,SCRATCH3,4000

$^{7}_{9}$RELEASE,ALL

$^{77}_{88}$end-of-file

A disk pack must be entered with a properly allocated and established PERT TIME overlay file. A properly allocated and established PERT TIME master file must also be entered if a master file is to be updated.

No special provisions are necessary to assign the PERT TIME master files to magnetic tape except to equip units 4 and 6 to magnetic tape drives and to eliminate all MSIO control cards pertaining to the master files.

5.5 PERT COST

To generate a PERT COST system, use the following short-form procedure.

> Punch the first file (PERT COST binary deck with control cards) of the release tape† and incorporate necessary MSOS control cards to create an overlay file. If a listing is desired, mount the release tape as standard input, skip to the second file (source cards for the PERT COST variant with control cards for compiling and listing on standard output), and compile it.

If re-creation of the overlay preparation deck is necessary, use the following procedure.

1. Punch the second file on cards.
2. Replace the FORTRAN and COMPASS control cards with cards that will produce punched and listed output.
3. Compile the program.
4. Insert main and overlay cards in the designated positions, using the listed output from the previous step as a guide.
5. Add MSOS control cards and run the job to create the overlay file.

Detailed procedures for generating and installing a PERT COST system include assembly and the execution run.

5.5.1 ESTABLISH MEMORY AND MASS STORAGE REQUIREMENTS

The size of core memory determines the maximum summary numbers and charge numbers in the work breakdown structure. These requirements are in addition to those of the operating system and priority programming. The capacity of each variant depends on the availability of a sufficient quantity of disk storage drives to handle PERT COST master and scratch files.

Memory Used	Maximum Account Numbers	Maximum Summary Numbers	Total
8K	1500	1000	2500
20K	3000	2000	5000

† If tape units are not available, the release consists of a binary card deck of the PERT COST program with necessary overlay control cards. Use this deck with the appropriate MSOS control cards to create the overlay file.

In addition to disk storage required by MSOS, the PERT COST system may be run with a minimum of two 852 Disk Storage Drives or the equivalent. However, at least four 852 Disk Storage Drives or the equivalent are recommended.

The logical file allocation for four 852 Disk Storage Drives is:

File Name	Logical Unit	Mode	Block Size	File Size (tracks)
OLD SUMMARY NUMBERS	11	Sector	100	25
OLD ACCOUNT NUMBERS	12		200	100
OLD COSTS	13		200	194
OLD RATES	14		300	150
OLD F & G CATEGORY	15		512	20
OLD HIERARCHY TABLE	16		512	10
NEW SUMMARY NUMBERS	21		100	25
NEW ACCOUNT NUMBERS	22		200	100
NEW COSTS	23		200	194
NEW RATES	24		300	150
NEW F & G CATEGORY	25		512	20
NEW HIERARCHY TABLE	26		512	10
SCRATCH1	01		4000	758
SCRATCH2	02		4000	758
SCRATCH3	03		4000	758
EVEN1	None		2000	1
EVEN2	None		2000	1
EVEN3	None		2000	1
COVERLAY	05	Sector	100	240

These file sizes allow processing of 500 summary numbers and 1000 account numbers. The master cost files provide for a maximum of 1940 records. Each six months of cost (budgets, estimates, and actuals) entered for an account number, performing organization, or resource code combination generates one master cost record. The PERT COST basic work file (scratch file) cannot exceed 6820 records. A record for the work file is generated for each summary number and account number, along with each month of costs associated with each account number, performing organization, and resource code combination.

The above file allocation capacities are determined by the number of records stored on an 852 disk track, which will store one of the following:

20 summary records (40 in an 853 or 854 disk track)

10 account number records (20 in an 853 or 854 disk track)

10 master cost records (20 in an 853 or 854 disk track)

9 basic work (scratch) file records (18 in an 853 or 854 disk track)

These factors can be used to compute the number of disk storage tracks that must be allocated to obtain a desired capacity.

Additional 852 Disk Storage Drives will obtain the maximum capacities. Maximum capacities are also achieved by replacing 852 Disk Storage Drives with 853 or 854 Disk Storage Drives. File sizes can then be increased to use the added disk storage capacity.

When files are allocated with the mode and block size indicated above, the PERT COST system operates most efficiently and optimizes use of the disk storage. EVEN1, EVEN2, and EVEN3 must be the first files on the 852 Disk Storage Drive containing SCRATCH1, SCRATCH2, and SCRATCH3. These files are not used by PERT COST, but they ensure that input/output requests do not cross cylinders on the 852 Disk Storage Drives. Files EVEN1, EVEN2, and EVEN3 are not required when 853 or 854 Disk Storage Drives are used. With these configurations, the block size of file COVERLAY should be changed to 256.

If only two 852 Disk Storage Drives are available, SCRATCH1, SCRATCH2, and SCRATCH3 can be allocated to the second disk storage drive with a file size of 250 tracks. File EVEN1 must be the first file on this 852 Disk Storage Drive. Files EVEN2 and EVEN3 are not allocated. The basic work file cannot exceed 2700 records.

5.5.2 ALLOCATE SCRATCH FILES

The control card examples illustrate the allocation of disk storage for scratch files. For complete format and discussion of control cards, see the MSOS Reference Manual.

$^7_9$SEQUENCE

$^7_9$JOB,,,

$^7_9$RAT,852/31

$^7_9$FET,PERT,EVEN1,2000

$^7_9$ALLOCATE,1,,S,NOSEG,852

$^7_9$FET,PERT,SCRATCH1,4000

$^7_9$ALLOCATE,758,,S,NOSEG,852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB,,,

$^{7}_{9}$RAT,852/32

$^{7}_{9}$FET,PERT,EVEN2,2000

$^{7}_{9}$ALLOCATE,1,S,NOSEG,852

$^{7}_{9}$FET,PERT,SCRATCH2,4000

$^{7}_{9}$ALLOCATE,758,,S,NOSEG,852

$^{7}_{9}$FET,PERT,COVERLAY,100

$^{7}_{9}$ALLOCATE,240,,S,NOSEG,852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB,,,

$^{7}_{9}$RAT,852/33

$^{7}_{9}$FET,PERT,EVEN3,2000

$^{7}_{9}$ALLOCATE,1,,S,NOSEG,852

$^{7}_{9}$FET,PERT,SCRATCH3,4000

$^{7}_{9}$ALLOCATE,758,,S,NOSEG,852

$^{77}_{88}$end-of-file

5.5.3 CREATE OVERLAY FILE AND RUN WITH PERT COST MASTER FILES ON TAPE

Execute PERT COST with SELECT JUMP 1 on. This is necessary for proper handling of the six individual files that make up the PERT COST data master. Construct a program to create a PERT COST overlay file and execute with master files on tape as follows:

```
  7
  9SEQUENCE
  7
  9JOB,,,
† 7
  9EQUIP,11=MT,12=11,13=11,14=11,15=11,16=11
† 7
  9EQUIP,21=MT,22=21,23=21,24=21,25=21,26=21
  7
  9FET,PERT,SCRATCH1,4000
  7
  9OPEN,01
  7
  9FET,PERT,SCRATCH2,4000
  7
  9OPEN,02
  7
  9FET,PERT,SCRATCH3,4000
  7
  9OPEN,03
  7
  9FET,PERT,COSTLINK,4000
  7
  9OPEN,07
  7
  9FET,PERT,COVERLAY,100
  7
  9OPEN,05

  +
  005
  7
  9

  binary cards for main program

  +
  0
  305,1
  7
  9
```

† When using magnetic tape files to save master files, SELECT JUMP 1 must be set for execution.

binary cards for first overlay

+
0
305,2
7
9

binary cards for second overlay

+
0
305,. . .
7
9

binary cards for last overlay

$^{7}_{9}$RUN

T control card
U control card

S cards
C cards
B cards
E cards
P cards optional
Q cards
R cards
F cards
G cards
H control card

report request cards in ascending order by type number

V control card

$^{7}_{9}$CLOSE, 05

$^{7}_{9}$CLOSE, 07

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , ,

$^{7}_{9}$FET, PERT, EVEN1, 2000

$^{7}_{9}$RELEASE, ALL

$^{7}_{9}$FET, PERT, SCRATCH1, 4000

$^{7}_{9}$RELEASE, ALL

$^{7}_{9}$FET, PERT, EVEN2, 2000

$^{7}_{9}$RELEASE, ALL

$^{7}_{9}$FET, PERT, SCRATCH2, 4000

$^{7}_{9}$RELEASE, ALL

$^{7}_{9}$FET, PERT, EVEN3, 2000

$^{7}_{9}$RELEASE, ALL

$^{7}_{9}$FET, PERT, SCRATCH3, 4000

$^{7}_{9}$RELEASE, ALL

$^{77}_{88}$end-of-file

When the COSTLINK or OVERLAY files are not being saved for future executions, the CLOSE card with the appropriate file ordinal should be followed by FET and RELEASE cards to remove the label from the file label directory.

To assign the PERT TIME summary (COSTLINK) file to magnetic tape, equip unit 7 to a magnetic tape drive.

5.5.4 ALLOCATION OF FILES WITH PERT COST MASTER FILES ON MASS STORAGE

The control card examples illustrate the allocation of PERT COST files to four 852 Disk Storage Drives. These cards must be in the order given. See the MSOS Reference Manual for complete format and discussion of control cards.

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , ,

$^{7}_{9}$RAT, 852/30

$^{7}_{9}$FET, PERT, OLDSUMMARYNUMBERS, 100

$^{7}_{9}$ALLOCATE, 25, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, OLDACCOUNTNUMBERS, 200

$^{7}_{9}$ALLOCATE, 100, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, OLDCOSTS, 200

$^{7}_{9}$ALLOCATE, 194, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, OLDRATES, 300

$^{7}_{9}$ALLOCATE, 150, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, OLDF+GCATEGORIES, 512

$^{7}_{9}$ALLOCATE, 20, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, OLDHIERARCHYTABLE, 512

$^{7}_{9}$ALLOCATE, 10, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, NEWSUMMARYNUMBERS, 100

$^{7}_{9}$ALLOCATE, 25, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, NEWACCOUNTNUMBERS, 200

$^{7}_{9}$ALLOCATE, 100, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, NEWCOSTS, 200

$^{7}_{9}$ALLOCATE, 194, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, NEWRATES, 300

$^{7}_{9}$ALLOCATE, 150, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, NEWF+GCATEGORIES, 512

$^{7}_{9}$ALLOCATE, 20, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, NEWHIERARCHYTABLE, 512

$^{7}_{9}$ALLOCATE, 10, , S, NOSEG, 852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , ,

$^{7}_{9}$RAT, 852/31

$^{7}_{9}$FET, PERT, EVEN1, 2000

$^{7}_{9}$ALLOCATE, 1, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, SCRATCH1, 4000

$^{7}_{9}$ALLOCATE, 758, , S, NOSEG, 852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , ,

$^{7}_{9}$RAT, 852/32

$^{7}_{9}$FET, PERT, EVEN2, 2000

$^{7}_{9}$ALLOCATE, 1, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, SCRATCH2, 4000

$^{7}_{9}$ALLOCATE, 758, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, COVERLAY, 100

$^{7}_{9}$ALLOCATE, 240, , S, NOSEG, 852

$^{77}_{88}$end-of-file

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , ,

$^{7}_{9}$RAT, 852/33

$^{7}_{9}$FET, PERT, EVEN3, 2000

$^{7}_{9}$ALLOCATE, 1, , S, NOSEG, 852

$^{7}_{9}$FET, PERT, SCRATCH3, 4000

$^{7}_{9}$ALLOCATE, 758, , S, NOSEG, 852

$^{77}_{88}$end-of-file

5.5.5 CREATE OVERLAY FILE AND RUN WITH PERT COST MASTER FILES ON MASS STORAGE

Construct a program to create a PERT COST overlay file and execute with master files on mass storage as follows.

$^{7}_{9}$SEQUENCE

$^{7}_{9}$JOB, , ,

$^{7}_{9}$FET, PERT, OLDSUMMARYNUMBERS, 100

$^{7}_{9}$OPEN, 11

$^{7}_{9}$FET, PERT, OLDACCOUNTNUMBERS, 200

$^{7}_{9}$OPEN, 12

$^{7}_{9}$FET, PERT, OLDCOSTS, 200

$^{7}_{9}$OPEN, 13

$^{7}_{9}$FET, PERT, OLDRATES, 300

$^{7}_{9}$OPEN, 14

$^{7}_{9}$FET, PERT, OLDF+GCATEGORIES, 512

$^{7}_{9}$OPEN, 15

$^{7}_{9}$FET, PERT, OLDHIERARCHYTABLE, 512

$^{7}_{9}$OPEN, 16

$^{7}_{9}$FET, PERT, NEWSUMMARYNUMBERS, 100

$^{7}_{9}$OPEN, 21

$^{7}_{9}$FET, PERT, NEWACCOUNTNUMBERS, 200

$^{7}_{9}$OPEN, 22

$^{7}_{9}$FET, PERT, NEWCOSTS, 200

$^{7}_{9}$OPEN, 23

$^{7}_{9}$FET, PERT, NEWRATES, 300

$^{7}_{9}$OPEN, 24

$^{7}_{9}$FET, PERT, NEWF+GCATEGORIES, 512

$^{7}_{9}$OPEN, 25

$^{7}_{9}$FET, PERT, NEWHIERARCHYTABLE, 512

$^{7}_{9}$OPEN, 26

$^{7}_{9}$FET, PERT, SCRATCH1, 4000

$^{7}_{9}$OPEN, 01

$^{7}_{9}$FET, PERT, SCRATCH2, 4000

$^{7}_{9}$OPEN, 02

$^{7}_{9}$FET, PERT, SCRATCH3, 4000

$^{7}_{9}$OPEN, 03

$^{7}_{9}$FET, PERT, COSTLINK, 4000

```
7OPEN,07
9
7
9FET,PERT,COVERLAY,100
7
9OPEN,05

+
0
7 05
9

binary cards for main program

+
0
305,1
7
9

binary cards for first overlay

+
0
305,2
7
9

binary cards for second overlay

.
.
.

+
0
305,n
7
9

binary cards for nth overlay

7
9RUN

T control card
U control card
```

$$\left.\begin{array}{l}\text{S cards}\\\text{C cards}\\\text{B cards}\\\text{E cards}\\\text{P cards}\\\text{Q cards}\\\text{R cards}\\\text{F cards}\\\text{G cards}\end{array}\right\}\quad\text{optional}$$

H control card

report request cards in ascending order by type number

V control card

$^7_9$CLOSE, 05

$^7_9$CLOSE, 07

$^7_9$CLOSE, 11

$^7_9$CLOSE, 12

$^7_9$CLOSE, 13

$^7_9$CLOSE, 14

$^7_9$CLOSE, 15

$^7_9$CLOSE, 16

$^7_9$CLOSE, 21

$^7_9$CLOSE, 22

$^7_9$CLOSE, 23

$^7_9$CLOSE, 24

$^7_9$CLOSE, 25

$^7_9$CLOSE, 26

$^{77}_{88}$end-of-file

${}^{7}_{9}$SEQUENCE

${}^{7}_{9}$JOB,,,

${}^{7}_{9}$FET,PERT,EVEN1,2000

${}^{7}_{9}$RELEASE,ALL

${}^{7}_{9}$FET,PERT,SCRATCH1,4000

${}^{7}_{9}$RELEASE,ALL

${}^{7}_{9}$FET,PERT,EVEN2,2000

${}^{7}_{9}$RELEASE,ALL

${}^{7}_{9}$FET,PERT,SCRATCH2,4000

${}^{7}_{9}$RELEASE,ALL

${}^{7}_{9}$FET,PERT,EVEN3,2000

${}^{7}_{9}$RELEASE,ALL

${}^{7}_{9}$FET,PERT,SCRATCH3,4000

${}^{7}_{9}$RELEASE,ALL

${}^{77}_{88}$end-of-file

When the master files, COSTLINK, or overlay files are not being saved for future executions, the CLOSE cards with the appropriate file ordinals should be followed by FET and RELEASE cards to remove the label from the file label directory.

5.6 USASI FORTRAN

5.6.1 USASI FORTRAN SYSTEM PREPARATION

Prepare the MSOS 4.2 library according to the following steps:

1. Install MSOS 4.2 with USASI option = 1 in LOADER, POSTLOAD, and OVPRO (Part II, Sections 2.2.15 through 2.2.17)
2. File 55 must be at least the size specified in the following table to execute the USASI FORTRAN compiler:

FILE 55	813	841	852	853	854	863	Device
	100	100	430	200	200	200	Tracks

3. Assign INP to the USASI FORTRAN release tape.
4. The following jobs are run. (See binary source listings in Section 5.6.2.)

   SEQUENCE, 001 allocates a mass storage file for the USASI FORTRAN compiler overlays. The user may obtain operator control at sequence time in order to RAT the overlay file to the system pack.

   SEQUENCE, 002 creates the USASI FORTRAN overlay file. Nine error messages should result from this job (one for each overlay or segment). The message is:

   I SYS 0164 subroutine UD MAP4MAIN

   MAP4MAIN is label intended to be unsatisfied by the system for the purpose of creating a printed map of MAIN.

   SEQUENCE, 003 allocates a mass storage file for USASI FORTRAN object-time auxiliary library. If the user desires, he can obtain operator control at SEQUENCE, 003 release the unused portion of the USASI FORTRAN compiler mass storage file and RAT the auxiliary library file to the system device.

   SEQUENCE, 004 creates a new edition (UF) of the MSOS library and an auxiliary library (UF) for the object-time routines. The user should obtain operator control at sequence time in order to RAT the PRELIB scratch files to a scratch pack (Part II, Section 4.6.5).
5. When installation is complete, run the verification deck (Part II, Section 6).

5.6.2 USASI FORTRAN BINARY SOURCE LISTINGS

One release tape is provided with USASI FORTRAN, containing three sections:

Binary source files

COSY formatted source

List file

Figure II-3 contains a diagram of the USASI FORTRAN compiler overlays as they occur during execution. The pages following Figure II-3 contain the binary source file listings.

