

V *k m
A ?;

1
FLOAT - ADDRESSES AND VECTORS MACRO Y05.02 Thursday 23-May 85 10:38 Page 1

SEQ 000

1

.REM *

IDENTIFICATION

PRODUCT CODE: AC 8789C-MC
PRODUCT NAME: CZFLACO FLOAT UTILITY PROG
PRODUCT DATE: MAY 1985
MAINTAINER: DIAGNOSTIC ENGINEERING

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE :
USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT
SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1977,1985 BY DIGITAL EQUIPMENT CORPORATION

CONTENTS

- 1.0 ABSTRACT
- 2.0 REQUIREMENTS
- 2.1 EQUIPMENT
- 3.0 LOADING PROCEDURE
- 3.1 METHOD
- 4.0 STARTING PROCEDURE
- 4.1 STARTING ADDRESSES
- 4.2 RESTART ADDRESS
- 5.0 OPERATING PROCEDURE
- 5.1 OPERATOR ACTION
- 5.2 'FA' FLOATING ADDRESS OPTION
- 'VA' FLOATING VECTOR OPTION
- 6.0 ERRORS
- 6.1 HALTS, TRAPS, OTHER FAILURES
- 6.2 INVALID RESPONSES
- 7.0 RESTRICTIONS
- 8.0 MISCELLANEOUS
- 8.1 TERMINAL ADDRESS MODIFICATION
- 8.2 DELETE FEATURE
- 8.3 CONTROL/C, CONTROL/Z, CONTROL/A, CONTROL/S AND CONTROL/Q
- 8.4 DN11'S AND PAC11'S
- 8.5 CHANGES TO PROGRAM

1.0 ABSTRACT

Float is a utility program to aide the operator with determining the addresses and vector of devices in the floating address or vector areas. The addresses and vectors given are completely compatible with all DEC standard software.

2.0 REQUIREMENTS

2.1 Equipment

Any PDP-11 processor with 4k of memory and a terminal.

3.0 LOADING PROCEDURE

3.1 Method

The program is supplied on the diagnostic media. Refer to the XXDP operating manual for further information.

4.0 STARTING PROCEDURE

4.1 Starting Addresses

By starting at address 200(8) or 204(8) the restart address is initialized, the program types name and version, control characters and will prompt operator to set up terminal fill count.

4.2 Restart Address

Location 204(8) is the only valid restart address.

5.0 OPERATING PROCEDURE

5.1 Operator Action

When the program is loaded and started the title and version are typed, control characters are listed, and the operator is prompted with:

TERMINAL TYPES:

- A = LA36 NO FILL
- L = LA120 100 FILL CHARACTER
- V = VT52 OR VT100 50 FILL CHARACTER

ENTER TERMINAL TYPE. (A,L,V)?

To this the operator responds

'A'<cr> If terminal is an LA36 or requires nofill.

'L'<cr> If terminal is a LA120 or requires fill for
a carriage return.
'V'<cr> If terminal is a VT52 OR VT100 @ 300 baud or more,
or requires fill for a line feed.

After the terminal fill has been entered or program has been restarted
the program types.

POSSIBLE OPTIONS RE
FA-FLOATING ADDRESSES. (DJ,DH,DQ,ETC.)
VA-FLOATING ADDRESSES AND VECTORS
HE-HELP
EX-EXIT
ENTER OPTION:

At this point the program is waiting for a two (2) character response
followed by a carriage return <cr> defining which option is wanted.
The valid responses are:

'FA' if the floating address option is desired.
'VA' if the floating vector option is desired.
'HE' if the help message is desired.
'EX' if it is desired to exit back to the XXDP monitor.

The program will then ask:

"How many of each does the system have." and then type a list
of devices depending on the "option" selected. The list is typed one
device at a time followed by a question mark. The operator responds
to each question with the number of each device he has in decimal
until the list is completed. (For exceptions and control characters
see miscellaneous.) At the end of the list the program will type the
address or vector information based on the number supplied by the
operator. (See para. 5.2 and 5.3 for format of data.)

5.2 'FA' - Floating Address Option 'VA' - Vector And Address Option

This option will ask for the decimal number of each device in the
floating address range or floating vector range.

After receiving input for each device the program will type the device
name, the address, the module number with the addressing log'c, and
the jumper(s) or switches(es) to be cutout or turned off in the
following format.

DEVICE	CSR	VECT	COMMENT
----	---	----	-----
NAME			
	ADDRESS	VECTOR	NOTES

WHERE:

NAME = ACTUAL DEVICE NAME (I.E. DJ11)
 ADDRESS = CSR ADDRESS OF THE DEVICE
 VECTOR = VECTOR ADDRESS

NOTE: DEVICES ARE LISTED IN ASCENDING ADDRESS ORDER.

6.0 ERRORS

6.1 Halts, Traps, Other Failures.

This program is not intended to be a diagnostic or system sizer. It does no checking of device addresses present or perform any other diagnostic functions. If program halts, traps, gets caught in a loop run CPU, memory and/or terminal diagnostics.

"Do not use program to trouble shoot any failures."

6.2 Invalid Responses

6.2.1 Invalid Response To 'Terminal Type (A,L,V)?'. - The program will type:

VALID RESPONSES ARE--

'A' - LA36 OR NO FILL
 'L' - LA120 OR FILL FOR <CR>
 'V' - VT52 AND VT100 OR FILL FOR <LF>?

And wait for response.

6.2.2 Invalid Response To "Option." - The program will type:

FA - FLOATING ADDRESSES (DJ,DH,DQ,DU,DUP,LK,DMC/R,DZ,KMC,LPP)
 (VM21,VM31,DWR,RL,LPA,KW,RES,RX,DRW,DRB,DMP,DPV,ISB,DMV)
 (UNA,UDA,DMF,VS100,TU81,KMV,DHV/DHU,DMZ32/CPI32(ASYN)
 (CPI32(SYN),QVSS) ➡

VF - VECTORS AND ADDRESSES OF DEVICES IN FLOATING VECTOR AREA.
 (DC,TU58,KL,DLA/B/W,DP,DMA,DN,DMB,DR,PAR,PAP,LPD,DT,DX)
 (DLC/D/E,DJ,DH,GT,VSV,LPS,DQ,KW,DU,DUP,DV/DM,LK,DWN,DMC/R)
 (DZ,KMC,LPP,VM21,VM31,VTV,DRW,RL,TS,LPA,IP,KW,RX,DRW,DRB)
 (DMP,DPV,ML,ISB,DMV,UNA,UDA,DMF,PCL,VS100,TU81,KMV,KCT32)
 (IEX,DHV/DHU,DMZ32/CPI32(ASYN),CPI32(SYN),QNA,QVSS,LNV11)
 HE - HELP MESSAGE
 EX - EXIT COMMAND

And will ask again "Option:".

6.2.3 Invalid Response When Typing Device Counts. - The program will type:

"That's not a valid number!"

If an alpha character or negative value is entered.

or

"That's too many! Only XX allowed!"

If a number greater than the maximum allowed for that device on a single system is supplied.

7.0 RESTRICTIONS

None

8.0 MISCELLANEOUS

8.1 Using Other Than Console Terminal.

Locations 700-706 contain the addresses of the terminal status buffer registers. To use another DL11-A just change these four (4) locations to the desired addresses.

8.2 Delete Feature

The character delete is proceeded and followed with a backslash on a hardcopy terminal. On a video terminal, the characters will be removed from the screen.

8.3 Control/C, Control/Z, Control/S, Control/Q and Control/A

Control C (+C)

Program is restarted at 200 when a CNTRL/C is typed with program waiting for input.

Control Z (+Z)

Program is restarted at print when a CNTRL/Z is typed and will print the selected devices. It is a means of getting an abbreviated list of devices rather than going through the entire list during question and answer session.

Control S (+S)

When CNTRL/S is typed the program will stop printing and wait for a CNTRL/Q.

Control Q (+Q)

When a CNTRL/Q is typed the program will continue printing.

Control A (+A)

When a CNTRL/A is typed the program will back up one device on the list. It allows one to correct errors made during the question and answer session by backing up the list and changing the value assigned to the number of devices.

8.4 DN11'S AND PA611'S

8.4.1 The DN11 Can Have Four (4) Lines Per Block - (1 control module). Specify the number of controls.

8.4.2 The PA611's Can Have Two (2) Readers/Punches Per Controller Specify the number of controllers.

8.5 CHANGES TO PROGRAM

REVISION C INCLUDED THE FOLLOWING CHANGES.

1. PROVISION HAS BEEN MADE TO ALLOW AT LEAST 2 DEVICES PER SYSTEM. NOT ALL DEVICES CAN BE EXPANDED TO THREE OR MORE WITHOUT MODIFICATIONS TO THE DEVICE HANDLER WITHIN THE PROGRAM.

2. THE FOLLOWING DEVICES WERE ADDED:

KMC11,LPP11,VMV21,VMV31,VTVO1,DWR70,RL11,RLV11
TS11,TU80,LPA11,IP11,IP300,KW11-C,RESER,RX11
RX211,RXV11,RXV21,DR11-W,DR11-B,DMP11,DPV11,ML11
ISB11,DMV11,UNA,UDA,RQDX1,DMF32,KMS11,PCL11,VS100
TU81,KMV11,KCT32,IEX,DHV11,DHU11,DMZ32,CPI32(ASYNCH)
CPI32(SYNCH),QNA,QVSS,VS31,LNV11

3. THIS IS A LIST OF DEVICES WHICH HAVE ONE FIXED AND THE REMAINDER FLOATING CSR'S.

RL11,RLV11,LPA11,RX11,RX211,RXV11,RXV21,UNA,UDA,RQDX1

TU81.QVSS

4. THIS IS A LIST OF DEVICES WHICH HAVE TWO FIXED AND THE
REMAINDER FLOATING CSR'S.
DR11-B

5. Changes made to make messages to operator easier to understand.
- LWL002 -

*

431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476

```
*****
????????????????????????????????????????????????????????
; * THE COMMENTS AND CODE BETWEEN THE *****/!!!!*****
; * IS DEPENDENT ON THE DEVICES. THIS CODE SHOULD BE CHECKED AND
; * MODIFIED WHEN ADDING OR REMOVING A DEVICE FOR THE
; * PROGRAM TO SUPPORT.
; *
; STEPS TO ADDING A DEVICE:
;
; 1. DETERMINE IF DEVICE IS A FLOATING CSR OR VA OR BOTH.
; IS THE 1ST ADDRESS FIXED? WHAT IS ITS RANK OF ORDER IN THE
; FLOATING SCHEME?
; 2. FOR FLOATING CSR ADDRESS DEVICES:
;     A. ENTER THE DEVICE IN QT2,DT2,MT2,ASCDEV,DEVSTR,
;     FATABL,Q??:, IN ITS RANK OF ORDER.
;     B. ADD OR MODIFY ROUTINES WITHIN THE PROGRAM FOR
;     SPECIAL CIRCUMSTANCES IN AREAS SUCH AS QUES,GFA,DONE.
; 3. FOR FLOATING VECTOR ADDRESS DEVICES:
;     A. ENTER THE DEVICE IN QT1,DT1,MT1,NAT,ADDRT,
;     VST,DEVSTR,Q??:, IN ITS RANK OF ORDER.
;     B. ADD OF MODIFY ROUTINES WITHIN THE PROGRAM FOR SPECIAL
;     CIRCUMSTANCES IN AREAS SUCH AS QUES,GFA,DONE.
;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
*****
```

000000
000001
000002
000003
000004
000005
000006
000007
000015
000200
000012
000126
000001
000002
000004
000010
000020
000040
000100

R0=#0
R1=#1
R2=#2
R3=#3
R4=#4
R5=#5
SP=#6
PC=#7
CR= 15
NOCRLF= 200
LF= 12
V= 126
BIT0= 1
BIT1= 2
BIT2= 4
BIT3= 10
BIT4= 20
BIT5= 40
BIT6= 100

;REGISTER DEFINITIONS

;BIT DEFINITIONS

;LWL002

DEFINITIONS

477	000200	BIT7=	200	
478	000400	BIT8=	400	
479	001000	BIT9=	1000	
480	002000	BIT10=	2000	
481	004000	BIT11=	4000	
482	010000	BIT12=	10000	
483	020000	BIT13=	20000	
484	040000	BIT14=	40000	
485	100000	BIT15=	100000	
486	104000	INPUT=	104000	; EMT'S- GET INPUT
487	104001	DATO=	104001	; DECIMAL ASCII TO OCTAL
488	104002	ERRCK=	104002	; ERROR CHECK
489	104003	TYPE=	104003	;
490	104004	OTOA=	104004	; OCTAL TO OCTAL ASCII
491	104005	SHIFT=	104005	; SPECIAL MULT & ADD
492	104006	OTDA=	104006	; OCTAL TO DECIMAL ASCII
493	104007	SHIFT1=	104007	; MULT BY 20 AND ADD GAP
494	104010	SHIFT2=	104010	; MULT BY 40 AND ADD GAP
495	104011	SHIFT3=	104011	; MULT BY 100 AND ADD GAP
496	104012	SHIFT4=	104012	; MULT BY 4 AND ADD GAP
497				
498				
499		.MACRO	ERCHK OK	; ERROR CHECKING MACRO
500		ERRCK		; GO CHECK FOR ERROR
501		BR	OK	; RETURN .+2 ON ERROR
502		.ENDM		
503				
504		.MACRO	PRINT ADR	; OUTPUT MACRO
505		MOV	ADR,RS	; ADDRESS TO RS
506		TYPE		; GO TYPE IT
507		.ENDM		
508				
509		.SBTTL	BEGIN	
510	000000	.=0		; TRAP CATCHERS
518	000030	.=30		
519	000030	.WORD	EMTHND	;
520	000042	.WORD	0	
521				
522	000042	.=42		;
523	000042	.WORD	0	; LOCATION FOR XXDP STACK POINTER ;LWL002
524				;LWL002
525	000200	.=200		; START AT 200
526	000200	JMP	STARTA	; ACTUALLY 1000
527	000204	JMP	STARTA	; RESTART AT 204...MUST GOTO START FIRST TIME THROUGH.
528				
529	000700	.=700		
530	000700	TRS:	177560	
531	000702	TRB:	177562	
532	000704	TPS:	177564	
533	000706	TPB:	177566	
534				
535		.LIST	MEB	

START

```

537 .SBTTL START
538 .=1000
539 001000 010637 000042 STARTA: MOV SP,#42 ;STORE THE XXDP STACK ADDRESS ;LWL002
540 001004 000005 START: RESET ;INIT THE WORLD
541 001006 012706 000700 MOV #TRS,SP ;SET THE STACK
542 001012 012737 001120 000206 MOV #RES,#206 ;SETUP RESTART ADDRESS
543 001020 012737 002260 000030 MOV #EMTHND,30 ;SETUP EMULATOR TRAP
544 001026 012705 015027 PRINT #HELLO ;IDENTIFY FLOAT ;LWL002
    001026 012705 015027 MOV #HELLO,R5 ;ADDRESS TO R5
    001032 104003 TYPE ;GO TYPE IT
545 001034 012705 015102 PRINT #HELP ;HOW TO GET HELP ;LWL002
    001034 012705 015102 MOV #HELP,R5 ;ADDRESS TO R5
    001040 104003 TYPE ;GO TYPE IT
546 001042 012705 015345 1$: PRINT #TERMQ ;ASK FOR TERMINAL TYPE
    001042 012705 015345 MOV #TERMQ,R5 ;ADDRESS TO R5
    001046 104003 TYPE ;GO TYPE IT
547 001050 012701 012144 MOV #TERMST,R1 ;FILL TABLE
548 001054 104000 INPUT ;GET IT
549 ;LWL0023$: CMPB (R1)+,(R0) ;WHICH ONE -A,L,V
550 001056 121110 3$: CMPB (R1),(R0) ;WHICH ONE -A,L,V ;LWL002
551 001060 001411 BEQ 4$ ;FOUND IT
552 ;LWL002 ADD #2,R1 ;SKIP OVER FILL CHAR. AND COUNT
553 001062 062701 000003 ADD #3,R1 ;SKIP OVER TYPE,FILL CHAR. AND COUNT ;LWL002
554 001066 020127 012155 CMP R1,#TERMST+11 ;NO - END OF TABLE
555 001072 001371 BNE 3$ ;NO MAYBE THE NEXT WILL MATCH
556 001074 012705 015556 PRINT #TERMQ2 ;BAD INPUT --TELL HIM WHAT WE WANT
    001074 012705 015556 MOV #TERMQ2,R5 ;ADDRESS TO R5
    001100 104003 TYPE ;GO TYPE IT
557 001102 000757 BR 1$ ;AND DO IT AGAIN
558 001104 112137 012277 4$: MOVB (R1)+,TERMT ;STORE TERMINAL TYPE ;LWL002
559 001110 112137 012140 MOVB (R1)+,CHAR ;SET FILL CHARACTER
560 001114 112137 006332 MOVB (R1)+,FILCNT ;SET FILL COUNT
561
562 ; THIS ROUTINE CLEARS OUT THE DEVICE DATA TABLE.
563 001120 000005 RES: RESET
564 001122 012706 001004 MOV #START,SP
565 001126 012700 000201 MOV #DEVCNT*2.+1,R0 ;# OF LOCATIONS TO CLEAR
566 001132 012701 010430 MOV #DC,R1 ;START OF DATA TABLE
567 001136 005021 1$: CLR (R1)+ ;CLEAR IT!
568 001140 005300 DEC R0 ;DECREMENT COUNT
569 001142 001375 BNE 1$ ;DO AGAIN IF NOT EQUAL
570 001144 005037 011740 CLR FLAG ;CLEAR PROGRAM CONDITION BITS
571 001150 005037 011742 CLR FLAG1 ;IF DM11-A AND DV11 WERE SELECTED PRINT DVMES
572 001154 005037 011744 CLR FLAG2 ;IF DV11 WERE DECREASED PRINTS DVMSS
573 001160 005037 011746 CLR FLAG3 ;IF DV11 WERE DECREASED TO ZERO PRINT DVNMS
574 001164 005037 011750 CLR FLAG4 ;IF PA611-P AND LPD11 WERE SELECTED PRINT LPDMES
575 001170 005037 011752 CLR FLAG5 ;IF LPD11 WERE DECREASED TO ZERO PRINT LPDMSS
576 001174 005037 011754 CLR FLAG6 ;IF PA611-R AND VTV01 WERE SELECTED PRINT VO1MES
577 001200 005037 011756 CLR FLAG7 ;IF VTV01 WAS DECREASED PRINT VO1MSS
578 001204 005037 011760 CLR FLAG8 ;USED FOR CONTROL A
579 001210 005037 011762 CLR FLAG9 ;USED FOR THE SELECTION OF DC11 AND DL11-A
580 001214 005037 011764 CLR FLAG10 ;IF LPD11 WAS DECREASED PRINTS LPDESS
581 001220 005037 011766 CLR FLAG11 ;USED IF PA611-R, PA611-P OR LPD11 WAS SELECTED
582 001224 005037 011770 CLR FLAG12 ;FOR PRINTING CSRVEC HEADER ;LWL002
583 ;LWL002 PRINT #HEAD ;TITLE-ASK OPTION
584 ; GET THE OPTION AND SETUP THE POINTERS FOR IT
585 ;

