

00010

XLIST

01970

```

00030
00040 ;          PUSHJ PDP, CORE1      ;ITEM = JOB #. TAC = 1K CORE BLOCKS
00050 ;                                ;REQUESTED
00060 ;          RET1                  ;OLD CORE RETURNED.
00070 ;          RET2                  ;OLD CORE RETURNED.  JBTADR, JOBRCL ARE
00080 ;                                ;SET UP
00090 ; JBTADR (ITEM) = POINTER TO JOB AREA. JOBRCL (JBTADR) = HIGHEST LOCAT
00100 ; C(RH) JBTADR IS SET TO 0 OR REQUESTED CORE. JOBRCL IS SET IF CORE G
00110 ; TAC = CORTALLY AT EXIT.
00120 ; NOT A PURE PROCEDURE ...
00130 ;CORTAL: 0                        ;COUNT OF FREE CORE
00140 ;CORTAB: BLOCK +D2                ;TABLE OF 72 BITS. 1 BIT FOR
00150 ;1K OF CORE. 1 = USED. 0 = FREE
00160 ;CORLST: POINT 1, CORTAB, 32      ;POINTER FIRST OUT OF BOUNDS
00170 ;
00180 ;TEMPORARY USAGE
00190 CORUSZ: 0                        ;SIZE OF USERS CODE, OLD.-NEW
00200 CORLOC: 0                        ;POSITION OF 1ST FREE BLOCK
00210 ;
00220 ;AC ASSIGNMENTS
00230 COR1=BUFPNT
00240 COR2=BUFWRD
00250 COR3=UUO
00260 INTERNAL CORE1
00270 EXTERNAL JBTADR, JOBRCL, CORLST, CORTAL, CORTAB

```

```

00280
00290
000002 135500 000042' 00300
000003 202500 000001' 00310
000004 322500 000013' 00320
000005 554504 000000 00330
000006 350000 000012 00340
000007 240500 777766 00350
000010 202500 000000' 00360
000011 402000 000014 00370
000012 260140 000056' 00380
000013 402004 000005' 00390
000014 322040 000037' 00400
000015 200600 000043' 00410
000016 476000 000001' 00420
000017 260140 000044' 00430
000020 254000 000040' 00440
000021 315500 000001 00450
000022 254000 000017' 00460
000023 213000 000012 00470
000024 272500 000001' 00480
000025 202040 000000' 00490
000026 201600 000001 00500
000027 260140 000056' 00510
000030 200500 000001' 00520
000031 240500 000012 00530
000032 542504 000013' 00540
000033 240040 000012 00550
000034 275040 000001 00560
000035 542052 000000 00570
000036 506044 000032' 00580
000037 350003 000000 00590
000040 200040 000000 00600
000041 263140 000000 00610
00620
000042 121004 000036' 00630
000043 440100 000000 00640

```

```

CORE1:  LDB COR1, CORE1P
        MOVEM COR1, CORLOC
        JUMPE COR1, CORGET      ;IF 0 NO CORE NOEW.
        HLRZ COR1, JBTADR(ITEM) ;CLEAR OLD CORE
        AOS COR1                ;HIGHEST ADDRESS AVAIL. FORM
        ASH COR1, -D10
        MOVEM COR1, CORUSZ
CORRET:  CLEARM COR3            ;RETURN OLD CORE
        PUSHJ PDP, CORSTG
CORGET:  SETZM JBTADR(ITEM)     ;CLEAR JOB ADDRESS TO 0
        JUMPE TAC, COROK       ;IF NO CORE TO GET, RETURN
        MOVE COR3, CORE2P      ;SET UP FOR
        SETOM CORLOC           ;HOLE SEARCH
CORG1:   PUSHJ PDP, CORHOL
        JRST CORNG             ;UNSUCCESSFUL SEARCH
        CAMGE COR1, TAC        ;HOLE FOUND
        JRST CORG1
CORGVE:  MOVNS COR1            ;BACKUP CORLOC
        ADDM COR1, CORLOC
        MOVEM TAC, CORUSZ      ;RESERVE HOLE FOR JOB
        MOVEI COR3, 1          ;SET CORTAB
        PUSHJ PDP, CORSTG
        MOVE COR1, CORLOC      ;C(RH)=RELOC, C(LH)=PROTECT
        ASH COR1, D10
        HRRM COR1, JBTADR(ITEM)
        ASH TAC, D10
        SUBI TAC, 1
        HRRM TAC, JOBREL (COR1)
        HRLM TAC, JBTADR(ITEM)
COROK:   AOS (PDP)
CORNG:   MOVE TAC, CORTAL
        POPJ PDP,
CORE1P:  POINT 8, JBTADRR(ITEM),25
CORE2P:  POINT 1, CORTAB