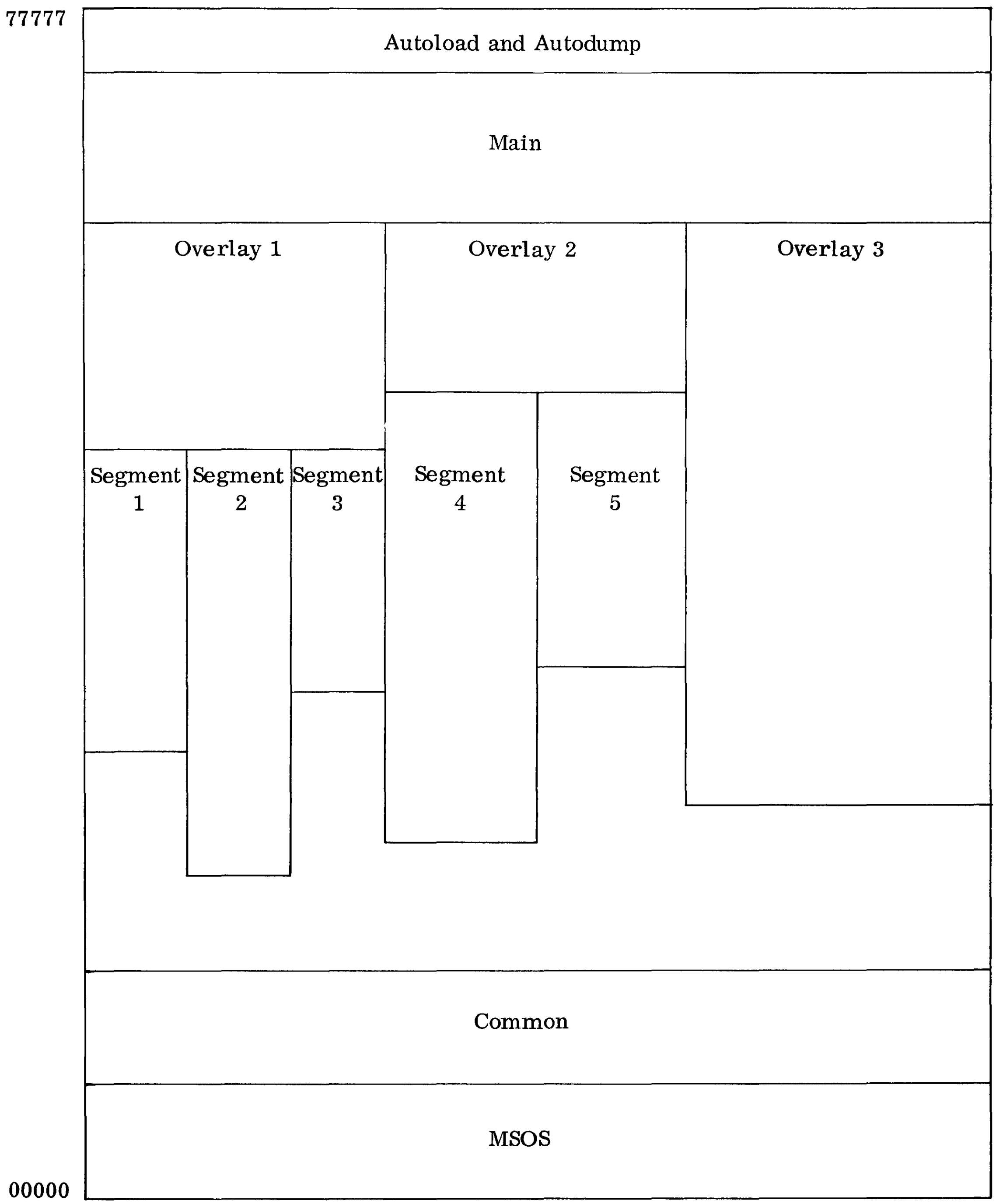


Figure II-3. Contents of Core During USASI FORTRAN Compilation

```
CN=00001    C.C.  SEQUENCE,001,ALLOCATE USASI FORTRAN OVERLAY FILES
CN=00002    C.C.  JOB,,,,
CN=00003    C.C.  FET,UFTN,USASIFORTRAN/MSOS,960,01,0000,0000
CN=00004    C.C.  ALLOCATE,70,991231,,NOSEG
           EOF


I PLIBEDIT 099 PLIBEDIT COMPLETED.
```

```
CN=00001     C.C.  SEQUENCE,002,USASI FORTRAN OVERLAYS
CN=00002     C.C.  JOB,,,
CN=00003     C.C.  FET,UFTN,USASIFORTRAN/MSOS,960,01,0000,0000
CN=00004     C.C.  OPEN,49
CN=00005     MAIN  49                 USASI FORTRAN MAIN OVERLAY CONTROL CARD
CN=00006     IDC  FTNUSASI  PROG.LENGTH=00530
CN=00022     IDC  COMPIO    PROG.LENGTH=03610
CN=00128     IDC  UFBINBCD  PROG.LENGTH=00013
CN=00132     IDC  ADDNAME   PROG.LENGTH=00041
CN=00137     IDC  UFMOVER   PROG.LENGTH=00052
CN=00143     IDC  QABORT    PROG.LENGTH=00047
CN=00150     IDC  LISTIN    PROG.LENGTH=00070
CN=00157     IDC  GETROOM   PROG.LENGTH=01107
CN=00180     IDC  LEXPAND   PROG.LENGTH=00641
CN=00198     IDC  MISQU     PROG.LENGTH=00163
CN=00207     IDC  COGEN     PROG.LENGTH=00672
CN=00226     OV    49,01              USASI FORTRAN OVERLAY ONE CONTROL CARD
CN=00227     IDC  Q.ENTRY1  PROG.LENGTH=00013
CN=00233     IDC  LABSRCH   PROG.LENGTH=00053
CN=00240     IDC  CONTRANS  PROG.LENGTH=01243
CN=00274     IDC  COMCHAR   PROG.LENGTH=00073
CN=00281     IDC  CHEKINT   PROG.LENGTH=00054
CN=00289     IDC  GETCARDS  PROG.LENGTH=01122
CN=00311     IDC  Q.MDDD    PROG.LENGTH=00415
CN=00325     IDC  CONTABS   PROG.LENGTH=00136
CN=00336     IDC  UFKEYWRD  PROG.LENGTH=01225
CN=00362     IDC  LABDEF    PROG.LENGTH=00106
CN=00370     IDC  SCAN3     PROG.LENGTH=00432
CN=00384     IDC  SCAN1     PROG.LENGTH=00401
CN=00397     IDC  CONGETTR  PROG.LENGTH=00714
CN=00417     IDC  LOGRELOP  PROG.LENGTH=00165
CN=00426     IDC  COMSTRNG  PROG.LENGTH=00071
CN=00433     IDC  DATATRAN  PROG.LENGTH=00161
CN=00442     IDC  SUBCRACK  PROG.LENGTH=02475
CN=00494     IDC  AOPERAND  PROG.LENGTH=00106
CN=00501     IDC  ENDDO     PROG.LENGTH=00620
CN=00521     SEG   49,01              USASI FORTRAN OVERLAY ONE - SEGMENT ONE
CN=00522     IDC  Q.ENTRY2  PROG.LENGTH=00003
CN=00527     IDC  DECLAR    PROG.LENGTH=06224
CN=00630     IDC  EQUSIZE   PROG.LENGTH=00213
CN=00640     IDC  CLEAR     PROG.LENGTH=00032
CN=00645     IDC  GETBSS    PROG.LENGTH=00206
CN=00654     IDC  COMBINE   PROG.LENGTH=00105
CN=00661     IDC  GETRUNS   PROG.LENGTH=00113
CN=00668     IDC  IALREADY  PROG.LENGTH=00063
CN=00674     SEG   49,02              USASI FORTRAN OVERLAY ONE - SEGMENT TWO
CN=00675     IDC  Q.ENTRY3  PROG.LENGTH=00003
CN=00680     IDC  FAZ1      PROG.LENGTH=00717
CN=00701     IDC  IODELAY   PROG.LENGTH=00226
CN=00713     IDC  IOSWITCH  PROG.LENGTH=01305
CN=00759     IDC  IOCONTRL  PROG.LENGTH=00077
```

```
CN=00765     IDC   POLCNTRL   PROG.LENGTH=04326
CN=00847     IDC   SN2HOLD    PROG.LENGTH=00106
CN=00856     IDC   DOER       PROG.LENGTH=00353
CN=00885     IDC   QCONPRO    PROG.LENGTH=01511
CN=00916     SEG    49,03              USASI FORTRAN OVERLAY ONE - SEGMENT THREE
CN=00917     IDC   Q.ENTRY4   PROG.LENGTH=00003
CN=00922     IDC   ISWITCH    PROG.LENGTH=02256
CN=00984     IDC   BKRWEF     PROG.LENGTH=03223
CN=01060     IDC   ENDFAZ1    PROG.LENGTH=00321
CN=01075     IDC   SCAN4      PROG.LENGTH=00051
CN=01083     IDC   GETDIMS    PROG.LENGTH=00376
CN=01098     OV     49,02              USASI FORTRAN OVERLAY TWO CONTROL CARD
CN=01099     IDC   Q.ENTRY5   PROG.LENGTH=00010
CN=01105     IDC   FPCOGEN    PROG.LENGTH=00160
CN=01117     IDC   TABLEIT    PROG.LENGTH=00300
CN=01130     IDC   ADDFUNC    PROG.LENGTH=00030
CN=01136     SEG    49,04              USASI FORTRAN OVERLAY TWO   - SEGMENT ONE
CN=01137     IDC   Q.ENTRY6   PROG.LENGTH=00003
CN=01142     IDC   ITOR       PROG.LENGTH=00026
CN=01146     IDC   ITOD       PROG.LENGTH=00040
CN=01150     IDC   RTOI       PROG.LENGTH=00104
CN=01157     IDC   FAZ3       PROG.LENGTH=00743
CN=01176     IDC   GENERAL    PROG.LENGTH=00066
CN=01183     IDC   INDXCALL   PROG.LENGTH=00201
CN=01192     IDC   DOPRO      PROG.LENGTH=00427
CN=01230     IDC   DOPARAM    PROG.LENGTH=00545
CN=01253     IDC   CALLPAUL   PROG.LENGTH=00776
CN=01274     IDC   CONGEN     PROG.LENGTH=01045
CN=01297     IDC   COMPUTE    PROG.LENGTH=00371
CN=01315     IDC   FRSTLEVL   PROG.LENGTH=05747
CN=01428     IDC   SCNDLEVL   PROG.LENGTH=00757
CN=01452     IDC   THRDLEVL   PROG.LENGTH=02072
CN=01494     IDC   FRTHLEVL   PROG.LENGTH=01435
CN=01527     IDC   FILTER     PROG.LENGTH=00372
CN=01545     IDC   GETREGS    PROG.LENGTH=00361
CN=01559     IDC   CODER      PROG.LENGTH=00137
CN=01566     IDC   RELOCATE   PROG.LENGTH=00120
CN=01574     IDC   INSTR      PROG.LENGTH=00514
CN=01591     IDC   STL2OPL    PROG.LENGTH=00346
CN=01606     IDC   INTRNSIC   PROG.LENGTH=00543
CN=01624     IDC   CONV       PROG.LENGTH=00104
CN=01632     SEG    49,05              USASI FORTRAN OVERLAY TWO   - SEGMENT TWO
CN=01633     IDC   Q.ENTRY7   PROG.LENGTH=00003
CN=01638     IDC   UFAND      PROG.LENGTH=00011
CN=01642     IDC   ERRPRO     PROG.LENGTH=07510
CN=01770     IDC   INFGEN     PROG.LENGTH=01561
CN=01818     IDC   QSWITCH    PROG.LENGTH=00060
CN=01823     IDC   QPOWER     PROG.LENGTH=00055
CN=01828     IDC   RISROS     PROG.LENGTH=00124
CN=01836     IDC   EXTGEN     PROG.LENGTH=01126
CN=01867     IDC   INITCODE   PROG.LENGTH=01012
CN=01908     IDC   QRADIX     PROG.LENGTH=00331
CN=01921     IDC   FMTSUBS    PROG.LENGTH=01347
CN=01944     IDC   FMTSYN     PROG.LENGTH=00661
CN=01983     IDC   GETEL      PROG.LENGTH=00107
```

```
CN=01991     IDC   GETSUB     PROG.LENGTH=00423
CN=02014     IDC   DATASUBS   PROG.LENGTH=00425
CN=02040     IDC   DATAGEN    PROG.LENGTH=00313
CN=02063     IDC   TABCHK1    PROG.LENGTH=00173
CN=02071     IDC   TABCHK2    PROG.LENGTH=00132
CN=02078     IDC   BINBCD2    PROG.LENGTH=00126
CN=02086     OV     49,03               USASI FORTRAN OVERLAY THREE CONTROL CARD
CN=02087     IDC   Q.ENTRY8   PROG.LENGTH=00003
CN=02092     IDC   CODEGEN    PROG.LENGTH=23120
CN=02170     C.C.   CTO,    END OF OVERLAY PREPARATION
            EOF


I PLIBEDIT 099 PLIBEDIT COMPLETED.
```

```
CN=00001     C.C.  SEQUENCE,003,ALLOCATE UFTN AUX FILE
CN=00002     C.C.  JOB,,,,
CN=00003     C.C.  FET,MSOS,AUXFILE,500,UF
CN=00004     C.C.  ALLOCATE,100,991231
            EOF


I PLIBEDIT 099 PLIBEDIT COMPLETED.
```

```
CN=00001    C.C.  SEQUENCE,004,UFTN AUX LIB PRELIB
CN=00002    C.C.  JOB,,,,
CN=00003    C.C.  FET,MSOS,AUXFILE,500,UF
CN=00004    C.C.  OPEN,30
CN=00005    C.C.  PRELIB,,,,,UF
CN=00006    C.C.  FILE
CN=00007    C.C.  INSERT
CN=00008    IDC  UFORTRAN   PROG.LENGTH=21700
CN=00020    C.C.  FILE,30
CN=00021    IDC  UFOVER     PROG.LENGTH=00263
CN=00033    IDC  Q.IODONE   PROG.LENGTH=00013
CN=00037    IDC  UFBNARR    PROG.LENGTH=00110
CN=00044    IDC  UFBINARY   PROG.LENGTH=00302
CN=00056    IDC  UFBUFFIO   PROG.LENGTH=00345
CN=00072    IDC  UFBCDIN    PROG.LENGTH=02067
CN=00123    IDC  UFBCDOUT   PROG.LENGTH=02160
CN=00177    IDC  UFIOUTIL   PROG.LENGTH=01614
CN=00287    IDC  UFIO       PROG.LENGTH=03165
CN=00369    IDC  UFLITES    PROG.LENGTH=00064
CN=00376    IDC  Q.ERROR    PROG.LENGTH=00167
CN=00384    IDC  UFSELECT   PROG.LENGTH=00307
CN=00414    IDC  Q.STOP     PROG.LENGTH=00163
CN=00422    IDC  Q.SCHECK   PROG.LENGTH=00014
CN=00427    IDC  UFAIMAG    PROG.LENGTH=00005
CN=00432    IDC  UFAINT     PROG.LENGTH=00154
CN=00441    IDC  UFALOG10   PROG.LENGTH=00015
CN=00446    IDC  Q.EXISC    PROG.LENGTH=00012
CN=00452    IDC  Q.EXIC     PROG.LENGTH=00031
CN=00458    IDC  Q.EXCIS    PROG.LENGTH=00016
CN=00464    IDC  Q.EXCI     PROG.LENGTH=00032
CN=00470    IDC  Q.EXCD     PROG.LENGTH=00016
CN=00476    IDC  Q.EXDC     PROG.LENGTH=00013
CN=00482    IDC  Q.EXRC     PROG.LENGTH=00016
CN=00488    IDC  Q.EXCR     PROG.LENGTH=00017
CN=00494    IDC  Q.EXCC     PROG.LENGTH=00035
CN=00500    IDC  UFCLOG     PROG.LENGTH=00074
CN=00507    IDC  UFATAN     PROG.LENGTH=00242
CN=00518    IDC  Q.EXSIS    PROG.LENGTH=00116
CN=00526    IDC  Q.EXII     PROG.LENGTH=00137
CN=00536    IDC  Q.EXRIS    PROG.LENGTH=00020
CN=00542    IDC  Q.EXRI     PROG.LENGTH=00177
CN=00553    IDC  Q.EXISR    PROG.LENGTH=00036
CN=00559    IDC  Q.EXIR     PROG.LENGTH=00040
CN=00565    IDC  Q.EXRR     PROG.LENGTH=00063
CN=00573    IDC  UFALOG     PROG.LENGTH=00204
CN=00583    IDC  UFCABS     PROG.LENGTH=00054
CN=00590    IDC  UFCEXP     PROG.LENGTH=00101
CN=00598    IDC  UFCMPLX    PROG.LENGTH=00012
CN=00603    IDC  UFCONJ     PROG.LENGTH=00012
CN=00608    IDC  UFCSIN     PROG.LENGTH=00077
CN=00615    IDC  UFDABS     PROG.LENGTH=00017
```

```
CN=00620      IDC   UFDBLE     PROG.LENGTH=00012
CN=00625      IDC   UFDCOSSN   PROG.LENGTH=00534
CN=00645      IDC   Q.EXDIS    PROG.LENGTH=00017
CN=00651      IDC   Q.EXDI     PROG.LENGTH=00214
CN=00662      IDC   Q.EXDR     PROG.LENGTH=00025
CN=00668      IDC   Q.EXISD    PROG.LENGTH=00012
CN=00674      IDC   Q.EXID     PROG.LENGTH=00053
CN=00681      IDC   Q.EXRD     PROG.LENGTH=00021
CN=00687      IDC   Q.EXDD     PROG.LENGTH=00075
CN=00695      IDC   UFDEXP     PROG.LENGTH=00342
CN=00710      IDC   UFDIMDIM   PROG.LENGTH=00027
CN=00715      IDC   UFDLOG2    PROG.LENGTH=00510
CN=00734      IDC   UFDMOD     PROG.LENGTH=00166
CN=00745      IDC   UFDSIGN    PROG.LENGTH=00030
CN=00750      IDC   UFDSQRT    PROG.LENGTH=00123
CN=00758      IDC   UFEXP      PROG.LENGTH=00157
CN=00768      IDC   UFFLOAT    PROG.LENGTH=00016
CN=00773      IDC   UFIDINT    PROG.LENGTH=00036
CN=00778      IDC   UFIFIX     PROG.LENGTH=00016
CN=00783      IDC   UFMINMAX   PROG.LENGTH=00246
CN=00793      IDC   UFMINMXD   PROG.LENGTH=00125
CN=00801      IDC   UFSINCOS   PROG.LENGTH=00330
CN=00818      IDC   UFSIGN     PROG.LENGTH=00042
CN=00824      IDC   UFSNGL     PROG.LENGTH=00010
CN=00829      IDC   UFSQRT     PROG.LENGTH=00121
CN=00837      IDC   Q.ADCR     PROG.LENGTH=00020
CN=00843      IDC   Q.ADID     PROG.LENGTH=00045
CN=00850      IDC   Q.ADDC     PROG.LENGTH=00023
CN=00856      IDC   Q.ADRD     PROG.LENGTH=00022
CN=00862      IDC   Q.ADDIS    PROG.LENGTH=00041
CN=00869      IDC   Q.ADDR     PROG.LENGTH=00025
CN=00876      IDC   Q.ADISD    PROG.LENGTH=00035
CN=00882      IDC   Q.ADDI     PROG.LENGTH=00051
CN=00889      IDC   Q.ADCD     PROG.LENGTH=00012
CN=00895      IDC   Q.ADISC    PROG.LENGTH=00012
CN=00901      IDC   Q.ADCIS    PROG.LENGTH=00016
CN=00907      IDC   Q.ADCI     PROG.LENGTH=00032
CN=00913      IDC   Q.ADIC     PROG.LENGTH=00032
CN=00919      IDC   Q.ADRC     PROG.LENGTH=00016
CN=00925      IDC   Q.ADRI     PROG.LENGTH=00026
CN=00931      IDC   Q.ADRIS    PROG.LENGTH=00027
CN=00937      IDC   Q.ADISR    PROG.LENGTH=00023
CN=00943      IDC   Q.ADIR     PROG.LENGTH=00022
CN=00949      IDC   Q.ADCC     PROG.LENGTH=00013
CN=00955      IDC   Q.DVRD     PROG.LENGTH=00021
CN=00961      IDC   Q.DVID     PROG.LENGTH=00044
CN=00968      IDC   Q.DVCR     PROG.LENGTH=00020
CN=00975      IDC   Q.DVISD    PROG.LENGTH=00035
CN=00981      IDC   Q.DVCD     PROG.LENGTH=00020
CN=00988      IDC   Q.DVDI     PROG.LENGTH=00050
CN=00995      IDC   Q.DVDIS    PROG.LENGTH=00041
CN=01001      IDC   Q.DVDC     PROG.LENGTH=00016
CN=01007      IDC   Q.DVDR     PROG.LENGTH=00025
CN=01013      IDC   Q.DVISC    PROG.LENGTH=00012
CN=01019      IDC   Q.DVIC     PROG.LENGTH=00031
```

```
CN=01025     IDC  Q.DVCIS   PROG.LENGTH=00016
CN=01031     IDC  Q.DVCI    PROG.LENGTH=00032
CN=01038     IDC  Q.DVRC    PROG.LENGTH=00016
CN=01044     IDC  Q.DVRIS   PROG.LENGTH=00033
CN=01050     IDC  Q.DVRI    PROG.LENGTH=00032
CN=01056     IDC  Q.DVIR    PROG.LENGTH=00022
CN=01062     IDC  Q.DVISR   PROG.LENGTH=00023
CN=01068     IDC  Q.DVCC    PROG.LENGTH=00121
CN=01079     IDC  Q.STRC    PROG.LENGTH=00023
CN=01085     IDC  Q.STRD    PROG.LENGTH=00030
CN=01091     IDC  Q.STDR    PROG.LENGTH=00011
CN=01097     IDC  Q.STISD   PROG.LENGTH=00042
CN=01104     IDC  Q.STDC    PROG.LENGTH=00022
CN=01110     IDC  Q.STCD    PROG.LENGTH=00030
CN=01116     IDC  Q.STID    PROG.LENGTH=00054
CN=01123     IDC  Q.STRI    PROG.LENGTH=00111
CN=01132     IDC  Q.STCIS   PROG.LENGTH=00054
CN=01140     IDC  Q.STCI    PROG.LENGTH=00061
CN=01148     IDC  Q.STISC   PROG.LENGTH=00037
CN=01154     IDC  Q.STISR   PROG.LENGTH=00043
CN=01162     IDC  Q.STIC    PROG.LENGTH=00035
CN=01168     IDC  Q.STCR    PROG.LENGTH=00011
CN=01174     IDC  Q.STDIS   PROG.LENGTH=00056
CN=01182     IDC  Q.STDI    PROG.LENGTH=00067
CN=01190     IDC  Q.STIR    PROG.LENGTH=00041
CN=01198     IDC  Q.STRIS   PROG.LENGTH=00106
CN=01207     IDC  Q.STCC    PROG.LENGTH=00020
CN=01214     IDC  Q.SBRC    PROG.LENGTH=00016
CN=01220     IDC  Q.SBCR    PROG.LENGTH=00020
CN=01226     IDC  Q.SBDI    PROG.LENGTH=00051
CN=01233     IDC  Q.SBRD    PROG.LENGTH=00021
CN=01239     IDC  Q.SBDC    PROG.LENGTH=00022
CN=01245     IDC  Q.SBCD    PROG.LENGTH=00012
CN=01251     IDC  Q.SBISD   PROG.LENGTH=00035
CN=01257     IDC  Q.SBID    PROG.LENGTH=00045
CN=01264     IDC  Q.SBDIS   PROG.LENGTH=00041
CN=01270     IDC  Q.SBDR    PROG.LENGTH=00025
CN=01276     IDC  Q.SBCIS   PROG.LENGTH=00016
CN=01282     IDC  Q.SBCI    PROG.LENGTH=00032
CN=01288     IDC  Q.SBISC   PROG.LENGTH=00012
CN=01294     IDC  Q.SBIC    PROG.LENGTH=00031
CN=01300     IDC  Q.SBISR   PROG.LENGTH=00023
CN=01306     IDC  Q.SBRI    PROG.LENGTH=00026
CN=01312     IDC  Q.SBRIS   PROG.LENGTH=00027
CN=01318     IDC  Q.SBIR    PROG.LENGTH=00022
CN=01324     IDC  Q.SBCC    PROG.LENGTH=00013
CN=01330     IDC  Q.SBDD    PROG.LENGTH=00026
CN=01336     IDC  Q.MURC    PROG.LENGTH=00016
CN=01342     IDC  Q.MUDC    PROG.LENGTH=00013
CN=01348     IDC  Q.MUDR    PROG.LENGTH=00025
CN=01354     IDC  Q.MUCD    PROG.LENGTH=00013
CN=01360     IDC  Q.MURD    PROG.LENGTH=00021
CN=01366     IDC  Q.MUISD   PROG.LENGTH=00034
CN=01372     IDC  Q.MUDIS   PROG.LENGTH=00041
CN=01378     IDC  Q.MUID    PROG.LENGTH=00045
```

```
CN=01385     IDC  Q.MUDI     PROG.LENGTH=00051
CN=01392     IDC  Q.MUCIS    PROG.LENGTH=00016
CN=01398     IDC  Q.MUISC    PROG.LENGTH=00012
CN=01404     IDC  Q.MUCI     PROG.LENGTH=00032
CN=01410     IDC  Q.MUIC     PROG.LENGTH=00032
CN=01416     IDC  Q.MUCR     PROG.LENGTH=00020
CN=01422     IDC  Q.MURI     PROG.LENGTH=00026
CN=01428     IDC  Q.MUIR     PROG.LENGTH=00022
CN=01434     IDC  Q.MUISR    PROG.LENGTH=00023
CN=01440     IDC  Q.MURIS    PROG.LENGTH=00027
CN=01446     IDC  Q.MUCC     PROG.LENGTH=00030
CN=01452     IDC  Q.ADDD     PROG.LENGTH=01306
CN=01482     IDC  Q.LDCC     PROG.LENGTH=00020
CN=01493     IDC  UFFORCON   PROG.LENGTH=00450
CN=01508     IDC  UFUTIL     PROG.LENGTH=00140
CN=01520     IDC  Q.XOACC    PROG.LENGTH=00022
CN=01530     IDC  UFTANH     PROG.LENGTH=00215
CN=01541     IDC  UFCSQRT    PROG.LENGTH=00132
CN=01548     IDC  UFDATAN2   PROG.LENGTH=00133
CN=01557     IDC  UFDATAN    PROG.LENGTH=00654
CN=01607     C.C.  FILE
            EOF
            ENDSCOPE,R
            EOF

I PLIBEDIT 099 PLIBEDIT COMPLETED.
```

5.6.3 COSY FILE CONTENTS

The following list gives the sequence of the routines found on the COSY formatted source of the USASI FORTRAN release tape.

The deck name of each routine in the list is suffixed with code A. Code A is a specified routine used for all MSOS libraries.

The deck name is also suffixed with a source language code, defined as follows:

SMP	COMPASS
UFTN	USASI FORTRAN

USASI FORTRAN COSY Tape Structure

Table II-7 lists the USASI FORTRAN COSY deck names in order of occurrence on the COSY tape. The table also shows the organization of the compiler on the overlay file. Compiler routines compiled with USASI FORTRAN must be compiled with the S option.† Object time routines compiled with USASI FORTRAN must be compiled without the S option.

† The presence of the S parameter on the $^{7}_{9}$UFORTRAN card specifies integers and logical variables compiles as 24-bit entities; the absence of the S parameter as 48-bit entities.

TABLE II-7. USASI FORTRAN COSY TAPE

Deck Name	Code	Source Language	Product Name	Deck Name	Code	Source Language	Product Name
FTNUSASI	A	CMP	USASI FORTRAN COMPILER ROUTINES	Q. ENTRY4	A	CMP	USASI FORTRAN COMPILER ROUTINES
COMPIO	A	CMP		ISWITCH	A	UFTN	
UFBINBCD	A	CMP		BKRWEF	A	UFTN	
ADDNAME	A	CMP		ENDFAZ1	A	UFTN	
UFMOVER	A	CMP		SCAN4	A	UFTN	
QABORT	A	CMP		GETDIMS	A	UFTN	
LISTIN	A	UFTN		Q. ENTRY5	A	CMP	
GETROOM	A	UFTN		FPCOGEN	A	UFTN	
LEXPAND	A	UFTN		TABLEIT	A	UFTN	
MISQU	A	UFTN		ADDFUNC	A	UFTN	
COGEN	A	UFTN		Q. ENTRY6	A	CMP	
Q. ENTRY1	A	CMP		ITOR	A	CMP	
LABSRCH	A	CMP		ITOD	A	CMP	
CONTRANS	A	CMP		RTOI	A	CMP	
COMCHAR	A	CMP		FAZ3	A	UFTN	
CHEKINT	A	CMP		GENERAL	A	UFTN	
GETCARDS	A	CMP		INDXCALL	A	UFTN	
Q. MDDD	A	CMP		DOPRO	A	UFTN	
CONTABS	A	CMP		DOPARAM	A	UFTN	
UFKEYWRD	A	CMP		CALLPAUL	A	UFTN	
LABDEF	A	UFTN		CONGEN	A	UFTN	
SCAN3	A	UFTN		COMPUTE	A	UFTN	
SCAN1	A	UFTN		FRSTLEVL	A	UFTN	
CONGETTR	A	UFTN		SCNDLEVL	A	UFTN	
LOGRELOP	A	UFTN		THRDLEVL	A	UFTN	
COMSTRNG	A	UFTN		FRTHLEVL	A	UFTN	
DATATRAN	A	UFTN		FILTER	A	UFTN	
SUBCRACK	A	UFTN		GETREGS	A	UFTN	
AOPERAND	A	UFTN		CODER	A	UFTN	
ENDDO	A	UFTN		RELOCATE	A	UFTN	
Q. ENTRY2	A	CMP		INSTR	A	UFTN	
DECLAR	A	UFTN		STL20PL	A	UFTN	
EQUSIZE	A	UFTN		INTRNSIC	A	UFTN	
CLEAR	A	UFTN		CONV	A	UFTN	
GETBSS	A	UFTN		Q. ENTRY7	A	CMP	
COMBINE	A	UFTN		UFAND	A	CMP	
GETRUNS	A	UFTN		ERRPRO	A	CMP	
IALREADY	A	UFTN		INFGEN	A	UFTN	
Q. ENTRY3	A	CMP		QSWITCH	A	UFTN	
FAZ1	A	UFTN		QPOWER	A	UFTN	
IODELAY	A	UFTN		RISROS	A	UFTN	
IOSWITCH	A	UFTN		EXTGEN	A	UFTN	
IOCONTRL	A	UFTN		INITCODE	A	UFTN	
POLCNTRL	A	UFTN		QRADIX	A	UFTN	
SN2HOLD	A	UFTN		FMTSUBS	A	UFTN	
DOER	A	UFTN		FMTSYN	A	UFTN	
QCONPRO	A	UFTN		GETEL	A	UFTN	

TABLE II-7. USASI FORTRAN COSY TAPE (cont'd)

Deck Name	Code	Source Language	Product Name
GETSUB	A	UFTN	USASI FORTRAN COMPILER ROUTINES
DATASUBS	A	UFTN	
DATAGEN	A	UFTN	
TABCHK1	A	UFTN	
TABCHK2	A	UFTN	
BINBCD2	A	UFTN	
Q.ENTRY8	A	CMP	
CODEGEN	A	CMP	
UFORTRAN	A	CMP	USASI FTN LDR
UFOVER	A	CMP	USASI FORTRAN OBJECT TIME ROUTINES
Q.IODONE	A	CMP	
UFBNARR	A	CMP	
UFBINARY	A	CMP	
UFBUFFIO	A	CMP	
UFBCDIN	A	CMP	
UFBCDOUT	A	CMP	
UFIOUTIL	A	CMP	
UFIO	A	CMP	
UFLITES	A	CMP	
Q.ERROR	A	CMP	
UFSELECT	A	CMP	
Q.STOP	A	CMP	
Q.SCHECK	A	CMP	
UFAIMAG	A	CMP	
UFAINT	A	CMP	
UFALOG10	A	CMP	
Q.EXISC	A	CMP	
Q.EXIC	A	CMP	
Q.EXCIS	A	CMP	
Q.EXCI	A	CMP	
Q.EXCD	A	CMP	
Q.EXDC	A	CMP	
Q.EXRC	A	CMP	
Q.EXCR	A	CMP	
Q.EXCC	A	CMP	
UFCLOG	A	CMP	
UFATAN	A	CMP	
Q.EXSIS	A	CMP	
Q.EXII	A	CMP	
Q.EXRIS	A	CMP	
Q.EXRI	A	CMP	
Q.EXISR	A	CMP	
Q.EXIR	A	CMP	
Q.EXRR	A	CMP	
UFALOG	A	CMP	
UFCABS	A	CMP	

Deck Name	Code	Source Language	Product Name
UFCEXP	A	CMP	USASI FORTRAN OBJECT TIME ROUTINES
UFCMPLX	A	CMP	
UFCONJG	A	CMP	
UFCSIN	A	CMP	
UFDABS	A	CMP	
UFDBLE	A	CMP	
UFDCOSSN	A	CMP	
Q.EXDIS	A	CMP	
Q.EXDI	A	CMP	
Q.EXDR	A	CMP	
Q.EXISD	A	CMP	
Q.EXID	A	CMP	
Q.EXRD	A	CMP	
Q.EXDD	A	CMP	
UFDEXP	A	CMP	
UFDIMDIM	A	CMP	
UFDLOG2	A	CMP	
UFDMOD	A	CMP	
UFDSIGN	A	CMP	
UFDSQRT	A	CMP	
UFEXP	A	CMP	
UFFLOAT	A	CMP	
UFIDINT	A	CMP	
UFIFIX	A	CMP	
UFMINMAX	A	CMP	
UFMINMXD	A	CMP	
UFSINCOS	A	CMP	
UFSIGN	A	CMP	
UFSNGL	A	CMP	
UFSQRT	A	CMP	
Q.ADCR	A	CMP	
Q.ADID	A	CMP	
Q.ADDC	A	CMP	
Q.ADRD	A	CMP	
Q.ADDIS	A	CMP	
Q.ADDR	A	CMP	
Q.ADISD	A	CMP	
Q.ADDI	A	CMP	
Q.ADCD	A	CMP	
Q.ADISC	A	CMP	
Q.ADCIS	A	CMP	
Q.ADCI	A	CMP	
Q.ADIC	A	CMP	
Q.ADRC	A	CMP	
Q.ADRI	A	CMP	
Q.ADRIS	A	CMP	
Q.ADISR	A	CMP	
Q.ADIR	A	CMP	
Q.ADCC	A	CMP	
Q.DVRD	A	CMP	

TABLE II-7. USASI FORTRAN COSY TAPE (cont'd)

Deck Name	Code	Source Language	Product Name
Q.DVID	A	CMP	
Q.DVCR	A	CMP	
Q.DVISD	A	CMP	
Q.DVCD	A	CMP	
Q.DVDI	A	CMP	
Q.DVDIS	A	CMP	
Q.DVDC	A	CMP	
Q.DVDR	A	CMP	
Q.DVISC	A	CMP	
Q.DVIC	A	CMP	
Q.DVCIS	A	CMP	
Q.DVCI	A	CMP	
Q.DVRC	A	CMP	
Q.DVRIS	A	CMP	
Q.DVRI	A	CMP	
Q.DVIR	A	CMP	
Q.DVISR	A	CMP	
Q.DVCC	A	CMP	
Q.STRC	A	CMP	
Q.STRD	A	CMP	USASI
Q.STDR	A	CMP	FORTRAN
Q.STISD	A	CMP	OBJECT
Q.STDC	A	CMP	TIME
Q.STCD	A	CMP	ROUTINES
Q.STID	A	CMP	
Q.STRI	A	CMP	
Q.STCIS	A	CMP	
Q.STCI	A	CMP	
Q.STISC	A	CMP	
Q.STISR	A	CMP	
Q.STIC	A	CMP	
Q.STCR	A	CMP	
Q.STDIS	A	CMP	
Q.STDI	A	CMP	
Q.STIR	A	CMP	
Q.STRIS	A	CMP	
Q.STCC	A	CMP	
Q.SBRC	A	CMP	
Q.SBCR	A	CMP	
Q.SBDI	A	CMP	
Q.SBRD	A	CMP	
Q.SBDC	A	CMP	
Q.SBCD	A	CMP	
Q.SBISD	A	CMP	
Q.SBID	A	CMP	
Q.SBDIS	A	CMP	
Q.SBDR	A	CMP	
Q.SBCIS	A	CMP	
Q.SBCI	A	CMP	

Deck Name	Code	Source Language	Product Name
Q.SBISC	A	CMP	
Q.SBIC	A	CMP	
Q.SBISR	A	CMP	
Q.SBRI	A	CMP	
Q.SBRIS	A	CMP	
Q.SBIR	A	CMP	
Q.SBCC	A	CMP	
Q.SBDD	A	CMP	
Q.MURC	A	CMP	
Q.MUDC	A	CMP	
Q.MUDR	A	CMP	
Q.MUCD	A	CMP	
Q.MURD	A	CMP	
Q.MUISD	A	CMP	
Q.MUDIS	A	CMP	USASI
Q.MUID	A	CMP	FORTRAN
Q.MUDI	A	CMP	OBJECT
Q.MUCIS	A	CMP	TIME
Q.MUISC	A	CMP	ROUTINES
Q.MUCI	A	CMP	
Q.MUIC	A	CMP	
Q.MUCR	A	CMP	
Q.MURI	A	CMP	
Q.MUIR	A	CMP	
Q.MUISR	A	CMP	
Q.MURIS	A	CMP	
Q.MUCC	A	CMP	
Q.ADDD	A	CMP	
Q.LDCC	A	CMP	
UFFORCON	A	CMP	
UFUTIL	A	CMP	
Q.XOACC	A	CMP	
UFTANH	A	UFTN	
UFCSQRT	A	UFTN	
UFDATAN2	A	UFTN	
UFDATAN	A	UFTN	

5.7 USASI COBOL

5.7.1 USASI COBOL SYSTEM PREPARATION

Prepare the MSOS 4.2 library according to the following steps:

1. Install MSOS 4.2 with USASI OPT = 1 in LOADER, POSTLOAD, and OVPRO (Part II, Section 2.2.15 through 2.2.17) and USARDT = 1 in xCICREC1 where x = B, S, or M (Part II, Sections 2.2.10 through 2.2.12).
2. File 55 must be at least the size specified in the following table to assemble the USASI COBOL compiler:

File 55	813	841	852	853	854	863	device
	250	220	1000	500	500	500	tracks

3. Select the assembly options (Part II, Sections 5.7.2 through 5.7.11) and, if necessary run COSY and COMPASS jobs to incorporate these options and assemble the new binary decks. (Part II, Section 3.2).
4. If new binary decks for the assembly options have been created in step 3, run a PLIBEDIT job to generate the USASI COBOL PRELIB source (Part II, Section 4.5 and Part III, Section 10). Note that, if user has a Memory Protect system, the P parameter must be specified on the PRELIB control card.
5. Run the PRELIB job with input assigned to the PRELIB source created in step 4. This step generates the new edition (UC) of the MSOS library.
6. When installation is complete, run the verification deck (Part II, Section 6).

5.7.2 USASI COBOL OPTION SELECTION

The following definitions explain the assembly options within each particular routine. The user is cautioned to correlate the particular assembly option setting with the major installation option on the USASI COBOL binary source file to determine if the assembly option must be changed. Note there is a difference between the COSY option setting and the assembled value on the binary source in some cases. The format of each definition is in the following form:

Routine

Assembly Option (mnemonic)

Definition:

Option:	COSY modifier	
mnemonic	operator	address

Deck Card:

Dependency:

Release Value:

Γable II-8 shows which assembly option is applicable to each routine:

TABLE II-8. USASI COBOL ASSEMBLY OPTIONS

Reassembled Routine	Assembly Options		
	BDP MULDV	BDP 3312	NONUSASI
UDCT			X
UCIE			X
UCDD			X
UCR1			X
UCP1			X
UCG1	X	X	
UCBLSORT		X	
UFIGCON		X	
UVARAN		X	

The assembly options are listed in a alphabetical order and the routines are in the order they appear on the USASI COBOL binary source file.

5.7.3 UDCT

USASI COBOL Option (NONUSASI)

Definition: The option NONUSASI depicts two different versions of the USASI COBOL compiler. The USASI only version conforms to the American Nation Standards Institute COBOL standards. The NONUSASI version contains additional COBOL features over and above the USASI standard. The NONUSASI assembly option is incorporated into five routines: UDCT, UCDD, UCIE, UCR1, and UCP1.

Option:		DELETE /	10	
	NONUSASI	EQU	0	NON-USASI COBOL version
	NONUSASI	EQU	1	USASI COBOL only version
Deck Card:	UDCT	DECK/	I=22, H	(see Dictionary Assembly Option)

Release Value: NONUSASI=0 on the UCBL COSY tape and USASI COBOL binary source.

NOTE

DICT is a program consisting of USASI COBOL reserved words and hash tables to aid in the syntax analysis of COBOL source text. The binary deck is created by executing the program UDCT which is located on COSY tape. All other binary decks are created by the normal execution of a COMPASS assembly.

The following job should be run to produce the binary deck for UDCT:

$^{7}_{9}$SEQUENCE, 001

$^{7}_{9}$JOB, 33L13, , ,

$^{7}_{9}$EQUIP, 22=MT

$^{7}_{9}$COSY

UDCT	DECK/	I=22, H
	ENDCOSY/	

$^{7}_{9}$COMPASS, I=54, X

$^{7}_{9}$LOAD, 56

$^{7}_{9}$RUN

77
88

5.7.4 UCIE

USASI COBOL Option (NONUSASI)

Definition: See Section 5.7.3.

Option:		DELETE/	436	
	NONUSASI	EQU	0	NONUSASI COBOL version
	NONUSASI	EQU	1	USASI COBOL only version
Deck Card:	UCIE	DECK/	p_1, p_2	

Release Value: NONUSASI=0 on the UCBL COSY tape and USASI COBOL binary source.

5.7.5 UCDD

USASI COBOL Option (NONUSASI)

Definition: See Section 5.7.3.

Option:		DELETE/	823	
	NONUSASI	EQU	0	NON-USASI COBOL version
	NONUSASI	EQU	1	USASI COBOL only version
Deck Card:	UCDD	DECK/	p_1, p_2	

Release Value: NONUSASI=0 on the UCBL COSY tape and USASI COBOL binary source.

5.7.6 UCR1

USASI COBOL Option (NONUSASI)

Definition: See Section 5.7.3.

Option:		DELETE/	121	
	NONUSASI	EQU	0	NON-USASI COBOL version
	NONUSASI	EQU	1	USASI COBOL only version
Deck Card:	UCR1	DECK/	p_1, p_2	

Release Value: NONUSASI=0 on the UCBL COSY tape and the USASI COBOL binary source.

5.7.7 UCP1

USASI COBOL Option (NONUSASI)

Definition: See Section 5.7.3.

Option:		DELETE/	67	
	NONUSASI	EQU	0	NON-USASI COBOL version
	NONUSASI	EQU	1	USASI COBOL only version
Deck Card:	UCP1	DECK/	p_1, p_2	

Release Value: NONUSASI=0 on the UCBL COSY tape and the USASI COBOL binary source.

5.7.8 UCG1

BDP Option (BDP3312)

Definition: The option BDP3312 allows the USASI COBOL compiler and object time routines to utilize either the non-enhanced or enhanced BDP hardware. The option BDP3312 contains four routines UCG1, UCBLSORT, UFIGCON and UVARAN.

Option:		DELETE/	479	
	BDP3312	EQU	1	non-enhanced BDP hardware
	BDP3312	EQU	2	enhanced BDP hardware
Deck Card:	UCG1	DECK/	p_1, p_2	

Dependency: Type of BDP hardware available to run USASI COBOL.

Release Value: BDP3312=1 on the UCBL COSY tape and the USASI COBOL binary source.

Floating Point Hardware Option (BDPMULDV)

Definition: The option BDPMULDV allows the compiler to generate or not to generate double precision instructions (MUAQ, DVAQ).

Option:		DELETE/	499	
	BDPMULDV	EQU	0	no floating-point hardware
	BDPMULDV	EQU	1	floating-point hardware
Deck Card:	UCG1	DECK/	p_1, p_2	

Dependency: Availability of floating-point hardware.

Release Value: BDPMULDV equals 0 on the UCBL COSY tape and the USASI COBOL binary source.

5.7.9 UCBLSORT

BDP Option (BDP3312)

Definition: The option BDP3312 allows the USASI COBOL compiler and object time routines to utilize either the non-enhanced or enhanced BDP hardware. The option BDP3312 is contained in four routines: UCG1, UCBLSORT, UFIGCON and UVARAN.

Option:		DELETE/	6	
	BDP3312	EQU	1	non-enhanced BDP hardware
	BDP3312	EQU	2	enhanced BDP hardware

Deck Card:	UCBLSORT	DECK/	p_1, p_2	

Dependency: Type of BDP hardware available to run USASI COBOL.

Release Value: BDP3312 =1 on the UCBL COSY tape and the USASI COBOL binary source.

5.7.10 UFIGCON

BDP Option (BDP3312)

Definition: See 5.7.9.

Option:		DELETE/	3	
	BDP3312	EQU	1	non-enhanced BDP hardware
	BDP3312	EQU	2	enhanced BDP hardware
Deck Card:	UFIGCON	DECK/	p_1, p_2	

Dependency: Type of BDP hardware available to run USASI COBOL.