```

START

```

586 001230          OPTION: PRINT  #0P          ;ASK OPTION                      ;LWL002
    001230 012705 014612      MOV      #0P,R5      ;ADDRESS TO R5
    001234 104003          TYPE          ;GO TYPE IT
587 001236          OPTNA: PRINT  #0P2          ;SKIP FIRST LINE                      ;LWL002
    001236 012705 014643      MOV      #0P2,R5      ;ADDRESS TO R5
    001242 104003          TYPE          ;GO TYPE IT
588 001244 104000          INPUT          ;WHICH OPTIN?
589 001246 021027 040526      CMP      (R0),#"VA      ;VECTORS AND ADDRESSES
590 001252 001425          BEQ      6$          ;BR IF YES
591 001254 021027 040506      CMP      (R0),#"FA      ;FLOATING ADDR ONLY
592 001260 001434          BEQ      7$          ;BRANCH IF MATCH
593 001262 021027 042510      CMP      (R0),#"HE      ;HELP?                                ;LWL002
594 001266 001413          BEQ      5$          ;BR IF YES                            ;LWL002
595 001270 021027 054105      CMP      (R0),#"EX      ;EXIT?                                ;LWL002
596 001274 001404          BEQ      4$          ;BR IF YES                            ;LWL002
597 001276          PRINT  #TERMQ2          ;NO MATCH - TELL WHAT THEY ARE
    001276 012705 015556      MOV      #TERMQ2,R5      ;ADDRESS TO R5
    001302 104003          TYPE          ;GO T E IT
598 001304 000754          BR      OPTNA          ;AND TRY AGAIN                      ;LWL002
599
600          ; GET HELP OR EXIT
601          ;
602
603 001306 013706 000042      4$:      MOV      #42,SP      ;GET THE XXDP STACK ADDRESS          ;LWL002
604 001312 000207          RTS      PC          ;RETURN TO XXDP                      ;LWL002
605 001314 000000          HALT          ;SHOULDN'T GET HERE                  ;LWL002
606 001316          5$:      PRINT  #HELP          ;PRINT HELP MESSAGE                  ;LWL002
    001316 012705 015102      MOV      #HELP,R5      ;ADDRESS TO R5
    001322 104003          TYPE          ;GO TYPE IT
607 001324 000741          BR      OPTION          ;ASK OPTION AGAIN                      ;LWL002
608
609          ; SETUP POINTERS FOR THE 'VA' OPTION
610          ;
611
612 001326 042737 000200 011740 6$:      BIC      #BIT7,FLAG      ;DOING THE 'VA' OPTION
613 001334 012703 006362      MOV      #QT1,R3      ;R3<=Q TABLE !
614 001340 012702 006562      MOV      #DT1,R2      ;R2<=DATA TABLE 1
615 001344 012701 007204      MOV      #MT1,R1      ;R1<=MAXIMUM NUMBER TABLE 1
616 001350 000411          BR      8$          ;START THE QUESTIONS
617
618          ; SETUP POINTERS FOR THE 'FA' OPTION
619          ;
620 001352 052737 000200 011740 7$:      BIS      #BIT7,FLAG      ;'FA' OPTION
621 001360 012703 006762      MOV      #QT2,R3      ;TABLE 2
622 001364 012702 007072      MOV      #DT2,R2
623 001370 012701 007406      MOV      #MT2,R1
624
625
626          ; ASK FOR THE NUMBER OF DEVICES.
627
628          ;
629 001374 012737 000001 011760 8$:      MOV      #1,FLAG8      ;OK TO USE CONTROL A
630 001402          PRINT  #Q          ;START QUESTIONS HERE.
    001402 012705 012210      MOV      #Q,R5      ;ADDRESS TO R5
    001406 104003          TYPE          ;GO TYPE IT
631 001410          QUES: PRINT  (R3)          ;TYPE DEVICE NAME
    001410 011305          MOV      (R3),R5      ;ADDRESS TO R5

```

START

```

001412 104003
632 001414 012705 015024
001414 104003
001420 104003
633 001422 104000
634 001424 104001
635 001426 104002
001426 104002
001430 000401
636 001432 000766
637
638 001434 010032
639 001436 022123
640 001440 005711
641 001442 001362
642 001444 012705 014510
001444 104003
001450 104003
643 001452 012705 014510
001452 104003
001456 104003
644
645
646
647
648 001460 012737 000300 012030
649 001466 012737 160010 012034
650 001474 012700 007720
651 001500 012701 010432
652 001504 012702 000100
653 001510 012021
654 001512 005721
655 001514 005302
656 001516 001374
657 001520 012700 012034
658 001524 012702 007072
659 001530 012203
660 001532 001406
661 001534 011063 000002
662 001540 011301
663 001542 000240
664 001544 000137 001554
665 001550 000137 002346
666
667
668
669
670
671
672
673
674
675 001554 020327 010660
676 001560 001003
677 001562 005701
678 001564 001401
679 001566 005301

TYPE
PRINT #QMARK
MOV #QMARK,R5
TYPE
INPUT
DATO
ERCHK 2$
ERRCK
BR 2$
BR QUES
2$: MOV R0,8(R2)+
CMP (R1)+,(R3)+
TST (R1)
BNE QUES
PRINT: PRINT #CRLF
MOV #CRLF,R5
TYPE
PRINT #CRLF
MOV #CRLF,R5
TYPE
.SBTTL GET FIRST ADDRESS
; THIS ROUTINE LOADS THE STARTING ADDRESS INTO THE DEVICE DATA TABLE
; FOR ALL DEVICES ( 0 FOR FLOATING ADDRESS DEVICES ).
GFA: MOV #300,VECTOR
MOV #160010,ADDR
MOV #ADDR,R0
MOV #DC+2,R1
MOV #DEVcnt,R2
1$: MOV (R0)+,(R1)+
TST (R1)+
DEC R2
BNE 1$
MOV #ADDR,R0
MOV #DT2,R2
TAG3: MOV (R2)+,R3
BEQ 4$
MOV @R0,2(R3)
MOV @R3,R1
NOP
JMP NEXT
4$: JMP DONE

;GO TYPE IT
;APPEND WITH '?'
;ADDRESS TO R5
;GO TYPE IT
;HOW MANY?
;CONVERT TO OCTAL
;ANY ERRORS?
;GO CHECK FOR ERROR
;RETURN .+2 ON ERROR
;YES!! - ASK AGAIN!
;STORE ANSWER
;UPDATE POINTERS
;END OF LIST?
;THERES MORE
;END OF QUESTIONS
;ADDRESS TO R5
;GO TYPE IT
;
;ADDRESS TO R5
;GO TYPE IT
;FIRST FLOATING VECTOR STARTS AT 300
; " ADDRESSES START AT 160010
;R0<=ADDR. TABLE
;R1<=DEVICE ADDR.DATA TABLE
;R2<=NUMBER OF DEVICES
;SET-UP FIRST ADDR.IN DEV.DATA TABLE
;R1+2-2 WORDS/DEVICE
;ONE LESS TO DO
;DO IT AGAIN?
; CURRENT ADDRESS LOCATION
; FLOATING DEVICE TABLE POINTER
;DEVICE DATA POINTER
;BRANCH IF NO MORE
;STORE ADDRESS
;GET DEVICE COUNT
;BRANCH TO 2$ IF NO SPACES WANTED BETWEEN DEVICES.
;DONE CALCULATING FIRST ADDRESSES
;CHECK FOR SPECIAL DEVICES. RESR IS NOT USED BUT MUST HAVE A SPACE OF 10 FOR FA.
;RL11, RLV11, RX11, RX211, RXV11, RXV21, UNA, RQDX1,UDA, TU81 BOTH HAVE ONE
;STANDARD ADDRESS AND VECTOR. THE REST ARE FLOATING. THE LPA11 AND QVSS HAVE ONE
;STANDARD ADDRESS AND THE REST ARE FLOATING. THE DR11-B HAS A STANDARD ADDRESS
;AND STANDARD VECTOR FOR THE FIRST. THE NEXT HAS A STANDARD ADDRESS AND FLOATING
;VECTOR, THE REST ARE FLOATING ADDRESSES AND VECTOR.
NEXT: CMP R3,@RESR
BNE 40$
TST R1
BEQ 40$
DEC R1
;IS IT RESERVED
;BRANCH IF NOT
;CHECK IF RESERVED WAS SELECTED
;BRANCH IF NOT
;DEC RESERVED SO IT WILL LEAVE A GAP

```

GET FIRST ADDRESS

680	001570	020327	010634	40\$:	CMP	R3,#RL	;IS IT THE RL11
681	001574	001003			BNE	41\$	;BRANCH IF NOT
682	001576	005701			TST	R1	;CHECK IF ANY RL11 WERE SELECTED
683	001600	001401			BEQ	41\$	;BRANCH IF NOT
684	001602	005301			DEC	R1	;DEC FOR STANDARD ADDRESS AND VECTOR
685	001604	020327	010664	41\$:	CMP	R3,#RX	;IS IT THE RX11
686	001610	001003			BNE	42\$	;BRANCH IF NOT
687	001612	005701			TST	R1	;CHECK IF ANY RX11 WERE SELECTED
688	001614	001401			BEQ	42\$	;BRANCH IF NOT
689	001616	005301			DEC	R1	;DEC FOR STANDARD ADDRESS AND VECTOR
690	001620	020327	010644	42\$:	CMP	R3,#LPAK	;IS IT THE LPA11-K
691	001624	001003			BNE	43\$	;BRANCH IF NOT
692	001626	005701			TST	R1	;CHECK IF ANY LPA11-K WERE SELECTED
693	001630	001401			BEQ	43\$	;BRANCH IF NOT
694	001632	005301			DEC	R1	;DEC STANDARD ADDRESS
695	001634	020327	010674	43\$:	CMP	R3,#DRB	;IS IT THE DR11-B
696	001640	001006			BNE	44\$	;BRANCH IF NOT
697	001642	005701			TST	R1	;CHECK IF ANY DR11-B WERE SELECTED
698	001644	001404			BEQ	44\$	;BRANCH IF NOT
699	001646	005301			DEC	R1	;DEC FOR STANDARD ADDRESS AND VECTOR
700	001650	005701			TST	R1	;CHECK IF A SECOND DR11-B WAS SELECTED
701	001652	001401			BEQ	44\$	;BRANCH IF NOT
702	001654	005301			DEC	R1	;DEC FOR SECOND STANDARD ADDRESS
703	001656	020327	010724	44\$:	CMP	R3,#UNA	;IS IT THE DEUNA
704	001662	001003			BNE	UDA\$	;BRANCH IF NOT
705	001664	005701			TST	R1	;CHECK IF ANY DEUNA WERE SELECTED
706	001666	001401			BEQ	UDA\$	;BRANCH IF NOT
707	001670	005301			DEC	R1	;DEC FOR STANDARD ADDRESS AND VECTOR
708	001672	020327	010730	UDA\$:	CMP	R3,#UDA	;IS IT THE UDA
709	001676	001003			BNE	TU81\$	;BRANCH IF NOT
710	001700	005701			TST	R1	;CHECK IF ANY UDA WERE SELECTED
711	001702	001401			BEQ	TU81\$	;BRANCH IF NOT
712	001704	005301			DEC	R1	;DEC FOR STANDARD ADDRESS AND VECTOR
713	001706	020327	010754	TU81\$:	CMP	R3,#TU81	;IS IT THE TU81
714	001712	001003			BNE	QVSS\$	;BRANCH IF NOT
715	001714	005701			TST	R1	;CHECK IF ANY TU81 WERE SELECTED
716	001716	001401			BEQ	QVSS\$	;BRANCH IF NOT
717	001720	005301			DEC	R1	;DEC FOR STANDARD ADDRESS AND VECTOR
718	001722	020327	011014	QVSS\$:	CMP	R3,#QVSS	;IS IT THE QVSS
719	001726	001003			BNE	5\$	;BRANCH IF NOT
720	001730	005701			TST	R1	;CHECK IF ANY QVSS WERE SELECTED
721	001732	001401			BEQ	5\$	;BRANCH IF NOT
722	001734	005301			DEC	R1	;DEC FOR STANDARD ADDRESS
723							
724							
725							
726							
727							
728							
729	001736	020327	010520	5\$:	CMP	R3,#DH	
730	001742	001432			BEQ	TAG1	; DH11 - 20
731	001744	020327	010620		CMP	R3,#V31	
732	001750	001427			BEQ	TAG1	; VMV31- 20
733	001752	020327	010644		CMP	R3,#LPAK	
734	001756	001424			BEQ	TAG1	; LPAK - 20
735	001760	020327	010720		CMP	R3,#DMV	
736	001764	001421			BEQ	TAG1	; DMV - 20

;*****
 ;??
 ; DO A CHECK FOR THE DEVICES THAT DON'T ADDRESS-UP BY 10

GET FIRST ADDRESS

737	001766	020327	010734		CMP	R3,DMF	
738	001772	001425			BEQ	TAG2	; DMF32- 40
739	001774	020327	010730		CMP	R3,UDA	
740	002000	001410			BEQ	TU\$	; UDA50- 4
741	002002	020327	010740		CMP	R3,KMS	
742	002006	001410			BEQ	TAG1	; KMS11- 20
743	002010	000137	002102		JMP	TAG	
744					;!!		
745					;*****		
746							
747	002014	104005		TAG4:	SHIFT		;CALCULATE NEXT DEVICE ADDRESS
748	002016	000137	001530		JMP	TAG3	;PROCESS NEXT
749	002022	104012		TU\$:	SHIFT4		;CALCULATE DEVICE ADDRESS BY 4
750	002024	000137	001530		JMP	TAG3	;PROCESS NEXT
751	002030	004737	005764	TAG1:	JSR	PC,BITE	;MAKE ADDR. DIVISIBLE BY 20
752	002034	011063	000002		MOV	0R0,2(R3)	;STORE IT
753	002040	104007			SHIFT1		;20 BYTES/UNIT
754	002042	000137	001530		JMP	TAG3	
755	002046	004737	006004	TAG2:	JSR	PC,BITE1	;MAKE ADDR. DIVISIBLE BY 40
756	002052	011063	000002		MOV	0R0,2(R3)	;STORE IT
757	002056	104010			SHIFT2		;40 BYTES/UNIT
758	002060	000137	001530		JMP	TAG3	
759	002064	004737	006042	QVS\$:	JSR	PC,BITE2	;MAKE ADDR. DIVISIBLE BY 100
760	002070	011063	000002		MOV	0R0,2(R3)	;STORE IT
761	002074	104011			SHIFT3		;100 BYTES/UNIT
762	002076	000137	001530		JMP	TAG3	
763							
764							
765	002102	020327	010750	TAG:	CMP	R3,VS100	
766	002106	001750			BEQ	TAG1	; VS100- 20
767	002110	020327	010754		CMP	R3,TU81	
768	002114	001742			BEQ	TU\$	; TU81 - 4
769	002116	020327	010760		CMP	R3,KMV	
770	002122	001742			BEQ	TAG1	; KMV11- 20
771	002124	020327	010774		CMP	R3,DHV	
772	002130	001737			BEQ	TAG1	; DHV11- 20
773	002132	020327	011000		CMP	R3,DMZ32	
774	002136	001743			BEQ	TAG2	; DMZ32- 40
775	002140	020327	011004		CMP	R3,CPI32	
776	002144	001740			BEQ	TAG2	; CPI32- 40
777	002146	020327	011014		CMP	R3,QVSS	
778	002152	001744			BEQ	QVS\$	; QVSS 200
779	002154	000137	002014		JMP	TAG4	
780							
781	002160	006301		SSHIF3:	ASL	R1	;MULT. BY 100 OCTAL
782	002162	006301		SSHIF2:	ASL	R1	;MULT. BY 40 OCTAL
783	002164	006301		SSHIF1:	ASL	R1	;MULT. BY 20 OCTAL
784	002166	006301		SSHIFT:	ASL	R1	;MULT. BY 10 OCTAL
785	002170	006301		SSHIF4:	ASL	R1	;MULT. BY 4 OCTAL
786	002172	006301			ASL	R1	
787	002174	020327	010724		CMP	R3,UNA	;IF UNA LEAVE GAP OF 4
788	002200	001423			BEQ	GAP	;FOR THE UDA
789	002202	020327	010750		CMP	R3,VS100	;IF VS100 LEAVE GAP OF 4
790	002206	001420			BEQ	GAP	;FOR THE TU81
791	002210	020327	010730		CMP	R3,UDA	;IF UDA CHECK FOR BIT 2
792	002214	001003			BNE	GAP3	;BRANCH IF NOT
793	002216	032701	000004		BIT	04,R1	;CHECK IF EVEN

SEQ 0015

GET FIRST ADDRESS

[illegible]

COMPUTE THE ADDR., JUMPERS AND VECTORS

```

851      ; SET THE DEVICE ADDRESS TO A MINUS ONE IF IT FOLLOWS THE DJ AND
852      ; IS NOT A FLOATING ADDRESS DEVICE.  FOR THOSE DEVICES THAT ARE A COMBINATION,
853      ; SPECIAL HANDLING IS REQUIRED TO LIST THE STANDARD CSR ADDRESS
854      ; AND VECTOR ADDRESS AND THEN LIST THE FLOATING RANGE FOR EACH.
855
856
857 002364 012701 177777      MOV      @-1,R1
858 002370 010137 010526      MOV      R1,GT+2      ;IF ADDRESS=-1 THEN
859 002374 010137 010532      MOV      R1,VSV+2      ;THE DEVICE IS SKIPPED
860 002400 010137 010536      MOV      R1,LPS+2      ;GT40,VSV11,LPS11,KW11-W,DV11,DWUN,VTVO1
861 002404 010137 010546      MOV      R1,KWW+2      ;TS11,IP11,ML11,PCL11,KCT32,IEX,QNA,LNV11
862 002410 010137 010562      MOV      R1,DV+2      ;HAVE STANDARD ADDRESS
863 002414 010137 010572      MOV      R1,DWUN+2
864 002420 010137 010626      MOV      R1,V01+2
865 002424 010137 010642      MOV      R1,TS11+2
866 002430 010137 010652      MOV      R1,IP+2
867 002434 010137 010712      MOV      R1,ML+2
868 002440 010137 010746      MOV      R1,PCL+2
869 002444 010137 010766      MOV      R1,KCT32+2
870 002450 010137 010772      MOV      R1,IEX+2
871 002454 010137 011012      MOV      R1,QNA+2
872 002460 010137 011026      MOV      R1,LNV+2
873
874      ;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
875      ;*****
876
877 002464 012701 010514      MOV      @DJ,R1      ;DJ IS FIRST FOR 'FA' OPTION
878 002470 012700 000032      MOV      @DJ-DC/2,R0      ;SET OFFSET INTO 'N'EXT 'A'DDR.'T'ABLE
879 002474 000405              BR      DONE1      ;SET TO PROCEED!!
880 002476 012702 006362      1$: MOV      @QT1,R2      ;USE THE SAME TABLE AS FOR THE QUESTIONS
881 002502 012701 010430      MOV      @DC,R1      ;R1<=START OF DATA TABLE
882 002506 005000              CLR      R0      ;0 OFFSET INTO NAT
883 002510 005721      DONE1: TST      (R1)+      ;DO WE HAVE ANY?
884 002512 001031              BNE     2$      ;BR IF YES
885 002514 022022              CMP      (R0)+,(R2)+      ;INC POINTERS
886 002516 022127 177777      CMP      (R1)+,@-1      ;DO WE SKIP THIS ONE?
887 002522 001372              BNE     DONE1      ;NO (TWO -1'S TOGETHER IS END OF LIST)
888 002524 005742              TST      -(R2)      ;BACKUP FOR THE 'FA' OPTION
889 002526 021127 177777      CMP      (R1),@-1      ;ANOTHER -1 IS THE END!!!!
890 002532 001366              BNE     DONE1      ;BRANCH IF SOME MORE!
891 002534 005737 011752      TST      FLAG5      ;CHECK TO SEE IF LPD11 WERE DECREASED TO ZERO
892 002540 001403              BEQ     37$      ;BRANCH IF NOT
893 002542              PRINT     @LPDMSS      ;PRINT THE MESSAGE
894      002542 012705 017753      MOV      @LPDMSS,R5      ;ADDRESS TO R5
895      002546 104003              TYPE     ;GO TYPE IT
896 002550 005737 011746      37$: TST      FLAG3      ;CHECK TO SEE IF THE DV11 WERE DECREASED TO ZERO
897 002554 001403              BEQ     38$      ;BRANCH IF NOT
898 002556              PRINT     @DVNMS      ;PRINT MESSAGE
899      002556 012705 020151      MOV      @DVNMS,R5      ;ADDRESS TO R5
900      002562 104003              TYPE     ;GO TYPE IT
901 002564              38$: PRINT     @END      ; ***** THE END *****
902      002564 012705 014503      MOV      @END,R5      ;ADDRESS TO R5
903      002570 104003              TYPE     ;GO TYPE IT
904 002572 000137 001120      JMP      RES      ;RESTART-(ACT-11 PATCH SHOULD GO HERE!)
905
906      ;*****

