

INTINIT
STORAGE ALLOCATION.

ADDRESS	LENGTH
0	633
633	

BINARY CONTROL CARDS.

IDENT INTINIT
END

BLOCKS	TYPE	ADDRESS	LENGTH
ABSOLUTE*	ABSOLUTE	0	167
PROGRAM*	LOCAL	0	633

ENTRY POINTS.

INTINIA - 13 INTINIB - 17 INTINIC - 0 MC.INT - 12

EXTERNAL SYMBOLS.

CALLSYS	S.TTY	DISKINT0	DISKWB0	DISKRSP	DISKADR	DSPCLKN	SPYTMFS
CAPAB	S.NUMTT	DISKR00	DISKWB1	DISKBUF	DSPPNT	DSPCLKP	
MAKEOBJ	MUXWRBT	DISKR01	DISKWB2	DISKSLT	DSPQNT	INTSCR	
S.ECSE	PROTEK	DISKR02	DISKWB3	DISKRQC	DSPKNT	SPYPPNT	
MUXQADD	SDVCPNT	DISKR03	DISKUF	DISKSF	DSPPPNP	SPYQNT	
MUXPNTS	SDVCQPT		DISKREQ	DISKMOT	DSPSCRN	SPYTMFA	


```

*      CACT  CREBLK  .. WRITE FILE  ECSACT
CACT  WRFILE  .. READ FILE  ECSACT
CACT  REFILE  .. READ FILE  ECSACT
*
*      EACT  MKEC    .. MAKE EVENT CHANNEL  ECSACT
*
*      CACT  EVENT   .. SEND AN EVENT  ECSACT
CACT  HANG    .. GET AN EVENT OR HANG  ECSACT
*
*      EACT  CRECC   .. CREATE A CLASS CODE  ECSACT
EACT  NWTMP   .. CREATE NEW TEMP PART OF CLASS CODE  ECSACT
*
*      CACT  SREG   .. SAVE REGISTERS  ECSACT
CACT  RREG   .. RESTORE REGISTERS  ECSACT
*
*      EACT  MKSUBP .. MAKE A SUBPROCESS  ECSACT
EACT  DSCAP   .. DISPLAY A CAPABILITY  ECSACT
EACT  MOVEC   .. MOVE CAP WITHIN FULL CLIST, MASK OPTION  ECSACT
*
*      EACT  CAPIN  .. MOVE A CAPABILITY INTO FULL CLIST  ECSACT
EACT  CAPOU   .. MOVE A CAPABILITY OUT OF FULL CLIST  ECSACT
*
*      CACT  CALSUB .. CALL A SUBPROCESS  ECSACT
CACT  JUMSUB .. JUMP CALL  ECSACT
CACT  RETURN  .. RETURN  ECSACT
CACT  FRETUR  .. FRETURN  ECSACT
CACT  ESMGEN  .. NEW ERROR SELECTION MASK (ANY SUBP)  ECSACT
CACT  ESMLOC  .. NEW ERROR SELECTION MASK (LOCAL SUBP)  ECSACT
*
*      EACT  MKOPER .. MAKE AN OPERATION  ECSACT
*
*      PARAMETER SPECIFICATION CHANGES TO OPERATIONS:
*
*      CACT  USRDAT .. PS TO USER DATUM  ECSACT
CACT  FIXDAT .. PS TO FIXED DATUM  ECSACT
CACT  USRCAP .. PS TO USER CAPABILITY  ECSACT
CACT  FIXCAP .. PS TO FIXED CAPABILITY  ECSACT
CACT  ANYCAP .. PS TO ANY CAPABILITY  ECSACT
CACT  BLKDAT .. PS TO BLOCK DATA  ECSACT
CACT  BLKCAP .. PS TO BLOCK CAPABILITY  ECSACT
*
*      CACT  ADDOPT .. ADD OPTION BITS  ECSACT
*
*      EACT  COPYOP .. COPY AN OPERATION  ECSACT
*
*      CACT  CHKBLK ..  ECSACT
CACT  DELBLK ..  ECSACT
*
*      EACT  DELFIL .. DELETE A FILE  ECSACT
EACT  REDSHP .. READ SHAPE NOS.  ECSACT
*
*      CACT  MAPZRO .. ZERO A MAP ENTRY  ECSACT

```

CACT	MPCHRO	.. CHANGE MAP ENTRY (R-O)	ECSACT	1
CACT	MPCHRW	.. CHANGE MAP ENTRY (R/W)	ECSACT	1
CACT	MOVBLK	.. MOVE A FILE BLOCK	ECSACT	1
*				
EACT	DISMAP	.. DISPLAY MAP	ECSACT	1
EACT	JPRET	.. JUMP RETURN	ECSACT	1
EACT	RINT		ECSACT	1
EACT	NEWUN	.. CHANGE UNIQUE NAMES	ECSACT	1
EACT	DISPST	.. DISPLAY STACK	ECSACT	1
EACT	DISSEN	.. DISPLAY STACK ENTRY	ECSACT	1
EACT	DSFMAP	.. DISPLAY FULL MAP ENTRY	ECSACT	1
EACT	DELCL	.. DELETE A CLIST	ECSACT	1
EACT	ADDORD	.. ADD AN ORDER TO AN OPERATION	ECSACT	1
EACT	CCCLOA		ECSACT	1
EACT	DONATE	.. TRANSFER BETWEEN ALLOCATION BLOCKS	ECSACT	1
EACT	CRALBK	.. CREATE ALLOCATION BLOCK	ECSACT	1
EACT	MODPC	.. MODIFY PCOUNTER OF STACK ENTRY	ECSACT	1
EACT	DLPRO	.. DESTROY A PROCESS	ECSACT	1
*				
CACT	GETEVF	.. GET AN EVENT OR F-RETURN	ECSACT	1
*				
EACT	DELAB	.. DESTROY AN ALLOCATION BLOCK	ECSACT	1
EACT	DELSUB	.. DELETE A SUBPROCESS	ECSACT	1
EACT	DPROD	.. DISPLAY A PROCESS	ECSACT	1
*				
CACT	CLRDAE	.. CLEAR DIRECT ACCESS TO ECS	ECSACT	1
CACT	SETDAE	.. SET DIRECT ACCESS TO ECS	ECSACT	1
CACT	SETIIB	.. SET INTERRUPT INHIBIT BIT	ECSACT	1
CACT	CLRIIB	.. CLEAR INTERRUPT INHIBIT BIT	ECSACT	1
CACT	MGETH	.. GET EVENT FROM MULTIPLE EVNT CHNNLS/HANECSACT	ECSACT	1
,				
CACT	MGETF	.. GET EVENT FROM MULTIPLE EVNT CHNNLS/FREECSACT	ECSACT	1
,				
*				
EACT	DELEC	.. DESTROY EVENT CHANNEL	ECSACT	1
*				
CACT	DCLOX	.. DISPLAY CLOCK TIMES IN USER CORE	ECSACT	1
*				
EACT	DSPAB	.. DISPLAY ALLOCATION BLOCK	ECSACT	1
EACT	TIMDT	.. OPERATOR TIME AND DATE	ECSACT	1
EACT	USRER	.. USER INITIATED ERROR	ECSACT	1
*				
CACT	RETPAR	.. RETURN WITH PARAMETERS	ECSACT	1
*				
EACT	DISPOP	.. DISPLAY AN OPERATION	ECSACT	1
EACT	CAGEN	.. MAKE CAP CREATING AUTHORIZATION	ECSACT	1
EACT	CGEN	.. CREATE A CAPABILITY	ECSACT	1
EACT	DSPSP	.. DISPLAY SUBPROCESS DESCRIPTOR	ECSACT	1
*				
CACT	TRDB	.. TEST A RESET DIRTY BIT	ECSACT	1
*				
EACT	INCHR	.. INCREMENT CHARGE RATE	ECSACT	1
EACT	DSPOB	.. DISPLAY OBJECT	ECSACT	1

EACT	DSPALC	.. DISPLAY ALLOCATOR CONSTANTS	ECSACT	1
CACT	CHMPRW	.. CHANGE EXISTING MAP ENTRY (R/W)	ECSACT	1
CACT	CHMPRO	.. CHANGE EXISTING MAP ENTRY (R-0)	ECSACT	1
CACT	DSCLX	.. DISPLAY SYSTEM CLOCKS IN USER CORE	ECSACT	1
EACT	FSON	.. FIND NTH SON OF SUBPROCESS	ECSACT	1
EACT	SPRET	.. SPECIAL RETURN: DECREMENT PCOUNTER	ECSACT	1
EACT	CPZRO	.. ZERO A CAPABILITY	ECSACT	1
EACT	INMTR	.. INCREMENT CHARGE METER	ECSACT	1
EACT	MOVCP	.. MOVE CP TIME	ECSACT	1
EACT	MOVMT	.. MOVE MOT SLOTS	ECSACT	1
EACT	DLOPR	.. DESTROY AN OPERATION	ECSACT	1
EACT	GRAB	.. STEAL ECS SPACE	ECSACT	1

