



DECUS

PROGRAM LIBRARY

DECUS NO.	8-922
TITLE	IKT-TUB FOCAL '81
AUTHOR	Klaus Lickteig
COMPANY	Institut fur Kerntechnik, Technische Universitat Berlin, West-Germany
DATE	April 1981
SOURCE LANGUAGE	PAL-8/PAL-III

ATTENTION

This is a USER program. Other than requiring that it conform to submittal and review standards, no quality control has been imposed upon this program by DECUS.

The DECUS Program Library is a clearing house only; it does not generate or test programs. No warranty, express or implied, is made by the contributor, Digital Equipment Computer Users Society or Digital Equipment Corporation as to the accuracy or functioning of the program or related material, and no responsibility is assumed by these parties in connection therewith.

IKT-TUB FOCAL '81

Version: April 1981

Author: Klaus Lickteig, Institut für Kerntechnik, Technische
Universität Berlin, West-Germany

Operating System: OS/8

Source Language: PAL-8/PAL-III

Memory Required: Minimum 12k, Maximum 28k

Special Hardware Required: If available; AX08 A/D-Converter, KV8I
Display, KL3-JA Interface; AF04 A/D-Converter DK8-EP Real time
clock.

This overlay will control within PS/8-FOCAL, now IKT-TUB FOCAL
'80, the handling of data files ("FX" function), the AX08
A/D-converter ("FLAB", "FADC(K)" functions), the AF04
A/D-Converter ("FADC(K,F,R,S,)" function), the KV8I-display ("FD"
function) and the KL3-JA interface ("FIO" functions) and the
DK8-EP programmable real time clock ("FN(3,C)", "FN(4)"
functions).

Some new functions "FN " are added. The old display function
"FDIS" is available too.

The program will check the computer-hardware and will modify
itself.

Note: No sources are available on tape for overlay to PS/8 FOCAL
1971, though the listing is available in hard copy. Binary
tape is complete.

IKT-TUB FOCAL '81

Problem: With VT-101 Terminal at 9600 Band with smooth scroll and Auto X-on,
an X-off is transmitted to CPU and program locks in loop

Solution: Before running program, terminal receive buffer must be cleared,
jump scroll enabled and Auto X-on disabled

1. Set-up A
2. Reset
3. Set-up B
4. 1-1=0; 2-4=0
5. Set-up
6. R IKTFOC

TITLE: IKT-TUB FOCAL '81

AUTHOR: Klaus Lickteig

COMPANY: Institut für Kerntechnik
Technische Universität Berlin
Marchstr. 18, 1000 Berlin 10
Germany

DATE: April 1981

SOURCE LANGUAGE: PAL 8

/ IKT-TUE FOCAL (2. APRIL 1931)

1713-USE 22/02/31 PAGE 1

/ IKT-TUE FOCAL (2. APRIL 1931)

/ IKT-TUE FOCAL '31

/ *****

/ COMPUTER: DIGITAL EQUIPMENT CORP.
/ FDF-3/E OF FDF-3/I

/ ASSEMBLER: PAL3

/ MINIMUM HARDWARE TO RUN THIS PROGRAM:
/ FDF-3 WITH 12 K MEMORY
/ KEYBOARD READER, TELETYPE

/ MAXIMUM HARDWARE:
/ FDF-3 WITH 23 K MEMORY
/ KEYBOARD READER, TELETYPE
/ DECTAPE AND/OR CICH
/ HIGH-SPEED-READER AND -PUNCH
/ (ALL FDF-3 FOCAL HARDWARE)
/ EXTENDED ARITHMETIC ELEMENT
/ AK23 A/D-CONVERTER (LAB-3)
/ APC4 A/D-CONVERTER
/ KLS-JA INTERFACE
/ KVS1 DISPLAY
/ DK3-ET PROGRAMMABLE
/ REAL TIME CLOCK

/ THE DECUS BINARY TAPE OF IKT-TUE FOCAL '31
/ CONTAINS ALL NECESSARY SOFTWARE

/ THIS OVERLAY WILL CONTROL WITHIN FS/3-FOCAL
/ (NOW IKT-TUE FOCAL '31) THE HANDLING OF DATA
/ FILES ("FN"-FUNCTION), THE AX03 A/D-CONVERTER
/ ("FLAB"-, "FALCK")-FUNCTIONS, THE
/ AF04 A/D-CONVERTER ("FALCK,F,F,S")-FUNCTION,
/ THE K081-DISPLAY ("FE")-FUNCTION, THE
/ KL3-JA INTERFACE ("FIO"-FUNCTION) AND
/ THE LK3-BF PROGRAMMABLE REAL TIME CLOCK
/ ("FN(3,C)"-, "FN(4)"-FUNCTIONS).
/ SOME NEW FUNCTIONS "FN" ARE ADDED. THE OLD
/ DISPLAY FUNCTION "FEIS" IS AVAILABLE
/ TOO.
/
/ THE PROGRAM WILL CHECK THE COMPUTER-HARDWARE
/ AND WILL MODIFY ITSELF.

/ WRITTEN BY:
/ KLAUS LICHTIG
/ INSTITUT FÜR KERntechnik
/ TECHNISCHE UNIVERSITÄT BERLIN
/ MARCHSTRASSE 13
/ 1000 BERLIN 12

/ TO BUILD THIS OVERLAY, THE FOLLOWING SOFTWARE
/ WAS USED:

/ SCHNEIDER, D; SMITH, B.:
/ PS/3 FOCAL, 1971
/ DECUS PROGRAM LIBRARY FOCAL3-177
/
/ VAN ZEE, J.:
/ IMPROVED EAE ROUTINE FOR FOCAL
/ DECUS PROGRAM LIBRARY FOCAL3-233
/
/ MOSS, L.:
/ FX FUNCTION FOR RANDOM ACCESS FILES
/ DECUS PROGRAM LIBRARY FOCAL3-263
/
/ ERICKSON, M.B.:
/ OVERLAY FOR KV81 - OMSI FOCAL 1971
/ DECUS PROGRAM LIBRARY FOCAL3-312
/
/ LICKTEIG, K.:
/ FREQUENCY TRANSFORMATION PROGRAM
/ DECUS PROGRAM LIBRARY FOCAL3-226
/
/ LICKTEIG, K.:
/ ADDITIONAL FUNCTIONS TO CONTROL THE
/ AX08 A/D-CONVERTER
/ DECUSCOPE 14(1975)3, P. 61-67

/

START ADDRESS: 10200

[illegible]

/ 1) EAE ?

/ 2) AX08 ?

/ 3) AF04 ?

/ 4) KL3-JA ?

/ 5) KUBI ?

1 6) EK3-EF ?

/ 7) FILE COMMANDS ?

NO: DELETE FUNCTION "FX"

INTRODUCTION

PS/8 FOCAL, 1971 MAKES THE DEVICE-INDEPENDENT FILE HANDLING FACILITIES OF PS/8 DIRECTLY AVAILABLE TO FOCAL USERS. IT IS A COLLECTION OF PATCHES TO FOCAL, 1969 --- MOST PROGRAMS WRITTEN IN FOCAL, 1969 WILL BE PROCESSED BY PS/8 FOCAL, 1971 (SEE LIST OF DIFFERENCES ON PAGE 2).

PS/8 FOCAL, 1971 SOURCE AND BINARY FILES PLUS DEMONSTRATION PROGRAMS ARE AVAILABLE ON DECTAPE OR PAPER TAPE. ENCLOSE \$5 OR AN EXTRA DECTAPE TO COVER HANDLING AND MAILING. (ADD \$10 FOR DECTAPE VERSION IF YOU DO NOT INCLUDE YOUR OWN PS/8 FORMATTED DECTAPE.) SEND REQUESTS TO:

SOFTWARE DEVELOPMENT GROUP, OMSI, PORTLAND, OREGON 97221

SUGGESTIONS FOR NOVICES:

THIS MANUAL IS INTENDED AS A REVIEWING AID FOR PERSONS WHO ALREADY HAVE SOME FAMILIARITY WITH FOCAL. A GOOD INTRODUCTION TO FOCAL PROGRAMMING IS CHAPTER 11 OF 'PROGRAMMING LANGUAGES' (1970, #2); ALSO USEFUL IS THE PS/8 USER'S GUIDE (NO. DEC-08-MEFA-D, #3). BOTH MAY BE PURCHASED FROM THE DEC PROGRAM LIBRARY, BLDG. 3-5, MAYNARD, MASS. 01754.

TO USE PS/8 FOCAL, 1971 ON DECTAPE SYSTEMS (DISK USERS SEE PAGE 24):

- 1) BOOTSTRAP INTO PS/8 MONITOR;
- 2) AT MULTIPLE DECTAPE INSTALLATIONS, THE AUTHORS SUGGEST THAT THE PS/8 COMMAND 'ASSIGN OTA1: DSK:' BE GIVEN TO PERMIT THE USER TO USE HIS OWN DECTAPE AS THE DEFAULT DEVICE (DSK:); THIS IS REQUIRED BY THE 'HELP' PACKAGE AVAILABLE ON DECTAPE FOR PS/8 FOCAL, 1971;
- 3) TYPE 'RUN DSK: PFOCAL';
- 4) THE INITIAL DIALOGUE WILL ASK IF YOU WISH TO RETAIN STANDARD FEATURES OF PS/8 FOCAL, 1971. A 'YES' WILL PROVIDE 3500 CHARACTERS OF PROGRAM SPACE, 98 VARIABLES, 10 DIGIT PRECISION, FILES, AND TRIG FUNCTIONS. SEE 'INITIAL DIALOGUE' ON PAGE 22 FOR OTHER OPTIONS;
- 5) THE 'LIBRARY EXIT' COMMAND WILL RETURN YOU TO THE PS/8 MONITOR;
- 6) A 'HELP' PACKAGE FOR PS/8 FOCAL, 1971 IS AVAILABLE ON DECTAPE. MOUNT THE HELP TAPE ON DECTAPE DRIVE #1; MOUNT THE PS/8 SYSTEM TAPE ON DRIVE #0. YOU MAY ENTER 'HELP' WHILE WRITING A PS/8 FOCAL PROGRAM BY TYPING THE COMMAND 'L G HELP'. YOUR PROGRAM WILL BE SAVED UNDER THE NAME 'FOCAL.TM'. WHEN YOU DESIRE NO MORE HELP, YOU WILL BE RETURNED TO YOUR OWN PROGRAM. SEE ALSO THE HELP INDEX ON PAGE 27;
- 7) CTRL/P IS PS/8 FOCAL'S BREAK CHARACTER.

THIS MANUAL WAS TYPESET ON AN INEXPENSIVE MATRIX PRINTER MADE BY:

CENTRONICS DATA COMPUTER CORP.
ONE WALL STREET
HUDSON, NEW HAMPSHIRE 03051
TELEPHONE (603) 883-0111

***** CHANGES FROM FOCAL, 1969:

PROGRAMS WRITTEN IN FOCAL, 1969 SHOULD REQUIRE ONLY ONE CHANGE TO RUN UNDER PS/8 FOCAL, 1971:

THE INPUT DEVICE IS SWITCHED TO THE HIGH SPEED PAPER TAPE READER WITH THE 'OPEN INPUT PTR:' COMMAND AND SWITCHED BACK TO THE TERMINAL WITH 'OPEN INPUT TTY:, ECHO' INSTEAD OF WITH THE '*' COMMAND OF FOCAL, 1969.

OTHER CHANGES INCLUDE:

- 1) EXTENDED LIBRARY FEATURES INCLUDING DEVICE-INDEPENDENT CHAINING AND SUBROUTINE CALLS BETWEEN PROGRAMS.
- 2) FILE READING AND WRITING COMMANDS, 10 DIGIT PRECISION, OR STANDARD TRIG FUNCTIONS MAY BE DELETED BY THE INITIAL DIALOGUE TO ALLOW UP TO 222 VARIABLES WITH 5500 CHARACTER PROGRAMS; WITH ALL FEATURES RETAINED, 98 VARIABLES AND 3500 CHARACTERS REMAIN. SEE PAGE 22.
- 3) COMPUTED LINE NUMBERS; SEE PAGE 20.
- 4) TABULATION.
- 5) IMPROVED DISTRIBUTION AND 'RANDOM' INITIALIZATION OF 'FRAN()', THE PSEUDO-RANDOM NUMBER FUNCTION.
- 6) UNLIMITED LINE LENGTHS.
- 7) CHARACTER MANIPULATIONS HANDLED WITH 'FIN()' AND 'FOUT()'.
/ FSGN(0)=0 [FSGN(0)=1 IN FOCAL, 1969]
- 8) 'ERASE ALL' AND 'MODIFY' DO NOT ERASE VARIABLES.
- 9) IN ADDITION TO THE FDIS() DISPLAY FUNCTION FOR DEC'S 340 DISPLAY, VERSIONS ARE AVAILABLE WITH DISPLAY AND JOYSTICK FUNCTIONS FOR THE TEKTRONIX T-4002 GRAPHICS TERMINAL.
- 10) 3 SPECIAL VARIABLES ARE PROTECTED FROM THE 'ERASE' COMMAND --- '!', '"', AND '#'. '#' IS SET TO 1 BY THE INITIAL DIALOGUE IF ALL OPTIONAL FEATURES OF PS/8 FOCAL ARE DELETED AND TO 2310 IF STANDARD FEATURES ARE RETAINED; SEE 'INITIAL DIALOGUE', PAGE 22.
- 11) PS/8 FOCAL'S STARTING ADDRESS IS 10200 (FIELD 1).

CTRL/P IS THE BREAK CHARACTER

NOTE: PS/8 FOCAL DATA FILES ARE COMPATIBLE WITH EDIT AND TECO-8; HOWEVER, PS/8 FOCAL PROGRAM FILES ARE SAVED AS CORE IMAGES.

COMMANDS

*** DIRECT COMMANDS ARE GIVEN WHILE FOCAL IS IN COMMAND MODE. THEY ARE TYPED WITHOUT LINE NUMBERS AND FOCAL EXECUTES THEM AS SOON AS THE RETURN BUTTON IS HIT --- FOR EXAMPLE:

```
*TYPE 3+4, !      (FOCAL OUTPUTS THE VALUE OF 3+4)
  7.0000
```

*

(OR SEVERAL COMMANDS MAY BE GIVEN IN SEQUENCE:)

```
*SET X=3
*SET Y=2
*TYPE 3+X+Y, !
  8.0000
```

*

*** INDIRECT COMMANDS ARE USED FOR LONGER PROGRAMS. THEY ARE TYPED FOLLOWING LINE NUMBERS BETWEEN 1.01 AND 31.99 NOT INCLUDING INTEGERS AND MAY BE EXECUTED BY A DIRECT 'GO':

```
*2.1 SET X=3
*2.2 SET Y=2
*2.3 TYPE 3+X+Y, !
*GO
  8.0000
```

*

COMMAND FORMAT :

THE GENERAL FORM OF EACH IS GIVEN FOLLOWED BY EXAMPLES IN WHICH:

<> ENCLOSE REQUIRED TERMS. [] ENCLOSE OPTIONAL TERMS.
 () ENCLOSE COMMENTS. ONE LETTER ABBREVIATIONS MAY BE USED FOR COMMAND WORDS. X REPRESENTS A VARIABLE. E1, E2 AND E3 ARE ARITHMETIC EXPRESSIONS. L1, L2 AND L3 ARE LINE NUMBERS. G1 IS EITHER A LINE OR GROUP NUMBER. L1, L2, L3 AND G1 MAY BE REPLACED WITH ANY ARITHMETIC EXPRESSION NOT BEGINNING WITH THE LETTER 'A'.

MOST COMMANDS MUST BE FOLLOWED BY A SPACE.

ELEMENTARY COMMANDS

ASK

*ASK ["PRINT OUT IN QUOTES"] X[,Y,Z] (INPUT VALUES FOR VARIABLES)

ASK Y (EXPECTS A VALUE TO BE INPUT FOR Y)
 A Z (A IS THE ABBREVIATION FOR 'ASK')
 A BZ,X,NUMBER (EXPECTS THREE VALUES TO BE INPUT)
 A "HOW MANY?"M (TYPES 'HOW MANY' THEN SETS M EQUAL TO THE RESPONSE)

* * * SPACE, COMMA OR CARRIAGE RETURN WILL TERMINATE EACH NUMBER INPUT. TYPE A '←' (BACK ARROW) TO ERASE A PARTIALLY ENTERED NUMBER WHICH MAY THEN BE RE-ENTERED. ALT MODE WILL CAUSE FOCAL TO LEAVE THE VARIABLE WITH ITS FORMER VALUE. IF 'ASKING' FROM A FILE --- A SPACE, COMMA OR OTHER DELIMITER MUST PRECEDE A NEGATIVE SIGN, '-', OR THE NUMBER WILL BE READ AS POSITIVE. RESPONDING TO INITIAL DIALOGUE WITH 'YES' OR '4,5' WILL REMEDY THIS BY OUTPUTTING A LEADING SPACE WHEN USING 'TYPE'.

COMMENT

*COMMENT (LINES BEGINNING WITH A 'C' WILL BE IGNORED)
 COMMENT SQRT (COMMENTS ARE USED TO DESCRIBE PROGRAMS)
 C (ABBREVIATION)
 CONTINUE (DUMMY LINE)

DO

*DO [LINE OR GROUP NUMBER] (SUBROUTINE CALL)
 DO 7.24 (TELLS FOCAL TO EXECUTE LINE 7.24 THEN RETURN
 D 4.48 TO THE COMMAND FOLLOWING THE 'DO')
 DO 7 (EXECUTES GROUP 7)
 DO ALL (STARTS EXECUTION OF PROGRAM AT FIRST LINE)
 D G1 D 7.6 (EXECUTES LINE OR GROUP G1 THEN LINE 7.6)

* * * DO NOT USE AN ARITHMETIC EXPRESSION STARTING WITH 'A' AS A COMPUTED LINE NUMBER OR IT WILL 'DO ALL'.

ERASE

*ERASE [LINE OR GROUP NUMBER] (ZEROES VARIABLES IF NO NUMBER GIVEN)
 ERASE 2.17 (DELETES LINE 2.17 FROM INDIRECT PROGRAM)
 E 3 (DELETES ALL GROUP 3 LINES)
 ERASE (ZEROES ALL VARIABLES)
 ERASE ALL (ERASES ENTIRE INDIRECT PROGRAM)

* * * 'ERASE' MUST BE THE LAST COMMAND ON A LINE. 'ERASE <LINE OR GROUP NUMBER>' WILL CAUSE FOCAL TO RETURN TO COMMAND MODE AFTER ERASING.

FOR

*FOR X=E1[, E2], E3] [COMMAND TO BE REPEATED (E3-E1)/E2+1 TIMES]

FOR P=1, 7; TYPE 3 (WILL TYPE A '3' SEVEN TIMES)
 FOR J=1, 5; DO 17. 1 (WILL DO LINE 17. 1 FIVE TIMES)
 F X=2, 2, 8; T X*10 (WILL TYPE '20 40 60 80')
 FOR Z=1, Y; DO 7 (WILL DO GROUP 7, Y TIMES)

* * * THE 'FOR' COMMAND IS REPEATED --- ADDING 1 (OR E2) TO E1 EACH CYCLE UNTIL E1 EXCEEDS E3. EXECUTION THEN GOES TO THE NEXT LINE.

GO

*GO [LINE NUMBER] (STARTS EXECUTING PROGRAM AT DESIGNATED LINE)

GO (STARTS EXECUTING INDIRECT PROGRAM AT FIRST LINE)
 G (ABBREVIATION)
 GOTO 3. 1 (STARTS EXECUTING PROGRAM AT LINE 3. 1)
 GO ? ('TRACE FEATURE' -- PRINTS PROGRAM DURING EXECUTION
 UNTIL NEXT OCCURRENCE OF '?')
 G X (STARTS EXECUTING PROGRAM AT LINE X)

IF

*IF (E1)L1[, L2, L3] (GOES TO LINE L1, L2, OR L3 DEPENDING ON WHETHER E1 IS -, 0, OR +)

IF (X)2. 7; 1. 2; 3. 8 (FOCAL GOES TO LINE 2. 7, 1. 2 OR 3. 8 IF X IS -, 0 OR +)
 I (H-5)1. 3; 1. 5; 1. 7 ('IF' CHECKS THE VALUE INSIDE THE PARENTHESES)
 I (B-3)2. 8; 7. 9; Q ('QUITS' EXECUTION IF (B-3) IS POSITIVE)
 I (Z)2. 7; T 21 (IF Z IS NEGATIVE THEN DO TO 2. 7; ELSE 'TYPE 21')
 I (G)2. 1; Y; 2. 4 (IF G=0 THEN GOES TO LINE Y)

MODIFY

*MODIFY <LINE NUMBER> (EDIT LINE)

'MODIFY 3. 72' FOLLOWED BY A CARRIAGE RETURN AND A CHARACTER 'X' WILL OUTPUT LINE 3. 72 THROUGH ITS FIRST 'X'. YOU MAY THEN:

- 1) INPUT THE REST OF THE REVISED LINE AND HIT 'RETURN'
- 2) USE 'RUB OUT' TO ERASE SINGLE CHARACTERS
- 3) HIT 'LINE FEED' TO KEEP REMAINDER OF OLD LINE
- 4) HIT CTRL/L TO GO TO NEXT OCCURRENCE OF 'X'
- 5) HIT CTRL/BELL TO CHANGE SEARCH CHARACTER; THEN HIT NEW CHARACTER
- 6) HIT '←' TO KILL LINE OVER TO LEFT MARGIN

QUIT

*QUIT (TERMINATES PROGRAM EXECUTION)

Q (ABBREVIATION)

RETURN

*RETURN (CAUSES ESCAPE FROM A SUBROUTINE TO THE COMMAND FOLLOWING THE CALLING 'DO' OR 'LIBRARY GOSUB')

R (ABBREVIATION)

SET

*SET Y=<NUMBER, VARIABLE OR EXPRESSION> (SETS VARIABLE VALUES)

SET Y=37 (CAUSES 'Y' TO TAKE THE VALUE 37)
 S A=110/P+32 (CAUSES 'A' TO ASSUME THE VALUE OF 110/P+32)
 S Y=Y+1 (SETS THE NEW VALUE OF Y EQUAL TO 1 PLUS OLD VALUE)
 S Z=# (SETS Z EQUAL TO VALUE OF PROTECTED VARIABLE '#')
 S #=0 (ONLY WAY TO ZERO PROTECTED VARIABLE '#')
 S FIN(<X>) (SETS X TO DECIMAL ASCII CODE OF NEXT CHARACTER INPUT)
 S DUMMY=FOUT(203) (OUTPUTS LETTER 'K' AND SETS DU=203)

TYPE

*TYPE [NUMBERS, VARIABLES, EXPRESSIONS, TEXT IN QUOTES, !, #, ., \$ OR %]

TYPE 3 (OUTPUTS A 3 ON THE TERMINAL OR OTHER OUTPUT DEVICE)
 T X (OUTPUTS THE VALUE OF X)
 T X/Y-1 (OUTPUTS THE VALUE OF THE EXPRESSION X/Y-1)
 T "HI THERE" (OUTPUTS THE GREETING 'HI THERE')
 T ! (OUTPUTS A CARRIAGE RETURN/LINE FEED)
 T "A"!"B"!"C"! (OUTPUTS CHARACTERS 'A', 'B' AND 'C' VERTICALLY)
 T # (OUTPUTS A CARRIAGE RETURN WITHOUT LINE FEED)
 T (#) (OUTPUTS VALUE OF SYSTEM STATUS VARIABLE, '#')
 T "ANS: ", 26+5 (OUTPUTS 'ANS: 31')
 T !:8, "*" (OUTPUTS RETURN/LINE FEED THEN A '#' IN POSITION 8)
 T * (OUTPUTS THE SYMBOL TABLE)
 T .:0. 07 (SETS THE OUTPUT FORMAT TO ALLOW UP TO 10 DIGITS TO BE PRINTED, OF WHICH 7 MAY BE DECIMAL PLACES)
 T % (FORMATS OUTPUT IN POWERS-OF-TEN NOTATION)

* * * THE INITIAL DIALOGUE'S STANDARD FEATURES OUTPUTS A LEADING SPACE BEFORE EACH NUMBER IS 'TYPED' --- THIS PERMITS NEGATIVE NUMBERS TO BE READ AS NEGATIVE FROM A FILE WRITTEN IN THIS MANNER. FEATURE 6 PERMITS OUTPUTTING A '#' AS IN FOCAL, 1969 (NOT RECOMMENDED FOR WRITING FILES).

WRITE

*WRITE [GROUP OR LINE NUMBER] (LISTS INDIRECT PROGRAM)

WRITE 2. 27 (OUTPUTS LINE 2. 27 OF THE INDIRECT PROGRAM)
 W 2 (OUTPUTS GROUP 2)
 WRITE ALL (OUTPUTS ENTIRE INDIRECT PROGRAM)
 W A (ABBREVIATION FOR 'WRITE ALL')
 W (ABBREVIATION FOR 'WRITE ALL')
 W G1 (OUTPUTS GROUP OR LINE GIVEN BY 'G1')

* * * DO NOT USE AN ARITHMETIC EXPRESSION STARTING WITH 'A' AS A COMPUTED LINE NUMBER OR IT WILL 'WRITE ALL'. 'WRITE' MUST BE THE LAST COMMAND ON A LINE.

LIBRARY COMMANDS

PS/8 DEVICE NAMES:

SYS: SYSTEM DEVICE (DSK: IN DISK SYSTEM; DTA0: IN DECTAPE SYSTEM)
 DSK: THE DISK IN DISK SYSTEMS; DECTAPE #0 IN DECTAPE SYSTEMS
 (THE AUTHORS SUGGEST THAT DECTAPE SYSTEMS 'ASSIGN DTA1:DSK:')
 ('DSK:' IS ASSUMED IF A DEVICE IS NOT SPECIFIED)
 DTA0: - DTA7: DECTAPE DRIVES
 LTA0: - LTA7: LINCTAPE DRIVES
 MTA0: - MTA7: MAGTAPE DRIVES
 PTR: HIGH SPEED READER
 PTP: HIGH SPEED PUNCH
 LPT: LINE PRINTER
 TTY: TERMINAL

REFERENCING A NON-EXISTENT DEVICE MAY CAUSE THE COMPUTER TO ENTER AN
 INFINITE LOOP WAITING FOR THE DEVICE. RECOVERY IS MADE BY USING THE
 SWITCH REGISTER --- LOAD ADD 10200 AND START.

LIBRARY CALL

*LIBRARY CALL [DEVICE:]KPROGRAM NAME) (LOADS SAVED PROGRAM)

LIBRARY CALL CHISQR (LOADS PROGRAM 'CHISQR.FC' FROM DSK:)
 L C DTA3:PROGRAM (LOADS 'PROGRAM.FC' FROM DECTAPE DRIVE #3)
 L C TEST1 (LOADS 'TEST1.FC' FROM DSK:)

*** FOCAL RETURNS TO COMMAND MODE FOLLOWING A 'LIBRARY CALL'.

LIBRARY DELETE

*LIBRARY DELETE [DEVICE:]KPROGRAM NAME) (DELETES FILE FROM DEVICE)

LIBRARY DELETE TTEST (DELETES PROGRAM 'TTEST.FC' FROM DSK:)
 L D DTA7:PROG (DELETES 'PROG.FC' FROM DECTAPE DRIVE #7)
 L D HOLD.FD (DELETES DATA FILE 'HOLD.FD')

*** LIBRARY DELETE ASSUMES '.FC' FILE NAME EXTENSION; OTHERS MUST
 BE SPECIFIED. 'LIBRARY DELETE' CLOSSES ANY OPEN OUTPUT FILES.

LIBRARY GOSUB

*LIBRARY GOSUB [DEVICE:]KPROGRAM NAME) [GROUP OR LINE NUMBER]

LIBRARY GOSUB TEXT 13.7 (LINE 13.7 OF 'TEXT.FC' BECOMES SUBROUTINE WHICH
 RETURNS TO THE COMMAND FOLLOWING 'GOSUB')
 L G SUMSQR (TREATS WHOLE 'SUMSQR.FC' PROGRAM AS SUBROUTINE)
 L G DTA3:X.Y 7 (TREATS GROUP 7 OF 'X.Y' LIKE A 'DO' SUBROUTINE)
 L G BOBO G1 ('GOSUB' TO LINE OR GROUP EQUAL TO VALUE OF G1
 DO NOT USE EXPRESSIONS BEGINNING WITH 'A'.)

*** IF THE VERSION OF A PROGRAM ISSUING A 'GOSUB' IS NOT SAVED, ANY
 OPEN OUTPUT FILE IS CLOSED AND THE PROGRAM IS SAVED AS 'FOCAL.TM' BEFORE
 EXECUTION OF THE 'GOSUB'.

LIBRARY LIST

*LIBRARY LIST [DEVICE:] [FIRST FILE NAME TO BE LISTED]
 (LISTS PROGRAM AND DATA FILE NAMES WITH FOCAL
 EXTENSIONS '.FC' =PROGRAM; '.FD' =DATA FILES)

LIBRARY LIST (LISTS FOCAL .FC AND .FD FILES SAVED ON DSK:)
 L L (ABBREVIATION)
 LIBRARY LIST DTA3: (LISTS FILES SAVED ON DECTAPE DRIVE #3)
 L L DTA6:TEST (STARTS LISTING WITH 'TEST.FC')

* * * FILE LENGTH IS PRINTED IN BLOCKS; 1 BLOCK IS ABOUT 500 CHARACTERS.

LIBRARY RUN

*LIBRARY RUN [DEVICE:] [PROGRAM NAME] [LINE NUMBER]
 (BEGINS EXECUTION OF NAMED PROGRAM)

LIBRARY RUN JOHN (LOADS 'JOHN.FC' THEN BEGINS PROGRAM EXECUTION)
 L R DTA2:ZONK (RUNS 'ZONK.FC' FROM DECTAPE DRIVE #2)
 L R POP 22.81 (STARTS EXECUTING 'POP.FC' AT LINE 22.81)
 L R ZOWIE E1 (STARTS EXECUTING 'ZOWIE.FC' AT LINE E1)

* * * PROGRAMS CONTAINING 'RUNS' SHOULD BE SAVED BEFORE EXECUTION.

LIBRARY SAVE

*LIBRARY SAVE [DEVICE:] [PROGRAM NAME]

LIBRARY SAVE PROG (SAVES INDIRECT PROGRAM PRESENTLY IN THE TEXT
 BUFFER AREA UNDER THE NAME 'PROG.FC' ON DSK:)
 L S DTA6:ZAAP (SAVES 'ZAAP.FC' ON DECTAPE DRIVE #6)
 L S PRGNAM (SAVES 'PRGNAM.FC' ON DSK:)

* * * OLD 'PRGNAM' IS DELETED WHEN NEW 'PRGNAM' IS SAVED. THIS
 COMMAND CLOSES OPEN OUTPUT FILES.

LIBRARY EXIT

*LIBRARY EXIT (LEAVES FOCAL; RETURNS TO PS/8 MONITOR)
 L E (ABBREVIATION)

FILE COMMANDS

THE EXPERIENCED PROGRAMMER MAY READ AND WRITE PS/8 COMPATIBLE DATA FILES WITH MANY DEVICES. SEE PS/8 DEVICE NAMES ON PAGE 7.

PS/8 FOCAL ASSUMES .FC AND .FD AS NAME EXTENSIONS FOR PROGRAM AND DATA FILES RESPECTIVELY. DATA FILES ARE SAVED IN STANDARD PS/8 ASCII FORMAT AND ARE COMPATIBLE WITH EDIT AND TECO-8. PROGRAM FILES ARE SAVED IN CORE IMAGE FORMAT AND MAY BE TRANSFERRED BY PIP ONLY WITH THE '/I' OPTION. TO PRODUCE AN ASCII FILE CONTAINING A FOCAL PROGRAM, OPEN AN OUTPUT FILE; 'WRITE ALL' THEN 'OUTPUT CLOSE'.

