

00010

XLIST

01970

```

00030
00040 ;ROUTINE TO RETURN NEXT ALPHANUMERIC STRING
00050 ; IN COMMAND LINE (SIXBIT)
00060 ; CALL: MOVE TAC, BYTE POINTER TO FIRST CHAR.-1
00070 ; PUSHJ PDP, CTEXT
00080 ; SIXBIT STRING RETURN LEFT JUSTIFIED IN AC TAC1
00090
00100 INTERNAL CTEXT, CTEXT1
00110
00120 T=BUFPNT ;TEMPORARY ACS
00130 T1=BUFWRD
00140 T2=UU0
00150
000000 201100 000000 00160 CTEXT1: MOVEI TAC1, 0 ;CHECK PREVIOUS BREAK CHAR.
000001 135500 000001 00170 LDB T, TAC
000002 306500 000015 00180 CAIN T, 15 ;RETURN IF CR.
000003 263140 000000 00190 POPJ PDP,
00200
000004 260140 000035' 00210 CTEXT: PUSHJ PDP, SKIPS ;SKIP LEADING SPACES, TABS
000005 200500 000231' 00220 MOVE T,[POINT 6,TAC1]
000006 201100 000000 00230 MOVEI TAC1, 0
000007 135540 000001 00240 LDB T1, TAC
000010 254000 000012' 00250 JRST CTEX1
000011 134540 000001 00260 CTEX0: ILDB T1,TAC
000012 260140 000024' 00270 CTEX1: PUSHJ PDP, CTEX ;IS IT ALPHANUMERIC
000013 263140 000000 00280 POPJ PDP, ;NO, EXIT
000014 640540 000040 00290 TRC T1,40 ;CONVERT TO SIXBIT
000015 136540 000012 00300 IDPB T1,T ;YES, STORE AWAY
000016 312500 000232' 00310 CAME T,[POINT 6,TAC1,35] ;SIX CHARACTER YET
000017 254000 000011' 00320 JRST CTEX0 ;NO
000020 134540 000001 00330 CTEX2: ILDB T1, TAC ;YES
000021 260140 000024' 00340 PUSHJ PDP, CTEX ;IS IT ALPHANUMERIC
000022 263140 000000 00350 POPJ PDP, ;NO, EXIT
000023 254000 000020' 00360 JRST CTEX2
00370
00380 ;SCAN FOR ALPHANUMERIC CHAR IN T1
00390 CTEX: CAIL T1, "0"
00400 CAILE T1, "Z" ;LETTERS ARE LARGER THAN NOS.
00410 POPJ PDP, ;NEITHER
00420 CAILE T1, "9"
00430 CAIL T1, "A"
00440 AOS (PDP) ;LETTER OR NUMBER
00450 POPJ PDP,

```

```
00460
00470 ;ROUTINE TO IGNORE LEADING SPACES AND TABS
00480 ;CALL: MOVE TAC, BYTE POINTER TO CHAR.
00490 ; PUSHJ PDP, SKIPS1
00500
00510 INTERNAL SKIPS1
00520
000033 135500 000001 00530 SKIPS1: LDB T, TAC
000034 254000 000036' 00540 JRST SKIPSA
00550
00560
00570 ;ROUTINE TO IGNORE LEADING SPACES AND TABS
00580 ;CALL: MOVE TAC, BYTE POINTER TO CHAR.-1
00590 ; PUSHJ PDP, SKIPS
00600
00610 INTERNAL SKIPS
00620
000035 134500 000001 00630 SKIPS: ILDB T, TAC
000036 302500 000040 00640 SKIPSA: CAIE T, " "
000037 306500 000011 00650 CAIN T, 11
000040 254000 000035' 00660 JRST SKIPS
000041 263140 000000 00670 POPJ PDP,
```

```

00680
00690 ;ROUTINE TO APPEND A "?" TO INPUT STRING AND SET AS OUTPUT
00700 ;CALL: MOVE TAC, BYTE POINTER TO LAST CHAR. IN INPUT STRING
00710 ; PUSHJ PDP, COMERA
00720
00730 INTERNAL COMERA
00740
000042 133000 000001 00750 COMERA: IBP TAC
00760
00770
00780 ;ROUTINE TO REPLACE LAST CHARACTER IN INPUT STRING BY "?"
00790 ;AND SET AS OUTPUT
00800 ;CALL: MOVE TAC, BYTE POINTER TO LAST CHAR. IN INPUT STRING
00810 ; PUSHJ PDP, COMERR
00820
00830 INTERNAL COMERR
00840
00850 EXTERNAL TTYBUF
00860
000043 200240 000001 00870 COMERR: MOVE DAT,TAC
000044 201506 000000 00880 MOVEI T,TTYBUF(DEVDAT) ;IS THERE MORE THAN 3 WORDS I
000045 305505 777775 00890 CAIGE T,-3(DAT)
000046 541252 000003 00900 HRRJ DAT,3(T) ;YES, TYPE ONLY 3
000047 201500 000077 00910 MOVEI T, "?" ;APPEND "?"
000050 137500 000005 00920 DPB T, DAT
000051 265040 000060 00930 JSP TAC,CONMES
00940
000052 202331 747222 00950 ASCIZ / MONITOR COMMAND ERROR
000053 522372 220206
000054 476331 540634
000055 421010 551244
000056 476441 505000 00960 /
00970