BIGIF	IFGT	TABLE.377B	ECSACT	1
BIGIF	ENDIF		ECSACT	1
PROCSYM	XTEXT		ECSACT	1

0	0	NEWSYS	EQU	0	. 0 FOR REGULAR SYSTEM, 1 FOR VV SYSTEM	PROCSYM	1
---	---	--------	-----	---	---	---------	---

PROCESS SYMBOLS

40	P.PARAML	EQU	408			PROCSYM	1
35	P.APLL	EQU	P.PARAML-3		ACTUAL SPACE FOR PARAMETERS	PROCSYM	1
		IFEQ	NEWSYS,1,1			PROCSYM	1
		IFEQ	NEWSYS,0,1			PROCSYM	1
5	P.SCHEDL	EQU	5		. LENGTH OF SCHEDULER AREA	PROCSYM	1
50	P.SCR1	EQU	508		. LENGTH OF PRINCIPAL SCRATCH AREA	PROCSYM	1
6	P.SCR2L	EQU	6			PROCSYM	1
372	P.PBUFL	EQU	250		BLOCK PARAMETER BUFFER LENGHT	PROCSYM	1

0		ORG	0			PROCSYM	1
0	P.SCR	BSS	P.SCR1		. PRIMARY SCRATCH AREA	PROCSYM	1
50	P.SCR2	BSS	P.SCR2L		. ANOTHER SCRATCH AREA	PROCSYM	1
56	P.TEMP1	BSS			. MORE SCRATCH STORAGE	PROCSYM	1
57	P.TEMP2	BSS				PROCSYM	1
60	P.TEMP3	BSS				PROCSYM	1
61	P.TEMP4	BSS				PROCSYM	1
62	P.TEMP5	BSS				PROCSYM	1
63	P.TEMP6	BSS				PROCSYM	1
64	P.TEMP7	BSS				PROCSYM	1

65	1	BSS	1		DEAD CELL BEFORE ACRUAL PARAM AREA	PROCSYM	1
66	P.PARAM	BSS	P.PARAML		. ACTUAL PARAMETER AREA	PROCSYM	1
125	P.PARAMC	EQU	*-1		. END OF ACTUAL PARM AREA (CLASS CODE GOES	PROCSYM	1

. HERE ON SUBP CALL

PROCESS READ ONLY DISCRIPTOR

3	MP.SIZE	EQU	3	• SIZE OF ONE LOGICAL MAP ENTRY	PROCSYM	1
0	MP.FILE	EQU	0	• FILE ID, REL THIS MAP ENTRY	PROCSYM	1
1	MP.FADR	EQU	1	• FILE ADDR, REL THIS MAP ENRY	PROCSYM	1
2	MP.CADR	EQU	2	• CM ADDR, REL THIS MAP ENTRY	PROCSYM	1

* THIS IS WHERE THE CLOCK FLAG IS SET

2	S.FNFLG	EQU	2	• THATS RIGHT, CELL 2 ABSOLUTE	PROCSYM	1
---	---------	-----	---	--------------------------------	---------	---

* STACK FLAG NAMES.

* SF.PCQ1 AND SF.PCQ2 ARE FOR CROSS-REFERNCE ONLY

0	SF.II	EQU	0	• INTERRUPT INHIBITED	PROCSYM	1
4	SF.PCQ1	EQU	4	• ABOUT TO EXECUTE BIT	PROCSYM	1
5	SF.PCQ2	EQU	5	• ALMOST FINISHED BIT	PROCSYM	1

TYPES XTEXT

OBJECT TYPE ASSIGNMENTS

CAPABILITY TYPES

777	T.PROC	EQU	511		TYPES	1
777	T.PSEUDO	EQU	T.PROC		TYPES	1
1377	T.CLIST	EQU	767		TYPES	1
1577	T.FILE	EQU	895		TYPES	1
1677	T.OPER	EQU	959		TYPES	1
1737	T.CLSCD	SET	991		TYPES	1
1757	T.EVCH	EQU	1007		TYPES	1
1767	T.ALLOC	EQU	1015		TYPES	1
1773	T.CAUCH	EQU	1019		TYPES	1

CAPABILITY CREATING AUTHORIZATION

ALLOCATION TYPES (USED IN BLOCK HEADER WORDS)

0	AT.DATA	EQU	0	DATA BLOCK TYPE, MUST BE ZERO ***	TYPES	1
1	AT.PROC	EQU	1		TYPES	1
2	AT.CLIST	EQU	2		TYPES	1
4	AT.OPER	EQU	4		TYPES	1
5	AT.EVCH	EQU	5		TYPES	1
6	AT.ALBK	EQU	6		TYPES	1
40	AT.FILE	EQU	40B	• MUST BE .GE. 40B	TYPES	1
41	NUMTYPES	EQU	41B		TYPES	1

COMPACTION TYPES (USED IN ALLOCATOR WORD OF BLOCK)

1	GT.PTRB	EQU	1	• FILE POINTER BLOCK	TYPES	1
2	GT.DAE	EQU	2	• DIRECT ACCESS BLOCK	TYPES	1
0	GT.NRMB	EQU	0	• NORMAL BLOCKS	TYPES	1

* DIRECTORIES				OBBITS	1
	OPTORG	2	(SINCE OB.CHNAM DOES NOT APPLY)	OBBITS	1
ACC	OPT		USER DIRECTORY ACCESS	OBBITS	1
IMPL	OPT		IMPLICIT ACCESS	OBBITS	1
*CRFIL	OPT		CREATE (DISK) FILE OWNERSHIP ENTRY	OBBITS	1
CRLE	OPT		CREATE (HARD- OR SOFT-) LINK ENTRY	OBBITS	1
DLE	OPT		DELETE LINK ENTRY	OBBITS	1
INSL	OPT	***** PUT	(DISK OBJECT) IN SOFT-LINK	OBBITS	1
AAP	OPT		ADD ACCESS-PAIR	OBBITS	1
DAP	OPT		DELETE ACCESS-PAIR	OBBITS	1
SSUL	OPT		EQU SUCCESSOR LINK	OBBITS	1
CSUL	OPT		CLEAR SUCCESSOR LINK	OBBITS	1
CRDR	OPT		CREATE DIRECTORY OWNERSHIP ENTRY	OBBITS	1
CHNM	OPT		CHANGE NAME OF EXISTING DIRECTORY ENTRY	OBBITS	1
SETAG	OPT		SET ACCOUNTING TAG OF DAR IN FUNDING DIRECTION	OBBITS	1
NPSW	OPT		CHANGE THE PASSWORD IN THE PROFILE	OBBITS	1
RESO	OPT		MOVE A RESOURCE OUT OF A DIRECTORY	OBBITS	1
RESI	OPT		MOVE A RESOURCE INTO A DIRECTORY	OBBITS	1
DSPAC	OPT		DISK SPACE	OBBITS	1
SBUSR	OPT		SUB-USERS	OBBITS	1
MONEY	OPT		MONEY (\$)	OBBITS	1

*
* MACROS FOR READ AND WRITE ECS
*

RECS MACRO CNT
 RE CNT
 RJ S.ECSER
 ENDM

*
WECS MACRO CNT
 WE CNT
 RJ S.ECSER
 ENDM

* MACRO TO PUT THE ADDRESS OF THE APPROPRIATE OPTION BIT MASK IN M.ADDR
*

OPTPTR MACRO ADDRESS
 SX7 ADDRESS
 SA7 M.ADDR
 ENDM

*
*
*
*
*
*
*

COMPUTE MASTER CLIST NEEDS

THIS IS THE FIRST CALL ON INTINIT DURING INITIALIZATION.
M.ADDR IS INITIALIZED TO POINT TO A MASK OF ALL ONES

```

0
1 7170000520 +
2 7110000001 X
   10211
   20201
3 36612
   5110000164 +
   10211
4 20201
   36112
   36616
5 7110000014
   36616
6 7110000000 X
   36661
   20101
7 36661
   7110000000 X
   36661
10 7110000002
   36616
11 5160000012 +
   0200000000 +
    
```