THE TERMINAL (TTY:) IS PS/8 FOCAL'S NORMAL INPUT AND OUTPUT DEVICE. THE 'OPEN INPUT' COMMAND PERMITS SELECTING ANOTHER DEVICE FOR INPUT (DEVICES LISTED ON PAGE 7). 'OPEN OUTPUT' DOES THE SAME FOR OUTPUT. BE SURE TO CLOSE ONE OUTPUT FILE BEFORE OPENING ANOTHER.

'ECHO' ADDED TO THE INPUT COMMANDS ECHOES THE INPUT DATA ON THE OUTPUT DEVICE. WHEN ADDED TO OUTPUT COMMANDS, 'ECHO' CAUSES THE OUTPUT TO BE ECHOED ON THE TERMINAL.

'OUTPUT CLOSE' EMPTIES THE OUTPUT BUFFER ONTO THE DEVICE AND SAVES THE FILE IF THE DEVICE IS FILE STRUCTURED (DISK OR MAGNETIC TAPE).

THE 'OPEN INPUT TTY:,ECHO' COMMAND WILL RESTORE THE TERMINAL AS THE INPUT DEVICE WITH ECHO. AFTER AN ERROR OCCURS OR 'CTRL/P' IS STRUCK, THE TERMINAL IS RESTORED AS BOTH INPUT AND OUTPUT DEVICE.

READING AND WRITING CAN BE RESUMED WITH PREVIOUSLY OPENED NON-TTY: DEVICES THROUGH 'OPEN RESTORE INPUT' AND 'OPEN RESTORE OUTPUT' COMMANDS.

DIRECT FILE COMMANDS MAY BE EXECUTED AS MENTIONED UNDER 'MERGING PROGRAMS' OR A SERIES OF DIRECT COMMANDS MAY BE EXECUTED AUTOMATICALLY FROM A FILE. FOR EXAMPLE, IF YOU WISHED TO GET LISTINGS OF SEVERAL PROGRAMS, YOU COULD PREPARE THE FILE 'LISTAL.FD' USING EDIT OR TECO-8:

```
TYPE !!!!!"LISTING OF TEST1-TEST3:"!!!!
LIBRARY CALL TEST1
WRITE ALL
TYPE !!!!!
LIBRARY CALL TEST2
WRITE ALL
TYPE !!!!!
LIBRARY CALL TEST3
WRITE ALL
TYPE !!!!!
OPEN INPUT TTY:,ECHO
```

THEN, BY GIVING THE DIRECT FOCAL COMMAND 'OPEN INPUT LISTAL', ALL 3 PROGRAMS WOULD BE OUTPUT WITHOUT YOUR INTERVENTION.

NOTE: WHEN WRITING FOCAL DATA FILES, IT IS NECESSARY TO INCLUDE A SPACE, COMMA, CARRIAGE RETURN OR OTHER DELIMITER PRECEDING A MINUS SIGN, OTHERWISE THE NUMBER WILL APPEAR POSITIVE WHEN 'ASKED'. A PRECEDING SPACE WILL BE AUTOMATICALLY 'TYPED' IF THE INITIAL DIALOGUE RECEIVES AN ANSWER OF 'YES' OR '4,5'.

OPEN INPUT

*OPEN INPUT [DEVICE:] [FILE NAME] [ECHO] (PREPARES A FILE TO BE READ WITH 'ASK' OR 'FIN')
 OPEN INPUT BLOOP (OPENS 'BLOOP.FD' FILE FROM DSK:)
 O I DTA1:RED,ECHO (OPENS 'RED.FD' FILE FROM DECTAPE DRIVE #4 AND WILL ECHO ON OUTPUT DEVICE WHILE READING 'RED')
 O I PTR: (PS/8 WILL TYPE AN 'X' ON THE TERMINAL; HIT THE SPACE BAR AFTER PAPER TAPE IS LOADED IN READER.)
 OPEN INPUT TTY:,ECHO (RESTORES TERMINAL TO NORMAL STATUS)

* * * CTRL/Z IS THE LAST CHARACTER IN A FOCAL DATA FILE. ATTEMPTS TO READ PAST THE END-OF-FILE WILL CAUSE A '?' TO BE OUTPUT ON THE TERMINAL AND WILL RESTORE THE TERMINAL AS INPUT DEVICE.

OPEN OUTPUT

*OPEN OUTPUT [DEVICE:] [FILE NAME] [ECHO] (PREPARES A FILE TO BE WRITTEN WITH 'TYPE', 'FOUT(A)' OR ECHOING FROM INPUT FILE)
 OPEN OUTPUT ABC (OPENS 'ABC.FD' TO BE WRITTEN ON THE DSK:)
 O O DTA7:ZZ (OPENS 'ZZ.FD' FILE ON DECTAPE DRIVE #7)
 O O PTP:,ECHO (ECHOES ON TTY: WHILE OUTPUTTING TO PUNCH; BE SURE TO GIVE AN 'OUTPUT CLOSE' WHEN DONE)

* * * OPEN OUTPUT FILES WILL BE CLOSED IF THE COMMAND 'LIBRARY SAVE' OR 'LIBRARY DELETE' IS GIVEN OR IF 'LIBRARY GOSUB' IS GIVEN BY A VERSION OF A PROGRAM THAT HAS NOT BEEN SAVED.

OPEN RESTORE INPUT

*OFF RESTORE INPUT [ECHO] (RESUMES INPUT FROM FILE)
 OPEN RESTORE INPUT (RESUMES INPUTTING DATA FROM A PREVIOUSLY OPENED INPUT FILE AFTER USING THE TTY: WITH AN 'OPEN INPUT TTY:,ECHO')
 O R I (SAME)
 O R I,E (SAME PLUS ECHO)

OPEN RESTORE OUTPUT

*OPEN RESTORE OUTPUT [ECHO] (RESUMES OUTPUT TO FILE)
 OPEN RESTORE OUTPUT (RESUMES OUTPUTTING ON PREVIOUSLY OPENED OUTPUT DEVICE AFTER USING TTY:)
 O R O (SAME)
 O R O,ECHO (SAME PLUS ECHO)

OUTPUT CLOSE

*OUTPUT CLOSE (ENDS FILE WRITING AND SAVES OUTPUT FILE IF DEVICE IS FILE STRUCTURED --- DISK OR MAGNETIC TAPE)
 O C (ABBREVIATION)

FILE AND CHARACTER HANDLING PROGRAMS

"WORDS"

GROUP 1 SETS UP THE "WORDS" FILE. GROUP 2 SCANS IT, ECHOING ONLY THE WORDS WHICH START WITH THE SEARCH LETTER. OBSERVE THE PROGRAM LISTING AND EXECUTION; ALSO NOTE COMPUTED LINE NUMBER (P) USED IN LINE 2.60 TO MINIMIZE TIME IN THE CHARACTER INPUT LOOP:

```
*WRITE ALL
C-PS/8 FOCAL, 1971
```

```
01.05 COMMENT: "WORDS" 7/3/71 BY J. HUISMAN OF OMSI
01.10 OPEN OUTPUT DTA2:WORDS; OPEN OUTPUT TTY; ERASE
01.20 TYPE !"ENTER WORDS; FOLLOW EACH WITH A SPACE:"
01.30 TYPE "(ENTER A SPACE BY ITSELF TO CLOSE FILE)!!"
01.40 TYPE :5"? " ; OPEN RESTORE OUTPUT, ECHO
01.50 IF (-FABS(FIN()-160))1.5; OPEN OUTPUT TTY:
01.60 TYPE !"MORE? " ; OPEN RESTORE OUTPUT, ECHO
01.70 IF (-FABS(FIN()-160))1.5; OUTPUT CLOSE; GOTO 2.7

02.10 TYPE !"ENTER SEARCH LETTER: " ; SET L=FIN(); T !!!
02.20 TYPE "WORDS STARTING WITH "; S D=FOUT(L); T " : " : 5
02.30 OPEN INPUT DTA2:WORDS; SET S=160; SET P=2.6
02.40 SET C=FIN(); IF (FABS(C-S))2.7; I (-FABS(L-C))2.6
02.50 OPEN RESTORE INPUT, ECHO; T : 5; SET DUMMY=FOUT(C)
02.60 I (-FABS(FIN()-S))P; OPEN RESTORE INPUT; G 2.4
02.70 OPEN INPUT TTY; ECHO; TYPE !"DONE. !!!" ; QUIT
*GO
```

```
ENTER WORDS; FOLLOW EACH WITH A SPACE:
(ENTER A SPACE BY ITSELF TO CLOSE FILE)
```

```
? KINCAID
MORE? BUTLER
MORE? CRAWFORD
MORE? KAY
MORE? POLLARD
MORE? KLATT
MORE?
```

```
DONE.
```

```
*GOTO 2.1
```

```
ENTER SEARCH LETTER: K
```

```
WORDS STARTING WITH K:
```

```
KINCAID
KAY
KLATT
```

```
DONE.
```

```
*
```

'ALPHA'

'ALPHA' ALPHABETIZES A 'WORDS.FD' FILE PREPARED BY THE 'WORDS.FC' PROGRAM ON PAGE 11.

NOTE THAT THESE PROGRAMS ARE EXAMPLES ONLY AND NOT RECOMMENDED FOR GENERAL USE SINCE ERROR CHECKING THAT WOULD MAKE THEM USER-ORIENTED WAS EXCLUDED FOR THE SAKE OF BREVITY.

*WRITE ALL

C-PS/8 FOCAL, 1971

```
01. 05 COMMENT: 'ALPHA' 8/1/71 BY B. SMITH OF OMSI
01. 10 OPEN INPUT DTA2:WORDS
01. 15 OPEN OUTPUT DTA2:WORDS, ECHO; ERASE
01. 20 DO 10; IF (LE), 3, 3
01. 25 SET LE(1)=LE; FOR J=1, LE; SET B(J)=A(J); SET A(J)=0
01. 30 DO 10; IF (LE), 3, 1
01. 40 IF (LE-LE(1))>1.45; SET N=LE; GOTO 1. 5
01. 45 SET N=LE(1)
01. 50 FOR I=1, N; DO 2
01. 60 GOTO 1. 3
```

```
02. 10 IF (A(I)-B(I))>2.4; 2, 3
02. 15 FOR J=1, LE(1); SET D=FOUT(B(J)); SET B(J)=0
02. 20 TYPE " "; SET I=N+1; DO 1. 25; RETURN
02. 30 IF (I-N); DO 2. 4; TYPE " "; RETURN
02. 40 FOR J=1, LE; SET D=FOUT(A(J)); SET A(J)=0
02. 50 SET I=N+1; SET CHANGE=CHANGE+1; TYPE " "
```

```
03. 10 FOR I=1, LE(1); SET D=FOUT(B(I))
03. 20 TYPE " "
03. 30 OUTPUT CLOSE; TYPE !; IF (CHANGE), 3, 4,
03. 40 OPEN INPUT TTY; ECHO; TYPE !, "DONE. ", !!; QUIT
```

```
10. 10 SET LE=0
10. 20 SET A(LE+1)=FIN(); IF (A(LE+1)-154)>10. 3,
10. 30 IF (A(LE+1)-160)>10. 4, 10. 5
10. 40 SET LE=LE+1; GOTO 10. 2
10. 50 SET A(LE+1)=0
```

*GO

```
791 AND QUITE 124 123 ALL NUMERICS ORDERED ALPHABETICS USEFULLY
791 AND 124 123 ALL NUMERICS ORDERED ALPHABETICS QUITE USEFULLY
791 124 AND 123 ALL NUMERICS ORDERED ALPHABETICS QUITE USEFULLY
124 791 123 ALL AND NUMERICS ALPHABETICS ORDERED QUITE USEFULLY
124 123 791 ALL AND ALPHABETICS NUMERICS ORDERED QUITE USEFULLY
123 124 791 ALL ALPHABETICS AND NUMERICS ORDERED QUITE USEFULLY
123 124 791 ALL ALPHABETICS AND NUMERICS ORDERED QUITE USEFULLY
```

DONE.

*

SYMBOLS

ARITHMETIC SYMBOLS:

EXPONENTIATION
TYPE 3~2
SET X=Y~Z
TYPE 2^3.5

[^] (INDICATES AN INTEGER EXPONENT)
(OUTPUTS A '9' --- 3 TO THE SECOND POWER)
(SETS X EQUAL TO Y RAISED TO THE INTEGRAL Z POWER)
(PRINTS AN '8' --- THE .5 IS IGNORED)

MULTIPLICATION
TYPE 2*4
SET X=X*Z

[*]
(OUTPUTS AN '8')
(SETS THE NEW VALUE OF X EQUAL TO X TIMES Z)

DIVISION
TYPE 6/2
SET X=Y/Z

(OUTPUTS A '3')
(SETS X EQUAL TO Y DIVIDED BY Z)

ADDITION
TYPE 2+2
SET X=Y+Z

[+]
(OUTPUTS A '4')
(SETS X EQUAL TO Y PLUS Z)

SUBTRACTION
TYPE 3-X
SET X=Y-Z

[-]
(OUTPUTS THE VALUE OF 3-X)
(SETS X EQUAL TO Y MINUS Z)

ENCLOSURES

(<), [], AND <> MAY BE USED IN MATCHED PAIRS
TO ENCLOSE QUANTITIES WHICH ARE TO BE OPERATED
ON AS A UNIT.

SPECIAL CHARACTERS:

RETURN/LINE FEED [!]
 TYPE !!!!!X,!! (OUTPUTS 4 CARRIAGE RETURN/LINE FEEDS, THE VALUE OF
 X, AND THEN 2 MORE CARRIAGE RETURN/LINE FEEDS.

CARRIAGE RETURN [#]
 SEE WHAT THE '#' DOES TO LINE 10.35 FIRST WRITTEN THEN EXECUTED:
 (NOTE THAT THE '/' AND THE '=' ARE SUPERIMPOSED)
 10.35 TYPE "X=5",#," /",!!!!
 X=5

TAB [:]
 TAB OUTPUTS ENOUGH SPACES SO THAT THE NEXT CHARACTER WILL BE OUTPUT
 AT THE POSITION ON THE LINE INDICATED BY THE VALUE AFTER THE ':'.
 TAB WILL HAVE NO EFFECT IF THE POSITION SPECIFIED HAS BEEN EXCEEDED.
 SEE WHAT THE ':' DOES TO LINE 11.37 FIRST WRITTEN THEN EXECUTED:
 11.37 TYPE "NAME---":25,"ROGER":40,"AGE = ",2*13,!!!!
 NAME--- ROGER AGE = 26.0000

OUTPUT FORMATTER [%]
 SEE WHAT THE '%' DOES TO LINE 12.30 FIRST WRITTEN THEN EXECUTED:
 12.30 SET X=123.456;TYPE %6.04 X,!!;TYPE % 4.02 X,!!;T % 3.0 X,!!;T % X,!
 123.456
 123.5
 .4
 0.1234560000E+03
 '%4.02' PERMITS FOUR DIGITS TO BE OUTPUT INCLUDING UP TO TWO DECIMAL
 PLACES; 'TYPE %' ALONE GIVES POWERS-OF-TEN NOTATION. '%' DOES NOT
 CHANGE CALCULATION PRECISION; THAT IS DONE WITH THE INITIAL DIALOGUE.

SYMBOL TABLE [\$]
 TYPE \$ (OUTPUTS THE LIST OF DEFINED VARIABLES NOT INCLUDING
 PROTECTED VARIABLES. BE SURE TO USE AN OUTPUT
 FORMAT (%) THAT WILL NOT MASK PRECISION. SUB-
 SCRIPTS OVER 99 WILL NOT OUTPUT ACCURATELY.)

QUOTATION MARKS ["]
 WORDS MAY BE TYPED BY ENCLOSING THEM IN QUOTES.
 SEE WHAT THE " DOES TO LINE 14.65 FIRST WRITTEN THEN EXECUTED:
 14.65 TYPE "THE ANSWER IS ",3+8,!!!!
 THE ANSWER IS 11.0000

TRACE [?]
 THE FIRST TIME FOCAL READS A '?' (EXCEPT WITHIN QUOTES) IT WILL
 START OUTPUTTING THE PROGRAM WHILE IT IS BEING EXECUTED. THE NEXT
 TIME IT MEETS A '?' IT WILL STOP OUTPUTTING THE PROGRAM.
 SEE WHAT THE '?' DOES TO LINE 15. 60 FIRST WRITTEN THEN EXECUTED:
 15. 60 SET X=3; ? SET Y=5; TYPE X/Y; !?
 SET Y=5; TYPE X/Y, 0. 6000!
 A 'GO ?' COMMAND WILL CAUSE THE WHOLE PROGRAM TO BE 'TRACED'.

AN ATTEMPT TO READ PAST A CTRL/Z END-OF-FILE CHARACTER WILL OUTPUT
 A '?' ON THE TERMINAL AND RESTORE THE TERMINAL AS THE INPUT DEVICE.

POWERS-OF-TEN [E OR &]
 'E' MAY BE CHANGED TO '&' IN THE INITIAL DIALOGUE.
 SEE WHAT THE 'E' DOES TO LINE 6. 80 FIRST WRITTEN THEN EXECUTED:
 06. 80 TYPE 1E1, 7E3, 1. 23E-2, 3. 76E217, !!!!!
 10. 0000 7000. 00 0. 0123 0. 376000E+218

RUBOUT KEY
 WHEN WRITING A PROGRAM, ONE CHARACTER WILL BE DELETED FROM THE CURREN
 LINE WHEN THE RUBOUT KEY IS STRUCK; RUBOUT WILL ECHO AS A '\ ' --- THUS
 'PLWEA\\EASE' BECOMES 'PLEASE'.

CTRL/C (PRESSING 'CTRL' AND 'C' AT THE SAME TIME)
 WILL RETURN USER TO THE PS/8 MONITOR WHEN FOCAL IS USING PS/8
 ROUTINES --- THE 'LIBRARY EXIT' COMMAND IS PREFERRED FOR
 LEAVING PS/8 FOCAL. 1971. CTRL/P IS THE BREAK CHARACTER.

CTRL/G [BELL] (PRESSING 'CTRL' AND 'G' AT THE SAME TIME)
 USED WITH THE 'MODIFY' COMMAND TO CHANGE THE SEARCH CHARACTER.

CTRL/L (PRESSING 'CTRL' AND 'L' AT THE SAME TIME)
 USED WITH THE 'MODIFY' COMMAND TO SKIP TO THE NEXT SEARCH CHARACTER.

CTRL/P (PRESSING 'CTRL' AND 'P' AT THE SAME TIME)
 IS PS/8 FOCAL'S BREAK CHARACTER --- INTERRUPTS PROGRAM EXECUTION AND
 CAUSES FOCAL TO PRINT ITS '*'.
 *

CTRL/Z (PRESSING 'CTRL' AND 'Z' AT THE SAME TIME)
 IS THE LAST CHARACTER IN A PS/8 FOCAL DATA FILE. ATTEMPTS TO READ
 PAST THE END-OF-FILE WILL CAUSE A '?' TO BE TYPED ON THE TERMINAL AND
 WILL RESTORE THE TERMINAL AS THE INPUT DEVICE.

LINE FEED

USE WITH THE 'MODIFY' COMMAND TO RETAIN REMAINDER OF MODIFIED LINE.

KILL LINE

[←]

WHEN WRITING A PROGRAM, '←' DELETES EVERYTHING OVER TO LEFT MARGIN.
IN RESPONSE TO 'ASK' COMMAND, '←' CANCELS THE NUMBER BEING ENTERED.

RETURN KEY

USED WITH THE 'MODIFY' COMMAND TO DELETE REMAINDER OF MODIFIED LINE.
'RETURN' IS A LEGAL SYMBOL FOR SEPARATING DATA BEING REQUESTED BY
AN 'ASK' COMMAND.

ALT MODE KEY

IN RESPONSE TO AN 'ASK' COMMAND, ALT MODE CAUSES THE VARIABLE
TO RETAIN ITS PREVIOUS VALUE.

SPACE KEY

IT IS A LEGAL SYMBOL FOR SEPARATING DATA REQUESTED BY AN 'ASK'.

COMMA

[,]

IT IS A LEGAL SYMBOL FOR SEPARATING DATA IN 'TYPE' AND 'ASK' COMMANDS.
SEE WHAT THE ',' DOES TO LINE 26. 40 FIRST WRITTEN THEN EXECUTED:

26. 40 TYPE 2, 3, 5/6, 12~3, !!!!!
2. 0000 3. 0000 0. 9333 1728. 00

SEMICOLON

[;]

SEPARATES COMMANDS WHEN PLACED TOGETHER ON ONE LINE.

SEE WHAT THE ';' DOES TO LINE 27. 42 FIRST WRITTEN THEN EXECUTED:

27. 42 SET X=17; TYPE 3*X; SET X=3/7; TYPE " ", X+3, !!!!!
51. 0000 3. 4286

FUNCTIONS

FOR MOST FUNCTIONS, A SAMPLE LINE IS SHOWN AND THEN EXECUTED:
 '(A)' REPRESENTS THE ARGUMENT OF THE FUNCTIONS

NOTE: ARCTAN, LOG AND EXP FUNCTIONS ARE PRECISE TO LESS THAN 7 DIGITS.
 ARCTAN'S PRECISION IS MUCH REDUCED FOR ARGUMENTS NEAR ZERO.
 SINE AND COSINE ARE PRECISE TO 10 DIGITS.

FABS(A) ABSOLUTE VALUE
 EXAMPLE (WRITTEN THEN EXECUTED):
 03. 40 TYPE FABS(-3), FABS(2), !!!!
 3. 0000 2. 0000

FADC(A) ANALOG TO DIGITAL INPUT FUNCTION
 04. 22 SET X=FADC(A), TYPE !!!!

FATN(A) ARCTANGENT ARGUMENT RANGE: $0 < A < 10^{-6}$
 05. 25 TYPE FATN(1), !!!!; COM: OUTPUTS RADIAN ANGLE WHOSE TANGENT = 1. 00
 0. 7854

----- TANGENT 'FTAN(A)' IS NOT AVAILABLE
 TANGENT IS COMPUTED AS THE QUOTIENT (SINE/COSINE)
 05. 55 TYPE FSIN(1)/FCOS(1), !!!!; COMMENT: OUTPUTS TANGENT OF 1 RADIAN
 1. 5574

FCOS(A) COSINE ARGUMENT RANGE: $-1 \times 10^{-4} < A < 10^{-4}$ (RADIAN)
 06. 27 TYPE FCOS(1), FCOS(3/2), !!!!; COM: OUTPUTS COSINES OF 1 & 1.5 RADIAN
 0. 5403 0. 0707

FDIS(A) DISPLAY FUNCTION FOR DEC TYPE 340 DISPLAY
 EXAMPLE:
 07. 20 SET DUMMY=FDIS(X, Y)
 WOULD CAUSE A POINT TO BE PLOTTED AT POSITION (X, Y) ON DISPLAY

FEXP(A) EXPONENTIAL
 08. 24 TYPE FEXP(1), FEXP(2.17), !!!!; COM: NATURAL BASE TO THE POWER (A)
 2. 7183 8. 7583

FITR(A) INTEGER

09.18 TYPE FITR(3/2), FITR(23.719), FITR(-2.99), !!!!

1.0000 23.0000- 2.0000

WARNING: LACK OF PRECISION IN FOCAL 1969'S FLOATING POINT PACKAGE
MAY CAUSE FITR(A) TO PRODUCE ROUNDING ERRORS WITH NUMBERS OF
MORE THAN FOUR OR FIVE DIGITS. YOU MAY PROGRAM AROUND THIS
PROBLEM BY ADDING EMPIRICAL CONSTANTS TO THE FUNCTION ARGUMENT.

FLOG(A) NATURAL LOGARITHM

10.14 TYPE FLOG(1), FLOG(4.237), FLOG(10), !!!!

0.0000 1.4439 2.3026

FRN () RANDOM NUMBER

11.22 TYPE FRAN(), FRAN(), FRAN(), FRAN(), FRAN(), !

0.5444 0.7732 0.3680 0.8852 0.8446

THE PSEUDO-RANDOM NUMBERS PRODUCED ARE PART OF A VERY LONG AND WELL
DISTRIBUTED BUT DETERMINISTIC SERIES. YOU WILL USUALLY OBSERVE AN
ENTIRELY DIFFERENT SERIES EACH TIME FOCAL IS INITIALIZED. HOWEVER,
FRAN() DOES NOT GENERATE TRULY RANDOM NUMBERS.

FSGN(A) SIGN

12.34 T %1.0 FSGN(2.78), " ", FSGN(-299), " ", FSGN(0), " ", FSGN(-1), !!!!

1 -1 0 -1

FSIN(A) SINE ARGUMENT RANGE: $-1 \times 10^{-4} < A < 10^{-4}$ (RADIAN)

13.52 TYPE FSIN(1), FSIN(0), FSIN(37-2.22), !!!!

0.8415 0.0000- 0.2207

FSQT(A) SQUARE ROOT OF A POSITIVE NUMBER

14.40 TYPE FSQT(4), FSQT(391), FSQT(.0038953), !!!!

2.0000 19.7737 0.0624

FIN() CHARACTER INPUT FUNCTION

ASKS FOR A CHARACTER FROM THE TERMINAL (OR FILE); EXAMPLE:

09.14 SET A=FIN()

SETS 'A' EQUAL TO THE DECIMAL ASCII CODE OF THE CHARACTER INPUT.

FOUT(A) CHARACTER OUTPUT FUNCTION

09.25 SET DUMMY=FOUT(A)

OUTPUTS THE CHARACTER WHOSE DECIMAL ASCII CODE EQUALS 'A'.

IF AN 'L' WERE INPUT IN RESPONSE TO LINE 9.14, 'A' WOULD BE SET
EQUAL TO 204. THEN LINE 9.25 WOULD CAUSE AN 'L' TO BE OUTPUT.
THE 'A' ON LINE 9.25 IS A DUMMY VARIABLE. SEE DECIMAL ASCII CODES
FOR FIN() AND FOUT(A) ON PAGE 28.

/ III. NEW COMMANDS

/ SCOPE COMMANDS:

/ * WORKS ONLY IN SCOPE MODE

/ CTRL/A- HOME(UPPER LEFT HAND CORNER)
/ CTPL/B- * LINE FEED
/ CTPL/E- * CAUSES END OF PAGE CONDITION
/ CTPL/H- * BACK SPACE
/ CTPL/N- * UP SPACE
/ CTPL/O- * CARRIAGE RETURN
/ CTRL/Q- INCREMENT SIZE
/ CTPL/R- RESET SIZE
/ CTPL/S- CHANGE TO SCOPE MODE
/ CTPL/T- CHANGE TO TTY MODE
/ CTPL/U- ERASE SCREEN & RETURN HOME
/ CTPL/X- ERASE SCREEN

/ +A - FOUT(01) - HOME
/ +B - FOUT(02) - * LINE FEED
/ +E - FOUT(05) - * END OF PAGE
/ +H - FOUT(08) - * BACK SPACE
/ +N - FOUT(14) - * UP SPACE
/ +O - FOUT(15) - * CARRIAGE RETURN
/ +Q - FOUT(17) - INCREMENT SIZE
/ +R - FOUT(18) - RESET SIZE
/ +S - FOUT(19) - SCOPE MODE
/ +T - FOUT(20) - TTY MODE
/ +U - FOUT(21) - ERASE & HOME
/ +X - FOUT(24) - ERASE

/ IV. NEW FOCAL COMMANDS

```
/ FX      BUILD AND HANDLE DATA FILES IN A
/          RANDOM FASHION
/
/          FX(1,A)      STORE THE VALUE "A"
/          FX(1)        RETRIEVE THE VALUE

/ FLAB    CONTROL THE AX29 A/D-CONVERTER, THE
/          CLOCKS, THE OUTPUTS AND INPUTS OF
/          THE LAB-3 SYSTEM
/
/          FLAB(0)      CLEAR FLACS, REGISTERS
/          FLAB(1,K1)    PLOT WITH X-Y-RECORDER
/          FLAB(2,K1)    PULSE AT RELAY-OUTPUTS
/                          S0, R1, R2, R3
/          FLAB(3,K1,K2) SWITCH RELAY-OUTPUTS
/                          S0, R1, R2, R3 ON/OFF
/          FLAB(4,K1,K2) SWITCH RC- OP EXTERNAL-
/                          CLOCK ON/OFF
/          FLAB(5)       WAIT FOR RC- OP EXTERNAL-
/                          CLOCK FLAG
/          FLAB(6,K1,K2) WAIT FOR K2 PULSES OF
/                          THE SPECIFIED CLOCK
/          FLAB(7,K1)    SYNCHRONIZE RC- OP
/                          EXTERNAL-CLOCK
/          FLAB(8,K1)    READ/CLEAR EXTERNAL
/                          SENSE REGISTER
/
/          FLAB(9)       FREE
/          FLAB(10)      FREE
```

```

/ FADC CONTROL THE AX03 AND/OR AF04
/ A/D-CONVERTER
/
/ FADC(K) AX03 A/D-CONVERSION
/ FADC(K, F, P, S) AF04 A/D-CONVERSION

```

```

/ FN NEW FUNCTIONS
/
/ FN(0, U, NU) ROUTINE TO SERVE A FFT
/ FN(1, I) GET DATA FROM BUFFER
/ IN FIELD 2
/ FN(1, I, A) STORE DATA INTO BUFFER
/ IN FIELD 2
/ FN(2, I) GET DATA FROM BUFFER
/ IN FIELD 3 FF
/ FN(2, I, A) STORE DATA INTO BUFFER
/ IN FIELD 3 FF
/ FN(3, C) START PROGRAMMABLE
/ REAL TIME CLOCK
/ FN(4) READ NUMBER OF CLOCK
/ FLAGS
/
/ FN(5) FREE
/ FN(6) FREE
/ FN(7) FREE
/ FN(8) FREE
/ FN(9) FREE
/ FN(10) FREE

```

```

/ FD SCOPE FUNCTION FOR THE KVB1-DISPLAY
/
/ FD(I, X, Y) DISPLAY A POINT,
/ VECTOR OR CIRCLE

```

```

/ FIO  RECEIVE OR TRANSMIT DATA VIA THE
/      KL3-JA INTERFACE
/
/      FIO(-1,1)  CHANGE OUTPUT FOR
/                  TRANSMIT DATA
/      FIO(2,1)   GET A RECEIVED DATA
/                  INTO THE FL.P.AC.
/      FIO(1,1)   RECEIVE DATA
/
/      FIO(2,K1)   FREE
/      FIO(3,K1)   FREE
/      FIO(4,K1)   FREE

```



```

/      FUNCTION          FX
/      *****

```

/ TAPE - OVERLAY TO PS/3 FOCAL.

```

/  SUBROUTINE BY LAWRENCE MOSS
/  FX FUNCTION FOR RANDOM ACCESS FILES
/  DECUS PROGRAM LIBRARY FOCAL3-203

```

```

/  THE FUNCTION FX IS USED AS FOLLOWS:
/  TO STORE A VALUE INTO THE BUFFER USE

```

```

/      SET D=FX(I,A)
/      *****

```

```

/  WHERE "D" IS A DUMMY VARIABLE, "I" IS THE
/  MATRIX LOCATION TO BE USED, AND "A" IS THE
/  VARIABLE TO BE STORED INTO LOCATION "I".

```

/ TO RETRIEVE A VALUE FROM THE BUFFER USE

```

/      SET X=FX(I)
/      *****

```

```

/  WHERE VARIABLE "X" IS ASSIGNED THE VALUE OF
/  ARRAY ELEMENT "I".

```

```

/  FOURWD=2  ASSEMBLES FOR 3 WORD VARIABLES AND
/             34 VARIABLES PER BLOCK
/  FOURWD=1  ASSEMBLES FOR 4 WORD VARIABLES AND
/             63 VARIABLES PER BLOCK

```

```

/  TO BUILD AN OUTPUT FILE, THE FILE NAMED MUST
/  FIRST BE OPENED FOR OUTPUT BY USING THE PS/3
/  FOCAL 'O O NAME' INSTRUCTION. EACH VARIABLE MAY
/  THEN BE STORED IN RANDOM OR SEQUENTIAL
/  PATTERN USING THE FORM      FX(I,A).