```

```

00980
00990
01000 ;ROUTINE TO PRINT CARRIAGE RETURN-LINE-FEED
01010 ;CALL: MOVE DAT,BYTE POINTER TO OUTPUT
01020 ; PUSHJ PDP,CRLF
01030
01040 INTERNAL CRLF
01050
000057 201040 000233' 01060 CRLF: MOVEI TAC,[ASCII /
01070 /]
01080
01090 ;ROUTINE TO MOVE ASCII CHAR. STRING TO CONSOLE OUTPUT BUFFER
01100 ; CALL: MOVE DAT, BYTE POINTER TO LAST OUTPUT CHARACTER
01110 ; MOVEI TAC, ADDRESS OF ASCII MESSAGE
01120 ; PUSHJ PDP, CONMES
01130 ; STRING TERMINATED BY NULL
01140 ; CONMES DOES NOT START TTY
01150
01160 INTERNAL CONMES
01170
000060 261140 000002 01180 CONMES: PUSH PDP, TAC1
000061 505040 440700 01190 HRLI TAC, 440700 ;FORM BYTE POINTER
000062 134100 000001 01200 CON0: ILDB TAC1, TAC
000063 322100 000066' 01210 JUMPE TAC1, CON1
000064 136100 000005 01220 IDPB TAC1, DAT
000065 254000 000062' 01230 JRST CON0
000066 262140 000002 01240 CON1: POP PDP, TAC1
000067 263140 000000 01250 POPJ PDP,
01260
01270 ;ROUTINE TO PRINT "TOO FEW ARGUMENTS"
01280 ;CALL: MOVE DAT,BYTE POINTER
01290 ; PUSHJ PDP,NOTENF
01300
01310 INTERNAL NOTENF
01320
000070 265040 000060' 01330 NOTENF: JSP TAC,CONMES
01340
000071 522371 720214 01350 ASCIIZ /TOO FEW ARGUMENTS
000072 426564 040644
000073 436531 542634
000074 522461 505000 01360 /

```

```

01370
01380
01390 ;ROUTINE TO INITIALIZE A JOB
01400 ;CALL: MOVE ITEM, JOB NUMBER
01410 ; PUSHJ PDP, JOBINI
01420 ; DEVDAT IS SETUP ON RETURN
01430
01440 INTERNAL JOBINI, JOBZER
01450 EXTERNAL JBTSTS, PJOBN, CORE1, MONJOB, ESTOP, TYATT
01460
000075 326200 000100' 01470 JOBINI: JUMPJ ITEM, JOB0 ;HAS A JOB NUMBER BEED ASSIGNED?
000076 260140 000135' 01480 PUSHJ PDP, ASSJOB ;NO, THEN ASSIGN ONE.
000077 254000 000060' 01490 JRST CONMESS ;NONE LEFT
000100 260140 000000 01500 JOB0: PUSHJ PDP, TTYATT ;ATTACH TTY
000101 255000 000000 01510 JFCL ;ALREADY ATTACHED
000102 205040 004000 01520 MOVSI TAC, JNA ;CLEAR JOB NO. ASSIGNED BIT
000103 412044 000000 01530 ANDCAM TAC, JBTSTS(ITEM) ;TO FORCE MESSAGE
000104 205040 646471 01540 MOVSI TAC, 646471 ;SIXBIT /TTY/
000105 202046 000005 01550 MOVEM TAC, DEVLOG(DEVDAT) ;SET LOGICAL NAME
000106 201500 400000 01560 MOVEI T, ASSCON ;SET ASSIGNED BY CONSOLE BIT
000107 436506 000004 01570 IORM T, DEVMOD(DEVDAT)
000110 260140 000172' 01580 PUSHJ PDP, DEASTY ;DE-ASSIGNED ALL DEVICES EXCEPT TTY
000111 322440 000000 01590 JUMPE JDAT, CPOPJ ;DOES JOB ALREADY HAVE CORE?
01600
01610
01620 ;ROUTINE TO RELEASE ALL DEVICES AND FLUSH CORE
01630 ;RUN AS MONITOR JOB
01640 ;CALL: SAME AS JOBINI
01650
01660 EXTERNAL IOKILL
01670
000112 541100 000115' 01680 JOBZER: HRFI TAC1, JOB1 ;ADDRESS OF SUB. TO RELEASE DEV.
000113 505100 000000 01690 JOBZR1: HRFI TAC1, ESTOP ;RETURN ADDRESS AFTER IORELS
000114 254000 000000 01700 JRST MONJOB ;GO SCHEDULE MONITOR JOB
01710
000115 260140 000000 01720 JOB1: PUSHJ PDP, IOKILL ;KILL ALL IO DEVICES
000116 201040 000000 01730 MOVEI TAC, 0 ;FLUSH ALL CORE
000117 370003 000000 01740 SOS (PDP) ;CORE1 WILL SKIP RETURN
000120 254000 000000 01750 JRST CORE1
01760