Release Value: BDP3312=1 on the UCBL COSY tape and the USASI COBOL binary source.

5.7.11 UVARAN

BDP Option (BDP3312)

Definition: See 5.7.9.

Option:		DELETE/	3	
	BDP3312	EQU	1	non-enhanced BDP hardware
	BDP3312	EQU	2	enhanced BDP hardware
Deck Card:	UVARAN	DECK/	p_1, p_2	

Dependency: Type of BDP hardware available to run USASI COBOL.

Release Value: BDP3312=1 on the UCBL COSY tape and the USASI COBOL binary source.

5.7.12 ASSEMBLY OPTION CODING SHEETS

It is recommended that the coding sheets in Table II-9 be used as an aid in selecting options. Unused lines can be crossed out and parameters supplied for chosen lines. These coding sheets contain all options on the MSOS COSY tape.

TABLE II-9. USASI COBOL ASSEMBLY OPTION CODING SHEET

LOCATION	OPERATION	ADDRESS FIELD	COMMENTS
1 8	10	20	41
	DELETE/	10	
NONUSASI	EQU		TYPE OF USASI COBOL
UDCT	DECK/		
	DELETE/	436	
NONUSASI	EQU		TYPE OF USASI COBOL
UCIE	DECK/		
	DELETE/	823	
NONUSASI	EQU		TYPE OF USASI COBOL
UCDD	DECK/		
	DELETE/	121	
NONUSASI	EQU		TYPE OF USASI COBOL
UCR1	DECK/		
	DELETE/	67	
NONUSASI	EQU		TYPE OF USASI COBOL
UCP1	DECK/		
	DELETE/	479	
BDP3312	EQU		BDP OPTION
	DELETE/	499	
BDPMULDV	EQU		FLOATING POINT HARDWARE
UCG1	DECK/		
	DELETE/	6	
BDP3312	EQU		BDP OPTION
UCBLSORT	DECK/		
	DELETE/	3	
BDP3312	EQU		BDP OPTION
UFIGCON	DECK/		
	DELETE/	3	
BDP3312	EQU		BDP OPTION
UVARAN	DECK/		

5.7.13 UCBL COSY FILE CONTENTS

Table II-10 lists the COSY decks and the sequence in which they occur on the USASI COBOL COSY tape. The deck name of each routine in the list is suffixed with a special code. The definition of this code is as follows:

U Specified routine is used for USASI COBOL

Z Specified routine has a special assembly option.

TABLE II-10. USASI COBOL COSY TAPE

Deck Name	Code	Source Language	Product Name
UCDV	U	CMP	USASI COBOL COMPILER ROUTINES
UDCT	UZ	CMP	
UCIE	UZ	CMP	
UCDD	UZ	CMP	
UCR1	UZ	CMP	
UCP1	UZ	CMP	
UCR2	U	CMP	
UCAL	U	CMP	
UCP2	U	CMP	
UCG1	UZ	CMP	
UCFA	U	CMP	
UCDP	U	CMP	
UDP1	U	CMP	
UDP2	U	CMP	
UCBL	U	CMP	USASI COBOL LDR.
UTIMER	U	CMP	USASI COBOL OBJECT TIME ROUTINES
USYSTIME	U	CMP	
USYSDATE	U	CMP	
UOVER	U	CMP	
UCOBOLIO	U	CMP	
UCBWRITA	U	CMP	
UNSLABEL	U	CMP	
UCBLSORT	UZ	CMP	
UACCEPT	U	CMP	
UALPHA	U	CMP	
UANEDIT	U	CMP	
UBCDFP	U	CMP	
UBINFP	U	CMP	
UBIN2AN	U	CMP	
UCOMPARE	U	CMP	
UDEEDIT	U	CMP	
UDISPLAY	U	CMP	
UDIVIDE	U	CMP	
UEXAMINE	U	CMP	
UFIGCON	UZ	CMP	
UFPBCD	U	CMP	
UFPBIN	U	CMP	
UFP2FPE	U	CMP	
UMULT	U	CMP	
UMVG4095	U	CMP	
UODDEDIT	U	CMP	
USUBSC	U	CMP	

Deck Name	Code	Source Language	Product Name
UVARAN	U	CMP	USASI COBOL OBJECT TIME ROUTINES
UVARC1	U	CMP	
UVARC2	U	CMP	
UVARN	UZ	CMP	
UEXPONC	U	CMP	
UDBINCON	U	CMP	
UDFPCONV	U	CMP	
UDLOG	U	CMP	
UDNTILOG	U	CMP	
UDASMD	U	CMP	
UDCH1WRK	U	CMP	
UNUMERIC	U	CMP	

5.7.14 USASI COBOL BINARY SOURCE LISTINGS

One release tape is provided with this release of USASI COBOL. It includes the following sections:

Binary source file

COSY formatted source

List file

The following section contains the listings of the binary source file.

```
CN=00001     C.C.  SEQUENCE,914,USASI COBOL PRELIB SOURCE
CN=00002     C.C.  JOB,,,,
CN=00003     C.C.  PRELIB,,A,,,UC
CN=00004     EXS   PROMEM=START2
CN=00005     C.C.  RECORD
CN=00006     C.C.  ORIGIN,PROMEM
CN=00007     IDC  UCDV       PROG.LENGTH=04257
CN=00100     C.C.  RECORD
CN=00101     C.C.  ORIGIN,CDICT
CN=00102     IDC  DICT       PROG.LENGTH=02577
CN=00148     C.C.  RECORD
CN=00149     C.C.  ORIGIN,PROMEM,7056
CN=00150     IDC  UCIE       PROG.LENGTH=17143
CN=00299     C.C.  RECORD
CN=00300     C.C.  ORIGIN,PROMEM,7056
CN=00301     IDC  UCDD       PROG.LENGTH=24675
CN=00533     C.C.  RECORD
CN=00534     C.C.  ORIGIN,PROMEM,7056
CN=00535     IDC  UCR1       PROG.LENGTH=23541
CN=00731     C.C.  RECORD
CN=00732     C.C.  ORIGIN,PROMEM,7056
CN=00733     IDC  UCP1       PROG.LENGTH=26727          09/05/70
CN=01013     C.C.  RECORD
CN=01014     C.C.  ORIGIN,PROMEM,2567
CN=01015     IDC  UCR2       PROG.LENGTH=22115
CN=01160     C.C.  RECORD
CN=01161     C.C.  ORIGIN,PROMEM,2567
CN=01162     IDC  UCAL       PROG.LENGTH=13112
CN=01298     C.C.  RECORD
CN=01299     C.C.  ORIGIN,PROMEM,2567
CN=01300     IDC  UCP2       PROG.LENGTH=25761          09/10/70
CN=01637     C.C.  RECORD
CN=01638     C.C.  ORIGIN,PROMEM,2567
CN=01639     IDC  UCG1       PROG.LENGTH=27212          09/08/70
CN=02148     C.C.  RECORD
CN=02149     C.C.  ORIGIN,PROMEM,2567
CN=02150     IDC  UCFA       PROG.LENGTH=13110
CN=02264     C.C.  RECORD
CN=02265     C.C.  ORIGIN,PROMEM,2567
CN=02266     IDC  UCDP       PROG.LENGTH=06512
CN=02298     C.C.  RECORD
CN=02299     C.C.  ORIGIN,UCDPEND
CN=02300     IDC  UDP1       PROG.LENGTH=11062
CN=02448     C.C.  RECORD
CN=02449     C.C.  ORIGIN,UCDPEND
CN=02450     IDC  UDP2       PROG.LENGTH=11150
CN=02600     C.C.  FILE
CN=02601     C.C.  INSERT
CN=02602     IDC  UCBL       PROG.LENGTH=00605
CN=02624     IDC  UTIMER     PROG.LENGTH=00123
CN=02634     IDC  USYSTIME   PROG.LENGTH=00013
```

```
CN=02639     IDC   USYSDATE   PROG.LENGTH=00012
CN=02644     IDC   UOVER      PROG.LENGTH=01124
CN=02668     IDC   UCOBOLIO   PROG.LENGTH=00213
CN=02678     IDC   UCBWRITA   PROG.LENGTH=00365
CN=02690     IDC   UNSLABEL   PROG.LENGTH=00111
CN=02696     IDC   UCBLSORT   PROG.LENGTH=03604
CN=02762     IDC   UACCEPT    PROG.LENGTH=00305
CN=02776     IDC   UALPHA     PROG.LENGTH=00045
CN=02782     IDC   UANEDIT    PROG.LENGTH=00177
CN=02791     IDC   UBCDFP     PROG.LENGTH=00206
CN=02800     IDC   UBINFP     PROG.LENGTH=00021
CN=02804     IDC   UBIN2AN    PROG.LENGTH=00102
CN=02809     IDC   UCOMPARE   PROG.LENGTH=00102
CN=02815     IDC   UDEEDIT    PROG.LENGTH=00234
CN=02823     IDC   UDISPLAY   PROG.LENGTH=01027
CN=02850     IDC   UDIVIDE    PROG.LENGTH=01227
CN=02872     IDC   UEXAMINE   PROG.LENGTH=00161
CN=02879     IDC   UFISCON    PROG.LENGTH=00174
CN=02886     IDC   UFPBCD     PROG.LENGTH=00245
CN=02896     IDC   UFPBIN     PROG.LENGTH=00031
CN=02900     IDC   UFP2FPE    PROG.LENGTH=00341
CN=02916     IDC   UMULT      PROG.LENGTH=00375
CN=02925     IDC   UMVG4095   PROG.LENGTH=00176
CN=02933     IDC   UODDEDIT   PROG.LENGTH=00110
CN=02940     IDC   USUBSC     PROG.LENGTH=00040
CN=02945     IDC   UVARAN     PROG.LENGTH=00257
CN=02954     IDC   UVARC1     PROG.LENGTH=00056
CN=02959     IDC   UVARC2     PROG.LENGTH=00057
CN=02964     IDC   UVARN      PROG.LENGTH=00071
CN=02969     IDC   UEXPONC    PROG.LENGTH=00231
CN=02979     IDC   UDBINCON   PROG.LENGTH=00067
CN=02985     IDC   UDFPCONV   PROG.LENGTH=00020
CN=02991     IDC   UDLOG      PROG.LENGTH=00413
CN=03005     IDC   UDNTILOG   PROG.LENGTH=00306
CN=03016     IDC   UDASMD     PROG.LENGTH=00727
CN=03037     IDC   UDCH1WRK   PROG.LENGTH=00020
CN=03041     IDC   UNUMERIC   PROG.LENGTH=00046
CN=03046     C.C.   FILE
            EOF
            ENDSCOPE,R
            EOF

I PLIBEDIT 099 PLIBEDIT COMPLETED.
```

VERIFICATION DECK OUTPUT 6

```
SEQUENCE,123, MSSORT UNDER MSOS 4.2                    *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,MSSORT,,
FET,QA,VERIFICATION,512,01,QAQA
ALLOCATE,10,,S
FET,QA,INT1,512,01,QAQA,QAQA
ALLOCATE,10,,S
FET,QA,INT2,512,01,QAQA,QAQA
ALLOCATE,10,,S
```

```
SEQUENCE,123, MSSORT UNDER MSOS 4.0                                                        *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,MSSORT,,
EFT,QA,INT1,512,01,QAQA,QAQA
OPEN,01
EFT,QA,INT2,512,01,QAQA,QAQA
OPEN,02
COBOL,X
LOAD,56
RUN,,NM
EFT,QA,VERIFICATION,512,01,QAQA
OPEN,45,0
MSSORT
   0110111               5100100005
   1ADS01680512F      45
   1CIT01680512F      01QA      INT1                          01QAQAQAQA
   1DIT01680512F      02QA      INT2                          01QAQAQAQA
   19D001680512F      45QA      VERIFICATION                  01QAQA    S
   9ENDMSS
I MSRT 111  B       100  IN
I MSRT 114  B       100 OUT
I MSRT 115  B         1 SEQ
I MSRT 120  B   FINAL MERGE
I MSRT 111  B       100 IN
I MSRT 114  B       100 OUT
CLOSE,01
CLOSE,02
```

```
SEQUENCE,123, MSSORT UNDER MSOS 4.0                    *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,MSSORT,,
EFT,OA,VERIFICATION,512,01,QAQA
RELEASE,ALL
EFT,OA,INT1,512,01,QACA,QAQA
RELEASE,ALL
EFT,OA,INT2,512,01,QACA,QAQA
RELEASE,ALL
```

```
SEQUENCE,123, LISA UNDER MSOS 4.2                                    *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,ACCTNO,ID,5,NP,ND
FORTRAN,X
                    FORTRAN DIAGNOSTIC RESULTS  FOR    VERIFY

 NULL STATEMENT NUMBERS
       30
LOAD,56
RUN,5,NM
LISA/MSOS HAS BEEN INSTALLED ON THIS LIBRARY.

I FTNO 0060  STOP
```

```
SEQUENCE,123, UTILITY UNDER MSOS 4.2                    *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,HKE,8PL78,,ND,NP
UTILITY,60,61.
I UTIL 000 BEGIN UTILITY
COPYS,60,61,1.
SOS UTILITY IS INSTALLED ON THIS LIBRARY.
I UTIL 110       1 BCD RECORDS COPIED
T UTIL 108 EOF
I UTIL 011 TASK COMPLETED
SCOPE.
```

```
SEQUENCE,123, USASI FORTRAN UNDER MSOS 4.2                                  *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,,
FFT,MSOS,AUXFILE,500,UF
OPEN,20
UFORTRAN,X
```

```
USASI FORTRAN(1.1)/MSOS 4.2                               DATE    12/04/70      PAGE    001

  USASI FORTRAN DIAGNOSTIC RESULTS FOR VERIFY

      NO ERRORS
```

```
AUX,20
LOAD,56
RUN,,NM

USASI FORTRAN PRODUCT VERIFICATION COMPLETED
```

```
SEQUENCE,123, TAPE/SORT UNDER MSOS 4.2                                              *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,,,
EQUIP,10=MT,11=MT,12=MT
COBOL,X
LOAD,56
RUN,,NM
SORT
   01122211R 101112                          1100100005
   1ISI01680504F       CM12  01X
   1B 001680504F       CM      X
   9ENDSORT
 I TSRT 213 B   INTERNAL MERGE IP LOG UNITS
10,11.
 I TSRT 214 B   INTERNAL MERGE OP LOG UNITS
12.
 A TSRT 323 B   UNIT 12 FOR SORT IP.
 I TSRT 239 B   IB 126      OB 1974      G 392
 A TSRT 303 B   UNIT 12.    MOUNT SCRATCH.
 I TSRT 222 B       500  IN
 I TSRT 223 B       500 OUT
 I TSRT 232 B         2 SEQ
 I TSRT 236 B   FINAL MERGE
 A TSRT 305 B   UNIT 12.    MOUNT O/P TAPE.
 I TSRT 216 B   UNIT 12.    FINAL OUTPUT.  REEL  01.
 I TSRT 222 B       500 IN
 I TSRT 223 B       500 OUT
```

```
SEQUENCE,123, ALGOL UNDER MSOS 4.2                    *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,,,
ALGOL,X
                FINIS
```

```
ALGOL/MSOS(1.3)                 VERIFY.                          12/04/70    0000 HRS        PAGE   1***

                  LINE 0003    PROGRAM BEGINS                    (01)
                  LINE 0005    PROGRAM ENDS                      (01)
                  LINE 0005    SOURCE DECK ENDS                  (01)
LOAD,56
RUN,,NM
   CHANNEL,60=LU60,P80
   CHANNEL,61=LU61,P136,PP60
   CHANNEL,END
```

ALGOL PRODUCT VERIFICATION COMPLETED

```
END OF ALGOL RUN
```

```
SEQUENCE,123, COSY UNDER MSOS 4.2                              *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,,,
COSY
```

```
COSY V3.2      - MSOS V4.2     12/04/70
 VERIFY   DECK/     H,L
         THE PURPOSE OF THIS PROGRAM IS TO VERIFY THE PRESENCE OF THE
                   COSY PRODUCT ON THE MSOS LIBRARY.
 *
   COSY PRODUCT VERIFICATION COMPLETED
         ENDCOSY/
```

```
SEQUENCE,123, FORTRAN UNDER MSOS 4.2                                        *** MSOS V4.2 EDITION=UP DATE=12/04/70.
JOB,,,,,ND
FORTRAN,X

                    FORTRAN DIAGNOSTIC RESULTS  FOR    VERIFY

NO ERRORS
LOAD,56
RUN,,NM

 MS FORTRAN PRODUCT VERIFICATION COMPLETED
```

```
SEQUENCE,123, COMPASS UNDER MSOS 4.2                    *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,,,
COMPASS,X
```

```
                                        NUMBER OF LINES WITH DIAGNOSTICS         0
LOAD,56
RUN,,NM
COMPASS HAS BEEN INSTALLED ON THIS LIBRARY.
```

```
SEQUENCE,123, COBOL UNDER MSOS 4.2                    *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,,,
COBOL,X
LOAD,56
RUN,,NM

 MS COBOL PRODUCT VERIFICATION COMPLETED
```

```
SEQUENCE,123, USASI COBOL UNDER MSOS 4.2                                          *** MSOS V4.2 EDITION=UF DATE=12/04/70.
JOB,,,,
UCBL,X
```

```
USASI COBOL 2.1/MSOS 4.2              USASI-COBOL-VERIFICATION              12/04/70         PAGE     1

              MPUT
              MOPENF
              MCLOSEF
              ABNORMAL


    0 ERRORS IN USASI-CO
LOAD,56
RUN,,NM

 USASI COBOL PRODUCT VERIFICATION COMPLETED
```

SYSTEM SCRATCH FILES 7

7.1 ALLOCATION OF FILES 54, 55 AND 56

Table II-11 defines the system scratch file usage.

Users of 863 interim library should release and reallocate files 54 and 56 on the far extremity of the drum unit to provide the contiguous space necessary for PRELIB execution. (See example 3 below.) Other users may use the same procedure. Scratch files 54 and 55 and load-and-go file 56 are not restricted to the system device. The placement and size of these files is installation-dependent. As installed, files 54, 55, and 56 immediately follow MSOS system files, and are allocated for 1 track each.

Since files 54 and 56 can be on any available mass storage device maintained by the system, PRELIB cannot arbitrarily release and reallocate these files. Before a PRELIB run, the user should release any of these files that occur on the system device and restore them after the PRELIB run. If file 54, 55, or 56 occurs on a device other than the system device, it need not be released prior to a PRELIB run as long as enough disk space remains to satisfy the requirements of PRELIB.

The following are examples of jobs that release and reallocate files 54 to 56:

1. Assume that files 54, 55, and 56 are on the system pack. The following job releases them and allocates file 55 on a scratch pack for use by PRELIB. The scratch pack is 854/2.

   $^{7}_{9}$SEQUENCE,001

   $^{7}_{9}$JOB,,,

   $^{7}_{9}$RAT,854/2

   $^{7}_{9}$CLOSE,54

   $^{7}_{9}$FET,MSOS,FILE54,480,00,0000,0000†

   $^{7}_{9}$RELEASE,ALL

   $^{7}_{9}$ALLOCATE,1

   $^{7}_{9}$CLOSE,56

   $^{7}_{9}$FET,MSOS,FILE56,960,00,0000,0000†

   $^{7}_{9}$RELEASE,ALL

---

† If the master privacy codes were changed, the 0000,0000 fields must be changed accordingly.

${}^{7}_{9}$ALLOCATE,1

${}^{7}_{9}$CLOSE,55

${}^{7}_{9}$FET,MSOS,FILE55,480,00,0000,0000†

${}^{7}_{9}$ALLOCATE,30

${}^{77}_{88}$end-of-file

2. After the deck and PRELIB have run, assume the above environment. This example allocates files 54 and 56 on the system pack and leaves file 55 on scratch device 854/2. This job assumes the device number of the system device is 8541.

${}^{7}_{9}$SEQUENCE,002

${}^{7}_{9}$JOB,,,

${}^{7}_{9}$RAT,854/8541

${}^{7}_{9}$CLOSE,54

${}^{7}_{9}$FET,MSOS,FILE54,480,00,0000,0000†

${}^{7}_{9}$RELEASE,ALL

${}^{7}_{9}$ALLOCATE,50,991230

${}^{7}_{9}$CLOSE,56

${}^{7}_{9}$FET,MSOS,FILE56,960,00,0000,0000†

${}^{7}_{9}$RELEASE,ALL

${}^{7}_{9}$ALLOCATE,130,991230

${}^{77}_{88}$end-of-file

† If the master privacy codes were changed, the 0000,0000 fields must be changed accordingly.

3. The following job releases the system scratch files and reallocates them on the most extreme tracks.

$^{7}_{9}$SEQUENCE, 001

$^{7}_{9}$JOB,,,

$^{7}_{9}$CLOSE, 54

$^{7}_{9}$FET, MSOS, FILE54, 480, 00, 0000, 0000

$^{7}_{9}$RELEASE,ALL

$^{7}_{9}$CLOSE, 55

$^{7}_{9}$FET, MSOS, FILE55, 480, 00, 0000, 0000

$^{7}_{9}$RELEASE, ALL

$^{7}_{9}$CLOSE, 56

$^{7}_{9}$RAT, 854/8541

$^{7}_{9}$FET, MSOS, FILE56, 960, 00, 0000, 0000

$^{7}_{9}$RELEASE, ALL

$^{7}_{9}$FET, DUMMY, TEMPORARY, 4096

$^{7}_{9}$ALLOCATE, xxx, 000001†

$^{7}_{9}$FET, MSOS, FILE55, 480, 00, 0000, 0000

$^{7}_{9}$ALLOCATE, 30

$^{7}_{9}$OPEN, 55

$^{7}_{9}$FET, MSOS, FILE54, 480, 00, 0000, 0000

$^{7}_{9}$ALLOCATE, 1

† xxx = the number of free tracks beyond the MSIO and library files minus 2 tracks for files 54 and 56.

$^{7}_{9}$OPEN,54

$^{7}_{9}$FET,MSOS,FILE56,960,00,0000,0000

$^{7}_{9}$ALLOCATE,1

$^{7}_{9}$OPEN,56

$^{7}_{9}$FET,DUMMY,TEMPORARY,4096

$^{7}_{9}$RELEASE,ALL

$^{77}_{88}$end-of-file

7.2 UTILIZATION OF SYSTEM SCRATCH FILES

TABLE II-11. SYSTEM SCRATCH FILE USAGE

	FILE54	FILE55	FILE56
COSY	Used as default for H parameter on DECK/card	Used only if additional space is required for corrections	
COMPASS†		Work area	Used only if X is specified on $^{7}_{9}$COMPASS card
MS COBOL	Work area	Work area	Used only if X is specified on $^{7}_{9}$COBOL card
UCBL	Default value for S parameter on $^{7}_{9}$UCBL card	Work area	Used only if X is specified on $^{7}_{9}$UCBL card
MS FORTRAN		Work area	Work area; Load and go, if X specified
UFTN		Work area	Used only if X is specified

† During assembly of PRELOAD, FILE55 must be at least 250 tracks. All system scratch files may be segmented and need not reside on the system device.

COMPASS OVERLAY PREPARATION 8

Figure II-4 and II-5 illustrate the layout of core and deck structures for the COMPASS assembler. Whenever a $^{7}_{9}$COMPASS card is encountered, COMPASSB is loaded in high core. COMPASSB loads COMPASS (the main subprogram) and then passes control to COMPASS. COMPASS controls the assembly process and calls the overlays, OVERLAY1, PASSONE, PASSTWO, and CRT for each deck assembled.

COMPASS also contains the common routines available to each of the overlays. These are linked to the overlays by the BSS blocks which appear at the beginning of each deck. Further, each overlay uses BSS blocks to line up labels internal to the overlay with the location of the same label in COMPASS.

When any overlay is loaded, it should be loaded above COMPASS and FTN0 without disturbing either of these two routines. PRELIB provides the facility to create an absolute routine which will load above another without destroying the routine already in core. By placing an offset parameter on the $^{7}_{9}$ORIGIN card when creating the absolute file, a deck can be loaded higher than it would normally be loaded. To determine this offset parameter for COMPASS and each of the overlays, find the relative address (from the listings) of the first instruction other than a BSS pseudo-instruction. These octal values should be placed on the $^{7}_{9}$ORIGIN cards for each deck of the PRELIB run. The offset parameters for OVERLAY1, PASSONE, PASSTWO, and CRT should be the same.

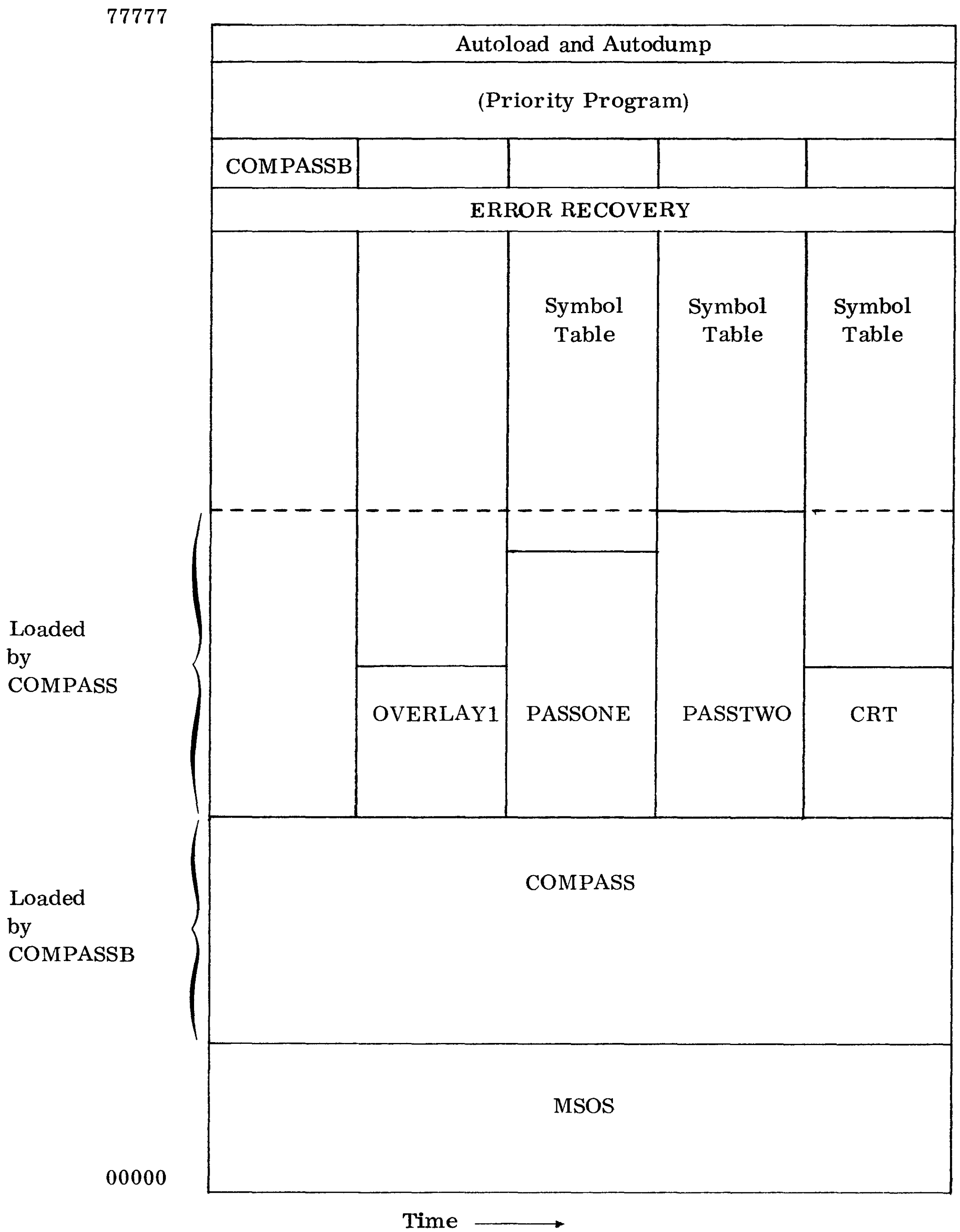


Figure II-4. Contents of Core During COMPASS Assembly

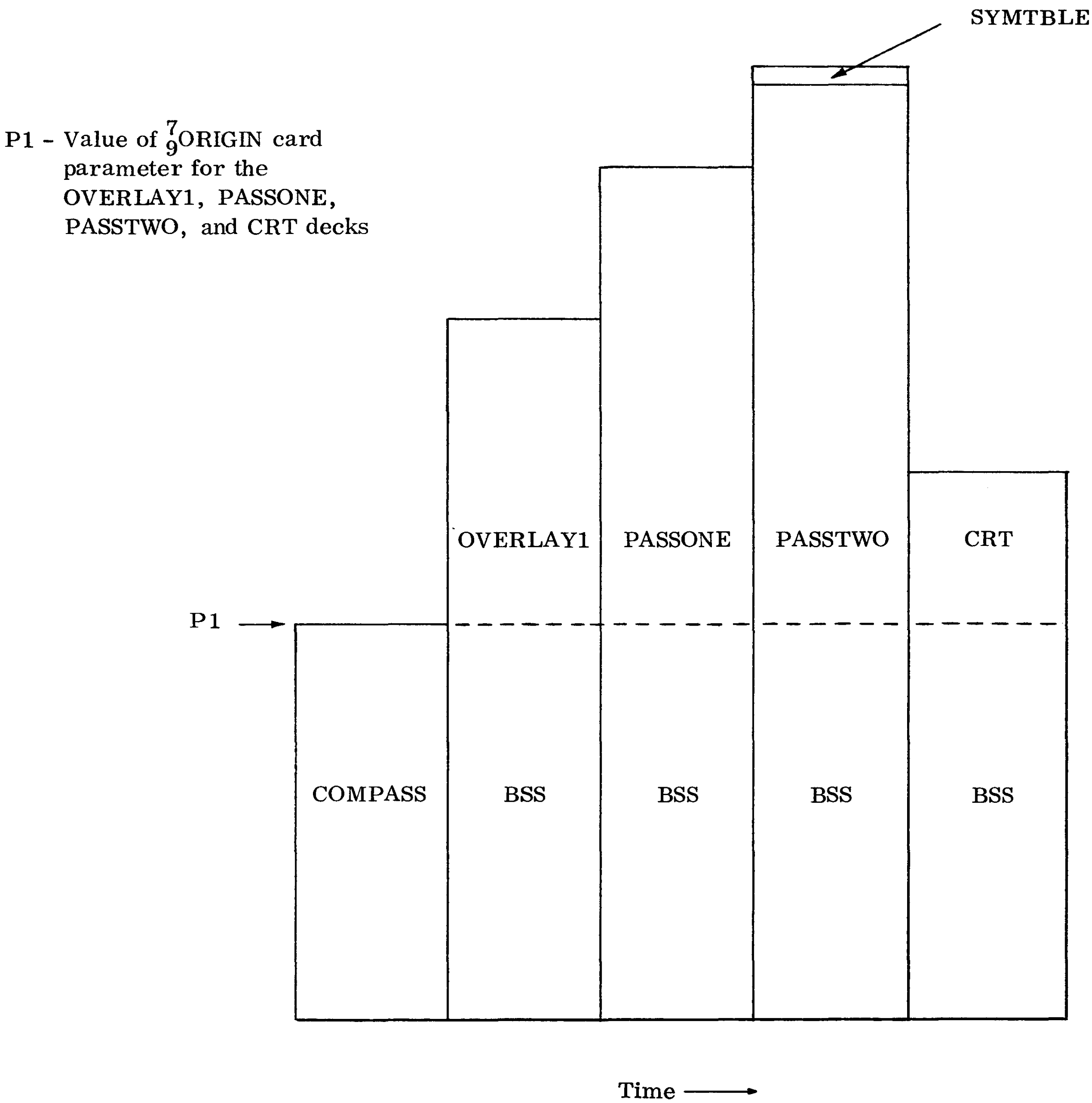


Figure II-5. COMPASS Overlay Deck Structure

When loading a FORTRAN absolute segment into core, it is necessary to preserve a common area to link routines from the previous overlay. PRELIB uses an increment parameter on the ORIGIN card when creating the absolute file of a library as shown in Figure II-6. In order to preserve the correct routines for the FORTRAN compiler, it is expedient to note the proper address for the increment for each absolute segment.