INTINIC

```

BSSZ 1
OPTPTR M.ALL .. INITIALIZE M.ADDR
SX1 S.NUMTT+1
BX2 X1
LX2 1
IX6 X1+X2 TTY NEEDS
SA1 SDVCCNT
BX2 X1
LX2 1
IX1 X1+X2
IX6 X1+X6 SDVC NEEDS
SX1 I+11
IX6 X1+X6 DISK NEEDS
SX1 DSPSCRN
IX6 X6+X1
LX1 1
IX6 X6+X1 . DISPLAY DRIVER STUFF
SX1 DSPCLKN
IX6 X6+X1 . CLOCK EVENT CHANNELS
SX1 SPYOBJS SPY NEEDS IN MASTER CLIST
IX6 X1+X6
SA6 MC.INT
JP INTINIC
    
```

MC.INT

BSSZ 1

INITIALIZE INTERRUPT OBJECTS

```

13
14 7160000012
   0100000416 +
15 10600
   5160000032 +
16 0200000013 +
    
```

INTINIA

```

BSSZ 1
SX6 NITYPES
RJ MFILE CREATE FILE FOR C LIST INDEXES
BX6 X0
SA6 DVCFILE
JP INTINIA
    
```

```

17
20 7160000024
   0100000416 +
21 10600
   5160000033 +
22 5111000070
   7261777776
23 5161000070
    
```

INTINI@

```

BSSZ 1
SX6 NITYPES*2
RJ MFILE CREATE FILE FOR INTERRUPT QUEUES
BX6 X0
SA6 INTQS
SA1 B1+P.PARAM+2
SX6 X1-1
SA6 B1+P.PARAM+2 DESTROY C LIST ENTRY FOR QUEUES
    
```


35 000000000
36 020000035 +

*
*
*

ERROR

PS

JP

ERROR

CREATE MUX DEVICES

```

*
*
*
*
EXT      MUXQADD
EXT      MUXPNTS
EXT      S.TTY
EXT      S.NUMTT
EXT      MUXWRBIT
EXT      PROTEK
*
0 X MUXWRBIT EQU MUXWRBIT
*
1 MUXINTX EQU 1
*
*
37      MMUX      BSSZ      1
40 6120000527 +
      6130000000 X      SB2      P.QUEUE
41 6140000000 X      SB3      MUXPNTS
      6150000001      SB4      MUXQADD
42 6160000001 X      SB5      MUXINTX
      0100000463 +      SB6      MUXWRBIT*1
*                               RJ      NEWCLASS
43 5111000070      SA1      BI+P.PARAM+2      PUTCLIST INDEX OF
      7261000001      SX6      X1+1      SPECIAL TTY IN IPROC
44 5160000000 X      SA6      S.TTY
*
      7160000000 X      SX6      MUXWRBIT
45 5160000542 +      SA6      INADDR
      0100000054 +      RJ      MMUXDVC      MAKE SPECIAL TTY
*
46 76600      SX6      B0
      5160000542 +      SA6      INADDR
*
47 0100000054 +      MMUX1      RJ      MMUXDVC
50 5110000542 +      SA1      INADDR
      7261000001      SX6      X1+1
51 5160000542 +      SA6      INADDR
      7120000000 X      SX2      S.NUMTT
52 37262      IX2      X6=X2
      0332000047 +      NG      X2*MMUX1
*
53 0200000037 +      JP      MMUX

```

```

*
*
* CODE TO CREATE ECS OBJECTS FOR MUX DEVICE WHOSE
* INPUT PHYSICAL ADDRESS IS AT INADDR
*
54 MMUXDVC BSSZ 1
55 7160000021 0100000416 + SX6 TTYFSIZE
56 10100 20122 5120000532 + RJ MFILE CONSTRUCT THE FILE
57 43346 20322 11623 15113 SA2 LX1 18 FILEADDR PLACE FILE ADDR IN P PROC
MX3 60-22
LX3 18
BX6 X2*X3
BX1 -X3*X1
BX6 X1+X6
SA6 FILEADDR
SA0 PFILE
WECS TTYDTA WRITE FILE DATA BLOCK
*
63 7170000522 + OPTPTR M.SENDE ** ADDRESS OF OPTION BIT MASK FOR PROTECTING
EVENT CHANNELS (OB.SENDEV+OB.CHNAM)
*
*
64 7160000012 SX6 ECOUTCNT MAX EVENTS IN OUTPUT E C
65 5170000540 + RJ MEC CONSTRUCT THE OUTPUT EVENT CHANNEL
SA7 OUTCHN
*
67 0100000373 + 7160000003 SX6 ECINCNT MAX EVENTS IN INPUT E C
70 5170000541 + RJ MEC CONSTRUCT INPUT EVENT CHANNEL
SA7 INCHN
*
71 0100000411 + 6120000021 SB2 PPROC SZ SIZE OF PSEUDO PROCESS
72 10644 5160000543 + RJ Mps
BX6 X4
SA6 PMOT SAVE P PROC UNIQUE NAME AND MOT INDEX
BX0 X5
SA0 PPROC
WECS PPROC SZ WRITE P PROC DATA TO P PROC
SX0 X6
SA0 TEMP
RECS 1 GET POINTER TO THIS P PROC
SA1 MUXPNTS
SA2 INADDR
IX0 X1+X2
SA0 TEMP
WECS 1 WRITE POINTER OF THIS DEVICE TO ECS
*
100 36012 5100000455 + JP MMUXDVC
101 0120000001
102 0200000054 +

```


133 36012
 5100000455 +
 134 5110000455 +
 43647
 15616
 135 5160000455 +
 136 0120000001
 137 7262000001
 5160000572 +
 *
 140 5110000164 +
 7261777776
 141 5160000164 +
 0306000103 +
 *
 142 5110000147 +
 63210
 143 5112000003
 7261777776
 144 54610
 0316000110 +
 *
 145 7162000004
 5160000147 +
 146 0200000110 +

IX0 X1+X2
 SA0 TEMP
 SA1 TEMP
 MX6 60-21
 BX6 X6*X1
 SA6 TEMP
 WECS 1
 SX6 X2+1
 SA6 SDVCCNT
 SA1 SDVCCNT
 SX6 X1-1
 SA6 SDVCCNT
 ZR X6,MSDVC
 SA1 SDVCCPNT
 SB2 X1
 SA1 B2+3
 SX6 X1-1
 SA6 A1
 NZ X6,MSDVCLP
 SX6 B2+4
 SA6 SDVCCPNT
 JP MSDVCLP

REMOVE UNIQUE NAME
 WRITE DEVICE POINTER TO ECS
 STEP S DEVICE NUMBER

ALL S DEVICES COMPLETED

GO TO NEXT TYPE OF S DEVICE

S DEVICE DESCRIPTIONS

IN FORM FILE SIZE
 REQUEST EC SIZE
 RESPONSE EC SIZE
 NUMBER OF THIS TYPE

SDVCCNT MUST AGREE WITH TOTAL NUMBER OF S DEVICES

147
 147 000000000000000000150 +
 150 0000000000000000004012
 154 0000000000000000001000
 160 0000000000000000001000
 *
 164 0000000000000000000003

SDVCCPNT BSS 0
 VFD 60/#+1
 DATA 4012B,6,6,1
 DATA 1000B,4,4,1
 DATA 1000B,4,4,1
 SDVCCNT DATA 3

TAPE DRIVE
 PRINTER
 CARD READER

CREATE OBJECTS FOR DISK

```

*
*
*
EXT      DSKPNT,DSKINTQ
*
3      DSKINTX EQU      3
*
*
165      MDISK      BSSZ      1
166      6120000605 +      SB2      DPQUEUE
           6130000000 x      SB3      DSKPNT
167      6140000000 x      SB4      DSKINTQ
           6150000003      SB5      DSKINTX
170      6160000011      SB6      9
           0100000463 +      RJ      NEWCLASS MAKE A NEW CLASS WITH 9 PSEUDO PROCESSES
*                                     THE PPROC POINTER FILE WILL NOT BE USED
*
*
MAKE DISK PSEUDO PROCESSES
*
171      7110777776      SX1      1
           7120000010      SX2      2
172      0100000252 +      RJ      DSKPP
173      7110000003      SX1      3
           20122          LX1      18
174      7130000001      SX3      1
           12113          BX1      X1+X3
175      7222777776      MDISK1  SX2      X2-1
           7232777774      SX3      X2-3
176      0303000201 +      ZR      X3,MDISK2
           0100000252 +      RJ      DSKPP
177      5140000205 +      SA4      MDISKX
           37114          IX1      X1-X4      STEP SZX FIELD
200      0200000175 +      JP      MDISK1
201      7110000003      MDISK2  SX1      3
           20122          LX1      18
202      0332000206 +      MDISK3  NG      X2,MDISK4
           0100000252 +      RJ      DSKPP
203      5140000205 +      SA4      MDISKX
           37114          IX1      X1-X4      STEP SZX FIELD
204      7222777776      SX2      X2-1
           0200000202 +      JP      MDISK3
*
*
205      00000000000001000000 MDISKX DATA 1000000B INCREMENT FOR SZX FIELD
*
*
MAKE USER FILE
*
EXT      DISKR00,DISKR01,DISKR02,DISKR03
EXT      DISKW00,DISKW01,DISKW02,DISKW03
EXT      DISKUF
*
*
206      7160000000 x      MDISK4  SX6      DISKR00 COMPUTE SIZE NEEDED