```

```

01770
01780
01790 ;ROUTINE TO KILL CURRENT JOB
01800 ;CALL PUSHJ PDP,JOBKIL
01810
01820 INTERNAL JOBKIL
01830 EXTERNAL JBTSTS,TTYKIL
01840
000121 350003 000000 01850 JOBKIL: AOS (PDP) ;SUPRESS CR-LF AND JOB INIT. CHECK
000122 201040 000001 01860 MOVEI TAC,1
000123 260140 000173' 01870 PUSHJ PDP,DEASAL ;DEASSIGN ALL DEVICES
000124 322440 000132' 01880 JUMPE JDAT,JOBKA ;DOES JOB HAVE CORE?
000125 541100 000127' 01890 HRRJ TAC1,JOBK2 ;YES,RELEAS DEVICES AND CLEAR CORE
000126 254000 000113' 01900 JRST JOBZR1
01910
000127 261140 000006 01920 JOBK2: PUSH PDP,DEV DAT ;SAVE DEV DAT FOR TTYKIL
000130 260140 000115' 01930 PUSHJ PDP,JOB1 ;FLUSH CORE AND RELEASE DEVICES
000131 262140 000006 01940 POP PDP,DEV DAT
000132 205040 002000 01950 JOBKA: MOVSI TAC,JERR ;CLEAR ALL BUT JOB ERROR BIT
000133 202044 000103' 01960 MOVEM TAC,JBTSTS(ITEM) ;IN JOB STATUS WORD
000134 254000 000000 01970 JRST TTYKIL ;RETURN TTY TO VIRGIN STATE
01980
01990
02000 ;ROUTINE TO ASSIGN AN UNUSED JOB NUMBER
02010 ;CALL PUSHJ PDP,ASSJOB
02020 ; ERROR RETURN TAC DESTROYED
02030 ; OK RETURN ITEM SET, TAC AND TAC1 PRESERVED
02040
02050 INTERNAL ASSJOB
02060 EXTERNAL MJOBN,JBTSTS,CPOPJ1
02070
000135 205200 000000 02080 ASSJOB: MOVSI ITEM,MJOBN
000136 253200 000137' 02090 AOBJN ITEM,+.1
000137 205540 004000 02100 MOVSI T1,JNA
000140 612544 000133' 02110 TDNE T1,JBTSTS(ITEM)
000141 253200 000140' 02120 AOBJN ITEM,-.1
000142 325200 000145' 02130 JUMPGE ITEM,ASSJB1
000143 405200 777777 02140 ANDI ITEM,-1
000144 254000 000000 02150 JRST CPOPJ1
02160
000145 265040 000060' 02170 ASSJB1: JSP TAC,CONMES
000146 452370 220206 02180 ASCIZ /JOB CAPACITY EXCEEDED
000147 406410 141622
000150 522624 042660
000151 416130 542212
000152 420321 200000 02190 /