The BSS blocks for use as registers and PICKBUF (buffer for blocked universal input) must remain in core throughout the entire compilation process. The origin for FTN1 will immediately follow these two blocks.

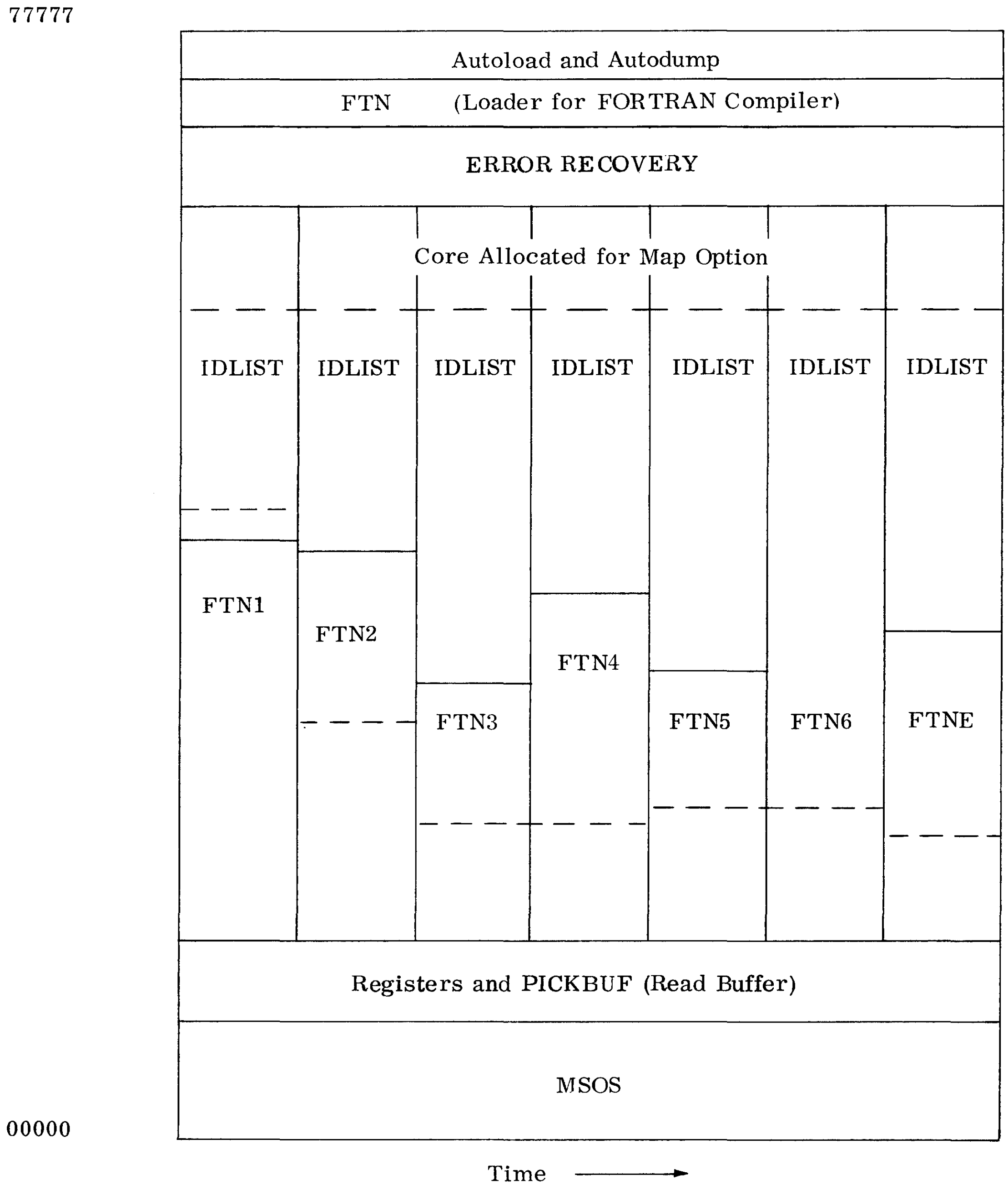
FTN2 uses the READ and SCAN routine in FTN1; therefore it is necessary for the origin in FTN2 to coincide with the address of the declarative processor (DEPROS) in FTN1.

In addition to some I/O routines carried through from FTN1 and FTN2, FTN3 uses six octal blocks as counters. These blocks, ending with NEXTWORD, immediately precede the origin point for FTN3. Since FTN4 uses the same six indicators, the origin for FTN4 should be identical to the origin for FTN3.

For FTN5 and FTN6, in addition to the octal blocks for FTN3 and FTN4 and a BSS of 3 for FTN4 linkage, a BSS block of 47 precedes the origin point. This is a temporary indexing block so the origins for FTN5 and FTN6 should be identical and should follow the 47-word BSS block. The tag on this BSS block is NTST in FTN5 and ZTEMPINX in FTN6.

The additional BSS blocks mentioned for FTN3 through FTN6 are not needed for FTNE (error processor). Therefore the origin for FTNE should immediately follow the message block titled AMESS which is a BSS 6.

The FORTRAN loader (FTN) remains in core throughout the entire compilation process and loads each of the absolute segments. IDLIST (identifier list), which is formed by FORTRAN, follows each overlay in core. The initialized portion of the FORTRAN loader is now located at the end of FTN1 and is overlaid by the first portion of IDLIST. If the map option parameter is requested, the area reserved for storing variables starts just below ERROR RECOVERY and utilizes the core space between ERROR RECOVERY and the end of IDLIST.



NOTE

Dotted lines in the FTN2 - FTNE overlay area indicate the amount of core that is not overlayed (i.e., the relative address of the increment for the origin card).

Figure II-6. Contents of Core During a FORTRAN Compilation

FCO LEVELS 10

MSOS Version 4.2 and product set and all subsequent additional products have been tested on the following configurations on which the following field change orders were incorporated. Enhanced BDP hardware was available in these configurations.

Product	Equipment	Product Designation	FCO Level
Control Data 3304A Basic Processor		A24	
Control Data 3304B Basic Processor		B07	
Control Data 3311 Multiprogramming Module		A12	
Control Data 3302 Storage Module		A16	
Control Data 3309 Storage Module		A10	
Control Data 3306 Communication Channel		A09	
		B04	
		C04	
Control Data 3307 Communication Channel		A12	
Control Data 3310 Floating Point Module		A09	
Control Data 3312 Business Data Processing Module		A16	
Control Data 3501 Console (for 3504)	CR105	A09	
Control Data 3501 Console (for 3514)	CR106	A07	
Control Data 3502 Storage Module			
(1st 16K with cabinet)	BB314	A05	
(2nd 16K)	BB313	A03	
(refrigeration unit)	GH104	A03	
Access channel)	AT105	A01	
Control Data 3504 Processor (includes floating point, page file and BDP)	AC104	A17	
Control Data 3507 Communication Channel	DC111	A02	
Control Data 3514 Basic Processor			
(includes floating point)	AC106	A13	
(page file)	AT124	A03	
(BDP)	AF102	A04	
(real-time processor)	AT122	A05	
(add interrupt lines)	AT123	A01	
Control Data 3170 Storage Module	(BB105-A)	PD(A01)	

Product	Equipment	Product Designation	FCO Level
Control Data 3228 Magnetic Tape Controller		A10	
		B02	
Control Data 3423 Magnetic Tape Controller		A09	
		B02	
Control Data 3518 Magnetic Tape Controller	FA416B	B18	
	FA417B	B04	
	FA420A	B03	
Control Data 3232 Storage Drive Controller		A05	
Control Data 3234 Disk Storage Controller		A11	
		B05	
Control Data 3637 Drum Storage Controller		A09	
		B05	
		C03	
Control Data 3553-2 Mass Storage Controller		A12	CA26675
		B10	
Control Data 3256 Line Printer Controller		A11	
		B07	
		C04	
Control Data 3659 Line Printer Controller		A14	
		B04	
		C02	
Control Data 3446 Card Punch Controller		A12	
		B01	
Control Data 3644 Card Punch Controller		A16	
		B03	
Control Data 3447 Card Reader Controller		A16	
		B04	
Control Data 3649 Card Reader Controller		A24	
Control Data 3290-2 Display Controller		D03	CD03353
Control Data 211 Display		P01	

PART III

RELEASE AIDS

INSTALLATION AUTOLOAD ROUTINE 1

Each installation must manually establish an autoload routine in the autoload region of core. This routine is a permanent machine entry. It initiates IUP. This routine in conjunction with an autoload card, autoloads the interim and final MSOS libraries.

The format of the installation autoload routine follows:

Location	Instruction	Comment
xxx40	006xxx52	If select jump 6 is set, load card
xxx41	770ce000	Connect channel c, equipment e, tape unit 0
xxx42	010xxx41	Reject
xxx43	771c0010	Rewind channel c
xxx44	010xxx43	Reject
xxx45	740xxx70	Read first tape record into location 00000
xxx46	c0000000	
xxx47	010xxx45	Reject
xxx50	77770000	Stop
xxx51	01000000	Go to program, location 00000
xxx52	770ce000	Connect card reader unit 0, channel c, equipment e
xxx53	010xxx52	Reject
xxx54	740xxx37	Read first record into location 00000
xxx55	c0000000	
xxx56	010xxx54	Reject
xxx57	010xxx50	Go to stop instruction

c Channel number appropriate to equipment

e Equipment number appropriate to unit

xxx is dependent on machine size and type as follows:

Memory	Type	xxx
16K	3100/3150	376
32K	3100/3150	776
16K	3200/3300	377
32K	3200/3300	777

The autoload procedure autoloads from tape unit 0 or from the card reader if SELECT JUMP 6 is set. It halts at location xxx50 to allow operation to complete. There is not enough space to check status. When device motion stops, press GO to continue.

PAGE INDEX FILE SETTING 2

MP MSOS users with the 3311 Multiprogramming Option must set the page index file to reflect contiguous memory.

1. Press: MC (master clear)
2. Press: ENTER
3. Enter the following instructions:

Address	Instruction
00000	14600074
00001	14200017
00002	77644000
00003	15477773
00004	02600002
00005	14604000
00006	14200020
00007	77644000
00010	10200177
00011	01000007
00012	00000000

4. Press: KYBD OFF (keyboard off)
5. Press: MC (master clear)
6. Press: EXECUTIVE MODE
7. Press: GO

The machine halts with P = 00000012 and F = 00000000.

8. Press: EXECUTIVE MODE

MASS STORAGE INITIALIZATION 3

3.1 853/854

The write address and clear operations occur simultaneously through the interaction of the 3234 Controller and the particular mass storage unit. The CPU is not used.

1. Ready disk pack.
2. Turn key lock switch to ON.
3. If disk unit n is the unit to address and clear, interchange the wire leads so that physical unit n corresponds to logical unit 00.
4. Set HEADER and DATA switches to WRITE.
5. Set DEVICE SELECT to DISK PACK.
6. Press FILE/CELL.
7. Set REGISTER SELECT on DATA and press MC.
8. Set REGISTER SELECT on LOWER address and press MC.
9. Set REGISTER SELECT on UPPER address and press MC.
10. Press SEEK ADRS (writes addresses and clears the pack simultaneously).
11. When writing is complete (upper address is 313_8 on 853/854), restore the wire leads to the proper connections, clear FILE/CELL switch, reset HEADER and DATA switches to READ and turn the key lock switch to OFF.

3.2 841

Address headers for the 841 are written from the 3553 controller maintenance panel. In order to write the header the following procedure should be performed:

1. Ready the disk pack.
2. Turn the maintenance panel key switch to on.
3. Set the DEVICE SELECT switch to the desired device type.
4. Select the desired ending mode by pressing the appropriate mode switch (TRACK, FILE SELECTED, etc.) and the SELECT switch.
5. Set the INTERLACE switch to 2:1.

6. Press MASTER CLEAR.

7. If addressing the whole pack, skip to step 9. Otherwise, select the appropriate portion of the address register via the REGISTER SELECT switch

8. Enter the starting addresses into the upper and lower portions of the address register.

9. If there is no usable data in the sectors where new headers are to be written, both the HEADER and DATA switch should be set to the WRITE position. If there is useful data in the sectors and only headers are to be written, the HEADER switch must be set to the WRITE position and the DATA switch to the READ position.

NOTE

Do not use the REPEAT mode switch when writing headers. The REPEAT switch must be OFF.

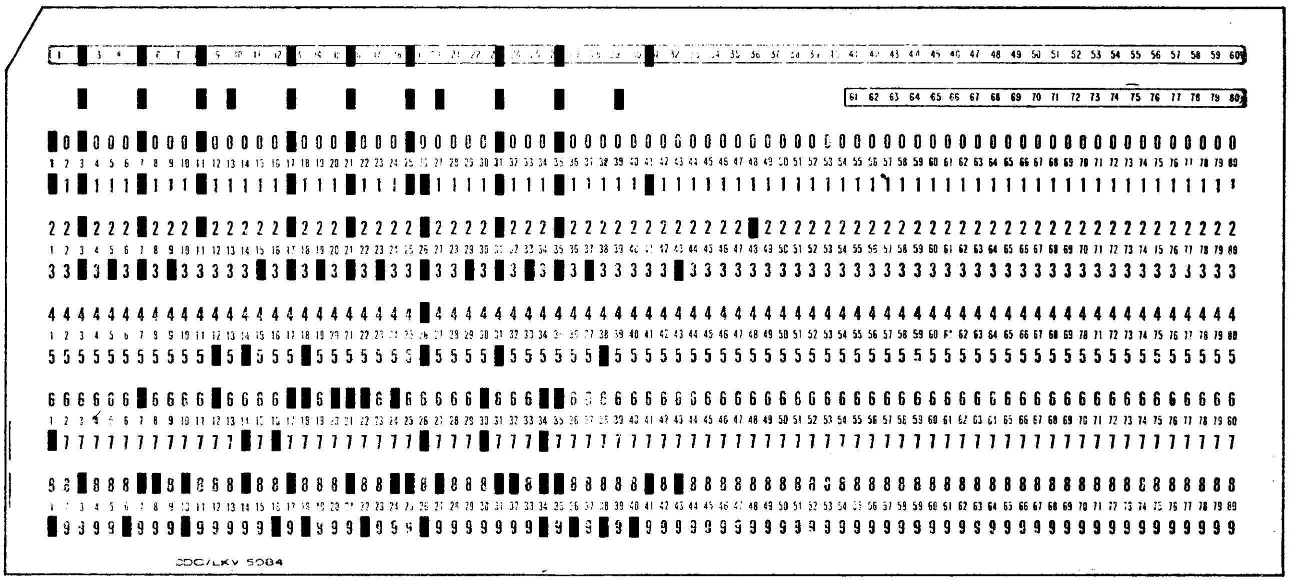
10. Press the SEEK ADRS. switch.

11. The seek address operation ends when the address of the last sector of the selected storage area has been written, or an abnormal condition is displayed in the status register. Reset HEADER and DATA switches to READ.

4 AUTOLOAD CARD PREPARATION

To autoload both the interim library and the final MSOS library, each installation must punch an autoload card. The autoload card is used in conjunction with the autoload routine. The cards are punched in column binary format. See Autoload Card Generation (AGEN), Part III, Section 5.2.8.

4.1 852 SAMPLE CARD



Machine Code	Mnemonic	
14050000	NOP	50000B
770ce000	CON	e000B, c
01000001	UJP	1
771c0002	SEL	2, c
01000003	UJP	3
76000030	OUTW	c,26B, 30B
c0000026		
01000005	UJP	5
771c0031	SEL	31B, c
01000010	UJP	10B
771c0011	SEL	11B, c
01000012	UJP	12B
74020765	INPW	c, 20000B, 20765B
c0020000		
01000014	UJP	14B
772c0002	EXS	2, c
01000017	UJP	17B
771c0001	SEL	1, c
01000021	UJP	21B
20000001	LDA	1
44020000	SWA	20000B
01020000	UJP	20000B
00000000	OCT	0
00000200	OCT	200

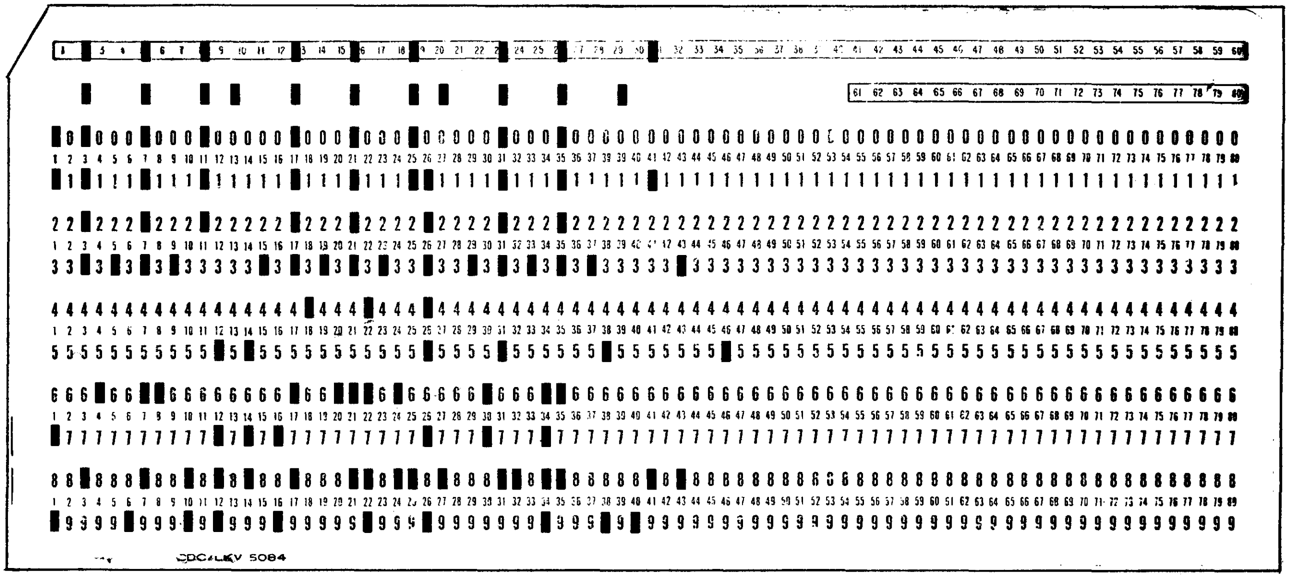
The lower case characters represent octal digits which must be supplied by each installation

c Channel of disk drive containing MSOS library

e Equipment of disk drive containing MSOS library

The values of c and e are 0 on the card shown.

4.2 853/854 SAMPLE CARD



	Machine Code	Mnemonic	
00	14050000	NOP	50000B
01	770ce010	CON	e010B, c
02	01000001	UJP	1
03	771c0010	SEL	10B, c
04	01000003	UJP	3
05	76000027	OUTW	c, 26B, 27B
06	c0000026		
07	01000005	UJP	5
10	771c0040	SEL	40B, c
11	01000010	UJP	10B
12	771c0053	SEL	53B, c
13	01000012	UJP	12B
14	74020765	INPW	c, 20000B, 20765B
15	c0020000		
16	01000014	UJP	14B
17	772c0002	EXS	2, c
20	01000017	UJP	17B
21	771c0000	SEL	0, c
22	01000021	UJP	21B
23	20000001	LDA	1
24	44020000	SWA	20000B
25	01020000	UJP	20000B
	00000020	OCT	20

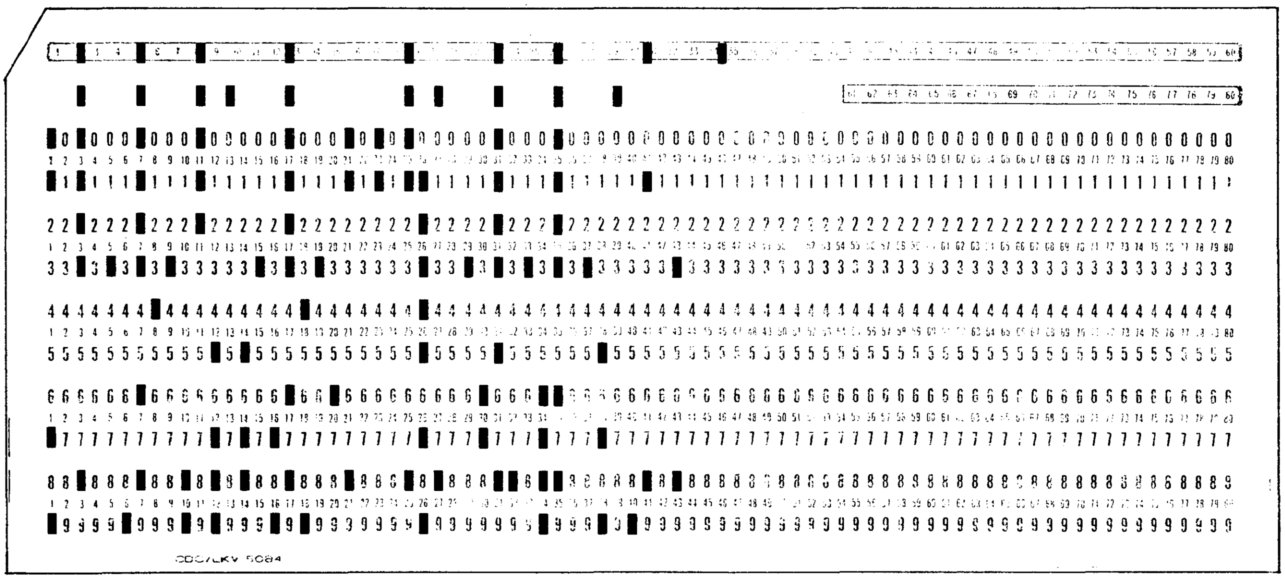
The lower case characters represent octal digits which must be supplied by each installation

c Channel of the disk drive containing the MSOS library

e Equipment number of the disk drive containing the MSOS library

The values of c and e are 0 on the card shown.

4.3 863 SAMPLE CARD



Machine Code	Mnemonic	
14050000	NOP	50000B
770ce000	CON	e000B,c
01000001	UJP	1
771c0040	SEL	40B,c
01000003	UJP	3
76000027	OUTW	c, 26B, 27B
c0000026		
01000005	UJP	5
771c0041	SEL	41B,c
01000010	UJP	10B
14000000	NOP	
14000000	NOP	
74020765	INPW	c,20000B,20765B
c0020000		
01000014	UJP	14B
772c0002	EXS	2,c
01000017	UJP	13B
771c0000	SEL	0,c
01000021	UJP	21B
20000001	LDA	1
44020000	SWA	20000B
01020000	UJP	20000B
00004000	OCT	4000

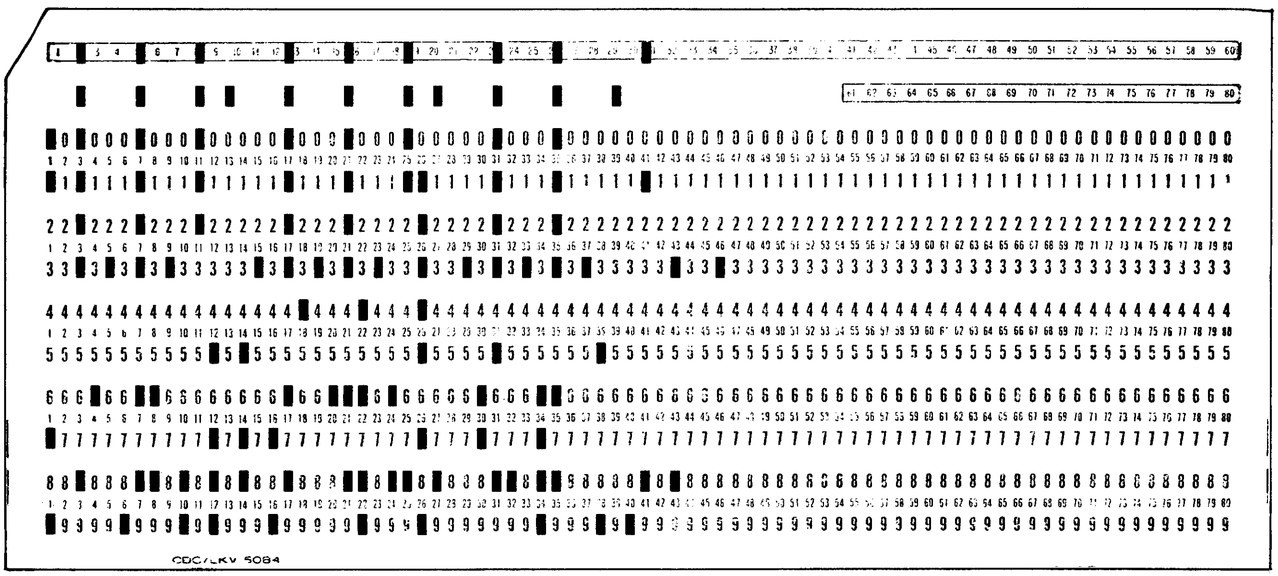
The lower case characters represent octal digits which must be supplied by each installation

c   Channel of drum containing MSOS library

e   Equipment of drum containing MSOS library

The values of c and e are 0 on the card shown.

4.4 841 SAMPLE CARD



Machine Code	Mnemonic	
14050000	NOP	50000B
770ce010	CON	c010B, c
01000001	UJP	1
771c0012	SEL	12B, c
01000003	UJP	3
76000027	OUTW	c, 26B, 27B
c0000026		
01000005	UJP	5
771c0040	SEL	40B, c
01000010	UJP	10B
771c0053	SEL	53B, c
01000012	UJP	12B
74020765	INPW	c, 20000B, 20765B
c0020000		
01000014	UJP	14B
772c0002	EXS	2, c
01000017	UJP	17B
771c0000	SEL	0, c
01000021	UJP	21B
20000001	LDA	1
44020000	SWA	20000B
01020000	UJP	20000B
00000100	OCT	100

The lower case characters represent octal digits which must be supplied by each installation

c Channel of the disk drive containing the MSOS library

e Equipment number of the disk drive containing the MSOS library

The values of c and e are 0 on the card shown.

MSOS INSTALLATION UTILITY PACKAGE 5

The MSOS Installation Utility Package is the initialization routine for the MSOS 4.2 system. IUP is a set of routines which perform the following functions:

Equips devices

Changes source of control statements

Inspects mass storage devices

Changes information on mass storage

Rewinds magnetic tape

Skips file forward on magnetic tape

Initializes 512 and MMTC image memory

Generates autoload cards

Generates one-card loaders

Dumps a library onto magnetic tape or cards in special format

Installs a library from magnetic tape or cards of special format

5.1 MSOS IUP OPERATION

To operate MSOS IUP, establish an internal autoload routine (Part III, Section 1) and autoload IUP into memory. MSOS IUP is released on tape in binary format or as binary cards preceded by a two-card loader and followed by an EOF card. The two-card loader is a bootstrap routine that loads MSOS IUP and terminates upon encountering an end-of-file.

When IUP is loaded, it types:

I IUP 033 INSTALLATION UTILITY PROGRAM LOADED

Then IUP=LUN is typed to allow the input unit to be designated (Part III, Section 5.2.2).

5.2 MSOS IUP CONTROL STATEMENTS

IUP accepts control statements from the typewriter, card reader, or magnetic tape. Card images have one control statement per card. Leading, embedded, and trailing blanks are ignored.

For typewriter input, IUP types:

A IUP 006 READY FOR INPUT

IUP is now ready to accept the 3 or 4 character control statement name. Parameters are then requested. When IUP is accepting a parameter and REPEAT is pushed, IUP will:

Either repeat the present request if some character have already been entered for this request, or

Repeat the previous request if no characters have been entered.

Using this feature, it is possible to return to the beginning of a control statement or to an erroneous parameter by hitting repeat until the desired request is reached.

5.2.1 DEV STATEMENTS

DEV statements define equipment available to IUP and assign logical unit numbers. The first statement entered must be a DEV statement. Otherwise, an error occurs. It is good practice to enter all DEV statements first because some operational control statements may cause device drivers of unequipped devices to be destroyed.

IUP allows up to 64 devices to be equipped.

Card input:

DEV, lun, ht, c, e, uu, $\left\{\begin{matrix}\text{dt}\\ \text{cdt}\end{matrix}\right\}$, [C], [W], [E],

Typewriter input:

DEV LUN lun HT ht C c E e UU uu $\left\{\begin{matrix}\text{DT dt}\\ \text{CDT cdt}\end{matrix}\right\}$, $\left[\text{CLR} \left\{\begin{matrix}\text{Y}\\ \text{N}\end{matrix}\right\}\right] \left[\text{WA} \left\{\begin{matrix}\text{Y}\\ \text{N}\end{matrix}\right\}\right] \left[\text{EXT} \left\{\begin{matrix}\text{Y}\\ \text{N}\end{matrix}\right\}\right]$

lun — Logical unit number, $1 \leq$ lun ≤ 63; unique for each equipment entered. All references to the device are in terms of the logical unit number.

ht — Hardware type

MT	Magnetic tape
DP	Disk pack
CP	Card punch
CR	Card reader
TY	Console typewriter
PR	Printer
DR	Drum
DF	Disk file

c — Channel number

e — Equipment number of controller

u — Unit number

dt — Unit device type if hardware type is MT, PR, DP, DF, or DR

cdt — Controller device type if hardware type is CR or CP

C — Write zeros on mass storage device (card format)

CLR — Write zeros on mass storage device (typewriter request). Respond Y for yes, N for no.

W — Write addresses on 852 or 1311 (card format)

WA — Write addresses on 852 or 1311 (typewriter request). Respond Y for yes, N for no.

E Extend mass storage on 813 or 814 (card format)

EXT Extend mass storage on 813 or 814 (typewriter request). Respond Y for yes, N for no.

C, W, and E may appear in any order on the card. No error occurs if write addresses or extend mass storage requests are for the wrong device type. The request is ignored.

5.2.2 INP STATEMENT

The INP statement allows the user to chagne the source of input to a designated unit. The input unit may be the console typewriter, the card reader, or magnetic tape. The INP statement also equips the unit when necessary.

Card input:

$$\text{INP,}\left\{\begin{matrix}\text{TY}\\ \text{lun}\end{matrix}\left[\text{, ht }\left[\text{,c,e,uu,, }\left\{\begin{matrix}\text{dt}\\ \text{cdt}\end{matrix}\right\}\right]\right]\right\}$$

Typewriter input:

$$\text{INP LUN}\left\{\begin{matrix}\text{TY}\\ \text{lun}\end{matrix}\right\}\text{HT ht C c E e UU uu}\left\{\begin{matrix}\text{DT dt}\\ \text{CDT cdt}\end{matrix}\right\}$$

lun Logical unit number of new control statement input device

TY indicates typewriter

HT Hardware type

c Channel number of input device

e Equipment number of controller

uu Unit number

dt Device type if hardware type equals MT

cdt Controller device type if hardware type is CR

Examples:

INP, 60, CR 1, 0, 00, 3248

This statement equips logical unit 60 to the card reader and designates it as the source input unit.

INP, 60, CR (If typewriter request, press FINISH following the C request.)

This statement allows the unit last equipped with the same hardware type to be assigned logical unit 60. Input is accepted from this unit

INP, 60 (If typewriter input, press FINISH following the HT request.)

This statement changes source input to logical unit 60, but logical unit 60 must be previously equipped.

5.2.3 INSP STATEMENT

The INSP statement allows the inspection of information on mass storage. The output can be by number of sectors or by number of words. The beginning location of the dump may be any sector or word within a sector. The console typewriter, printer, or magnetic tape can be designated for output, but the typewriter should only be used for samll inspections.