```



```

207 20606      7266000000 X
           7110000000 X
210 7211000000 X
           36676
           20107
211 36616      7110000000 X
           7211000000 X
           36676
           20110
212 36616      7110000000 X
           7211000000 X
           36676
           20111
213 36616      7110000000 X
           7211000000 X
           36676
           20111
214 36616      7110000401
           36616
215 0100000416 +
216 10600
217 5160000000 X

```

```

SX6      X6+DISKWB0
LX6      6
SX1      DISKRB1
SX1      X1+DISKWB1
IX6      X1+X6
LX1      7
IX6      X1+X6
SX1      DISKRB2
SX1      X1+DISKWB2
IX6      X1+X6
LX1      8
IX6      X1+X6
SX1      DISKRB3
SX1      X1+DISKWB3
IX6      X1+X6
LX1      9
IX6      X1+X6
SX1      256+1
IX6      X1+X6
RJ       MFILE
BX6      X0
SA6      DISKUF

```

SPACE FOR BUFFER POINTERS AND USER THIS

MAKE UP 11 EVENT CHANNELS

```

*
*
*
*
DODBUF  MACRO  COUNT*X
          SX6  COUNT+1
          RJ   MEC
          SA7  DISKBUF*X
          ENDM

```

```

EXT     DISKREQ*DISKRSP*DISKBUF*DISKSLT
EXT     DISKRQC

```

```

220 7160000001 X
           0100000373 +
221 5170000000 X
           7160000001 X
222 0100000373 +
223 5170000000 X

```

```

SX6      DISKRQC*1
RJ       MEC
SA7      DISKREQ
SX6      DISKRQC*1
RJ       MEC
SA7      DISKRSP

```

```

*
*
*
*
*
*
*
*
*
*
*
*
DODBUF  DISKRQ0*0
DODBUF  DISKRQ1*1
DODBUF  DISKRQ2*2
DODBUF  DISKRQ3*3
DODBUF  DISKWB0*4
DODBUF  DISKWB1*5
DODBUF  DISKWB2*6
DODBUF  DISKWB3*7

```

244 0100000373 + 7160000001 x
 245 5170000000 x

SX6 DISKRQC+1
 RJ MEC
 SA7 DISKSLT

*
*
*
*

MAKE UP PRIVATE FILE

246 0100000416 + 7160001500
 247 10600
 250 5111000070 5160000000 x
 251 54610 7261777776

EXT DISKSF
 SX6 256+64+512
 RJ MFILE
 BX6 X0
 SA6 DISKSF
 SA1 B1+P.PARAM+2
 SX6 X1-1
 SA6 A1

DESTROY C [IST] ENTRY FOR THIS FILE

0200000165 +

JP MDISK

*
*
*
*
*
*
*
*
*

MAKE UP A DISK PSEUDO PROCESS

X1 DATA WD
X2 INDEX OF PPROC

THESE WILL BE RETURNED

		EXT	DISKMOT?DISKADR
252		DSKPP	BSSZ
253	10611		1
	5160000611 +	BX6	X1
	10622	SA6	DDATA
254	5160000265 +	BX6	X2
	6120000010	SA6	DSKPPSV
255	0100000411 +	SB2	DPPROCSZ
256	10644	RJ	MPS
	5160000610 +	BX6	X4
257	5110000611 +	SA6	DMOT
	5120000265 +	SA1	DDATA
260	5262000000 X	SA2	DSKPPSV
	10655	SA6	DISKMOT*X2
261	5262000000 X	BX6	X5
	10066	SA6	DISKADR*X2
262	5100000602 +	BX0	X6
263	0120000010	SA0	DPPROC
264	0200000252 +	WECS	DPPROCSZ
		JP	DSKPP
265		DSKPPSV	BSSZ
			1

311 0100000373 + 7160000014
 312 5110000000 X 5120000331 +
 313 5170000455 + 54070 36012
 314 0120000001

SX6
 RJ
 SA1
 SA2
 SA7
 SA0
 IX0
 WECS

I2
 MEC
 DSPKPNT
 MDSPT
 TEMP
 A7
 X1+X2
 I

• KEYBOARD EVCH SIZE
 • ECS ADDRESS OF POINTER FILE
 • INDEX
 •
 • WRITE POINTER TO POINTER FILE

315 6120000012 0100000411 +
 316 10644 5160000621 +
 317 5110000000 X 5120000331 +
 320 10722 36012 5170000623 +
 321 5160000455 + 5100000455 +
 322 0120000001

SB2
 RJ
 BX6
 SA6
 SA1
 SA2
 BX7
 IX0
 SA7
 SA6
 SA0
 WECS

DSPPROCL
 MPS
 X4
 DSPMOT
 DSPPPNP
 MDSPT
 X2
 X1+X2
 DSPSCRN
 TEMP
 TEMP
 I

• P-PROC LEN
 • MAKE P-PROC
 • PUT MOT IN P-PROC
 • SCREEN INDEX IN P-PROC
 • SAVE MOT IN DSPPPNP FILE

323 10055 10600 5160000622 +
 324 5100000612 +
 325 0120000012

BX0
 BX6
 SA6
 SA0
 WECS

X5
 X0
 DSPPROCA
 DSPPROC
 DSPPROCL

• ECS ADDRESS OF P-PROC
 • WRITE P-PROC TO ECS

326 7277000001 7160000000 X
 327 54720 37776

SX7
 SX6
 SA7
 IX7

X7+1
 DSPSCRN
 A2
 X7-X6

• BUMP COUNTER
 • CHECK FOR END

330 0400000266 + 0337000302 +

NG
 EQ

X7, MDSPLP
 MDSPI

• LOOP IF MORE
 • ELSE RETURN

331 MDSPT

BSSZ

I

INTINIT

COMPASS - VER 2. 10/20/71 11.15.08.

PAGE 28

372 020000350 *

JP

MSPY

ALL DONE WITH SPY STUFF

```

*
* MAKE AN EVENT CHANNEL TO HOLD NUMB OF EVENTS IN X6
* RETURN WITH ABS ECS ADDRESS IN X7
* LOOKS AT M.ADDR TO FIND ADDRESS OF OPTION BIT MASK TO BE USED BY
* PROTEK. PROTEK IS A ROUTINE IN INITL WHICH PRESERVES
* SELECTED OPTION BITS ON A CAPABILITY.
* OBJECT SINCE IT USES THE ACTUAL PARAMETER LIST TO FIND THE INDEX
* PROTEK SHOULD BE CALLED IMMEDIATELY AFTER CREATING AN
* OF THE OBJECT IN THE MASTER CLIST. M.ADDR IS RESET TO POINT
* TO AN OPTION BIT MASK WITH ALL BITS ON.
* (AFTER THE EVCH IS CREATED AND ITS SPECIFIED OPTION
* BITS ARE TURNED OFF)
*

```

```

373
374 5161000071
      5111000070
375 7261000001
      54610
376 6150000011
      6160000400 +
377 0200000000 X
      *
400 5110000517 +
      63610
401 0100000000 X
402 7170000520 +
      *
403 5110000001 X
      73010
404 5100000410 +
405 0110000001
406 5110000410 +
      43047
      15710
407 0200000373 +
      *
410
      MEC2

```

```

BSSZ 1
SA6 B1+P.PARAM+3
SA1 B1+P.PARAM+2
SX6 X1+1 STEP C LIST INDEX
SA6 A1
SBS J.MKEC
SB6 MEC1
JP CALLSYS
*
MEC1 SA1 M.ADDR
      SB6 X1
      RJ PROTEK
OPTPTR M.ALL .. RESET M.ADDR (NOT EVERYONE WANTS
      TO TURN OFF OPTION BITS)
*
SA1 CAPAB+1
SX0 X1
SA0 MEC2
RECS 1 GET ABS ECS ADDRESS
SA1 MEC2
MX0 60-21
BX7 -X0#X1
JP MEC
*
MEC2 BSSZ 1