```

/ DATA MAY ALSO BE RETRIEVED AT THE SAME TIME
/ BUT TO MAKE THE FILE PERMANENT, IT MUST BE
/ CLOSED WITH THE 'O C' COMMAND. NOTE THAT ANY
/ TYPING BY THE PROGRAM WILL GO INTO AND
/ DESTROY THE CONTENTS OF THE DATA FILE
/ UNLESS THE PROGRAM FIRST OPENS OUTPUT TO
/ THE TTY TO RESTORE THAT DEVICE. HOWEVER IT
/ IS NOT NECESSARY TO RESTORE OUTPUT TO
/ THE FILE DEVICE, AND 'O O TTY:' CAN BE
/ ISSUED IMMEDIATELY AFTER THE COMMAND TO
/ OPEN THE DATA FILE.
/

/ TO USE AN ALREADY EXISTING FILE, THE FILE
/ MUST FIRST BE OPENED FOR INPUT (!!) USING
/ THE 'O I NAME' COMMAND. THIS ESTABLISHES
/ THE FILE OPEN OPERATION WITHOUT ERASING
/ THE EXISTING FILE AND BRINGS THE REQUIRED
/ DEVICE HANDLER INTO THE MEMORY.
/

/ BEFORE ATTEMPTING TO USE THE FILE, YOU
/ MUST (!!) CALL TO FX WITH AN ARGUMENT
/ OF ZERO! THIS CALL TRANSFERS THE FILE
/ ADDRESS PARAMETERS AND DEVICE HANDLER
/ POINTERS FROM INPUT TO OUTPUT. INPUT
/ AND OUTPUT MAY THEN BE RESTORED TO THE
/ TELETYPE (NOT A PS/3 DEVICE) AND FX
/ USED AS DESCRIBED ABOVE. DATA CAN BE
/ INPUT AND OUTPUT AS LONG AS YOU DO NOT
/ ATTEMPT TO EXCEED THE LENGTH OF YOUR
/ FILE, IN WHICH CASE AN ERPOP MESSAGE
/ WILL BE GENERATED (13.45).
/

/ (WHEN BUILDING A DATA FILE YOU
/ CAN PAD IN ROOM FOR EXTRA DATA BY MAKING
/ IT LONGER THAN NECESSARY. E.G. STORING
/ FX(2047,0) AS THE LAST INSTRUCTION
/ BEFORE CLOSING, MAKING THE FILE
/ THE MAXIMUM SIZE ALLOWABLE).
/

/ DO NOT CLOSE AN EXISTING FILE. WHEN ADDING
/ TO OR CHANGING AN EXISTING FILE, TO
/ INSURE THAT THE LAST CHANGE IS SAVED,
/ READ IN A VALUE WITH A POSITION 100
/ LARGER OR SMALLER THAN THE LAST TO
/ INSURE THAT THE LAST CHANGE IS
/ SWAPPED OUT OF MEMORY ONTO THE FILE
/ DEVICE.

```

/      FUNCTION      FLAB
/      *****

```

```

/ ROUTINE TO CLEAR THE EXTERNAL SENSE REGISTER,
/ TO DISABLE THE ENABLE REGISTER AND TO CLEAR
/ THE FLAGS OF THE AX03 A/D-CONVERTER
/

```

```

/ FOCAL FUNCTION:
/

```

```

/      SET DUMMY=FLAB(0)
/      *****

```

```

/ ROUTINE TO CONTROL THE PEN LIFT OF A
/ X-Y-RECORDER WITH/WITHOUT NULL-DETEKTOR.
/
/ WITH NULL-DETEKTOR: CHANNEL "S0" WILL GIVE
/ THE "SEEK-PULSE". THE "PLOT-COMplete-
/ PULSE" OF THE RECORDER MUST BE NEGATIVE
/ AND SEND TO THE "EXTERNAL-CLOCK-INPUT".
/

```

```

/ WITHOUT NULL-DETEKTOR: CHANNEL "S0" GIVES A
/ PULSE TO CONTROL THE PEN LIFT. THE
/ SPEED IS CONTROLLED BY THE RC-CLOCK.
/

```

```

/ FOCAL FUNCTION:
/

```

```

/      SET DUMMY=FLAB(1,K1)
/      *****

```

```

/      K1=0      WITH NULL-DETEKTOR
/      1        WITHOUT NULL-DETEKTOR

```



```
/ ROUTINE TO SEND A PULSE TO OUTPUTS
/ S0, R1, R2, R3; PULSE LENGTH IS
/ 12(S0) OR 7 (R1...R3) CYCLE TIMES.
```

```
/ FOCAL FUNCTION:
```

```
/
/      SET DUMMY=FLAB(2,K1)
/      *****
```

```
/
/      K1=1,...,15
```

```
/ ROUTINE TO SWITCH THE OUTPUTS S0, R1, R2, R3
/ ON/OFF
```

```
/ FOCAL FUNCTION:
```

```
/
/      SET DUMMY=FLAB(3,K1,K2)
/      *****
```

```
/
/      K1=1,...,15
/      1 - R1
/      2 - R2
/      4 - R3
/      8 - S0
/      OR ANY COMBINATION
/      K2=0 DISABLE
/      1 ENABLE
```

```
/ ROUTINE TO SWITCH THE PC-CLOCK OR THE
/ EXTERNAL-CLOCK ON/OFF.
```

```
/ FOCAL FUNCTION:
```

```
/
/      SET DUMMY=FLAB(4,K1,K2)
/      *****
```

```
/
/      K1=1 PC-CLOCK, SLOW
/      2 PC-CLOCK, FAST
/      4 EXTERNAL-CLOCK
/      K2=0 DISABLE
/      1 ENABLE
```

```

/ ROUTINE TO WAIT ON A RC-CLOCK- OR
/ EXTERNAL-CLOCK-PULSE
/
/ FOCAL FUNCTION:
/
/   SET DUMMY=FLAB(5)
/   *****

```

```

/ ROUTINE TO WAIT FOR K2 PULSES OF THE
/ SPECIFIED CLOCK OF INPUT.
/
/ FOCAL FUNCTION:
/
/   SET DUMMY=FLAB(6,K1,K2)
/   *****
/
/   K1=1 RC-CLOCK, SLOW
/       2 RC-CLOCK, FAST
/       3 CRYSTAL-CLOCK
/       4 EXTERNAL-CLOCK
/   K2=1, ..., 4295
/   COUNTER

```

```

/ ROUTINE TO SYNCHRONIZE AND COUNT THE RC-CLOCK
/ OR EXTERNAL CLOCK TIME. THE TIME IS HELD IN
/ THE FL.F.AC..
/
/ FOCAL FUNCTION:
/
/   SET TI=FLAB(7,K1)
/   *****
/
/   K1=CLOCK FLAG COUNTER
/       K1=1, ..., 2247
/       K1*(CLOCK TIME) MUST BE
/       SMALLER THAN 1677.7216 SEC
/   TI=TIME OF CLOCK
/
/   TIME=(TI/K1)*0.1 MSEC

```

/ ROUTINE TO READ THE EXTERNAL SENSE REGISTER
/ AND TO CLEAR THE BITS GIVEN WITH K1.

/ FOCAL FUNCTION:

/ SET A=FLAB(8,K1)
/ *****

/ K1=0,...,2047
/ BITS GIVEN AS A DECIMAL
/ NUMBERS ARE CLEARED
/ A: BITS OF THE EXTERNAL
/ SENSE REGISTER AS A
/ DECIMAL NUMBER

/ THIS FUNCTION ARE FREE

/ FLAB(9)
/ FLAB(10)

/ IN LOCATION NEULAB=20503 (FF) YOU HAVE TO
/ STORE A POINTER TO YOUR NEW FUNCTION.
/ YOU HAVE TO MOVE LOCATION POINTER
/ (POINTER = END OF PROGRAM IN FIELD 2):
/ STORE THE NEW POINTER INTO LOCATION
/ IPOIN=20113 AND INTO POINT=20114 THE
/ VALUE OF POINTER-2).

/ FOR RETURN TO FOCAL, MAKE A
/ JMP PFOCAL = 5005 IN FIELD 2
/ OR
/ JMP I EFUN31 = 5506 IN FIELD 1


```

/      FUNCTION          FADC
/      *****

```

```

/ ROUTINE FOR ANALOG TO DIGITAL CONVERSION VIA
/ THE AX03 A/D-CONVERTER.
/

```

```

/ FOCAL FUNCTION:
/

```

```

/      1.1 SET A=FADC(K)
/      1.2 IF (A-256) 1.3,1.3; SET A=A-4096
/      1.3 SET A=A/256
/      *****
/

```

```

/      K = CHANNEL-NUMBER
/

```

```

/      CHANNEL-NUMBER "K":          0 - 31
/

```

```

/      THE VARIABLE "A" CONTAINS THE DECIMAL
/      DATA
/

```

/ ROUTINE FOR ANALOG TO DIGITAL CONVERSION VIA
 / THE AF04 A/D-CONVERTER

/ FOCAL FUNCTION:

/ SET A=FALC(K,F,P,S)

/ *****

/ K = CHANNEL-NUMBER

/ F = FUNCTION

/ P = RANGE

/ S = RESOLUTION

/ CHANNEL-NUMBER "K": 0 - 999

		(3)	(12)
FUNCTION "R": AC-VOLTAGE	=	400	256
DC-VOLTAGE	=	200	128
RESISTANCE	=	100	64
FREQUENCY	=	40	32
PERIOD	=	20	16
SCAN RESET	=	10	8
SWITCH CLOSURE	=	4	4

RANGE "R": 10 MV	=	1
100 MV	=	2
1000 MV	=	3
10 V / 10K	=	4
100 V / 100K	=	5
1000 V	=	6
AUTO RANGE	=	7

RESOLUTION "S": 0.1 %	=	1
0.01 %	=	2
0.001 %	=	3

/ THE VARIABLE "A" CONTAINS THE DECIMAL
 / DATA.

/ FUNCTION FM
/ *****

/ SUBROUTINE TO SERVE THE FAST FOURIER
/ TRANSFORMATION (FOCAL PROGRAM)

/ FOCAL FUNCTION:

/ SET A=FN(2,U,NU)
/ *****

/ SUBROUTINE BY JAMES E. ROTHMAN
/ DECUSCOPE 1968 VOL.7, NO.3, PP.3-10

/ SEE ALSO:
/ FREQUENCY TRANSFORMATION PROGRAM
/ DECUS PROGRAM LIBRARY FOCAL3-226


```
/ ROUTINE TO GET/STORE DATA FROM/INTO
/ THE DATA BUFFER OF FIELD 2
/
/ FOCAL FUNCTION:
/
/   SET A=FN(1,1)
/   *****
/                               /DATA AT PLACE "1"
/                               /IS HELD IN "A"
/   SET LUMMY=FN(1,1,A)
/   *****
/                               /DATA "A" IS STORED
/                               /AT PLACE "1"
/
/   I=0,1,...,593   /FOURWD=1
/                   / 4-WORD-PRECISION
/
/   I=0,1,...,799   /FOURWD=0
/                   / 3-WORD-PRECISION
/
/ THE DATA OF THIS FN-FUNCTION USE THE
/ SAME MEMORY PART AS THE FIO-FUNCTION !!
/ THE DATA OF THE FIO-FUNCTION ARE STORED
/ IN 2-WORD-PRECISION; E.G. IF YOU HAVE
/ 200 DATA WITHIN THE FIO-FUNCTION, YOU CAN
/ START THIS FN-FUNCTION WITH I=100
/ (4-WORD-PRECISION)
/
/ THE LAST PAGE OF FIELD 2 IS PROTECTED
```

```

/ ROUTINE TO GET/STORE DATA FROM/INTO
/ FIELD 3 FF.
/
/ FOCAL FUNCTION:
/
/   SET A=FN(2,I)
/   ****+*****
/                               /DATA AT PLACE "I"
/                               /IS HELD IN "A"
/   SET LUNNY=FN(2,I,A)
/   ****+*****
/                               /DATA "A" IS STORED
/                               /AT PLACE "I"
/
/                               /FOURWE=1
/   I=0,1,...,1023           /FIELD 3
/   =1024,...,2047           /FIELD 4
/   =2048,...,3071           /FIELD 5
/   =3072,...,4095           /FIELD 6
/
/                               /FOURWE=0
/   I=0,1,...,1364           /FIELD 3
/   I(MAX)=4095              /ETC.

```

/ ROUTINE TO SET THE PROGRAMMABLE REAL TIME
/ CLOCK DK3-EF TO ZERO AND TO START IT.

/ FOCAL FUNCTION:

/ SET DUMMY=FN(3,C)
/ *****

/ C = CLOCK TIME FLAG AT
/ "C" MILLISECONDS
/ C = 1, ..., 4096

/ ROUTINE TO READ THE NUMBER OF CLOCK FLAGS OF
/ THE DK3-EF PROGRAMMABLE REAL TIME CLOCK.

/ FOCAL FUNCTION:

/ SET TI=FN(4)
/ *****

/ THE VARIABLE "TI" CONTAINS THE NUMBER OF
/ CLOCK FLAGS.

/ REAL TIME = TI * C [MILLISECONDS]

/ WITH E.G. C=100 YOU GET A CLOCK FLAG AT
/ EVERY 100 MILLISECONDS, THE MAXIMUM TIME
/ WILL BE 33336.03 SECONDS OR
/ 23 HOURS 13 MINUTES AND 6.03 SECONDS.

```

/ THIS FUNCTIONS ARE FREE:
/
/      FN(5)
/      FN(6)
/      FN(7)
/      FN(3)
/      FN(9)
/      FN(10)
/
/ IN LOCATIONS NEWFN=20372 (FF) YOU HAVE
/ TO STORE A POINTER TO YOUR NEW FUNCTION
/ YOU HAVE TO MOVE LOCATION POINTER
/ (POINTER = END OF PFOCPAM IN FIELD 2):
/ STORE THE NEW POINTER INTO LOCATION
/ IPOIN=20113 AND INTO POINT=20114 THE
/ VALUE OF POINTER-2
/
/ FOR RETURN TO FOCAL, MAKE A
/      JMP PFOCAL = 5205      IN FIELD 2
/ OR
/      JMP I EFUN3I = 5536      IN FIELD 1

```


/ FUNCTION FD
/ *****

/ SCOPE FUNCTIONS

/ OVERLAY FOR KV31-OMSI FOCAL 71

/ SUBROUTINE BY MICHAEL E. EPICKSON
/ OVERLAY FOR KV31-OMSI FOCAL 71
/ DECUS PROGRAM LIBRARY FOCAL3-210

/ THIS PATCH PROVIDES OSMI FOCAL
/ THE ABILITY TO GENERATE
/ CHARACTERS ON A KV31 SCREEN,
/ BY USING THE VSCG. THE
/ ROUTINE ARE ALSO INSEPTED
/ INTO THE FUNCTION TABLE TO BE
/ USE ON A KV31 SCREEN.

/ THE OLD FOCAL FUNCTION "FLIS(X,Y)"
/ IS ACTIVE TOO.

/ KVS1-DISPLAY FUNCTION

/ FOCAL FUNCTION:

/ SET DUMMY=FD(1,X,Y)
/ *****

/ I-PEN VALUE

/ 1 : DRAW A VISIBLE VECTOR
/ TO LOCATION X & Y
/ 0 : DRAW AN INVISIBLE VECTOR
/ TO LOCATION X & Y
/ -1 : DRAW A POINT AT LOCATION X & Y
/ -N : DRAW A VISIBLE CIRCLE AROUND
/ LOCATION X & Y
/ WITH RADIUS N,
/ N < -1

/ X- X COORDINATE (HORIZONTAL)
/ Y- Y COORDINATE (VERTICAL)

/ * WORKS ONLY IN SCOPE MODE

/ +A - FOUT(01) - HOME
/ +B - FOUT(02) - * LINE FEED
/ +E - FOUT(05) - * END OF PAGE
/ +H - FOUT(03) - * BACK SPACE
/ +N - FOUT(14) - * UP SPACE
/ +O - FOUT(15) - * CARRIAGE RETURN
/ +C - FOUT(17) - INCREMENT SIZE
/ +R - FOUT(13) - RESET SIZE
/ +S - FOUT(19) - SCOPE MODE
/ +T - FOUT(20) - TTY MODE
/ +U - FOUT(21) - ERASE & HOME
/ +X - FOUT(24) - ERASE

```

/      FUNCTION      FIO
/      *****

```

```

/ WITH THIS FUNCTION YOU CAN RECEIVE AND
/ TRANSMIT DATA VIA THE KLB-JA INTERFACE.
/
/ THE ROUTINES ARE BUILT TO SPEAK WITH THE
/ CANBERRA MULTICHANNEL ANALYZER 3100.
/ BY CHANGING THE OUTPUT FROM TTY TO KLB-JA,
/ A SIGN "212" (LOCATION C212=22630) IS
/ SENT; IT STARTS THE CANBERRA 3100.
/ AT THE END OF TRANSMIT A SIGN "173"
/ (LOCATION C173=22627) IS SENT.

```

```

/ ROUTINE FOR OUTPUT DURING A FOCAL
/ TYPE COMMAND.
/

```

```

/ FOCAL FUNCTION:
/

```

```

/      SET DUMMY=FIO(-1,A)
/      *****
/

```

```

/      A = 0  OUTPUT ONTO TTY
/      A > 0  OUTPUT TO KLB-JA
/              TRANSMIT DATA FOR
/              INPUT INTO THE
/              CANBERRA MULTI-
/              CHANNEL ANALYZER
/

```

```

/ EXAMPLE:
/

```

```

/      SET DU=FIO(-1,1)
/      FOR I=1,N; T 26.00,A(I)
/      SET DU=FIO(-1,0)

```

```

/ ROUTINE TO GET AND FLOAT A DATA OF THE
/ DATA BUFFER INTO THE FL.F.AC..
/
/ FOCAL FUNCTION:
/
/     SET A=FIO(2,1)
/     *****
/                                     I > 0
/
/ THE VARIABLE "A" CONTAINS THE DATA.

```

```

/ ROUTINE FOR INPUT DATA DURING FOCAL VIA THE
/ KLS-JA INTERFACE: RECEIVE DATA.
/
/ FOCAL FUNCTION:
/
/     SET D=FIO(1,1)
/     *****
/                                     I > 0
/
/ AFTER INPUT THE NUMBER OF DATA CONTAINS
/ VARIABLE "D"; THAT IS: "I" OF THE MAXIMUM
/ POSSIBLE NUMBER, I(MAX)=1199.
/
/ THIS FUNCTION STORES THE DATA AS A UNSIGNED
/ 2-WORD-PRECISION NUMBER. THE DATA ARE
/ ALWAYS STORED AT THE PLACES I=1,2,...,I(MAX).
/
/ THE FN-FUNCTION FN(1,1,A) USES THE SAME
/ LOCATIONS !!

```



```

/ THIS FUNCTIONS ARE FREE:
/
/      FIO(2,K1)
/      FIO(3,K1)
/      FIO(4,K1)
/
/ IN THE LOCATIONS NEWFIO=22573 (FF) YOU HAVE
/ TO STORE A POINTER TO YOUR NEW FUNCTION
/ (IT MUST HAVE THE 1. APC. AND 2. APC. K1).
/ YOU HAVE TO MOVE LOCATION  POINTER
/ (POINTER = END OF PROGRAM IN FIELD 2):
/ STORE THE NEW POINTER INTO LOCATION
/ IPOIN=22113 AND INTO POINT=22114 THE
/ VALUE OF POINTER-2
/
/ FOR RETURN TO FOCAL, MAKE A
/      JMP PFOCAL = 5025      IN FIELD 2
/ OR
/      JMP I EFUN3I = 5536      IN FIELD 1

```

C IKT-TUB FOCAL '31

```

01.01 C TEST PROGRAMM
01.02 C
01.03 C     AX08 A/D-CONVERTER: LAB-3 SYSTEM
01.04 C     DK3-EP PROGRAMMABLE REAL TIME CLOCK
01.05 C     AF04 A/D-CONVERTER
01.06 C     KLS-JA INTERFACE WITH CANBERRA 8100
01.07 C     KVS1 STORAGE DISPLAY
01.08 C     DATA FILES IN A RANDOM FASHION ONTO DTA0:
01.09 C     FFT ROUTINE
01.10 C
01.11 C
01.12 S D=FOUT(19); F I=1,50; S D=I*I
01.13 S D=FOUT(13)+FOUT(17)+FOUT(21)
01.15 A !!,"AX08 - TEST ? ",A; I (A-25) 1.25,1.2,1.25
01.20 L G DTA0:AX08 2
01.25 D 2; D 3; D 4; D 5; D 10; D 12; D 19
01.30 Q

02.01 C DK3-EP PROGRAMMABLE REAL TIME CLOCK TEST
02.02 C
02.03 C
02.10 D 1.13; A !!,"DK3-EP REAL TIME CLOCK TEST ? ",A
02.15 I (A-25) 2.45,2.2,2.45
02.20 S D=FN(3,1000); S I=0; S J=5
02.23 T "BELL 1 MINUTE EVERY 5 SECONDS",!; S D=FOUT(20)
02.25 S A=FN(4)
02.30 I (A-J) 2.25,2.35
02.35 S D=FOUT(135); S J=A+5; S I=I+1
02.40 I (I-12) 2.25,2.45,2.25
02.45 D 1.12; R

03.01 C AF04 A/D-CONVERTER TEST
03.02 C
03.03 C
03.10 D 1.13; A !!,"AF04 - TEST ? ",A; I (A-25) 3.3,3.2,3.3
03.20 L G DTA0:AF04
03.30 R

04.01 C KLS-JA INTERFACE WITH CANBERRA 8100 TEST
04.02 C
04.03 C
04.10 D 1.13; S CA=0; A !!,"CANBERRA 8100 INPUT/OUTPUT TEST ? ",A,!!
04.12 I (A-25) 4.4,4.15,4.4
04.15 A "MEMORY: 1024",!,"I/O DEVICE: TTY IN",!,"I/O",!,"?",!,"A
04.20 S CA=1; S D=FIO(-1,1); S X=0; F I=0,1024; T %6.20,I; S X=X+I
04.25 S D=FIO(-1,0); T !,"I/O DEVICE: TTY OUT",!,"INTEGRAL",!,"I/O"
04.30 S X=X-1024; S D=FIO(1,1039); I (X-FIO(0,3)) 4.35,4.4,4.35
04.35 S D=FOUT(17); T !!!!!,"INPUT/OUTPUT ERROR"; G
04.40 R

```

```

05.01 C KVBI STOPAGE DISPLAY TEST
05.02 C
05.03 C
05.05 D 1.13; A !!,"KVBI-DISPLAY TEST ? ",A; I (A-25) 5.12,5.1,5.12
05.10 D 1.13; F I=1,20,340; D 6
05.12 R

06.10 S D=FD(0,512-1,512-1); S D=FD(1,512-1,512+1)
06.20 S D=FD(1,512+1,512+1); S D=FD(1,512+1,512-1)
06.30 S D=FD(1,512-1,512-1); S D=FD(-1,512,512)

10.01 C DATA FILES IN A RANDOM FASHION ONTO DTA0:
10.02 C
10.03 C
10.05 D 12.1; A !!,"DATA FILES IN A RANDOM FASHION ? ",D
10.07 I (D-25) 10.99,10.1,10.99
10.10 O 0 DTA0:TEST; O 0 TTY:
10.15 I (CA) 10.2,10.17,10.2
10.17 F I=1,1025; S X=FX(I,I)
10.18 G 10.25
10.20 S I=1; F J=6,101,1017; F K=1,100; S I=I+1; S X=FX(I,F10(0,J+K))
10.25 D 1.13; T !!,"DATA FILES IN A RANDOM FASHION ONTO DTA0:",!!
10.30 S D=FOUT(13); T !; F I=1,100,1001; T %10.00,FX(I),!
10.40 O C
10.45 T !
10.50 O I DTA0:TEST; O I TTY:,E
10.60 S X=FX(0); C IMPERATIVE
10.70 D 10.3; T !,FX(1024),!
10.80 S X=FX(400,999); S X=FX(250,33333); S X=FX(50,1234567890)
10.90 S X=FX(150); C SWAP OUT LAST CHANGE
10.95 T !,%10.00,FX(400),!,FX(250),!,FX(50),!; E
10.96 L D DTA0:TEST.FD
10.99 R

12.01 C FFT ROUTINE: TEST FREQUENCY = COSINUS 3 HZ
12.02 C
12.03 C
12.10 S D=FOUT(13)+FOUT(17)
12.20 A !!,"FFT OF 3 HZ COSINUS ? ",D; I (D-25) 12.5,12.3,12.5
12.30 L G DTA0:FFT !; E
12.40 D 12.1; A !!,"NEXT ? ",D
12.50 R

19.01 C END OF TEST PROGRAM
19.02 C
19.03 C
19.10 D 1.13
19.20 F I=1,6; S D=FOUT(17)
19.30 T !!!,"TEST O.K."; S D=FOUT(13)+FOUT(17)+FOUT(20)+FOUT(01)
19.40 T !!,"DTA0:!!; L L DTA0:; T !!,"SYS:",!; L L

```


C IKT-TUB FOCAL '91

```

01.10 C AX08 A/D-CONVERTER TEST
01.15 C
01.20 C TEST: POTENTIOMETER 0-1-2-3
01.25 C X-Y-RECORDER WITH NULL-DETEKTOR
01.30 C X-Y-RECORDER, SPEED CONTROLLED BY RC-CLOCK
01.35 C PULSE OUTPUTS
01.40 C RELAY OUTPUTS
01.45 C SLOW RC-CLOCK RATE
01.50 C EXTERNAL SENSE REGISTER
01.60 C
01.70 C
01.80 C
01.90 D 2; Q

02.01 S D=FOUT(0); S D=FOUT(18); S D=FOUT(17); S D=FOUT(21)
02.06 A !,"ADC-POTENTIOMETER TEST ? ",A; I (A-25) 2.27,2.99,2.27
02.07 D 2.01
02.08 A !,"X-Y-RECORDER WITH NULL-DETEKTOR ? ",A
02.10 I (A-25) 2.14,2.11,2.14
02.11 S D=FLAB(3,8,1)
02.12 S D=FLAB(3,8,0); F I=0,2,510; S D=FDIS(1,1-255); S D=FLAB(1,0)
02.14 A !,"TEST X-Y-RECORDER",!,"SPEED CONTROLLED BY RC-CLOCK ? ",A
02.15 I (A-25) 2.2,2.16,2.2
02.16 F I=0,2,510; S D=FDIS(1,1-255); S D=FLAB(1,1)
02.20 A !,"TEST OF PULSE-OUTPUTS ? ",A; I (A-25) 2.3,2.22,2.3
02.22 F I=1,15; S D=FLAB(2,1)
02.24 F J=1,9; D 2.22
02.30 A !,"TEST RELAY-OUTPUTS ? ",A; I (A-25) 2.4,2.32,2.4
02.32 S D=FLAB(6,3,4000)
02.34 F I=1,15; S D=FLAB(3,1,1); D 2.32; S D=FLAB(3,1,0)
02.40 D 2.01
02.42 T !,"TEST SLOW RC-CLOCK SPEED "
02.44 A !,"POTENTIOMETER 0 FAP LEFT ? ",A; I (A-25) 2.5,2.45,2.5
02.45 T !,"END OF TEST: FAP RIGHT",!
02.46 S A=FLAB(4,1,1); S D=FLAB(7,2)
02.48 T !,10.05,0.0001*D/2," SEC"; I (256-FADC(23)) 2.46, 2.5
02.50 D 2.01; T !,"TEST EXTERNAL SENSE REGISTER",!
02.51 A "POTENTIOMETER 0 FAP LEFT ? ",A; I (A-25) 2.7,2.53,2.7
02.53 D 2.45; T !,"TTY-INPUT (DECIMAL): BITS TO CLEAR",!; G 2.56
02.54 A A; S D=FLAB(8,A)
02.56 T 14.00,D," (8) IS SET CLEAR (10): "; I (256-FADC(23)) 2.54
02.60 S D=FLAB(0); E
02.70 R
02.99 T !; D 3; G 2.01

03.10 S J=0
03.20 F I=23,31; D 4
03.30 T !; S J=J+1; I (J-20) 3.2; R

04.10 S A=FADC(1); I (A-256) 4.2,4.2; S A=A-4096
04.20 T 14.00,A/250," "

```


-C IKT-TUB FOCAL'S I

-03.01 C AF04 A/D-CONVERTER TEST

03.02 C

03.03 C

03.15 S A=FADC(0,8,1,1); T %5.04

03.20 F I=1,3; F J=3,7; S A(I+(J-1)*7)=FADC(J,128,J,I)

03.25 F I=1,3; T !; F J=3,7; T A(I+(J-1)*7)

03.30 T !; D 3.2; D 3.25

03.35 D 3.3; T !!

03.40 R

*B

```

01.01 C FFT ROUTINE
01.02 C
01.03 C
01.04 C INPUT DATA FOR GROUP 16:
01.05 C      X      GROUP NUMBER OF HEADER AND OUTPUT
01.06 C      NU     NUMBER OF DATA IS N=2*NU
-01.07 C      IF     FFT=1      IFFT=-1
01.08 C      TM     N POINTS REPRESENTED THE TIME "TM"
01.09 C      FFT     REAL PART: DATA IN FN(1,1)
01.10 C      IMAG.PART: DATA IN FN(2,1)=0
01.11 C      IFFT    REAL PART: DATA IN FN(1,1)
01.12 C      IMAG.PART: DATA IN FN(2,1)
01.13 C
01.15 C TEST FREQUENCY: COSINUS 3 HZ
01.16 C
01.17 C
01.20 S NU=5; S P2=6.283185303; S N=2*NU; S K=3
01.25 F I=0,N-1; S D=FCOS(3*P2*I/N); S E=FN(1,1,D); S L=FN(2,1,D)
01.30 S IF=1; S TM=1
01.40 D 16

02.10 C

03.01 S D=FOUT(18)+FOUT(17)+FOUT(21)
03.02 T !!,"TEST FFT",!!
03.04 T "FREQUENCY = 3 HZ",!!!; S D=FOUT(13)
03.08 T !!!
03.10 T "      [HZ]      REAL      IMAGINARE      POWER",!
03.12 T "      PART      PART      SPECTRUM"
03.14 T !!!
03.20 S DF=1/TM; F I=0,N/2-1; D 17

16.04 S N=2*NU; S P2=6.283185303
16.10 S T=P2/N; S S=N/2; S L=1; S Q=S-1; S H=1-NU
16.14 S SR=FN(1,Q+S)+FN(1,Q); S DD=FN(1,Q)-FN(1,Q+S)
16.15 S D=FN(1,Q+S,DD); S D=FN(1,Q,SR)
16.16 I [Q] 16.18,16.18; S Q=Q-1; G 16.14
16.18 I [L-NU] 16.20,16.42,16.20
16.20 S L=L+1; S S=S/2; S H=H+1; S P=N-1; S Z=1/[2*(-H)]
16.22 S C=1
16.24 S U=FITR(P*Z); S K=T*FN(0,U,NU)
16.26 S CO=FCOS(K); S SN=IF*FSIN(K)
16.28 S GR=CO*FN(1,P)+SN*FN(2,P); S GI=CO*FN(2,P)-SN*FN(1,P)
16.30 S Q=P-S; S SR=GR+FN(1,Q); S SI=GI+FN(2,Q)
16.31 S DD=FN(1,Q)-GR; S D=FN(1,Q,DD)
16.32 S DD=FN(2,Q)-GI; S D=FN(2,Q,DD); S D=FN(1,P,SP)
16.33 S D=FN(2,P,SI)
16.34 S P=P-1; I [C-S] 16.36,16.38,16.36
16.36 S C=C+1; G 16.24
16.38 I [F-S+1] 16.40,16.13,16.40
16.40 S P=P-S; G 16.22
16.42 D X
16.50 R

17.10 T 17.04,I*DF," "; S K=FN(2,1,NU)
17.12 S SP=2*FN(1,K)/N; S SI=2*FN(2,K)/N; S PS=FSGT(SP*SP+SI*SI)
17.14 T 17.06,SP," ",SI," ",PS,!