Card input:

INSP, lun1, lun2, s, b, n

Typewriter input:

INSP FROM lun1 TO lun2 SECTOR s BEGIN b COUNT n

lun1	Logical unit number of mass storage device to be inspected	
lun2	Logical unit number of device where information is to be output	
s	Beginning sector on mass storage device	
b	Numeric	Indicates the beginning word index of the beginning sector
	or S	Indicates that parameter n is the number of sectors dumped rather than the number of words
n	Number of words or sectors to be listed depending on b 60,	

Sector numbers for the INSP and CHNG control statements must be calculated using 1 to indicate the first sector of the mass storage device. Thus, all sector addresses given on an FLD listing (which uses 0 for the first sector) must be increased by one for use in these two statements.

5.2.4 CHNG STATEMENT

CHNG statements permit the changing of information on a mass storage device.

Card input:

CHNG, , lun, s, b, w1, [w2...wn]

Typewriter input:

CHNG LUN lun SECTOR s BEGIN b WORD w1 [WORD w2 ... WORD wn]

lun	Logical unit number of mass storage device
s	Sector number on mass storage device
b	Beginning word index
w_i	List of words to be written consecutively beginning at relative word 6. On card images this information can only continue to the end of card. When using the typewriter, information is requested until FINISH is hit and no new characters are input. The data should be in octal numbers. If eight digits are not entered, the information is right justified with leading zeros.

5.2.5 INIT STATEMENT

The INIT statement directs the initialization of the MMTC or 512 image memory. Initialization can use a standard or a non-standard image.

Card input:

INIT, lun1, n, lun2

Typewriter input:

INIT LUN lun1 IMAGE n LUN lun2

lun1	Logical unit number of device to initialize		
n	For 512, the train number		
		1	501 compatible
		2	AN compatible
		3	HN compatible
		4	595-2 train image
		Any other character indicates non-standard image	
	For MMTC, image number		
		blank, 0, 1	ASCII standard image
		other	non-standard
lun2	Logical unit number for input of non-standard image memory (CR or MT)		

Non-standard images must be entered in 80 character records from the card reader or magnetic tape. The card contains the octal digits for each conversion code, separated by commas. Blanks are ignored and each code must contain 1-4 octal digits. If less than 4 characters are entered, leading zeros are added.

The 512 initialization requires 288 conversion codes for the image memory. Illegal codes produce error. The MMTC conversion memory consists of two parts:

A 32-word section used for output conversion, and

A 128-word section used for input conversion.

However, IUP requires only the 64 codes for the 32-word output conversion section. This input is used to generate the full image.

Example:

The following is an example of EBCDIC conversion codes used to load a non-standard EBCDIC image memory for MMTC.

```
        343,344,345,346,347,350,351,117,153,115,155,173,120
      326,327,330,331,320,133,120,166,155,156,100,141,342
    300,113,135,340,137,136,140,321,322,323,324,325
  177,114,74,175,105,301,302,303,304,305,306,307,310,311
360,361,362,363,364,365,366,367,370,371,172,176
```

MMTC drivers may have two-channel controllers. Thus, both channels must be initialized. After initializing the first channel, IUP requests a second channel via the console typewriter. If a channel is present, the operator types the channel number and the second channel will be initialized. If there is no second channel, press FINISH, allowing IUP to continue normally.

5.2.6 SKEF STATEMENT

The SKEF statement directs IUP to skip forward to end-of-file on designated magnetic tapes.

Card input:

SKEF, lun

Typewriter input:

SKEF LUN lun

lun	Logical unit number of magnetic tape

5.2.7 RWND STATEMENT

The RWND statement directs IUP to rewind the designated tape unit.

Card input:

RWND, lun

Typewriter input:

RWND LUN lun

lun	Logical unit number of magnetic tape

5.2.8 AGEN STATEMENT

The AGEN statement directs IUP to punch autoload cards for the designated library device.

Card input:

AGEN, lun1, lun2, n

Typewriter input:

AGEN, LIB lun1 CP lun2 NUMBER n

lun1	Logical unit number of library device
lun2	Logical unit number of card punch
n	Number of cards to be punched

5.2.9 LGEN STATEMENT

The LGEN statement punches one-card loader cards.

Card input:

LGEN, lun1, lun2, n

Typewriter input:

LGEN CR lun1 CP lun2 NUMBER n

lun1	Logical unit number of card reader
lun2	Logical unit number of card punch
n	Number of cards to punch

5.2.10 DUMP STATEMENT

The DUMP statement produces a dump file of a designated MSOS library edition in a format suitable for install processing (INST control statement).

Card input:

DUMP, lun1, lun2, ed

Typewriter input:

DUMP LIB lun1 DUMP UNIT lun2 EDITION ed

lun1	Logical unit number of mass storage containing MSOS library
lun2	Logical unit number of unit on which the library is to be dumped. The unit can only be the card punch or magnetic tape.
ed	Edition of library to be installed using this dump

5.2.11 INST STATEMENT

The INST statement processes the dump file produced by the IUP DUMP function and installs an MSOS library edition on a designated mass storage device.

Card input:

INST, lun1, lun2, ed, dn, mfc, msc,[exid], [mode]

Typewriter input:

INST LIB lun1 DUMP UNIT lun2 EDITION ed DN dn MFC mfc MSC msc [EXID exid] [MODE $\begin{Bmatrix} S \\ T \end{Bmatrix}$]

lun1	Logical unit number of mass storage device on which library is to be installed
lun2	Logical unit number of unit containing the library dump file
ed	Edition of library to be installed
dn	Device number used in the device label
mfc	Maximum file count for the new system
msc	Maximum segment count per mass storage file for new system
exid	External identification used in device label (optional)
mode	Mode if lun1 is 852 or 1311

S	Sector
T or blank	Track

5.2.12 END STATEMENT

The END statement terminates MSOS IUP.

Card and Typewriter input:

END

5.3 MSOS IUP MESSAGES

MSOS IUP communicates with the operator through the console typewriter. Messages are preceded by letter A, I, or D.

A	Operator intervention is required
I	Informative
D	Destructive

IUP messages are listed in Tables III-1 thru III-3.

TABLE III-1. IUP ACTION MESSAGES

Type	Source	No.	Message	Unit	Significance	Result/Action
A	IUP	006	READY FOR INPUT	CTO	IUP is ready to accept a control statement name from CTO.	Type control statement name.
A	IUP	009	END-OF-TAPE ON INPUT UNIT TYPE Y IF NEW TAPE MOUNTED TYPE N FOR IUP TERMINATION	CTO	End-of-tape found on magnetic tape presently used as input.	Type N if no other tape is to be mounted or Type Y after new tape is mounted.
A	IUP	012	UNIT NOT READY FO fo LUN lun	CTO	Mass storage device not ready.	Type R to retry, or A to abandon the call.
A	IUP	013	END-OF-TAPE ON OUTPUT TAPE TYPE Y IF NEW TAPE MOUNTED TYPE N FOR IUP TERMINATION	CTO	Written to end of output tape.	Type Y after mounting new tape, or Type N to terminate IUP if no new tape is mounted.
A	IUP	018	TURN ON WRITE ADDRESS SWITCH PRESS FINISH WHEN READY	CTO	WA option specified on DEV statement for 852/ 1311.	1. Turn on WRITE address switch. 2. Press FINISH.
A	IUP	021	UNABLE TO LOAD IMAGE MEMORY CORRECTLY	CTO	IUP compared written image memory and one read-in. They are not equal.	Type R to rewrite the image, or Type A to abandon initialization of the unit.
A	IUP	022	{SEL / CON} REJECTED - yyyyyyyy	CTO	A select or connect rejected while trying to initialize image memory yyyyyyyy=rejected instruction	Type R to entry or A to abandon initialization.
A	IUP	023	IMAGE MEMORY {OUTW / INPW} REJECTED	CTO	OUTW or INPW rejected while trying to initialize memory.	Type R to retry or A to abandon initializing the unit.

TABLE III-1. IUP ACTION MESSAGES (cont'd)

Type	Source	No.	Message	Unit	Significance	Result/Action
A	IUP	026	CONTROLLER BEING INITIALIZED IS BUSY AFTER 10 SECONDS	CTO	I/O not completed after 10 seconds when trying to write image memory.	Type R to retry for another 10 seconds, or Type A to abandon initialization of the unit.
A	IUP	030	BUSY CHANNEL OR UNIT REJECTED I/O CALL 10000 TIMES $\left\{\begin{matrix} \text{FO} & \text{fo} \\ \text{LUN} & \text{lun} \end{matrix}\right\}$	CTO	I/O call to mass storage device rejected 10000 times because of busy channel or unit.	Type R to retry another 10000 times, or Type A to abandon I/O call and present operational task.

5.3.1 INFORMATIVE MESSAGES

IUP informative messages are listed in Table III-2.

Some of the informative messages can be corrected immediately or require reassigning of source input. If input is from the typewriter, one of the following is requested:

The parameter causing the error is requested again, or

The operator may reassign input. INP = LUN is typed by IUP. The operator then reassigns input using the INP control statement format (Section 5.7.2).

If input is from the card reader, the operator may:

Reassign input with an INP statement to a different device.

Reassign to the same unit and skip a bad card.

Remove the bad card from the receiving tray, correct it, return it to the input tray, and reassign input to the same unit.

Assign input to the typewriter, type in the correct control statement, and reassign input back to the card reader using the INP statement.

If input is from magnetic tape, input may be reassigned as from the card reader.

TABLE III-2. IUP INFORMATIVE MESSAGES

Type	Source	No.	Message	Unit	Significance	Result/Action
I	IUP	001	ILLEGAL CONTROL STATEMENT	CTO	Control statement name on card or typewriter is invalid.	NOTE Refer to Section 5.3.1 for operator action when type I message occurs.
I	IUP	002	cnst CARD PARAMETER ERROR	CTO	Illegal parameter on cnst control statement where cnst is the control statement name.	
I	IUP	004	CONTROL STATEMENT SEQUENCE ERROR - NO DEVICE CARDS HAVE BEEN PROCESSED.	CTO	A DEV control statement has to be the first control statement.	
I	IUP	005	UNABLE TO EQUIP THIS DEVICE	CTO	Operational control statement caused device drivers of unequipped device to be destroyed.	
I	IUP	007	LOGICAL UNIT ALREADY ASSIGNED	CTO	Logical unit number not unique	

TABLE III-2. IUP INFORMATIVE MESSAGES (cont'd)

Type	Source	No.	Message	Unit	Significance	Result/Action
I	IUP	008	NO DEVICE OF THIS HARDWARE TYPE HAS BEEN EQUIP-PED	CTO	Only the logical unit number and hardware type given in INP statement but no device of same hardware type has been equipped.	
I	IUP	010	LOGICAL UNIT NUMBER SPECIFIED IN CONTROL STATE-MENT IS UNASSIGNED	CTO	Logical unit number used in control statement has not been assigned in DEV statement.	
I	IUP	014	ILLEGAL DEVICE TYPE LUN lu	CTO	IUP control statement contains the logical unit number, lu, of a device type which can not be used for this operation.	
I	IUP	016	CONTROL STATE-MENT READ = card read	CTO	This message is produced when a control card is read from a source other than the typewriter.	
I	IUP	017	CONFLICTING HARD-WARE TYPE AND DEVICE TYPE OR CONTROLLER DEVICE TYPE	CTO	The hardware type speci-fied on a DEV or INP statement cannot have the device type or controller device type specified.	
I	IUP	019	DUPLICATE MMTC CONVERSION CODES (xx) AND (yy)	CTO	While entering a nonstand-ard image for MMTC image memory duplicate codes were found.	
I	IUP	020	ONLY 4 MASS STOR-AGE DEVICES CAN BE EQUIPPED	CTO	More than 4 different mass storage devices trying to be equipped.	
I	IUP	021	IMAGE MEMORY PARITY ERROR	CTO	While reading or writing image memory, a parity error occurred.	
I	IUP	025	UNEXPECTED STATUS WORD II (xxxx)	CTO	During initialization of MMTC image memory an I/O status was returned that was not expected.	

TABLE III-2. IUP INFORMATIVE MESSAGES (cont'd)

Type	Source	No.	Message	Unit	Significance	Result/Action
I	IUP	027	NONSTANDARD IMAGE CODE CONTAINS TOO MANY DIGITS	CTO	Non-standard image memory being entered contains a conversion code greater than 4 digits.	
I	IUP	029	INSTALLATION UTILITY PROGRAM TERMINATED	CTO	IUP program complete. END statement processed.	
I	IUF	033	ATTEMPT TO EQUIP MORE THAN 30 DEVICES	CTO	More than 30 devices trying to be equipped.	
I	IUP	036	DATA CARD CHECKSUM ERROR	CTO	Data card input by INSTALL function has a checksum error.	

5.3.2 DESTRUCTIVE MESSAGES

IUP destructive messages are listed in Table III-3.

These messages are output for errors that terminate an operational function. IUP does not terminate. The operator can reassign input. The messages INP = LUN is typed following the destructive message. The operator may then reassign input using the INP statement. Reassigning input gives the operator the following options:

Reassign the same unit and skip the control card causing the problem

For card reader input, the card can be corrected and replaced in the input tray before assigning input to the same unit

For card reader or magnetic tape input, the source input can be reassigned to the typewriter, the correct control statement typed, the input original unit

Assign input to a new device

TABLE III-3. IUP DESTRUCTIVE MESSAGES

Type	Source	No.	Message	Unit	Significance	Result/Action
D	IUP	011	MSIO REJECT {FO = fo / LUN = lun} REJECT CODE zz	CTO	MSIO reject on I/O operation. zz = MSIO reject code	
D	IUP	015	LUN lun message	CTO	I/O error recovery returned one of the following errors: SCAR REJECT, RAAR REJECT, EQUIPMENT DOWN, IRRECOVERABLE ERROR	NOTE
D	IUP	028	UNEXPECTED EOF ON LUN lun	CTO	IUP found EOF before expected.	Refer to Section 5.3.2 for operator action when a type D message occurs.
D	IUP	031	DUMP FILE FORMAT ERROR	CTO	Information being read by install function contains unrecognizable data.	
D	IUP	032	INSUFFICIENT MASS STORAGE	CTO	Mass storage device on which library is being installed does not have room for complete library.	
D	IUP	034	DATA CARD FORMAT ERROR	CTO	The data cards input by the install function do not have correct format.	
D	IUP	035	DATA CARD SEQUENCE ERROR	CTO	Data cards being input by the install function are out of order.	

AET

At installation time, the available equipment table AET must define the equipment available to the system. Each AET entry consists of two 24-bit words and an external definition which describes the hardware device and the system driver servicing it. Any channel or equipment dedicated to real-time or specialized operations such as RESPOND channels must not appear in the AET. Each physical unit constitutes one complete AET entry except an 814 disk file which is represented by two separate AET entries.

6.1 AET FORMAT

The format of the AET entry (word 1) is:

23	22 18	17	16	15	14	13 12	11 00
a	h	s	r	cp	m		cc

The format of the EXT driver entry point name (word 2) is:

23	22 15	14 00
e	c	daddr

If the same driver services more than one device, the additional external definition may be omitted since only one definition is necessary.

The fields are defined as follows.

Word	Bit	Field	Significance
1	23	a	0 Unit not assigned to a logical unit 1 Unit assigned to a logical unit
	22-18	ht	Numeric code 1-37 designates hardware type

Code	Type
01	Magnetic Tape (MT)
02	Card Reader (CR)
03	Printer (PR)
04	Card Punch (CP)
05	Console Typewriter (TY)
06	Paper Tape Reader (TR)
07	Paper Tape Punch (TP)
10	Typewriter Station (TS)
11	Plotter (PL)
12	Satellite Controller (SL)
13	Disk Pack Controller (DP)
14	Disk File (DF)
15	Drum (DR)
16	Optical Character Reader (OR)
17	Seismic (SP)
20-37	Unassigned

Word	Bit	Field		Significance
1	17	s	0	Unit operable
			1	Unit inoperable
	16	r	0	Not reserved
			1	Reserved for another computer
	15	cp	0	No transmission parity error detected; should be set to 0 at installation
			1	Transmission parity error occurred
	14	m	0	Batch interrupt
			1	Priority interrupt
	13-12			Reserved
	11-00	cc		12-bit connect code for each unit; the hardware code used by MSOS in the I/O instruction cc = euuu e = One octal digit equipment number uuu = Up to 3 octal digits right-justified represent the unit number †
2	23	e	0	No action
			1	Unit to be assigned. Set e to 0 at installation time as this bit is manipulated only by MSOS.
	22-15	c		Eight-bit channel code specifying the channels available to an equipment. A bit set at 1 means the corresponding channel is available to a unit; a bit set at 0 means the channel is not available. The following table shows the correspondence between bit positions and channel numbers. ††

† The unit number in the connect code for mass storage is:

00U = 852/1311/863/865
0dU = 853/854/813/814/841
U = Unit number
d: 1 = Disk drive; 2 = Disk file

†† The console typewriter (non-channel) c should equal 0.

Word	Bit	Field	Significance
2	14-00	daddr	A 15-bit address that is the entry point name of the appropriate driver (part II, paragraph 2.13). A zero value in this field indicates a nonresident driver and, thus, a nonsystem unit. When MSOS loads a nonresident driver, it supplies a driver address in this field. A system unit entry must have driver address in the daddr field.

Bit Position	22	21	20	19	18	17	16	15
Corresponding Channel	0	1	2	3	4	5	6	7

Entry point names to be used for the various equipments and their drivers are listed in Table 3-4. Specific COSY corrections for adapting the AET to a particular installation are not given because of the diversity of configurations. A COSY correction causes the complete deletion of the AET and the user must insert his complete hardware description. A general format of the AET is as follows:

AET		entry 1, word 1	Library
	EXT	driver entry point name	
+1		entry 1, word 2	
+2		entry 2, word 1	CTO/CFO
	EXT	driver entry point name	
+3		entry 2, word 2	
+4		entry 3, word 1	INP
	EXT	driver entry point name	
+5		entry 3, word 2	
+6		entry 4, word 1	OUT
	EXT	driver entry point name	
+7		entry 4, word 2	
+8		entry 5, word 1	PUN
	EXT	driver entry point name	
+9		entry 5, word 2	

In this scheme, the preceding are system units and the following are available units for Batch or priority use.

+10		entry 6, word 1	Available equipment
	EXT	driver entry point name	
+11		entry 6, word 2	
+12		entry 7, word 1	Available equipment
	EXT	driver entry point name	
+13		entry 7, word 2	

The available equipment scheme may continue. A maximum of 50 two-word entries may be defined.

Table III-4 may be used to determine driver entry point names.

TABLE III-4. HARDWARE DRIVER REFERENCE

Hardware	Controllers	Equipment (in combination)	COSY Name	Entry Point Name
MT	362X 342X 322X	604 607 608	DRIV607	DRIVER01
MT	362X 342X 322X	601 603 606	DRIV606	DRIVER01
MT	3518 3528	657 659	DRIV3528	DRIVER01
CR	3447 3649	405	DRIV3649	DRIVER02
CR	3248	405	DRIV3248	DRIVER02
PR	3256 3659 3254	501 505	DRIV3659	DRIVER03
PR	3555	512	DRIV3555	DRIVER03
CP	3644 3446	415	DRIV3644	DRIVER04
CP	3245	415	DRIV3245	DRIVER04
TY	Console Typewriter		DRIVTYWR	DRIVER05
PL	3293	Plotter	DRIV3293	DRIVER11
DP	3232	852	MSIO3232	MSIO3232
DP	3234	853, 854 813, 814	MSIO3234	MSIO3234
DP	3553	841	MSIO3553	MSIO3553
DR	3436 3637	863	MSIO3436	MSIO3436
OR	3195	915	DRIV3195	DRIVER16
TR TP	3691	Paper Tape Station	DRIV3691	DRIVER06 DRIVER07

6.2 AET MACRO

AET (ht, c, e, u, driver, RES, DOWN)

ht	Hardware type of device corresponding to the AET entry	
	MT	Magnetic tape
	CR	Card reader
	PR	Printer
	CP	Card punch
	TY	Typewriter
	TR	Paper tape reader
	TP	Paper tape punch
	TS	Typewriter station
	PL	Plotter
	SL	Satellite controller
	DP	Disk pack
	DF	Disk file
	DR	Drum
	OR	Optical character reader
c	Channel code	
	200	Channel 0
	100	Channel 1
	040	Channel 2
	020	Channel 3
	010	Channel 4
	004	Channel 5
	002	Channel 6
	001	Channel 7

This code may have more than one bit set indicating more than one channel is connected to a specific device.

Example: 204 Channels 0 and 5

e	One octal digit equipment number
u	One octal digit unit number
driver	The entry point name of the appropriate driver. For a nonresident (file 2) driver, the entry point name must be [redacted]. NRD.
RES	This parameter is included if the unit is assigned to a logical unit
DOWN	This parameter is included if the device is inoperable.

RHT AND BRHT 7

7.1 RHT MODIFICATION

The RHT (running hardware table) contains 63 one-character entries that define the equipment used during a Batch run and the equipment assigned as system units (PUN, INP, OUT, CTO, and CFO). In MSOS, nonmass storage input/output units are referenced by logical unit numbers (LUN) rather than by physical units. The character entries in the RHT, from character RHT+1 to RHT+63, correspond to LUNs 01 to 63, respectively. Each RHT entry defining an equipment contains a nonzero value which is the ordinal in octal of the AET (available equipment table) and UST (unit status table) entry of the physical unit to which it is assigned. Certain system units must have permanent RHT entries as follows:

PUN	LUN 62
OUT	LUN 61
INP	LUN 60
CTO	LUN 59
CFO	LUN 58

Because the MSOS library and certain scratch LUNs are on mass storage, their LUN entries must be zero. The following table shows what the RHT must look like for these RHT entries.

RHT Entry	Character Value in RHT
63 (LIB)	0
54	0
55	0
56	0

Each RHT position for LUNs 58-62 must point to the appropriate AET and UST entry of an equipment with a resident driver. All nonsystem entries and the LIB entry (63) must initially be zero. The initial entries for system units (PUN=05, CTO/CFO=02, OUT=04, and INP=03) need not be changed if the AET is in the described format (part II, section 2, paragraph 2.5.11.)

7.2 BRHT MODIFICATION

The BRHT (priority running hardware table) is similar to the RHT, except that only logical units 1-49, 58, and 59 are available for reference by a priority program. Logical units 50-57 and 60-63 are illegal units for a priority program and their BRHT entries must be zero; logical units 58 and 59 are assembled to 02. The BRHT contains 63 one-character entries from BRHT+1 to BRHT+63 that correspond to priority LUNs 1-63, respectively. Entries for LUNs 50-57 and 60-63 must be included in the BRHT but must have a zero value for detection of illegal priority program references.

INTERIM LIBRARY 8

The operator defines the interim library environment through the CTO. The interim library types the system unit, requests the hardware type, and returns the carriage when the operator completes the response.

The interim library routine requests system units in the following order.

LIB	(library)
INP	(standard input)
OUT	(standard output)
PUN	(standard punch)
SCR	(system scratch)

After each system unit request, the operator types the mnemonic representing the hardware type (hh). The interim library routine types C and the operator types the channel numbers; the interim library routine types E and the operator types the equipment controller number; the interim library routine types U and the operator types the device unit number. The carriage returns ana the interim library routine types DT (device type) or CDT (controller device type); the operator responds with the 3- or 4-digit device type. Operator responses to all interim librarv reauests are listed in Table III-5.

Prior to typing the last character in any line, the operator can correct an error by pressing REPEAT. The line repeats and the operator types the accurate response.

The operator must define LIB, INP, and OUT. He may omit the punch by pressing FINISH when PUN is requested. The AET permits 15 entries and is arranged in the following order.

AET Ordinal	System Unit	
1	Library	
2	Console Typewriter	
3	Input	
4	Output	
5	Punch	
6	Scratch	
7	Scratch	
10	Scratch	Only 2 of the 10 scratch entries may be mass storage devices
11	Scratch	
12	Scratch	
.	.	
.	.	
.	.	
15	Scratch	

8.1 DEFINITION OF ENVIRONMENT

```
* INTERIM LIBRARY *
DEFINE ENVIRONMENT

LIB = hhCcEeUuu

†DT = ttt

INP = hhCcEeUuu

CDT = cccc

OUT = hhCcEeUuu

DT = ttt

PUN = hhCcEeUuuu

CDT = cccc

SCR = hhCcEeUuu

DT = ttt
```

SCR = (By pressing FINISH, the interim library environment is defined and DATE is typed.)

hh = hardware type:

hh	meaning
DP	Disk pack controller
CR	Card reader
PR	Printer
CP	Card punch
MT	Magnetic tape

c = channel number 0-7

e = equipment number (controller)

uu = unit number (device)

ttt = device number

cccc = controller device type

When the standard units are defined, press FINISH. The request for DATE signals the end of interim library environment definition.

† The interim library is designed to have only one driver available for each hardware type. No check is made for a hardware device being entered twice. The last driver entered for each device type is the driver used for that device type.

TABLE III-5. INTERIM LIBRARY RESPONSE TABLE

CTO Request	Unit DT/ CDT	Hardware Type	Channel Number	Equipment Number	Unit Number
LIB		DP	0 to 7	0 to 7	10 for 854/853/841 00 for 1311/852/863
DT	1311, 852, 853, 854, 841, 863				
INP		CR	0 to 7	0 to 7	00 to 07
CDT	3248, 3447, 3649				
INP		MT	0 to 7	0 to 7	00 to 07
DT	607, 606, 604, 603, 601, 657, 659				
OUT		PR	0 to 7	0 to 7	00 to 07
DT	501, 505, 512				
OUT		MT	0 to 7	0 to 7	00 to 07
DT	607, 606, 604, 603, 601, 657, 659				
PUN		CP	0 to 7	0 to 7	00 to 07
CDT	3245, 3446, 3644				
PUN		MT	0 to 7	0 to 7	00 to 07
DT	607, 606, 604, 603, 601, 657, 659				
SCR		MT	0 to 7	0 to 7	00 to 07
DT	607, 606, 604, 603, 601, 657, 659				
SCR		DP	0 to 7	0 to 7	11, 12 for 853/854/841
DT	1311, 852, 853, 854, 841, 863, 813, 814				01, 02 for 1311/852/863 21, 22 for 813/814

8.2 INTERIM LIBRARY RESTRICTIONS

Each system and scratch unit can use only one controller device type and one magnetic tape, card reader, card punch, printer, and disk pack or file. If more than one type is assigned for a system or scratch unit, the interim library routine types the following:

NO GOOD (device/controller device - CONFLICT)

This negates the erroneous assignment and the interim library again requests assignment of the unit.

8.3 AUTOLOADING INTERIM LIBRARY EXAMPLES

Example 1: Device type 852 and standard input magnetic tape.

```
* INTERIM SYSTEM *
DEFINE ENVIRONMENT
LIB = DPC0E1U00
DT = 852
INP = MTC1E1U01
DT = 607
OUT = PRC3E4U00
DT = 501
PUN = CPC3E3U00
CDT = 3644
SCR = DPC0E1U02
DT = 852
SCR =                              (Press:  FINISH)
DATE
Interim library is now autoloaded.
```

Example 2: Device type 854

```
* INTERIM SYSTEM *
DEFINE ENVIRONMENT
LIB = DPC0E0U10
DT = 854

INP = CRC1E1U00
CDT = 3648
CDT = 3649
OUT = PRC1E2U00
DT = 501
PUN = CPC7E7U00
CDT = 3446
SCR = DPC0E0U11
DT = 854
SCR = MTC4E4U00
DT = 604
SCR = MTC4E4U01
DT = 604
SCR =                          (Press:  FINISH)
DATE
Interim library is now autoloaded.
```

Example 3: Device type 863.

```
* INTERIM SYSTEM *
DEFINE ENVIRONMENT
LIB = DRC2E0U00
DT = 863
INP = CRC1E1U00
CDT = 3447
OUT = PRC1E2U00
DT = 501
PUN = CPC7E7U00
CDT = 3446
SCR = MTC4E4U00
DT = 604
SCR =                          (Press:  FINISH)
DATE
Interim library is now autoloaded.
```

8.4 COMPASS

The COMPASS supplied with the MSOS 4.2 interim library is COMPASS 3.2 released as part of MSOS 4.2. Some 16K users may encounter problems with insufficient core to assemble PRELOAD. For this reason, a second version of COMPASS is also available on the interim library. It is COMPASS 2.1 as released under MSOS 3.0 with additional PSR's included. COMPASS 2.1 is called by a ${}^{7}_{9}$CMPS2.1 card. All other operating characteristics are the same as COMPASS 2.1/MSOS 3.0.

PLIBEDIT 9

9.1 I/O FILES

The PLIBEDIT I/O files are used in every execution. The standard units are assumed to be present and defined. CPF and NPF are equipped via loader declaration cards included in the library program. Additional input (logical units and file ordinals) may be used via FILE/ or UNIT/ control cards. These must be equipped and opened by the user. Any logical unit number or file ordinal which does not conflict with systems devices or the above mentioned units may be used. !

LED-cards: siehe MSØS 10-13.

PLIBEDIT I/O Files

CPF - logical unit 01 - current PLIBEDIT file

NPF - logical unit 02 - new PLIBEDIT file

CFO - logical unit 58 - comment from operator

CTO - logical unit 59 - comment to operator

INP - logical unit 60 - standard input file

OUT - logical unit 61 - standard output file

9.2 PLIBEDIT STATEMENTS

Execution of PLIBEDIT requires PLIBEDIT and RTS or MSOS/MSIO control cards.

PLIBEDIT is called by the $^7_9$PLIBEDIT control card

PLIBEDIT processing is directed by 11 user prepared control cards:

DELETE/

INSERT/

REPLACE/

LOCATE/

SKIP/

POSITION/

FILE/

UNIT/

COMMENT/

REWIND/

ENDEDIT/

MCHANGE/

Use the following symbols:

X	one to eight BCD characters or numbers of a deck name
X_1	one to eight BCD characters or numbers of a deck name
Y	one to twenty BCD characters or numbers of a control card, excluding column one
Z	one to eight BCD characters or numbers of macro name
F	file ordinal
U	logical unit of magnetic tape

9.2.1 DELETE/

PLIBEDIT deletes specified decks, control cards, or macro decks and copies from standard input onto the NPF any binary cards following the DELETE/ card until the next PLIBEDIT control card is encountered.

X	DELETE/		deck X is deleted
MACRO	DELETE/	MACRO, Z	MACRO set with name Z is deleted
X	DELETE/	X_1	decks X through X_1 are deleted
	DELETE/	Y	control card Y is deleted

9.2.2 INSERT/

PLIBEDIT inserts the binary cards or macro decks following an INSERT/ card up to the next PLIBEDIT control card.

X	INSERT/		binary cards which follow are inserted in front of deck X
MACRO	INSERT/	X	the following ${}^{7}_{9}$MACRO card and BCD cards are inserted in front of deck X
	INSERT/	Y	the following binary cards are inserted in front of control card Y

9.2.3 REPLACE/

PLIBEDIT deletes the specified decks, control cards, or macro decks and replaces these decks with the cards following the REPLACE/ card until a PLIBEDIT control card is encountered.

X	REPLACE/		deck X is replaced
MACRO	REPLACE/	MACRO, Z	MACRO set with name Z is replaced
	REPLACE/	Y	control card Y is replaced

9.2.4 MCHANGE/

PLIBEDIT places on the designated LUN or FO the binary decks following an MCHANGE/ card up to the next PLIBEDIT control card. A table is generated with a deck name for each entry. Each time an IDC card is encountered on the current PLIBEDIT file, a search of the table occurs. If a matching deck name is found, the deck is placed on the new PLIBEDIT file. Multiple replacement is done with an MCHANGE/ card.

FILE	MCHANGE/	F
UNIT	MCHANGE/	U

9.2.5 LOCATE/

PLIBEDIT locates the current PLIBEDIT file to the specified deck, control card, or n number of end-of-files.