```

COMPUTE THE ADDR., JUMPERS AND VECTORS

```

902      ;????????????????????????????????????????????????????????????
903      ; SPECIAL DEVICE DEPENDENT ROUTINES FOR ADDRESS.
904
905      ;* CHECK IF DC11 AND DL11-A WAS SELECTED
906      ;
907 002576 020127 010432      2$:    CMP      R1,#DC+2      ;CHECK IF IT IS THE DC11
908 002602 001011              BNE      9$              ;BRANCH IF NOT
909 002604 005737 010430      TST      DC              ;CHECK IF DC11 WERE SELECTED
910 002610 001406              BEQ      9$              ;BRANCH IF NOT
911 002612 005737 010434      TST      DL              ;CHECK IF DL11-A WERE SELECTED
912 002616 001403              BEQ      9$              ;BRANCH IF NOT
913 002620 012737 000001 011762      MOV      #1,FLAG9      ;SET FLAG FOR DC11 AND DL11-A
914
915      ;* DV11 ADDRESS DEPENDS ON DM11-A ADDRESS
916      ;
917 002626 020127 010446      9$:    CMP      R1,#DMA+2      ;CHECK IF IT IS THE DM11-A?
918 002632 001105              BNE      17$             ;BR IF NOT
919 002634 005737 010444      TST      DMA              ;CHECK IF DM11-A WERE SELECTED
920 002640 001502              BEQ      17$             ;BRANCH IF NOT
921 002642 005737 010560      TST      DV              ;CHECK IF DV11 WERE SELECTED
922 002646 001477              BEQ      17$             ;BRANCH IF NOT
923 002650 012737 000001 011742      MOV      #1,FLAG1      ;SET FLAG THAT DM11-A AND DV11 WERE SELECTED
924 002656 013737 010444 012044      MOV      DMA,SAVE3      ;GET THE NUMBER OF DM11-A SELECTED
925 002664 006337 012044      ASL      SAVE3              ;FORM NEW ADDRESS FOR DV11
926 002670 006337 012044      ASL      SAVE3
927 002674 006337 012044      ASL      SAVE3
928 002700 032737 000010 012044      BIT      #10,SAVE3      ;CHECK IF BIT THREE IS SET
929 002706 001403              BEQ      33$             ;BRANCH IF NOT
930 002710 062737 000010 012044      ADD      #10,SAVE3      ;ADD 10 TO MAKE IT EVEN
931 002716 032737 000020 012044 33$:  BIT      #20,SAVE3      ;CHECK IF BIT FOUR IS SET
932 002724 001406              BEQ      34$             ;BRANCH IF NOT
933 002726 062737 000020 012044      ADD      #20,SAVE3      ;ADD 20 TO BE ON RIGHT BOUNDARY AND SAVE
934 002734 063737 012044 010562      ADD      SAVE3,DV+2      ;STORE NEW ADDRESS
935 002742 013737 010560 012046 34$:  MOV      DV,SAVE4      ;GET NUMBER OF DV11 SELECTED
936 002750 005337 012046      DEC      SAVE4              ;DEC BECAUSE SAVE3 HAS FIRST DV11 ADDRESS
937 002754 006337 012046      ASL      SAVE4              ;FORM ADDRESS OF LAST DV11
938 002760 006337 012046      ASL      SAVE4
939 002764 006337 012046      ASL      SAVE4
940 002770 006337 012046      ASL      SAVE4
941 002774 006337 012046      ASL      SAVE4
942 003000 063737 012044 012046      ADD      SAVE3,SAVE4      ;ADD FIRST ADDRESS TO FORM LAST ADDRESS
943 003006 022737 000140 012046      CMP      #140,SAVE4      ;CHECK TO SEE IF IT IS TO HIGH
944 003014 103014              BHIS     17$             ;BRANCH IF NOT
945 003016 012737 000001 011744      MOV      #1,FLAG2      ;SET FLAG THAT DV11 HAVE BEEN DECREASED
946 003024 005337 010560      DEC      DV              ;DECREMENT NUMBER THAT WAS SELECTED
947 003030 005737 010560      TST      DV              ;CHECK TO SEE IF ANY LEFT
948 003034 001401              BEQ      35$             ;BRANCH IF NOT
949 003036 000741              BR       34$             ;GO BACK AND TRY AGAIN
950 003040 012737 000001 011746 35$:  MOV      #1,FLAG3      ;SET FLAG THAT DV11 WERE DECREASED TO ZERO
951
952      ;
953      ;* VTV01 ADDRESS IS AFTER THE PA6-11R'S
954      ;
955 003046 020127 010466      17$:  CMP      R1,#PAR+2      ;CHECK IF THE PA6-11R
956 003052 001007              BNE      47$             ;BRANCH IF NOT
957 003054 005737 010464      TST      PAR              ;CHECK IF ANY PA6-11R WERE SELECTED
958 003060 001404              BEQ      47$             ;BRANCH IF NOT

```

COMPUTE THE ADDR., JUMPERS AND VECTORS

959	003062	012737	000001	011766	MOV	#1,FLAG11	;SET FLAG FOR PAR SELECTION
960	003070	000431			BR	49\$	;BRANCH TO FORM VTV01 ADDRESS
961	003072	020127	010472		47\$: CMP	R1,#PAP+2	;CHECK IF THE PA611-P
962	003076	001012			BNE	48\$	;BRANCH IF NOT
963	003100	005737	011766		TST	FLAG11	;CHECK IF VTV01 ADDRESS HAS BEEN FORMED
964	003104	001050			BNE	45\$	;BRANCH IF YES
965	003106	005737	010470		TST	PAP	;CHECK IF ANY PA611-P WERE SELECTED
966	003112	001404			BEQ	48\$	;BRANCH IF NO
967	003114	012737	000001	011766	MOV	#1,FLAG11	;SET FLAG FOR PAP SELECTION
968	003122	000414			BR	49\$	;BRANCH TO FORM VTV01 ADDRESS
969	003124	020127	010476		48\$: CMP	R1,#LPD+2	;CHECK IF LPD11
970	003130	001036			BNE	45\$	;BRANCH IF NOT
971	003132	005737	011766		TST	FLAG11	;CHECK IF VTV01 ADDRESS HAS BEEN FORMED
972	003136	001033			BNE	45\$	;BRANCH IF YES
973	003140	005737	010474		TST	LPD	;CHECK IF LPD11 WERE SELECTED
974	003144	001430			BEQ	45\$	;BRANCH IF NO
975	003146	012737	000001	011766	MOV	#1,FLAG11	;SET FLAG FOR LPD11 SELECTION
976	003154	005737	010624		49\$: TST	V01	;CHECK IF ANY VTV01 WERE SELECTED
977	003160	001422			BEQ	45\$	;BRANCH IF NOT
978	003162	005737	011766		TST	FLAG11	;CHECK IF PAR, PAP OR LPD WERE SELECTED
979	003166	001417			BEQ	45\$	;BRANCH IF NOT
980	003170	012737	000001	011754	MOV	#1,FLAG6	;SET FLAG THAT PA6-11R WERE SELECTED
981	003176	062737	000160	010626	ADD	#160,V01+2	;FORM NEW ADDRESS FOR VTV01
982	003204	022737	000002	010624	CMP	#2,V01	;CHECK FOR TWO VTV01
983	003212	001005			BNE	45\$	;BRANCH IF NOT
984	003214	012737	000001	011756	MOV	#1,FLAG7	;SET FLAG THAT VTV01S WERE DECREASED
985	003222	005337	010624		DEC	V01	;DECREASE VTV01S
986					:		
987					;*	LPD11 ADDRESS IS AFTER THE PA6-11P'S	
988					:		
989	003226	020127	010472		45\$: CMP	R1,#PAP+2	;CHECK IF THE PA611 P
990	003232	001063			BNE	10\$	;BRANCH IF NOT
991	003234	005737	010470		TST	PAP	;CHECK IF ANY PA611-P WERE SELECTED
992	003240	001460			BEQ	10\$	;BRANCH IF NOT
993	003242	005737	010474		TST	LPD	;CHECK IF ANY LPD11 WERE SELECTED
994	003246	001455			BEQ	10\$	;BRANCH IF NOT
995	003250	012737	000001	011750	MOV	#1,FLAG4	;SET FLAG THAT PA611-P AND LPD11 WERE SELECTED
996	003256	013737	010470	012050	MOV	PAP,SAVE5	;GET THE NUMBER OF PA611-P SELECTED
997	003264	006337	012050		ASL	SAVE5	;FORM ADDRESS OF LPD11
998	003270	006337	012050		ASL	SAVE5	
999	003274	006337	012050		ASL	SAVE5	
1000	003300	063737	012050	010476	ADD	SAVE5,LPD+2	;ADD ADDRESS TO TABLE
1001	003306	013737	010474	012052	29\$: MOV	LPD,SAVE6	;GET NUMBER OF LPD11
1002	003314	005337	012052		DEC	SAVE6	;DECREMENT BECAUSE LPD+2 HAS FIRST ADDRESS
1003	003320	006337	012052		ASL	SAVE6	;FORM NEW ADDRESS
1004	003324	006337	012052		ASL	SAVE6	
1005	003330	006337	012052		ASL	SAVE6	
1006	003334	063737	012050	012052	ADD	SAVE5,SAVE6	;ADD FIRST ADDRESS TO FORM LAST ADDRESS
1007	003342	022737	000070	012052	CMP	#70,SAVE6	;CHECK TO SEE IF IT IS TOO HIGH
1008	003350	103014			BHIS	10\$	;BRANCH IF NOT
1009	003352	012737	000001	011764	MOV	#1,FLAG10	;SET FLAG THAT LPD11 WERE DECREASED TO ZERO
1010	003360	005337	010474		DEC	LPD	;DECREASE LPD
1011	003364	005737	010474		TST	LPD	;CHECK TO SEE IF ANY ARE LEFT
1012	003370	001401			BEQ	36\$	;BRANCH IF NOT
1013	003372	000745			BR	29\$	;GO BACK AND TRY AGAIN
1014	003374	012737	000001	011752	36\$: MOV	#1,FLAG5	;LPD11 WERE DECREASED TO ZERO
1015							

COMPUTE THE ADDR., JUMPERS AND VECTORS

```

1016 ;*****
1017 ;????????????????????????????????????????????????????????????
1018 ; CHECK HERE FOR A DEVICE THAT NEEDS A COMMENT TYPED.
1019 ; DURING THE PRINT ROUTINE, THE DL11,LPD11,DV11,VTVO1,RL11,TS11,LPA11,IP11
1020 ; RESER,RX11,DR11-B,ML11,UNA,UDA,KMS11,TU81,QVSS
1021 ; WILL PRINT OUT STANDARD CSR AND VECTOR ADDRESSES
1022 ; AND THEN LIST THE FLOATING VALUES FOR
1023 ; ADDITIONAL DEVICES REQUESTED BY OPERATOR.
1024
1025 003402 105737 011770 10$: TSTB FLAG12 ;HAVE WE BEEN THROUGH ONCE? ;LWL002
1026 003406 100406 BMI 4$ ;IF YES, BRANCH ;LWL002
1027 003410 PRINT #CSRVEC ;'CSR VECTOR 'HEADER ;LWL002
1028 003410 012705 015746 MOV #CSRVEC,R5 ;ADDRESS TO R5
1029 003414 104003 TYPE ;GO TYPE IT
1030 003416 052737 000200 011770 BIS #BIT7,FLAG12 ;WE'VE BEEN THRU HERE ;LWL002
1031 003424 4$: PRINT (R2)+ ;TYPE DEVICE NAME
1032 003424 012205 MOV (R2)+,R5 ;ADDRESS TO R5
1033 003426 104003 TYPE ;GO TYPE IT
1034 003430 020127 010436 CMP R1,#DL+2 ;IS IT THE DC11
1035 003434 001006 BNE 39$ ;BRANCH IF NOT
1036 003436 005737 011762 TST FLAG9 ;CHECK IF DC11 AND DL11-A WERE SELECTED
1037 003442 001403 BEQ 39$ ;BRANCH IF NOT
1038 003444 PRINT #DLMESS ;PRINT MESSAGE
1039 003444 012705 020772 MOV #DLMESS,R5 ;ADDRESS TO R5
1040 003450 104003 TYPE ;GO TYPE IT
1041 003452 020127 010476 39$: CMP R1,#LPD+2 ;IS IT THE LPD11
1042 003456 001022 BNE 26$ ;BRANCH IF NOT
1043 003460 005737 011750 TST FLAG4 ;CHECK IF PA611-P AND LPD11 WERE SELECTED
1044 003464 001403 BEQ 50$ ;BRANCH IF NOT
1045 003466 PRINT #LPDMES ;PRINT MESSAGE
1046 003466 012705 017335 MOV #LPDMES,R5 ;ADDRESS TO R5
1047 003472 104003 TYPE ;GO TYPE IT
1048 003474 005737 011764 50$: TST FLAG10 ;HAS THE LPD11 BEEN DECREASED
1049 003500 001403 BEQ 51$ ;BRANCH IF NOT
1050 003502 PRINT #LPDESS ;PRINT MESSAGE
1051 003502 012705 017542 MOV #LPDESS,R5 ;ADDRESS TO R5
1052 003506 104003 TYPE ;GO TYPE IT
1053 003510 005737 011752 51$: TST FLAG5 ;HAS THE LPD11 BEEN DECREASED TO ZERO
1054 003514 001403 BEQ 26$ ;BRANCH IF NOT
1055 003516 PRINT #LPDMES ;PRINT MESSAGE
1056 003516 012705 017335 MOV #LPDMES,R5 ;ADDRESS TO R5
1057 003522 104003 TYPE ;GO TYPE IT
1058 003524 020127 010562 26$: CMP R1,#DV+2 ;IS IT THE DV11?
1059 003530 001021 BNE 11$ ;BRANCH IF NOT
1060 003532 PRINT #DVMESS ;PRINT DV11 MESSAGE
1061 003532 012705 016121 MOV #DVMESS,R5 ;ADDRESS TO R5
1062 003536 104003 TYPE ;GO TYPE IT
1063 003540 005737 011742 TST FLAG1 ;CHECK IF DM11-A AND DV11 WERE SELECTED
1064 003544 001403 BEQ 11$ ;BRANCH IF NOT
1065 003546 PRINT #DVMES ;PRINT MESSAGE THAT THEY WERE
1066 003546 012705 016716 MOV #DVMES,R5 ;ADDRESS TO R5
1067 003552 104003 TYPE ;GO TYPE IT
1068 003554 005737 011744 11$: TST FLAG2 ;CHECK IF DV11 WERE DECREASED TO ZERO
1069 003560 001403 BEQ 20$ ;BRANCH IF NOT
1070 003562 PRINT #DVMSS ;PRINT MESSAGE THAT THEY WERE
1071 003562 012705 017121 MOV #DVMSS,R5 ;ADDRESS TO R5
1072 003566 104003 TYPE ;GO TYPE IT

```

COMPUTE THE ADDR., JUMPERS AND VECTORS

1055	003570	000137	004332	20\$:	JMP	PRI	;BACK TO ROUTINE
1056	003574	020127	010626	13\$:	CMP	R1,#V01+2	;IS IT THE VTV01
1057	003600	001014			BNE	46\$	;BRANCH IF NOT
1058	003602	005737	011754		TST	FLAG6	;CHECK IF BOTH PA6-11RS AND VTV01 WERE SELECTED
1059	003606	001411			BEQ	46\$	;BRANCH IF NOT
1060	003610				PRINT	#V01MES	;PRINT MESSAGE
	003610	012705	020345		MOV	#V01MES,R5	;ADDRESS TO R5
	003614	104003			TYPE		;GO TYPE IT
1061	003616	005737	011756		TST	FLAG7	;CHECK IF THE VTV01 WAS DECREASED
1062	003622	001403			BEQ	46\$	;BRANCH IF NOT
1063	003624				PRINT	#V01MSS	;PRINT MESSAGE
	003624	012705	020561		MOV	#V01MSS,R5	;ADDRESS TO R5
	003630	104003			TYPE		;GO TYPE IT
1064	003632	020127	010636	46\$:	CMP	R1,#RL+2	;IS IT THE RL?
1065	003636	001007			BNE	14\$	;IF NOT, BRANCH
1066	003640	005337	010634		DEC	RL	;DECREMENT DEVICE COUNT
1067	003644	012705	016201		MOV	#RLMESS, R5	;THEN PRINT STANDARD CSR
1068	003650	104003			TYPE		;GO TYPE IT
1069	003652	000137	004310		JMP	CLEAN	;BACK TO ROUTINE
1070	003656	020127	010642	14\$:	CMP	R1,#TS11+2	;IS IT THE TS11?
1071	003662	001007			BNE	15\$	;IF NOT, BRANCH
1072	003664	005337	010640		DEC	TS11	;DECREMENT DEVICE COUNT
1073	003670	012705	016244		MOV	#TSMESS,R5	;THEN PRINT STANDARD CSR
1074	003674	104003			TYPE		;GO TYPE IT
1075	003676	000137	004310		JMP	CLEAN	;BACK TO ROUTINE
1076	003702	020127	010646	15\$:	CMP	R1,#LPAK+2	;IS IT THE LPAK?
1077	003706	001022			BNE	18\$	;IF NOT, BRANCH
1078	003710	105737	011740		TSTB	FLAG	;CHECK IF IN FA OR VA
1079	003714	100410			BMI	19\$	;BRANCH IF IN FA
1080	003716	013737	010646	012040	MOV	LPAK+2,SAVE1	;SAVE FLOATING ADDRESS
1081	003724	012737	170460	010646	MOV	#170460,LPAK+2	;REPLACE FLOATING ADDRESS WITH STANDARD
1082	003732	000137	004332		JMP	PRI	;BRANCH FOR VECTOR
1083	003736	005337	010644	19\$:	DEC	LPAK	;DECREMENT DEVICE COUNT
1084	003742	012705	016307		MOV	#LPKMES,R5	;THEN PRINT STANDARD CSR
1085	003746	104003			TYPE		;GO TYPE IT
1086	003750	000137	004310		JMP	CLEAN	;BACK TO ROUTINE
1087	003754	020127	010652	18\$:	CMP	R1,#IP+2	;IS IT THE IP11
1088	003760	001007			BNE	21\$	;IF NOT, BRANCH
1089	003762	005337	010650		DEC	IP	;DECREMENT DEVICE COUNT
1090	003766	012705	016334		MOV	#IPMESS,R5	;PRINT STANDARD CSR
1091	003772	104003			TYPE		;GO TYPE IT
1092	003774	000137	004310		JMP	CLEAN	;CONTINUE TEST
1093	004000	020127	010662	21\$:	CMP	R1,#RESR+2	;IS IT THE RESR
1094	004004	001007			BNE	22\$	;IF NOT BRANCH
1095	004006	005337	010660		DEC	RESR	;DECREMENT COUNT
1096	004012	012705	016377		MOV	#REMES,R5	;PRINT NOT USED
1097	004016	104003			TYPE		;GO TYPE IT
1098	004020	000137	004310		JMP	CLEAN	;COUNTINE TEST
1099	004024	020127	010666	22\$:	CMP	R1,#RX+2	;IS IT THE RX11
1100	004030	001007			BNE	23\$	;BRANCH IF NOT
1101	004032	005337	010664		DEC	RX	;DECREMENT DEVICE COUNT
1102	004036	012705	016412		MOV	#RXMES,R5	;PRINT STANDARD CSR
1103	004042	104003			TYPE		;GO TYPE IT
1104	004044	000137	004310		JMP	CLEAN	;CONTINUE TEST
1105	004050	020127	010676	23\$:	CMP	R1,#DRB+2	;IS IT THE DRB
1106	004054	001015			BNE	24\$	;BRANCH IF NOT
1107	004056	005337	010674		DEC	DRB	;DECREMENT DEVICE COUNT

COMPUTE THE ADDR., JUMPERS AND VECTORS

```

1108 004062 013737 010676 012042      MOV      DRB+2,SAVE2      ;SAVE FLOATING ADDRESS
1109 004070 012737 172430 010676      MOV      #172430,DRB+2  ;REPLACE FLOATING ADDRESS WITH STANDARD
1110 004076 012705 016455              MOV      #DRBMES,R5      ;PRINT STANDARD CSR
1111 004102 104003              TYPE              ;GO TYPE IT
1112 004104 000137 004310              JMP      CLEAN          ;CONTINUE TEST
1113 004110 020127 010712      24$:      CMP      R1,#ML+2      ;IS IT THE ML11
1114 004114 001002              BNE      25$            ;BRANCH IF NOT
1115 004116 000137 004332              JMP      PRI            ;GET VECTOR
1116 004122 020127 010726      25$:      CMP      R1,#UNA+2      ;IS IT THE DEUNA
1117 004126 001007              BNE      UNA$          ;IF NOT BRANCH
1118 004130 005337 010724              DEC      UNA          ;DECREMENT DEVICE COUNT
1119 004134 012705 016520              MOV      #UNAMES,R5    ;THEN PRINT STANDARD CSR
1120 004140 104003              TYPE              ;GO TYPE IT
1121 004142 000137 004310              JMP      CLEAN          ;BACK TO ROUTINE
1122 004146 020127 010732      UNA$:      CMP      R1,#UDA+2      ;IS IT THE UDA
1123 004152 001007              BNE      52$            ;BRANCH IF NOT
1124 004154 005337 010730              DEC      UDA          ;DECREMENT DEVICE COUNT
1125 004160 012705 016563              MOV      #UDAMES,R5    ;PRINT STANDARD CSR AND VECTOR
1126 004164 104003              TYPE              ;GO TYPE IT
1127 004166 000137 004310              JMP      CLEAN          ;CONTINUE TEST
1128 004172 020127 010742      52$:      CMP      R1,#KMS+2      ;IS IT THE KMS11
1129 004176 001005              BNE      27$            ;BRANCH IF NOT
1130 004200 012705 021334              MOV      #KMSMES,R5    ;PRINT MODEM MESS
1131 004204 104003              TYPE              ;GO TYPE IT
1132 004206 000137 004310              JMP      CLEAN          ;CONTINUE TEST
1133 004212 020127 010756      27$:      CMP      R1,#TU81+2      ;IS IT THE TU81
1134 004216 001007              BNE      QV$           ;BRANCH IF NOT
1135 004220 005337 010754              DEC      TU81          ;DECREMENT DEVICE COUNT
1136 004224 012705 016626              MOV      #TU81MS,R5    ;PRINT STANDARD CSR AND VECTOR
1137 004230 104003              TYPE              ;GO TYPE IT
1138 004232 000137 004310              JMP      CLEAN          ;BACK TO ROUTINE
1139 004236 020127 011016      QV$:      CMP      R1,#QVSS+2      ;IS IT THE QVSS
1140 004242 001007              BNE      DR$           ;BRANCH IF NOT
1141 004244 005337 011014              DEC      QVSS          ;DECREMENT DEVICE COUNT
1142 004250 012705 016671              MOV      #QVSSMS,R5    ;PRINT STANDARD CSR
1143 004254 104003              TYPE              ;GO TYPE IT
1144 004256 000137 004310              JMP      CLEAN          ;BACK TO ROUTINE
1145 004262 020127 010462      DR$:      CMP      R1,#DR+2        ;IS IT THE DR11?
1146 004266 001021              BNE      PRI            ;BRANCH IF NOT
1147 004270 013705 010460              MOV      DR,R5         ;# OF DR11
1148 004274 006305              ASL      R5             ;MULTIPLY BY 10
1149 004276 006305              ASL      R5
1150 004300 006305              ASL      R5
1151 004302 005405              NEG      R5             ;SET UP TO SUBTRACT
1152 004304 000137 004332              JMP      PRI            ;GET OVER CLEAN UP ROUTINE
1153 004310 005761 177776      CLEAN:    TST      -(R1)        ;ARE THERE ANY LEFT?
1154 004314 001006              BNE      PRI            ;YES, THEN BRANCH
1155 004316              PRINT      #CRLF          ;CR/LF
1156 004324 005741              MOV      #CRLF,R5       ;ADDRESS TO R5
1157 004326 000137 004426              TYPE              ;GO TYPE IT
1158              TST      -(R1)        ;ADJUST POINTER
1159              JMP      PRIN        ;CLEAN UP
1160
1161
1162 004332      PRI:      PRINT      #CRLF          ;CR/LF