```

		00650	
		00660	
		00670	; PUSHJ PDP, CORHOL ;COR1-HOLES FREE, COR2=TEMP
		00680	; RETURN1 ;COR3=POINTER. RETURN1=END OF STRING RE
		00690	; RETURN2 ;HOLE FOUND.
		00700	; CORLOC CONTAINS 0,1.. N, THE CORE BLK PTR. CORLST IS PTR TO
		00710	; STOP STRING TEST
000044	402000 000012	00720	CORHOL: CLEARM COR1
000045	316600 000000	00730	CORH01: CAMN COR3, CORLST ;TEST DONE
000046	263140 000000	00740	POPJ PDP,
000047	134540 000014	00750	ILDB COR2, COR3 ;TEST NEXT BLOCK
000050	350000 000001	00760	AOS CORLOC
000051	336000 000013	00770	SKIPN COR2
000052	344500 000045	00780	AOJA COR1, CORH01 ;COUNT FREE
000053	322500 000045	00790	JUMPE COR1, CORH01
000054	350003 000000	00800	AOS (PDP)
000055	263140 000000	00810	POPJ PDP,
		00820	:: PUSHJ PDP, CORSTG ;CORUSZ=SIZE RESERVED. CORLOC=LOCATION
		00830	; IF COR3=0, THEN FREE CORE
		00840	; IF COR3=1, THEN RESERVE CORE.
		00850	; USES COR1, COR2, COR3
		00860	; CORTAL=COUNT OF FREE CORE, AND IS UPDATED
000056	261140 000001	00870	CORSTG: PUSH PDP, TAC ;SAVE NO. OF BLOCKS BEING REQ
000057	200540 000000	00880	MOVE COR2,CORUSZ ;FIX CORTALLY
000060	332000 000014	00890	SKIPE COR3
000061	213000 000013	00900	MOVNS COR2
000062	272540 000040	00910	ADDM COR2, CORTAL
000063	200040 000001	00920	MOVE TAC, CORLOC ;GET POINTER-1
000064	231040 000044	00930	IDIVI TAC, +D36 ;TAC=WORD,TAC1=BIT
000065	270040 000076	00940	ADD TAC, CORLP1 ;FORM BYTE POINTER
000066	213000 000002	00950	MOVNS TAC1
000067	271100 000044	00960	ADDI TAC1, +D36
000070	137100 000077	00970	DPB TAC1, [POINT 6, TAC, 5]
000071	200540 000000	00980	MOVE COR2, CORUSZ
000072	136600 000001	00990	CORLOP: DPBI COR3, TAC ;SET OR CLEAR EACH BIT IN USE
000073	367540 000072	01000	SOJG COR2, CORLOP
000074	262140 000001	01010	POP PDP, TAC
000075	263140 000000	01020	POPJ PDP,
		01030	
000076	000100 000043	01040	CORLP1: XWD 100, CORTAB
		01050	
000077	360600 000001	01060	END,