```

*
*
*
*
*
*

CREATE PSEUDO PROCESS OF SIZE GIVEN IN B2
MAKE NO C LIST ENTRY
RETURN WITH ABS ADDRESS OF TH P PROC IN X5
MOT INDEX IN X4

411
412 6140000414 +
5151000067
413 7170000001
0200000000 x

414 5140000001 x
73440
415 0200000411 +

MPS

MPS1

MPS1

BSSZ
SB4
SA5
SX7
JP

SA4
SX4
JP

1
MPS1
B1+P.PARAM+1 GET U NAME AND MOT OF ALLOC BLK
AT.PROC
MAKEOBJ

CAPAB+1
Y4
MPS
STRIP OFF UNIQUE NAME

```

*
* ROUTINE TO CREATE A 1 LEVEL FILE WITH DATA BLOCK OF
*   SIZE GIVEN IN X6
* RETURNS WITH ECS ADDRS OF THE DATA BLK IN X0
*

```

```

* OPTION BITS FOR THE FILE ARE MASKED ACCORDING TO THE PTR
* STORED IN M.ADDR. (A CALL IS MADE TO ↑PROTEK↑)
* M.ADDR IS RESET TO POINT TO A MASK WITH ALL BITS ON.
*

```

```

416 MFILB BSSZ 1
417 5160000462 + SA6 BSIZE
      7160000001 SX6 1
420 5161000071 SA6 B1+P.PARAM+3 SET NUMBER OF LEVELS = 1
      7160000462 + SX6 BSIZE
421 5161000072 SA6 B1+P.PARAM+4 POINT TO BLOCK SIZE
      5111000070 SA1 B1+P.PARAM+2
422 7261000001 SX6 X1+1 STEP CLIST INDEX
      54610 SA6 A1
423 6150000004 SB5 J.CREFIL
      6160000425 + SB6 MFILE1
424 0200000000 X JP CALLSYS
*
425 6121000066 MFILB1 SB2 B1+P.PARAM
      6130000003 SB3 3 FIRST 3 ACTUAL PARAMETERS
426 56120 MFILB2 SA1 B2
      10611 BX6 X1
      5163000455 + SA6 B3+TEMP
427 6122000001 SB2 B2+1
      6133777776 SB3 B3-1
430 0530000426 + NE B3+B0,MFILE2
*
431 10611 5110000000 X SA1 CAPAB SET UP FILE CAPABILITY
      5161000066 BX6 X1 IN ACTUAL PARAM AREA
432 5110000001 X SA6 B1+P.PARAM
      10611 SA1 CAPAB+1
      5161000067 BX6 X1
433 5161000067 SA6 B1+P.PARAM+1
      5160000461 + SA6 FPOINT SAVE MOT INDES OF FILE
434 7160000000 SX6 0
      5161000070 SA6 B1+P.PARAM+2 PLACE AN ADDRS OF A WORD IN BLOCK
435 6150000006 SB5 J.CREBLK IN ACTUAL PARAM AREA
      6160000437 + SB6 MFILE3 AND CALL FOR BLOCK CREATION
436 0200000000 X JP CALLSYS
*
437 6121000066 MFILB3 SB2 B1+P.PARAM SET UP TO RESTORE FIRST
      6130000003 SB3 3 3 ACTUAL PARAMS
440 5113000455 + MFILB4 SA1 B3+TEMP
      10611 BX6 X1
      56620 SA6 B2
441 6122000001 SB2 B2+1
      6133777776 SB3 B3-1
442 0530000440 + NE B3+B0,MFILE4
*

```

```

5110000517 +
443 63610 0100000000 X
444 7170000520 +
445 5110000461 +
      73010
446 5100000461 +
447 0110000001
450 5110000461 +
      43247
      15012
451 5100000461 +
452 0110000001
453 5110000461 +
      43247
      15012
454 0200000416 +
*
*
455 TEMP
461 FPOINT
462 BSIZE

```

```

SA1 M.ADDR
SB6 Y1
RJ PROTEK
OPTPTR M.ALL

SA1 FPOINT
SX0 X1
SA0 FPOINT
RECS 1
SA1 FPOINT
MX2 60-21
BX0 -X2*X1
SA0 FPOINT
RECS 1
SA1 FPOINT
MX2 60-21
BX0 -X2*X1
JP MFILE

BSSZ 4
BSSZ 1
BSSZ 1

```

.. NOW THAT WE HAVE CREATED THE FILE BLOCKS
TURN OFF THE OPTION BITS.

NOW COMPUTE BLOCK ADDRESS
COMPUTE MOT INDEX OF THE FILE

READ IN MOT ENTRY

COMPUTE ADDRS OF FILE DESC BLOCK

COMPUTE ADDRESS OF DATA BLOCK
EXIT

```

*
*
*      INITIALIZE      A NEW CLASS OF OBJECTS
*
*      B2      LOCATION OF PROTOTYPE P PROC INT QUEUE POINTER
*      B3      LOCATION OF POINTER TO P PROC POINTERS
*      B4      LOCATION OF POINTER TO INTERRUPT QUEUE
*      B5      INTERRUPT INDEX
*      B6      NUMBER OF DEVICES IN CLASS
*
*
463      NEWCLASS BSSZ      1
464      76620      SX6      B2
           5160000512 +    SA6      NEWC2
           76630      SX6      B3
465      5160000513 +    SA6      NEWC3
           76640      SX6      B4
466      5160000514 +    SA6      NEWC4
           76650      SX6      B5
467      5160000515 +    SA6      NEWC5
           76660      SX6      B6
470      5160000516 +    SA6      NEWC6
*
471      0725000035 +    SB2      NITYPES-1
           6120000011      LT      B2,B5,ERROR  CHECK SIZE OF INTERRUPT INDEX
           76660      SX6      B6
472      0100000416 +    RJ      MFILE      NUMBER OF DEVICES IN CLASS
473      10600      BX6      X0      FILE TO CONTAIN POINTERS TO P PROCS
           5110000513 +    SA1      NEWC3
           53610      SA6      X1
474      5111000070      SA1      B1+P.PARAM+2  SAVE POINTER TO P PROC POINTERS
           7261777776      SX6      X1-1
475      54610      SA6      A1      DESTROY C LIST ENTRY FOR THIS FILE
*
476      5120000515 +    SA1      INTQS      COMPUTE ADDRESS OF INTERRUPT QUEUE
           5110000033 +    SA2      NEWC5
           20201      LX2      1
           36612      IX6      X1+X2
477      5110000514 +    SA1      NEWC4
           53610      SA6      X1
           20622      LX6      18
500      5130000512 +    SA3      NEWC2
           53130      SA1      X3
           43246      MX2      60-22
501      20222      LX2      18
           11121      BX1      X2*X1
           15662      BX6      -X2*X6
           12616      BX6      X1+X6
502      53630      SA6      X3      PLACE QUEUE POINTER IN P PROC DATA
*
503      7261000001      SA1      DVCTYPE
           5160000034 +    SX6      X1+1
           SA6      DVCTYPE

```

504	722177765		SX2	X1=NITYPES	
		0322000035 +	PL	X2=ERROR	CHECK DEVICE TYPE
505	5121000070		SA2	B1+P.PARAM+2	
		7262000001	SX6	X2+1	FIRST C LIST INDEX FOR THESE OBJECTS
506	5160000455 +		SA6	TEMP	
		5120000032 +	SA2	DVCFIL	
507	36012		IX0	X1*X2	
		5100000455 +	SA0	TEMP	
510	0120000001		WECS	1	SAVE FIRST C LIST INDEX
		*			
511	0200000463 +		JP	NEWCLASS	
		*			
		*			
512			NEWC2	BSSZ	1
513			NEWC3	BSSZ	1
514			NEWC4	BSSZ	1
515			NEWC5	BSSZ	1
516			NEWC6	BSSZ	1
517			M.ADDR	BSS	1
		*			HOLDS ADDRESS OF CURRENT OPTION BIT MASK (FOR USE WHILE CREATING OBJECTS)
520	77777777777777777777		M.ALL	DATA	0
521	0000000000000000000032		M.GETE	VFD	60/OB.GETE+OB.CHNAM+OB.GTEVF
522	0000000000000000000006		M.SENDE	VFD	60/OB.SNDEV+OB.CHNAM
523	0000000000000000000016		M.FILE	VFD	60/OB.PLMAP+OB.CHNAM+OB.RDFIL+OB.WFILE
			M.RDFIL	VFD	60/OB.PLMAP+OB.RDFIL

```

*
*
*
524 11000001000000000000 PPROC VFD 3/1,3/1,6/0,12/1,18/FIELD1,18/0
525 BSSZ 1
526 00000000000000000000 DATA 0
527 000000000000000000001 PQUEUE VFD 20/0,22/FIELD2,18/MUXINTX
530 BSSZ 2
*
532 000000000000000000021 FILEADDR VFD 22/0,20/FIELD3,18/TTYFSIZE
533 000000000000000000007 VFD 60/7
534 000000000000000000007 VFD 60/7
535 000000000000000000014 VFD 60/7+TTYOSIZE
536 000000000000000000014 VFD 60/7+TTYOSIZE
537 000000000000000000021 VFD 60/TTYFSIZE
540 000000000000000000000 OUTCHN VFD 60/FIELD4
541 000000000000000000000 INCHN VFD 60/FIELD5
542 000000000000000000000 INADDR VFD 60/FIELD6
543 000000000000000000000 PMOT VFD 60/FIELD7
544 000000000000000000000 VFD 60/FIELD8
*
21 PPROCSZ EQU *_PPROC
*
*