```

MISCELLANEOUS

EXPRESSIONS — ARITHMETIC EXPRESSIONS INCLUDE ANY EVALUATABLE GROUP OF NUMBERS OR VARIABLES SUCH AS: A7, 2+3*X OR 22/7

HIGH SPEED PAPER TAPE READER CAN READ IN FOCAL PROGRAMS:

- 1) TYPE 'OPEN INPUT PTR:'
- 2) AFTER '~' IS TYPED OUT, HIT SPACE BAR TO READ TAPE
- 3) AFTER '?' IS TYPED OUT, HIT 'CTRL/P'

USING YOUR TERMINAL'S LOW SPEED READER, YOU CAN INPUT PROGRAM TAPES BY FIRST TURNING OFF THE ECHO WITH 'OPEN INPUT TTY:;' READING IN THE TAP. AND THEN TURNING ON THE ECHO WITH CTRL/P OR 'OPEN INPUT TTY:;E'.

THE FOLLOWING 3 LINE PROGRAM WILL ASK FOR 10 NUMBERS FROM THE READER:

```
10. 60 OPEN INPUT PTR:
    (PS/8 WILL NOW OUTPUT A '~' ON THE TERMINAL AND WAIT FOR YOU
    TO SIGNIFY THAT THE TAPE IS LOADED BY HITTING THE SPACE BAR.)
10. 65 FOR X=1,10: ASK A(X)
10. 70 OPEN INPUT TTY:;ECHO: COMMENT --- RESTORES TERMINAL TO NORMAL
```

HIGH SPEED PUNCH MAY BE USED TO PUNCH PROGRAM TAPES:

- 1) TYPE 'OPEN INPUT TTY:; OPEN OUTPUT PTP:;W'
- 2) THEN TYPE 'OUTPUT CLOSE: OPEN INPUT TTY:;E'

LINE NUMBERS — MAY RANGE FROM 1.01 THROUGH 31.99 BUT DO NOT INCLUDE WHOLE NUMBERS. VARIABLES (OR ARITHMETIC EXPRESSIONS NOT STARTING WITH 'A') OF THE CORRECT VALUE MAY BE USED IN PLACE OF LINE NUMBER REFERENCES (EXAMPLE: 'GOTO X'). GROUP NUMBERS ARE INTEGERS FROM 1 THROUGH 31 AND REFERENCE GROUPS OF LINES IN 'DO', 'WRITE', AND 'ERASE' COMMANDS.

MERGING PROGRAMS 'A' AND 'B' IS DONE WITH A SERIES OF COMMANDS WHICH CONVERT 'A' INTO A PS/8 EDITOR COMPATIBLE DATA FILE AND THEN BRING IT IN 'ON TOP' OF PROGRAM 'B':

- 1) LIBRARY CALL A
- 2) OPEN INPUT TTY:
- 3) OPEN OUTPUT A (STEPS 3-4 WILL NOT OUTPUT ON TERMINAL)
- 4) OUTPUT CLOSE: OPEN INPUT TTY:;ECHO
- 5) LIBRARY CALL B
- 6) OPEN INPUT A (WAIT FOR '?' ON TERMINAL THEN HIT 'CTRL/P')
- 7) LIBRARY DELETE A. FD

THE MERGED PROGRAM WILL NOW BE IN YOUR PROGRAM BUFFER.

PRECISION --- IF 10 DIGIT PRECISION IS NOT RETAINED THROUGH THE INITIAL DIALOGUE, YOU WILL HAVE 20% MORE VARIABLES BUT ONLY 6 DIGIT PRECISION.

PIP --- A PS/8 SYSTEM PROGRAM WHICH CAN LIST YOUR ENTIRE DEVICE DIRECTORY. TYPE 'R PIP' TO PS/8 MONITOR THEN 'DEVICE:/L' TO LIST DIRECTORY. WHEN TRANSFERRING FOCAL PROGRAM FILES WITH PIP, BE SURE TO USE THE '/I' OPTION AS THEY ARE CORE IMAGE FILES --- NOT ASCII!!!

PROGRAM NAMES --- MAY HAVE UP TO SIX ALPHABETIC OR NUMERIC CHARACTERS. FOCAL ASSUMES '.FC' OR '.FD' EXTENSIONS FOR PROGRAM AND DATA FILE NAMES.

PROGRAM SIZE --- WITH STANDARD FEATURES, PS/8 FOCAL PROGRAMS MAY BE 7 BLOCKS LONG (ABOUT 3500 CHARACTERS). IF FILES ARE DELETED THROUGH THE INITIAL DIALOGUE, 11 BLOCK PROGRAMS ARE POSSIBLE.

PUSHDOWN LIST OVERFLOW --- WHEN A PROGRAM USES A 'DO' OR 'LIBRARY GOSUB' COMMAND, FOCAL RECORDS THE LOCATION IN ITS PUSHDOWN LIST SO THAT IT CAN 'RETURN' TO THE NEXT COMMAND AFTER COMPLETING THE 'DO' SUBROUTINE. IF TOO MANY SUBROUTINES ARE REQUESTED, THE LIST WILL OVERFLOW. TO REMEDY THIS SITUATION, REDUCE THE NUMBER OF BLOCKS OR SUBROUTINES IN PROGRAM.

STRINGS --- ARE ANY SERIES OF CHARACTERS SUCH AS:

```
HELLO
122.5
$99.95
NOW IS THE TIME FOR ALL GOOD MEN ....
TEST34
```

THE ABOVE STRINGS WERE TYPED BY SURROUNDING THEM WITH QUOTATION MARKS AS IN LINE 22.25:
22.25 TYPE " HELLO"

STRINGS MAY ALSO BE PRINTED OUT CHARACTER BY CHARACTER USING THE 'FOUT()' FUNCTION; INPUT OF STRINGS MAY BE HANDLED WITH THE 'FIN()' FUNCTION. LIST THE INITIAL DIALOGUE FOR AN EXAMPLE.

THE ASK COMMAND MAY BE USED TO INPUT AND COMPARE 6 OR 10 CHARACTER STRINGS (DEPENDING ON PRECISION RETAINED IN INITIAL DIALOGUE). IT IS BEST TO RETAIN FEATURE '8' IN DIALOGUE WHEN USING THIS TECHNIQUE.

```
22.80 ASK "ANSWER YES OR NO? ", AN; COMMENT 'YES OR NO' SUBROUTINE
22.82 IF (AN=0YES)22.84, 22.86
22.84 IF (AN=0NO)22.8, 22.88, 22.8
22.86 SET X=2; RETURN
22.88 SET X=1; RETURN
```

```
*00 22.8
ANSWER YES OR NO? YEP
ANSWER YES OR NO? NOPE
ANSWER YES OR NO? YES
```

PROGRAM CONTROL WOULD THEN RETURN TO THE COMMAND FOLLOWING THE SUBROUTINE CALL ('DO' OR 'LIBRARY GOSUB') WITH X EQUAL TO 2.

VARIABLES

VARIABLES --- SOME EXAMPLES INCLUDE: A, X, Z7, P2, AN, QU, PI, NUMBER (ONLY THE FIRST TWO LETTERS WILL BE USED) --- VARIABLES MAY NOT BEGIN WITH AN 'F' OR A DIGIT.

SUBSCRIPTING --- VARIABLES MAY HAVE UNIQUE SUBSCRIPTS FROM -2048 THROUGH 2047 ALTHOUGH THE TOTAL NUMBER OF VARIABLES IS STILL LIMITED BY CHOICES MADE IN THE INITIAL DIALOGUE. NOTE THAT THE SYMBOL TABLE WILL ONLY ACCURATELY OUTPUT SUBSCRIPTS BETWEEN 0 AND 99.

PROTECTED VARIABLES --- '!', '" AND '*' CANNOT BE 'TYPED', 'ASKED' OR ZEROED WITH THE 'ERASE' COMMAND DIRECTLY BUT ARE OTHERWISE NORMAL VARIABLES AND MAY BE USED AS SUCH IN 'SET', 'IF', 'GOTO', 'FOR' AND OTHER COMMANDS. '*', THE SYSTEM STATUS VARIABLE, IS 'SET' TO SPECIAL CODES BY THE INITIAL DIALOGUE.

INITIAL DIALOGUE

INITIAL DIALOGUE --- ASK FOCAL'S INITIAL DIALOGUE FOR 'HELP' WHICH WILL LIST OPTIONS OTHER THAN STANDARD FEATURES. BY NOT RETAINING FILES (FEATURE #4) PROGRAM SPACE IS INCREASED FROM ABOUT 3500 CHARACTERS (7 BLOCKS) UP TO ABOUT 5500 CHARACTERS (11 BLOCKS). IF FEATURES 5 AND 6 ARE BOTH REQUESTED, 6 WILL BE IGNORED.

THE SYSTEM STATUS VARIABLE, '*', LIKE '!' AND '"' CANNOT BE 'TYPED', 'ASKED' OR 'ERASED' BUT IS OTHERWISE A NORMAL VARIABLE. THE SYSTEM STATUS VARIABLE IS 'SET' TO THE PRODUCT OF THE CODES REPRESENTING FEATURES CHOSEN IN THE INITIAL DIALOGUE.

FEATURE	CODE
'0' NO OPTIONAL FEATURES	1
'1' FSIN AND FCOS FUNCTIONS	2
'2' FSIN, FCOS, FLOG, FEXP AND FATN	6
'3' EXTENDED PRECISION (10 DIGITS)	5
'4' FILE COMMANDS	7
'5' LEADING SPACE IN 'TYPE'	11
'6' '*' AUTOMATICALLY PRINTED BY 'TYPE' ..	13
'7' ':' AUTOMATICALLY PRINTED BY 'ASK' ..	17
'8' '&' IN POWERS-OF-TEN NOTATION	-1
'9' FEATURES 2, 3, 4	210
'YES' STANDARD FEATURES 2, 3, 4, 5	2310

THE NUMBER OF ORDINARY VARIABLES PERMITTED DEPENDS ON WHETHER EXTENDED PRECISION OR EXTENDED FUNCTIONS ARE RETAINED:

	NO EXTENDED FUNCTIONS	FSIN & FCOS ONLY (1)	FSIN, FCOS, FLOG FLOG, & FEXP (2)
6 DIGIT PRECISION:	222	196	118
10 DIGIT PRECISION:	185	163	98

COMMAND EXECUTION TIMES

6 & 10 DIGIT PRECISION VERSIONS OF PS/8 FOCAL, 1971 WERE USED TO TEST THE COMMAND EXECUTION TIMES IN MILLISECONDS ON A PDP-8 USING A TU55 DECTAPE FOR INPUT AND HIGH SPEED TERMINAL FOR OUTPUT. TIMES INCREASE AS LINES ARE REFERENCED TOWARD THE END OF A PROGRAM AND VARIABLES ARE USED FROM THE END OF THE SYMBOL TABLE.

COMMAND	6 DIG	10 DIG	COMMAND	6 DIG	10 DIG
ASK X	13.42	13.57	S X=H-H	6.98	7.02
A X	13.08	13.09	S X=G*G	7.73	8.53
A X,X	25.69	25.96	S X=H*H	8.47	9.99
COMMENT	1.50	1.50	S X=G/G	8.71	10.34
C	0.51	0.51	S X=H/H	8.76	10.39
DO 1.3	16.06	18.38	S X=444444/444444	15.48	17.12
D 1.3	15.91	18.22	S X=444444/333333	15.56	17.35
FOR X=1.1	14.94	15.01	S X=1E10/1E-10	50.11	79.77
F X=1.1	14.60	14.67	S X=1E100/1E-100	357.57	657.90
F X=1.2	18.13	18.23	S X=FABS()	6.09	6.15
F X=1.10	50.27	50.63	S X=FABS(G)	7.31	7.37
GO 1.3	14.43	16.74	S X=FABS(4)	9.59	9.66
G 1.3	14.27	16.59	S X=FABS(H)	7.35	7.40
G Z	9.45	10.28	S X=FABS(444444)	10.69	10.7
IF (-1)1.3,1.3,1.3	20.87	23.22	S X=FSQT(G)	22.79	31.77
I (-1)1.3,1.3,1.3	20.69	23.04	S X=FSQT(4)	25.07	34.06
I (L)1.3,1.3,1.3	16.78	19.10	S X=FSGN(G)	8.01	7.90
I (1)1.3,1.3,1.3	20.47	22.79	S X=FITR(G)	9.29	9.34
I (N)1.3,1.3,1.3	18.04	20.38	S X=FRAN()	7.43	7.50
I (-1)Z,Z,Z	15.85	16.70	S X=FEXP(G)	26.50	36.56
I (L)Z,Z,Z	11.93	12.77	S X=FSIN(G)	31.69	48.29
I (1)Z,Z,Z	15.04	15.88	S X=FCOS(G)	33.32	50.06
I (N)Z,Z,Z	12.63	13.46	S X=FATN(G)	25.09	37.10
RETURN	1.42	1.42	S X=FLOG(G)	14.78	48.25
R	.61	.61	S X=FIN()	6.24	6.30
S X=N	4.10	4.11	S X=FIN()+FIN()	11.29	11.38
S X=1	6.47	6.52	S X=[TEN FIN() S]	65.37	65.60
S X=G	4.15	4.15	S X=FOUT(K)	9.19	9.26
S X=4	6.35	6.37	S X=FOUT(100)	11.91	11.98
S X=1+1	11.67	11.72	S X=FOUT(K)+FOUT(K)	17.28	17.29
S X=H	4.16	4.18	S X=[10 FOUT(K) S]	82.46	82.88
S X=444444	7.43	7.46	TYPE	1.19	1.19
S X=G+G	6.87	6.91	T	.71	.71
S X=G+1	9.35	9.39	T !	1.32	1.34
S X=1+1	11.67	11.71	T #	1.38	1.38
S X=H+H	6.94	6.97	T "4"	1.46	1.46
S X=444444+H	10.38	10.47	T " 444444"	3.38	3.38
S X=444444+444444	13.69	13.72	T " 444444" !	4.06	4.06
S X=G+G+G	9.73	9.79	T " 444444", !	4.29	4.29
S X=G+(G+G)	11.51	11.59	T H	16.20	26.40
S X=G~J	26.80	34.47	T H, !	16.87	27.06
S X=4~J	29.15	36.82	T 1E25	91.20	163.49
S X=4~10	31.62	39.29	T 0001E5/0001E5	33.27	43.78
S X=4~100	185.37	261.78	T 100000/100000	18.20	20.83
S X=H+H	6.94	6.98	T "[20 SPACES]" !	7.76	7.77
S X=G-G	6.91	6.95	T :20, !	11.56	11.59

CHANGING PS/8 FOCAL

FOR ADDITIONAL INFORMATION, YOU MAY CONSULT 'ADVANCED FOCAL TECHNICAL SPECIFICATIONS' (ORDER NUMBER DEC-08-AJ88-DL FROM THE DEC PROGRAM LIBRARY, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS).

NEARLY ALL USER PATCHES TO FOCAL 1969 CAN BE ADDED TO PS/8 FOCAL 1971 WITH ONLY MINOR ADAPTATIONS. SOME POINTS WHICH YOU MAY WISH TO CHECK TO INSURE COMPATABILITY INCLUDE:

1) SOME FOCAL INTERNAL SUBROUTINES AND POINTERS HAVE BEEN CHANGED (E.G. 'POPA' IS NOW A SUBROUTINE CALL). PS/8 FOCAL ROUTINES WHICH PERFORM IDENTICAL FUNCTIONS ARE INCLUDED FOR ALL COMMONLY USED ROUTINES.

2) IN MOST CASES, THE 'HOLES' IN FOCAL 1969 HAVE BEEN FILLED WITH PATCHES. A USER ROUTINE WHICH USES THESE LOCATIONS MUST BE RELOCATED. THE VARIABLE SYMBOL TABLE AREA IS RECOMMENDED FOR THIS PURPOSE (CHANGE 'STVAR' TO ALLOW FOR PATCHES).

USER-DEFINED FUNCTIONS REQUIRE POINTERS IN 'FNTABF' AND 'FNTABL'. THE 'FX' AND 'FNEW' SLOTS HAVE BEEN FILLED WITH 'FIN' AND 'FOUT'. RECOMMENDED FUNCTIONS TO DELETE FOR EXTRA SLOTS ARE FDIS, FADC AND FRAN DEPENDING ON THE NEEDS OF YOUR INSTALLATION.

4) FOCAL PROPER NOW RESIDES IN FIELD 1 (STARTING ADDRESS 10200). ANY PATCHES TO FOCAL MUST BE LOADED INTO FIELD 1 TO OVERLAY FOCAL. THE LIBRARY ROUTINES, PUSHDOWN STACK, AND TEXT STORAGE USE FIELD 0.

THE FOCOV.R PA SOURCE FILE CONTAINS ALL PATCHES MADE BY PS/8 FOCAL TO FOCAL PROPER AND SHOULD BE EXAMINED TO INSURE THERE ARE NO CONFLICTS WITH A USER PATCH.

SOME TERMINALS HAVE 'ESCAPE' OR 'PREFIX' IN PLACE OF 'ALT MODE'. TO CAUSE PS/8 FOCAL TO RECOGNIZE THE NON-ALT MODE, FIRST SAVE IT ON THE SYSTEM DEVICE. THEN GIVE THE PS/8 MONITOR THE FOLLOWING COMMANDS:

```
. GET SYS:PFOCAL
. ODT
17002/0375 233 (ESCAPE) OR 376 (PREFIX)
^C
. SAVE SYS:XFOCAL
```

IF PS/8 FOCAL IS TO BE USED IN THE PREPARATION OF NUMERICAL CONTROL TAPES, IT WILL BE NECESSARY TO MAKE A CHANGE SO THAT FOUT(141) WILL NOT AUTOMATICALLY OUTPUT BOTH A CARRIAGE RETURN AND LINE FEED. TO DO THIS FIRST SAVE PS/8 FOCAL ON THE SYSTEM DEVICE. THEN GIVE THE PS/8 MONITOR THE FOLLOWING COMMANDS:

```
. GET SYS:PFOCAL
. ODT
17541/4451 4463
^C
. SAVE SYS:NFOCAL
```

CHARACTERS OUTPUT WITH 'FOUT' WILL NOT CHANGE THE 'TAB' POSITION.

INDEX FOR HELP

COMMANDS	SYMBOLS	FUNCTIONS
<u>ELEMENTARY COMMANDS</u>	<u>ARITHMETIC SYMBOLS</u>	FABS() ABSOLUTE VALUE FADC() ANALOG-DIGITAL FATN() ARCTANGENT FCOS() COSINE FDIS() DISPLAY FEXP() EXPONENTIAL FITR() INTEGER FLOG() LOG (BASE E) FRAN() RANDOM NUMBER FSGN() SIGN PART FSIN() SINE FSQT() SQUARE ROOT FINK() CHARACTER INPUT FOUT() CHARACTER OUTPUT
ASK COMMENT CONTINUE DO ERASE FOR GO GOTO IF MODIFY QUIT RETURN SET TYPE WRITE	^ EXPONENTIATION * MULTIPLICATION / DIVISION + ADDITION - SUBTRACTION () PARENTHESES [] SQUARE BRACKETS < > ANGLE BRACKETS <u>SPECIAL CHARACTERS</u> ! RETURN/LINE FEED # CARRIAGE RETURN : TAB % OUTPUT FORMATTER \$ SYMBOL TABLE " QUOTATION MARKS ? TRACE E POWER-OF-TEN [&] RUBOUT KEY CTRL/C CTRL/G [BELL] CTRL/L CTRL/P LINE FEED KEY ← KILL LINE RETURN KEY ALT MODE KEY SPACE KEY , COMMA ; SEMICOLON	<u>MISCELLANEOUS</u> CHANGES FROM FOCAL, 1969 DEVICE NAMES DIRECT COMMANDS ERROR CODES EXPRESSIONS FILES HIGH SPEED READER INDIRECT COMMANDS INITIAL DIALOGUE LINE NUMBERS MERGING PROGRAMS NAMES OF FILES PRECISION PIP PUSHDOWN LIST OVERFLOW SAMPLE FILES PROGRAM SIZE OF PROGRAMS STRINGS SUBSCRIPTING SUMMARY OF COMMANDS TAB VARIABLES
<u>LIBRARY COMMANDS</u>		
LIBRARY CALL LIBRARY DELETE LIBRARY GOSUB LIBRARY LIST LIBRARY RUN LIBRARY SAVE LIBRARY EXIT		
<u>FILE COMMANDS</u>		
OPEN INPUT OPEN OUTPUT OPEN RESTORE INPUT OPEN RESTORE OUTPUT OUTPUT CLOSE		

THE HELP PACKAGE FOR PS/8 FOCAL, 1971 PERMITS THE ON-LINE REVIEW OF ANY OF THE ABOVE WORDS, PHRASES OR SYMBOLS. HELP IS AVAILABLE ON A DECtape WHICH SHOULD BE MOUNTED ON DRIVE #1 HAVING BEEN ASSIGNED THE NAME 'DSK:'. 'HELP' IS REQUESTED WITH THE 'L O HELP' COMMAND ('LET'S GET HELP') WHILE THE USER IS WRITING A PS/8 FOCAL PROGRAM. HE WILL BE RETURNED TO HIS PROGRAM (SAVED AS 'FOCAL TM') WHEN HE HAS HAD ENOUGH 'HELP'.

DECIMAL ASCII CODES

FOR FINK<> AND FOUT<>

CODE	CHARACTER	CODE	CHARACTER	CODE	CHARACTER
128	CTRL/SHIFT/P (LEADER)	160	SPACE	193	A
129	CTRL/A	161	!	194	B
130	CTRL/B	162	"	195	C
131	CTRL/C	163	#	196	D
132	CTRL/D	164	\$	197	E
133	CTRL/E	165	%	198	F
134	CTRL/F	166	&	199	G
135	CTRL/G (BELL)	167	'	200	H
136	CTRL/H	168	(	201	I
137	CTRL/I	169	)	202	J
138	LINE FEED	170	*	203	K
139	CTRL/K	171	+	204	L
140	CTRL/L	172	,	205	M
141	CARRIAGE RETURN	173	-	206	N
142	CTRL/N	174	.	207	O
143	CTRL/O	175	/	208	P
144	CTRL/P	176	0	209	Q
145	CTRL/Q	177	1	210	R
146	CTRL/R	178	2	211	S
147	CTRL/S	179	3	212	T
148	CTRL/T	180	4	213	U
149	CTRL/U	181	5	214	V
150	CTRL/V	182	6	215	W
151	CTRL/W	183	7	216	X
152	CTRL/X	184	8	217	Y
153	CTRL/Y	185	9	218	Z
154	CTRL/Z	186	:	219	[
155	CTRL/SHIFT/K	187	;	220	\
156	CTRL/SHIFT/L	188	<	221	]
157	CTRL/SHIFT/M	189	=	222	^
158	CTRL/SHIFT/N	190	>	223	_
159	CTRL/SHIFT/O	191	?	253	ALT MODE
		192	@	255	RUBOUT

FOUT(141) WILL OUTPUT A RETURN/LINEFEED; FOUT(13) WILL OUTPUT A CARRIAGE RETURN ONLY. CODE 144 (CTRL/P) IS PS/8 FOCAL'S BREAK CHARACTER. 224 THROUGH 250 ARE LOWER-CASE LETTERS ON SOME TERMINALS. MANY TERMINALS USE SHIFT/K, /L, AND /M FOR '[', '\', AND ']'.

CODES 0 THROUGH 127 ARE SIMILAR TO 128-255 EXCEPT THAT THE HIGH ORDER BIT OF ITS BINARY REPRESENTATION IS EQUAL TO ZERO.

PS/8 FOCAL, 1971 ERROR CODES

ERROR CODES VALID ONLY FOR 6/1/71 VERSION OF PS/8 FOCAL

page 57

CODE EXPLANATION

200. 00	MANUAL START FROM CONSOLE
201. 00	BREAK FROM KEYBOARD VIA 'CTRL/P'
201. 40	BAD LINE NUMBER
201. 91	GROUP NUMBER LARGER THAN 31
201. 13	ZERO IS ILLEGAL LINE NUMBER
202. 32	'DO' REFERENCED NON-EXISTENT GROUP
202. 52	'DO' REFERENCED NON-EXISTENT LINE
203. 05	NON-EXISTENT LINE AFTER 'GOTO' OR 'IF'
203. 28	ILLEGAL COMMAND
204. 39	BAD VARIABLE IN 'FOR' OR 'SET' COMMAND
204. 52	EXCESS RIGHT TERMINATORS
204. 60	ILLEGAL TERMINATOR IN 'FOR' OR 'SET' COMMAND
204. 13	MISSING ARGUMENT IN DISPLAY FUNCTION
205. 48	BAD ARGUMENT IN 'MODIFY' COMMAND
206. 06	ILLEGAL USE OF A FUNCTION OR NUMBER
206. 54	TOO MANY VARIABLES
207. 38	NO OPERATOR BEFORE PARENTHESIS
207. 19	FUNCTION ARGUMENT MISSING
207. 16	ILLEGAL FUNCTION OR DOUBLE OPERATORS
208. 47	ENCLOSURES DO NOT MATCH
209. 11	BAD ARGUMENT IN 'ERASE' COMMAND
210. 15	STORAGE FILLED BY TEXT
211. 35	INPUT BUFFER OVERFLOW
215. 28	BAD RESTORE COMMAND
215. 11	ATTEMPT TO WRITE PAST END-OF-FILE
220. 34	LOGARITHM OF ZERO REQUESTED
222. 23	NO OUTPUT FILE TO RESTORE
222. 51	BAD FILES 'OPEN' COMMAND
223. 05	DATA FILE NOT FOUND
223. 15	NO INPUT FILE TO RESTORE
223. 36	TOO MANY DIGITS IN NUMBER
223. 39	PUSHDOWN OVERFLOW; PROGRAM TOO LONG
224. 60	TWO PERIODS IN FILE NAME OR NO NAME
225. 37	BAD DEVICE OR 2 PAGE HANDLER
226. 18	BAD LIBRARY COMMAND
226. 58	NO PROGRAM NAMED BY 'SAVE' COMMAND
226. 99	EXPONENT TOO LARGE OR NEGATIVE
227. 10	PROGRAM NOT FOUND
227. 14	'LIBRARY CALL' FROM NON-DIRECTORY DEVICE
228. 06	'LIBRARY LIST' OF NON-DIRECTORY DEVICE
228. 73	DIVISION BY ZERO
229. 53	FILE ALREADY DELETED
229. 68	DEVICE ERROR: WRITE LOCK OR PARITY
229. 91	CANNOT OPEN OUTPUT FILE
230. 05	IMAGINARY SQUARE ROOTS REQUIRED
231. <7	UNAVAILABLE COMMAND OR FUNCTION
?	ATTEMPT TO READ PAST END-OF-FILE (INPUT IS SWITCHED TO TERMINAL AND PROGRAM CONTINUES)

NOTE: TYPE =, TYPE >, TYPE 1, OR TYPE > OUTPUTS A STRING OF ZEROES. THIS CAN BE STOPPED BY HITTING 'CTRL/P'.

SHOULD YOU EXPERIENCE A PS/8 MONITOR ERROR SUCH AS ATTEMPTING TO WRITE ON AN UNSELECTED DEVICE, IT IS BEST TO RE-ENTER FOCAL THROUGH THE INITIAL DIALOGUE.

PS/8 FOCAL, 1971

SUMMARY OF COMMANDS

*** ELEMENTARY COMMANDS:

*ASK X ACCEPT VALUE OF X FROM INPUT DEVICE
 *COMMENT IGNORE REST OF LINE
 *CONTINUE DUMMY COMMAND
 *DO [LINE OR GROUP NUMBER] SUBROUTINE CALL
 *ERASE ZERO VARIABLES
 *ERASE <LINE OR GROUP NUMBER> DELETE LINE OR GROUP
 *ERASE ALL DELETE ENTIRE INDIRECT PROGRAM
 *FOR X=E1[, E2], E3; [COMMAND] COMMAND REPEATED (E3-E1)/E2+1 TIMES
 *GO START PROGRAM EXECUTION AT FIRST LINE
 *GOTO <L1> SEND PROGRAM EXECUTION TO LINE L1
 *IF (E1)>L1[, L2, L3]; -, 0, OR + VALUE OF E1 GIVES NEXT LINE EXECUTED
 *MODIFY L1 EDIT LINE L1
 *QUIT STOP EXECUTION
 *RETURN TERMINATE A 'DO' OR 'LIBRARY GOSUB' SUBROUTINE
 *SET X=<NUMBER, VARIABLE OR EXPRESSION> ASSIGN VALUE TO VARIABLE
 *TYPE [NUMBERS, VARIABLES, EXPRESSIONS, TEXT IN QUOTES, !, #, %, \$ OR %]
 *WRITE [GROUP OR LINE NUMBER] LIST PROGRAM

*** LIBRARY COMMANDS:

*LIBRARY CALL [DEVICE:]<PROGRAM NAME> LOAD SAVED PROGRAM
 *LIBRARY DELETE [DEVICE:]<PROGRAM NAME> UNSAVE PROGRAM
 *LIBRARY GOSUB [DEVICE:]<PROGRAM NAME> [GROUP OR LINE NUMBER] 'DO'
 *LIBRARY LIST [DEVICE:]<FIRST FILE TO BE LISTED> CATALOG
 *LIBRARY RUN [DEVICE:]<PROGRAM NAME> [LINE NUMBER] CHAIN
 *LIBRARY SAVE [DEVICE:]<PROGRAM NAME> SAVE PROGRAM
 *LIBRARY EXIT RETURN TO PS/8 MONITOR

*** FILE COMMANDS:

*OPEN INPUT [DEVICE:]<FILE NAME>[, ECHO] PREPARE TO READ FILE
 *OPEN OUTPUT [DEVICE:]<FILE NAME>[, ECHO] PREPARE TO WRITE FILE
 *OPEN RESTORE INPUT[, ECHO] RESUME INPUT FROM FILE
 *CLOSE RESTORE OUTPUT[, ECHO] RESUME OUTPUT TO FILE
 *OUTPUT CLOSE OUTPUTS BUFFER; CLOSES OUTPUT FILE
 ---INPUT---, ECHO INPUT IS ECHOED ON OUTPUT DEVICE
 ---OUTPUT---, ECHO OUTPUT IS ECHOED ON TERMINAL

*** GENERAL FORM:

<> ENCLOSE REQUIRED TERMS AND [] ENCLOSE OPTIONAL TERMS.
 X IS A VARIABLE. E1, E2 AND E3 ARE ARITHMETIC EXPRESSIONS.
 L1, L2, AND L3 ARE LINE NUMBERS. LINE AND GROUP NUMBERS MAY
 BE REPLACED BY ANY EXPRESSION NOT BEGINNING WITH THE LETTER 'A'.
 .FC AND .FD ARE ASSUMED PROGRAM AND DATA FILE NAME EXTENSIONS.

FOLLOW COMMANDS WITH A SPACE. ONE LETTER ABBREVIATIONS PERMITTED.