```



```

02200
02210
02220 ;ROUTINE TO DEASSIGN A DEVICE
02230 ;CALL: MOVE DEVDAT, DEVICE DATA BLOCK
02240 ; MOVE ITEM, JOB NUMBER
02250 ; PUSHJ PDP, DEASG
02260 ; ERROR NOT PREVIOUSLY ASSIGNED
02270 ; OK RETURN WITH DEVICE DEASSIGNED
02280
02290 INTERNAL DEASG
02300 EXTERNAL PJOBN, CPOPJ1
02310
000153 135500 000000 02320 DEASG: LDB T, PJOBN ;WAS DEVICE ASSIGNED TO THIS JOB?
000154 312500 000004 02330 CAME T, ITEM
000155 263140 000000 02340 POPJ PDP, ;NO, RETURN
000156 205500 400000 02350 MOVSI T,DVDIRIN ;CLEAR DIRECTORY IN CORE BIT
000157 412506 000004 02360 ANDCAM T,DEVMOD(DEVDAT)
000160 201500 400000 02370 MOVEI T, ASSCON ;YES, IS ASSIGNED BY CONSOLE BIT ON?
000161 616506 000004 02380 TDNN T, DEVMOD(DEVDAT)
000162 263140 000000 02390 POPJ PDP, ;NO
000163 413506 000004 02400 ANDCAB T,DEVMOD(DEVDAT) ;YES, CLEAR BIT
000164 402006 000005 02410 SETZM DEVLOG(DEVDAT) ;CLEAR LOGICAL NAME
000165 602500 200000 02420 TRNE T, ASSPRG ;IS ASSIGNED BY PROGRAM BIT ON?
000166 254000 000144' 02430 JRST CPOPJ1 ;YES
000167 201500 000000 02440 MOVEI T,0 ;NO, CLEAR JOB NUMBER
000170 137500 000153' 02450 DPB T, PJOBN
000171 254000 000166' 02460 JRST CPOPJ1
02470
02480 ;ROUTINE TO DEASSIGN ALL DEVICES EXCEPT LOGICAL TTY
02490 ;CALL: MOVE ITEM, JOB NUMBER
02500 ; PUSHJ PDP, DEASTY
02510
02520 INTERNAL DEASTY
02530
000172 205040 646471 02540 DEASTY: MOVSI TAC, 646471 ;SIXBIT /TTY/
02550 ;FALL INTO DEASAL
02560
02570 ;ROUTINE TO DEASSIGN ALL DEVICES EXCEPT ONE
02580 ;CALL: MOVE ITEM, JOB NUMBER
02590 ; MOVE TAC, LOGICAL DEVICE NOT TO BE DEASSIGNED
02600 ; PUSHJ PDP, DEASAL
02610
02620 INTERNAL DEASAL
02630 EXTERNAL DEVLST
02640
000173 261140 000006 02650 DEASAL: PUSH PDP,DEVDAT
000174 554300 000000 02660 HLRZ DEVDAT,DEVLST ;SEARCH ALL DBBS
000175 312046 000005 02670 DEAS1: CAME TAC, DEVLOG(DEVDAT) ;IS THIS THE LOGICAL NAME
000176 260140 000153' 02680 PUSHJ PDP, DEASG ;NO, TRY TO DEASSIGN IT
000177 255000 000000 02690 JFCL ;IGNORE IF CANT
000200 554306 000003 02700 HLRZ DEVDAT, DEVSER(DEVDAT)
000201 326300 000175' 02710 JUMPN DEVDAT, DEAS1
000202 262140 000006 02720 POP PDP,DEVDAT
000203 263140 000000 02730 POPJ PDP,

```

```

02740
02750
02760 ;ROUTINE TO READ CONSOLE AND CONVERT ANY RADIX NUMBER
02770 ; CALL: MOVE TAC1, DESIRED RADIX
02780 ; MOVE TAC, BYTE POINTER TO FIRST CHARACTER
02790 ; PUSHJ PDP, ANYRIN
02800 ; ERROR EXIT ;AC TAC1=0 IF NO NUMBER BEFORE FIRST NO
02810 ; NORMAL EXIT ;AC TAC1 CONTAINS NUMBER
02820 ; SCAN STOPS ON FIRST ILLEGAL CHARACTER OR CR
02830 ; IGNORES SPACES, TABS, RUBOUTS
02840
02850 INTERNAL OCTIN,ANYRIN,DECIN,DECIN1
02860 EXTERNAL CPOPJ1, CPOPJ
02870
02880 C=BUFPNT ;CHARACTER AC
02890 R=BUFWRD ;RADIX AC
02900
000204 133000 000001 02910 DECIN1: IBP TAC ;SKIP SEPARATION CHARACTER
000205 201100 000012 02920 DECIN: MOVEI TAC1,12 ;DECIMAL INPUT
000206 254000 000210' 02930 JRST ANYRIN
02940
000207 201100 000010 02950 OCTIN: MOVEI TAC1, 10
000210 200540 000002 02960 ANYRIN: MOVE R, TAC1
000211 201100 000000 02970 MOVEI TAC1, 0
000212 260140 000033' 02980 PUSHJ PDP, SKIPS1 ;SKIP LEADING SPACES, TABS
000213 135500 000001 02990 LDB C, TAC
000214 306500 000015 03000 CAIN C, 15 ;CHECK FOR INITIAL CR
000215 263140 000000 03010 POPJ PDP, ;EXIT WITH 0 IN TAC1
000216 302500 000055 03020 OCT0: CAIE C,"-" ;DASH?
000217 306500 000015 03030 CAIN C,15 ;CR?
000220 254000 000171' 03040 JRST CPOPJ1 ;YES, ONLY LEGAL TERMINATORS
000221 275500 000060 03050 SUBI C,60
000222 321500 000224' 03060 JUMPL C, OCTE ;ERROR IF NEGATIVE
000223 311500 000013 03070 CAML C, R ;OR .GE. RADIX
000224 344100 000111' 03080 OCTE: AOJA TAC1,CPOPJ ;ERROR,INSURE TAC1 NON-ZERO
000225 220100 000013 03090 IMUL TAC1, R
000226 270100 000012 03100 ADD TAC1, C
000227 134500 000001 03110 OCT1: ILDB C, TAC
000230 254000 000216' 03120 JRST OCT0
03130
000231 440600 000002 03140 END,
000232 000600 000002
000233 064240 000000