X	LOCATE/		TRA card of deck X is located and all preceding decks or cards including the TRA card are copied onto NPF
	LOCATE/	Y	Control card Y is located and preceding decks or cards are copied onto NPF. Card Y is copied onto NPF. Y must be 1-20 BCD characters or numbers, excluding column one.
	LOCATE/	EOF=n	Locate CPF past n end-of-files. The value of n must be 1-9.
X	LOCATE/	n	TRA card of deck X plus n card images are located. Preceding decks or cards are copied onto NPF. Card n is copied onto NPF. The value of n must be 1-99.
	LOCATE/	n	Locate n card images. The value of n must be 1-99. Preceding decks or cards and card n are copied onto NPF.

9.2.6 SKIP/

PLIBEDIT skips cards or skips cards following a specified deck.

	SKIP/	n	Skip n cards. The value of n must be 1-99. CPF is ready to process card n+1.
X	SKIP/	n	Locate to TRA of deck X and skip n cards at that point. TRA card of deck X is copied onto NPF with all preceding decks or cards. CPF is ready to process card n+1.

9.3 PLIBEDIT MESSAGES

Message: I PLIBEDIT 001 PLIBEDIT BEGIN . . . DATE= dd/mm/yr . . . PAGE 01
Significance: Heading of first page.
Action: none

Message: I PLIBEDIT 002 PLIBEDIT CONTINUED . . .
Significance: Heading of pages 2-n.
Action: none

Message: D PLIBEDIT 003 ILLEGAL USE OF FILE CARD
Significance: File card on input inconsistent with PLIBEDIT deck assembly. Run aborted.
Action: Reassemble PLIBEDIT deck with file card denoting MSOS.

Message: A PLIBEDIT 004 INFORM USER OF I/O TROUBLE ON LUN 61
Significance: Irrecoverable error on logical unit 61.
Action: Operator must inform programmer of abort conditions.

Message: D PLIBEDIT 005 IRRECOVERABLE WRITE ERROR LUN xx
Significance: Unit is down or operation is abandoned as a result of a write error on logical unit xx.
Action: In response to error recovery message, operator typed D, down, or A, abandon.

Message: D PLIBEDIT 006 IRRECOVERABLE READ ERROR LUN xx
Significance: Unit is down or operation is abandoned as a result of a read error on logical unit xx.
Action: In response to error recovery message, operator typed D, down, or A, abandon.

Message: I PLIBEDIT 007 CARD CHECKSUM ERROR W.C. = xx, A.F. = xxxxx.
Significance: Checksum error on card with word count xx and address field xxxxx.
Action: Correct error condition.

Message: I PLIBEDIT 008 CHECKSUM ERROR IN ABOVE DECK
Significance: Checksum error in last deck named.
Action: Correct error condition.

Message: D PLIBEDIT 009 Y-FIELD ERROR-RUN ABORTED
Significance: Illegal Y field in last control card printed.
Action: Correct Y field on control card.

Message: I PLIBEDIT 010 FORGOTTEN JOB CARD-IGNORED
Significance: A JOB card is part of input data. Card is ignored.
Action: Remove card before next run.

Message: I PLIBEDIT 011 FORGOTTEN BLANK CARD-IGNORED
Significance: A blank card is part of data. Card is ignored.
Action: Remove card before next run.

Message: D PLIBEDIT 012 UNIDENTIFIABLE CARD-RUN ABORTED
Significance: BCD card unidentifiable. Run aborts.
Action: Identify correctly or remove before next run.

Message: I PLIBEDIT 013 FORGOTTEN EOF CARD-IGNORED
Significance: An end-of-file card is part of data. Card is ignored.
Action: Remove card before next run.

Message: D PLIBEDIT 014 REQUESTED C.C/DECK NAME CANNOT BE FOUND-RUN ABORTED.
Significance: Last control card or deck name printed before diagnostic cannot be found on current PLIBEDIT file. Run is aborted.
Action: Check deck name spelling. Reassemble.

Message: D PLIBEDIT 015 MCHANGE TABLE OVERFLOW-RUN ABORTED
Significance: Multiple change table has overflowed. Run aborts.
Action: Maximum number of change decks has been exceeded.

Message: D PLIBEDIT 020 IRRECOVERABLE REJ. ON LUN xx
Significance: Irrecoverable reject on logical unit xx. Run aborts. RAARREJ was entered.
Action: In response to ready message, operator typed D, down, or A, abandon.

Message: D PLIBEDIT 021 IRRECOVERABLE MSREJ.xx ON FO.xx
Significance: Irrecoverable reject xx on file ordinal xx. Run aborts.
Action: Locate xx reject error in MSIO. Correct error condition.

Message: I PLIBEDIT 099 PLIBEDIT COMPLETED
Significance: PLIBEDIT run is completed.
Action: none

The MSOS product set includes PRELIB for the purposes of generating new library editions and creating auxiliary relocatable library files. PRELIB permits the user to absolutize frequently used system programs (COMPASS, FORTRAN, etc.) and eliminate the necessity of relocating and linking their routines each time they are called. The capabilities to DELETE, INSERT, and REPLACE routines from the input edition to the output edition are also available.

PRELIB generates a new edition using the input edition (IE) if specified on the parameter or the autoloaded edition (AE) for generating the new output edition (OE). OE may not be equal to any edition currently in existence on the system device. It is impossible to destroy the old library. Old editions may be purged via MSUTIL (see Operator's Guide).

When modifying file 2 absolute records, and/or relocatable subprograms, no other routines need be changed. Whenever resident or variable resident is modified, all other routines must be reabsolutized.

Before a PRELIB run, use Mass Storage Utility (MSUTIL) (See MSOS Operator's Guide) to enter a scratch pack into the MST so that the system can allocate PRELIB scratch files on it. This greatly reduces processing time because less head movement is necessary during the PRELIB run. Table III-6 lists all temporarily allocated files.

10.1 LOADING AND EXECUTION

PRELIB requires all of memory and may not share time with another program. The PRELIB user must be familiar with the PRELIB source deck supplied for installation.

PRELIB detail cards (PDC) are binary cards for the relocatable programs to be placed on the new library. Each deck must begin with an IDC card and end with a TRA card.

10.2 PRELIB CONTROL CARDS

PRELIB control cards (PCC) direct PRELIB in the generation of a new library. They allow addition and deletion of records and entry points.

10.2.1 PRELIB CARD

The user loads PRELIB into core with the PRELIB statement. When PRELOAD encounters a PRELIB card, it calls LOADER which in turn calls OVPRO to load PRELIB and its overlays and segments from file 2 of the library. SEQUENCE, JOB, and any required EQUIP cards precede the PRELIB card.

MSOS gives control to PRELIB, which reads from INP. PRELIB executes automatically when called; a RUN card is not needed. There are two formats of the PRELIB control card. If modifications are to be made to the main library, it has the following form:

```
7
9PRELIB,S,A,P,IE,OE
```

S — If S is present, PRELIB generates a suppressed history of updated files on OUT. If S is omitted, PRELIB generates a detailed history of updated files on OUT.

A — Must be present if modifications include file 1 absolute records to be absolutized; A is omitted if modifications are for the relocatable file only or if the entire system is being rebuilt.

P — If present, P allows use of PROMEM EXS card which is required during generation on a Memory Protect PRELIB source.† This parameter has no influence on LIBFILE modifications.

IE — Two alphanumeric characters designating the input edition for this PRELIB run. If omitted, the edition autoloaded is assumed.

OE — Two alphanumeric characters designating the name for the output edition. OE must be specified and distinct from any other edition.

When only an auxiliary library is to be generated, the PRELIB control card is as follows:

```
7
9PRELIB,S,AUX,fo,n
```

S — Same as above

AUX — Required.

fo — File ordinal of the auxiliary library file. The AUX library file must be allocated and opened by the user prior to entering PRELIB. Its block size is 500 characters

n — Number of blocks to be saved for the AUX DRS. If omitted, 20 blocks are saved (41 symbols can be placed in each block; 20 blocks are sufficient for all the primary entry points in the UFTN object time routines).

10.2.2 FILE CARD

A FILE card signals the completion of RESFILE and ABSFILE processing. If the FILE card is the first card in the PRELIB input deck, the old RESFILE and ABSFILE are merely copied. The second FILE card signals the end of the relocatable file (LIBFILE) processing.

Für AUX nur eine Filekarte

† P must be present for an MP/USASI COBOL PRELIB run.

$^{7}_{9}$FILE, fo, n

fo — File ordinal of the auxiliary library file. The AUX library file must be allocated and opened by the user prior to entering PRELIB. Its block size is 500 characters.

n — Number of blocks to be saved for the AUX DRS (See Section 10.2.1).

10.2.3 EXPAND CARD

The EXPAND control card is provided as a means for the installation to expand the size of the MSDFILE, LABELFILE, and the IDFILE at PRELIB time.

Although this card is optional, when it is used, it must follow the PRELIB control card.

$^{7}_{9}$EXPAND, n_1, n_2, n_3

n_1 — Decimal digits, $1 \leq n \leq 99$; number of tracks to expand MSD. If omitted, a comma must delineate the field.

n_2 — Decimal digits, $1 \leq n \leq 99$; number of tracks to expand LABELFILE. If omitted, a comma must delineate the field.

n_3 — Decimal digits, $1 \leq n \leq 99$; number of tracks to expand IDFILE.

10.2.4 REPLACE CARD

A REPLACE card indicates that an entire relocatable file is to be replaced or identifies particular records in file 2 to be replaced. A REPLACE card may be followed by a file 2 UNIT card if the replacement records are not on INP.

Absolute records that are replaced are entered at the end of the absolute file on the new library.

$^{7}_{9}$REPLACE, $name_i$, $name_n$

$name_i$, $name_n$ — Subprogram on existing relocatable file or blank. The possible combinations specify the following actions.

$name_i$	$name_n$	Action
specified	specified	Replace subprograms $name_i$ through $name_n$ with PDC records following REPLACE on INP, or with PDC records on unit specified by unit card following REPLACE.
blank	specified	Replace subprograms from beginning of relocatable file through $name_n$ with indicated PDC's.
specified	blank	Replace subprograms from $name_i$ to end of relocatable file with indicated PDC's.
blank	blank	Replace entire relocatable file or RESFILE/ABSFILE with PDC's which follow.

When $name_i$ is the same as $name_n$, only the one subprogram is replaced.

10.2.5 DELETE CARD

A DELETE control card prevents the transfer of the designated records from the old library to the new one. The user must delete subprograms in the order of their appearance in the absolute file and in the relocatable file. There is no provision for deletion of individual resident subprograms since any modification of resident requires a complete regeneration of the system.

Vorkehrung

$^{7}_{9}$DELETE, $name_i$, $name_n$

$name_i$, $name_n$	Subprograms on existing relocatable file or blank. The possible combinations specify the following actions.

$name_i$	$name_n$	Action
specified	specified	Delete subprograms $name_i$ through $name_n$ on relocatable file.
blank	specified	Delete subprograms from beginning of relocatable file through $name_n$.
specified	blank	Delete subprograms from $name_i$ through end of relocatable file.
blank	blank	Delete entire relocatable file.
name		Delete named absolute record; name must not be followed by a comma.

When $name_i$ is the same as $name_n$, only the one subprogram is deleted.

10.2.6 INSERT CARD

INSERT control cards add records. Records are inserted on the relocatable file immediately after the named subprogram. If no subprogram is specified, the new records become the first records of file 2.

$^{7}_{9}$INSERT, name

name	Subprogram on existing relocatable file or blank

10.2.7 UNIT CARD

UNIT control cards specify the logical unit, other than INP, containing PRELIB detail cards. When a UNIT card is used to change resident or variable resident, it must follow an origin card or the PRELIB detail TRA card. When a UNIT card is used to control modifications of a relocatable file, it may follow a REPLACE, INSERT, or DELETE card. When used within the absolute file, a file ordinal is flagged with D as the second parameter on the card.

During execution, PRELIB uses file ordinals 1-13 for its own scratch files.

$^{7}_{9}$UNIT, u_1, D

u_1	Logical unit or file ordinal containing PRELIB detail deck
D	Indicates u_1 is a mass storage file (absolute file only).

10.2.8 MACRO CARD

MACRO control cards direct PRELIB to read Hollerith cards and write on the new library in a form compatible with the system languages.

$^{7}_{9}$MACRO,p,u

p Symbolic name, such as COMAC, that identifies a group of macros to be written.

u Logical unit containing macro cards; if blank, INP is assumed.

PRELIB adds a pseudo IDC card (word count of 41_8) and then processes the macro cards.

Each 80-column Hollerith card is read into core as a 20-word BCD record modified as follows.

Word 20 (columns 77-80) is deleted.

Words 1-19 are renumbered 2-20.

A new word 1 with a w field of 71_8 and with a $^{7}_{9}$ punch in column 1 is added. Two such cards are then written as a 40-word BCD card image.

MACROS used by COMPASS are required to occur in COMAC.

An END card (columns 10-12) must follow the last macro in the PRELIB input stream.

10.2.9 RECORD CARD

2

A RECORD control card marks the beginning of a new record to be absolutized. A RECORD card must be followed by an ORIGIN card.

$^{7}_{9}$RECORD, CLST

CLST When present, CLST causes PRELIB's symbol table to be cleared and initialized with only system entry points. In addition, if resident was replaced during the run, all resident entry points are retained.

10.2.10 ORIGIN CARD

An ORIGIN control card designates a loading position within a simulated target memory.

$^{7}_{9}$ORIGIN,n,i

n Unsigned 1- to 5-digit octal number or previously defined entry point in file 1. Previously defined entry point may be modified by signed (+ or -) 1- to 5-digit octal number.

i Unsigned 1- to 5-digit octal number indicating the number of words to be deleted from beginning of routine when record is written from core. May be blank.

10.2.11 SEPOINT CARD

A SEPOINT control card must be used to define and enter any entry point into a permanent position in the loader symbol table (LST).

```
7
9SEPOINT,name
```

name — File 1 entry point name; if name is undefined in file 1 and the symbol ABNORMAL is defined, name is equated to ABNORMAL. If name and ABNORMAL are undefined in resident, name is assigned the value 77777_8. In these cases, a job which transfers control to name terminates abnormally.

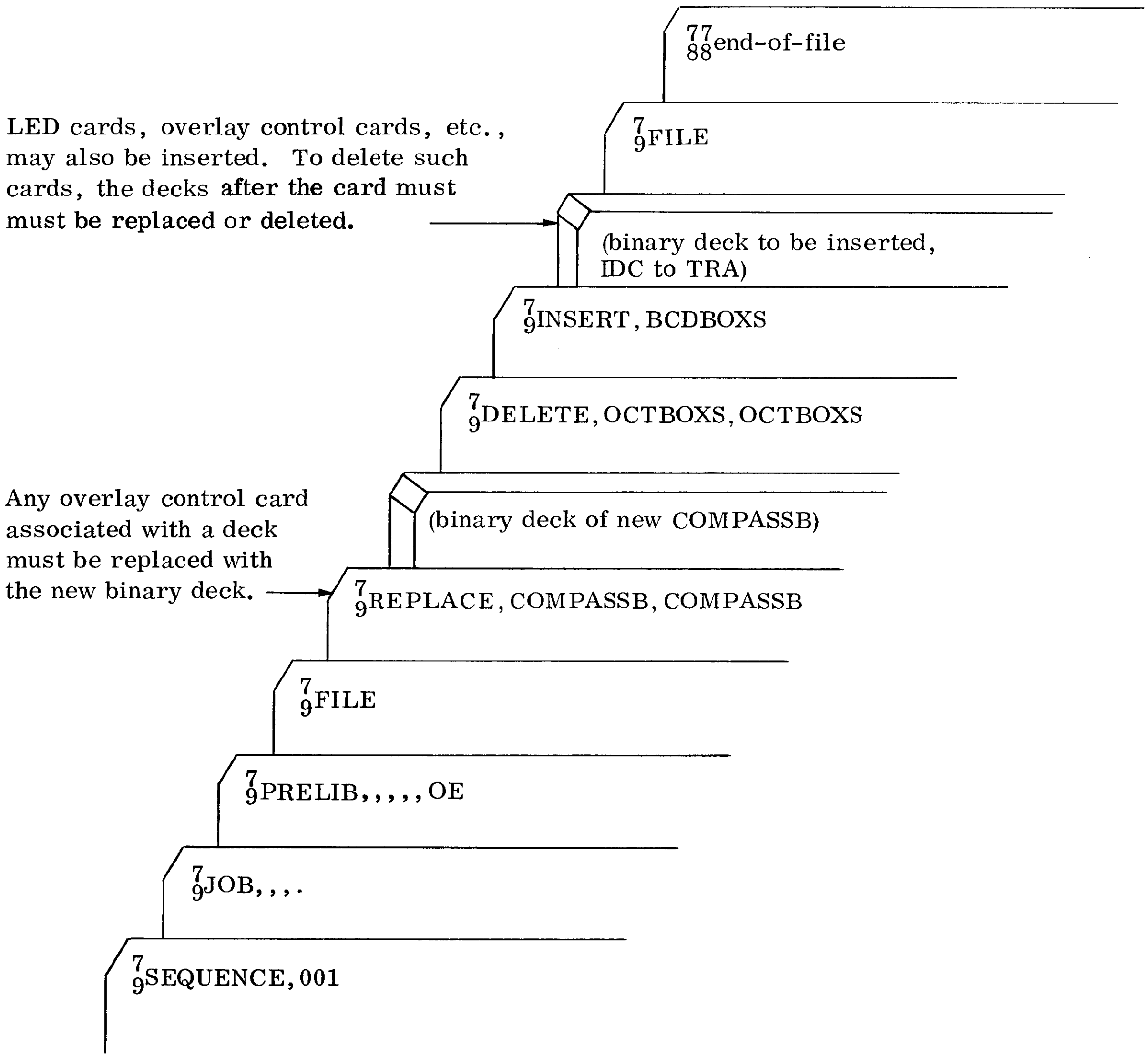
A SEPOINT card must be inserted directly after the LOADER record. When only system routines (e.g., COMPASS) are absolutized, SEPOINT cards are illegal.

For linking file 2 absolute records to system entry points, PRELIB uses the permanent LST of the current edition. Therefore, to avoid improper addresses being used for linking, the current edition should be the same as the input edition.

10.3 PRELIB EXAMPLES

10.3.1 MODIFY RELOCATABLE ROUTINES

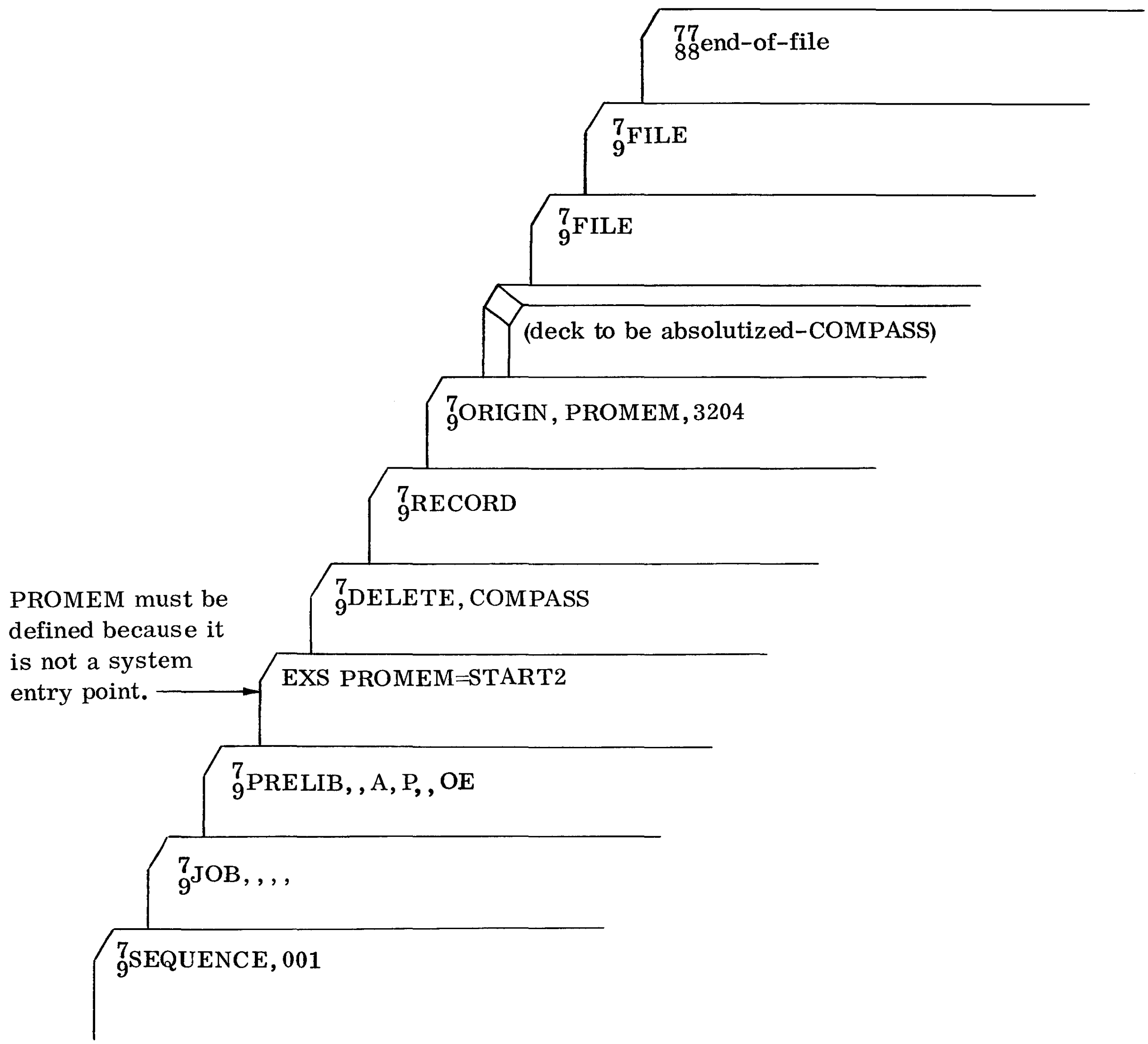
Replace, delete, and insert relocatable routines in file 2.



10.3.2 REPLACE ABSOLUTIZED PROGRAMS

Those products in file 2 partially comprised of absolute routines with overlays or segments (COMPASS, FORTRAN, etc.) can be modified. Resident and variable resident need not be changed.

For example, to replace only COMPASS:



10.3.3 REPLACE ENTIRE SYSTEM

${}^{7}_{9}$RECORD

PRELOAD binary deck

${}^{7}_{9}$ORIGIN, BINBUF+50

${}^{7}_{9}$RECORD

RDUMP binary deck

${}^{7}_{9}$ORIGIN, BINBUF+50

${}^{7}_{9}$RECORD

resident binary decks - ZERO,
CIO, MSIO, DRIVERS, CICREC1,
INITIAL

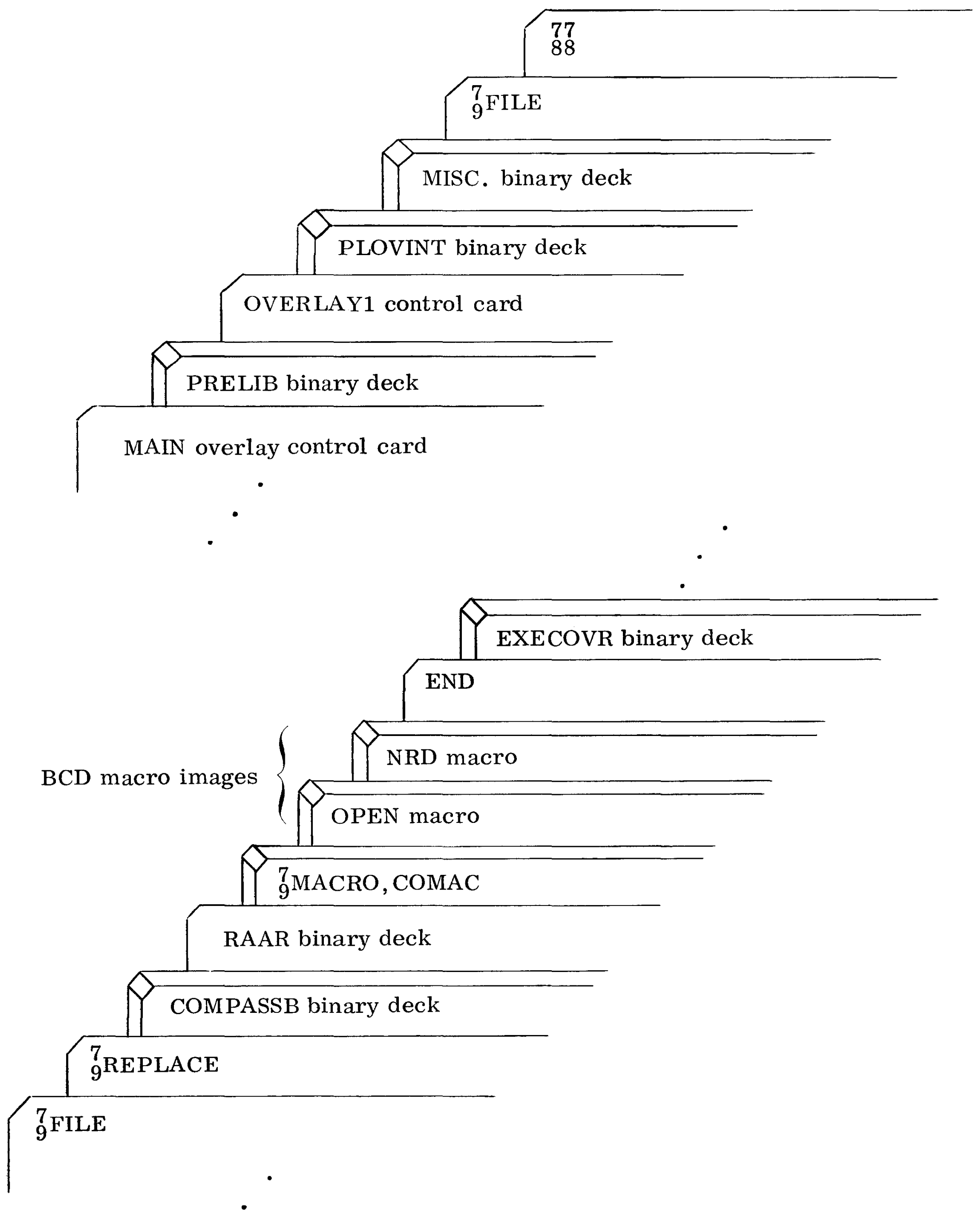
${}^{7}_{9}$ORIGIN, 0

${}^{7}_{9}$RECORD

${}^{7}_{9}$PRELIB, , , , , OE

${}^{7}_{9}$JOB, , , ,

${}^{7}_{9}$SEQUENCE, 069


77
88
$^{7}_{9}$FILE
MISC. binary deck
PLOVINT binary deck
OVERLAY1 control card
PRELIB binary deck
MAIN overlay control card
EXECOVR binary deck
END
BCD macro images
NRD macro
OPEN macro
$^{7}_{9}$MACRO, COMAC
RAAR binary deck
COMPASSB binary deck
$^{7}_{9}$REPLACE
$^{7}_{9}$FILE

10.3.4 GENERATE AUX LIB FOR UFTN

${}^{7}_{9}$SEQUENCE, 001

${}^{7}_{9}$JOB, , , ,

${}^{7}_{9}$FET, MSOS, AUXLIB, 500, UF

${}^{7}_{9}$ALLOCATE, 150, 991231, , NOSEG

${}^{7}_{9}$OPEN, 40

${}^{7}_{9}$PRELIB, , AUX, 40

(USASI FORTRAN object-time binary decks)

${}^{7}_{9}$FILE

77
88

10.3.5 ADD UFTN TO RELOCATABLE FILE AND GENERATE AUX LIB FOR OBJECT TIME ROUTINES

${}^{7}_{9}$SEQUENCE, 002

${}^{7}_{9}$JOB, , , ,

${}^{7}_{9}$FET, MSOS, AUXFILE, 500, UF

${}^{7}_{9}$ALLOCATE, 150, 991231, , NOSEG

${}^{7}_{9}$OPEN, 40

${}^{7}_{9}$PRELIB, , , , , OE

${}^{7}_{9}$FILE

${}^{7}_{9}$INSERT

(Binary deck of UFORTRAN)

$^{7}_{9}$FILE, 40

(Binary decks of UFORTRAN object time)

$^{7}_{9}$FILE

77
88

TABLE III-6. TEMPORARILY ALLOCATED FILES

Program	File Ordinal	File Name	Block Size	852/4	841	852
PRELIB	2	RESX	4	10	5	20
	4	ABSX	4	175	88	350
	6	LIBX	960	315	158	630
	8	DRSX	500	4	2	7
	9	SCRX	4	325	163	650
	10	ABUFX	4	10	5	20
	11	BBUFX	4	10	5	20
	12	CBUFX	4	10	5	20
	13	DBUFX	4	10	5	20
RESPOND	49 (overlay file)		960	20	10	40
UFORTRAN	49 (overlay file)		960	70		
L-MSIO	44 (overlay file)		10240	10		
AUX LIB			500	variable	variable	variable

10.4 ALLOCATION OF THE ABS AND RES FILES

When generating a new edition of MSOS, PRELIB allocates the new ABS, RES, DRS, and LIB files. However, the ABS and RES files require contiguous space. Sometimes there will not be enough contiguous space to allocate the new edition, although sufficient non-contiguous space is available. In this case, one of the following steps may be necessary:

1. Release unnecessary files from the system pack. System scratch files 54, 55, and 56 can be reallocated on a scratch pack (see Part II, Section 7).

2. Release unused portions of files where possible. For PRELIB only file 55 is needed at 25 tracks (853, 854, 863) or 10 tracks (841).

3. Dump files onto tape using MSUTIL. They may be returned to mass storage after PRELIB. This method is especially useful if the file is bounded by unallocated area.

4. If there is enough non-contiguous space available for the ABS or RES files, a dummy file may be used to reallocate this space so it is contiguous. An example of this is illustrated in Figure III-1, where edition E1 is to be purged and edition E3 created. The ABS file for edition E3 does not fit the space which E1 occupied prior to being purged. Allocation of a dummy file before the PRELIB for edition E2 will provide the needed contiguous space the PRELIB for edition E3, if the dummy file and E1 are released before the PRELIB for edition E3.

5. The relocatable routines that are used less frequently can be stripped from the main LIB file and placed on an AUX LIB file. This file does not have to reside on a system device.

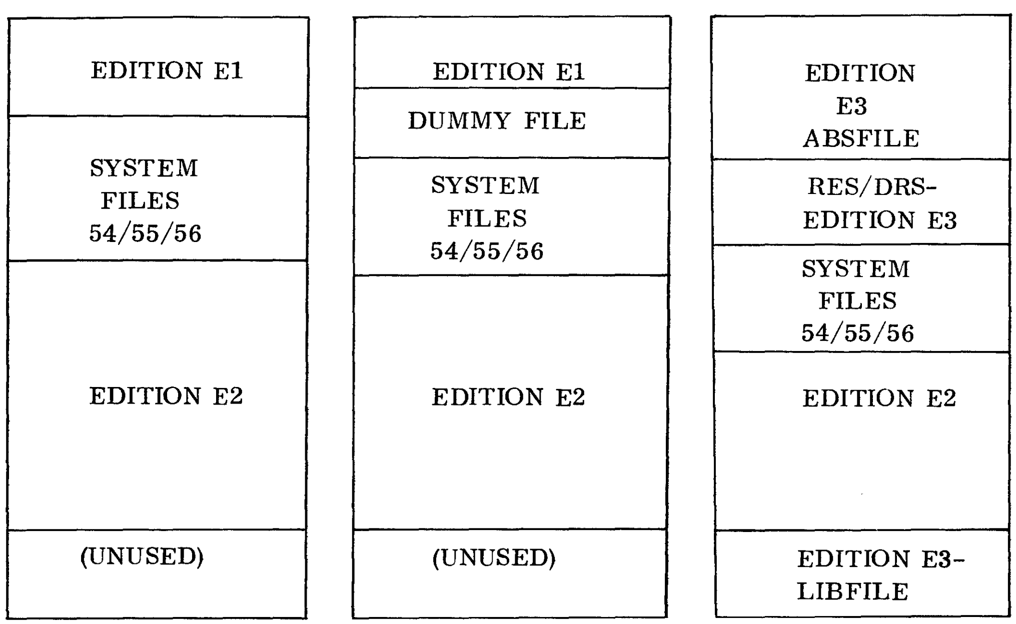


Figure III-1. Use of Dummy File to Facilitate Generating New Library Editions.