CTRL/P IS PS/8 FOCAL'S BREAK CHARACTER


```
1          / IKT-TUB FOCAL ( 2. APRIL 1981 )
2
3
4
5
6
7
8
9          / IKT-TUB FOCAL '81
10         / *****
11
12         /
13         / OVERLAY FOR PS/3 FOCAL
14
15
16
17
18
19         / COMPUTER:   DIGITAL EQUIPMENT CORP.
20         /             PDP-3/E OR PDP-3/I
21         /
22         / ASSEMBLER:   PAL3
23         /
24         / MINIMUM HARDWARE TO RUN THIS PROGRAM:
25         /             PDP-3 WITH 12 K MEMORY
26         /             KEYBOARD READER, TELETYPE
27         /
28         / MAXIMUM HARDWARE:
29         /             PDP-3 WITH 23 K MEMORY
30         /             KEYBOARD READER, TELETYPE
31         /             DECTAPE AND/OR DISK
32         /             HIGH-SPEED-READER AND -PUNCH
33         /             (ALL PS/3 FOCAL HARDWARE)
34         /             EXTENDED ARITHMETIC ELEMENT
35         /             AX23 A/D-CONVERTER (LAB-3)
36         /             AF24 A/D-CONVERTER
37         /             KL3-JA INTERFACE
38         /             KV31 DISPLAY
39         /             DK3-BF PROGRAMMABLE
40         /             REAL TIME CLOCK
41         /
42         / SOFTWARE:    PS/3 FOCAL, 1971
43         /             DECUS NO. FOCAL3-177
```

44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

/ THIS OVERLAY WILL CONTROL WITHIN PS/3-FOCAL
/ (NOW IKT-TUB FOCAL '31) THE HANDLING OF DATA
/ FILES ("FX"-FUNCTION), THE AX23 A/D-CONVERTER
/ ("FLAB"-, "FADC(K)"-FUNCTIONS), THE
/ AF24 A/D-CONVERTER ("FADC(K,F,R,S)"-FUNCTION),
/ THE KV31-DISPLAY ("FD"-FUNCTION), THE
/ KL3-JA INTERFACE ("FIO"-FUNCTIONS) AND
/ THE DK3-EF PROGRAMMABLE REAL TIME CLOCK
/ ("FX(3,0)"-, "FX(4)"-FUNCTIONS).
/ SOME NEW FUNCTIONS "FX" ARE ADDED. THE OLD
/ DISPLAY FUNCTION "FDIS" IS AVAILABLE
/ TOO.
/
/ THE PROGRAM WILL CHECK THE COMPUTER-HARDWARE
/ AND WILL MODIFY ITSELF.

/ WRITTEN BY:
/ KLAUS LICKTEIG
/ INSTITUT FÜR KERntechnik
/ TECHNISCHE UNIVERSITÄT BERLIN
/ MARCHSTRASSE 13
/ 1000 BERLIN 10

31
32
33
34
35 / TO BUILD THIS OVERLAY, THE FOLLOWING SOFTWARE
36 / WAS USED:
37
38
39
40
41
42 / SCHNEIDER, D; SMITH, B.:
43 / PS/3 FOCAL, 1971
44 / DECUS PROGRAM LIBRARY FOCAL3-177
45 /
46 / VAN ZEE, J.:
47 / IMPROVED EAE ROUTINE FOR FOCAL
48 / DECUS PROGRAM LIBRARY FOCAL3-270
49 /
50 / MOSS, L.:
51 / FK FUNCTION FOR RANDOM ACCESS FILES
52 / DECUS PROGRAM LIBRARY FOCAL3-263
53 /
54 / EPICKSON, M.B.:
55 / OVERLAY FOR KV31 - OHSI FOCAL 1971
56 / DECUS PROGRAM LIBRARY FOCAL3-312
57 /
58 / LICKTEIG, K.:
59 / FREQUENCY TRANSFORMATION PROGRAM
60 / DECUS PROGRAM LIBRARY FOCAL3-226
61 /
62 / LICKTEIG, K.:
63 / ADDITIONAL FUNCTIONS TO CONTROL THE
64 / AK23 A/D-CONVERTER
65 / DECUSCOPE 14(1975)3, P. 61-67

```
116
117
118 / 1. LOADING PROCEDURE
119 /
120 /
121 /
122 /
123 / LOAD PS/3 FOCAL (DECUS NO.: FOCAL3-177) AND
124 / THIS OVERLAY WITH AN OS/3 SYSTEM OR WITH THE
125 / BINARY LOADER VIA THE HIGH SPEED READER.
126 /
127 / START ADDRESS: 10200
128 / *****
129
130
131
132
133
134 / II. PROGRAM START AND INITIALIZATION
135 /
136 /
137 /
138 /
139 / THE PROGRAM WILL START THE PS/3 FOCAL INITIAL
140 / DIALOGUE. AFTER YOUR REPLY THE PROGRAM WILL
141 / CHECK THE COMPUTER HARDWARE:
142 /
143 / 1) EAE ?
144 / YES: LOAD THE FOCAL PATCHES
145 / FOR 3/1 EAE
146 /
147 / 2) AX23 ?
148 / NO: DELETE FUNCTIONS "FLAB"
149 / AND "FADCK(K)"
150 /
151 / 3) AF24 ?
152 / NO: DELETE FUNCTION
153 / "FADCK(K,F,P,S)"
154 /
155 / 4) KL3-JA ?
156 / NO: DELETE FUNCTIONS "FIO"
157 /
158 / 5) KV31 ?
159 / NO: DELETE FUNCTION "FD"
160 / SCOPE COMMANDS INACTIVE
161 /
162 / 6) DK3-EF ?
163 / NO: DELETE FUNCTIONS
164 / "FN(3,C)" AND "FN(4)"
165 /
166 / 7) FILE COMMANDS ?
167 / NO: DELETE FUNCTION "FX"
```

163
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215

/ III. NEW COMMANDS

/ SCOPE COMMANDS:

/ * WORKS ONLY IN SCOPE MODE

/ CTPL/A- HOME(UPPER LEFT HAND CORNER)
/ CTPL/B- * LINE FEED
/ CTPL/E- * CAUSES END OF PAGE CONDITION
/ CTPL/H- * BACK SPACE
/ CTPL/N- * UP SPACE
/ CTPL/O- * CARPAGE RETURN
/ CTPL/C- INCREMENT SIZE
/ CTPL/P- RESET SIZE
/ CTPL/S- CHANGE TO SCOPE MODE
/ CTPL/T- CHANGE TO TTY MODE
/ CTPL/U- ERASE SCREEN & RETURN HOME
/ CTPL/X- ERASE SCREEN

/ *A - FOUT(01) - HOME
/ *B - FOUT(02) - * LINE FEED
/ *E - FOUT(05) - * END OF PAGE
/ *H - FOUT(03) - * BACK SPACE
/ *N - FOUT(14) - * UP SPACE
/ *O - FOUT(15) - * CARPAGE RETURN
/ *C - FOUT(17) - INCREMENT SIZE
/ *P - FOUT(13) - RESET SIZE
/ *S - FOUT(19) - SCOPE MODE
/ *T - FOUT(22) - TTY MODE
/ *U - FOUT(21) - ERASE & HOME
/ *X - FOUT(24) - ERASE

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

/ IV. NEW FOCAL COMMANDS

/ FX BUILD AND HANDLE DATA FILES IN A
RANDOM FASHION/ FX(1,A) STORE THE VALUE "A"
/ FX(1) RETRIEVE THE VALUE/ FLAB CONTROL THE AM03 A/D-CONVERTER, THE
CLOCKS, THE OUTPUTS AND INPUTS OF
THE LAB-3 SYSTEM/ FLAB(0) CLEAR FLAGS, REGISTERS
/ FLAB(1,K1) PLOT WITH X-Y-RECORDER
/ FLAB(2,K1) PULSE AT RELAY-OUTPUTS
SQ, P1, P2, P3/ FLAB(3,K1,K2) SWITCH RELAY-OUTPUTS
SQ, P1, P2, P3 ON/OFF/ FLAB(4,K1,K2) SWITCH PC- OR EXTERNAL-
CLOCK ON/OFF/ FLAB(5) WAIT FOR PC- OR EXTERNAL-
CLOCK FLAG/ FLAB(6,K1,K2) WAIT FOR K2 PULSES OF
THE SPECIFIED CLOCK/ FLAB(7,K1) SYNCHRONIZE PC- OR
EXTERNAL-CLOCK/ FLAB(8,K1) READ/CLEAR EXTERNAL
SENSE REGISTER

/ FLAB(9) FREE

/ FLAB(10) FREE

```

257
258
259
260
261
262      / FADC  CONTROL THE AX23 AND/OR AF24
263      /      A/D-CONVERTER
264      /
265      /      FADC(X)      AX23 A/D-CONVERSION
266      /      FADC(X, F, P, S)  AF24 A/D-CONVERSION
267
268
269
270
271
272      / FN      NEW FUNCTIONS
273      /
274      /      FN(0,0,NU)      ROUTINE TO SERVE A FFT
275      /      FN(1,1)      GET DATA FROM BUFFER
276      /      IN FIELD 2
277      /      FN(1,1,A)      STORE DATA INTO BUFFER
278      /      IN FIELD 2
279      /      FN(2,1)      GET DATA FROM BUFFER
280      /      IN FIELD 3 FF
281      /      FN(2,1,A)      STORE DATA INTO BUFFER
282      /      IN FIELD 3 FF
283      /      FN(3,C)      START PROGRAMMABLE
284      /      REAL TIME CLOCK
285      /      FN(4)      READ NUMBER OF CLOCK
286      /      FLAGS
287      /
288      /      FN(5)      FREE
289      /      FN(6)      FREE
290      /      FN(7)      FREE
291      /      FN(8)      FREE
292      /      FN(9)      FREE
293      /      FN(10)     FREE
294
295
296
297
298
299      / FE      SCOPE FUNCTION FOR THE KV31-DISPLAY
300      /
301      /      FE(1,X,Y)      DISPLAY A POINT,
302      /      VECTOR OR CIRCLE

```

303			
304			
305			
306			
307	/ FIO	RECEIVE OR TRANSMIT DATA VIA THE	
308	/	KL3-JA INTERFACE	
309	/		
310	/	FIO(-1,1)	CHANGE OUTPUT FOR
311	/		TRANSMIT DATA
312	/	FIO(0,1)	GET A RECEIVED DATA
313	/		INTO THE FL.F.AC.
314	/	FIO(1,1)	RECEIVE DATA
315	/		
316	/	FIO(2,K1)	FREE
317	/	FIO(3,K1)	FREE
318	/	FIO(4,K1)	FREE


```
319
320
321
322
323 /      FUNCTION      FX
324 /      *****
325
326
327
328
329
330
331
332 /      TAPE - OVERLAY TO PS/3 FOCAL.
333
334 /      SUBROUTINE BY LAWRENCE MOSS
335 /      FX FUNCTION FOR RANDOM ACCESS FILES
336 /      DECUS PROGRAM LIBRARY FOCAL3-263
337
338
339 /      THE FUNCTION FX IS USED AS FOLLOWS:
340 /      TO STORE A VALUE INTO THE BUFFER USE
341 /
342 /      SET D=FX(I,A)
343 /      *****
344 /
345 /      WHERE "D" IS A DUMMY VARIABLE, "I" IS THE
346 /      MATRIX LOCATION TO BE USED, AND "A" IS THE
347 /      VARIABLE TO BE STORED INTO LOCATION "I".
348 /
349 /      TO RETRIEVE A VALUE FROM THE BUFFER USE
350 /
351 /      SET X=FX(I)
352 /      *****
353 /
354 /      WHERE VARIABLE "X" IS ASSIGNED THE VALUE OF
355 /      ARRAY ELEMENT "I".
356
357
358
359 /      FOURWD=0  ASSEMBLES FOR 3 WORD VARIABLES AND
360 /                34 VARIABLES PER BLOCK
361 /      FOURWD=1  ASSEMBLES FOR 4 WORD VARIABLES AND
362 /                63 VARIABLES PER BLOCK
363
364
365 /      TO BUILD AN OUTPUT FILE, THE FILE NAMED MUST
366 /      FIRST BE OPENED FOR OUTPUT BY USING THE PS/3
367 /      FOCAL 'O O NAME' INSTRUCTION. EACH VARIABLE MAY
368 /      THEN BE STORED IN RANDOM OR SEQUENTIAL
369 /      PATTERN USING THE FORM      FX(I,A).
```

370
371
372
373
374 / DATA MAY ALSO BE RETRIEVED AT THE SAME TIME
375 / BUT TO MAKE THE FILE PERMANENT, IT MUST BE
376 / CLOSED WITH THE 'O C' COMMAND. NOTE THAT ANY
377 / TYPING BY THE PROGRAM WILL GO INTO AND
378 / DESTROY THE CONTENTS OF THE DATA FILE
379 / UNLESS THE PROGRAM FIRST OPENS OUTPUT TO
380 / THE TTY TO RESTORE THAT DEVICE. HOWEVER IT
381 / IS NOT NECESSARY TO RESTORE OUTPUT TO
382 / THE FILE DEVICE, AND 'O O TTY:' CAN BE
383 / ISSUED IMMEDIATELY AFTER THE COMMAND TO
384 / OPEN THE DATA FILE.
385 /
386 / TO USE AN ALREADY EXISTING FILE, THE FILE
387 / MUST FIRST BE OPENED FOR INPUT (!!) USING
388 / THE 'O I NAME' COMMAND. THIS ESTABLISHES
389 / THE FILE OPEN OPERATION WITHOUT ERASING
390 / THE EXISTING FILE AND BRINGS THE REQUIRED
391 / DEVICE HANDLER INTO THE MEMORY.
392 / BEFORE ATTEMPTING TO USE THE FILE, YOU
393 / MUST (!!) CALL TO FX WITH AN ARGUMENT
394 / OF ZERO! THIS CALL TRANSFERS THE FILE
395 / ADDRESS PARAMETERS AND DEVICE HANDLER
396 / POINTERS FROM INPUT TO OUTPUT. INPUT
397 / AND OUTPUT MAY THEN BE RESTORED TO THE
398 / TELETYPE (NOT A PS/3 DEVICE) AND FX
399 / USED AS DESCRIBED ABOVE. DATA CAN BE
400 / INPUT AND OUTPUT AS LONG AS YOU DO NOT
401 / ATTEMPT TO EXCEED THE LENGTH OF YOUR
402 / FILE, IN WHICH CASE AN ERROR MESSAGE
403 / WILL BE GENERATED (13.45).
404 / (WHEN BUILDING A DATA FILE YOU
405 / CAN PAD IN ROOM FOR EXTRA DATA BY MAKING
406 / IT LONGER THAN NECESSARY. E.G. STOPPING
407 / FX(2047,0) AS THE LAST INSTRUCTION
408 / BEFORE CLOSING, MAKING THE FILE
409 / THE MAXIMUM SIZE ALLOWABLE).
410 /
411 / DO NOT CLOSE AN EXISTING FILE. WHEN ADDING
412 / TO OR CHANGING AN EXISTING FILE, TO
413 / INSURE THAT THE LAST CHANGE IS SAVED,
414 / READ IN A VALUE WITH A POSITION 100
415 / LARGER OR SMALLER THAN THE LAST TO
416 / INSURE THAT THE LAST CHANGE IS
417 / SWAPPED OUT OF MEMORY ONTO THE FILE
418 / DEVICE.

```

419
420
421
422
423
424      /      FUNCTION      FLAB
425      /      *****
426
427
428
429
430
431
432      / ROUTINE TO CLEAR THE EXTERNAL SENSE REGISTER,
433      / TO DISABLE THE ENABLE REGISTER AND TO CLEAR
434      / THE FLAGS OF THE AX23 A/D-CONVERTER
435      /
436      / FOCAL FUNCTION:
437      /
438      /      SET DUMMY=FLAB(0)
439      /      *****
440
441
442
443
444
445      / ROUTINE TO CONTROL THE PEN LIFT OF A
446      / X-Y-RECORDER WITH/WITHOUT NULL-DETECTOR.
447      /
448      / WITH NULL-DETECTOR: CHANNEL "S0" WILL GIVE
449      / THE "SEEK-PULSE". THE "PLOT-COMPLETE-
450      / PULSE" OF THE RECORDER MUST BE NEGATIVE
451      / AND SEND TO THE "EXTERNAL-CLOCK-INPUT".
452      /
453      / WITHOUT NULL-DETECTOR: CHANNEL "S0" GIVES A
454      / PULSE TO CONTROL THE PEN LIFT. THE
455      / SPEED IS CONTROLLED BY THE RC-CLOCK.
456      /
457      / FOCAL FUNCTION:
458      /
459      /      SET DUMMY=FLAB(1,K1)
460      /      *****
461      /
462      /      K1=0      WITH NULL-DETECTOR
463      /      1      WITHOUT NULL-DETECTOR

```

```
464
465
466
467 / ROUTINE TO SEND A PULSE TO OUTPUTS
468 / S0, R1, R2, R3; PULSE LENGTH IS
469 / 12(S0) OR 7 (R1...R3) CYCLE TIMES.
470 /
471 / FOCAL FUNCTION:
472 /
473 /     SET DUMMY=FLAB(2,K1)
474 /     *****
475 /
476 /     K1=1,...,15
477
478
479
480
481 / ROUTINE TO SWITCH THE OUTPUTS S0, R1, R2, R3
482 / ON/OFF
483 /
484 / FOCAL FUNCTION:
485 /
486 /     SET DUMMY=FLAB(3,K1,K2)
487 /     *****
488 /
489 /     K1=1,...,15
490 /         1 - R1
491 /         2 - R2
492 /         4 - R3
493 /         3 - S0
494 /         OR ANY COMBINATION
495 /     K2=0 DISABLE
496 /         1 ENABLE
497
498
499
500
501 / ROUTINE TO SWITCH THE RC-CLOCK OR THE
502 / EXTERNAL-CLOCK ON/OFF.
503 /
504 / FOCAL FUNCTION:
505 /
506 /     SET DUMMY=FLAB(4,K1,K2)
507 /     *****
508 /
509 /     K1=1 RC-CLOCK, SLOW
510 /         2 RC-CLOCK, FAST
511 /         4 EXTERNAL-CLOCK
512 /     K2=0 DISABLE
513 /         1 ENABLE
```



```
514
515
516
517 / ROUTINE TO WAIT ON A RC-CLOCK- OR
518 / EXTERNAL-CLOCK-PULSE
519 /
520 / FOCAL FUNCTION:
521 /
522 /     SET DUMMY=FLAB(5)
523 /     *****
524
525
526
527
528
529
530 / ROUTINE TO WAIT FOR K2 PULSES OF THE
531 / SPECIFIED CLOCK OF INPUT.
532 /
533 / FOCAL FUNCTION:
534 /
535 /     SET DUMMY=FLAB(6,K1,K2)
536 /     *****
537 /
538 /     K1=1 RC-CLOCK, SLOW
539 /         2 RC-CLOCK, FAST
540 /         3 CRYSTAL-CLOCK
541 /         4 EXTERNAL-CLOCK
542 /     K2=1, ..., 4095
543 /     COUNTER
544
545
546
547
548
549 / ROUTINE TO SYNCHRONIZE AND COUNT THE RC-CLOCK
550 / OF EXTERNAL CLOCK TIME. THE TIME IS HELD IN
551 / THE FL.P.AC..
552 /
553 / FOCAL FUNCTION:
554 /
555 /     SET TI=FLAB(7,K1)
556 /     *****
557 /
558 /     K1=CLOCK FLAG COUNTER
559 /     K1=1, ..., 2247
560 /     K1*(CLOCK TIME) MUST BE
561 /     SMALLER THAN 1677.7216 SEC
562 /     TI=TIME OF CLOCK
563 /
564 /     TIME=(TI/K1)*0.1 MSEC
```

```
564
565
566
567
568 / ROUTINE TO READ THE EXTERNAL SENSE REGISTER
569 / AND TO CLEAR THE BITS GIVEN WITH K1.
570 /
571 / FOCAL FUNCTION:
572 /
573 /     SET A=FLAB(3,K1)
574 /     *****
575 /
576 /     K1=Z,....,2247
577 /     BITS GIVEN AS A DECIMAL
578 /     NUMBER ARE CLEARED
579 /     A: BITS OF THE EXTERNAL
580 /     SENSE REGISTER AS A
581 /     DECIMAL NUMBER
582 /
583
584
585
586
587 / THIS FUNCTION ARE FREE
588 /
589 /     FLAB(9)
590 /     FLAB(12)
591 /
592 / IN LOCATION NEWLAE=22523 (FF) YOU HAVE TO
593 / STORE A POINTER TO YOUR NEW FUNCTION.
594 / YOU HAVE TO MOVE LOCATION POINTER
595 / (POINTER = END OF PROGRAM IN FIELD 2):
596 / STORE THE NEW POINTER INTO LOCATION
597 / IPOIN=22113 AND INTO POINT=22114 THE
598 / VALUE OF POINTER-21.
599 /
600 / FOR RETURN TO FOCAL, MAKE A
601 /     JMP PFOCAL = 5225      IN FIELD 2
602 / OP
603 /     JMP I EFUN3I = 5536    IN FIELD 1
```



```

635
636
637
638 / ROUTINE FOR ANALOG TO DIGITAL CONVERSION VIA
639 / THE AF24 A/D-CONVERTER
640 /
641 / FOCAL FUNCTION:
642 /
643 / SET A=FADC(K,F,P,S)
644 / *****
645 /
646 / K = CHANNEL-NUMBER
647 / F = FUNCTION
648 / P = RANGE
649 / S = RESOLUTION
650 /
651 / CHANNEL-NUMBER "K": 0 - 299
652 /
653 / (3) (12)
654 / FUNCTION "F": AC-VOLTAGE = 400 256
655 / DC-VOLTAGE = 200 123
656 / RESISTANCE = 100 64
657 / FREQUENCY = 40 32
658 / PERIOD = 20 16
659 / SCAN RESET = 10 3
660 / SWITCH CLOSURE = 4 4
661 /
662 / RANGE "P": 10 MV = 1
663 / 100 MV = 2
664 / 1000 MV = 3
665 / 10 V / 10K = 4
666 / 100 V / 100K = 5
667 / 1000 V = 6
668 / AUTO RANGE = 7
669 /
670 / RESOLUTION "S": 0.1 % = 1
671 / 0.01 % = 2
672 / 0.001 % = 3
673 /
674 / THE VARIABLE "A" CONTAINS THE DECIMAL
675 / DATA.

```



```
706
707
708
709
710
711 / ROUTINE TO GET/STORE DATA FROM/INTO
712 / THE DATA BUFFER OF FIELD 2
713 /
714 / FOCAL FUNCTION:
715 /
716 /     SET A=FN(1,I)
717 /     *****
718 /                                     /DATA AT PLACE "I"
719 /                                     /IS HELD IN "A"
720 /     SET DUMMY=FN(1,I,A)
721 /     *****
722 /                                     /DATA "A" IS STORED
723 /                                     /AT PLACE "I"
724 /
725 /     I=2,1,...,593 /FOURWD=1
726 /                                     / 4-WORD-PRECISION
727 /
728 /     I=2,1,...,799 /FOURWD=2
729 /                                     / 3-WORD PRECISION
730 /
731 / THE DATA OF THIS FN-FUNCTION USE THE
732 / SAME MEMORY PART AS THE FIO-FUNCTION !!
733 / THE DATA OF THE FIO-FUNCTION ARE STORED
734 / IN 2-WORD-PRECISION; E.G. IF YOU HAVE
735 / 200 DATA WITHIN THE FIO-FUNCTION, YOU CAN
736 / START THIS FN-FUNCTION WITH I=100
737 / (4-WORD-PRECISION)
738 /
739 / THE LAST PAGE OF FIELD 2 IS PROTECTED
```

```

740
741
742
743 / ROUTINE TO GET/STORE DATA FROM/INTO
744 / FIELD 3 FF.
745 /
746 / FOCAL FUNCTION:
747 /
748 / SET A=FN(2,I)
749 / *****
750 / /DATA AT PLACE "I"
751 / /IS HELD IN "A"
752 / SET DUMMY=FN(2,I,A)
753 / *****
754 / /DATA "A" IS STORED
755 / /AT PLACE "I"
756 /
757 / /FOURND=1
758 / I=0,1,...,1023 /FIELD 3
759 / =1024,...,2047 /FIELD 4
760 / =2048,...,3071 /FIELD 5
761 / =3072,...,4095 /FIELD 6
762 /
763 / /FOURND=0
764 / I=0,1,...,1364 /FIELD 3
765 / I(MAX)=4095 /ETC.

```

```
766
767
763
769
770
771 / ROUTINE TO SET THE PROGRAMMABLE REAL TIME
772 / CLOCK DK3-EP TO ZERO AND TO START IT.
773 /
774 / FOCAL FUNCTION:
775 /
776 / SET DUMMY=FN(3,C)
777 / *****
778 /
779 / C = CLOCK TIME FLAG AT
780 / "C" MILLISECONDS
781 / C = 1, ..., 4096
782
783
784
785
786
787
788
789 / ROUTINE TO READ THE NUMBER OF CLOCK FLAGS OF
790 / THE DK3-EP PROGRAMMABLE REAL TIME CLOCK.
791 /
792 / FOCAL FUNCTION:
793 /
794 / SET TI=FN(4)
795 / *****
796 /
797 / THE VARIABLE "TI" CONTAINS THE NUMBER OF
798 / CLOCK FLAGS.
799 /
800 / REAL TIME = TI * C [MILLISECONDS]
801 /
802 / WITH E.G. C=100 YOU GET A CLOCK FLAG AT
803 / EVERY 100 MILLISECONDS, THE MAXIMUM TIME
804 / WILL BE 33336.00 SECONDS OR
805 / 23 HOURS 13 MINUTES AND 6.00 SECONDS.
```


306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329

/ THIS FUNCTIONS ARE FREE:

/
/ FN(5)
/ FN(6)
/ FN(7)
/ FN(8)
/ FN(9)
/ FN(10)

/ IN LOCATIONS NEWFN=20372 (FF) YOU HAVE
/ TO STORE A POINTER TO YOUR NEW FUNCTION
/ YOU HAVE TO MOVE LOCATION POINTER
/ (POINTER = END OF PROGRAM IN FIELD 2):
/ STORE THE NEW POINTER INTO LOCATION
/ IPOIN=20113 AND INTO POINT=20114 THE
/ VALUE OF POINTER-2

/ FOR RETURN TO FOCAL, MAKE A
/ JMP FFOCAL = 5005 IN FIELD 2
/ OF
/ JMP I EFUN31 = 5536 IN FIELD 1

330

331

332

333

334

335

/ FUNCTION FD

336

/ *****

337

338

339

340

341

342

/ SCOPE FUNCTIONS

343

344

345

346

347

/ OVERLAY FOR KUBI-OSMI FOCAL 71

348

349

350

/ SUBROUTINE BY MICHAEL B. EPICKSON

351

/ OVERLAY FOR KUBI-OSMI FOCAL 71

352

/ DECUS PROGRAM LIBRARY FOCAL3-312

353

354

355

356

357

/ THIS PATCH PROVIDES OSMI FOCAL

358

/ THE ABILITY TO GENERATE

359

/ CHARACTERS ON A KUBI SCREEN,

360

/ BY USING THE VSCG. THE

361

/ ROUTINE ARE ALSO INSERTED

362

/ INTO THE FUNCTION TABLE TO BE

363

/ USE ON A KUBI SCREEN.

364

365

366

367

/ THE OLD FOCAL FUNCTION "FDIS(X,Y)"

368

/ IS ACTIVE TOO.

369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915

/ KUBI-DISPLAY FUNCTION

/ FOCAL FUNCTION:

/ SET DUMMY=FD(I,X,Y)

/ *****

/ I-PEN VALUE

/ 1 : DRAW A VISIBLE VECTOR
/ TO LOCATION X & Y/ 2 : DRAW AN INVISIBLE VECTOR
/ TO LOCATION X & Y

/ -1 : DRAW A POINT AT LOCATION X & Y

/ -N : DRAW A VISIBLE CIRCLE AROUND
/ LOCATION X & Y

/ WITH RADIUS N,

/ N < -1

/ X- X COORDINATE (HORIZONTAL)

/ Y- Y COORDINATE (VERTICAL)

/ * WORKS ONLY IN SCOPE MODE

/ +A - FOUT(01) - HOME

/ +B - FOUT(02) - * LINE FEED

/ +E - FOUT(05) - * END OF PAGE

/ +H - FOUT(03) - * BACK SPACE

/ +N - FOUT(14) - * UP SPACE

/ +O - FOUT(15) - * CARRIAGE RETURN

/ +G - FOUT(17) - INCREMENT SIZE

/ +R - FOUT(13) - RESET SIZE

/ +S - FOUT(19) - SCOPE MODE

/ +T - FOUT(20) - TTY MODE

/ +U - FOUT(21) - ERASE & HOME

/ +X - FOUT(24) - ERASE

```

916
917
918
919
920      /      FUNCTION      FIO
921      /      *****
922
923
924
925
926      / WITH THIS FUNCTION YOU CAN RECEIVE AND
927      / TRANSMIT DATA VIA THE KL3-JA INTERFACE.
928      /
929      / THE ROUTINES ARE BUILT TO SPEAK WITH THE
930      / CANBERRA MULTICHANNEL ANALYZER 3100.
931      / BY CHANGING THE OUTPUT FROM TTY TO KL3-JA,
932      / A SIGN "212" (LOCATION C212=22630) IS
933      / SENT; IT STARTS THE CANBERRA 3100.
934      / AT THE END OF TRANSMIT A SIGN "170"
935      / (LOCATION C170=22627) IS SENT.
936
937
938
939
940
941      / ROUTINE FOR OUTPUT DURING A FOCAL
942      / TYPE COMMAND.
943      /
944      / FOCAL FUNCTION:
945      /
946      /      SET DUMMY=FIO(-1,A)
947      /      *****
948      /
949      /      A = 0   OUTPUT ONTO TTY
950      /      A > 0   OUTPUT TO KL3-JA
951      /      TRANSMIT DATA FOR
952      /      INPUT INTO THE
953      /      CANBERRA MULTI-
954      /      CHANNEL ANALYZER
955      /
956      / EXAMPLE:
957      /      SET DU=FIO(-1,1)
958      /      FOR I=1,N; T %6.20,A(I)
959      /      SET DU=FIO(-1,0)

```



```
960
961
962
963
964
965 / ROUTINE TO GET AND FLOAT A DATA OF THE
966 / DATA BUFFER INTO THE FL.P.AC..
967 /
968 / FOCAL FUNCTION:
969 /
970 /     SET A=FIO(2,1)
971 /     *****
972 /                                     I > 2
973 /
974 / THE VARIABLE "A" CONTAINS THE DATA.
975
976
977
978
979
980
981
982 / ROUTINE FOR INPUT DATA DURING FOCAL VIA THE
983 / KL3-JA INTERFACE: RECEIVE DATA.
984 /
985 / FOCAL FUNCTION:
986 /
987 /     SET D=FIO(1,1)
988 /     *****
989 /                                     I > 2
990 /
991 / AFTER INPUT THE NUMBER OF DATA CONTAINS
992 / VARIABLE "D"; THAT IS:  "I"  OF THE MAXIMUM
993 / POSSIBLE NUMBER, I(MAX)=1199.
994 /
995 / THIS FUNCTION STORES THE DATA AS A UNSIGNED
996 / 2-WORD-PRECISION NUMBER. THE DATA ARE
997 / ALWAYS STORED AT THE PLACES I=1,2,...,I(MAX).
998 /
999 / THE FN-FUNCTION  FN(1,1,A) USES THE SAME
/ LOCATIONS !!
```

```
1000
1001
1002
1003
1004      / THIS FUNCTIONS ARE FREE:
1005      /
1006      /      FIO(2,K1)
1007      /      FIO(3,K1)
1008      /      FIO(4,K1)
1009      /
1010      / IN THE LOCATIONS NEWFIO=22573 (FF) YOU HAVE
1011      / TO STORE A POINTER TO YOUR NEW FUNCTION
1012      / (IT MUST HAVE THE 1. APG. AND 2. APG. K1).
1013      / YOU HAVE TO MOVE LOCATION POINTER
1014      / (POINTER = END OF PROGRAM IN FIELD 2):
1015      / STORE THE NEW POINTER INTO LOCATION
1016      / IFOIN=22113 AND INTO POINT=22114 THE
1017      / VALUE OF POINTER-2
1018      /
1019      / FOR RETURN TO FOCAL, MAKE A
1020      /      JMP PFOCAL = 5205      IN FIELD 2
1021      / OR
1022      /      JMP I EFUNCI = 5536      IN FIELD 1
```

1023

1024

1025

1026

1027

1028

1029

/ DEFINITIONS FOR INSTRUCTIONS.