```

THERE ARE NO ERRORS

PROGRAM BREAK IS 000234

COMCSS - COMMAND COMMON ROUTINES  
 SYMBOL TABLE

A	000000	INT
AC1	000015	INT
AC2	000016	INT
AC3	000017	INT
AL	000001	INT
ANYRIN	000210'	INT
ASSCON	400000	INT
ASSJB1	000145'	
ASSJOB	000135'	INT
ASSPRG	200000	INT
R	000014	INT
BUFPNT	000012	INT
BUFWRD	000013	INT
C	000012	
CLOSB	002000	INT
CLSIN	000002	INT
CLSOUT	000001	INT
COMERA	000042'	INT
COMERR	000043'	INT
CON0	000062'	
CON1	000066'	
CONMES	000060'	INT
CORE1	000120'	EXT
CPOPJ	000224'	EXT
CPOPJ1	000220'	EXT
CRLF	000057'	INT
CTEX	000024'	
CTFX0	000011'	
CTFX1	000012'	
CTFX2	000020'	
CTFXT	000004'	INT
CTFXT1	000000'	INT
D	000017	INT
DAT	000005	INT
DCL	000001	INT
DCW	020000	INT
DDI	000007	INT
DDO	000006	INT
DDTMEM	000037	INT
DDTSYM	000036	INT
DEA1	000175'	
DEASAL	000173'	INT
DEASG	000153'	INT
DEASTY	000172'	INT
DECIN	000205'	INT
DECIN1	000204'	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

DEVLST	000174'	EXT
DEVMOD	000004	INT
DEVNAM	000000	INT
DEVOAD	000010	INT
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
DVTTY	000010	INT
FNTRB	020000	INT
ESTOP	000113'	EXT
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
IOEND	000040	INT
IOFST	000004	INT
IOIMPM	400000	INT
IOKILL	000115'	EXT
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
ITFM	000004	INT

JBFADR	000000	INT
JBFCTR	000002	INT
JBFPTR	000001	INT
JBTSTS	000140'	EXT
JBUF	000005	INT
JDAT	000011	INT
JERR	002000	INT
JIOW	100000	INT
JNA	004000	INT
JOR0	000100'	
JOR1	000115'	
JORINI	000075'	INT
JORK2	000127'	
JORKA	000132'	
JORKIL	000121'	INT
JORZER	000112'	INT
JORZR1	000113'	
LOOKR	040000	INT
MJOBN	000135'	EXT
MONJOB	000114'	EXT
MTW	010000	INT
NOTENF	000070'	INT
OBUFB	100000	INT
OCT0	000216'	
OCT1	000227'	
OCTE	000224'	
OCTIN	000207'	INT
OUTPB	004000	INT
PDP	000003	INT
PICHN	000100	INT
PJOBN	000170'	EXT
PROG	000007	INT
R	000013	
RUN	200000	INT
RUNABL	204000	INT
SKIPS	000035'	INT
SKIPS1	000033'	INT
SKIPSA	000036'	
T	000012	
T1	000013	
T2	000014	
TAC	000001	INT
TAC1	000002	INT
TEM	000010	INT
TTYATC	020000	INT
TTYATT	000100'	EXT
TTYBUF	000044'	EXT
TTYKIL	000134'	EXT
TTYUSE	010000	INT
USRMOD	010000	INT
UUO	000014	INT

END OF ASSEMBLY