```

DESCRIPTION OF FIELDS WHICH RECIEVE VALUES DYNAMICALLY

```

0 FIELD1 EQU 0 MOT INDEX
0 FIELD2 EQU 0 QUEUE ADDRESS
0 FIELD3 EQU 0 ECS ADDR FILE DATA BLOCK
0 FIELD4 EQU 0 MOT, UNIQUE NAME OUT EVENT CHANNEL
0 FIELD5 EQU 0 MOT, UNIQUE NAME OF IN EVENT CHANNEL
0 FIELD6 EQU 0 INPUT PHYSICAL ADDRESS
0 FIELD7 EQU 0 MOT, UNIQUE NAME OF THIS P PROC
0 FIELD8 EQU 0 HOLDS BROADCAST IN PROGRESS FLAG, AND COUNT ETC

```


PROTOTYPE FILE DATA FOR MUX DEVOCE

545 000000000000000000000007
 546 000000000000000000000007
 547 000000000000000000000014
 550 000000000000000000000014
 551 000000000000000000000021
 552 000000000000000000000012
 553 000000000000000000000014

*
 *
 *
 PFILE VFD 60/7
 VFD 60/7
 VFD 60/7+TTYOSIZE
 VFD 60/7+TTYOSIZE
 VFD 60/TTYFSIZE
 VFD 60/TTYDTA
 VFD 60/7+TTYOSIZE
 *

554 00150012012401230123
 555 02400116011701270240
 556 01250120001500124001

DATA 00150012012401230123B CR LF TSS
 DATA 02400116011701270240B NOW
 DATA 01250120001500124001B UP CR LF EOM

*
 *
 *
 DATA 00150012010701170240B
 DATA 01010127010101310241B
 DATA 02410241001500124001B
 EQU *-PFILE

CR LF GO
 ALWAYS
 SS CR LF EOM

12 TTYDTA

```
*  
*  
*  
5 TTYOSIZE EQU 5  
5 TTYISIZE EQU 5  
21 TTYFSIZE EQU 7+TTYOSIZE+TTYISIZE  
12 ECOUTCNT EQU 10  
3 ECINCNT EQU 3  
*  
*  
COMPUTE ITY FILE AND EVENT CHANNEL CONSTANTS
```

```

*
*
*
557 1100000100000000000000 SPPROC VFD 3/1,3/1,6/0,12/1,36/0
560 BSSZ 1
561 0000000000000000000000 DATA 0
562 0000000000000000000002 SPQUEUE VFD 22/0,20/SFIELD1,18/SDVCINTX
563 BSSZ 2
*
565 SFILE BSSZ 1 BUFFER FILE DATA BLOCK ADDR
566 SFILESZ BSSZ 1 SIZE OF BUFFER FILE DATA BLOCK
567 SECREQ BSSZ 1 REQ EVENT CHANNEL
570 SECRSP BSSZ 1 RESPONSE EVENT CHANNEL
571 SMOT BSSZ 1 SPPROC MOT AND UNIQUE NAME
572 SDVCNUMB BSSZ 1 NUMBER OF THIS DEVICE
*
573 SEFLAG BSSZ 1
574 SREQLEFT BSSZ 1
575 SREQDONE BSSZ 1
576 SBUFPOS BSSZ 1
577 SBUFSZ BSSZ 1
600 SACTION BSSZ 1
601 SUSERREQ BSSZ 1
*
23 SPPROCSZ EQU *-SPPROC
*
*
*
0 SFIELD1 EQU 0 QUEUE ADDRESS

```

PROTOTYPE OF S DEVICE PSEUDO PROCESS

DESCRIPTIONS OF SOME FIELDS WHICH RECIEVE VALUES

```

*
*
*
602 110000010000000000000000 DPPROC VFD 3/1,3/1,6/0,12/1,36/0
603 BSSZ 1
604 000000000000000000000000 DATA 0
605 000000000000000000000003 DPQUEUE VFD 22/0,20/SFIELD1,18/DSKINTX
606 BSSZ 2
*
610 DMOT BSSZ 1 DISK PPROC MOT AND UNIQUE NAME
611 DDATA BSSZ 1 DISKDATA WORD
*
10 DPPROC$Z EQU *-DPPROC

```

*
*
*
*
*
*

PROTOTYPE FOR DISPLAY PSEUDO-PROCESSES

612	11000001000000000000	DSPPROC	VFD	3/1,3/1!6/0,12/1,36/0
613			BSSZ	1
614	00000000000000000000		DATA	0
615	000000000000000000004	DSPQUEUE	VFD	22/0,20/0,18/DSPINTX
616			BSSZ	2 . EVENT WORDS
620		DSPECREQ	BSSZ	1
621		DSPMOT	BSSZ	1
622		DSPPROCA	BSSZ	1
623		DSPSCREEN	BSSZ	1
		*		
		*		
	12	DSPPROCL	EQU	*-DSPPROC

```

*
*
*
*
*
624 11000001000000000000 SPYSP VFD 3/1,3/1,6/0,12/1,18/FIELD1,18/0
625 BSSZ 1
626 00000000000000000000 DATA 0
627 000000000000000000005 SPYQUEUE VFD 20/0,22/FIELD2,18/SPYINTX
630 BSSZ 2
*
632 SPYSPMOT BSSZ 1
*
* 7 SPYSPSZ EQU *-SPYSP
*
* (FOR FIELDS 1 AND 2 SEE PPROC )
*
633 END

36213 STORAGE USED 1801 STATEMENTS 440 SYMBOLS
6600 ASSEMBLY 9.036 SECONDS 1144 REFERENCES

```


INTINIT SYMBOLIC REFERENCE TABLE.

DVCFIL	32	PROGRAM*	13/42 S	14/12 L	34/06				
DVCTYP	34	PROGRAM*	14/14 L	33/51	33/53 S				
ECINCNT	3		17/32	37/08 D					
ECOUTCNT	12		17/24	37/07 D					
ERROR	35	PROGRAM*	15/04 L	15/05	33/24	34/02			
FIELD1	0		35/04	35/27 D	41/06				
FIELD2	0		35/07	35/28 D	41/09				
FIELD3	0		35/10	35/29 D					
FIELD4	0		35/16	35/30 D					
FIELD5	0		35/17	35/31 D					
FIELD6	0		35/18	35/32 D					
FIELD7	0		35/19	35/33 D					
FIELD8	0		35/20	35/34 D					
FILEADDR	532	PROGRAM*	17/10	17/16 S	35/10 L				
FPOINT	461	PROGRAM*	31/38 S	32/06	32/08	32/10	32/13	32/15	32/22 L
GT.DAE	2		8/52 D						
GT.NRMB	0		8/53 D						
GT.PTRB	1		8/51 D						
INADDR	542	PROGRAM*	16/31 S	16/35 S	16/38	16/40 S	17/47	35/18 L	
INCHN	541	PROGRAM*	17/34 S	35/17 L					
INTINIA	13	PROGRAM*	2/04 E	13/38 L	13/43				
INTINIB	17	PROGRAM*	2/04 E	13/46 L	14/09				
INTINIC	0	PROGRAM*	2/05 E	13/08 L	13/30				
INTQS	33	PROGRAM*	13/50 S	14/13 L	33/35				
INTSCR	0	EXTERNAL*	26/14						
I.NUNCH	5		2/16 D						
I.NUSUB	5		2/18 D						
I.PAUSE	454		2/14 D						
J.ADDOPT	54		3/43 D						
J.ADDORD	111		4/13 D						
J.ANYCAP	51		3/39 D						
J.BLKCAP	53		3/41 D						
J.BLKDAT	52		3/40 D						
J.CAGEN	156		4/46 D						
J.CALSUB	35		3/24 D						
J.CAPIN	31		3/21 D						
J.CAPOU	33		3/22 D						
J.CCCLOA	113		4/14 D						
J.CGEN	160		4/47 D						
J.CHKBLK	57		3/47 D						
J.CHMPRO	174		5/04 D						
J.CHMPRW	173		5/03 D						
J.CLRDAE	134		4/26 D						
J.CLRIB	137		4/39 D						
J.COPYOP	55		3/45 D						
J.CPROC	2		2/52 D						
J.CPZRO	202		5/09 D						
J.CRALBK	117		4/16 D						
J.CREBLK	6		3/02 D	31/41					
J.CRECC	15		3/11 D						
J.CRECL	0		2/51 D						
J.CREFIL	4		2/53 D	31/19					
J.DCLOX	144		4/37 D						

INTINIT
SYMBOLIC REFERENCE TABLE.