```

COMPUTE THE ADDR., JUMPERS AND VECTORS

```

004332 012705 014510      MOV    #CRLF,R5      ;ADDRESS TO R5
004336 104003              TYPE                    ;GO TYPE IT
1163 004340              PRINT   #TAB              ;PRINT TAB TO OFFSET ADDRESSES      ;LWL002
004340 012705 016105      MOV    #TAB,R5          ;ADDRESS TO R5
004344 104003              TYPE                    ;GO TYPE IT
1164 004346 104004      OTOA                      ;CHANGE ADDRESS TO ASCII
1165              ;LWL002 TSTB   FLAG              ;WHICH OPTIN?
1166              ;LWL002 BMI    6$                ;BR FOR 'FA'
1167 004350 004737 004444 4$: JSR    PC,VECT        ;GO GET THE VECTOR
1168              ;LWL002 BR     7$                ;GO TYPE IT
1169              ;LWL0026$: JSR    PC,JUMPER        ;GET JUMPER OR SWITCH SETTING
1170 004354              7$: PRINT   #A            ;TYPE THE ADDRESS AND VECTOR/JUMPERS
004354 012705 012054      MOV    #A,R5            ;ADDRESS TO R5
004360 104003              TYPE                    ;GO TYPE IT
1171 004362 005341      DEC     -(R1)              ;ANY MORE OF THE SAME?
1172 004364 001420      BEQ     PRIN              ;BRANCH IF NO MORE
1173 004366 005721      TST     (R1)+              ;POINT TO ADDR OF DEVICE
1174 004370 022711 170460 CMP     #170460,(R1)      ;CHECK TO SEE IF ITS THE LPA11-K
1175 004374 001003      BNE     30$                ;BRANCH IF NOT
1176 004376 013711 012040 MOV     SAVE1,(R1)        ;RESTORE FLOATING ADDRESS
1177 004402 000410      BR      32$                ;BRANCH TO CONVERT
1178 004404 022711 172430 30$: CMP     #172430,(R1) ;CHECK TO SEE IF ITS THE DR11-B
1179 004410 001003      BNE     31$                ;BRANCH IF NOT
1180 004412 013711 012042 MOV     SAVE2,(R1)        ;RESTORE FLOATING ADDRESS
1181 004416 000402      BR      32$                ;BRANCH TO CONVERT
1182 004420 066011 007520 31$: ADD     NAT(R0),(R1)   ;ADD GAP BETWEEN SAME DEVICES
1183 004424 000745      32$: BR      5$            ;GO CONVERT
1184 004426              PRIN: PRINT   #CRLF        ;NEXT DEVICE
004426 012705 014510      MOV    #CRLF,R5          ;ADDRESS TO R5
004432 104003              TYPE                    ;GO TYPE IT
1185 004434 005721      TST     (R1)+              ;ADJUST POINTER
1186
1187
1188      ;* END OF CHECKS
1189
1190
1191
1192 004436 022120      12$:  CMP     (R1)+,(R0)+      ;UPDATE POINTERS
1193 004440 000137 002510 JMP     DONE1          ;SEE IF ANY MORE
1194
1195      ;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
1196      ;*****
1197      ;.SBTTL CALCULATE THE VECTOR
1198      ; THIS ROUTINE, AFTER CHECKING THE VECTOR, CONVERTS IT TO ASCII,
1199      ; THEN UPDATES IT FOR THE NEXT UNIT OR DEVICE.
1200
1201 004444 010146      VECT:  MOV     R1,-(SP)
1202
1203      ;*****
1204      ;????????????????????????????????????????????????????????????????
1205      ; SPECIAL DEVICE VECTOR CHECKING.....
1206      ;
1207      ;* THE VECTOR IS OK BEFORE WE GET TO THE DMA'S
1208      ;
1209 004446 020127 010444 4$:  CMP     R1,#DMA          ;WORRY ABOUT CORRECT VECTOR?
1210 004452 002501      BLT     6$            ;BR IF NO
1211

```

CALCULATE THE VECTOR

```

1212          ;* THE PA6'S FOLLOW THE DR'S BUT HAVE A VECTOR GAP OF ONLY 4
1213          ;
1214 004454 020127 010452      CMP      R1,#DN+2      ;DN VECOR INCREASE BY 4
1215 004460 001476          BEQ      6$
1216 004462 020127 010456      CMP      R1,#DMBB+2     ;DMBB INCREASE BY 4
1217 004466 001473          BEQ      6$
1218 004470 020127 010636      CMP      R1,#RL+2      ;RL11 VECTOR INCREASE BY 4
1219 004474 001470          BEQ      6$
1220 004476 020127 010642      CMP      R1,#TS11+2     ;TS11 VECTOR INCREASES BY 4
1221 004502 001465          BEQ      6$
1222 004504 020127 010652      CMP      R1,#IP+2      ;TP11 VECTOR INCREASES BY 4
1223 004510 001462          BEQ      6$
1224 004512 020127 010666      CMP      R1,#RX+2      ;RX11 VECTOR INCREASES BY 4
1225 004516 001457          BEQ      6$
1226 004520 020127 010672      CMP      R1,#DRW+2     ;DRW VECTOR INCREASES BY 4
1227 004524 001454          BEQ      6$
1228 004526 020127 010676      CMP      R1,#DRB+2     ;DRB VECTOR INCREASES BY 4
1229 004532 001451          BEQ      6$
1230 004534 020127 010712      CMP      R1,#ML+2      ;ML11 VECTOR INCREASE BY 4
1231 004540 001446          BEQ      6$
1232 004542 020127 010726      CMP      R1,#UNA+2     ;UNA VECTOR INCREASE BY 4
1233 004546 001443          BEQ      6$
1234 004550 020127 010732      CMP      R1,#UDA+2     ;UDA VECTOR INCREASE BY 4
1235 004554 001440          BEQ      6$
1236 004556 020127 010752      CMP      R1,#VS100+2    ;VS100 VECTOR INCREASE BY 4
1237 004562 001435          BEQ      6$
1238 004564 020127 010756      CMP      R1,#TU81+2    ;TU81 VECTOR INCREASE BY 4
1239 004570 001432          BEQ      6$
1240 004572 020127 011012      CMP      R1,#QNA+2     ;QNA VECTOR INCREASE BY 4
1241 004576 001427          BEQ      6$
1242 004600 020127 011022      CMP      R1,#VS31+2    ;VS31 VECTOR INCREASE BY 4
1243 004604 001424          BEQ      6$
1244 004606 020127 011026      CMP      R1,#LNV+2     ;LNV-11 VECTOR INCREASE BY 4
1245 004612 001421          BEQ      6$
1246
1247
1248          ;* CORRECT VECTOR IF NEEDED
1249          ;
1250 004614 032737 000004 012030      BIT      #BIT2,VECTOR      ;CHECK TO SEE IF VECTOR IS AT AN EVEN 10
1251 004622 001403          BEQ      1$      ;BR IF OK
1252 004624 062737 000004 012030      ADD      #4,VECTOR      ;ELSE ADD 4
1253
1254          ;* GT40 VECTOR GOES UP BY 20 AND IS DIVISIBLE BY 20
1255          ;
1256 004632 020127 010526      1$:      CMP      R1,#GT+2      ;ARE WE THERE?
1257 004636 001007          BNE      6$      ;DON'T WORRY ABOUT IT
1258 004640 032737 000010 012030      BIT      #BIT3,VECTOR      ;CHECK IT
1259 004646 001403          BEQ      6$
1260 004650 062737 000010 012030      ADD      #10,VECTOR      ;MAKE IT RIGHT
1261
1262
1263          ;* AND FINALLY MAKE SURE IT'S IN BOUNDS
1264          ;
1265 004656 012701 012030      6$:      MOV      #VECTOR,R1      ;SET R1 FOR OTOA ROUTINE
1266 004662 021127 000774      CMP      @R1,#774      ;CHECK FOR HIGH VECTOR LIMIT
1267 004666 003405          BLE      2$      ;IT'S OK!
1268 004670          PRINT      #BADCON

```

CALCULATE THE VECTOR

```

004670 012705 015605      MOV      #BADCON,R5      ;ADDRESS TO R5
004674 104003      TYPE      ;GO TYPE IT
1269 004676 000137 001120      JMP      RES      ;DONE!
1270
1271      ;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
1272      ;*****
1273
1274
1275
1276
1277
1278 004702 012737 012076 004754 2$:      MOV      #DBUF+6,SOTOA+2 ;MODIFY OTOA ROUTINE - BUFFER START
1279 004710 012737 012070 005000      MOV      #DBUF,SOTOA+26 ;BUFFER END
1280 004716 005037 012076      CLR      DBUF+6      ;ZERO END OF BUFFER FOR CR/LF
1281 004722 104004      OTOA      ;CONVERT VECTOR
1282 004724 012737 012062 004754      MOV      #ABUF,SOTOA+2 ;RESET OTOA ROUTINE
1283 004732 012737 012054 005000      MOV      #A,SOTOA+26
1284 004740 066037 010120 012030      ADD      VST(R0),VECTOR ;ADJUST FOR NEXT
1285 004746 012601      5$:      MOV      (SP)+,R1      ;RESTORE R1
1286 004750 000207      RTS      PC
1287
1288
1289
1290
1291      .SBTTL OCTAL TO OCTAL ASCII
1292 004752 012704 012062      SOTOA:      MOV      #ABUF,R4      ;START OF BUFFER TO R4
1293 004756 010046      MOV      R0,-(SP)      ;SAVE R0
1294 004760 011103      MOV      (R1),R3      ;GET NUMBER
1295 004762 010300      MOV      R3,R0      ;SO DOES R0
1296 004764 042700 177770      1$:      BIC      #177770,R0      ;THE LAST 3 BITS ARE OF INTEREST!
1297 004770 052700 000060      BIS      #60,R0      ;MAKE IT AN ASCII NUMBER!
1298 004774 110044      MOV      R0,-(R4)      ;STORE IT
1299 004776 022704 012054      CMP      #A,R4      ;ARE WE DOWN?
1300 005002 001002      BNE      2$      ;NO-BR
1301 005004 012600      MOV      (SP)+,R0      ;YEP! RESTORE R0
1302 005006 000002      RTI
1303 005010 006203      2$:      ASR      R3      ;GET THE NEXT 3 BITS
1304 005012 006203      ASR      R3
1305 005014 006203      ASR      R3
1306 005016 010300      MOV      R3,R0      ;MOVE TO R0
1307 005020 000761      BR      1$      ;DOITTOITAGAIN!!!
1308
1309
1310      .SBTTL JUMPER SELECT
1311      ; CONVERT THE ADDRESS BITS THAT ARE SET TO THEIR ASCII EQUIVALENT
1312      ;
1313 005022 010046      JUMPER:      MOV      R0,-(SP)      ;SAVE R0
1314 005024 011100      MOV      @R1,R0      ;R0<=ADDRESS
1315 005026 012703 012070      MOV      #DBUF,R3      ;R3<=ASCII BUFFER
1316 005032 012704 177774      MOV      #-4,R4      ;GO FROM -4 TO 12
1317
1318      ;*****
1319      ;????????????????????????????????????????????????????????????
1320      ; THE DEVICES THAT USE JUMPERS USE THE JUMPER TABLE (JT).
1321      ;
1322 005036 020127 010516      CMP      R1,#DJ+2      ;THE DJ'S, DH'S, AND DQ'S USE JUMPERS
1323 005042 001437      BEQ      1$

```

JUMPER SELECT

```

1324 005044 020127 010522      CMP      R1, #DH+2
1325 005050 001434      BEQ      1$
1326 005052 020127 010542      CMP      R1, #DQ+2
1327 005056 001431      BEQ      1$
1328      ;LWL002 CMP      R1, #LKK+2
1329 005060 020127 010566      CMP      R1, #LK+2      ;LWL002
1330 005064 001426      BEQ      1$
1331
1332      ;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
1333      ;*****
1334      ;THE DEVICES THAT USE SWITCHES USE THE JUMPER TABLE (ST)
1335
1336 005066 020127 010552      CMP      R1, #DU+2
1337 005072 001420      BEQ      4$
1338 005074 020127 010556      CMP      R1, #DUP+2
1339 005100 001415      BEQ      4$
1340 005102 020127 010576      CMP      R1, #DMC+2
1341 005106 001412      BEQ      4$
1342 005110 020127 010602      CMP      R1, #DZ+2
1343 005114 001407      BEQ      4$
1344 005116 020127 010606      CMP      R1, #KMC+2
1345 005122 001404      BEQ      4$
1346 005124 020127 010726      CMP      R1, #UNA+2
1347 005130 001401      BEQ      4$
1348 005132 000416      BR      3$
1349 005134 012705 011774      4$: MOV      #ST, R5      ;R5<= SWITCH TABLE
1350 005140 000402      BR      2$
1351 005142 012705 012000      1$: MOV      #JT, R5      ;R5<= JUMPER TABLE
1352 005146 005204      2$: INC      R4      ;THE FIRST 3 BITS WILL BE CLEAR
1353 005150 006000      ROR      R0      ;SHIFT THE ADDRESS
1354 005152 103375      BCC      2$      ;IF CLR - TRY THE NEXT
1355 005154 020427 000012      CMP      R4, #12      ;
1356 005160 001403      BEQ      $      ;BR IF FINISHED
1357 005162 004737 005176      JSR      PC, CVTA      ;CONVERT TO ASCII
1358 005166 000767      BR      2$      ;DO IT AGAIN
1359 005170 012600      3$: MOV      (SP)+, R0      ;RESTORE R0
1360      ;LWL002 CLRB      -(R3)      ;ZERO BYTE FOR OUTPUT
1361 005172 105013      CLRB      (R3)      ;ZERO BYTE FOR OUTPUT      ;LWL002
1362 005174 000207      RTS      PC
1363
1364
1365      .SBTTL CONVERT BIT TO ASCII #
1366 005176 006304      CVTA: ASL      R4      ;R4 HAS A # BETWEEN 0 AND 11
1367 005200 060405      ADD      R4, R5      ;ADD IT TO R5 ---POINTS TO CORRECT ASCII CHAR.S
1368 005202 111523      MOV      (R5), (R3)+      ;MOV CHARACTERS TO BUFFER
1369 005204 116513 000001      MOV      1(R5), (R3)
1370 005210 160405      SUB      R4, R5      ;RESET R5
1371 005212 006204      ASR      R4      ;RESET R4
1372 005214 122327 000054      CMP      (R3)+, #',      ;TO MAKE IT NEAT---MAKE SURE THERE IS A COMMA
1373 005220 001402      BEQ      1$      ;YEP! OK!!
1374 005222 112723 000054      MOV      #'., (R3)+      ;PUT IT THERE
1375 005226 000207      1$: RTS      PC
1376
1377
1378
1379      .SBTTL INPUT ROUTINE
1380

```

INPUT ROUTINE

1381	005230	012700	021542	SINPUT: MOV	#BUFF,R0	;START OF BUFF TO R0	
1382	005234	105777	173442		TSTB	#TRB	;KILL LAST CHAR.
1383	005240	105777	173434	TST:	TSTB	#TRS	;SEE IF HE TYPED ANYTHING
1384	005244	100375			BPL	TST	;BR IF NO
1385	005246	117710	173430		MOVB	#TRB,(R0)	;STORE IT
1386	005252	142710	000200		BICB	#BIT7,(R0)	;STRIP PARITY
1387	005256	121027	000141		CMPB	(R0),#141	;SEE IF ITS LOWER CASE
1388				;LWL002	BLT	2#	;BR IF NO
1389	005262	002405			BLT	11#	;BR IF NO
1390	005264	121027	000172		CMPB	(R0),#172	;MAYBE ;LWL002
1391				;LWL002	BGT	2#	;BRANCH IF DEFFINATELY NOT
1392	005270	003002			BGT	11#	;BRANCH IF DEFINITELY NOT ;LWL002
1393	005272	142710	000040		BICB	#BIT5,(R0)	;IT IS - MAKE IT UPPER CASE IN BUFFER
1394	005276	121027	000023	11#:	CMPB	(R0),#23	;IS THIS XOFF? ;LWL002
1395	005302	001001			BNE	12#	;BRANCH IF NOT ;LWL002
1396	005304	000751			BR	SINPUT	;IF YES, START OVER ;LWL002
1397	005306	121027	000021	12#:	CMPB	(R0),#21	;IS THIS XON? ;LWL002
1398	005312	001001			BNE	2#	;BRANCH IF NOT ;LWL002
1399	005314	000745			BR	SINPUT	;IF YES, START OVER ;LWL002
1400	005316	121027	000003	2#:	CMPB	(R0),#3	;IS IT A ^C? ;LWL002
1401	005322	001002			BNE	6#	;NO
1402	005324	000137	001004		JMP	START	;LETS START IT AGAIN!!!
1403	005330	121027	000032	6#:	CMPB	(R0),#32	;IS IT A CONTROL Z
1404	005334	001002			BNE	7#	;BRANCH IF NOT
1405	005336	000137	001444		JMP	PRINT	;PRINT SELECTED LIST
1406	005342	005737	011760	7#:	TST	FLAGB	;IS IT OK TO USE CONTROL A
1407	005346	001430			BEQ	5#	;BRANCH IF NO
1408	005350	121027	000001		CMPB	(R0),#1	;IS IT A CONTROL A
1409	005354	001025			BNE	5#	;BRANCH IF NOT
1410	005356	105737	011740		TSTB	FLAG	;CHECK FOR VA OR FA
1411	005362	001404			BEQ	8#	;BRANCH IF FA
1412	005364	022702	007072		CMP	#DT2,R2	;CHECK IF AT BEGINING OF VA TABLE
1413	005370	001411			BEQ	10#	;BRANCH IF AT BEGINING
1414	005372	000403			BR	9#	;BRANCH TO BACK UP TABLES
1415	005374	022702	006562	8#:	CMP	#DT1,R2	;CHECK IF AT BEGINING OF FA TABLE
1416	005400	001405			BEQ	10#	;BRANCH IF AT BEGINING
1417	005402	162701	000002	9#:	SUB	#2,R1	;BACK UP MT1 OR MT2
1418	005406	162703	000002		SUB	#2,R3	;BACK UP QT1 OR QT2
1419	005412	005052			CLR	#-(R2)	;BACK UP DT1 OR DT2 AND CLEAR NUMBER
1420	005414	012626		10#:	MOV	(SP)+,(SP)+	;RESTORE STACK
1421	005416				PRINT	#CRLF	;DO A CRRAGE RETURN AND LINE FEED
	005416	012705	014510		MOV	#CRLF,R5	;ADDRESS TO R5
	005422	104003			TYPE		;GO TYPE IT
1422	005424	000137	001410		JMP	QUES	;START AGAIN
1423	005430	121027	000177	5#:	CMPB	(R0),#177	;IS IT A DELETE?
1424	005434	001431			BEQ	RUBOUT	;YEP!
1425	005436	032737	000002	011740	BIT	#BIT1,FLAG	;ARE WE IN DELETE MODE?
1426	005444	001411			BEQ	3#	;NO-BRANCH
1427	005446	042737	000002	011740	BIC	#BIT1,FLAG	;YES..NOT ANY MORE!
1428	005454	012777	000134	173224	MOV	#'\,STPB	;SEPARATE THE DELETED FROM THE NEW.
1429	005462	105777	173216	4#:	TSTB	#TPS	;WAIT FOR READY.
1430	005466	100375			BPL	4#	
1431	005470	121027	000015	3#:	CMPB	(R0),#CR	;A TERMINATOR?
1432	005474	001403			BEQ	1#	;YES'
1433	005476	112077	173204		MOVB	(R0)+,STPB	;NO..JUST ECHO AND
1434	005502	000656			BR	TST	;GET NEXT CHARACTER.
1435	005504			1#	PRINT	#CRLF	;PRINT A CR AND LF.