1030

1031

1032

1033

1034

1035

1036

1037

/ DEFINITIONS FOR THE SAE.

1038

1039

7501 MGA=7501

/ INCLUSIVE OF THE AC WITH THE AC

1040

7421 MGL=7421

/ LOAD MC FROM AC, THEN CLEAR AC

1041

7415 ASF=7415

/ ARITHMETIC SHIFT RIGHT

1042

7405 MUY=7405

/ MULTIPLY

1043

7407 DVI=7407

/ DIVIDE

1044

7411 NMI=7411

/ NORMALIZE

1045

7413 SHL=7413

/ SHIFT LEFT

1046

7417 LSR=7417

/ LOGICAL SHIFT RIGHT

1047

7441 SCA=7441

/ READ SC INTO AC

1048

7621 CAM=7621

/ CLEAR AC AND MC

1049

7403 SCL=7403

/ LOAD THE STEP COUNTER

1050

1051

1052

1053

1054

1055

/ DEFINITIONS FOR THE DK3-EP

1056

/ PROGRAMMABLE REAL TIME CLOCK.

1057

1058

6131 CLSK=6131

/ SKIP ON CLOCK INTERRUPT

1059

6133 CLAB=6133

/ (AC) TO CLOCK BUFFER

1060

6130 CLZE=6130

/ CLEAR CLOCK ENABLE REG. PER AC

1061

6132 CLDE=6132

/ SET CLOCK ENABLE REGISTER PER AC

1062

6134 CLEN=6134

/ LOAD CLOCK ENABLE REGISTER

1063

6135 CLSA=6135

/ CLOCK STATUS TO AC

1064

6136 CLBA=6136

/ CLOCK BUFFER TO AC

1065

6137 CLCA=6137

/ CLOCK COUNTER TO AC

1266

1267

1268

1269

1270

/ DEFINITIONS FOR THE AF04 A/D-CONVERTER

1271

6542 VSEL=6542

/ SELECT FUNCTION, RANGE AND GATE

1272

6541 VCONV=6541

/ SELECT CHANNEL AND CONVERT

1273

6544 VINX=6544

/ INDEX CHANNEL AND CONVERT

1274

6561 VSDF=6561

/ SKIP ON DATA READY

1275

6562 VPL=6562

/ READ DATA AND CLEAR FLAG

1276

6564 VBA=6564

/ BYTE ADVANCE

1277

6571 VSOC=6571

/ SAMPLE CURRENT CHANNEL

1278

1279

1280

1281

1282

1283

1284

/ DEFINITIONS FOR THE AX03 A/D-CONVERTER.

1285

6301 DXC=6301

/ CLEAR X REGISTER

1286

6302 DXL=6302

/ LOAD X REGISTER

1287

6311 DYC=6311

/ CLEAR Y REGISTER

1288

6312 DYL=6312

/ LOAD Y REGISTER

1289

6304 DIS=6304

/ DISPLAY ON OSCYLOSCOPE

1290

6324 DSB=6324

/ SET BRIGHTNESS

1291

6341 SKPK=6341

/ SKIP ON PC CLOCK TIMING FLAG

1292

6321 SKKK=6321

/ SKIP ON CRYSTAL CLOCK

1293

/ TIMING FLAG

1294

6332 SKAD=6332

/ SKIP ON ADC DONE

1295

6322 SKEP=6322

/ SKIP ON ADC TIMING ERROR

1296

6342 ZTEN=6342

/ READ IN ENABLE REGISTER

1297

6344 OTEN=6344

1298

6351 CLEP=6351

/ CLEAR ADC TIMING ERROR FLAG

1299

6352 CLXK=6352

/ CLEAR CRYSTAL CLOCK TIMING FLAG

1100

6354 CLPK=6354

/ CLEAR PC CLOCK TIMING FLAG

1101

6371 ACMX=6371

/ SET MULTIFLEX REGISTER FROM AC

1102

6361 ICMX=6361

/ INCREMENT MULTIFLEX CHANNEL

1103

6362 RADC=6362

/ ADC BUFFER TO AC

1104

6364 ADCV=6364

/ INITIATE CONVERSION

1105

6334 XPCL=6334

/ CLEAR ALL BITS OF EXTERNAL

1106

/ SENSE REGISTER

1107

6331 XRIN=6331

/ READ IN EXTERNAL SENSE REGISTER

1103

1109

1110

1111

1112

1113

1114

1115

1116

1117

1118

1119

1120

1121

1122

1123

1124

1125

1126

1127

1128

1129

1130

1131

1132

1133

1134

1135

1136

1137

1138

1139

1140

1141

1142

1143

/ DEFINITIONS FOR KV GRAPHIC DISPLAY SYSTEM.

6051 SNC=6051

/ SKIP THE NEXT PROGRAM

/ INSTRUCTION IF

/ THE CURSOR INTERRUPT

/ FLAG HAS NOT BEEN SET

6052 CCF=6052

/ UNCONDITIONALLY CLEAR THE CURSOR

/ FLAG

6062 SAC=6062

/ SELECT THE ANALOG COMPARATOR

6063 LDF=6063

/ LOAD FORMAT

6064 LEX=6064

/ LOAD THE X-AXIS SAMPLE AND HOLD

/ THE REGISTER

6065 LDY=6065

/ LOAD THE Y-AXIS SAMPLE AND HOLD

/ THE REGISTER

6066 EXC=6066

/ EXECUTE THE INSTRUCTION

/ ACCORDING TO

/ THE PARAMETER WORD

/ CURRENTLY HELD IN THE

/ ACCUMULATOR

6071 SPF=6071

/ SKIP IF THE READY FLAG IS SET

6072 CFF=6072

/ UNCONDITIONALLY CLEAR THE READY

/ FLAG

6073 SLA=6073

/ SKIP WHEN THE OUTPUT OF THE

/ DIGITAL-TO-ANALOG

/ CONVERTER OUTPUT IS

/ GREATER THAN THE

/ SELECTED SOURCE

6074 LDA=6074

/ LOAD THE DIGITAL-TO-ANALOG

/ CONVERTER

1144

1145

1146

1147

1148

1149

/ DEFINITIONS FOR THE INTERFACE

1150

/ KL3-JA TERM CONT/ASYNC DATA INT.

1151

1152

1153

1154

/ JUMPER DEFINITIONS:

1155

/ B1, B2, B3 JUMPERS CONTROLLE THE BAUD RATE.

1156

1157

1158

1159

/ INPUT:

1160

/ RECEIVE IOT (JUMPERS SP1): 44

1161

1162

1163

6440 VICF=6440

/ CLEAR KEYBOARD FLAG (KCF)

1164

6441 VISF=6441

/ SKIP IF KEYBOARD FLAG IS

1165

/ SET (KSF)

1166

6442 VICC=6442

/ CLEAR KEYBOARD FLAG AND

1167

/ SET PEADER RUN (KCC)

1168

6444 VIRS=6444

/ READ KEYBOARD STATIC (KRS)

1169

6445 VIIE=6445

/ SET/CLEAR INTERRUPT ENABLE (KIE)

1170

/ AC11

1171

6445 VISE=6445

/ SET/CLEAR STATUS ENABLE (KSE)

1172

/ AC12

1173

6446 VIRB=6446

/ READ KEYBOARD BUFFER

1174

/ DYNAMIK (KRB)

1175

1176

1177

/ OUTPUT:

1178

/ TRANSMIT IOT (JUMPERS SP2): 45

1179

1180

1181

6450 VOFL=6450

/ SET TELEPRINTER FLAG (TFL)

1182

6451 VOSF=6451

/ SKIP ON TELEPRINTER FLAG (TSF)

1183

6452 VOCF=6452

/ CLEAR TELEPRINTER FLAG (TCF)

1184

6454 VOFC=6454

/ LOAD TELEPRINTER & PRINT (TFC)

1185

6455 VOPI=6455

/ SKIP IF TELETYPE INTERRUPT (SPI)

1186

6456 VOLS=6456

/ PRINT CHARACTER (TLS)

1137

1133

1139

1190

/ DEFINITION FOR POWER FAILURE OPTION.

1191

1192

1193

1194

6102 SFL=6102

/ SKIP ON POWER LOW

1195

1196

1197

1198

1199

1220

/ DEFINITIONS FOR INSTRUCTIONS.

1221

/ BUT ONLY FOR PDP-3/E !!

1222

1223

1224

1225

6007 CAF=6007

/ CLEAR ALL FLAGS

1226

7002 BSW=7002

/ SWAP BYTES IN AC

1227

1223

1229

1210

1211

1212

/ NUMERICAL LOAD DEFINITIONS.

1213

1214

1215

7300 LW0=CLA CLL

/ + 0

1216

7301 LW1=CLA CLL IAC

/ + 1

1217

7326 LW2=CLA STL RTL

/ + 2

1213

7325 LW3=CLA STL IAC PAL

/ + 3

1219

7307 LW4=CLL CLA IAC RTL

/ + 4

1220

7327 LW6=CLA STL IAC RTL

/ + 6

1221

7332 LW2000=CLA STL RTR

/ +1024 (DEC) = 2000 (OCT)

1222

7350 LW3777=CLL STA PAR

/ +2047 (DEC) = 3777 (OCT)

1223

7330 LW4000=CLA STL PAR

/ - 0 (DEC)

1224

7352 LW5777=CLL STA RTR

/ -1025 (DEC) = 5777 (OCT)

1225

7333 LW6000=CLA STL IAC RTR

/ -1024 (DEC) = 6000 (OCT)

1226

7240 LWM1=STA

/ - 1 (DEC) = 7777 (OCT)

1227

7344 LWM2=CLL STA PAL

/ - 2 (DEC) = 7776 (OCT)

1223

7346 LWM3=CLL STA RTL

/ - 3 (DEC) = 7775 (OCT)

1229

1230

1231

1232

1233

/ FOCAL SYMBOLS:

1234

1235

0033

T3=33

1236

1237

0041

AC1H=41

1238

0042

AC1L=42

1239

0043

OVER1=43

1240

1241

0045

HORD=45

1242

0046

LOPD=46

1243

0047

OVER2=47

1244

1245

0025

F13=5

1246

0106

F177=106

1247

0113

C260=113

1248

0124

F4200=124

1249

1250

6000

FLOUTP=6000

1251

6136

SMIN=6136

1252

6154

OUTEG=6154

1253

7004

DMULT=7004

1254

7124

SIGN=7124

1255

1256

0066

CHAP=66

1257

0044

FLAC=44

1258

4551

PRINTC=JMS I 151

1259

4540

PUSHJ=JMS I 140

1260

4564

TESTC=JMS I 164

1261

2017

EFUNC=2017

1262

1613

EVAL=1613

1263

4453

INTEGER=JMS I 53

1264

1265

0136

EFUNCI=136

1266

6200

FLINTP=6200

1267

0756

INPUTT=0756

1268

0054

SOFTCN=54

1269

5666

XINPUT=5666

1270

6306

XIN=6306

1271

1272

2725

ERROR5=2725

1273

1274

1275

1276

0001 FOUFPD=1

1277

1278

1279

1280

1281

1282

1283

1284

1285

1286

1287

1288

1289

1290

1291

1292

1293

1294

1295 00212 0340

1296 00213 1113

1297 00214 2455

1298 00215 2425

1299 00216 0240

1300 00217 0617

1301 00220 0301

1302 00221 1447

1303 00222 7061

1304 00223 0000

1305

1306 0223 *-1

1307 00223 7715

/ NEW INITIAL DIALOGUE

0000 FIELD 0

0012 *212

TEXT "C IKT-TUB FOCAL '31"

7715

1308

1309

1310

1311

/ MODIFY PS/3-FOCAL INITIALIZATION

1312

1313

1314

1315

0001 FIELD 1

1316

1317

1318

4111 *4111

1319

1320

14111 6223

CDF CIF 20

1321

14112 5713

JMP I .+1

1322

14113 4000

EXTEND

1323

1324

1325

1326

1327

0002 FIELD 2

1328

1329

1330

4000 *4000

1331

1332

1333

24000 1221

EXTEND.

TAD EXT0

1334

24001 3010

DCA AUTO

1335

24002 1212

TAD M21

1336

24003 3104

DCA COUNT

1337

24004 4213

JMS EXTEN

1338

24005 2104

ISZ COUNT

1339

24006 5204

JMP .-2

1340

24007 6213

CDF CIF 10

1341

24010 5611

JMP I .+1

1342

24011 4363

PSCONT

1343

1344

24012 7757

M21,

-21

1345					
1346					
1347			/	STORE THE NORMAL PRECISION COMMANDS	
1348			/	(THREE WORD PRECISION) AT THEIR PLACES	
1349			/		
1350			/	FOURWD=0	
1351					
1352					
1353					
1354					
1355	24013	7402	EXTEN.	HLT	
1356	24014	1410		TAD I AUTO	/GET ADDRESS
1357	24015	3103		DCA LOW	
1358	24016	1410		TAD I AUTO	/GET COMMAND
1359	24017	3503		DCA I LOW	
1360	24020	5610		JMP I EXTEN	/RETURN
1361					
1362					
1363					
1364					
1365	24021	4021	EXTC,	EXTC	
1366					
1367	24022	4063	APGMA		/MODIFY INITIALIZATION
1368				DECIMAL	
1369	24023	5253		-1365	
1370				OCTAL	
1371					
1372					
1373	24024	4771	CF		/MODIFY EAE OVERLAY
1374	24025	1124		TAD P4000	
1375	24026	4772	CF+1		
1376	24027	5257		5257	// JMP BE-1
1377	24030	5042	EX1		
1378	24031	7425		MCL MUY	
1379	24032	5101	EX2		
1380	24033	7421		MCL	
1381					
1382					
1383	24034	0232	EX3		/MODIFY ROUTINE FN
1384	24035	7104		CLL PAL	
1385	24036	0233	EX3+1		
1386	24037	1117		TAD ARG2	
1387	24040	0274	EX4		
1388	24041	1117		TAD ARG2	
1389	24042	0300	EX5		
1390	24043	7000		NOP	
1391	24044	0343	EX6		
1392	24045	7300		CLA CLL	
1393	24046	0453	EX7		
1394	24047	7300		CLA CLL	

```
1395
1396
1397 24050 1061 EX10 /MODIFY ROUTINE FX
1398 24051 7124 CLL PAL
1399 24052 1062 EX10+1
1400 24053 1354 1354 // TAD MATLOC
1401 24054 1105 EX11
1402 24055 7410 SKP
1403 24056 1106 EX11+1
1404 24057 7000 NOP
1405 24060 1133 EX12
1406 24061 5340 5340 // JMP 21140
1407 24062 1173 M34
1408
1409 24063 7654 DECIMAL
1410 -34
1411 OCTAL
1412
1413
1414 4064 CONTIG=.
```



```
1415
1416
1417
1418      0001  FIELD 1
1419
1420
1421      4362  *4362
1422
1423      4207  CHANGE=4207
1424      4222  FORADD=4222
1425      14362 7747  M31,   -31
1426
1427      14363 4207  PSCONT, JMS CHANGE
1428      14364 4221                      FORADD-1
1429      14365 1362                      TAL M31
1430      14366 5767                      JMP I .+1
1431      14367 4114                      4114
1432
1433
1434      4370  CONTIN=.
1435
1436
1437
1438      4147  *4147
1439      14147 4370  CONTIN
1440
1441
1442
1443      4370  *CONTIN
1444
1445      14370 6223  CDF CIF 20
1446      14371 5772  JMP I .+1
1447      14372 4200  START0
```

1443

1449

1450 0202 FIELD 2

1451

1452

1453

1454 4064 *CONTI0

1455

1456

1457

1458

1459 / TYPE A MESSAGE ONTO THE TELETYPE.

1460

1461

1462

1463 24064 7402 MESSAGE, HLT

1464 24065 1664 TAD I MESSAGE

1465 24066 2264 ISZ MESSAGE

1466 24067 3104 DCA COUNT / SET POINTER

1467 24072 1504 MESAGI, TAD I COUNT

1468 24071 0340 AND 07700

1469 24072 7112 CLL RTR

1470 24073 7012 RTR

1471 24074 7012 RTR

1472 24075 4305 JMS MESLAM / PRINT FULL ASCII-CODE

1473 24076 1504 TAD I COUNT

1474 24077 0045 AND 077

1475 24100 7450 SNA / END ?

1476 24101 5664 JMP I MESSAGE / YES: RETURN

1477 24102 4305 JMS MESLAM / NO: PRINT

1478 24103 2104 ISZ COUNT

1479 24104 5270 JMP MESAGI / CONTINUE

1480

1481

1482

1483

1484 / THIS ROUTINE MAKES A FULL ASCII-CODE AND

1485 / PRINT IT; LF, CR, RUBOUT IS NOT POSSIBLE.

1486

1487

1488

1489 24105 7402 MESLAM, HLT

1490 24106 7450 SNA / WAS IT 00 ?

1491 24107 5705 JMP I MESLAM / YES: IGNORE

1492 24110 1070 TAD M40

1493 24111 7510 SPA / IS IT 1...39 ?

1494 24112 1334 TAD C100 / YES: IT MUST BE 321-339

1495 24113 1337 TAD CC240

1496 24114 4326 JMS PRINT

1497 24115 5705 JMP I MESLAM / RETURN

1493
1499
1500
1501 / ROUTINE TO PRINT CR AND LF.
1502
1503
1504
1505 24116 7402 CPLF, HLT
1506 24117 1336 TAD C215
1507 24120 4326 JMS PRINT
1508 24121 1335 TAD CC212
1509 24122 4326 JMS PRINT
1510 24123 1335 TAD CC212
1511 24124 4326 JMS PRINT
1512 24125 5716 JMP I CPLF / RETURN
1513
1514
1515
1516
1517
1518 / PRINT A CHARACTER.
1519 / INTERRUPT IS OFF !
1520
1521
1522 24126 7402 PRINT, HLT
1523 24127 6046 TFS / PRINT ONTO TTY.
1524 24130 6041 TSF
1525 24131 5330 JMP --1
1526 24132 7300 CLA CLL
1527 24133 5726 JMP I PRINT
1528
1529
1530
1531
1532
1533 / CONSTANTS.
1534
1535
1536
1537 24134 0100 C100, 100
1538 24135 0212 CC212, 212
1539 24136 0215 C215, 215
1540 24137 0240 CC240, 240
1541 24140 7700 C7700, 7700
1542
1543
1544 4200 PAGE

```

1545
1546
1547      / START LFOCAL: IT WILL MODIFY ITSELF.
1548
1549      / CHECK COMPUTER TYPE.
1550
1551      24200  7300  START0, CLA CLL
1552      24201  1045      TAD C77
1553      24202  7002      BSW
1554      24203  7700      SMA CLA      / WAS THERE A SWAP?
1555      24204  1350      TAD C400      / NO: THIS IS A FDP-3/1.
1556      24205  1351      TAD C500
1557      24206  3753      DCA I MES113
1558
1559
1560      / TEST THE MEMORY SIZE.
1561
1562      24207  7326      LMS
1563      24210  3104      DCA COUNT
1564      24211  3102      DCA HIGH
1565      24212  1366      TAD M5
1566      24213  3103      DCA LOW
1567      24214  1223      TAD CCDF
1568      24215  1365  START1, TAD C12
1569      24216  3220      DCA START2
1570      24217  7327      LMS
1571      24220  7402  START2, HLT
1572      24221  3764      DCA I STARTE
1573      24222  1764      TAD I STARTE
1574      24223  6221  CCDF,   CDF 20
1575      24224  7750      SPA SNA CLA      / YOU GET THE UPPER SIZE?
1576      24225  5232      JMP START3      / YES
1577      24226  2102      ISZ HIGH      / NO
1578      24227  2104      ISZ COUNT
1579      24230  1220      TAD START2
1580      24231  5215      JMP START1      / TRY NEXT FIELD
1581      24232  1102  START3, TAD HIGH
1582      24233  7550      SPA SNA      / WHICH MAX. FIELD
1583      24234  5247      JMP START5      / FIELD 2
1584      24235  7041      CIA
1585      24236  3102      DCA HIGH
1586      24237  2103  START4, ISZ LOW
1587      24240  7410      SKP
1588      24241  5245      JMP STAP6
1589      24242  1363      TAD ARGMA      / CALCULATE ARGMAX
1590      24243  2102      ISZ HIGH
1591      24244  5237      JMP START4
1592      24245  3121  STAP6,   DCA ARGMAX
1593      24246  5252      JMP START6
1594      24247  7300  START5, CLA CLL
1595      24250  1371      TAD IFUNC
1596      24251  3772      DCA I NFELD3      / NO FUNCTION FN(2,...)

```


1597					
1598	24252	1124	STAPT6,	TAD COUNT	
1599	24253	1373		TAD COPEX	
1600	24254	3124		DCA COUNT	/ CORE SIZE.
1601	24255	1524		TAD I COUNT	
1602	24256	3754		DCA I MES2I2	
1603					
1604			/ AX03		
1605					
1606	24257	6352		CLXX	
1607					
1608			/ AF04		
1609					
1610	24260	1347		TAD C125	
1611	24261	6542		VSEL	/ SCAN RESET
1612	24262	7300		CLA CLL	
1613	24263	6541		VCHV	/ SELECT CHANNEL 0; DO IT
1614	24264	6562		VFD	
1615	24265	7302		CLA CLL	
1616					
1617			/ KL3-JA		
1618					
1619	24266	6452		VOCF	
1620	24267	6456		VOLS	
1621					
1622			/ KV3I		
1623					
1624	24270	6072		CRF	
1625	24271	6052		CCF	
1626	24272	7327		LW6	
1627	24273	6066		EXC	
1628	24274	7300		CLA CLL	
1629					
1630			/ DK3-EF		
1631					
1632	24275	6134		CLEN	
1633	24276	6130		CLZE	/ CLEAR ENABLE REGISTER
1634	24277	6135		CLSA	
1635	24300	7240		LWM1	
1636	24301	6132		CLAB	
1637	24302	7300		CLA CLL	
1638	24303	1367		TAD ENABL	/ OVERFLOW+MODE01+RATE
1639	24304	6132		CLDE	/ SET ENABLE REGISTER
1640	24305	7300		CLA CLL	
1641					
1642	24306	4755		JMS I ICPLF	
1643	24307	4756		JMS I IMESA	/ TYPE: PDP-3/..
1644	24310	4505		MES1	
1645	24311	4756		JMS I IMESA	/ TYPE: ...K
1646	24312	4511		MES2	

1647

1648

1649

/ CHECK EXTENDED ARITHMETIC ELEMENT.

1650

1651 24313 7350

LW3777

1652 24314 7413

SKL

1653 24315 0000

0000

1654 24316 7710

SPA CLA

/ EAE AVAILABLE?

1655 24317 4770

JMS I EAE

/ YES. LOAD EAE OVERLAY

1656

1657

1658

/ CHECK AX03 A/D-CONVERTER.

1659

1660 24320 6321

SKXX

/ CRYSTAL CLOCK FLAG?

1661 24321 5335

JMP DELLAB

/ NO AX03: DELETE FOCAL-

1662 24322 4756

JMS I IMESA

/ FUNCTION FLAG, FADC(K)

1663 24323 4522

MES4

/ TYPE: AX03

1664 24324 6352

START7, CLXX

1665

1666

1667

/ CHECK AF04 A/D-CONVERTER.

1668

1669 24325 6561

VSEF

/ DATA READY FLAG?

1670 24326 5344

JMP DELAF04

/ NO AF04: DELETE FOCAL-

1671 24327 4756

JMS I IMESA

/ FUNCTION FADC(K, F, P, S)

1672 24328 4527

MESS

/ TYPE: AF04

1673 24331 6562

START3, VRE

1674 24332 7300

CLA CLL

1675 24333 5734

JMP I .+1

1676 24334 4403

CHECKL

1677

1678

1679

1680

1681

1682

/ SUBROUTINE TO DELETE THE FOCAL-FUNCTION FLAG

1683

/ AND FADC(K).

1684

1685

1686

1687 24335 1361

DELLAB, TAD C2725

1688 24336 6211

CDF 10

1689 24337 3762

ECA I C414

1690 24340 6221

CDF 20

1691 24341 1352

TAD JMPERROR

1692 24342 3760

ECA I INOAX03

1693 24343 5324

JMP START7

1694

1695

1696

1697

1698

1699

1700

1701

```

1702      24344  1352  DELAF0, TAD JMPERROR
1703      24345  3757          DCA I INOAF04
1704      24346  5331          JMP START3

```

1705

1706

1707

1708

1709

1710

1711

/ CONSTANTS, VARIABLES, POINTERS

```

1712      24347  2105  C125,   125
1713      24352  2400  C400,   400
1714      24351  0500  C500,   500
1715      24352  5512  JMPERR, JMP I IERROR
1716      24353  4512  MES1I3, MES1+3
1717      24354  4513  MES2I2, MES2+2
1718      24355  4116  ICPLF,  CPLF
1719      24356  4064  IMESA,  MESSAGE
1720      24357  2635  INOAF2, NOAF24
1721      24360  2636  INOAX2, NOAX28
1722      24361  2725  C2725,  2725
1723      24362  0414  C414,   414

```

1724

1725

1726

1727

1728

1729

1730

1731

1732

1733

1734

1735

1736

1737

1738

1739

1740

1741

1742

1743

1744

```

          IFZERO FOURWD <
1726      ARGMA,  -1365          /DATA PER FIELD>
1727          IFNZRO FOURWD <
1728      ARGMA,  -1024>
1729          OCTAL
1730      24363  6000  STARTE,  0
1731      24365  0010  C10,    10
1732      24366  7773  MS,     -5
1733      24367  5300  ENABL,  5300
1734      24370  4600  EABL,   EAE
1735      24371  1305  IFUNC,  NOFUNC
1736      24372  2367  NFELD3, FELD3N
1737      24373  4373  COREX,  COREX
1738      24374  4070          4070      /  3
1739      24375  6162          6162      / 12
1740      24376  6166          6166      / 16
1741      24377  6260          6260      / 20
1742      24400  6264          6264      / 24
1743      24401  6270          6270      / 28
1744      24402  6362          6362      / 32

```

```

1745
1746
1747
1748
1749      / CHECK KL3-JA INTERFACE.
1750
1751      24403      6451      CHECKL,      VOSF      /FLAG SKIP?
1752      24404      5244      JMP DELFIO      /NO KL3-JA: DELETE FOCAL-
1753      24405      4672      JMS I IMESSA      /FUNCTION FIO
1754      24406      4534      MES6      /TYPE: KL3-JA
1755      24407      6452      START9,      VOCF
1756
1757
1758      / CHECK KUB1-DISPLAY.
1759
1760      24412      6071      SRF      /READY FLAG SKIP?
1761      24411      5251      JMP DELRV      /NO KUB1: DELETE FOCAL-
1762      24412      4570      JMS I IMESSA      /FUNCTION FIO
1763      24413      4542      MES7      /TYPE: KUB1
1764      24414      6072      STAP12,      CRF
1765      24415      6052      CCF
1766
1767
1768      / CHECK DK3-EP PROGRAMMABLE REAL TIME CLOCK
1769
1770      24416      6131      CLSK      /CLOCK SKIP ?
1771      24417      5262      JMP DELDK3      /NO DK3-EP: DELETE FOCAL-
1772      24420      4672      JMS I IMESSA      /FUNCTION FN(3,C), FN(4)
1773      24421      4547      MES3      /TYPE: DK3-EP
1774      24422      6134      STAR11,      CLEN
1775      24423      6130      CLZE      /CLEAR ENABLE REGISTER
1776      24424      6135      CLSA      /CLEAR CLOCK FLAGS
1777
1778
1779      / CHECK FOR ACTIVE FILE COMMANDS.
1780
1781      24425      7320      CLA CLL
1782      24426      6211      CDF 10
1783      24427      1703      TAD I A1201
1784      24430      7041      CIA
1785      24431      1271      TAD CC2725
1786      24432      7640      SZA CLA      /FILE COMMANDS ACTIVE?
1787      24433      5236      JMP FILECO      /YES
1788      24434      1271      TAD CC2725      /NO: DELETE FOCAL-
1789      24435      3704      DCA I A411      /FUNCTION FK
1790      24436      6221      FILECO,      CDF 20

```


1791
1792
1793
1794 24437 4677 JMS I IICPLF
1795 24440 4677 JMS I IICPLF
1796 24441 6213 CDF CIF 10
1797 24442 5643 JMP I .+1 /NORMAL PS/3-FOCAL START
1793 24443 2214 2214
1799

1322

1321

1322

1323

1324

1325

1326

1327

1323 24444 1271 DELFIO, TAD CC2725

1329 24445 6211 CDF 10

1312 24446 3667 DCA I C413

1311 24447 6221 CDF 27

1312 24450 5227 JMP START9

1313

1314

1315

1316

1317

1313

1319

1320

1321 24451 1271 DELKV, TAD CC2725

1322 24452 6211 CDF 10

1323 24453 3670 DCA I C412

1324 24454 6221 CDF 20

1325 24455 1675 TAD I IELKV2

1326 24456 3674 DCA I IELKV1

1327 24457 1273 TAD CCNOP

1323 24460 3676 DCA I IELKV3

1329 24461 5214 JMP STAR10

1332

1331

1332

1333

1334

1335

1336

1337

1338

1339

1340

1341

1342

1343

1344

1345

1346

1347

1348

1349

1350

1351

1352

1353

1354

1355

1356

1357

1358

1359

1360

1361

1362

1363

1364

/ SUBROUTINE TO DELETE THE FOCAL-
/ FUNCTIONS FM(3,C) AND FM(4)

24462	1300	DELDK3,	TAD DELDK
24463	3701		DCA I DELDK1
24464	1300		TAD DELDK
24465	3702		DCA I DELDK2
24466	5222		JMP STAP11

/ CONSTANTS, POINTERS.

24467	2413	C413,	413
24470	2412	C412,	412
24471	2725	CC2725,	2725
24472	4064	IMESSA,	MESSAGE
24473	7002	CCNOP,	NOP
24474	1406	IELKV1,	DELDK1
24475	1412	IELKV2,	DELDK2
24476	1447	IELKV3,	DELDK3
24477	4116	IICRLF,	CRLF
24500	1305	DELDK,	NOFUNC
24501	2370	DELDK1,	FELDCN+1
24502	2371	DELDK2,	FELDCN+2
24503	1201	A1201,	1201
24504	2411	A411,	411

1365

1366

1367

1368

1369

1370

/ MESSAGES.