10.5 PLOV2 LISTABLE OUTPUT

A history of the second file update of the LIBT is produced on OUT during PLOV2 execution. This listing contains the names of all relocatable subprograms contained on the updated file in the following format:

```
T    NNNNNNNN LENGTH=XXXXX  COMM=XXXXX  DATA=XXXXX  **  MM/DD/YY
          ENTRY POINTS
          F PPPPPPPP          F PPPPPPPP    . . . . .
          EXTERNAL REFERENCES
          N RRRRRRRR          N RRRRRRRR    . . . . .
```

where T = V if the subprogram is a MAIN, OVERLAY, SEGMENT
= M if it is a macro
=blank if otherwise;

NNNNNNNN = the subprogram name from ${}^{7}_{9}$IDC card;

XXXXX = the lengths of the subprogram, length of common block and length of data block, respectively;

MM/DD/YY = the assembly date of the binary deck if specified.

F = D for doubly defined entry points
= S for uncallable entry points
= blank if otherwise;

PPPPPPPP = the entry point names as obtained from ${}^{7}_{9}$EPT card(s);

N = B if external reference refers to an entry point already defined,
= blank if otherwise;

RRRRRRRR = the names of the external references as obtained from the ${}^{7}_{9}$XNL card(s).

When the request has been made to suppress this listing, only T NNNNNNNN as defined above is listed.

MSOS/MSIO SYSTEM FILE MAINTENANCE 11

The following information is provided to aid in the maintenance of MSOS system files. At installation time, the MSIO related files are initialized. If special circumstances warrant modifying these files, this section should be consulted.

Under MSOS, one mass storage device is reserved for a system device which must remain on-line at all times. The utilization of this device is illustrated in Figure III-2. This device contains the following MSOS system files and MSIO files.

<u>MSOS System Files</u>

RESIDENT FILE (RESFILE)

ABSOLUTE FILE (ABSFILE - FO 67)

LIBRARY FILE (LIBFILE - FO 63)

DIRECTORY OF RELOCATABLE SUBPROGRAMS FILE (LIBDIRFILE - FO 68)

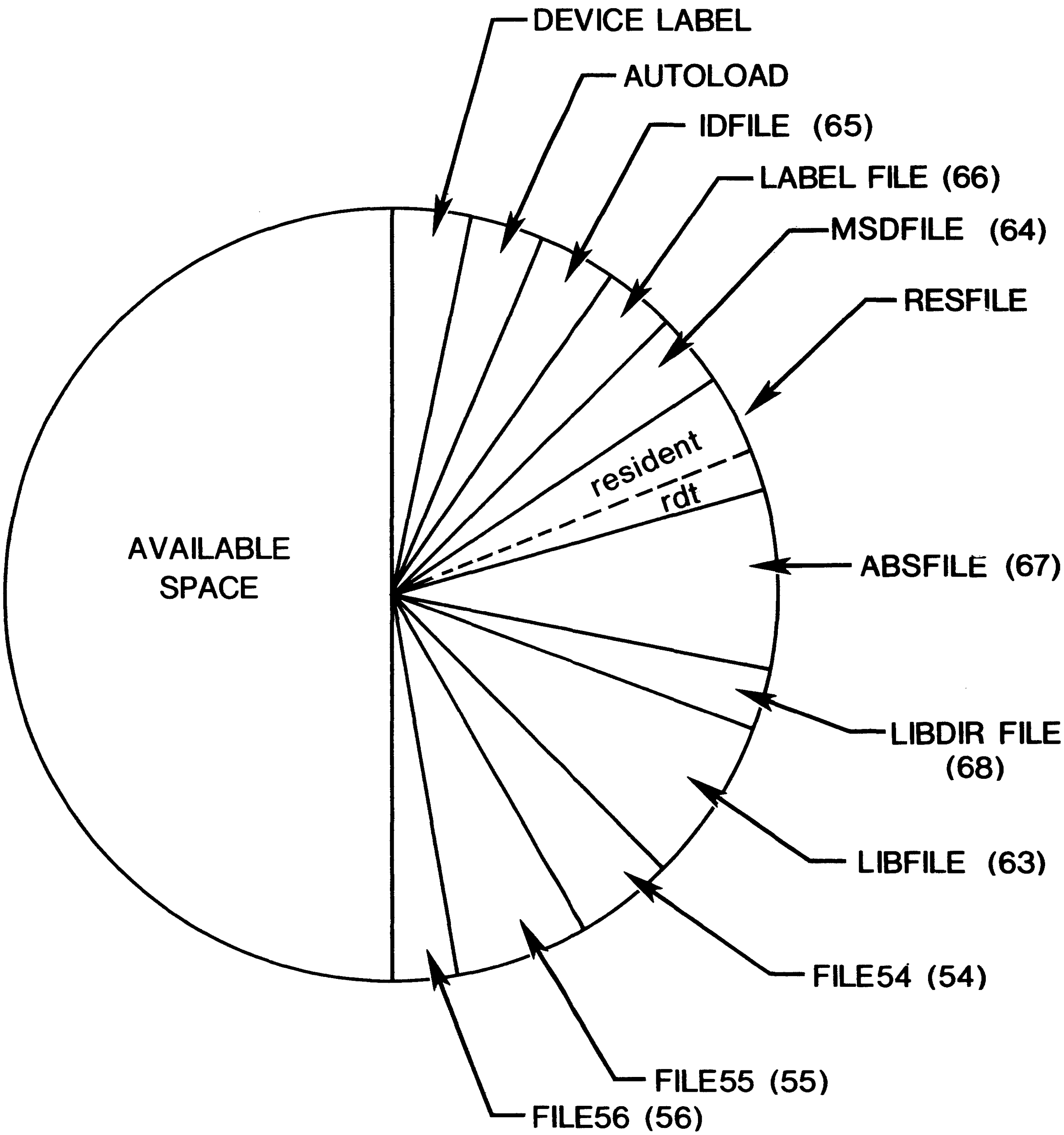
SCRATCH FILES (FILE54, FILE55, FILE56 - FO 54, 55, 56)

<u>MSIO Files</u>

FILE LABEL DIRECTORY FILE (LABELFILE - FO 66)

IDENTIFICATION FILE (IDFILE - FO 65)

MASS STORAGE DIRECTORY FILE (MSDFILE - FO 64)



Note:
FILE ORDINALS SHOWN IN PARENTHESIS

Figure III-2. MSOS System Device Utilization

11.1 MSOS SYSTEM FILES

11.1.1 RESIDENT FILE (RES)

Resident contains the following routines and tables which must be available at all times for reference by batch and priority programs:

ZERO	Defines linkage between EXEC and first 57_8 words of memory
CIC	Central interrupt control routine
CIO	Central input/output control routine
MSIO	Basic mass storage input/output resident routine
I/O DRIVERS	Drivers for standard system units
CIP	Central clock control routine
MANUAL INTERRUPT PROCESSORS	Routine for entering messages to MSOS, to batch programs, or to priority programs during system operation
CONTROL	Multipurpose resident routine which handles loading of variable resident and routes control through the variable resident routines.
TABLES	Tables may be referenced but not changed by users

11.1.2 ABSOLUTE FILE (ABS FILE - FO 67)

ABS contains all records not in resident which must be loaded into a specific core location prior to use. These routines, when loaded, overlay one another in core. The overlay strucuture is possible because the routines are needed only at particular points of processing. The variable resident routines are:

RDUMP	Recovery dump routine; prints out the contents of the console registers, the register file, and all of batch or priority memory. The routine is executed when a program terminates abnormally or when requested by manual interrupt.
PRELOAD	Clears memory and releases scratch units in preparation for loading; processes control statements.
MSIOCCP	Mass Storage Input/Output Control Card Processor.
CKREC/LOADER	Loads and links relocatable subprograms, library routines, and I/O drivers.
OVPRO	Overlay processor.
PROTECT	Prohibits certain operations on standard system units.
POSTLOAD	Processes control statements concerned with post-load operations and prepares MSOS for entry into user programs.

Other ABS subprograms, such as the COMPASS assembler and the FORTRAN compiler, are included with variable resident routines on the ABS files.

The resident directory table (RDT) provides linkage between resident and ABS. The RDCKF1 routine of resident refers to the RDT to load subprograms from ABS.

11.1.3 LIBRARY FILE (LIB-FO 63)

LIB, written in relocatable binary format, contains all library subprograms and macros.

11.1.4 DIRECTORY OF RELOCATABLE SUBPROGRAMS (DRS-FO 68)

When the user calls subprograms from LIB, the loader refers to the directory of relocatable subprograms (DRS). Every primary entry point of a LIB subprogram and every library macro name has a three word entry in the DRS of the following form:

word 1	entry name
word 2	entry name
word 3	fbn

entry name — Name of LIB subprogram or macro; eight BCD characters, left justified with blank fill

fbn — Block number, relative to LIB origin, of first binary card image of subprogram

The DRS consists of 125 word-blocks with a maximum of 41 entries per block. Unused areas of DRS contain zeros. The last word of each block is a pointer to the next block of DRS as follows:

Last word of block = 0 DRS continues to next block
$\neq$ 0 This block is the last block of DRS

11.1.5 SYSTEM SCRATCH FILES (FO 54,55, and 56)

The system library contains two scratch files which are automatically opened when MSOS is initialized. Programs such as FORTRAN and COMPASS use these files. The system assigns file ordinals 54 and 55 to the system scratch files; the user may perform input/output operations on system scratch using these file ordinals. At the end of a job, the system scratch files are reset to file origin to be available for the next job.

If the user requires larger scratch files, he may close the system scratch files and open his own scratch files using file ordinals 54 and 55. The block size of user scratch files should be 480 characters. At the end of the job, MSOS reassigns file ordinals 54 and 55 to the system scratch files.

The system load-and-go file (LGO) must be assigned when it is needed. Unless the programmer specifies otherwise, assembly and compilation load-and-go output appears on 56. The load-and-go file is reset to file origin when a LOAD,56 card is encountered, when loading is completed from 56, and when a job terminates.

If the user requires a larger load-and-go file, he may close the system LGO and open his own load-and-go using file ordinal 56. The block size of a user LGO should be 960 characters. MSOS reassigns file ordinal 56 to the system LGO at the end of the job.

The user may refer to the system scratch files and LGO, using the appropriate file ordinal, in the following statements: LOAD, CLOSE, or the MSIO input/output statements LOCATE, READ, WRITE, and WRITE CHECK.

11.1.6 ALTERING MSOS FILES

The user may alter and add to the MSOS system files, resident, ABS, and LIB, through the PRELIB routine which is contained in the LIB file (See PRELIB, Part III, Section 9).

11.2 MSIO FILES

11.2.1 FILE LABEL DIRECTORY (FLD)

The FLD is used to provide access to all files maintained by MSOS and is composed of two separate files, IDFILE and LABELFILE. An entry is made in each of the FLD files each time a file is created.

LABELFILE (FO 66)

The format of the LABELFILE is shown in Figure III-3 and the entries are described in Table III-7.

LABELFILE entries are fixed-length file labels having a block size of 50 + 3 (MSC) words, where MSC is the maximum allowable segment count (defined by the installation during PRELIB. IDFILE entries consist of file identifiers (words 1-10 of the file label). The IDFILE entries can be thought of as 12-word logical records written in blocks of 120 words each.

The entries for a given file occupy the same relative positions within IDFILE and LABELFILE. Thus, to locate a specific label one can search IDFILE until the desired identification is found, then compute the block number within LABELFILE at which the label is stored.

The first label in LABELFILE is for LABELFILE itself. The labels for IDFILE and MSDFILE occupy blocks two and three respectively, followed by the labels for the four MSOS system files.

A mass storage file label is composed of a fixed 50-word base plus three additional words for each segment of the file. MSIO is capable of processing files which contain up to 63 segments although installations may set the maximum allowable segment count to some value less than 63 if they wish. The labels for all files are stored in a central label directory (called LABELFILE) which is contained on the system disk pack. The format of a file label is shown in Figure III-3 and described in Table III-7.

Word	Contents				
1, 2	Owner Identification				File Identifier
3 – 9	File Name				File Identifier
10	File Name		Edition No.		File Identifier
11	Access Security				
12	Modification Security				
13	*				
14	*	Block Size			
15	Block Count				
16	Usage Count				
17	Creation Date				
18	Expiration Date				
19	Last Access Date				
20	DT	SC	P	*	
21	DTM	*			
22	*				
23	File Size				
24 . . . 49	*				
50	Checksum				
51		Device Number			These 3 words are repeated for each segment of the file.
52	Low Segment Limit				These 3 words are repeated for each segment of the file.
53	Segment Length				These 3 words are repeated for each segment of the file.

* These fields are reserved

Figure III-3. Label File Format

TABLE III-7. FILE LABEL FIELD DESCRIPTION

Field Name	Size	Description
File Identifier	40 Char.	Used to uniquely identify a file in the label directory. The standard identifier consists of: Owner Identification - 8 characters. File Name - 30 characters, Edition Number - 2 characters. The 40-character field may be divided in other ways if an installation chooses to do so.
Access Security	4 Char.	This field is supplied when the file is allocated and must be supplied for each succeeding OPEN request.
Modification Security	4 Char.	This field is supplied when the file is allocated and must be supplied for each RELEASE, EXPAND, and MODIFY request.
Block Size	3 Char.	This field contains, as a binary integer, the number of 6-bit characters in each record block ($0 < \text{Block Size} < 131072$).
Block Count	4 Char.	This field contains, as a binary integer, the highest block number that has been written. If the file is processed sequentially, this is actually the number of blocks written into the file ($0 < \text{Block Count} < 2^{23}$).
Usage Count	4 Char.	A binary count of the number of times the file has been opened.
Creation Date	4 Char.	A date in the form YYMMDD, stored as a binary integer, supplied by the I/O system when the file was allocated.
Expiration Date	4 Char.	A date in the form YYMMDD, stored as a binary integer, supplied by the user when the file was allocated. This field determines when a file may be deleted.
Last Access Date	4 Char.	A date in the form YYMMDD, stored as a binary integer, supplied by the I/O system each time the file is opened or changed.
DT (Device Type)	1 Char.	A 6-bit code to indicate the type of mass storage device on which the file is contained. Octal 40 indicates an 852 disk pack.
SC (Segment Count)	1 Char.	This field contains, as a binary integer, the number of segments in the file ($0 < \text{SC} < 64$).
P (Protection)	1 Char.	This field contains protection flags for use by the I/O system. The only values currently defined are: 0 - The file may be read or written, 1 - The file may not be written.

TABLE III-7. FILE LABEL FIELD DESCRIPTION (cont'd)

Field Name	Size	Description
DTM (Device Type Modifier)	1 Char.	A 6-bit code which provides further device information. The only values currently defined are: XXXXX0 - Track Mode XXXXX1 - Sector Mode
File Size	4 Char.	This field contains, as a binary integer, the number of allocatable units (tracks) assigned to the file ($0 < \text{File Size} < 2^{23}$).
Checksum	4 Char.	A 24-binary checksum of the entire label. This field is checked by the I/O system to detect accidental modification of the label.
Device Number	3 Char.	The number of the device on which this file segment is stored. This field is checked against the device label to insure that proper packs are mounted.
Low Segment Limit	4 Char.	The binary hardware address at which this file segment begins.
Segment Length	4 Char.	The number of allocatable units (tracks) in this segment.
* (Reserved)	117 Char.	These fields are reserved for future use by the I/O System.

ID FILE (FO 65)

The IDFILE entries consist of file identifier (words 1-10 of file label) and security codes (words 11 and 12 of the file label).

1	OWNER
2	IDENTIFICATION
3	
4	
5	
6	FILE NAME
7	
8	
9	
10	EDITION NO.
11	ACCESS SECURITY
12	MODIFICATION SECURITY

The IDFILE entries can be thought of as 12 word logical records written in blocks of 120 words each.

The entries for a given file occupy the same relative positions within IDFILE and LABELFILE. Thus, to locate a specific label, one can search IDFILE until the desired identification is found, then compute the block number within LABELFILE at which the label is stored.

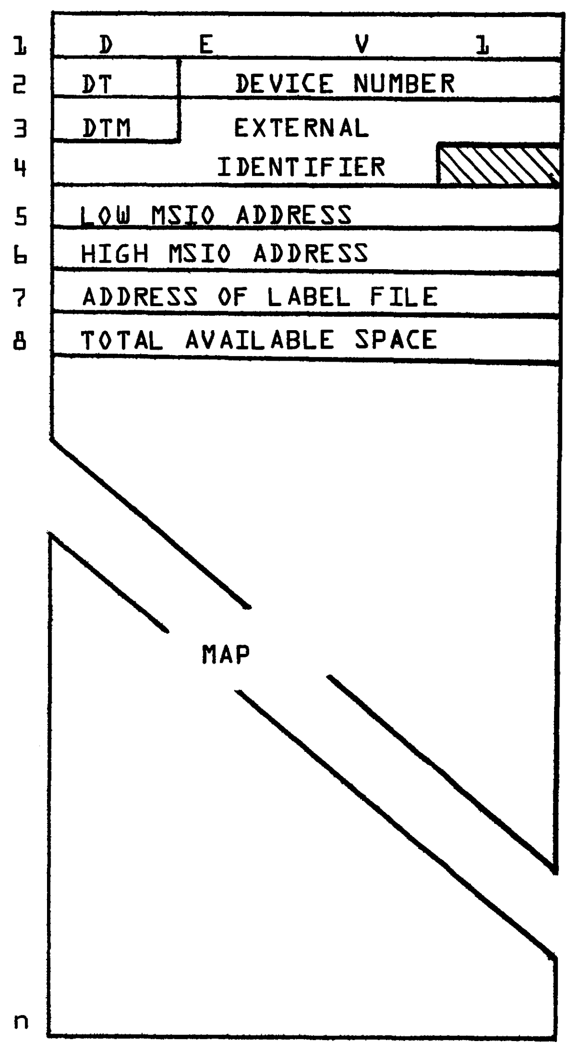
11.2.2 MASS STORAGE DIRECTORY (MSDFILE - FO 64)

The MSD contains one entry for each mass storage device maintained by MSOS. The format of the MSD file is shown in Figure III-4 and file entries are described in Table III-8. Each mass storage device is described in the file by an entry consisting of the following:

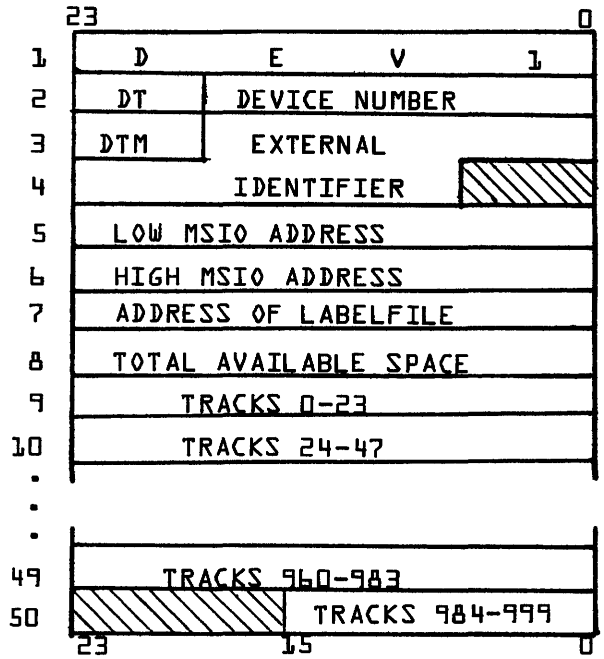
Words 1-7 of the device label

The number of unassigned tracks

A bit map of available and assigned tracks



EXAMPLE OF 853 ENTRY



The correspondence between bits and tracks is:

Bit Number	Word Number	Track Number
0	9	0
1	9.	1
23	9	23
0	10	24
1	10	25
i	j	24 {j-9}+i

Figure III-4. MSD File Format

TABLE III-8. MSD FILE FIELD DESCRIPTION

Field Name	Size	Description
DEV1	4 char.	A standard 4-character identifier which is prefixed to device labels.
DT	1 char.	A 6-bit code to represent device type. DT= octal 40 for 852 disk packs.
DEVICE NUMBER	3 char.	An 18-bit device number which matches an external number on each device.
DTM	1 char.	A 6-bit device type modifier. The only values defined for 852 disk packs are: xxxxx0 - this device is recorded in track mode. xxxxx1 - this dev. is recorded in sector md.
EXTERNAL IDENTIFIER	6 char.	Any alphanumeric chars. This field corresponds to an external ident. on each device.
LOW MSIO ADDRESS	4 char.	The lowest hardware address {bin.} that can be accessed by MSIO.
HIGH MSIO ADDRESS	4 char.	The highest hardware address {binary} that can be accessed by MSIO.
DIRECTORY ADDRESS	4 char.	The binary hardware address at which the file label directory is stored. This is the low address of the LABELFILE and is present only on the device which contains the label directory.

Map of Available and Assigned Tracks

The storage map occupies words 9 through 683 and represents tracks of the mass storage device. That is, the map contains x bits representing x tracks. A bit set to one indicates the corresponding track is assigned. A bit set to zero indicates the corresponding track is available. The values of variables n (number of words per MSD entry) and x (number of bits per map) are defined below:

Normal Capacity

DEVICE TYPE	{x} # TRACKS PER DEVICE	{n} WORD LENGTH OF MSD ENTRY
1311	1000	50
852	1000	50
853	1000	50
854	2000	92
814	12288	520
813	12288	520
863	1024	51
841	4060	178
Devices with extended capacity		
854	2030	93
813	16384	691
814	16384	691

11.2.3 ALTERING MSIO FILES

The user may alter the size of the MSIO files through the PRELIB routine (Part III, Section 9.2.3). Care must be taken to insure the following:

Only the proper file ordinals are used when referring to the MSIO files. The file ordinals are:

MSDFILE	64
IDFILE	65
LABELFILE	66

The BMSIO/MSIO values for MFC (maximum file count) and MSC (maximum segment count) must be consistent with the installation or expanded values. Errors are catastrophic.

The sizes of the LABELFILE and IDFILE must be of the same logical size. MFC must equal the number of entries that can be contained in both files.

Example:

If MSOS was installed with MFC = 100 and MSC = 16 on an 854 system, the FLD files would be of the following size:

LABELFILE	13 tracks
IDFILE	2 tracks

The BMSIO/MSIO values would be:

MFC	100
MSC	16

If the user needs the capability of 500 files he must add space for 400 files. On an 854 the FLD files require the following:

LABELFILE	63 tracks
IDFILE	7 tracks

PRELIB is used to replace BMSIO/MSIO with new values of MFC = 500. The EXPAND card is used to expand LABELFILE by 50 tracks and IDFILE by 5 tracks.

$^7_9$EXPAND,,50,5

The new edition must be autoloaded to create the system FDT showing the expanded capability.

NOTE

The MSC can only be set during an install. It determines the block size of the FLD for the life of the system.

EXAMPLES OF INSTALLATION PROCEDURES 12

12.1 COSY/COMPASS/PLIBEDIT RUN

The following example shows the use of PLIBEDIT, COSY, and COMPASS in generating a MSOS PRELIB source tape. Card and deck handling is kept to a minimum. The following site dependent hardware configuration is assumed:

Standard MSOS user
Floating-point hardware available; no BCD or BDP hardware.
32K 3100 computer with the following configuration:

Equipment type	Quantity	Controller	Channel	Equipment	Unit
console type-writer	1	none	none	none	none
854 disk drive	2	3234	2	0	10, 11
405 card reader	1	3248	1	0	0
501 printer	1	3256	1	1	0
415 card punch	1	3446	0	1	0
604 tape units	4	3228	0	0	0, 1, 2, 3

Part I shows a COSY job incorporating site dependent assembly options into MSIO and SCICREC1. Hollerith output is on the system scratch file 54.

Part II calls COMPASS. Input is from system scratch file 54. Output is an assembly list, cross reference table, and the binary decks of MSIO and SCICREC1 on system scratch file 56.

Part III uses PLIBEDIT to modify the Standard PRELIB source on binary release tape 1, LUN 01, and places the output on LUN 02. This is called the PRELIB source tape. LUN 01 and 02 are equipped through a LED card within the library program PLIBEDIT. The created PRELIB source tape may be assigned as standard input for a PRELIB run generating a MSOS system.

$^7_9$SEQUENCE, 416 PART I

$^7_9$JOB, , , ,

$^7_9$CTO, LUN03 IS COSY TAPE NO. 1

$^7_9$CTO, LUN01 IS BINARY RELEASE TAPE NO. 1

$^7_9$CTO, LUN02 IS NEW PLIBEDIT (SOURCE) TAPE

$^7_9$EQUIP, 03=MT

```
7
9COSY
            DELETE/     23
RATL        EQU         2
            DELETE/     52,55
MST         OCT         01000000,51000000
            OCT         12000000,51000000
MSIO        DECK/       I=03,H,R
            DELETE/     84
MAXCORE     EQU         77637B
            DELETE/     85
NCHANS      EQU         4
            DELETE/     2207,2223
            AET         (DP040,0,0,MSIO3234,RES)
            AET         (TY,,,,DRIVER05,RES)
            AET         (CR,100,0,0,DRIVER02,RES)
            AET         (PR,100,1,0,DRIVER03,RES)
            AET         (CP,200,1,0,DRIVER04,RES)
            AET         (MT,200,0.3,DRIVER01)
            AET         (MT,200,0,2,DRIVER01)
            AET         (MT,200,0,1,DRIVER01)
            AET         (MT,200,0,0,DRIVER01)
            AET         (DP,040,1,1,MSIO3234)
```

	DELETE/	2372, 2379	
	00	IOIP	CH 0
	00	IOIP	CH 1
	00	IOIP	CH 2
	00	ABINRT	CH 3
	00	ABINRT	CH 4
	00	ABINRT	CH 5
	00	ABINRT	CH 6
	00	ABINRT	CH 7
	DELETE/	2412	
SYST. IOM	OCT	7	CH 0, 1, 2
SCICREC1	DECK/	I=03, H, R	
	ENDCOSY/		END OF PART I
$^{7}_{9}$COMPASS, I=54, L, R, X			PART II
$^{7}_{9}$PLIBEDIT			PART III
	POSITION/	STD	
CIO	SKIP/	1	
MSIO	FILE/	56	
DRIV606	DELETE/	DRIV3649	
DRIV3555	DELETE/		
DRIV3245	DELETE/	MSIO3232	
MSIO3553	DELETE/	DRIV3691	
	SKIP/	1	
SCICREC1	FILE/	56	
	LOCATE/	SEPONT, FDPBOXS	
	SKIP/	1	
POSTLOAD	LOCATE/		
	POSITION/	COMMON	ABS
	DELETE/	BCD	
	DELETE/	BDP	
	POSITION/	COMMON	REL

ENDEDIT/ END OF PART III

${}^{77}_{88}$EOF

${}^{7}_{9}$ENDSCOPE

PLIBEDIT rewinds the PRELIB source tape. After the monitor reads the ENDSCOPE card,the operator responds with the following:

Press: MANUAL INTERRUPT

Type: GO

Press: MANUAL INTERRUPT

The monitor permits the operator to reassign the standard unit, equip the PRELIB source tape and run PRELIB to generate a new MSOS system.

12.2 PLIBEDIT/MSOS CONTROL CARD EXAMPLES

The following deck structures illustrate the use of PLIBEDIT/MSOS control cards.

The user must EQUIP or ALLOCATE and OPEN all needed nonstandard files except LUN 01 and 02.

Sample 1 replaces the binary deck of CIO and PRELOAD.

${}^{7}_{9}$JOB,4JL52,DLW,10 SAMPLE 1

${}^{7}_{9}$PLIBEDIT

CIO REPLACE/

binary deck of CIO

PRELOAD REPLACE/

binary deck of PRELOAD

ENDEDIT/

${}^{77}_{88}$

Sample 2 skips the first file on CPF and deletes control cards for FORTRAN and FORTRAN compiler routines.

${}^{7}_{9}$JOB,,, SAMPLE 2

${}^{7}_{9}$PLIBEDIT

COMMENT/ SKIP FIRST FILE OF INP

LOCATE/ EOF=1

COMMENT/ DELETE FORTRAN COMPILER AND CONTROL CARDS

```
ALG5               SKIP/          2

FTN1               DELETE/        FTNE

                   COMMENT/       NEW FORTRAN REV.  6*7*68

control cards and binary decks of FORTRAN compiler

                   ENDEDIT/

77
88
```

Sample 3 replaces macros named COMAC and inserts the routines PROG1 and PROG2 in front of SIPP.

```
7
9JOB,4JL52,DLW,10                              SAMPLE 3
7
9PLIBEDIT

                   COMMENT/       NEW MACROS

MACRO              REPLACE/       MACRO, COMAC

7
9MACRO,COMAC

deck of macros named COMAC

SIPP               INSERT/

binary deck of PROG 1

binary deck of PROG 2

                   ENDEDIT/

77
88
```

Sample 4 deletes SEPOINT, SETCLOCK, inserts SEPOINT, ACCOUNT1 before SEPOINT, START2, skips two cards after SEPOINT, UST, inserts SEPOINT, FDPBOXS, and deletes BSIPP routine.

```
7
9JOB, 4JL52,DLW,10                             SAMPLE 4

7
9PLIBEDIT

                   DELETE/        SEPOINT,SETCLOCK

                   INSERT/        SEPOINT, START2

7
9SEPOINT, ACCOUNT1

                   LOCATE/        SEPOINT,UST

                   SKIP/          2
```

$^{7}_{9}$SEPOINT, FDPBOXS

BSIPP DELETE/

ENDEDIT/

$^{77}_{88}$

Sample 5 replaces the old PRELIB control card, deletes binary decks MSIO and SCICREC1, inserts decks MSIO and MCICREC1, inserts SEPOINT, OPTBOXS before SEPOINT, BCDBOXS, and adds routine COPYTAPE after the COSY routine.

$^{7}_{9}$JOB, , , SAMPLE 5

$^{7}_{9}$PLIBEDIT

REPLACE/ PRELIB, , , , , xx

$^{7}_{9}$PRELIB, , , P, , 42

MSIO DELETE/

binary deck of MSIO

SCICREC1 DELETE/

binary deck of MCICREC1

INSERT/ SEPOINT, BCDBOXS

$^{7}_{9}$SEPOINT, OPTBOXS

COSY LOCATE/

binary deck of COPYTAPE

ENDEDIT/

$^{77}_{88}$

Sample 6 deletes MSSORT, accepts input from LUN 12 until encountering an EOF, deletes MSSMERG, and accepts MSSMERG from LUN 15.