J.DELAB	126	4/22	D
J.DELBLK	60	3/48	D
J.DELCL	107	4/12	D
J.DELEC	142	4/35	D
J.DELFIL	61	3/50	D
J.DELSUB	130	4/33	D
J.DISMAP	71	4/35	D
J.DISPOP	154	4/45	D
J.DISPST	101	4/09	D
J.DISSEN	103	4/10	D
J.DLOPR	212	5/13	D
J.DLPRO	123	4/18	D
J.DONATE	115	4/15	D
J.DPROD	132	4/24	D
J.DSCAP	25	3/18	D
J.DSCLX	175	5/05	D
J.DSFMAP	105	4/11	D
J.DSPAB	145	4/39	D
J.DSPALC	171	5/01	D
J.DSPOB	167	4/53	D
J.DSPSP	162	4/48	D
J.ESMGEN	41	3/28	D
J.ESMLOC	42	3/29	D
J.EVENT	13	3/08	D
J.FIXCAP	50	3/28	D
J.FIXDAT	46	3/26	D
J.FRETUR	40	3/7	D
J.FSON	176	5/07	D
J.GETEVF	125	4/30	D
J.GRAB	214	5/14	D
J.HANG	14	3/09	D
J.INCHR	165	4/52	D
J.INMTR	204	5/10	D
J.JPRET	73	4/06	D
J.JUMSUB	36	3/25	D
J.MAPZRO	65	3/23	D
J.MGETF	141	4/32	D
J.MGETH	140	4/30	D
J.MKEC	11	3/06	D
J.MKOPER	43	3/31	D
J.MKSUBP	23	3/17	D
J.MODPC	121	4/17	D
J.MOVBLK	70	4/03	D
J.MOVCP	206	5/11	D
J.MOVEC	27	3/19	D
J.MOVMT	210	5/12	D
J.MPCHRO	66	4/01	D
J.MPCHRW	67	4/02	D
J.NEWUN	77	4/08	D
J.NWTMP	17	3/12	D
J.PINT	75	4/07	D
J.REDSHP	63	3/51	D
J.REFILE	10	3/04	D

INTINIT SYMBOLIC REFERENCE TABLE.

J.RETPAR	153		4/43	D						
J.RETURN	37		3/26	D						
J.RREG	22		3/15	D						
J.SETDAE	135		4/27	D						
J.SETIIB	136		4/28	D						
J.SPRET	200		5/08	D						
J.SREG	21		3/14	D						
J.TIMDT	147		4/40	D						
J.TRDB	164		4/50	D						
J.USRCAP	47		3/37	D						
J.USRDAT	45		3/35	D						
J.USRFR	151		4/41	D						
J.WRFILE	7		3/03	D						
MAKEOBJ	0	EXTERNAL*	30/11							
MCLOCKS	332	PROGRAM*	14/06		26/08	L	26/37			
MCLOCKS1	337	PROGRAM*	26/22	L	26/36					
MC.INT	12	PROGRAM*	2/06	E	13/29	S	13/32	L		
MDISK	165	PROGRAM*	14/04		20/09	L	22/17			
MDISKX	205	PROGRAM*	20/31		20/38		20/44	L		
MDISK1	175	PROGRAM*	20/27	L	20/33					
MDISK2	201	PROGRAM*	20/29		20/34	L				
MDISK3	202	PROGRAM*	20/36	L	20/41					
MDISK4	206	PROGRAM*	20/36		20/53	L				
MDSP	266	PROGRAM*	14/05		24/06	L	25/38			
MDSPLP	302	PROGRAM*	24/38	L	25/37					
MDSPT	331	PROGRAM*	24/41		25/04		25/16	25/42	L	
MEC	373	PROGRAM*	17/25		18/38		21/46	21/49	21/52	24/50
			17/33		21/39		21/47	21/50	21/53	25/02
			18/31		21/42		21/48	21/51	22/02	26/24
MEC1	400	PROGRAM*	29/20		29/23	L				
MEC2	410	PROGRAM*	29/30		29/32		29/37	L		
MFILE	416	PROGRAM*	13/00		17/07		21/20	24/21	24/39	31/10
			13/48		18/23		22/10	24/29	27/51	32/18
MFILE1	425	PROGRAM*	31/20		31/23	L				
MFILE2	426	PROGRAM*	31/25	L	31/30					
MFILE3	437	PROGRAM*	31/42		31/45	L				
MFILE4	440	PROGRAM*	31/47	L	31/52					
MMUX	37	PROGRAM*	14/02		16/18	L	16/45			
MMUXDVC	54	PROGRAM*	16/32		16/37		17/05	L	17/52	
MMUX1	47	PROGRAM*	16/37	L	16/43					
MPS	411	PROGRAM*	17/37		18/42		23/18	25/12	27/27	30/07
MPS1	414	PROGRAM*	30/08		30/13	L				30/15
MP.CADR	2		8/04	D						
MP.CMAP	3		7/52	D						
MP.CNT	2		7/51	D						
MP.FADR	1		8/03	D						
MP.FILE	0		8/02	D						
MP.RAFL	1		7/50	D						
MP.SIZE	3		8/01	D						
MSDVC	103	PROGRAM*	14/03		18/09	L	19/14			
MSDVCLP	110	PROGRAM*	18/18	L	19/21		19/25			
MSPY	350	PROGRAM*	14/07		27/14	L	28/02			
MUXINTX	1		16/15	D	16/22		35/07			

INTINIT SYMBOLIC REFERENCE TABLE.

MUXPNTS	0	EXTERNAL*	16/20	17/46						
MUXQADD	0	EXTERNAL*	16/21							
MUXWRBIT	0	EXTERNAL*	16/12 D	16/23	16/30					
MUXWRBT	0	EXTERNAL*	16/12							
M.ADDR	517	PROGRAM*	13/10 S	17/29 S	18/31 S	26/24 S	29/23	32/01	34/19 L	
			17/21 S	18/22 S	18/38 S	27/49 S	29/27 S	32/05 S		
M.ALL	520	PROGRAM*	13/09	29/26	32/04	34/21 L				
M.FILE	523	PROGRAM*	18/21	27/48	34/24 L					
M.GETE	521	PROGRAM*	17/28	18/37	26/23	34/22 L				
M.SENDE	522	PROGRAM*	17/20	18/30	34/23 L					
NEWCLASS	463	PROGRAM*	16/24	18/16	20/15	24/17	26/17	27/20	33/11 L	34/11
NEWC2	512	PROGRAM*	33/13 S	33/42	34/14 L					
NEWC3	513	PROGRAM*	33/15 S	33/29	34/15 L					
NEWC4	514	PROGRAM*	33/17 S	33/39	34/16 L					
NEWC5	515	PROGRAM*	33/19 S	33/36	34/17 L					
NEWC6	516	PROGRAM*	33/21 S	34/18 L						
NEWSYS	0		5/21 D							
NITYPES	12		13/39	13/47	14/17 D	33/23	34/01			
NUMTYPES	41		8/47 D							
OB.AAP	100		11/11 D							
OB.ACC	2		11/05 D							
OB.ADDOR	4		10/50 D							
OB.ALORD	400		9/39 D							
OB.ATCH	10000		10/15 D							
OB.CALOP	100		10/36 D							
OB.CHMAP	400		10/38 D							
OB.CHNAM	2		9/32 D	34/22	34/23	34/24				
OB.CHNM	4000		11/16 D							
OB.CHOPT	20		10/52 D							
OB.CHTYP	10		10/11 D							
OB.CLOSE	1000		10/12 D							
OB.CPYIN	4		9/32 D							
OB.CPYOP	40		10/33 D							
OB.CPYOT	10		9/33 D							
OB.CRDR	2000		11/15 D							
OB.CREAB	4		9/33 D							
OB.CREBL	4		10/05 D							
OB.CRECL	10		9/34 D							
OB.CREEC	200		9/38 D							
OB.CREPR	40		9/36 D							
OB.CRESP	100		9/37 D							
OB.CRFIL	20		9/35 D							
OB.CRLE	10		11/08 D							
OB.CSUL	1000		11/14 D							
OB.DAE	1000		10/39 D							
OB.DAP	200		11/12 D							
OB.DCRBL	2000		10/13 D							
OB.DDLBL	4000		10/14 D							
OB.DELBL	10		10/06 D							
OB.DLE	20		11/09 D							
OB.DMAP	40000		10/17 D							
OB.DSPAC	200000		11/22 D							
OB.DSTRY	1		9/31 D							

INTINIT
SYMBOLIC REFERENCE TABLE.