1371 24505 2004 MES1, TEXT"PDP-3/ "

1372 24506 2055

1373 24507 7057

1374 24510 4000

1375

1376 24511 4040 MES2, TEXT" K "

1377 24512 4040

1378 24513 4040

1379 24514 4013

1380 24515 4000

1381

1382 24516 4040 MES3, TEXT" EAI"

1383 24517 4040

1384 24520 0501

1385 24521 0500

1386

1387 24522 4040 MES4, TEXT" AK03 "

1388 24523 4040

1389 24524 0130

1390 24525 6070

1391 24526 4000

1392

1393 24527 4040 MES5, TEXT" AF04 "

1394 24530 4040

1395 24531 0106

1396 24532 6064

1397 24533 4000

1398

1399 24534 4040 MES6, TEXT" KL3-JA "

1900 24535 4040

1901 24536 1314

1902 24537 7055

1903 24540 1201

1904 24541 4000

1905

1906 24542 4040 MES7, TEXT" KV31 "

1907 24543 4040

1908 24544 1326

1909 24545 7011

1910 24546 4000

1911

1912 24547 4040 MES8, TEXT" DK3-EP "

1913 24550 4040

1914 24551 0413

1915 24552 7055

1916 24553 0520

1917 24554 4000

1918

1919 4600 PAGE

```

1920
1921      / SUBROUTINE TO LOAD THE EAE OVERLAY
1922      / AND TO MOVE SOME ROUTINES INTO FREE
1923      / LOCATIONS.
1924
1925
1926
1927
1928      24600  7402  EAE  HLT
1929      24601  4273      JMS EAESTO
1930      24602  2441      2441
1931      24603  4716      EAE1
1932      24604  2464      2464
1933      24605  4273      JMS EAESTO
1934      24606  6123      6123
1935      24607  4741      EAE2
1936      24610  6131      6131
1937      24611  1252      TAD ANEMIN
1938      24612  3126      DCA NEWIN
1939      24613  1370      TAD EAE3
1940      24614  6211      CDF 12
1941      24615  3653      DCA I A6137
1942      24616  1255      TAD A413+1
1943      24617  3654      DCA I A413
1944      24620  1257      TAD A2616+1
1945      24621  3656      DCA I A2616
1946      24622  1261      TAD A2632+1
1947      24623  3660      DCA I A2632
1948      24624  1263      TAD A2647+1
1949      24625  3662      DCA I A2647
1950      24626  1265      TAD A2712+1
1951      24627  3664      DCA I A2712
1952      24630  1266      TAD EFOLOW3
1953      24631  3667      DCA I A31
1954      24632  1266      TAD EFOLOW3
1955      24633  3670      DCA I A134
1956      24634  1266      TAD EFOLOW3
1957      24635  3671      DCA I A175
1958      24636  6221      CDF 22
1959      24637  4273      JMS EAESTO
1960      24640  7011      EMULT+5
1961      24641  4770      EAE4
1962      24642  7100      MUL DIV-1
1963      24643  4273      JMS EAESTO
1964      24644  7177      7200-1
1965      24645  5057      EAE5
1966      24646  7252      7252
1967      24647  4672      JMS I IMESAC      / TYPE: EAE
1968      24650  4516      MES3
1969      24651  5600      JMP I EAE

```


1970

1971

1972

1973

1974

1975

/ POINTERS, CONSTANTS

1976	24652	2460	ANEWIN,	EFLINF
1977	24653	6137	A6137,	6137
1978	24654	0413	A413,	413
1979	24655	6123		EXF10
1980	24656	2616	A2616,	2616
1981	24657	7236		EOUT1
1982	24660	2630	A2630,	2630
1983	24661	2455		ERTRAP
1984	24662	2647	A2647,	2647
1985	24663	7224		ECLEARC
1986	24664	2712	A2712,	2712
1987	24665	7244		ECUTE
1988	24666	3343	EFOLLOW3,	FOLOGA+4
1989	24667	2231	A31,	31
1990	24670	2134	A134,	134
1991	24671	0175	A175,	175
1992	24672	4064	IMESA3,	MESAGE

1993

1994

1995

1996

1997

1998

1999

2000

2001

2002

2003

2004

2005

2006

2007

2008

2009

2010

2011

2012

2013

2014

2015

2016

2017

2018

2019

2020

2021

2022

2023

2024

2025

2026

24673 7402

24674 1673

24675 3011

24676 2273

24677 1673

24700 3012

24701 2273

24702 1011

24703 7041

24704 1673

24705 7041

24706 3124

24707 2273

24710 1412

24711 6211

24712 3411

24713 6221

24714 2104

24715 5310

24716 5673

/ SUBROUTINE TO STORE EAE COMMANDS

/

/ JMS EAESTO

/ START ADDRESS -1

/ COMMAND-LABEL

/ END ADDRESS

EAESTO, HLT

TAD I EAESTO

DCA ADDRESS

/GET START ADDRESS

ISZ EAESTO

TAD I EAESTO

DCA CONTENTS

/GET COMMAND

ISZ EAESTO

TAD ADDRESS

CIA

TAD I EAESTO

CIA

DCA COUNT

/SET COUNTER

ISZ EAESTO

EAESTI, TAD I CONTENTS

CDF 10

DCA I ADDRESS

CDF 20

ISZ COUNT

JMP EAESTI

JMP I EAESTO

/RETURN

2027			
2028			
2029			
2030	4716	EAE1=-1	
2031	24717	7402	7402
2032	24720	0106	0106
2033	24721	7427	7427
2034	24722	0012	0012
2035	24723	7521	7521
2036	24724	1113	1113
2037	24725	4551	4551
2038	24726	7521	7521
2039	24727	1113	1113
2040	24730	4551	4551
2041	24731	5642	5642
2042	24732	6223	CDF CIF 20 / "EFTPAF"
2043	24733	5657	5657
2044	24734	1445	TPAF
2045	24735	7402	HLT / "EFLINF"
2046	24736	6223	CDF CIF 20
2047	24737	4664	4664
2048	24740	5660	5660
2049	24741	0150	APPUT
2050			
2051			
2052			
2053	4741	EAE2=-1	
2054	24742	7700	7700
2055	24743	7144	7144
2056	24744	1336	1336
2057	24745	4551	4551
2058	24746	1033	1033
2059	24747	7510	7510
2060	24750	7041	7041
2061	24751	7427	7427
2062	24752	0144	0144
2063	24753	7521	7521
2064	24754	7440	7440
2065	24755	4354	4354
2066	24756	7521	7521
2067	24757	4737	4737
2068	24760	5600	5600
2069	24761	4453	INTEGER / "XFIO"
2070	24762	7421	MCL
2071	24763	4731	4731
2072	24764	6223	CDF CIF 20
2073	24765	5730	5730
2074	24766	2550	WAY
2075	24767	3245	GETARG

2276				
2277				
2278				
2279	24770	2442	EAE3,	2442
2280				
2281				
2282				
2283		4770	EAE4=.-1	
2284	24771	1047	CF,	1047
2285	24772	3216		3216
2286	24773	1043		1043
2287	24774	7425		7425
2288	24775	0000		0000
2289	24776	7421		7421
2290	24777	1046		1046
2291	25000	3224		3224
2292	25001	1043		1043
2293	25002	7525		7525
2294	25003	0000		0000
2295	25004	1124		1124
2296	25005	3301		3301
2297	25006	7004		7004
2298	25007	3324		3324
2299	25010	1047		1047
2100	25011	3235		3235
2101	25012	1042		1042
2102	25013	7525		7525
2103	25014	0000		0000
2104	25015	1301		1301
2105	25016	7421		7421
2106	25017	7430		7430
2107	25020	2324		2324
2108	25021	1045		1045
2109	25022	3246		3246
2110	25023	1043		1043
2111	25024	7525		7525
2112	25025	0000		0000
2113	25026	1324		1324
2114	25027	3324		3324
2115	25030	1047		1047
2116	25031	3255		3255
2117	25032	1041		1041
2118	25033	7525		7525
2119	25034	0000		0000
2120	25035	1324		1324
2121	25036	3324		3324
2122	25037	1046		1046
2123	25040	3264		3264
2124	25041	1042		1042

2125				
2126				
2127	25042	7525	EX1,	7525
2128	25043	0000		0000
2129	25044	1324		1324
2130	25045	4752		4752
2131	25046	1045		1045
2132	25047	3273		3273
2133	25050	1041		1041
2134	25051	7525		7525
2135	25052	0000		0000
2136	25053	3045		3045
2137	25054	7521		7521
2138	25055	3046		3046
2139	25256	4301		4301
2140	25057	5604		5604

2141			
2142			
2143			
2144		5057	EAE5=-.1
2145	25060	0000	0000
2146	25061	7421	7421
2147	25062	7004	7004
2148	25063	3047	3047
2149	25064	1045	1045
2150	25065	3210	3210
2151	25066	1042	1042
2152	25067	7525	7525
2153	25070	0000	0000
2154	25071	1047	1047
2155	25072	3047	3047
2156	25073	1046	1046
2157	25074	3217	3217
2158	25075	1041	1041
2159	25076	7525	7525
2160	25077	0000	0000
2161	25100	1047	1047
2162	25101	7521	EX2, 7521
2163	25102	3047	3047
2164	25103	5600	5600
2165	25104	6052	CCF / "ECLEAR"
2166	25105	6072	CRF
2167	25106	6022	PCF
2168	25107	6012	RRB
2169	25110	6562	VRD
2170	25111	7000	NOP
2171	25112	6442	VICC
2172	25113	6452	VOCF
2173	25114	5635	5635
2174	25115	3322	CLEARC
2175	25116	6223	CDF CIF 20 / "OUT1"
2176	25117	4652	4652
2177	25120	6044	TPC
2178	25121	3016	DCA 16
2179	25122	5643	5643
2180	25123	2617	2617
2181	25124	6223	CDF CIF 20 / "OUT2"
2182	25125	4652	4652
2183	25126	6046	TLS
2184	25127	3016	DCA 16
2185	25130	5651	5651
2186	25131	2713	2713
2187	25132	1400	OUTPUT

2133

2139

2190

NOPUNCH

2191

2192

2193

2194

/

FOCAL PATCHES FOR 3/I EAE

2195

2196

/

ROUTINE BY JAMES VAN ZEE

2197

/

IMPROVED EAE ROUTINE FOR FOCAL

2198

/

DECUS PROGRAM LIBRARY FOCAL3-233

2199

2200

2201

2202

2203

/

THESE ROUTINES UTILIZE THE

2204

/

STANDARD KES-1 (OR KES2) E.A.E.

2205

/

HARDWARE TO IMPROVE FOCAL'S

2206

/

MULTIPLY SPEED BY A FACTOR OF

2207

/

ABOUT FOURTEEN. THE CODING IS

2208

/

ENTIRELY DIFFERENT FROM THE

2209

/

PATCH SHOWN ON PAGE 39 OF THE

2210

/

LISTING WHICH ONLY GIVES AN

2211

/

IMPROVEMENT OF ABOUT FOUR.

2212

/

ADVANTAGE IS TAKEN OF TWO FEAT-

2213

/

URES OF THE HARDWARE WHICH

2214

/

ARE NOT WIDELY KNOWN: (1) THE

2215

/

COMBINATION MCA MCL INTERCHANGES

2216

/

THE AC 2 MC AND (2) THE

2217

/

MUY INSTRUCTION ADDS THE C(AC)

2218

/

TO THE RESULT. THE ACTUAL

2219

/

MULTIPLY TIME FOR A 36 BIT PRODUCT

2220

/

IS 232 MICROSECONDS,

2221

/

BUT SIGN-CHECKING AND NORMALIZATION

2222

/

APPROXIMATELY DOUBLE THIS.

2223

/

A TWO-BIT NORMALIZE REQUIRES

2224

/

55 MICROSECONDS MORE

2225

/

AND IF THE RESULT IS NEGATIVE A

2226

/

FURTHER 140 MICROSECONDS

2227

/

ARE NEEDED. THUS THE OVERALL

2228

/

IMPROVEMENT IN PROGRAM SPEED

2229

/

IS CONSIDERABLY LESS THAN FOURTEEN.

2230

2231

/

THE HARDWARE DIVIDE INSTRUCTION

2232

/

IS ALSO USED PRIMARILY TO

2233

/

SHORTEN TWO ROUTINES. THIRTY-NINE

2234

/

WORDS ARE MADE AVAILABLE

2235

/

BY THIS PATCH.

2236

2237

2238

/LETTER ASSIGNMENTS:

2239

2240

2241

0045

AA=HORD

2242

0046

BB=LORD

2243

0047

CC=OVER2

2244

2245

0041

D=AC1H

2246

0042

E=AC1L

2247

0043

F=OVER1

2248

2249

2250

/NUMBER ASSIGNMENTS:

2251

2252

2253

2254

0047

TWO=OVER2

2255

7124

THREE=SIGN

2256

7101

FOUR=MUL DIV

2257

2258

2259

2260

0001 FIELD 1

2261

2262

2263

2442 *2442

2264

2265

2266

/EAE1=.-1

2267

2268

2269

12442

7402

PRNT,

HLT

/PRINT TWO DECIMAL DIGITS

2270

12443

0106

AND P177

2271

12444

7427

MCL DVI

/DIVIDE BY TEN

2272

12445

0012

12

2273

12446

7521

MCL MGA

/GET QUOTIENT

2274

12447

1113

TAD C260

/FORM ASCII

2275

12450

4551

PRINTC

2276

12451

7521

MCL MGA

/GET REMAINDER

2277

12452

1113

TAD C260

2278

12453

4551

PRINTC

2279

12454

5642

JMP I PRNT

2280

2281

12455

0000

ZBLOCK 10

/EIGHT LEFT OVER

2282

2283

/ "ERTRAP", "EFLINP"-FUNCTION

2234

2235

2236

2237

/EXPONENT OUTPUT ROUTINE

2238

2239

2290

2291

2292

6104 *6104

2293

2294

/EAE2=-.1

2295

2296

2297 16104 7700

SMA CLA

/TEST SIGN OF EXPONENT

2298 16105 7144

CMA CLL PAL

/CREATE "+"

2299 16106 1336

TAD SMIN

/OR "-"

2300 16107 4551

FPINTC

2301 16108 1033

TAD TS

/GET DECIMAL EXPONENT

2302 16111 7512

SFA

2303 16112 7041

CIA

/TAKE ABSOLUTE VALUE

2304 16113 7427

MGL DVI

/DIVIDE BY ONE HUNDRED

2305 16114 0144

144

2306 16115 7521

MGL MCA

/PRINT QUOTIENT

2307 16116 7440

SZA

/UNLESS IT'S ZERO

2308 16117 4354

JMS OUTDG

2309 16120 7521

MGL MCA

/NOW PRINT REMAINDER

2310 16121 4737

JMS I FRNT1

2311 16122 5600

JMP I FLOUTP

2312

2313 16123 0000

ZBLOCK 10

/EIGHT FREE

2314

2315

/ "EXFIO"-FUNCTION

2316

2317

2318

2319

2320

2321

6137 *6137

2322

2323

/EAE3

2324

2325

2326 16137 2442 FRNT1, FRNT

```

2327
2328
2329 /      THREE WORD BY THREE WORD
2330 /      UNSIGNED MULTIPLY ROUTINE
2331
2332 /      (A+B+C)*(D+E+F) = (ONE+TWO+THREE+FOUR+FIVE+SIX)
2333
2334 /      THE METHOD GENERATES ALL
2335 /      6 WORDS BUT IS MODIFIED
2336 /      HERE TO PRESERVE ONLY THE
2337 /      3 MOST SIGNIFICANT ONES.
2338 /      ADDITIONALLY THE RESULT
2339 /      IS ROUNDED OFF RATHER THAN
2340 /      TRUNCATED WHICH IMPROVES
2341 /      THE ACCURACY.
2342
2343
2344 7212 *DMULT+6
2345
2346 /EAE4=-.1
2347
2348
2349 IFZERO FOURWD <
2350 17012 1047 TAD CC // CF,
2351 17013 3216 DCA .+3>
2352 IFZERO FOURWD <
2353 TAD P4000
2354 JMP BE-1>
2355 17014 1043 TAD F
2356 17015 7425 MQL MUY
2357 17016 0000 0
2358 17017 7421 MQL /SAVE HIGH ORDER AND EPASE
2359 /SIX
2360 17020 1046 BF, TAD BE
2361 17021 3224 DCA .+3
2362 17022 1043 TAD F
2363 17023 7525 MQL MGA MUY /USE PREVIOUS HIGH ORDER
2364 17024 0000 0 /ASPEMAINDER IN THIS
2365 /POSITION
2366 17025 1124 TAD P4000 /ROUND OFF
2367 17026 3301 DCA FOUR /SAVE INTERMEDIATE RESULT
2368 17027 7004 RAL
2369 17030 3324 DCA THREE /SAVE CARRY AND/OR CLEAR
2370 /THREE
2371 17031 1047 CE, TAD CC
2372 17032 3235 DCA .+3
2373 17033 1042 TAD E

```

2374					
2375					
2376	17034	7525		MQL MGA MUY	/ADD IN PREVIOUS
2377	17035	0000		0	/PARTIAL PRODUCT
2378	17036	1321		TAD FOUR	/SUM HIGH ORDER PARTS
2379	17037	7421		MQL	/DISCARD FIVE AND SAVE
2380	17040	7430		SZL	/CAPRY
2381	17041	2324		ISZ THREE	/ACCUMULATE CARRIES
2382	17042	1045	AF,	TAD AA	
2383	17043	3246		DCA .+3	
2384	17044	1043		TAD F	
2385	17045	7525		MQL MGA MUY	
2386	17046	0000		0	
2387	17047	1324		TAD THREE	/BUILD UP LEAST
2388					/SIGNIFICANT PART
2389	17050	3324		DCA THREE	/OF FINAL RESULT
2390					
2391	17051	1047	CL,	TAD CC	
2392	17052	3255		DCA .+3	
2393	17053	1041		TAD D	
2394	17054	7525		MQL MGA MUY	
2395	17055	0000		0	
2396	17056	1324		TAD THREE	
2397	17057	3324		DCA THREE	/ENTRY POINT FOR 2X2
2398	17060	1046	BE,	TAD BB	
2399	17061	3264		DCA .+3	
2400	17062	1042		TAD E	
2401				IFNZRO FOURWD <	
2402	17063	7525		MQL MGA MUY>	// EX1,
2403				IFZERO FOURWD <	
2404				MQL MUY>	
2405	17064	0000		0	
2406	17065	1324		TAD THREE	
2407	17066	4752		JMS I 7152	/CONTINUE IN MP4 SPACE
2408					
2409	17067	1045	AD,	TAD AA	/FINAL PARTIAL PRODUCT
2410	17070	3273		DCA .+3	
2411	17071	1041		TAD D	
2412	17072	7525		MQL MGA MUY	
2413	17073	0000		0	
2414	17074	3045		DCA HORD	/SAVE ONE
2415	17075	7521		MQL MGA	
2416	17076	3046		DCA LORD	/SAVE TWO
2417	17077	4301		JMS MULDIV	/COMPUTE SIGN & NORMALIZE
2418	17100	5604		JMP I DMULT	
2419		7101	MUL DIV=.		

```

2422
2421
2422      7200  *7200
2423
2424
2425
2426      /EAES=-1
2427
2428
2429      17200  0000  MP4,      0
2430      17201  7421      MGL      /DISCARD FOUR
2431      17202  7004      RAL
2432      17203  3047      DCA TWO  /BUT SAVE CARRY
2433
2434      17204  1245  AE,      TAD AA
2435      17205  3210      DCA .+3
2436      17206  1042      TAD E
2437      17207  7525      MQL MGA MUY
2438      17210  0000      0
2439      17211  1047      TAD TWO  /SUM CARRY BITS
2440      17212  3047      DCA TWO  /WITH CARRY WORD
2441
2442      17213  1046  ED,      TAD BB
2443      17214  3217      DCA .+3
2444      17215  1041      TAD D
2445      17216  7525      MQL MGA MUY
2446      17217  0000      0
2447      17220  1047      TAD TWO
2448      IFNZRO FOURWD <
2449      17221  7521      MQL MGA>      //      EX2,
2450      IFZERO FOURWD <
2451      MQL>
2452      17222  3047      DCA OVER2  /SAVE THREE
2453      17223  5600      JMP I MP4  /FINISH UP
2454
2455      17224  0000      ZBLOCK 27  /TWENTY-THREE FREE WORDS
2456
2457      / "ECLEAR", "EOUT1", "EOUT2" - FUNCTION

```


2458

2459

2460

2461

/ MORE COMMANDS INTO FREE LOCATIONS

2462

/

2463

/ FOLLOW=STVAR WILL CHANGE FROM FOLOWS TO FOLOWS

2464

2465

2466

2467

2468

2455 *2455

2469

2470

12455

6223

ERTRAP, CDF CIF 20

/ "RTRAP"-ROUTINE

2471

12456

5657

JMP I .+1

2472

12457

1445

TRAP

2473

2474

2475

2476

2477

12460

7400

EFLINP, HLT

/ "FLINP"-ROUTINE

2478

12461

6223

CDF CIF 20

2479

12462

4664

JMS I EAFPUT

2480

12463

5660

JMP I EFLINP

2481

12464

0150

EAFPUT, APUT

2482

2483

2484

2485

2486

6123 *6123

2487

2488

16123

4453

EXFIO, INTEGER

/ "XFIO"-ROUTINE

2489

16124

7421

MCL

2490

16125

4731

JMS I .+4

2491

16126

6223

CDF CIF 20

2492

16127

5730

JMP I .+1

2493

16130

2550

WAY

2494

16131

3245

GETARG

```
2495
2496
2497
2498
2499      7224  *7224
2500
2501      17224  6052  ECLEAR, CCF      / "CLEARQ"-ROUTINE
2502      17225  6072      CRF
2503      17226  6022      PCF
2504      17227  6012      RRB
2505      17230  6562      VRD
2506      17231  7000      NOP
2507      17232  6442      VICC
2508      17233  6452      VOCF
2509      17234  5635      JMP I .+1
2510      17235  3322      CLEARC
2511
2512
2513
2514
2515      17236  6223  EOUT1, CDF CIF 20      / "OUT1"-ROUTINE
2516      17237  4652      JMS I EOUT10
2517      17240  6044      TPC
2518      17241  3016      DCA 16
2519      17242  5643      JMP I .+1
2520      17243  2617      2617
2521
2522
2523
2524
2525      17244  6223  EOUT2, CDF CIF 20      / "OUT2"-ROUTINE
2526      17245  4652      JMS I EOUT10
2527      17246  6046      TLS
2528      17247  3016      DCA 16
2529      17250  5651      JMP I .+1
2530      17251  2713      2713
2531
2532      17252  1400  EOUT10, OUTFUT
2533
2534
2535      ENPUNCH
```

2536

2537

2538

2539

2540

2541

2542

2543

2544

2545

2546

2547

2548

2549

2550

2551

2552

2553

2554

2555

2556

2557

2558

2559

2560

2561

/ CORRECTIONS TO PS/3- OR OS/3-FOCAL

0001

FIELD 1

0001 *0001

10001

3407

STVAR / ADDRESS OF LAST VARIABLES

0134 *0134

10134

3407

STVAR

0175 *0175

10175

3407

STVAR / FOR SECRET VARIABLES

2562
2563
2564
2565
2566
2567
2568
2569

/ CORRECTIONS FOR A LONGER COMMAND-LIST:
/ FIVE NEW COMMANDS.

2570 2157 *157
2571 10157 3234 XRTL6 /CHANGE POINTER
2572
2573
2574
2575

2576 / LONGER FNTABF - LIST
2577
2578

2579 2410 *410
2580 10410 3241 XFN / FN
2581 10411 3267 XFX / FX
2582 10412 3337 XFD / FD
2583 10413 3374 XFIO / FIO
2584 10414 3263 XFLAB / FLAB
2585 10415 7522 7522 / FIN
2586 10416 7534 7534 / FOUT
2587 10417 0552 0552 / FIND
2588
2589
2590
2591
2592
2593
2594
2595

/ LONGER FNTABL - LIST

2596 2201 *2201
2597 12201 0316 0316 / FN
2598 12202 0330 0330 / FX
2599 12203 0304 0304 / FD
2600 12204 1141 1141 / FIO
2601 12205 2564 2564 / FLAB
2602 12206 1140 1140 / FIN
2603 12207 2672 2672 / FOUT
2604 12210 2604 2604 / FIND

2625

2626

2627

/ NEW START OF ERASE-COMMAND.

2628

2629

2610

2611

2612

2613

1174 *1174

2614

11174 3222

EPASENEW

/NEW POINTER

2615

2616

2617

2618

2619

3222 *3222

2620

2621

13222

4564

EPASEN, TESTC

/TEST SECOND WORD,

2622

/IF ANY

2623

13223

5631

JMP I IEPUK

/ERASE VARIABLES

2624

13224

5632

JMP I IEPL

/LINES OR GROUPS

2625

13225

5633

JMP I IEPR3

/ERROR

2626

13226

1266

TAD CHAR

2627

13227

5630

JMP I .+1

/CONTINUE

2628

13232

2211

2211

2629

2630

13231

2237

IEPUK, 2237

2631

13232

2222

IEPL, 2222

2632

13233

2213

IEPR3, 2213

2633

2634

2635

2636

2637

2638

/ NEW PLACE FOR "RTL6"

2639

2640

2641

13234

7402

XRTL6, HLT

2642

13235

7106

CLL RTL

2643

13236

7006

RTL

2644

13237

7006

RTL

2645

13240

5634

JMP I XRTL6

2646

2647

2648

2649

2650

2651

2652

2653

2654

2655

2656

2657

2658

2659

2660

2661

2662

2663

2664

2665

2666

2667

2668

2669

2670

2671

2672

2673

2674

2675

2676

2677

2678

2679

2680

2681

2682

2683

2684

2685

/ SUBROUTINES FOR NEW FUNCTIONS

13241	4453	XFN,	INTEGER	/GET 1. ARGUMENT
13242	6223		CDF CIF 20	
13243	5644		JMP I .+1	
13244	0351		FNEW	

13245	7402	GETARG,	HLT	
13246	4540		PUSHJ	/GET NEXT ARGUMENT
13247	1612		EVAL-1	
13250	4453		INTEGER	
13251	5645		JMP I GETARG	

13252	7402	GETAR,	HLT	
13253	4245		JMS GETARG	/GET NEXT ARGUMENT
13254	6223		CDF CIF 20	/AS AN INTEGER
13255	5652		JMP I GETAR	/RETURN

13256	7402	GETARI,	HLT	
13257	4540		PUSHJ	/GET NEXT ARGUMENT
13260	1612		EVAL-1	/IN FLOATING POINT
13261	6223		CDF CIF 20	
13262	5656		JMP I GETARI	

3263 FOLLOWI=.

```
2636
2637
2638
2639      0002  FIELD 2
2690
2691
2692      0004  *4
2693
2694
2695      / SUBROUTINE FOR RETURN TO FOCAL
2696
2697      20004  6001      ION
2698      20005  6213  FFOCAL, CDF CIF 10
2699      20006  5407      JMP I .+1      / RETURN TO FOCAL
2700      20007  2017      EFUNG
2701
2702
2703
2704
2705      / AUTO-INDEX-REGISTER
2706
2707      0010  *10
2708
2709      20010  0000  AUTO, 0
2710      20011  0000  ADDRESS, 0
2711      20012  0000  CONTEN, 0
2712
2713
2714      0020  *0020
2715
2716
2717      / SUBROUTINE TO SIMULATE A BCD-INPUT.
2718
2719      20020  7402  AFOUT, HLT
2720      20021  0057      AND C17
2721      20022  1066      TAD CC260      / BCD-CODE IN AC.
2722      20023  5420      JMP I AFOUT      / RETURN
2723
2724
2725
2726
2727      / SUBROUTINE TO GET AN ARGUMENT OF THE
2728      / FOCAL FUNCTION.
2729
2730      20024  7402  ARGUM, HLT
2731      20025  6213      CDF CIF 10
2732      20026  4430      JMS I .+2      / GET ARGUMENT.
2733      20027  5424      JMP I ARGUM      / RETURN.
2734      20030  3252      GETAR
```

2735					
2736					
2737			/CONSTANTS, VARIABLES, POINTERS		
2738					
2739	20031	7376	X0,	-402	/INITILIZE TO LEFT MARGIN
2740	20032	0506	Y0,	506	/INITILIZE TO TOP
2741					/OF SCREEN
2742	20033	0000	XLAST,	0000	/LAST COORDINATE STORED
2743					/HERE BY SCP
2744	20034	0000	YLAST,	0000	
2745	20035	0000	XTEMP,	0000	
2746	20036	0000	YTEMP,	0000	
2747	20037	7376	MARGIN,	-402	/LEFT MARGIN
2748	20040	0506	TOP,	506	/TOP OF PAGE
2749	20041	0000	IPEN,	0000	/PEN STORAGE LOCATION
2750	20042	0000	SCOPE,	0000	/SCOPE MODE SWITCH
2751	20043	0000	STOPE,	0	
2752	20044	0000	LINKL,	0000	
2753	20045	0077	C77,	0077	
2754	20046	1046	NUM0,	TAD	
2755	20047	4002	C4002,	4002	
2756	20050	0002		0002	
2757	20051	0000		0000	
2758	20052	0004		0004	
2759	20053	0003	C3,	3	
2760	20054	0005	C5,	5	
2761	20055	0007	C7,	0007	
2762	20056	0011	C11,	11	
2763	20057	0017	C17,	17	
2764	20060	0020	C20,	20	
2765	20061	7760	M20,	-22	
2766	20062	0060	C60,	60	
2767	20063	0000	C200,	200	
2768	20064	0005	C205,	205	
2769	20065	0140	C0140,	0140	
2770	20066	0060	CC260,	260	
2771	20067	0774	C774,	774	
2772	20070	7740	M40,	-40	
2773	20071	7524	M254,	-254	
2774	20072	0066	ICHAR,	CHAR	
2775	20073	0044	IFLAC,	FLAC	/ FLOATING POINT AC.
2776	20074	0045	IFLAC1,	FLAC+1	
2777	20075	0046	IFLAC2,	FLAC+2	
2778	20076	0047	IFLAC3,	FLAC+3	
2779	20077	2005	ISCALE,	SCALE	
2780	20100	0000	ISFLAG,	SFLAG	
2781	20101	2050	IDSPY1,	DSFY1	
2782	20102	0000	HIGH,	0	
2783	20103	0000	LOW,	0	
2784	20104	0000	COUNT,	0	


```

2735
2736      20105  0000  ARG,      0
2737      20106  7760  COUNT2, 7760
2738      20107  0000  TIMER,    0
2739      20110  0000
2790      0107  TEM1=TIMER
2791      0110  TEM2=TIMER+1
2792      0107  TESTE=TIMER
2793      20111  2571  WHICH0, WHICH
2794      20112  1305  IERROR, ERROR
2795      20113  3035  IPOIN,  POINTER
2796      20114  3033  POINT,  POINTER-2
2797      20115  1343  ILOOP,  ILOOP
2798      20116  1371  ITRANS, TRANSMIT
2799      20117  0000  ARG2,      0
2800      20120  0000  ARG3,      0
2801      20121  0000  ARGMAX,    0
2802      20122  0000  IFNUMB,  FNUMBER
2803      20123  3256  IGET1,  GETARI
2804      20124  2402  IDICON,  IDICONV
2805      20125  2400  IBCD,    BCD
2806      20126  3402  NEWIN,  FLINP
2807      20127  0756  OLDIN,  INPUTT
2808      20130  5666  AFLOP1,  KINPUT
2809      20131  6306  AFLOP2,  KIN
2810      20132  0135  AFINPU,  AFSIGN
2811      20133  1350  IFLINT,  FLINT
2812      20134  0054  ISORTC,  SORTON
2813
2814
2815
2816
2817      /  TABLE FOR SIMULATED FLOATING-POINT-INPUT
2818      /  IN BCD-CODE:
2819      /
2820      /      +XXXXXXE-Y
2821      /      -XXXXXXE-Y
2822
2823
2824      20135  0000  AFSIGN,    0      /  SIGN: + OR -
2825      20136  0000      0      /  1. CHAR.
2826      20137  0000      0      /  2. CHAR.
2827      20140  0000      0      /  3. CHAR.
2828      20141  0000      0      /  4. CHAR.
2829      20142  0000      0      /  5. CHAR.
2830      20143  0000      0      /  6. CHAR.
2831      20144  0305      305      /  E
2832      20145  0255  C255,    255      /  -
2833      20146  0000  AFEXP,    0      /  RANGE: 0...7
2834      20147  0215      215      /  TERMINATOR

```

2335

2336

2337

2338

2339

2340

2341

2342

2343

2344

2345

2346

2347

2348

2349

2350

2351

2352

2353

2354

2355

2356

2357

2358

2359

2360

2361

2362

2363

2364

2365

2366

2367

2368

2369

2370

2371

2372

2373

2374

2375

2376

2377

2378

2379

2380

2381

2382

2383

2384

/ SUBROUTINE TO GET THE SIMULATED FLOATING-
/ POINT-NUMBER HELD IN "AFSIGN"-FF

20150	7402	AFPUT,	HLT	
20151	1504		TAD I COUNT	/ GET CHARACTER
20152	2104		ISZ COUNT	
20153	6213		CDF C1F 10	
20154	3472		DCA I ICHAR	
20155	5550		JMP I AFPUT	/ RETURN

/ SUBROUTINE TO CHECK THE NEXT ARGUMENT
/ WITHIN A FOCAL-ROUTINE
/ CALL: JMS CHECK
/ EXIT1 /YES: NEXT ARG
/ EXIT2 /NO

20156	7402	CHECK,	HLT	
20157	6211		CDF 10	
20160	1472		TAD I ICHAR	
20161	6221		CDF 20	
20162	1071		TAD M254	
20163	7640		SZA CLA	/ WAS THERE A COMMA ?
20164	2156		ISZ CHECK	/ NO.
20165	5556		JMP I CHECK	/ RETURN.