INPUT ROUTINE

005504	012705	014510		MOV	#CRLF,R5		;ADDRESS TO R5
005510	104003			TYPE			;GO TYPE IT
1436 005512	012700	021542		MOV	#BUFF,R0		;POINT R0 TO BEGINNING OF BUFFER.
1437 005516	000002			RTI			;RETURN
1438 005520	022700	021542	RUBOUT:	CMP	#BUFF,R0		;CAN'T DELETE PAST BEGINNING OF BUFFER
1439 005524	002245			BGE	TST		
1440 005526	122737	000126	012277	CMPB	#V,TERMTP		;IS THIS A VIDEO TERMINAL? ;LWL002
1441 005534	001423			BEQ	4:		;BRANCH IF IT IS ;LWL002
1442 005536	032737	000002	011740	BIT	#BIT1,FLAG		;IN DELETE MODE?
1443 005544	001011			BNE	3:		;BR IF YES
1444 005546	052737	000002	011740	BIS	#BIT1,FLAG		;NOW WE ARE!
1445 005554	112777	000134	173124	MOVB	#'\,ATPB		;SEPARATE THAT THAT IS FROM THAT THAT IS NO MORE!!
1446 005562	105777	173116	2:	TSTB	ATPS		;WAIT
1447 005566	100375			BPL	2:		
1448 005570	114077	173112	3:	MOVB	-(R0),ATPB		;ECHO DELETE CHAR.
1449 005574	105777	17310:	1:	TSTB	ATPS		;WAIT FOR READY
1450 005600	100375			BPL	1:		
1451 005602	000616			BR	TST		;GET NEXT
1452 005604	105740		4:	TSTB	-(R0)		;BACK-UP POINTER ;LWL002
1453 005606	010046			MOV	R0,-(SP)		;SAVE R0 POINTER ;LWL002
1454 005610	012760	012273		MOV	#DCHAR,R0		;GET DELETE SEQUENCE OF CHARACTERS ;LWL002
1455 005614	112077	173066	6:	MOVB	(R0)+,ATPB		;OUTPUT FIRST CHARACTER ;LWL002
1456 005620	105777	173060	5:	TSTB	ATPS		;READY FOR NEXT ONE? ;LWL002
1457 005624	100375			BPL	5:		;BRANCH IF NOT ;LWL002
1458 005626	105710			TSTB	(R0)		;END OF STRING? ;LWL002
1459 005630	001371			BNE	6:		;NO,NEXT CHARACTER ;LWL002
1460 005632	012600			MOV	(SP)+,R0		;RESTORE R0 ;LWL002
1461 005634	000601			BR	TST		;BACK TO INPUT ROUTINE ;LWL002
1462							
1463							
1464							
1465 005636	010146			SDATO:	MOV R1,-(SP)		;SAVE R1
1466 005640	010346			MOV	R3,-(SP)		;AND R3
1467 005642	012701	021542		MOV	#BUFF,R1		;INPUT BUFF TO R1
1468 005646	005000			CLR	R0		;R0 GETS RESULT
1469 005650	112103			CMP1:	MOVB (R1)+,R3		;GET CHAR.
1470 005652	122703	000015		CMPB	#CR,R3		;EOL?
1471 005656	001433			BEQ	EXIT		;YES - BR
1472 005660	120327	000060		CMPB	R3,#60		;SEE IF IT'S A NUMBER
1473 005664	002414			BLT	IL		;CAN'T BE LESS THAN 60
1474 005666	120327	000071		CMPB	R3,#71		; OR
1475 005672	003011			BGT	IL		;GREATER THAN 71
1476 005674	162703	000060		SUB	#60,R3		;CHANGE TO OCTAL
1477 005700	006300			ASL	R0		;SHIFT OLD
1478 005702	060003			ADD	R0,R3		;ADD TO NEW
1479 005704	006300			ASL	R0		;OLD * 2 ; A DIVIDE BY 8.ROUTINE!
1480 005706	006300			ASL	R0		; *2 AGAIN
1481 005710	060300			ADD	R3,R0		;ADD (2*OLD+NEW) TO 10*OLD
1482 005712	103356			BCC	CMP1		;NO CARRY -- OK
1483 005714	000407			BR	TOB		;ELSE IT'S TOO BIG
1484 005716				IL:	PRINT #ILL		;IT'S AN ILLEGAL NUMBER
005716	012705	014513		MOV	#ILL,R5		;ADDRESS TO R5
005722	104003			TYPE			;GO TYPE IT
1485 005724	052737	100000	011740	ER:	BIS #BIT15,FLAG		;SET ERROR FLAG
1486 005732	000411			BR	OK		;EXIT
1487 005734	104006			TOB:	OTDA		
1488 005736				PRINT	#TB		;TOO BIG

DECIMAL ASCII TO OCTAL

```

005736 012705 014546      MOV    #TB,R5      ;ADDRESS TO R5
005742 104003      TYPE    ;GO TYPE IT
1489 005744 000767      BR      ER      ;GO SET ERROR
1490 005746 016601 000002 EXIT: MOV    2(SP),R1    ;RESTORE R1
1491 005752 021100      CMP    (R1),R0    ;CHECK FOR MAXIMUM NUMBER
1492 005754 002767      BLT    T0B      ;BR IF TOO BIG
1493 005756 012603      OK:   MOV    (SP)+,R3    ;RESTORE R3
1494 005760 012601      MOV    (SP)+,R1    ;RESTORE R1
1495 005762 000002      RTI
1496
1497      .SBTTL EVEN OR ODD?
1498 005764 032737 000010 012034 BITE:  BIT    #10,ADDR    ;ENTER FOR EVEN
1499 005772 001403      BEQ    BEX      ;BR IF EVEN
1500 005774 062737 000010 012034      ADD    #10,ADDR    ;ADD 10 - EVEN OR ODD
1501 006002 000207      BEX:   RTS    PC
1502
1503 006004 032737 000010 012034 BITE1: BIT    #10,ADDR    ;ENTER TO MAKE SURE ADDR 0 OR 40
1504 006012 001403      BEQ    BADD      ;BR IF EVEN
1505 006014 062737 000010 012034      ADD    #10,ADDR    ;ADD 10 TO MAKE IT EVEN
1506 006022 032737 000020 012034 BADD:  BIT    #20,ADDR    ;ENTER FOR EVEN
1507 006030 001403      BEQ    BEX1     ;BR IF EVEN
1508 006032 062737 000020 012034      ADD    #20,ADDR    ;ADD 20 TO MAKE IT EVEN
1509 006040 000207      BEX1:  RTS    PC
1510 006042 032737 000010 012034 BITE2: BIT    #10,ADDR    ;ENTER TO MAKE SURE ADDR 0 OR 100
1511 006050 001403      BEQ    BADS     ;BR IF ADDR BIT TWO IS CLEAR
1512 006052 062737 000010 012034      ADD    #10,ADDR    ;ADD 10 TO MAKE IT EVEN
1513 006060 032737 000020 012034 BADS:  BIT    #20,ADDR    ;CHECK FOR BIT FOUR
1514 006066 001403      BEQ    BADD5    ;BR IF ADDR BIT FOUR IS CLEAR
1515 006070 062737 000020 012034      ADD    #20,ADDR    ;ADD 20 TO MAKE IT EVEN
1516 006076 032737 000040 012034 BADD5: BIT    #40,ADDR    ;CHECK FOR ADDR BIT FIVE
1517 006104 001403      BEQ    BEX2     ;BR IF ADDR BIT FIVE IS CLEAR
1518 006106 062737 000040 012034      ADD    #40,ADDR    ;ADD 40 TO MAKE IT EVEN
1519 006114 000207      BEX2:  RTS    PC
1520
1521      .SBTTL OCTAL TO DECIMAL ASCII
1522      ;DIVIDE BY 1.*10.*N
1523      ; SET UP FOR A MAXIMUM OF TWO DIGITS.
1524      ;
1525 006116 012700 014575      SOTDA: MOV    #MAXIUM,R0    ;POINTER TO STORE ASCII.
1526 006122 012705 006176      MOV    #DIG2,R5      ;POINTER TO DIVISOR TABLE.
1527 006126 017604 000006      MOV    #6(SP),R4    ;GET DIVIDEND.
1528 006132 012701 177777      2$:   MOV    #-1,R1      ;RESET COUNTER.
1529 006136 005201      1$:   INC    R1      ;THE QUOTIENT!
1530 006140 161504      SUB    @R5,R4      ;DO THE PSEUDO DIVIDE!
1531 006142 103375      BCC    1$      ;NO CARRY-TRY AGAIN.
1532 006144 061504      ADD    @R5,R4      ;REPLACE WHAT WE TOOK.
1533 006146 005745      TST    -(R5)      ;EOT?
1534 006150 001404      BEQ    3$      ;BR IF YES
1535 006152 062701 000060      ADD    #'0,R1      ;MAKE IT ASCII
1536 006156 110120      MOVB   R1,(R0)+    ;STORE IT.
1537 006160 000764      BR      2$      ;NEXT CHARACTER.
1538 006162 062701 000060      3$:   ADD    #'0,R1      ;MAKE LAST ASCII
1539 006166 110120      MOVB   R1,(R0)+    ;AND STORE IT
1540 006170 000002      RTI      ;RETURN
1541
1542 006172 000000      .WORD   0
1543 006174 000001      .WORD   1

```

OCTAL TO DECIMAL ASCII

```

1544 006176 000012          DIG2:  .WORD  10.
1545
1546          .SBTTL OUTPUT ROUTINE
1547 006200 105777 172500      STYPE:  TSTB  @TPS      ;READY?
1548 006204 100375              BPL    STYPE      ;NO
1549 006206 105777 172466      TSTB  @TRS      ;CHECK FOR CHARACTER
1550 006212 100027              BPL    XON       ;BRANCH IF NOT
1551 006214 017737 172462 011772  MOV   @TRB,SVCHA ;SAVE CHARACTER
1552 006222 042737 177600 011772  BIC   @177600,SVCHA ;CLEAR UNWANTED BITS
1553 006230 122737 000023 011772  CMPB  @23,SVCHA  ;CHECK FOR XOFF OR CONTROL S
1554 006236 001015              BNE    XON       ;BRANCH IF NOT
1555 006240 105777 172434      XOFF:  TSTB  @TRS      ;CHECK FOR CHARACTER
1556 006244 100375              BPL    XOFF      ;WAIT FOR XON
1557 006246 017737 172430 011772  MOV   @TRB,SVCHA ;SAVE CHARACTER
1558 006254 042737 177600 011772  BIC   @177600,SVCHA ;CLEAR UNWANTED BITS
1559 006262 122737 000021 011772  CMPB  @21,SVCHA  ;CHECK FOR XON OR CONTROL Q
1560 006270 001363              BNE    XOFF      ;WAIT FOR CONTROL Q
1561 006272 111577 172410      XON:  MOVB  (R5),@TPB  ;PRINT IT
1562 006276 122537 012140      CMPB  (R5)+,CHAR  ;FILL CHAR.???
1563 006302 001412              BEQ    FILL      ;BR IF YES
1564 006304 105715              TENDM: TSTB  (R5)     ;EOD ??
1565 006306 001004              BNE    1$        ;NOT FOR SURE.
1566 006310              PRINT  @CRLF             ;YES
      006310 012705 014510      MOV   @CRLF,R5    ;ADDRESS TO R5
      006314 104003              TYPE              ;GO TYPE IT
1567 006316 000002              RTI
1568 006320 121527 000200      1$:  CMPB  (R5),@NOCRLF ;MAYBE ??
1569 006324 001325              BNE    STYPE      ;NO
1570 006326 000002              RTI              ;YES
1571 006330 012727      FILL:  MOV   (PC)+,(PC)+    ;SET FILL
1572 006332 000010      FILCNT: .WORD  10          ;10 TO START
1573 006334 000000      FILLER: .WORD  0
1574 006336 105777 172342      1$:  TSTB  @TPS      ;READY?
1575 006342 100375              BPL    1$        ;NO
1576 006344 112777 000000 172334  MOVB  @0,@TPB  ;TYPE A NULL.
1577 006352 005337 006334      DEC   FILLER      ;1 LESS TO DO
1578 006356 001367              BNE    1$        ;DO IT AGAIN
1579 006360 000751              BR     TENDM      ;GO TEST FOR END OF DATA
1580
1581
1582
1583
1584          .SBTTL DATA TABLES
1585          .NLIST BEX

```

DATA TABLES

```

1587 ;*****
1588 ;????????????????????????????????????????????????????????
1589 ; THE TABLES ARE THE GUTS OF THIS PROGRAM !
1590 ;
1591 ;* THESE TABLES MUST REMAIN IN ORDER !!
1592 ;*
1593 ;*
1594
1595
1596 ;* QUESTION TABLE (QT1) - POINTS TO THE ASCII DEVICE NAMES
1597 ;* USED FOR ASKING THE QUESTIONS AND
1598 ;* WHEN TYPING THE ANSWERS FOR THE 'VA' OPTION.
1599 ;*
1600 006362 012300 012321 012361 QT1: .WORD Q1,Q2,Q3,Q4,Q5 ;DC,DL,DP,DMA,DN
1601 006374 012444 012465 012516 .WORD Q6,Q7,Q8,Q9,Q10 ;DMB8,DR,PAR,PAP,LPD
1602 006406 012601 012622 012643 .WORD Q11,Q12,Q13,Q14,Q15 ;DT,DX,DLC,DJ,DH
1603 006420 012736 012757 013000 .WORD Q16,Q17,Q18,Q19,Q20 ;GT,VSV,LPS,DQ,KWW
1604 006432 013064 013106 013127 .WORD Q21,Q22,Q23,Q24,Q25 ;DU,DUP,DV,LK,DWUN
1605 006444 013212 013235 013270 .WORD Q26,Q27,Q28,Q29,Q30 ;DMC,DZ,KMC11,LPP11,V21
1606 006456 013353 013374 013415 .WORD Q31,Q32,Q33,Q34,Q35 ;V31,V01,DWR,RL11,TS11
1607 006470 013501 013522 013544 .WORD Q36,Q37,Q38,Q39,Q40 ;LPAK,IP,KWC,RESR,RX11
1608 006502 013642 013663 013704 .WORD Q41,Q42,Q43,Q44,Q45 ;DRW,DRB,DMP,DPV,ML
1609 006514 013767 014010 014031 .WORD Q46,Q47,Q48,Q49,Q50 ;ISB,DMV,UNA,UDA,DMF
1610 006526 014114 014135 014156 .WORD Q51,Q52,Q53,Q54,Q55 ;KMS,PCL,VS,TU,KMV
1611 006540 014241 014262 014303 .WORD Q56,Q57,Q58,Q59,Q60 ;KCT,IEX,DHV,DMZ,CPI
1612 006552 014377 014420 014441 .WORD Q61,Q62,Q63,Q64 ;QNA,QVS,VS31,LNV
1613
1614
1615 ;* DEVICE TABLE (VA OPTION) (DT1) - POINTERS TO THE DEVICE DATA TABLE
1616 ;* THE ORDER IS DIRECTLY RELATED TO THE
1617 ;* QT1 TABLE. SO THAT WHEN YOU ASK FOR
1618 ;* THE NUMBER OF DC11'S YOU STORE THE
1619 ;* ANSWER IN THE DEVICE DATA TABLE FOR
1620 ;* THE DC11 AND NOT THE PA6'S.
1621 006562 010430 010434 010440 DT1: .WORD DC,DL,DP,DMA,DN
1622 006574 010454 010460 010464 .WORD DMB8,DR,PAR,PAP,LPD
1623 006606 010500 010504 010510 .WORD DT,DX,DLC,DJ,DH
1624 006620 010524 010530 010534 .WORD GT,VSV,LPS,DQ,KWW
1625 006632 010550 010554 010560 .WORD DU,DUP,DV,LK,DWUN
1626 006644 010574 010600 010604 .WORD DMC,DZ,KMC,LPP,V21
1627 006656 010620 010624 010630 .WORD V31,V01,DWR,RL,TS11
1628 006670 010644 010650 010654 .WORD LPAK,IP,KWC,RESR,RX
1629 006702 010670 010674 010700 .WORD DRW,DRB,DMP,DPV,ML
1630 006714 010714 010720 010724 .WORD ISB,DMV,UNA,UDA,DMF
1631 006726 010740 010744 010750 .WORD KMS,PCL,VS100,TU81,KMV
1632 006740 010764 010770 010774 .WORD KCT32,IEX,DHV,DMZ32,CPI32
1633 006752 011010 011014 011020 .WORD QNA,QVSS,VS31,LNV

```

DATA TABLES

```

1635      ;*QUESTION TABLE 2 (QT2) POINTS TO THE ASCII DEVICE NAMES
1636      ;*      USED FOR ASKING THE QUESTIONS FOR
1637      ;*      THE 'FA' OPTION
1638      ;*
1639 006762 012674 012715 013021 QT2: .WORD Q14,Q15,Q19,Q21,Q22 ;DJ,DH,DQ,DU,DUP
1640 006774 013150 013212 013235      .WORD Q24,Q26,Q27,Q28,Q29 ;LK,DMC,DZ,KMC,LPP
1641 007006 013332 013353 013415      .WORD Q30,Q31,Q33,Q34,Q36 ;V21,V31,DWR,RL,LPA11-K
1642 007020 013544 013565 013606      .WORD Q38,Q39,Q40,Q41,Q42 ;KWC,RESER,RX11,DRW,DRB
1643 007032 013704 013725 013767      .WORD Q43,Q44,Q46,Q47,Q48 ;DMP,DPV,ISB,DMV,UNA
1644 007044 014052 014073 014114      .WORD Q49,Q50,Q51,Q53,Q54 ;UDA,DMF,KMS,VS100,TU81
1645 007056 014220 014303 014325      .WORD Q55,Q58,Q59,Q60,Q62 ;KMW,DHV,DMZ32,CPI32,QVSS
1646 007070 014441      .WORD Q63 ;VS31
1647
1648
1649      ;*DEVICE TABLE 2 (DT2) - USED WITH Q.2 FOR STORING ANSWERS AND
1650      ;*      FOR COMPUTING THE FLOATING ADDRESSES. IT
1651      ;*      THEREFORE MUST BE IN THE ORDER OF THE
1652      ;*      PRIORITY OF THE DEVICE. IT ALSO MUST
1653      ;*      END WITH A ZERO. (FA OPTION)
1654      ;*
1655 007072 010514 010520 010540 DT2: .WORD DJ,DH,DQ,DU,DUP
1656 007104 010564 010574 010600      .WORD LK,DMC,DZ,KMC,LPP
1657 007116 010614 010620 010630      .WORD V21,V31,DWR,RL,LPAK
1658 007130 010654 010660 010664      .WORD KWC,RESR,RX,DRW,DRB
1659 007142 010700 010704 010714      .WORD DMP,DPV,ISB,DMV,UNA
1660 007154 010730 010734 010740      .WORD UDA,DMF,KMS,VS100,TU81
1661 007166 010760 010774 011000      .WORD KMW,DHV,DMZ32,CPI32,QVSS
1662 007200 011020 000000      .WORD VS31,0
1663
1664