THERE ARE NO ERRORS

PROGRAM BREAK IS 000100

CORE - CORE ALLOCATION ROUTINE
SYMBOL TABLE

PAGE 6

A	000000	INT
AC1	000015	INT
AC2	000016	INT
AC3	000017	INT
AL	000001	INT
ASSCON	400000	INT
ASSPRG	200000	INT
B	000014	INT
BUFPNT	000012	INT
BUFWRD	000013	INT
CLOSB	002000	INT
CLSIN	000002	INT
CLSOUT	000001	INT
COR1	000012	
COR2	000013	
COR3	000014	
CORE1	000002'	INT
CORE1P	000042'	
CORE2P	000043'	
CORG1	000017'	
CORGET	000013'	
CORGVE	000023'	
CORH01	000045'	
CORH0L	000044'	
CORLOC	000001'	
CORLOP	000072'	
CORLP1	000076'	
CORLST	000045'	EXT
CORNG	000040'	
COROK	000037'	
CORRET	000011'	
CORSTG	000056'	
CORTAB	000076'	EXT
CORTAL	000062'	EXT
CORUSZ	000000'	
D	000017	INT
DAT	000005	INT
DCL	000001	INT
DCW	020000	INT
DDI	000007	INT
DDO	000006	INT
DDTMEM	000037	INT
DDTSYM	000036	INT
DEN	000004	INT
DEVADR	000007	INT
DEVBUF	000006	INT
DEVCHR	000001	INT
DEVCTR	000011	INT
DEVDAT	000006	INT
DEVIAD	000007	INT
DEVIOS	000002	INT
DEVLOG	000005	INT
DEVMOD	000004	INT
DEVNAM	000000	INT
DEVOAD	000010	INT

CORE - CORE ALLOCATION ROUTINE
SYMBOL TABLE

PAGE 7

DEVPTR	000010	INT
DEVSER	000003	INT
DGF	000012	INT
DIN	000003	INT
DLK	000005	INT
DOU	000002	INT
DR	000016	INT
DRL	000000	INT
DSI	000011	INT
DSO	000010	INT
DTW	040000	INT
DVAVAL	000040	INT
DVCDR	100000	INT
DVDIR	000004	INT
DVDIRI	400000	INT
DVIN	000002	INT
DVLPT	040000	INT
DVMTA	000020	INT
DVOUT	000001	INT
DVTY	000010	INT
ENTRB	020000	INT
I	000010	INT
IB	000013	INT
IBUFR	200000	INT
INITB	400000	INT
INPB	010000	INT
IO	000020	INT
IOACT	010000	INT
IOREG	000002	INT
IORKTL	040000	INT
IOCON	000040	INT
IODEND	020000	INT
IODERR	200000	INT
IODISC	400000	INT
IODONE	400000	INT
IODTER	100000	INT
IOFND	000040	INT
IOFST	000004	INT
IOIMPM	400000	INT
IONRCK	000100	INT
IORDEL	000100	INT
IORET	000020	INT
IOS	000000	INT
IOSTRT	000010	INT
IOUSE	400000	INT
IOW	000001	INT
IOWC	000020	INT
IOWS	400000	INT
ITEM	000004	INT
JBFAADR	000000	INT
JBFCR	000002	INT
JBFPTR	000001	INT
JBTAADR	000042	EXT
JBUF	000005	INT
JDAT	000011	INT

CORE - CORE ALLOCATION ROUTINE
SYMBOL TABLE

PAGE 8

JERR	002000	INT
JIOW	100000	INT
JNA	004000	INT
JORREL	000035	EXT
LOOKB	040000	INT
MTW	010000	INT
OBUEB	100000	INT
OUTPR	004000	INT
PDP	000003	INT
PICHN	000100	INT
PROG	000007	INT
RUN	200000	INT
RUNABL	204000	INT
TAC	000001	INT
TAC1	000002	INT
TEM	000010	INT
TTYATC	020000	INT
TTYUSE	010000	INT
USRMOD	010000	INT
UUO	000014	INT

END OF ASSEMBLY