OB.DTCH	20000		10/16	D	
OB.ECLM	100000		10/18	D	
OB.FATHR	4		10/32	D	
OB.FDAE	200		10/10	D	
OB.FREZ	1000000		10/21	D	
OB.GET	2000		9/41	D	
OB.GETCP	40000		9/45	D	
OB.GETEV	10		10/45	D	34/22
OB.GETMT	200000		9/47	D	
OB.GIVCP	20000		9/44	D	
OB.GIVE	1000		9/40	D	
OB.GIVMT	100000		9/46	D	
OB.GOD	4000		9/42	D	
OB.GTEVF	20		10/46	D	34/22
OB.IMPL	4		11/06	D	
OB.INCHR	10000		9/43	D	
OB.INMTR	400000		9/48	D	
OB.INSL	40		11/10	D	
OB.INTSP	40		10/35	D	
OB.LOCCL	20		10/01	D	
OB.MONEY	1000000		11/24	D	
OB.NPSW	20000		11/19	D	
OB.OPEN	400		10/11	D	
OB.PCNT	20		10/34	D	
OB.PLMAP	100		10/09	D	34/24
OB.RDFIL	20		10/07	D	34/24
OB.REL	400000		10/20	D	
OB.RESI	100000		11/21	D	
OB.RESO	40000		11/20	D	
OB.SBUSR	400000		11/23	D	
OB.SCLM	200000		10/19	D	
OB.SDINT	4		10/27	D	
OB.SETAG	10000		11/17	D	
OB.SNDEV	4		10/44	D	34/23
OB.SONSP	200		10/37	D	
OB.SPRET	10		10/33	D	
OB.SSUL	400		11/13	D	
OB.STESM	2000		10/40	D	
OB.TEMP	2		10/31	D	
OB.TRDB	200000		10/22	D	
OB.WFILE	40		10/08	D	34/24
OB.XOPN	4000000		10/23	D	
OUTCHN	540	PROGRAM*	17/26	S	35/16 L
PF.ILE	545	PROGRAM*	17/17		36/04 L 36/19
PF.C	6		6/13	D	
PF.CREA	10		6/17	D	
PF.D	4		6/11	D	
PF.E	5		6/12	D	
PF.EE	11		6/16	D	
PF.I	3		6/10	D	
PF.KILL	4200		6/18	D	
PF.P	0		6/07	D	
PF.R	2		6/09	D	

INTINIT
SYMBOLIC REFERENCE TABLE.

P.TEMP3	60		5/40	L					
P.TEMP4	61		5/41	L					
P.TEMP5	62		5/42	L					
P.TEMP6	63		5/43	L					
P.TEMP7	64		5/44	L					
P.USRTIM	131		6/27	D					
P.XPACK	135		6/40	L					
SACTION	600	PROGRAM*	38/22	L					
SBUFPOS	576	PROGRAM*	38/20	L					
SBUFSZ	577	PROGRAM*	38/21	L					
SDVCCNT	164	PROGRAM*	13/14		18/14	19/11	19/13 S	19/45 L	
SDVCDPNT	147	PROGRAM*	18/18		18/27	18/34	19/16	19/24 S	19/40 L
SDVCINTX	2		18/06	D	18/13	38/07			
SDVCNUMB	572	PROGRAM*	18/53		19/09 S	38/15	L		
SDVCPNT	0	EXTERNAL*	18/11		18/52				
SDVCGPT	0	EXTERNAL*	18/12						
SD.CC	1		7/19	D					
SD.CLST	3		7/21	D					
SD.ESM	2		7/20	D					
SD.FDIS	2		7/26	D					
SD.FINT	0		7/24	D					
SD.FIPT	0		7/17	D					
SD.FMPE	1		7/25	D					
SD.MAP	6		7/24	D	7/27				
SD.ORIG	5		7/23	D					
SD.PTRS	4		7/22	D					
SD.RAFL	7		7/27	D					
SD.SIZE	10		7/16	D					
SECRE0	567	PROGRAM*	18/32	S	38/12	L			
SECRSP	570	PROGRAM*	18/29	S	38/13	L			
SEFLAG	573	PROGRAM*	38/17	L					
SFIELD1	0		38/7		38/29	D	39/07		
SFILE	565	PROGRAM*	18/25	S	38/10	L			
SFILESZ	566	PROGRAM*	18/22	S	38/11	L			
SF.II	0		8/17	D					
SF.PCQ1	4		8/18	D					
SF.PCQ2	5		8/19	D					
SMOT	571	PROGRAM*	18/44	S	38/14	L			
SPPROC	557	PROGRAM*	18/47		38/04	L	38/25		
SPPROCSZ	23		18/41		18/48		38/25	D	
SPQUEUE	562	PROGRAM*	18/10		38/07	L			
SPYINTX	5		27/08	D	27/18		41/09		
SPYOBJ5	2		13/27		27/09	D			
SPYPPNT	0	EXTERNAL*	27/16		27/36				
SPYQNT	0	EXTERNAL*	27/17						
SPYQUEUE	627	PROGRAM*	27/15		41/09	L			
SPYSP	624	PROGRAM*	27/21		41/06	L	41/14		
SPYSPMOT	632	PROGRAM*	27/29	S	41/12	L			
SPYSPSZ	7		27/26		27/32		41/14	D	
SPYTMFA	0	EXTERNAL*	27/53	S					
SPYTMES	0	EXTERNAL*	27/50	S					
SPYTMFSZ	500		27/11	D	27/49				
SREQDONE	575	PROGRAM*	38/19	L					

INTINIT
SYMBOLIC REFERENCE TABLE.

T.CLSCD 1737
T.EVCH 1757
T.FILE 1577
T.OPER 1677
T.PROC 777
T.PSEUDO 777
IO: 2000000

8/31 D								
8/32 D								
8/29 D								
8/30 D								
8/26 D	8/27							
8/27 D								
11/04 D	11/08	11/10 D	11/12	11/15	11/17 D	11/20	11/23	
11/05	11/08 D	11/10	11/13	11/15 D	11/17	11/21	11/23 D	
11/05 D	11/08	11/11	11/13 D	11/15	11/19	11/21 D	11/23	
11/05	11/09	11/11 D	11/13	11/16	11/19 D	11/21	11/24	
11/06	11/09 D	11/11	11/14	11/16 D	11/19	11/22	11/24 D	
11/06 D	11/09	11/12	11/14 D	11/16	11/20	11/22 D	11/24	
11/06	11/10	11/12 D	11/14	11/17	11/20 D	11/22		

11.12.40. 10/20/71 SCOP32D OF 08/27/71
11.12.53.\$! CM=18944/045000R AT CP= 0 SEC
11.12.56.GET,INTINIT
11.13.05.GET,SWAP
11.13.12.GET,MYBINS
11.13.27.NOMPASS,I=SWAP,S=0,B=MYBINS
11.14.06. ASSEMBLY COMPLETE.
11.14.10.END
11.14.19.NOMPASS,I=INTINIT,S=0,B=MYBINS
11.15.15. ASSEMBLY COMPLETE.
11.15.21.END
11.15.27.FIN
11.15.27.\$! USER CPU - 14.977 SEC
11.15.27.\$! SCOPE CPU - 9.264 SEC
11.15.28.\$! SCOPE ECS - 7.309 SEC
11.15.28.\$! SCOPE SWAP - 18.334 SEC
11.15.28.\$! DISK CPU - 10.677 SEC
11.15.29.\$! DISK ECS - 11.179 SEC
11.15.29.\$! SYSTEXT - 3113 LINES