```

DATA TABLES

```

1666      ; * MAXIMUM NUMBER TABLES 1 & 2
1667      ; * (MT1,MT2) - USED FOR THE 'VA' AND 'FA' OPTIONS RESPECTIVELY
1668      ; *
1669      ; *   CONTAIN THE MAXIMUM NUMBER OF CONTROLLERS THAT
1670      ; *   IS SUPPORTED FOR EACH DEVICE. THEY MUST REMAIN
1671      ; *   IN THE SAME ORDER AS QT1,DT1 AND QT2,DT2 AND
1672      ; *   ALSO END WITH A ZERO.
1673 007204 000040 000020 000040 MT1: .WORD 32.,16.,32.,16.,16. ;DC,DL,DP,DMA,DN
1674 007216 000020 000020 000010      .WORD 16.,16.,8.,8.,8. ;DMBB,DR,PAR,PAP,LPD
1675 007230 000010 000004 000037      .WORD 8.,4.,31.,16.,16. ;DT,DX,DLC,DJ,DH
1676 007242 000004 000004 000001      .WORD 4,4,1,16.,1 ;GT,VSV,LPS,DQ,KWW
1677 007254 000020 000020 000004      .WORD 16.,16.,4,2,1 ;DU,DUP,DV,LK,DWUN
1678 007266 000020 000020 000004      .WORD 16.,16.,4,2,2 ;DMC,DZ,KMC11,LPP,V21
1679 007300 000002 000002 000002      .WORD 2,2,2,4,4 ;V31,V01,DWR,RL,TS11
1680 007312 000004 000002 000002      .WORD 4,2,2,1,4 ;LPAK,IP11,KW11C,RESR,RX11
1681 007324 000020 000020 000012      .WORD 16.,16.,10.,10.,1 ;DRW,DRB,DMP,DPV,ML
1682 007336 000004 000012 000012      .WORD 4,10.,10.,10.,10. ;ISB,DMV,UNA,UDA,DMF
1683 007350 000012 000004 000006      .WORD 10.,4,6,10.,10. ;KMS,PCL,VS100,TU81,KMV
1684 007362 000004 000001 000012      .WORD 4,1,10.,10.,10. ;KCT32,IEX,DHV,DMZ32,CPI32
1685 007374 000004 000004 000012      .WORD 4,4,10.,2,0 ;QNA,QVSS,VS31,LNV
1686
1687
1688 007406 000020 000020 000020 MT2: .WORD 16.,16.,16.,16.,16. ;DJ,DH,DQ,DU,DUP
1689 007420 000002 000020 000020      .WORD 2,16.,16.,4,2 ;LK,DMC,DZ,KMC11,LPP
1690 007432 000002 000002 000002      .WORD 2,2,2,4,4 ;V21,V31,DWR,RL,LPAK
1691 007444 000002 000001 000004      .WORD 2,1,4,16.,16. ;KWC,RESER,RX11,DRW,DRB
1692 007456 000012 000012 000004      .WORD 10.,10.,4,10.,10. ;DMP,DPV,ISB,DMV,UNA
1693 007470 000012 000012 000012      .WORD 10.,10.,10.,6,10. ;UDA,DMF,KMS,VS100,TU81
1694 007502 000012 000012 000012      .WORD 10.,10.,10.,10.,4 ;KMV,DHV,DMZ32,CPI32,QVSS
1695 007514 000012 000000      .WORD 10.,0 ;VS31

```

DATA TABLES

```

1697
1698
1699
1700
1701 007520 000010 000010 177770
1702 007532 000010 177770 000010
1703 007544 000002 000040 000010
1704 007556 000010 000010 000040
1705 007570 000010 000010 000040
1706 007602 000010 000010 000010
1707 007614 000020 000160 000010
1708 007626 000020 000400 000010
1709 007640 000010 000010 000010
1710 007652 000010 000020 000010
1711 007664 000020 000040 000020
1712 007676 000040 000020 000020
1713 007710 000020 000100 000010
1714
1715
1716
1717
1718
1719
1720
1721
1722 007720 174000 176500 174770
1723 007732 170500 167770 172600
1724 007744 177420 176200 175610
1725 007756 172000 172000 170400
1726 007770 000000 000000 175000
1727 010002 000000 000000 000000
1728 010014 000000 172600 000000
1729 010026 170460 171400 000000
1730 010040 000000 000000 000000
1731 010052 000000 000000 000000
1732 010064 000000 164200 000000
1733 010076 164400 164100 000000
1734 010110 174440 000000 000000
1735
1736
1737
1738
1739
1740 010120 000010 000010 000010
1741 010132 000004 000010 000010
1742 010144 000010 000010 000010
1743 010156 000020 000020 000030
1744 010170 000010 000010 000020
1745 010202 000010 000010 000010
1746 010214 000010 000010 000010
1747 010226 000010 000004 000010
1748 010240 000004 000004 000010
1749 010252 000010 000010 000004
1750 010264 000020 000010 000004
1751 010276 000010 000010 000010
1752 010310 000004 000020 000004
1753

; * NEXT ADDRESS TABLE
; * (NAT) - USED TO CALCULATE THE NEXT CSR ADDRESS (VA OPTION)
; * OF A LIKE DEVICE.
; *
; *
NAT: .WORD 10,10,-10,10,10 ;DC,DL,DP,DMA,DN
      .WORD 10,-10,10,10,10 ;DMBB,DR,PAR,PAP,LPD
      .WORD 2,40,10,10,20 ;DT,DX,DLC,DJ,DH
      .WORD 10,10,40,10,10 ;GT,VSV,LPS,DQ,KWW
      .WORD 10,10,40,10,0 ;DU,DUP,DV,LK,DWUN
      .WORD 10,10,10,10,10 ;DMC,DZ,KMC11,LPP11,V21
      .WORD 20,160,10,10,4 ;V31,V01,DWR,RL,TS11
      .WORD 20,400,10,10,10 ;LPAK,IP11,KWC,RESR,RX11
      .WORD 10,10,10,10,0 ;DRW,DRB,DMP,DPV,ML
      .WORD 10,20,10,4,40 ;ISB,DMV,UNA,UDA,DMF
      .WORD 20,40,20,4,20 ;KMS,PCL,VS100,TU81,KMV
      .WORD 40,20,20,40,40 ;KCT32,IEX,DHV,DMZ32,CPI32
      .WORD 20,100,10,40 ;QNA,QVSS,VS31,LNV

; * ADDRESS TABLE
; * (ADDRT) - USED FOR LOADING THE DEVICE DATA TABLE (VA + FA OPTION)
; * WITH THE FIRST ADDRESS FOR EACH DEVICE.
; * MUST HAVE A ZERO ENTRY FOR
; * FLOATING ADDRESS DEVICES
; *
; *
ADDRT: .WORD 174000,176500,174770,175000,175200 ;DC,DL,DP,DMA,DN
        .WORD 170500,167770,172600,172700,172700 ;DMBB,DR,PAR,PAP,LPD
        .WORD 177420,176200,175610,160010,000000 ;DT,DX,DLC,DJ,DH
        .WORD 172000,172000,170400,000000,172400 ;GT,VSV,LPS,DQ,KWW
        .WORD 000000,000000,175000,000000,172500 ;DU,DUP,DV,LK,DWUN
        .WORD 000000,000000,000000,000000,000000 ;DMC,DZ,KMC11,LPP,V21
        .WORD 000000,172600,000000,000000,172524 ;V31,V01,DWR,RL,TS11
        .WORD 170460,171400,000000,000000,000000 ;LPAK,IP11,KWC,RESR,RX11
        .WORD 000000,000000,000000,000000,176400 ;DRW,DRB,DMP,DPV,ML
        .WORD 000000,000000,000000,000000,000000 ;ISB,DMV,UNA,UDA,DMF
        .WORD 000000,164200,000000,000000,000000 ;KMS,PCL,VS100,TU81,KMV
        .WORD 164400,164100,000000,000000,000000 ;KCT32,IEX,DHV,DMZ32,CPI32
        .WORD 174440,000000,000000,176200 ;QNA,QVSS,VS31,LNV

; * VECTOR SIZE TABLE
; * (VST) - USED TO CALCULATE THE NEXT VECTOR (VA OPTION)
; *
; *
VST: .WORD 10,10,10,10,4 ;DC,DL,DP,DMA,DN
      .WORD 4,10,10,10,10 ;DMBB,DR,PAR,PAP,LPD
      .WORD 10,10,10,10,10 ;DT,DX,DLC,DJ,DH
      .WORD 20,20,30,10,10 ;GT,VSV,LPS,DQ,KWW
      .WORD 10,10,20,10,10 ;DU,DUP,DV,LK,DWUN
      .WORD 10,10,10,10,10 ;DMC,DZ,KMC11,LPP11,V21
      .WORD 10,10,10,4,4 ;V31,V01,DWR,RL,TS11
      .WORD 10,4,10,4,4 ;LPAK,IP,KWC,RESR,RX11
      .WORD 4,4,10,10,4 ;DRW,DRB,DMP,DPV,ML
      .WORD 10,10,4,4,40 ;ISB,DMV,UNA,UDA,DMF
      .WORD 20,10,4,4,10 ;KMS,PCL,VS100,TU81,KMV
      .WORD 10,10,10,30,30 ;KCT32,IEX,DHV,DMZ32,CPI32
      .WORD 4,20,4,4 ;QNA,QVSS,VS31,LNV

```

DATA TABLES

1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766

010320 011036 011052 011066
010332 011132 011146 011162
010344 011226 011242 011256
010356 011322 011336 011352
010370 011416 011432 011446
010402 011512 011526 011542
010414 011606 011622 011640
010426 011725

;* ASCDEV - USED INPLACE OF QT2 FOR THE 'FA' OPTION WHEN TYPING ANSWERS

```

;*
ASCDEV: .WORD JS,HS,QS,US,UPS
        .WORD LKS,MCS,ZS,KMCS,PPS
        .WORD MV21,MV31,WRS,RLS,LPAKS
        .WORD KWCS,RESRS,RXS,DRWS,DRBS
        .WORD DMPS,DPVS,ISBS,DMVS,UNAS
        .WORD UDAS,DMFS,KMSS,VS100S,TU81S
        .WORD KMVS,DHVS,DMZ32S,CPI32S,QVSSS
        .WORD VS31S
    
```

DATA TABLES

1768		;	*	DEVICE DATA TABLE	
1769		;	*		
1770		;	*	MUST REMAIN IN ORDER OF THE VECTOR ASSIGNMENTS	
1771		;	*		
1772		;	*	MUST END WITH TWO MINUS ONES	
1773		;	*		
1774	010430			DEVSTR = .	
1775		;	*		
1776	010430	DC:		.BLKW	2
1777	010434	DL:		.BLKW	2
1778	010440	DP:		.BLKW	2
1779	010444	DMA:		.BLKW	2
1780	010450	DN:		.BLKW	2
1781	010454	DMBB:		.BLKW	2
1782	010460	DR:		.BLKW	2
1783	010464	PAR:		.BLKW	2
1784	010470	PAP:		.BLKW	2
1785	010474	LPD:		.BLKW	2
1786	010500	DT:		.BLKW	2
1787	010504	DX:		.BLKW	2
1788	010510	DLC:		.BLKW	2
1789	010514	DJ:		.BLKW	2
1790	010520	DH:		.BLKW	2
1791	010524	GT:		.BLKW	2
1792	010530	VSV:		.BLKW	2
1793	010534	LPS:		.BLKW	2
1794	010540	DQ:		.BLKW	2
1795	010544	KWW:		.BLKW	2
1796	010550	DU:		.BLKW	2
1797	010554	DUP:		.BLKW	2
1798	010560	DV:		.BLKW	2
1799	010564	LK:		.BLKW	2
1800	010570	DWUN:		.BLKW	2
1801	010574	DMC:		.BLKW	2
1802	010600	DZ:		.BLKW	2
1803	010604	KMC:		.BLKW	2
1804	010610	LPP:		.BLKW	2
1805	010614	V21:		.BLKW	2
1806	010620	V31:		.BLKW	2
1807	010624	V01:		.BLKW	2
1808	010630	DWR:		.BLKW	2
1809	010634	RL:		.BLKW	2
1810	010640	TS11:		.BLKW	2
1811	010644	LPAK:		.BLKW	2
1812	010650	IP:		.BLKW	2
1813	010654	KWC:		.BLKW	2
1814	010660	RESR:		.BLKW	2
1815	010664	RX:		.BLKW	2
1816	010670	DRW:		.BLKW	2
1817	010674	DRB:		.BLKW	2
1818	010700	DMP:		.BLKW	2
1819	010704	DPV:		.BLKW	2
1820	010710	ML:		.BLKW	2
1821	010714	ISB:		.BLKW	2
1822	010720	DMV:		.BLKW	2
1823	010724	UNA:		.BLKW	2
1824	010730	UDA:		.BLKW	2

; 2 WORDS / DEVICE
 ; FIRST - # OF THEM
 ; SECOND - 1ST ADDRESS

DATA TABLES

1825	010734				DMF:	.BLKW	2
1826	010740				KMS:	.BLKW	2
1827	010744				PCL:	.BLKW	2
1828	010750				VS100:	.BLKW	2
1829	010754				TU81:	.BLKW	2
1830	010760				KMV:	.BLKW	2
1831	010764				KCT32:	.BLKW	2
1832	010770				IEX:	.BLKW	2
1833	010774				DHV:	.BLKW	2
1834	011000				DMZ32:	.BLKW	2
1835	011004				CPI32:	.BLKW	2
1836	011010				QNA:	.BLKW	2
1837	011014				QVSS:	.BLKW	2
1838	011020				VS31:	.BLKW	2
1839	011024				LNV:	.BLKW	2
1840					;	*	
1841		011030			DEVEND	=	.
1842					;	*	
1843	011030	000000	177777	177777		.WORD	0,-1,-1
1844					;	*	
1845					;	*	END OF DEVICE DATA TABLE
1846					;	*	
1847		000100				DEVcnt=DEVEND-DEVSTR/4	

DATA TABLES

```

1849      ;*
1850      ;* THESE ASCII MESSAGES ARE POINTED TO BY THE 'ASCDEV' TABLE.
1851      ;* USED FOR THE 'FA' OPTION
1852      ;* THEY DEFINE THE DEVICE, THE BOARD WITH THE JUMPERS/SWITCHES,
1853      ;* WHICH JUMPERS TO CUTOUT OR WHICH SWITCH AND THE SWITCHES TO
1854      ;* TURN ON OR OFF.
1855
1856      .REM †
1857      FATABL:
1858      JS:      .ASCII / DJ11'S      M105    JUMPER(S) TO CUTOUT/<NOCRLF>
1859      HS:      .ASCII / DH11'S      M7277   JUMPER(S) TO CUTOUT/<NOCRLF>
1860      QS:      .ASCII / DQ11'S      M105    JUMPER(S) TO CUTOUT/<NOCRLF>
1861      US:      .ASCII / DU11'S      M7822   SW SWITCHES IN OFF POSITION/<NOCRLF>
1862      UPS:     .ASCII / DUP11'S     M7867   SW1 SWITCHES IN OFF POSITION/<NOCRLF>
1863      LKS:     .ASCII / LK11'S      M7060   JUMPER(S) TO CUTOUT/<NOCRLF>
1864      MCS:     .ASCII / DMC11 DMR11'S M8200,E113,M8207,E127-SWITCHES IN OFF POSITION/<NOCRLF>
1865      ZS:      .ASCII / DZ11'S      M7819   E81 M7814 E72 SWITCHES IN ON POSITION/<NOCRLF>
1866      KMCS:    .ASCII / KMC11'S     M8204   E65 M8206 SWITCHES IN OFF POSITION / <NOCRLF>
1867      PPS:     .ASCII / LPP11'S     ?????? *****/<NOCRLF>
1868      MV21:    .ASCII / VMV21'S     M7067   *****/<NOCRLF>
1869      MV31:    .ASCII / VMV31'S     M7068   *****/<NOCRLF>
1870      WRS:     .ASCII / DWR70'S     ?????? *****/<NOCRLF>
1871      RLS:     .ASCII / RL11'S      M7762   *****/<NOCRLF>
1872      LPAKS:   .ASCII / LPA11'S     ?????? *****/<NOCRLF>
1873      KWCS:    .ASCII / KW11-C'S    ?????? *****/<NOCRLF>
1874      RESRS:   .ASCII / RESERVED   ?????? *****/<NOCRLF>
1875      RXS:     .ASCII / RX11'S      M7846   *****/<NOCRLF>
1876      DRWS:    .ASCII / DR11-W'S    M7816   *****/<NOCRLF>
1877      DRBS:    .ASCII / DR11-B'S    M7219   *****/<NOCRLF>
1878      DMPS:    .ASCII / DMP11'S     M8207-YA *****/<NOCRLF>
1879      DPVS:    .ASCII / DPV11'S     M8020   *****/<NOCRLF>
1880      ISBS:    .ASCII / ISB11'S     M8204   *****/<NOCRLF>
1881      DMVS:    .ASCII / DMV11'S     M8053-MA M8064-MA */<NOCRLF>
1882      UNAS:    .ASCII / UNA'S       M7792   E40 SWITCHES IN OFF POSITION/<NOCRLF>
1883      UDAS:    .ASCII / UDA'S       ?????? *****/<NOCRLF>
1884      DMFS:    .ASCII / DMF32'S     M8396   *****/<NOCRLF>
1885      KMSS:    .ASCII / KMS11'S     M8206   MOD M8640 ***/<NOCRLF>
1886      VS100S:  .ASCII / VS100'S     ?????? *****/<NOCRLF>
1887      TU81S:   .ASCII / TU81'S      ?????? *****/<NOCRLF>
1888      KMSV:    .ASCII / KMV11'S     ?????? *****/<NOCRLF>
1889      DHVS:    .ASCII / DHV11 DHU11 ??? *****/<NOCRLF>
1890      DMZ32S:  .ASCII / DMZ32,CPI32 (ASYNCH) *****/<NOCRLF>
1891      CPI32S:  .ASCII / CPI32 (SYNCH) *****/<NOCRLF>
1892      QVSSS:   .ASCII / QVSS        ?????? *****/<NOCRLF>
1893      VS31S:   .ASCII / VS31        ?????? *****/<NOCRLF>
1894      .ENDR †
1895
1896

```

DATA TABLES

1898	011036				FATABL:		
1899	011036	040	104	112	JS:	.ASCII	/ DJ11'S /<NOCRLF>
1900	011052	040	104	110	HS:	.ASCII	/ DH11'S /<NOCRLF>
1901	011066	040	104	121	QS:	.ASCII	/ DQ11'S /<NOCRLF>
1902	011102	040	104	125	US:	.ASCII	/ DU11'S /<NOCRLF>
1903	011116	040	104	125	UPS:	.ASCII	/ DUP11'S /<NOCRLF>
1904	011132	040	114	113	LKS:	.ASCII	/ LK11'S /<NOCRLF>
1905	011146	040	104	115	MCS:	.ASCII	/ DMC11 /<NOCRLF>
1906	011162	040	104	132	ZS:	.ASCII	/ DZ11'S /<NOCRLF>
1907	011176	040	113	115	KMCS:	.ASCII	/ KMC11'S /<NOCRLF>
1908	011212	040	114	120	PPS:	.ASCII	/ LPP11'S /<NOCRLF>
1909	011226	040	126	115	MV21:	.ASCII	/ VMV21'S /<NOCRLF>
1910	011242	040	126	115	MV31:	.ASCII	/ VMV31'S /<NOCRLF>
1911	011256	040	104	127	WRS:	.ASCII	/ DWR70'S /<NOCRLF>
1912	011272	040	122	114	RLS:	.ASCII	/ RL11'S /<NOCRLF>
1913	011306	040	114	120	LPAKS:	.ASCII	/ LPA11'S /<NOCRLF>
1914	011322	040	113	127	KWCS:	.ASCII	/ KW11-C'S /<NOCRLF>
1915	011336	040	122	105	RESRS:	.ASCII	/ RESERVED /<NOCRLF>
1916	011352	040	122	130	RXS:	.ASCII	/ RX11'S /<NOCRLF>
1917	011366	040	104	122	DRWS:	.ASCII	/ DR11-W'S /<NOCRLF>
1918	011402	040	104	122	DRBS:	.ASCII	/ DR11-B'S /<NOCRLF>
1919	011416	040	104	115	DMPS:	.ASCII	/ DMP11'S /<NOCRLF>
1920	011432	040	104	120	DPVS:	.ASCII	/ DPV11'S /<NOCRLF>
1921	011446	040	111	123	ISBS:	.ASCII	/ ISB11'S /<NOCRLF>
1922	011462	040	104	115	DMVS:	.ASCII	/ DMV11'S /<NOCRLF>
1923	011476	040	125	116	UNAS:	.ASCII	/ UNA'S /<NOCRLF>
1924	011512	040	125	104	UDAS:	.ASCII	/ UDA'S /<NOCRLF>
1925	011526	040	104	115	DMFS:	.ASCII	/ DMF32'S /<NOCRLF>
1926	011542	040	113	115	KMSS:	.ASCII	/ KMS11'S /<NOCRLF>
1927	011556	040	126	123	VS100S:	.ASCII	/ VS100'S /<NOCRLF>
1928	011572	040	124	125	TU81S:	.ASCII	/ TU81'S /<NOCRLF>
1929	011606	040	113	115	KMVS:	.ASCII	/ KMV11'S /<NOCRLF>
1930	011622	040	104	110	DHVS:	.ASCII	/ DHV11 DHU11 /<NOCRLF>
1931	011640	040	104	115	DMZ32S:	.ASCII	/ DMZ32,CPI32 (ASYNCH) /<NOCRLF>
1932	011667	040	103	120	CPI32S:	.ASCII	/ CPI32 (SYNCH) /<NOCRLF>
1933	011712	040	121	126	QVSSS:	.ASCII	/ QVSS /<NOCRLF>
1934	011725	040	126	123	VS31S:	.ASCII	/ VS31 /<NOCRLF>
1935					.EVEN		

DATA TABLES

[illegible]