$^{7}_{9}$JOB, 4JL52, DWL, 10 SAMPLE 6

$^{7}_{9}$EQUIP, 12=MT, 15=MT

$^{7}_{9}$PLIBEDIT

MSSORT DELETE/

UNIT/ 12

MSSMERG DELETE/

```
MSSMERG         UNIT/           15
                ENDEDIT/

77
88
```

Sample 7 deletes COSYRDWT through to COSY, accepts input from LUN 10 (mass storage file) until encountering an EOF, deletes SFP and accepts the binary image of SFP from file ordinal 12 (LUN 12).

```
7
9JOB,4JL52,XCL,875                                    SAMPLE 7

7
9FET,XCL,COSY,960

7
9OPEN,10

7
9FET,XCL1,SFP,960

7
9OPEN,12

7
9PLIBEDIT

COSYRDWT        DELETE/         COSY
                FILE/           10
SFP             DELETE/
SFP             FILE/           12
                ENDEDIT

77
88
```

Sample 8 replaces RAAR and SCAR whenever they appear

```
7
9JOB,4JL52,DWL,10                                     SAMPLE 8

7
9EQUIP,10=MT

7
9PLIBEDIT

UNIT            MCHANGE/        10
binary RAAR deck
binary SCAR deck
                COMMENT/        RAAR/SCAR REPLACED 01/23/69
                ENDEDIT

77
88
```

12.3 COSY CORRECTION SAMPLE OF SYSTEM WITH CARD PUNCH

The coding and control cards necessary to punch binary decks for insertion into the PRELIB source are shown below. The example does not represent a complete update (in this instance, OC and AREM are not modified). This example assumes the following system.

1. Floating point hardware
2. No BDP hardware
3. No clock interrupt processor (CIP) or system accounting (ACCT)
4. A 32K 3300 with the following configuration:

Equipment Type	Quantity	Controller	Channel	Equipment	Unit
Console Typewriter	1	none	none	none	none
853 Disk Drive	4	3234	0	4	0-3
415 Card Punch	1	3245	1	4	0
501 Printer	1	3659	0,1	5	0
405 Card Reader	1	3248	1	3	0
607 Magnetic Tape Drive	4	362X	2,3	4	0-3

All COSY/COMPASS updates which follow are set up to execute under Tape SCOPE, Real-Time SCOPE, or the interim MSOS library, if magnetic tapes are available.

Card Description			Comments
$^{7}_{9}$SEQUENCE,001,MODIFY MSIO			
$^{7}_{9}$JOB,12345,ABC			
$^{7}_{9}$EQUIP,01=MT, 02=MT			LUN 01 MSOS COSY tape
$^{7}_{9}$COSY			LUN 02 Hollerith output tape
	DELETE/	23	
RATL	EQU	4	The number of MST entries
	DELETE/	52,55	
MST	OCT	01000000,50000000	Set up the mass storage table to
	OCT	06000000,50000000	correspond to the hardware
	OCT	07000000,50000000	environment defined in
	OCT	10000000,50000000	MCICREC1 (MP MSOS)

Card Description			Comments
MSIO	DECK/ ENDCOSY/	I=01, H=02	Assemble and punch MSIO deck; this example allows no modification for: • Number of words for Batch standard units on mass storage, and priority file description labels • Privacy access code • Maximum number of tracks allowed per file Changes may be made if desired.
$^{7}_{9}$COMPASS, I=2, L, R, P			
$^{77}_{88}$end-of-file			
$^{7}_{9}$SEQUENCE, 002, MODIFY MCICREC1			
$^{7}_{9}$JOB, 12345, ABC			
$^{7}_{9}$EQUIP, 01=MT, 02=MT			LUN 01 = MSOS 4.2 COSY tape 1 LUN 02 = Hollerith output tape
$^{7}_{9}$COSY			The following changes apply to Memory Protect users exclusively
	DELETE/	109	
ACCT	EQU	0	No system accounting
	DELETE/	110	
BDP	EQU	0	BDP hardware not available
	DELETE/	111	
CLOCK	EQU	0	No clock interrupt processor
	DELETE/	112	
FDP	EQU	1	Floating-point hardware available
	DELETE/	113	
MAXCORE	EQU	77737B	32K 3300 core size
	DELETE/	114	
MP	EQU	0	3300 Memory Protect variant
	DELETE/	115	
NCHANS	EQU	4	Number of channels available
	DELETE/	116	
USARDT	EQU	1	USASI COBOL will be used

	Card Description		Comments
		DELETE/ 3271/3287	
	AET	(DP,200,4,0,MSIO3234,RES)	Library CH 0, Eq 4, Un 0
	AET	(TY,,,,DRIVER05,RES)	CTO/CFO Console Typewriter
	AET	(CR,100,3,0,DRIVER02,RES)	Standard Input Ch 1, Eq 3, Un 0
	AET	(PR,300,5,0,DRIVER03,RES)	Standard Output Ch 0/1, Eq 5, Un 0
	AET	(CP,100,4,0,DRIVER04,RES)	Standard Punch Ch 1, Eq 4, Un 0
	AET	(DP,200,4,1,MSIO3234)	Available disk Ch 0, Eq 4, Un 1
	AET	(DP,200,4,2,MSIO3234)	Available Disk Ch 0, Eq 4, Un 2
	AET	(DP,200,4,3,MSIO3234)	Available Disk Ch 0, Eq 4, Un 3
	AET	(MT,60,4,0,DRIVER01)	Mag Tape Ch 2/3, Eq 4, Un 0
	AET	(MT,60,4,1,DRIVER01)	Mag Tape Ch 2/3, Eq 4, Un 1
	AET	(MT,60,4,2DRIVER01)	Mag Tape Ch 2/3, Eq 4, Un 2
	AET	(MT,60,4,3,DRIVER01)	Mag Tape Ch 2/3, Eq 4, Un 3
	DELETE/	3354/3360	
RHT	OCT	0,0,0,0	Set up RHT to match AET+
	OCT	0,0,0,0	
	OCT	0,0,0,0	
	OCT	0,0,0202,03040500	
	DELETE/	3438,3445	
BRHT	OCT	0,0,0,0	Set up BRHT to match AET+
	OCT	0,0,0,0	
	OCT	0,0,0,0	
	OCT	0,0,0202,0	

+ Within this run, these changes are given only to show RHT or BRHT formats. Since the AET is in suggested release order, the changes replace identical release values.

Card Description			Comments
			CIT modification IOIP channel available ABINRT channel not available
	DELETE/	3438, 3445	
	00	IOIP	Channel 0, available
	00	IOIP	Channel 1, available
	00	IOIP	Channel 2, available
	00	IOIP	Channel 3, available
	00	ABINRT	Channel 4, not available
	00	ABINRT	Channel 5, not available
	00	ABINRT	Channel 6, not available
	00	ABINRT	Channel 7, not available
MCICREC1	DECK/	I=01, H=02	Merge corrections with job file corrections
	ENDCOSY/		

$^{7}_{9}$COMPASS, I=2, L, R, P

$^{77}_{88}$end-of-file

12.4 COSY CORRECTION SAMPLE OF PUNCHLESS SYSTEM

The following example demonstrates the coding and control cards necessary to generate decks on a punchless system. It does not represent a complete update (in this instance, OC and AREM are not modified). Example two assumes the following system.

1. A BATCH MSOS user

2. A 16K 3100 Computer with the following configuration.

Equipment Type	Quantity	Controller	Channel	Equipment	Unit
Console Typewriter	1	none	none	none	none
854 Disk Drive	2	3234	0	4	0-1
405 Card Reader	1	3649	1	3	0
501 Printer	1	3659	1	5	0

The COSY/COMPASS updates which follow execute under the interim MSOS library.

Card Description			Comments
$^{7}_{9}$SEQUENCE, 001, MODIFY BMSIO			
$^{7}_{9}$JOB, 12345, ABC			
$^{7}_{9}$FET, COSY, HOLLERITH-FILE, 512			Hollerith file contains the Hollerith output
$^{7}_{9}$ALLOCATE, 50, , S, , 854			
$^{7}_{9}$OPEN, 01			
$^{7}_{9}$FET, COMPASS, BINARY-FILE1, 960			Binary file 1 contains the binary of BMSIO
$^{7}_{9}$ALLOCATE, 20, , S, , 854			
$^{7}_{9}$OPEN, 50			
$^{7}_{9}$COSY			
	DELETE/	21	
RATL	EQU	2	Number of MST entries
	DELETE/	46, 49	
MST	OCT	01000000, 51000000	Set up the mass storage table to correspond to the hardware environment defined in BCICREC1 (BATCH MSOS)
	OCT	05000000, 51000000	
BMSIO	DECK/	H=01	
(insert the COSY deck of BMSIO)			
	ENDCOSY/		
$^{7}_{9}$COMPASS, I=01, L, R, X=50			
$^{7}_{9}$CLOSE, 50			
$^{77}_{88}$end-of-file			This update is not complete. There are no changes in the maximum file size, the privacy code, etc. They may be changed if desired.
$^{7}_{9}$SEQUENCE, 002, MODIFY BCICREC1			
$^{7}_{9}$JOB, 12345, ABC			
$^{7}_{9}$FET, COSY, HOLLERITH-FILE, 512			Hollerith file contains the Hollerith output
$^{7}_{9}$OPEN, 01			
$^{7}_{9}$FET, COMPASS, BINARY-FILE2, 960			Binary file 2 contains the binary of BCICREC1
$^{7}_{9}$ALLOCATE, 20, , S, , 854			
$^{7}_{9}$OPEN, 51			
$^{7}_{9}$COSY			

	Card Description		Comments
	DELETE/	81	
ACCT	EQU	0	No system accounting
	DELETE/	82	
BDP	EQU	0	BDP hardware not available
	DELETE/	83	
CLOCK	EQU	0	No clock interrupt processor
	DELETE/	84	
MAXCORE	EQU	37637B	16K 3100 core size
	DELETE/	85	
NCHANS	EQU	4	Number of channels available
	DELETE/	86	
SETCLK	EQU	0	No setclock option
	DELETE	87	
USARDT	EQU	1	USASI COBOL will be used
	DELETE/	805,812	
IOT	00	IOIP	Channel 0, available
	00	IOIP	Channel 1, available
	00	IOIP	Channel 2, not available
	00	ABINRT	Channel 3, not available
	00	ABINRT	Channel 4, not available
	00	ABINRT	Channel 5, not available
	00	ABINRT	Channel 6, not available
	00	ABINRT	Channel 7, not available
	DELETE/	1499,1515	
AET	(DP,200,4,0,MSIO3234,RES)		Library Ch 0, Eq 4, Un 0
AET	(TY,,,,DRIVER05,RES)		CTO/CFO Console Typewriter
AET	(CR,100,3,0,DRIVER02,RES)		Standard Input Ch 1, Eq 3, Un 0
AET	(PR,100,5,0,DRIVER03,RES)		Standard Output Ch 1, Eq 5, Un 0
AET	(DP,200,4,1,MSIO3234)		Available Disk Ch 0, Eq 4, Un 1

Card Description			Comments
	DELETE/	1580, 1586	
RHT	OCT	0, 0, 0, 0	Set up RHT to match AET†
	OCT	0, 0, 0, 0	
	OCT	0, 0, 0, 0	
	OCT	0, 0, 0202, 03040500	
	DELETE/	1625	
SYST. IOM	OCT	7	Set up system I/O mask to match available channels
BCICREC1	DECK/	H=01	

(insert the COSY deck of BCICREC1)

ENDCOSY/

$^{7}_{9}$COMPASS, I=01, L, R, X=51

$^{7}_{9}$CLOSE, 51

$^{77}_{88}$end-of-file

† The AET does not define PUN. Therefore, LUN 62 in RHT is set to 00. Since AET was not described as recommended these changes must appear.

MASS STORAGE REQUIREMENT TABLE 13

The following is a guide to determine what library options buy the most for the space occupied on the system disk. For safety, one sector has been added to all equations. Remainders of calculated values are ignored. PRELIB creates system files blocked according to sector size. Calculations using the following equations are therefore relative to sectors.

13.1 MASS STORAGE REQUIREMENT TABLE (MSRT) EQUATIONS

The following equations are dependent upon the type of library program being considered. The appropriate equation determines the number of sectors occupied.

13.1.1 ABSFILE OCCUPATION

$$\text{sectors} = \left(\frac{\text{program length - decrement}}{\text{words per sector}} + 1 \right)$$

The decrement is indicated by the I parameter on an ORIGIN card.

Example of routine ALG1 follows:

sector size	25 words (852 specification)
program length	10731_8
decrement	74_8

$$\text{sectors} = \left(\frac{10731_8 - 74_8}{31_8} + 1 \right) = 181 \text{ sectors}$$

13.1.2 LIBFILE OCCUPATION

$$\text{sectors} = \left(\frac{\text{number of binary cards in program}}{6} + 1 \right) \left(\frac{240}{\text{sector size}} + 1 \right)$$

The number of binary cards in a program includes LED, overlay, LRL, EXS, and BCT cards.

Example of routine PLOVINT follows:

sector size 25 words (852 specification)

number of cards 40

$$\text{sectors} = \left(\frac{40}{6} + 1\right)\left(\frac{240}{25} + 1\right) = 70 \text{ sectors}$$

13.1.3 MACRO OCCUPATION

$$\text{sectors} = \left(\frac{\text{number of macro cards}}{10} + 1\right)\left(\frac{240}{\text{sector size}} + 1\right)$$

The following exemplifies calculation for a given macro:

sector size 25 words (852 specification)

number of cards 4

$$\text{sectors} = \left(\frac{4}{10} + 1\right)\left(\frac{240}{25} + 1\right) = 10 \text{ sectors}$$

13.2 PRODUCT SET OCCUPATION

Table III-9 may be used to estimate the number of tracks product set members will occupy. Table III-10 lists the storage capacity of mass storage devices usable on the system.

TABLE III-9. MASS STORAGE SPACE REQUIRED FOR MSOS PRODUCT SET MEMBERS

PRODUCT	853/854/863 (841)†		852	
	ABSFILE	LIBFILE	ABSFILE	LIBFILE
SYSTEM[1] (RESFILE)	35		47	
BASIC LIBFILE[2]		86 (45)		195
ABSFILE[3] LERP = 0	86 (44)		175	
ABSFILE[3] LERP = 1	91 (46)		185	
USASI COBOL	48 (25)	27 (14)	151	54
RESPOND		14 (7)		26
ALGOL	21 (11)	15.5 (8)	43	31
SIPP		5 (3)		10
BSIPP		4.5 (2.5)		9
SFP		3.5 (2)		7
MS FORTRAN	30 (16)	27 (14)	60	54
MS COBOL		49 (25)		
MSSORT		20 (11)		40
TAPESORT		36 (19)		71
LISA		(8 +2.5) 15 + 4.25[4]		27 + 8.5[4]
MSOSUTIL		7.5		15
LOGICAL MSIO	11 (5)	6.5 (3.5)	8.5	13
SNAPSHOT		3 (1.5)		6
COMPASS	13.25 (6.75)	(.25 + 10.5) .50 + 21.75[4]	28	1.0 + 42.5[4]
COSY		6.5 (3.5)		13
SAINT		2.5 (1.5)		6
ERROR RECOVERY	1.75 (1.10)	12 (6.5)	3.5	25
USASI FORTRAN	OVERLAY FO49 57	75[5]	OVERLAY FO49 120	165[5]

† All numbers in parentheses refer to the 841.

Superscripts indicated have the following meanings.

1 The system size assumes the following:

	852	854
MP resident, 5 standard drivers	20	8
MSDFILE, IDFILE, LABELFILE, DRS, label	27	27
Total tracks	47	35

2 The basic relocatable file contains only routines essential to the system. The interim library PRELIB source shows these routines.

3 An additional 28 absolute records, includes ALGOL, as in standard PRELIB sources. Symbol LERP of the variable resident routine LOADER incorporates error recovery package.

4 Indicates macro adjustment

5 This does not take into account LIBFILE reduction due to removal of MS FORTRAN object time routines.

TABLE III-10. DEVICE STORAGE CAPACITIES

DEVICE	WORDS PER SECTOR	SECTORS PER DEVICE	SECTORS PER TRACK	WORDS PER TRACK
852	25	20,000	20	500
853	64	16,000	16	1,024
854	64	32,000	16	1,024
863	64	16,284	16	1,024
841	160	56,000	14	2,240

COSY UPDATE HISTORY FILE 14

14.1 COSY,* FUNCTION

COSY, * functions in the same manner as COSY, with the following exceptions:

The * must be used in place of the / on COSY control cards

An END* control card, not an ENDCOSY/ card, terminates a COSY run. END* must begin in column 10.

The dummy control card (INSERT/ 0) appears on listings when the L parameter, list function, is selected on a DECK/ card. The card (INSERT/ 0) has COSY sequence number of one. This permits the addition of revision cards ahead of cards not listed. To do so the user references COSY sequence number one. The new COSY ignores this card when a Hollerith revision tape is used as input.

COSY, * output may be used as input to COSY. An ENDCOSY/ control card used to terminate this COSY run is obtained on the COSY, * output by inputting as the last DECK* card the following.

```
ENDCOSY/      DECK*        I=xx, H=yy
```

This DECK* card will output an ENDCOSY/ control card.

14.2 LISTING THE COSY UPDATE HISTORY FILE

To obtain a listong of the COSY Update History file, insert a DECK* control card for each routine listed on the deck name list of the update history file. An example of listing two such routines follows.

$^{7}_{9}$SEQUENCE,403

$^{7}_{9}$JOB,,,

$^{7}_{9}$CTO,   LUN 03 = THE COSY UPDATE HISTORY FILE COSY TAPE

$^{7}_{9}$EQUIP,03-MT

$^{7}_{9}$COSY,8

```
MZERO         DECK*        I=03, L

ZERO          DECK*        I=03, L
insert additional DECK* cards

              END*
```

$^{77}_{88}$ EOF

Output from the previous example follows. Underlined cards are the control cards used to create the listing. All other cards are images which appear on tape.

```
SEQUENCE,403
JOB,,,,
EQUIP,03=MT
COSY,*
MZERO       DECK*       I=03,L
            INSERT/     0                                     00001
            INSERT/     7                                     00002
BDP3312     EQU         0                                     00003
            DELETE/     66,67                                 00004
            IFZ         BDP3312-2,17                          00005
*-------------BDP COLLATING SEQUENCE---------------           00006
            OCT         65666770                              00007
            OCT         71727374                              00008
            OCT         75766526                              00009
            OCT         27010203                              00010
            OCT         17313233                              00011
            OCT         34353637                              00012
            OCT         40413014                              00013
            OCT         15121316                              00014
            OCT         22434445                              00015
            OCT         46475051                              00016
            OCT         52534220                              00017
            OCT         21071011                              00018
            OCT         00235556                              00019
            OCT         57606162                              00020
            OCT         63645424                              00021
            OCT         25040506                              00022
MZERO       DECK/       I=1, H=2                              00023
ZERO        DECK*       I=03,L
```

```
          INSERT/    0                                        00001
          INSERT/    2                                        00002
BDP3312   EQU        0                                        00003
          DELETE/    45,46                                    00004
          IFZ        BDP3312-2,17                             00005
*-------------BDP COLLATING SEQUENCE-------------00006
          OCT        65666770                                 00007
          OCT        71727374                                 00008
          OCT        75766526                                 00009
          OCT        27010203                                 00010
          OCT        17313233                                 00011
          OCT        34353637                                 00012
          OCT        40413014                                 00013
          OCT        15121316                                 00014
          OCT        22434445                                 00015
          OCT        46475051                                 00016
          OCT        52534220                                 00017
          OCT        21071011                                 00018
          OCT        00235556                                 00019
          OCT        57606162                                 00020
          OCT        63645424                                 00021
          OCT        25040506                                 00022
ZERO      DECK/      I=1,H=2                                  00023
          END*
```

14.3 USING THE COSY UPDATE HISTORY FILE OUTPUT

Once a listing is obtained, site modifications may be applied to the COSY Update History file. COSY sequence numbers on the MSOS COSY tape may be easily located for insertion of site modifications.

Use the COSY sequence numbers found on the right side of the output listing for INSERT* and DELETE* control card parameters.

The following deck structure shows how these revisions may be made and a Hollerith tape of the combined COSY revisions created.

```
7
9SEQUENCE,404
7
9JOB,,,
7
9CTO, LUN 03 = COSY UPDATE HISTORY FILE
7
9CTO, LUN 04 = OUTPUT UNIT FORHOLLERITH REVISION TAPE
7
9EQUIP,03=MT,04=MT
7
9COSY,*
             INSERT*     3
             INSERT/     10
*                              INSTALLATION MODIFICATION INSERTED
             DELETE*     22
*                              RELEASE MODIFICATION REMOVED
MZERO        DECK*       I=03,H=04,L,R
             INSERT*     3
             INSERT/     3
*                              INSTALLATION MODIFICATION INSERTED
             DELETE*     22
*                              RELEASE MODIFICATION REMOVED
ZERO         DECK*       I=03,H=04,L,R
ENDCOSY/     DECK*       I=03,H=04
             END*
77
88 EOF
```

Once the Hollerith revision tape is created by the previous job, the COSY,I option allows its use as input to COSY.

The following example using COSY,I should follow:

```
7
9SEQUENCE,405
7
9JOB,,,
7
9CTO, LUN 01 = MSOS COSY TAPE 1
7
9CTO, LUN 02 = SCRATCH TAPE FOR HOLLERITH OUTPUT
7
9CTO, LUN 04 = HOLLERITH REVISION TAPE
7
9COSY,I=04
7
9COMPASS,I=02,L,R,P
77
88 EOF
```

When assembling from a Hollerith tape consisting of deck names contained on different COSY tapes, the following job sequence should be followed:

```
7
9SEQUENCE,406
7
9JOB,,,,
7
9EQUIP, 01=MT,02=MT,04=MT,05=MT
7
9CTO, LUN 01 = MSOS COSY TAPE #1
7
9CTO, LUN 02 = SCRATCH TAPE FOR HOLLERITH OUTPUT
7
9CTO, LUN04 = HOLLERITH REVISION TAPE
7
9CTO, LUN 05 = MSOS COSY TAPE #2
7
9COSY,I=04
7
9COMPASS,I=02,L,R,P
77
88EOF
```

The output from this job will be the revised routines inputted as DECK/ cards in the form of a listing, cross reference table, and binary decks.

The following deck names will generate a DECK/ card with the parameters I=01, H=02.

MZERO	POSTLOAD	BCDBOXS†	T.NOTRDY†	TAPEHAND
ZERO	AUTOLOAD†	OPTBOXS†	TYPEOUT.†	PAUSE
CIO	INITIAL†	BSIPP	WHATKIND†	CONTROL
BCIO	OC†	SIPP	COSYRDWT	CIO.MSIO
DRIVLABT†	AREM†	PRELIB	COSY2.0	LOCATE†
DRIV3649†	MISC†	PLOVINT	COMPASSB	DOUBLE
DRIV3248†	EXECOVR†	OV1	COMPASS	DFPRIME†
DRIV3659	MSUTIL†	PAHSE1	OVERLAY1	DFP†
DRV512	UDUMP†	PHASE2	PASSONE	Q1QADRI†
DRIV3644	LOAD†	PLOVINT1	PASSTWO	ITOJ†
DRIVTYWR†	ENTER†	PLOV2	SYMTBLE†	ITOX†
DRIV3245†	DMSD†	PLOV3	CRT	XTOI†
DRIVER12	PSM†	RAAR†	FT1	POWRF†
DRIV3293	PSM2†	SCAR	FT2	SINCOS†
DRIV3691	FLL2†	MTWPR†	FT3	ATANF†
MSIORADD†	FLL†	MTRPR†	FT4	EXPF†
MSIO3234†	PFLD†	MTLDACPR†	FT5	LOGF†
MSIO3436†	CLRDSK†	MTDER†	FT6	SIGNF†
MSIO†	BDTRCK†	PRCPR†	FTE	SQRTF†
BMSIO†	WTADR†	CRDER†	FTN	ABSF†
MCICREC1	MSDLRW†	CPDER†	FLOVER†	EXTREMA1†
SCICREC1	PRINT3†	MTWPRR†	BCDINP†	EXTREMA2†
BCICREC1	BINTOBCD†	KTRPRR	BCDOUT	FLOATF†
RDUMP	SPLTDATE†	MTLDCPRR†	FORMAT†	FIXF†
PRELOAD	DTB†	CMNRTNS†	BINARY	MASKINGF
MSIOCCP	IOPACK†	WHATISIT†	BUFFER	FAULTS†
LOADER	IODRAIN†	NRD†	UNIT	SNESLITE†
OVPRO	SNAPSHOT†	NRC	IOCHK†	SENSWTCH†
PROTECT	FDPBOXS	NWR†	EOFCHK†	Q8QERROR†

† Title change only.

The following deck names will generate a DECK/ card with the parameters I=05, H=02.

MSIOMAIN	DPMULDIV†	PCOBOLP2	PZIPPER†	SRTRBALF†
MSIOTPOV	STRIPPER†	PCOBOLDP	PLOGICAL†	SRTRPOLF†
MSIOMSOV	BSTRIPPR†	PCOBOLO	MSSORT	SRTRBALB†
MRESTART	SUBSCRP†	PERRSTOP†	MDYNALL	SRTRPOLB†
COBOL	DEEDIT†	PTYPELOO†	MSSEDIT	SRESTART†
COBOLIE	EXAMINE†	PFIGCON†	MSSIOP1	RESTART1†
COBOLD1	NUMERIC†	PCOMPARE	MSSINTS	RSTRTDUM†
COBOLP1	ALPHABET†	PMULTIPL†	MSSIOP2	SORTIOP2
COBOLD2	ACCEPT†	PDIVIDE†	MSSMERG	SRTEQUAL
COBOLP2	DISPLAY†	PSTRIPPE†	SORT	POLYFORW†
COBOLDP3	MVFIGCON†	PSUBSCRP†	TDYNALL	BALCFORW
COBOLIO	VARC1†	PDEEDIT†	BINANDEC†	BALCBACK
TRANSMIT†	VARN†	PEXAMINE†	SDUMP†	POLYBACK†
FIGCON†	VARAN†	PNUMERIC†	SRTPRINT†	WAITBEEP†
COMPARE†	ROUNDER†	PALPHABE†	SORTPOLY†	ALG1†
EDIT†	CONVERT	PACCEPT†	SRTMBALF†	ALG2†
EDITCOBL†	ZIPPER†	PDISPLAY†	SRTMBALB†	ALG3
MULTIPLY†	LOGICAL†	PMVFIGCO†	SRTMPOLF†	ALG4
DIVIDE†	PCOBOL	PVARC1†	SRTMPOLB†	ALG5
BMULTPLY†	PCOBOLIE	PVARN†	SORTPDMY†	ALGLIB00
BDIVIDE†	PCOBOLD1	PVARAN	SORTEDIT	ALGLIB01
ERRSTOP†	PCOBOLP1	PDPBINBC†	SORTIOP1†	ALGLIB02
TYPELOOP†	PCOBOLD2	PDPBCDBI†	SORTPHS1	ALGOLRUN

The following deck names will generate a DECK/ card with the parameters I=06, H=02.

EXPANDSF†	ISGET	BUFDEF	FD	VERIFY
REPLACE†	BUILD	FILEDEFF†	MSOSUTIL	DUMP†
DELETE†	BLOCKER	RECDEFF†	COPYS	ERROR
INSERT	FILEDEF†	BUFDEFF	COPYT	PLIBEDIT
UPDATE	RECDEF†			

† Title change only.

APPENDIX

ASSEMBLY REFERENCE GUIDE

A

The following is a reference guide to assemble configuration-dependent routines. Routines which must be modified by all installations are listed.

A.1 MS10 (STD, MP)

COSY NO.	1	10	20 (Released Value)	40
23	RATL	EQU	4	number of MST entries
25	FFDTL	EQU	250	BATCH words reserved
26	BFDTL	EQU	125	priority words reserved
27	SFDTL	EQU	165	sums words reserved
41		BCD	2,	privacy code
42		DEC	1	AETD
45		DEC	3996	maximum tracks/file
46		DEC	16	maximum segment count
47		DEC	100	maximum file count
52, 55	MST	OCT	modifications	

A.2 BMS10 (BATCH)

COSY NO.	1	10	20 (Released Value)	40
21	RATL	EQU	4	number of MST entries
23	FFDTL	EQU	250	BATCH words reserved
38		BCD	2,	privacy code
39		DEC	1	AETD
42		DEC	3996	maximum tracks/file
43		DEC	16	maximum segment count
44		DEC	100	maximum file count
46, 49	MST	OCT	modifications	

A.3 MCICRECI (MP)

COSY NO.	1	10	20(Released Value)	40
109	ACCT	EQU	0	accounting option
110	BDP	EQU	0	BDP module
111	CLOCK	EQU	0	clock interrupt processor
112	FDP	EQU	0	floating-point hardware
113	MAXCORE	EQU	77737_8	machine core size
114	MP	EQU	0	3300/3500 memory protect
115	NCHANS	EQU	4	highest channel available
116	USARDT	EQU	0	NON-USASI USER
121	INP	EQU	HOLL	ASCII input option
123	PUN	EQU	HOLL	ASCII punch option
124	PT	EQU	HOLL	ASCII paper tape option
3271,3287				AET
3354,3360	RHT	OCT	modifications	
3368,3371	BRHT	OCT	modifications	
3438		00	IOIP	channel 0
3439		00	IOIP	channel 1
3440		00	IOIP	channel 2
3441		00	IOIP	channel 3
3442		00	ABINRT	channel 4
3443		00	ABINRT	channel 5
3444		00	ABINRT	channel 6
3445		00	ABINRT	channel 7
3481	SYST.IOM	OCT	17	channel mask

A.4 SCICRECI (STD)

COSY NO.	1	10	20(Released Value)	40
81	ACCT	EQU	0	accounting option
82	BDP	EQU	0	BDP module
83	CLOCK	EQU	0	clock interrupt processor
84	MAXCORE	EQU	77737_8	machine core size
85	NCHANS	EQU	6	highest channel available
86	SETCLK	EQU	0	former clock routine
87	USARDT	EQU	0	NON-USASI USER
92	INP	EQU	HOLL	ASCII input option
94	PUN	EQU	HOLL	ASCII punch option
95	PT	EQU	HOLL	ASCII paper tape option
2207,2223				AET
2288,2294	RHT	OCT	modifications	
2302,2305	BRHT	OCT	modifications	
2372		00	IOIP	channel 0
2373		00	IOIP	channel 1
2374		00	IOIP	channel 2
2375		00	IOIP	channel 3
2376		00	ABINRT	channel 4
2377		00	ABINRT	channel 5
2378		00	ABINRT	channel 6
2379		00	ABINRT	channel 7
2412	SYS.IOM	OCT	17	channel mask

A.5 BCICRECI (BATCH)

COSY NO.	1	10	20(Released Value)	40
81	ACCT	EQU	0	accounting option
82	BDP	EQU	0	BDP module
83	CLOCK	EQU	0	clock interrupt processor
84	MAXCORE	EQU	77737_8	machine core size
85	NCHANS	EQU	4	highest channel available
86	SETCLK	EQU	0	former clock routine
87	USARDT	EQU	0	NON-USASI USER
92	INP	EQU	HOLL	ASCII input option
94	PUN	EQU	HOLL	ASCII punch option
95	PT	EQU	HOLL	ASCII paper tape option
805	IOT	00	IOIP	channel 0
806		00	IOIP	channel 1
807		00	IOIP	channel 2
808		00	IOIP	channel 3
809		00	ABINRT	channel 4
810		00	ABINRT	channel 5
811		00	ABINRT	channel 6
812		00	ABINRT	channel 7
1499,1515				AET
1580,1586	RHT	OCT	modifications	
1625	SYST. IOM	OCT	17	channel mask

INDEX

INDEX

COMMENT SHEET

MANUAL TITLE 3100/3200/3300/3500 MSOS Installation Handbook

PUBLICATION NO. 60280700C REVISION

FROM: NAME:

BUSINESS ADDRESS:

COMMENTS:

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

CUT ALONG LINE

AA3419 REV. 11/69

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

FOLD ON DOTTED LINES AND STAPLE

STAPLE

STAPLE

FOLD

FOLD

FIRST CLASS
PERMIT NO. 8241

MINNEAPOLIS, MINN.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
Technical Publications
4201 North Lexington Ave.
St. Paul, Minnesota 55112

ARH219

CUT ALONG LINE

FOLD

FOLD

CONTROL DATA CORPORATION
Software Documentation
4201 North Lexington Avenue
St. Paul, Minnesota 55112

Pub. No. 60280700