DATA TABLES

```

1984
1985
1986
1987
1988
1989 012300      104      103      061 Q1:  .ASCII /DC11'S                /<NOCRLF>
1990 012321      104      114      061 Q2:  .ASCII /DL11-A,B,W (TU58) DLV11-A,B,F /<NOCRLF>
1991 012361      104      120      061 Q3:  .ASCII /DP11'S                /<NOCRLF>
1992 012402      104      115      061 Q4:  .ASCII /DM11-A'S                /<NOCRLF>
1993 012423      104      116      061 Q5:  .ASCII /DN11-AA'S              /<NOCRLF>
1994 012444      104      115      061 Q6:  .ASCII /DM11-BB'S              /<NOCRLF>
1995 012465      104      122      061 Q7:  .ASCII /DR11-A,C,K'S DRV11      /<NOCRLF>
1996 012516      120      101      066 Q8:  .ASCII /PA611-R'S              /<NOCRLF>
1997 012537      120      101      066 Q9:  .ASCII /PA611-P'S              /<NOCRLF>
1998 012560      114      120      104 Q10: .ASCII /LPD11              /<NOCRLF>
1999 012601      104      124      061 Q11: .ASCII /DT11,07'S            /<NOCRLF>
2000 012622      104      130      061 Q12: .ASCII /DX11'S                /<NOCRLF>
2001 012643      104      114      061 Q13: .ASCII /DL11-C,D,E'S DLV11-E      /<NOCRLF>
2002 012674      104      112      061 Q14: .ASCII /DJ11'S                /<NOCRLF>
2003 012715      104      110      061 Q15: .ASCII /DH11'S                /<NOCRLF>
2004 012736      107      124      064 Q16: .ASCII /GT40                /<NOCRLF>
2005 012757      126      123      126 Q17: .ASCII /VSV11'S              /<NOCRLF>
2006 013000      114      120      123 Q18: .ASCII /LPS11              /<NOCRLF>
2007 013021      104      121      061 Q19: .ASCII /DQ11'S                /<NOCRLF>
2008 013042      113      127      061 Q20: .ASCII /KW11-W KWV11          /<NOCRLF>
2009 013064      104      125      061 Q21: .ASCII /DU11 DUV11 S          /<NOCRLF>
2010 013106      104      125      120 Q22: .ASCII /DUP11'S              /<NOCRLF>
2011 013127      104      126      061 Q23: .ASCII /DV11'S                /<NOCRLF>
2012 013150      114      113      061 Q24: .ASCII /LK11'S                /<NOCRLF>
2013 013171      104      127      125 Q25: .ASCII /DWUN S                /<NOCRLF>
2014 013212      104      115      103 Q26: .ASCII /DMC11 DMR11'S          /<NOCRLF>
2015 013235      104      132      061 Q27: .ASCII /DZ11 DZS11 DZV11 DZ32'S /<NOCRLF>
2016 013270      113      115      103 Q28: .ASCII /KMC11'S              /<NOCRLF>
2017 013311      114      120      120 Q29: .ASCII /LPP11'S              /<NOCRLF>
2018 013332      126      115      126 Q30: .ASCII /VMV21'S              /<NOCRLF>
2019 013353      126      115      126 Q31: .ASCII /VMV31'S              /<NOCRLF>
2020 013374      126      124      126 Q32: .ASCII /VTV01'S              /<NOCRLF>
2021 013415      104      127      122 Q33: .ASCII /DWR70'S              /<NOCRLF>
2022 013436      122      114      061 Q34: .ASCII /RL11 RLV11'S          /<NOCRLF>
2023 013460      124      123      061 Q35: .ASCII /TS11 TU80'S          /<NOCRLF>
2024 013501      114      120      101 Q36: .ASCII /LPA11'S              /<NOCRLF>
2025 013522      111      120      061 Q37: .ASCII /IP11 IP300'S          /<NOCRLF>
2026 013544      113      127      061 Q38: .ASCII /KW11-C S              /<NOCRLF>
2027 013565      122      105      123 Q39: .ASCII /RESERVED              /<NOCRLF>
2028 013606      122      130      061 Q40: .ASCII /RX11 RX211 RXV11 RXV21'S /<NOCRLF>
2029 013642      104      122      061 Q41: .ASCII /DR11-W S              /<NOCRLF>
2030 013663      104      122      061 Q42: .ASCII /DR11-B'S              /<NOCRLF>
2031 013704      104      115      120 Q43: .ASCII /DMP11'S              /<NOCRLF>
2032 013725      104      120      126 Q44: .ASCII /DPV11'S              /<NOCRLF>
2033 013746      115      114      061 Q45: .ASCII /ML11'S                /<NOCRLF>
2034 013767      111      123      102 Q46: .ASCII /ISB11 S              /<NOCRLF>
2035 014010      104      115      126 Q47: .ASCII /DMV11'S              /<NOCRLF>
2036 014031      125      116      101 Q48: .ASCII /UNA'S                /<NOCRLF>
2037 014052      125      104      101 Q49: .ASCII /UDA RQDX1'S          /<NOCRLF>
2038 014073      104      115      106 Q50: .ASCII /DMF32'S              /<NOCRLF>
2039 014114      113      115      123 Q51: .ASCII /KMS11'S              /<NOCRLF>
2040 014135      120      103      114 Q52: .ASCII /PCL11'S              /<NOCRLF>

```

SEQ 0041

DATA TABLES

2041	014156	126	123	061	Q53:	.ASCII	/VS100'S	/<NOCRLF>
2042	014177	124	125	070	Q54:	.ASCII	/TU81'S	/<NOCRLF>
2043	014220	113	115	126	Q55:	.ASCII	/KMW11'S	/<NOCRLF>
2044	014241	113	103	124	Q56:	.ASCII	/KCT32'S	/<NOCRLF>
2045	014262	111	105	130	Q57:	.ASCII	/IEX'S	/<NOCRLF>
2046	014303	104	110	126	Q58:	.ASCII	/DHV11 DHU11'S	/<NOCRLF>
2047	014325	104	115	132	Q59:	.ASCII	/DMZ32 CPI32 (ASYNCH)	/<NOCRLF>
2048	014355	103	120	111	Q60:	.ASCII	/CPI32 (SYNCH)	/<NOCRLF>
2049	014377	121	116	101	Q61:	.ASCII	/QNA'S	/<NOCRLF>
2050	014420	121	126	123	Q62:	.ASCII	/QVSS'S	/<NOCRLF>
2051	014441	126	123	063	Q63:	.ASCII	/VS31'S	/<NOCRLF>
2052	014462	114	116	126	Q64:	.ASCII	/LNV11'S	/<NOCRLF>

[illegible]

DATA TABLES

2058	014503	015	012	012	END:	.BYTE	15,12,12,12,0	
2059	014510	015	012	200	CRLF:	.BYTE	CR,LF,NOCRLF	
2060	014513	124	110	101	ILL:	.ASCIZ	/THAT'S NOT A VALID NUMBER! /	
2061	014546	124	110	101	TB:	.ASCII	/THAT'S TOO MANY! ONLY /	
2062	014575				MAXIUM:	.BLKB	2	
2063	014577	040	101	114		.ASCIZ	/ ALLOWED ! /	
2064	014612	015	012	120	OP:	.ASCII	<CR><LF>/POSSIBLE OPTIONS ARE/<CR><LF><NOCRLF>	;LWL002
2065	014643	011	106	101	OP2:	.ASCII	/ FA FLOATING ADDRESSES. (DJ,DH,DQ,ETC.)/<CR><LF>	
2066	014714	011	126	101		.ASCII	/ VA-FLOATING ADDRESSES AND VECTORS./<CR><LF>	;LWL002
2067	014761	011	110	105		.ASCII	/ HE-HELP/<CR><LF>	;LWL002
2068	014773	011	105	130		.ASCII	/ EX-EXIT/<CR><LF>	;LWL002
2069	015005	105	116	124		.ASCII	/ENTER OPTION: /<NOCRLF>	;LWL002
2070	015024	077	040	200	QMARK:	.ASCII	/? /<NOCRLF>	
2071	015027	015	012	012	HELLO:	.ASCII	<CR><LF><LF>/CZFLA FLOAT UTILITY PROG - /	;LWL002
2072	015065	126	105	122		.ASCII	/VERSION: C0 /<NOCRLF>	;LWL002
2073	015102	015	012	012	HELP:	.ASCII	<CR><LF><LF>/CONTROL C TO RESTART PROGRAM/<CR><LF>	;LWL002
2074	015143	103	117	116		.ASCII	/CONTROL Z TO PRINT SELECTED LIST/<CR><LF>	
2075	015205	103	117	116		.ASCII	/CONTROL S TO STOP PRINTING/<CR><LF>	
2076	015241	103	117	116		.ASCII	/CONTROL Q TO CONTINUE PRINTING/<CR><LF>	
2077	015301	103	117	116		.ASCII	/CONTROL A TO BACK UP DEVICE LIST/<CR><LF><LF><NOCRLF>	;LWL002
2078	015345	124	105	122	TERMQ:	.ASCII	/TERMINAL TYPES: /<CR><LF>	;LWL002
2079	015366	011	101	040		.ASCII	/ A = LA36 NO FILL/<CR><LF>	;LWL002
2080	015411	011	114	040		.ASCII	/ L = LA120 100 FILL CHARACTER/<CR><LF>	;LWL002
2081	015450	011	126	040		.ASCII	/ V = VT52 OR VT100 50 FILL CHARACTER/<CR><LF>	;LWL002
2082	015516	015	012	105		.ASCII	<CR><LF>/ENTER TERMINAL TYPE.(A,L,V)? /<NOCRLF>	
2083	015556	126	101	114	TERMQ2:	.ASCII	/VALID RESPONSES ARE-/<CR><LF><NOCRLF>	;LWL002
2084	015605	015	012	123	BADCON:	.ASCII	<CR><LF>/SORRY, BUT THE REST OF THE DEVICE(S) WILL NOT /<CR><LF>	
2085	015667	106	111	124		.ASCIZ	/FIT IN THE FLOATING VECTOR AREA!/<CR><LF><NOCRLF>	;LWL002

DATA TABLES

2087	015733	040	040	040	CSR:	.ASCII	/ CSR = /<NOCRLF>		;LWL002
2088	015746	040	104	105	CSRVEC:	.ASCII	/ DEVICE /		;LWL002
2089	015756	040	040	040		.ASCII	/ CSR VECT COMMENT/<CR><LF>		;LWL002
2090	016024	040	055	055		.ASCII	/ ----- /		;LWL002
2091	016034	040	040	040		.ASCII	/ --- ----		;LWL002
2092	016102	040	012	200		.ASCII	/ /<LF><NOCRLF>		;LWL002
2093	016105	040	040	040	TAB:	.ASCII	/ /<NOCRLF>		;LWL002
2094	016121	015	012	124	DVMES:	.ASCII	<CR><LF>/THE DV11 MODEM ADDRESS IS +20 , VECTOR IS +10/<NOCRLF>		;LWL002
2095	016201	015	012	061	RLMESS:	.ASCII	<CR><LF>/1ST CSR = 774400, 1ST VECT = 160/<NOCRLF>		
2096	016244	015	012	061	TSMESS:	.ASCII	<CR><LF>/1ST CSR = 772520, 1ST VECT = 224/<NOCRLF>		
2097	016307	015	012	061	LPKMES:	.ASCII	<CR><LF>/1ST CSR = 770460, /<NOCRLF>		
2098	016334	015	012	061	IPMESS:	.ASCII	<CR><LF>/1ST CSR = 771000, 1ST VECT = 234/<NOCRLF>		
2099	016377	015	012	116	REMES:	.ASCII	<CR><LF>/NOT USED/<NOCRLF>		
2100	016412	015	012	061	RXMES:	.ASCII	<CR><LF>/1ST CSR = 777170, 1ST VECT = 264/<NOCRLF>		
2101	016455	015	012	061	DRBMES:	.ASCII	<CR><LF>/1ST CSR = 772410, 1ST VECT = 124/<NOCRLF>		
2102	016520	015	012	061	UNAMES:	.ASCII	<CR><LF>/1ST CSR = 774510, 1ST VECT = 120/<NOCRLF>		
2103	016563	015	012	061	UDAMES:	.ASCII	<CR><LF>/1ST CSR = 772150, 1ST VECT = 154/<NOCRLF>		
2104	016626	015	012	061	TU81MS:	.ASCII	<CR><LF>/1ST CSR = 774500, 1ST VECT = 260/<NOCRLF>		
2105	016671	015	012	061	QVSSMS:	.ASCII	<CR><LF>/1ST CSR = 777200, /<NOCRLF>		
2106					:				
2107					:LWL002		FOLLOWING AREA MODIFIED TO FIT NEW FORMAT		;LWL002
2108					:				
2109	016716	015	012	040	DVMES:	.ASCII	<CR><LF>/		;LWL002
2110	016770	052	052	052		.ASCII	/***DM11-A COMES BEFORE DV11 /<CR><LF>		;LWL002
2111	017026	040	040	040		.ASCII	/		;LWL002
2112	017076	102	117	124		.ASCII	/BOTH WERE SELECTED/<NOCRLF>		;LWL002
2113	017121	015	012	040	DVMSS:	.ASCII	<CR><LF>/		;LWL002
2114	017173	052	052	052		.ASCII	/***THE DV11S HAVE BEEN DECREASED, /<CR><LF>		;LWL002
2115	017237	040	040	040		.ASCII	/		;LWL002
2116	017307	114	111	115		.ASCII	/LIMITED ADDRESS SPACE/<NOCRLF>		;LWL002
2117	017335	015	012	040	LPDMES:	.ASCII	<CR><LF>/		;LWL002
2118	017407	052	052	052		.ASCII	/***PA611 P COMES BEFORE LPD11 /<CR><LF>		;LWL002
2119	017447	040	040	040		.ASCII	/		;LWL002
2120	017517	102	117	124		.ASCII	/BOTH WERE SELECTED/<NOCRLF>		;LWL002
2121	017542	015	012	040	LPDESS:	.ASCII	<CR><LF>/		;LWL002
2122	017614	052	052	052		.ASCII	/***LPD11S HAVE BEEN DECREASED, /<CR><LF>		;LWL002
2123	017655	040	040	040		.ASCII	/		;LWL002
2124	017725	114	111	115		.ASCII	/LIMITED ADDRESS SPACE/<NOCRLF>		;LWL002
2125	017753	015	012	040	LPDMSS:	.ASCII	<CR><LF>/		;LWL002
2126	020025	052	052	052		.ASCII	/***THE PA611-P LEFT NO ROOM FOR /<CR><LF>		;LWL002
2127	020067	040	040	040		.ASCII	/		;LWL002
2128	020137	124	110	105		.ASCII	/THE LPD11/<NOCRLF>		;LWL002
2129	020151	015	012	040	DVNMS:	.ASCII	<CR><LF>/		;LWL002
2130	020223	052	052	052		.ASCII	/***THE DM11-A LEFT NO ROOM FOR /<CR><LF>	</	

DATA TABLES

2144	021154	124	110	105	.ASCII	/THERE IS NO STANDARD CONFIGURATION/<CR><LF>	
2145	021220	040	040	040	.ASCII	/	;LWL002
2146	021270	106	117	122	.ASCII	/FOR SYSTEMS WITH BOTH DC11 AND TU58/<NOCRLF>	
2147	021334	015	012	040	KMSMES: .ASCII	<CR><LF>/	;LWL002
2148	021406	052	052	052	.ASCII	/***IF IT HAS A M8640 ADDRESS IS /<CR><LF>	
2149	021450	040	040	040	.ASCII	/	;LWL002
2150	021520	053	061	060	.ASCII	/+10 VECTOR IS +10/<NOCRLF>	
2151							
2152					.EVEN		
2153	021542				BUFF:	;INPUT BUFFER!!!	
2154		000001			.END	;THAT'S IT FOLKS!!!	

Errors detected: 0

*** Assembler statistics

Work file reads: 0
 Work file writes: 0
 Size of work file: 8578 Words (34 Pages)
 Size of core pool: 19402 Words (74 Pages)
 Operating system: RSX-11M/PLUS (Under VAX/VMS)

Elapsed time: 00:00:59.72
 CZFLAC,CZFLAC/CR/-SP=CZFLAC

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES
A	012054	2-1170 2-1283 2-1299 #10-1970
ABUF	012062	2-1282 2-1292 #10-1971
ADDR	012034	*2-649 2-657 2-1498 *2-1500 2-1503 *2-1505 2-1506 *2-1508 2-1510
		*2-1512 2-1513 *2-1515 2-1516 *2-1518 #10-1957
ADDRT	007720	2-650 #6-1722
ASCDEV	010320	2-847 #6-1758
BADCON	015605	2-1268 #12-2084
BADD	006022	2-1504 #2-1506
BADDS	006076	2-1514 #2-1516
BADS	006060	2-1511 #2-1513
BEX	006002	2-1499 #2-1501
BEX1	006040	2-1507 #2-1509
BEX2	006114	2-1517 #2-1519
BITE	005764	2-751 #2-1498
BITE1	006004	2-755 #2-1503
BITE2	006042	2-759 #2-1510
BIT0	= 000001	#1-470
BIT1	= 000002	#1-471 2-1425 2-1427 2-1442 2-1444
BIT10	= 002000	#1-480
BIT11	= 004000	#1-481
BIT12	= 010000	#1-482
BIT13	= 020000	#1-483
BIT14	= 040000	#1-484
BIT15	= 100000	#1-485 2-831 2-1485
BIT2	= 000004	#1-472 2-1250
BIT3	= 000010	#1-473 2-1258
BIT4	= 000020	#1-474
BIT5	= 000040	#1-475 2-1393
BIT6	= 000100	#1-476
BIT7	= 000200	#1-477 2-612 2-620 2-1028 2 1386
BIT8	= 000400	#1-478
BIT9	= 001000	#1-479
BUFF	021542	2-1381 2-1436 2-1438 2-1467 #13 2153
CHAR	012140	*2-559 2-1562 #10-1973
CLEAN	004310	2-1069 2-1075 2-1086 2-1092 2-1098 2-1104 2-1112 2-1121 2-1127
		2-1132 2-1138 2-1144 #2-1153
CMP1	005650	#2-1469 2-1482
CPI32	011004	2-775 3-1632 4-1661 #7-1835
CPI32S	011667	6-1764 #9-1932
CR	= 000015	#1-466 2-1431 2-1470 10-1976 10-1976 10-1978 12-2059 12-2064 12-2064
		12-2065 12-2066 12-2067 12-2068 12-2071 12-2073 12-2073 12-2074 12-2075
		12-2076 12-2077 12-2078 12-2079 12-2080 12-2081 12-2082 12-2083 12-2084
		12-2084 12-2085 13-2089 13-2091 13-2094 13-2095 13-2096 13-2097 13-2098
		13-2099 13-2100 13-2101 13-2102 13-2103 13-2104 13-2105 13-2109 13-2110
		13-2113 13-2114 13-2117 13-2118 13-2121 13-2122 13-2125 13-2126 13-2129
		13-2130 13-2133 13-2134 13-2137 13-2138 13-2141 13-2142 13-2144 13-2147
		13-2148
CRLF	014510	2-642 2-643 2-1155 2-1162 2-1184 2-1421 2-1435 2-1566 #12-2059
CSR	015733	#13-2087
CSRVEC	015746	2-1027 #13-2088
CVTA	005176	2-1357 #2-1366
DATO	= 104001	#1-487 2-634

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES
DBUF	012070	2-1278 2-1279 *2-1280 2-1315 #10-1972
DC	010430	2-566 2-651 2-878 2-881 2-907 2-909 3-1621 #7-1776
DCHAR	012273	2-1454 #10-1979
DEVCNT	= 000100	2-565 2-652 #7-1847
DEVEND	= 011030	#7-1841 7-1847
DEVSTR	= 010430	#7-1774 7-1847
DH	010520	2-729 2-1324 3-1623 4-1655 #7-1790
DHV	010774	2-771 3-1632 4-1661 #7-1833
DHVS	011622	6-1764 #9-1930
DIG2	006176	2-1526 #2-1544
DJ	010514	2-877 2-878 2-1322 3-1623 4-1655 #7-1789
DL	010434	2-911 2-1030 3-1621 #7-1777
DLC	010510	3-1623 #7-1788
DLMESS	020772	2-1034 #13-2141
DMA	010444	2-917 2-919 2-924 2-1209 3-1621 #7-1779
DMBB	010454	2-1216 3-1622 #7-1781
DMC	010574	2-1340 3-1626 4-1656 #7-1801
DMF	010734	2-737 3-1630 4-1660 #7-1825
DMFS	011526	6-1763 #9-1925
DMP	010700	3-1629 4-1659 #7-1818
DMPS	011416	6-1762 #9-1919
DMV	010720	2-735 3-1630 4-1659 #7-1822
DMVS	011462	6-1762 #9-1922
DMZ32	011000	2-773 3-1632 4-1661 #7-1834
DMZ32S	011640	6-1764 #9-1931
DN	010450	2-1214 3-1621 #7-1780
DONE	002346	2-665 #2-844
DONE1	002510	2-879 #2-883 2-887 2-890 2-1193
DP	010440	3-1621 #7-1778
DPV	010704	3-1629 4-1659 #7-1819
DPVS	011432	6-1762 #9-1920
DQ	010540	2-1326 3-1624 4-1655 #7-1794
DR	010460	2-1145 2-1147 3-1622 #7-1782
DRB	010674	2-695 2-1105 *2-1107 2-1108 *2-1109 2-1228 3-1629 4-1658 #7-1817
DRBMES	016455	2-1110 #13-2101
DRBS	011402	6-1761 #9-1918
DRW	010670	2-1226 3-1629 4-1658 #7-1816
DRWS	011366	6-1761 #9-1917
DR#	004262	2-1140 #2-1145
DT	010500	3-1623 #7-1786
DT1	006562	2-614 2-1415 #3-1621
DT2	007072	2-622 2-658 2-1412 #4-1655
DU	010550	2-1336 3-1625 4-1655 #7-1796
DUP	010554	2-1338 3-1625 4-1655 #7-1797
DV	010560	*2-862 2-921 *2-934 2-935 *2-946 2-947 2-1046 3-1625 #7-1798
DVMES	016716	2-1051 #13-2109
DVMESS	016121	2-1048 #13-2094
DVMSS	017121	2-1054 #13-2113
DVNMS	020151	2-896 #13-2129
DWR	010630	3-1627 4-1657 #7-1808
DWUN	010570	*2-863 3-1625 #7-1800
DX	010504	3-1623 #7-1787

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES
DZ	010600	2-1342 3-1626 4-1656 #7-1802
EMTHND	002260	1-519 2-543 #2-808
EMTTAB	002276	2-812 #2-813
END	014503	2-897 #12-2058
ER	005724	#2-1485 2-1489
ERRCK	* 104002	#1-488 2-635
EXIT	005746	2-1471 #2-1490
FATABL	011036	#9-1898
FILCNT	006332	*2-560 #2-1572
FILL	006330	2-1563 #2-1571
FILLER	006334	#2-1573 *2-1577
FLAG	011740	*2-570 *2-612 *2-620 2-828 *2-831 2-845 2-1078 2-1410 2-1425
FLAG1	011742	*2-1427 2-1442 *2-1444 *2-1485 #10-1937
FLAG10	011764	*2-571 *2-923 2-1049 #10-1938
FLAG11	011766	*2-580 *2-1009 2-1040 #10-1947
FLAG12	011770	*2-581 *2-959 2-963 *2-967 2-971 *2-975 2-978 #10-1948
FLAG2	011744	*2-582 2-1025 *2-1028 #10-1949
FLAG3	011746	*2-572 *2-945 2-1052 #10-1939
FLAG4	011750	*2-573 2-894 *2-950 #10-1940
FLAG5	011752	*2-574 *2-995 2-1037 #10-1941
FLAG6	011754	*2-575 2-891 *2-1014 2-1043 #10-1942
FLAG7	011756	*2-576 *2-980 2-1058 #10-1943
FLAG8	011756	*2-577 *2-984 2-1061 #10-1944
FLAG9	011760	*2-578 *2-629 2-1406 #10-1945
GAP	011762	*2-579 *2-913 2-1032 #10-1946
GAP1	002250	2-788 2-790 2-794 2-798 #2-802
GAP3	002240	2-796 #2-799
GFA	002224	2-792 #2-795
GT	001460	#2-648
HEAD	010524	*2-858 2-1256 3-1624 #7-1791
HELLO	012155	#10-1976
HELP	015027	2-544 #12-2071
HS	015102	2-545 2-606 #12-2073
IEX	011052	6-1758 #9-1900
IL	010770	*2-870 3-1632 #7-1832
ILL	005716	2-1473 2-1475 #2-1484
INPUT	014513	2-1484 #12-2060
IP	* 104000	#1-486 2-548 2-588 2-633
IPMESS	010650	*2-866 2-1087 *2-1089 2-1222 3-1628 #7-1812
ISB	016334	2-1090 #13-2098
ISBS	010714	3-1630 4-1659 #7-1821
JS	011446	6-1762 #9-1921
JT	011036	6-1758 #9-1899
JUMPER	012000	2-1351 #10-1952
KCT32	005022	#2-1313
KMC	010764	*2-869 3-1632 #7-1831
KMCS	010604	2-1344 3-1626 4-1656 #7-1803
KMS	011176	6-1759 #9-1907
KMSMES	010740	2-741 2-1128 3-1631 4-1660 #7-1826
KMSS	021334	2-1130 #13-2147
KMV	011542	6-1763 #9-1926
	010760	2-769 3-1631 4-1661 #7-1830

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES
KMVS	011606	6-1764 #9-1929
KWC	010654	3-1628 4-1658 #7-1813
KWCS	011322	6-1761 #9-1914
KWW	010544	*2-861 3-1624 #7-1795
LF	= 000012	#1-468 10-1976 10-1976 10-1976 10-1978 10-1978 10-1978 12-2059 12-2064 12-2064
		12-2065 12-2066 12-2067 12-2068 12-2071 12-2071 12-2073 12-2073 12-2073
		12-2074 12-2075 12-2076 12-2077 12-2077 12-2078 12-2079 12-2080 12-2081
		12-2082 12-2083 12-2084 12-2084 12-2085 13-2089 13-2091 13-2092 13-2094
		13-2095 13-2096 13-2097 13-2098 13-2099 13-2100 13-2101 13-2102 13-2103
		13-2104 13-2105 13-2109 13-2110 13-2113 13-2114 13-2117 13-2118 13-2121
		13-2122 13-2125 13-2126 13-2129 13-2130 13-2133 13-2134 13-2137 13-2138
		13-2141 13-2142 13-2144 13-2147 13-2148
LK	010564	2-1329 3-1625 4-1656 #7-1799
LKS	011132	6-1759 #9-1904
LNV	011024	*2-872 2-1244 3-1633 #7-1839
LPAK	010644	2-690 2-733 2-1076 2-1080 *2-1081 *2-1083 3-1628 4-1657 #7-1811
LPAKS	011306	6-1760 #9-1913
LPD	010474	2-969 2-973 2-993 *2-1000 2-1001 *2-1010 2-1011 2-1035 3 1622
		#7-1785
LPDESS	017542	2-1042 #13-2121
LPDMES	017335	2-1039 2-1045 #13-2117
LPDMSS	017753	2-893 #13-2125
LPKMES	016307	2-1084 #13-2097
LPP	010610	3-1626 4-1656 #7-1804
LPS	010534	*2-860 3-1624 #7-1793
MAXIUM	014575	2-1525 #12-2062
MCS	011146	6-1759 #9-1905
ML	010710	*2-867 2-1113 2-1230 3-1629 #7-1820
MT1	007204	2-615 #5-1673
MT2	007406	2-623 #5-1688
MV21	011226	6-1760 #9-1909
MV31	011242	6-1760 #9-1910
NAT	007520	2-1182 #6-1701
NEXT	001554	2-664 #2-675
NOCRLF	= 000200	#1-467 2-1568 9-1899 9-1900 9-1901 9-1902 9-1903 9-1904 9-1905
		9-1906 9-1907 9-1908 9-1909 9-1910 9-1911 9-1912 9-1913 9 1914
		9-1915 9-1916 9-1917 9-1918 9-1919 9-1920 9-1921 9-1922 9 1923
		9-1924 9-1925 9-1926 9-1927 9-1928 9-1929 9-1930 9-1931 9-1932
		9-1933 9-1934 10-1977 11-1989 11-1990 11-1991 11-1992 11-1993 11-1994
		11-1995 11-1996 11-1997 11-1998 11-1999 11-2000 11-2001 11-2002 11-2003
		11-2004 11-2005 11-2006 11-2007 11-2008 11-2009 11-2010 11-2011 11-2012
		11-2013 11-2014 11-2015 11-2016 11-2017 11-2018 11-2019 11-2020 11-2021
		11-2022 11-2023 11-2024 11-2025 11-2026 11-2027 11-2028 11-2029 11-2030
		11-2031 11-2032 11-2033 11-2034 11-2035 11-2036 11-2037 11-2038 11-2039
		11-2040 11-2041 11-2042 11-2043 11-2044 11-2045 11-2046 11-2047 11-2048
		11-2049 11-2050 11-2051 11-2052 12-2059 12-2064 12-2069 12-2070 12-2072
		12-2077 12-2082 12-2083 12-2085 13-2087 13-2092 13-2093 13-2094 13-2095
		13-2096 13-2097 13-2098 13-2099 13-2100 13-2101 13-2102 13-2103 13-2104
		13-2105 13 2112 13-2116 13-2120 13-2124 13-2128 13-2132 13-2136 13-2140
		13-2146 13-2150
OFFSET	012032	#10-1956
OK	005756	2-1486 #2-1493

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES	CREF	V02
OP	014612	2-586 #12-2064		
OPTION	001230	#2-586 2-607		
OPTNA	001236	#2-587 2-598		
OP2	014643	2-587 #12-2065		
OTDA	= 104006	#1-492 2-1487		
OTOA	= 104004	#1-490 2-1164 2-1281		
PAP	010470	2-961 2-965 2-989	2-991	2-996 3-1622 #7-1784
PAR	010464	2-955 2-957 3-1622	#7-1783	
PCL	010744	*2-868 3-1631 #7-1827		
PPS	011212	6-1759 #9-1908		
PRI	004332	2-1055 2-1082 2-1115	2-1146	2-1152 2-1154 #2-1162
PRIN	004426	2-1157 2-1172 #2-1184		
PRINT	001444	#2-642 2-1405		
Q	012210	2-630 #10-1978		
QMARK	015024	2-632 #12-2070		
QNA	011010	*2-871 2-1240 3-1633	#7-1836	
QS	011066	6-1758 #9-1901		
QT1	006362	2-613 2-880 #3-1600		
QT2	006762	2-621 #4-1639		
QUES	001410	#2-631 2-636 2-641	2-1422	
QVSS	011014	2-718 2-777 2-1139	*2-1141	3-1633 4-1661 #7-1837
QVSSMS	016671	2-1142 #13-2105		
QVSSS	011712	6-1764 #9-1933		
QVSS\$	001722	2-714 2-716 #2-718		
QVS\$	002064	#2-759 2-778		
QV\$	004236	2-1134 #2-1139		
Q1	012300	3-1600 #11-1989		
Q10	012560	3-1601 #11-1998		
Q11	012601	3-1602 #11-1999		
Q12	012622	3-1602 #11-2000		
Q13	012643	3-1602 #11-2001		
Q14	012674	3-1602 4-1639 #11-2002		
Q15	012715	3-1602 4-1639 #11-2003		
Q16	012736	3-1603 #11-2004		
Q17	012757	3-1603 #11-2005		
Q18	013000	3-1603 #11-2006		
Q19	013021	3-1603 4-1639 #11-2007		
Q2	012321	3-1600 #11-1990		
Q20	013042	3-1603 #11-2008		
Q2	013064	3-1604 4-1639 #11-2009		
Q22	013106	3-1604 4-1639 #11-2010		
Q23	013127	3-1604 #11-2011		
Q24	013150	3-1604 4-1640 #11-2012		
Q25	013171	3-1604 #11-2013		
Q26	013212	3-1605 4-1640 #11-2014		
Q27	013235	3-1605 4-1640 #11-2015		
Q28	013270	3-1605 4-1640 #11-2016		
Q29	013311	3-1605 4-1640 #11-2017		
Q3	012361	3-1600 #11-1991		
Q30	013332	3-1605 4-1641 #11-2018		
Q31	013353	3-1606 4-1641 #11-2019		
Q32	013374	3-1606 #11-2020		

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES
Q33	013415	3-1606 4-1641 #11-2021
Q34	013436	3-1606 4-1641 #11-2022
Q35	013460	3-1606 #11-2023
Q36	013501	3-1607 4-1641 #11-2024
Q37	013522	3-1607 #11-2025
Q38	013544	3-1607 4-1642 #11-2026
Q39	013565	3-1607 4-1642 #11-2027
Q4	012402	3-1600 #11-1992
Q40	013606	3-1607 4-1642 #11-2028
Q41	013642	3-1608 4-1642 #11-2029
Q42	013663	3-1608 4-1642 #11-2030
Q43	013704	3-1608 4-1643 #11-2031
Q44	013725	3-1608 4-1643 #11-2032
Q45	013746	3-1608 #11-2033
Q46	013767	3-1609 4-1643 #11-2034
Q47	014010	3-1609 4-1643 #11-2035
Q48	014031	3-1609 4-1643 #11-2036
Q49	014052	3-1609 4-1644 #11-2037
Q5	012423	3-1600 #11-1993
Q50	014073	3-1609 4-1644 #11-2038
Q51	014114	3-1610 4-1644 #11-2039
Q52	014135	3-1610 #11-2040
Q53	014156	3-1610 4-1644 #11-2041
Q54	014177	3-1610 4-1644 #11-2042
Q55	014220	3-1610 4-1645 #11-2043
Q56	014241	3-1611 #11-2044
Q57	014262	3-1611 #11-2045
Q58	014303	3-1611 4-1645 #11-2046
Q59	014325	3-1611 4-1645 #11-2047
Q6	012444	3-1601 #11-1994
Q60	014355	3-1611 4-1645 #11-2048
Q61	014377	3-1612 #11-2049
Q62	014420	3-1612 4-1645 #11-2050
Q63	014441	3-1612 4-1646 #11-2051
Q64	014462	3-1612 #11-2052
Q7	012465	3-1601 #11-1995
Q8	012516	3-1601 #11-1996
Q9	012537	3-1601 #11-1997
REMES	016377	2-1096 #13-2099
RES	001120	2-542 #2-563 2-898 2-1269
RESR	010660	2-675 2-1093 #2-1095 3-1628 4-1658 #7-1814
RESRS	011336	6-1761 #9-1915 #2-1066 2-1218 3-1627 4-1657 #7-1809
RL	010634	2-680 2-1064
RLMESS	016201	2-1067 #13-2095
RLS	011272	6-1760 #9-1912
RUBOUT	005520	2-1424 #2-1438
RX	010664	2-685 2-1099 #2-1101 2-1224 3-1628 4-1658 #7-1815
RXMES	016412	2-1102 #13-2100
RXS	011352	6-1761 #9-1916
SAVE	012036	#10-1958
SAVE1	012040	#2-1080 2-1176 #10-1959
SAVE2	012042	#2-1108 2-1180 #10-1960

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES
SAVE3	012044	*2-924 *2-925 *2-926 *2-927 2-928 *2-930 2-931 *2-933 2-934
SAVE4	012046	2-942 #10-1961 *2-935 *2-936 *2-937 *2-938 *2-939 *2-940 *2-941 *2-942 2-943
SAVE5	012050	#10-1962 *2-996 *2-997 *2-998 *2-999 2-1000 2-1006 #10-1963
SAVE6	012052	*2-1001 *2-1002 *2-1003 *2-1004 *2-1005 *2-1006 2-1007 #10-1964
SDATO	005636	2-814 #2-1465
SERRCK	002324	2-815 #2-828
SHIFT	= 104005	#1-491 2-747
SHIFT1	= 104007	#1-493 2-753
SHIFT2	= 104010	#1-494 2-757
SHIFT3	= 104011	#1-495 2-761
SHIFT4	= 104012	#1-496 2-749
SINPUT	005230	2-813 #2-1381 2-1396 2-1399
SOTDA	006116	2-819 #2-1525
SOTOA	004752	2-817 *2-1278 *2-1279 *2-1282 *2-1283 #2-1292
SPEC	012142	*2-844 #10-1974
SSHIFT	002166	#2-784 2-818
SSHIF1	002164	#2-783 2-820
SSHIF2	002162	#2-782 2-821
SSHIF3	002160	#2-781 2-822
SSHIF4	002170	#2-785 2-823
ST	011774	2-1349 #10-1951
START	001004	#2-540 2-564 2-1402
STARTA	001000	1-526 1-527 #2-539
STYPE	006200	2-816 #2-1547 2-1548 2-1553 2-1569
SVCHA	011772	*2-1551 *2-1552 2-1553 *2-1557 *2-1558 2-1559 #10-1950
TAB	016105	2-1163 #13-2093
TAG	002102	2-743 #2-765
TAG1	002030	2-730 2-732 2-734 2-736 2-742 #2-751 2-766 2-770 2-772
TAG2	002046	2-738 #2-755 2-774 2-776
TAG3	001530	#2-659 2-748 2-750 2-754 2-758 2-762
TAG4	002014	#2-747 2-779
TB	014546	2-1488 #12-2061
TENDM	006304	#2-1564 2-1579
TERMQ	015345	2-546 #12-2078
TERMQ2	015556	2-556 2-597 #12-2083
TERMST	012144	2-547 2-554 #10-1975
TERMTF	012277	*2-558 2-1440 #10-1980
TOB	005734	2-1483 #2-1487 2-1492
TPB	000706	#1-533 2-1428 2-1433 2-1445 2-1448 2-1455 2-1561 2-1576
TPS	000704	#1-532 2-1429 2-1446 2-1449 2-1456 2-1547 2-1574
TRB	000702	#1-531 2-1382 2-1385 2-1551 2-1557
TRS	000700	#1-530 2-541 2-1383 2-1549 2-1555
TSMESS	016244	2-1073 #13-2096
TSI	005240	#2-1383 2-1384 2-1434 2-1439 2-1451 2-1461
TS11	010640	*2-865 2-1070 *2-1072 2-1220 3-1627 #7-1810
TU\$	002022	2-740 #2-749 2-768
TU81	010754	2-713 2-767 2-795 2-1133 *2-1135 2-1238 3-1631 4-1660 #7-1829
TU81MS	016626	2-1136 #13-2104
TU81S	011572	6-1763 #9-1928
TU81#	001706	2-709 2-711 #2-713

SYMBOL CROSS REFERENCE

CREF V02

SYMBOL	VALUE	REFERENCES
TYPE	= 104003	#1-489 2-544 2-545 2-546 2-556 2-586 2-587 2-597 2-606 2-630 2-631 2-632 2-642 2-643 2-893 2-896 2-897 2-1027 2-1029 2-1034 2-1039 2-1042 2-1045 2-1048 2-1051 2-1054 2-1060 2-1063 2-1068 2-1074 2-1085 2-1091 2-1097 2-1103 2-1111 2-1120 2-1126 2-1131 2-1137 2-1143 2-1155 2-1162 2-1163 2-1170 2-1184 2-1268 2-1421 2-1435 2-1484 2-1488 2-1566 2-708 2-739 2-791 2-1122 *2-1124 2-1234 3-1630 4-1660 #7-1824
UDA	010730	
UDAMES	016563	2-1125 #13-2103
UDAS	011512	6-1763 #9-1924
UDA#	001672	2-704 2-706 #2-708
UNA	010724	2-703 2-787 2-1116 *2-1118 2-1232 2-1346 3-1630 4-1659 #7-1823
UNAMES	016520	2-1119 #13-2102
UNAS	011476	6-1762 #9-1923
UNA#	004146	2-1117 #2-1122
UPS	011116	6-1758 #9-1903
US	011102	6-1758 #9-1902
V	= 000126	#1-469 2-1440
VECT	004444	2-1167 #2-1201
VECTOR	012030	*2-648 2-1250 *2-1252 2-1258 *2-1260 2-1265 *2-1284 #10-1955
VST	010120	2-1284 #6-1740
VSV	010530	*2-859 3-1624 #7-1792
VS100	010750	2-765 2-789 2-1236 3-1631 4-1660 #7-1828
VS100S	011556	6-1763 #9-1927
VS31	011020	2-1242 3-1633 4-1662 #7-1838
VS31S	011725	6-1765 #9-1934
V01	010624	*2-864 2-976 *2-981 2-982 *2-985 2-1056 3-1627 #7-1807
V01MES	020345	2-1060 #13-2133
V01MSS	020561	2-1063 #13-2137
V21	010614	3-1626 4-1657 #7-1805
V31	010620	2-731 3-1627 4-1657 #7-1806
WRS	011256	6-1760 #9-1911
XOFF	006240	#2-1555 2-1556 2-1560
XON	006272	2-1550 2-1554 #2-1561
ZS	011162	6-1759 #9-1906

MACRO CROSS REFERENCE

CREF V02

MACRO NAME

REFERENCES

ERCHK	01-499	2-635								
PRINT	01-504	2-544	2-545	2-546	2-556	2 586	2-587	2-597	2-606	2-630
	2-631	2-632	2-642	2-643	2-893	2-896	2-897	2-1027	2-1029	2-1034
	2-1039	2-1042	2-1045	2-1048	2-1051	2-1054	2-1060	2-1063	2-1155	2-1162
	2-1163	2-1170	2-1184	2-1268	2-1421	2-1435	2-1484	2-1488	2-1566	