/ TABLE FOR BINARY TO BCD CONVERSION

20166	1167	CONTRL,	TAD TABLE	
20167	6340	TABLE,	-1440	/ -300 (12)
20170	7160		-0620	/ -400
20171	7470		-0310	/ -200
20172	7634		-0144	/ -100
20173	7660		-0120	/ - 30
20174	7730		-0050	/ - 40
20175	7754		-0024	/ - 20
20176	7766		-0012	/ - 10

```

2335
2336      02202 *2002
2337
2338
2339
2390      / SUBROUTINE TO SERVE THE FAST FOURIER
2391      / TRANSFORMATION FFT
2392      /
2393      / FOCAL FUNCTION:
2394      /
2395      /      SET A=FN(2,U,NU)
2396
2397
2398
2399
2900
2901      / SUBROUTINE BY JAMES E. POTTMAN
2902      / DECUSCOPE 1963 VOL.7, NO.3, PP.3-12
2903      /
2904      / SEE ALSO:
2905      / FREQUENCY TRANSFORMATION PROGRAM
2906      / DECUS PROGRAM LIBRARY FOCAL3-226
2907
2908
2909
2910
2911
2912
2913      20200 4024 FFT, JMS ARGUM /GET 2. ARG.
2914      20201 3117 DCA ARG2
2915      20202 4024 JMS ARGUM /GET 3. ARG.
2916      20203 6211 CDF 10
2917      20204 1474 TAD I IFLAC1
2918      20205 7041 CIA
2919      20206 3104 DCA COUNT /NUMBER OF BITS
2920      20207 1117 TAD ARG2 /GET 2. ARG.
2921      20210 3474 DCA I IFLAC1 /PUT IT IN FLAC+1
2922      20211 3475 DCA I IFLAC2 /BUILD UP RESULT
2923      /IN FLAC+2
2924      20212 1474 FLOOP, TAD I IFLAC1 /TRANPOSE FLAC+1
2925      20213 7110 CLL RAR /ABOUT ITS CENTER BY
2926      20214 3474 DCA I IFLAC1 /ROTATING FLAC+1 RIGHT
2927      20215 1475 TAD I IFLAC2 /AND FLAC+2 LEFT
2928      20216 7004 FAL /BIT TO BE TRANPOSE
2929      20217 3475 DCA I IFLAC2 /IN LINK INSERT INTO
2930      /LOW ORDER BIT
2931      /OF FLAC+2
2932      20220 2104 ISZ COUNT /FOR ALL NU BITS
2933      20221 5212 JMP FLOOP /CONTINUE
2934      20222 3476 DCA I IFLAC+3
2935      20223 5005 JMP PFOCAL /RETURN TO FOCAL

```



```

2936
2937
2938 / ROUTINE TO GET/STORE DATA FROM/INTO
2939 / FIELD 2
2940 /
2941 / FOCAL FUNCTION:
2942 /
2943 / SET A=FN(1,1)
2944 / /DATA AT PLACE "1"
2945 / /IS HELD IN "A"
2946 / SET DUMMY=FN(1,1,A)
2947 / /DATA "A" IS STORED
2948 / /AT PLACE "1"
2949 /
2950 / I=0,1,...,593 /FOURWD=1
2951 /
2952 / I=0,1,...,799 /FOURWD=2
2953
2954
2955
2956 20224 4024 FELD2, JMS ARGUM /GET 2. APC.
2957 20225 3117 DCA APC2
2958 20226 1117 TAD ARG2
2959 20227 7021 IAC
2960 20230 7550 SPA SNA
2961 20231 5512 JMP I IERFOR /ERROR: <1 !
2962 /IFZERO FOURWD < /MULTIPLY BY
2963 EX3, CLL BAL /THREE
2964 TAD ARG2>
2965 IFNZRO FOURWD <
2966 20232 7106 EX3, CLL RTL /FOUR
2967 20233 7000 NOP>
2968 20234 1114 TAD POINT /SET START POINT
2969 20235 3010 DCA AUTO
2970 20236 1113 TAD IPOIN /AND CHECK IT
2971 20237 7141 CIA CLL
2972 20240 1010 TAD AUTO
2973 20241 7620 SNL CLA
2974 20242 5512 JMP I IERFOR
2975 20243 7100 CLL
2976 20244 1064 TAD C205
2977 20245 1010 TAD AUTO
2978 20246 7630 SZL CLA
2979 20247 5512 JMP I IERFOR
2980 20250 4156 JMS CHECK /NEXT ARGUMENT ?
2981 20251 5254 JMP .+3 /YES
2982 20252 4323 JMS FNGET /NO: GET DATA
2983 20253 6221 FELD20, CDF 20
2984 20254 4722 JMS I IFNSTO /YES: STORE DATA
2985 20255 6221 CDF 20

```



```

2936
2937
2938
2939 / ROUTINE TO GET/STORE DATA FROM/INTO
2990 / FIELD 3 FF.
2991 /
2992 / FOCAL FUNCTION:
2993 /
2994 / SET A=FN(2,1)
2995 / /DATA AT PLACE "I"
2996 / /IS HELD IN "A"
2997 / SET DUMMY=FN(2,1,A)
2998 / /DATA "A" IS STORED
2999 / /AT PLACE "I"
3000 /
3001 / /FOURWD=1
3002 / I=0,1,...,1023 /FIELD 3
3003 / =1024,...,2047 /FIELD 4
3004 / =2048,...,3071 /FIELD 5
3005 / =3072,...,4095 /FIELD 6
3006 /
3007 / /FOURWD=0
3008 / I=0,1,...,1364 /FIELD 3
3009 / I(MAX)=4095 /ETC.
3010
3011
3012
3013
3014 20256 4024 FEL D3, JMS ARGUM
3015 20257 3117 DCA ARG2
3016 20260 7300 CLA CLL
3017 20261 1117 TAD ARG2 /GET 2. ARG.
3018 20262 1121 TAD ARGMAX
3019 20263 7630 SZL CLA
3020 20264 5512 JMP I IERROR. /ERROR: TO HIGH!
3021 20265 7300 CLA CLL
3022 20266 3102 DCA HIGH
3023 20267 1117 TAD ARG2
3024 20270 7004 RAL /MULTIPLY 2. ARG.
3025 20271 7430 SZL
3026 20272 2102 ISZ HIGH
3027 20273 7100 CLL
3028 IFZERO FOURWD <
3029 EX4, TAD ARG2 > /BY THREE
3030 IFNZRO FOURWD <
3031 20274 7004 EX4, RAL> /BY FOUR
3032 20275 3103 DCA LOW
3033 20276 1102 TAD HIGH
3034 20277 7430 SZL

```

3035				IFZERO FOURWD <	
3036				NOP>	
3037				IFNZRO FOURWD <	
3038				PAL>	
3039			EXS,		
3040				RTL	
3041	20300	7004	EXS,		
3042	20301	7430		SZL	
3043	20302	7001		IAC	
3044	20303	7001		IAC	
3045	20304	7104		CLL RAL	
3046	20305	7006		RTL	
3047	20306	1253		TAD FELD20	
3048	20307	3315		DCA FELD30	/SET FIELD 3 FF
3049	20310	1103		TAD LOW	
3050	20311	3010		DCA AUTO	
3051	20312	4156		JMS CHECK	/NEXT ARGUMENT ?
3052	20313	5016		JMP .+3	/YES
3053	20314	4020		JMS FNGET	/NO: GET DATA
3054	20315	7402	FELD30,	HLT	
3055	20316	1015		TAD FELD30	
3056	20317	3321		DCA .+2	
3057	20320	4722		JMS I IFNSTO	/YES: STOPE DATA
3058	20321	7402		HLT	
3059					
3060					
3061	20322	0425	IFNSTO,	FNSTO	

```

3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072 20323 7402 FNGET, HLT
3073 20324 1723 TAD I FNGET /GET FIELD
3074 20325 3330 DCA FNGET1 /AND SET IT.
3075 20326 1723 TAD I FNGET
3076 20327 3340 DCA FNGET2
3077 20330 7402 FNGET1, HLT /CDF 20 ETC.
3078 20331 1410 TAD I AUTO
3079 20332 7421 MCL
3080 20333 1410 TAD I AUTO
3081 20334 6211 CDF 10
3082 20335 3474 DCA I IFLAC1
3083 20336 7501 MCA
3084 20337 3473 DCA I IFLAC
3085 20340 7402 FNGET2, HLT /CDF 22 ETC.
3086 20341 1410 TAD I AUTO
3087 20342 7421 MCL
3088
3089 IFZERO FOURWD <
3090 EX6, CLA CLL>
3091 IFNZRO FOURWD <
3092 EX6, TAD I AUTO>
3093 CDF 10
3094 DCA I IFLAC3
3095 MCA
3096 DCA I IFLAC2
3097 JMP PFOCAL /RETURN TO FOCAL

```

3097

3093

3099

/ NEW FUNCTIONS OF FOCAL

3100

3101

3102

3103 20351 7421 FNEW, MQL /SAVE 1. ARG.

3104 20352 7501 MGA /GET 1. ARG.

3105 20353 7510 SPA /POSITIVE?

3106 20354 5512 JMP I IERFOR /NO: EPROP

3107 20355 1364 TAD MUMB

3108 20356 7740 SMA SZA CLA /1. ARG. > 10 ?

3109 20357 5512 JMP I IERROR /YES: EPROP

3110 20360 7501 MGA

3111 20361 1122 TAD IFNUMBER

3112 20362 5763 JMP I .+1

3113 20363 2563 WAYOUT

3114

3115

3116

3117

/CONSTANTS, POINTERS

3113

3119 20364 7766 MUMB, -12

3120 20365 0200 FNUMBE, FFT /FOR FAST FOURIER TRANSFORMATION

3121 20366 0224 FELD2 /DATA INTO/FROM FIELD 2

3122 20367 0256 FELD3N, FELD3 /DATA INTO/FROM FIELD 3 FF

3123 20370 3011 CLOCKZ /START REAL TIME CLOCK

3124 20371 3023 CLOCKP /READ REAL TIME CLOCK FLAGS

3125 20372 1305 NEWFN, NOFUNC /POINTERS FOR NEW FUNCTIONS

3126 20373 1305 NOFUNC

3127 20374 1305 NOFUNC

3128 20375 1305 NOFUNC

3129 20376 1305 NOFUNC

3130 20377 1305 NOFUNC


```

3131
3132
3133      / BINARY TO BCD CONVERSION 2/6/65-HE-DEC
3134      / ENTER WITH BINARY NUMBER (<999<10)
3135      / IN ACCUMULATOR; EXIT WITH THREE CHARACTER
3136      / BCD NUMBER IN ACCUMULATOR
3137      / AC 0-3; AC 4-7; AC 8-11 WILL CONTAIN
3138      / THE BCD CHARACTER ON EXIT
3139      / WEIGHTING:          AC 0-3   100
3140      /                   AC 4-7   10
3141      /                   AC 8-11   1
3142      /
3143      / IF INPUT >999 (<10) RESULT IS UNSPECIFIED
3144
3145
3146      / EXAMPLE: INPUT  2726  (3)
3147      /             OUTPUT  0100/0111/0000  = 270 (10)
3148
3149
3150
3151

```

3152	20400	7402	BCD,	HLT	
3153	20401	3103		DCA LOW	/STORE BINARY
3154	20402	1166		TAD CONTEL	/SET UP TABLE
3155	20403	3210		DCA POINTP	/POINTERS
3156	20404	7100		CLL	
3157	20405	1060		TAD C00	/SET BIT 7=1; 3PAL'S
3158	20406	3104		DCA COUNT	/WILL PUT IT IN LINK
3159	20407	1103		TAD LOW	
3160	20410	1167	POINTP,	TAD TABLE	/OR TABLE+1, TABLE+2, ETC
3161	20411	7430		SZL	/IF C(L)=1, INPUT>-TABLE
3162	20412	3103		DCA LOW	/IF SO: INPUT=INPUT+TABLE
3163	20413	7200		CLA	
3164	20414	1104		TAD COUNT	
3165	20415	7004		RAL	/PUT THIS BIT IN ANSWER
3166	20416	2210		ISZ POINTR	/UPDATE TABLE POINTER
3167	20417	7420		SNL	/IF LINK=1, ALL DONE
3168	20420	5206		JMP POINTR-2	
3169	20421	7106		CLL RTL	/CONVERTED 2 BCD
3170	20422	7006		RTL	/CHARACTERS
3171	20423	1103		TAD LOW	/SHIFT LEFT AND ADD
3172	20424	5600		JMP I BCD	/THE THIRD

3173

3174

3175

3176

3177

3178

3179

3180

3181

3182

3183

3184

3185

3186

3187

3188

3189

3190

3191

3192

3193

3194

3195

3196

3197

3198

3199

3200

3201

3202

3203

3204

3205

3206

3207

3208

3209

3210

3211

3212

/ ROUTINE TO STORE A DATA INTO FIELD 2 OR 3 FF
/ FROM THE FL.P.AC.

20425	7402	FNSTO,	HLT	
20426	1625		TAD I FNSTO	/GET FIELD
20427	3240		DCA FNSTO1	/AND SET IT
20430	1625		TAD I FNSTO	
20431	3250		DCA FNSTO2	
20432	6213		CDF CIF 10	
20433	4523		JMS I IGET1	/GET 3. APG.
20434	6211		CDF 10	
20435	1474		TAD I IFLAC1	
20436	7421		MGL	
20437	1473		TAD I IFLAC	
20440	7402	FNSTO1,	HLT	/CDF 20 ETC.
20441	3410		DCA I AUTO	
20442	7501		MGA	
20443	3410		DCA I AUTO	
20444	6211		CDF 10	
20445	1476		TAD I IFLAC3	
20446	7421		MGL	
20447	1475		TAD I IFLAC2	
20450	7402	FNSTO2,	HLT	/CDF 20 ETC.
20451	3410		DCA I AUTO	
20452	7421		MGL	
			IFZERO FOURWD <	
		EX7,	CLA CLL>	
			IFNZRO FOURWD <	
20453	3410	EX7,	DCA I AUTO>	
20454	5005		JMP PFOCAL	/RETURN TO FOCAL
2455		CONT11=.		

3213

3214

3215

3216

/ FUNCTION TO CONTROLLE THE AX23 A/D-CONVERTER

3217

3218

3219

3220

2001 FIELD 1

3221

3222

3223

3224

3225

3263 *FOLLOW1

3226

3227

3228

3229

3230

13263

4453

XFLAB,

INTEGER

/GET 1. ARGUMENT

3231

13264

6223

CDF CIP 20

3232

13265

5666

JMP I .+1

3233

13266

0455

FLAB

3234

3235

3236

3267 FOLLOW2=.

3237

3238

3239

3240

3241

3242

3243

0002 FIELD 2

3244

3245

3246

3247

0455 *CONTI1

3248

3249

3250

3251

20455

7421

FLAB,

MCL

3252

20456

7501

MCA

3253

20457

7510

SFA

/1. APC. POSITIVE ?

3254

20460

5510

JMP I ERROR

/NO: ERROR

3255

20461

1271

TAD M12

/YES

3256

20462

7740

SMA SIA CLA

/WAS IT > 10 ?

3257

20463

5512

JMP I ERROR

/YES: ERROR

3258

20464

7501

MCA

/NO

3259

20465

1270

TAD IFLABN

3260

20466

5667

JMP I .+1

3261

20467

2563

WAYOUT

3262

3263

3264

3265

3266

20470

0472

IFLABN,

FLABN

3267

20471

7766

M12,

-12

3268

20472

0505

FLABN,

CLR

/CLEAR ALL REGISTERS

3269

20473

0517

FLO

/PLOT WITH X-Y-RECORDER

3270

20474

0633

PULSE

/PULSE AT S2, R1, R2, R3

3271

20475

0641

RELAY

/SWITCH S2, R1, R2, R3 ON/OFF

3272

20476

0652

RCLOCK

/RC-CLOCK ON/OFF

3273

20477

0513

RCWAIT

/WAIT FOR RC-CLOCK FLAG

3274

20500

0662

DEL

/WAIT FOR N RC-CL. FLAGS

3275

20501

0550

CLOCK

/SYNCHRONIZE CLOCK

3276

20502

0611

EXTEPN

/READ EXTERNAL REGISTER

3277

20503

1305

NEWLAB,

NOFUNC

/POINTERS FOR NEW FUNCTIONS

3278

20504

1305

NOFUNC


```

3279
3280
3281
3282      / ROUTINE TO CLEAR THE EXTERNAL SENSE REGISTER
3283      / AND TO DISABLE THE ENABLE REGISTER
3284      / OF THE AX03 A/D-CONVERTER
3285      /
3286      / FOCAL FUNCTION:
3287      /
3288      /      SET DUMMY=FLAB(0)
3289
3290
3291
3292
3293
3294      20505  7340  CLR,      CLA CLL CMA
3295      20506  6334          XPCL          /CLEAR EXTERNAL SENSE
3296      20507  7302          CLA CLL          /REGISTER
3297      20510  6342          ZTEN          /DISABLE ENABLE REGISTER
3298      20511  6357          CLPK CLXX CLR   /CLEAR ALL FLAGS
3299      20512  5005          JMP PFOCAL      /RETURN TO FOCAL
3300
3301
3302
3303
3304
3305
3306      / ROUTINE TO WAIT ON A RC-CLOCK- OR
3307      / EXTERNAL-CLOCK-PULSE
3308      /
3309      / FOCAL FUNCTION:
3310      /
3311      /      SET DUMMY=FLAB(5)
3312
3313
3314
3315
3316      20513  6341  RCWAIT,  SKPK
3317      20514  5313          JMP .-1          /WAIT
3318      20515  6354          CLRK
3319      20516  5005          JMP PFOCAL      /RETURN TO FOCAL

```

```

3322
3321
3322 / ROUTINE TO CONTROL THE PEN LIFT OF A
3323 / X-Y-RECORDER WITH/WITHOUT NULL-DETEKTOR.
3324 /
3325 / WITH NULL-DETEKTOR: CHANNEL "S2" WILL GIVE
3326 / THE "SEEK-PULSE". THE "PLOT-COMPLETE-
3327 / PULSE" OF THE RECORDER MUST BE NEGATIVE
3328 / AND SEND TO THE "EXTERNAL-CLOCK-INPUT".
3329 /
3330 / WITHOUT NULL-DETEKTOR: CHANNEL "S2" GIVES A
3331 / PULSE TO CONTROL THE PEN LIFT. THE
3332 / SPEED IS CONTROLLED BY THE RC-CLOCK.
3333 /
3334 / FOCAL FUNCTION:
3335 /
3336 / SET DUMMY=FLAB(1,K1)
3337 /
3338 / K1=0 WITH NULL-DETEKTOR
3339 / 1 WITHOUT NULL-DETEKTOR
3340
3341
3342
3343
3344 20517 4024 FLO, JMS ARGUM /GET K1
3345 20520 3107 DCA TEM1 /AND SAVE IT
3346 20521 1107 TAD TEM1
3347 20522 7650 SNA CLA
3348 20523 5331 JMP FLOTN /NULL-DETEKTOR
3349 20524 1047 TAD C4002 /RC-CLOCK CONTROLLED
3350 20525 6344 OTEN
3351 20526 6354 CLPK
3352 20527 6341 SKPK
3353 20530 5327 JMP .-1
3354 20531 1054 FLOTN, TAD C5 /PULSE AT OUTPUT S2
3355 20532 6344 OTEN
3356 20533 6354 CLPK
3357 20534 6341 SKPK
3358 20535 5334 JMP .-1 /WAIT FOR PULSE
3359 20536 7330 CLA STL PAR
3360 20537 6334 XPCL
3361 20540 7300 CLA CLL
3362 20541 1107 TAD TEM1
3363 20542 7640 SZA CLA
3364 20543 1047 TAD C4002
3365 20544 1054 TAD C5
3366 20545 7040 CMA
3367 20546 6342 ZTEN
3368 20547 5005 JMP PFOCAL /RETURN TO FOCAL

```

```

3369
3370 / ROUTINE TO SYNCHRONIZE AND COUNT THE RC-CLOCK
3371 / OR EXTERNAL CLOCK TIME. THE TIME IS HELD IN
3372 / THE FL.P.AC..
3373 /
3374 / FOCAL FUNCTION:
3375 /
3376 / SET TI=FLAB(7,K1)
3377 /
3378 / K1=CLOCK FLAG COUNTER
3379 / TI=TIME OF CLOCK
3380 /
3381 / TIME=(TI/K1)*2.1 MSEC
3382
3383
3384
3385 20550 4024 CLOCK, JMS ARGUM /GET K1
3386 20551 7550 SPA SNA /K1=1,...,2047 ?
3387 20552 5512 JMP I IERFOR /NO: ERROR
3388 20553 7041 CIA
3389 20554 3104 DCA COUNT /SET COUNTER
3390 20555 3102 DCA HIGH
3391 20556 3103 DCA LOW
3392 20557 6354 CLPK
3393 20560 6341 SKPK
3394 20561 5360 JMP --1 /SYNCHRONIZE
3395 20562 6354 CLOCK1, CLPK
3396 20563 6352 CLOCK2, CLKK
3397 20564 6341 SKPK
3398 20565 7410 SKP
3399 20566 5376 JMP CLOCK3
3400 20567 6321 SKXX
3401 20570 5367 JMP --1 /WAIT 2.1 MSEC
3402 20571 2103 ISZ LOW /COUNT CRYSTAL
3403 20572 5363 JMP CLOCK2 /CLOCK FLAGS
3404 20573 2102 ISZ HIGH
3405 20574 5363 JMP CLOCK2
3406 20575 5512 JMP I IERFOR /COUNTER TO HIGH
3407 20576 2104 CLOCK3, ISZ COUNT
3408 20577 5362 JMP CLOCK1
3409 20600 1122 TAD HIGH
3410 20601 7510 SPA
3411 20602 5512 JMP I IERFOR /COUNTER K1 TO HIGH
3412 20603 6211 CLOCK4, CDF 10
3413 20604 3474 DCA I IFLAC1
3414 20605 1103 TAD LOW
3415 20606 3475 DCA I IFLAC2
3416 20607 3476 DCA I IFLAC3
3417 20610 5005 JMP PFOCAL /RETURN TO FOCAL

```

```

3413
3419
3420
3421      / ROUTINE TO READ THE EXTERNAL SENSE REGISTER
3422      / AND TO CLEAR THE BITS GIVEN WITH K1.
3423      /
3424      / FOCAL FUNCTION:
3425      /
3426      /      SET A=FLAB(3,K1)
3427      /
3428      /      K1=0, ..., 2247
3429      /      BITS GIVEN AS A DECIMAL
3430      /      NUMBER ARE CLEARED
3431      /      A: BITS OF THE EXTERNAL
3432      /      SENSE REGISTER AS A
3433      /      DECIMAL NUMBER
3434
3435
3436
3437
3438      22611  4224  EXTERN, JMS ARGUM      /GET K1
3439      22612  7510      SPA              /K=0, ..., 2247 ?
3440      22613  5512      JMP I IERFOR     /NO: ERROR
3441      22614  3107      DCA TEM1
3442      22615  6331      XRIN             /READ EXTERNAL REG.
3443      22616  3112      DCA TEM2
3444      22617  1110      TAD TEM2
3445      22622  0107      AND TEM1
3446      22621  6334      XRCL            /CLEAR BITS THAT MATCH
3447      22622  7300      CLA CLL
3448      22623  3103      DCA LOW
3449      22624  1232      TAD C13
3450      22625  6211      CDF 10
3451      22626  3473      DCA I IFLAC
3452      22627  1110      TAD TEM2
3453      22630  5631      JMP I .+1       /RETURN TO FOCAL
3454      22631  0603      CLOCK4
3455
3456      22632  0013  C13,  13

```



```
3457
3458
3459      / ROUTINE TO SEND A PULSE TO OUTPUTS
3460      / S0, R1, R2, R3; PULSE LENGTH IS
3461      / 12(S0) OR 7 (R1...R3) CYCLE TIMES.
3462      /
3463      / FOCAL FUNCTION:
3464      /
3465      /      SET DUMMY=FLAB(2,K1)
3466      /
3467      /      K1=1,....,15
3468
3469
3470
3471
3472      20633      4350      PULSEJ      JMS CHANEL      /GET OUTPUT (K1)
3473      20634      1107      TAD TEXI
3474      20635      6044      OTEN      /PULSE ON
3475      20636      7300      CLA CLL
3476      20637      4036      JMS PULSOFF      /PULSE OFF
3477      20640      5025      JMP PFOCAL      /RETURN TO FOCAL
```

```

3473
3479
3480
3481      / ROUTINE TO SWITCH THE OUTPUTS S0, R1, R2, R3
3482      / ON/OFF
3483      /
3484      / FOCAL FUNCTION:
3485      /
3486      /      SET DUMMY=FLAB(3,K1,K2)
3487      /
3488      /      K1=1,....,15
3489      /      1 - R1
3490      /      2 - R2
3491      /      4 - R3
3492      /      3 - S2
3493      /      OR ANY COMBINATION
3494      /      K2=0 DISABLE
3495      /      1 ENABLE
3496
3497
3498
3499
3500      20641  4352  RELAY,  JMS CHANEL      /GET OUTPUT (K1)
3501      20642  4024      JMS ARGUM      /GET K2
3502      20643  7650      SNA CLA        /K2=1 ?
3503      20644  5250      JMP RELOUT     /NO: DISABLE
3504      20645  1107      TAD TEM1       /YES: ENABLE
3505      20646  6344      OTEN
3506      20647  5005      JMP PFOCAL      /RETURN TO FOCAL
3507      20650  4336  RELOUT, JMS PULSOF  /DISABLE
3508      20651  5005      JMP PFOCAL      /RETURN TO FOCAL

```

```

3509
3510
3511
3512 / ROUTINE TO SWITCH THE RCL-CLOCK OR THE
3513 / EXTERNAL-CLOCK ON/OFF.
3514 /
3515 / FOCAL FUNCTION:
3516 /
3517 /      SET DUMMY=FLAB(4,K1,K2)
3518 /
3519 /      K1=1 RC-CLOCK, SLOW
3520 /      2 RC-CLOCK, FAST
3521 /      4 EXTERNAL-CLOCK
3522 /      K2=0 DISABLE
3523 /      1 ENABLE
3524
3525
3526
3527
3528
3529 20652 4015 RCLOCK, JMS RCCK /GET K1' FOR ENABLE
3530 /REGISTER
3531 20653 4024 JMS ARGUM /GET K2
3532 20654 7652 SNA CLA /K2=1 ?
3533 20655 5307 JMP DELOUT /NO: DISABLE
3534 20656 1127 TAD TEM1 /YES: ENABLE
3535 20657 6344 OTEN
3536 20660 6354 CLPK
3537 20661 5005 JMP PFOCAL /RETURN TO FOCAL

```

```

3538
3539
3540
3541      / ROUTINE TO WAIT FOR K2 PULSES OF THE
3542      / SPECIFIED CLOCK OR INPUT
3543      /
3544      / FOCAL FUNCTION:
3545      /
3546      /      SET DUMMY=FLAB(6,K1,K2)
3547      /
3548      /      K1=1 RC-CLOCK, SLOW
3549      /      2 RC-CLOCK, FAST
3550      /      3 CRYSTAL-CLOCK
3551      /      4 EXTERNAL-CLOCK
3552      /      K2=1,...,4095
3553      /      COUNTER
3554
3555
3556
3557
3558      20662  4015  DEL.      JMS PCBI      /GET K1'
3559      20663  4024      JMS AFGUN      /GET K2
3560      20664  7041      CIA
3561      20665  3110      DCA TEM2      /SAVE COUNTER
3562      20666  1107      TAD TEM1
3563      20667  7650      SNA CLA      /CRYSTAL CLOCK?
3564      20670  1061      TAD M20
3565      20671  1313      TAD CSKPK
3566      20672  3303      DCA WHATS
3567      20673  1107      TAD TEM1
3568      20674  7650      SNA CLA
3569      20675  7344      CLL STA PAL
3570      20676  1314      TAD CCLPK
3571      20677  3302      DCA WHATC
3572      20700  1107      TAD TEM1
3573      20701  6344      OTEN      /ENABLE
3574      20702  7402  WHATC,  HLT      /CLXX  CLPK
3575      20703  7402  WHATS,  HLT      /SKXX  SKPK
3576      20704  5303      JMP .-1
3577      20705  2110      ISZ TEM2      /ENOUGH?
3578      20706  5302      JMP WHATC      /NO
3579      20707  1107  DELOUT, TAD TEM1      /YES
3580      20710  7040      CMA
3581      20711  6342      ZTEN      /DISABLE
3582      20712  5005      JMP PFOCAL      /RETURN TO FOCAL
3583
3584      20713  6341  CSKPK,  SKPK
3585      20714  6354  CCLPK,  CLPK

```


3536

3537

3533

3539

/ SUBROUTINE TO SELECT RC- OR EXTERNAL-CLOCK

3590

3591

3592

3593

20715 7402

RCEX,

HLT

3594

20716 4024

JMS ARGUM

/GET K1

3595

20717 7550

SPA SNA

/IS K1 > 0 ?

3596

20720 5512

JMP I IERROR

/NO: ERROR

3597

20721 3107

DCA TEM1

3598

20722 7307

LW4

3599

20723 7041

CIA

3600

20724 1107

TAD TEM1

3601

20725 7740

SMA SZA CLA

/IS K1 > 4 ?

3602

20726 5512

JMP I IERROR

/YES: ERROR

3603

20727 1107

TAD TEM1

3604

20730 0055

AND C7

3605

20731 1046

TAD NUM0

3606

20732 3300

DCA .+1

3607

20733 7402

HLT

3608

20734 3107

DCA TEM1

/SAVE K1' FOR

3609

/ENABLE REGISTER

3610

20735 5715

JMP I RCEX

/RETURN

3611

3612

3613

3614

3615

/ SUBROUTINE TO DISABLE OUTPUTS R1, R2, R3

3616

/ AND S0.

3617

3618

3619

3620

20736 7402

PULSOF,

HLT

3621

20737 1107

TAD TEM1

3622

20740 7040

CMA

3623

20741 6342

ZTEN

/PULSE R1-R3 OFF

3624

20742 7300

CLA CLL

3625

20743 1110

TAD TEM2

3626

20744 7112

CLL RTR

3627

20745 6334

XRCL

/PULSE S0 OFF

3628

20746 7300

CLA CLL

3629

20747 5736

JMP I PULSOF

/RETURN

3630

3631

3632

/ SUBROUTINE TO CONTROL THE OUTPUTS S0
/ AND R1, R2, R3.

3633

3634

3635

3636

3637 20750 7402 CHANEL, HLT

3638 20751 4024 JMS ARGUM /GET K1

3639 20752 7550 SPA SNA /K1 > 0 ?

3640 20753 5512 JMP I IERFOR /NO: ERFOR

3641 20754 3107 DCA TEM1

3642 20755 1107 TAD TEM1

3643 20756 1376 TAD M17

3644 20757 7740 SMA SZA CLA /K1 > 15 ?

3645 20760 5512 JMP I IERFOR /YES: ERFOR

3646 20761 1107 TAD TEM1

3647 20762 0075 AND 00010

3648 20763 7110 CLL RAR

3649 20764 7012 PTR

3650 20765 3110 DCA TEM2 /OUTPUT S2

3651 20766 1107 TAD TEM1

3652 20767 0055 AND C7 /OUTPUTS R1-R3

3653 20770 7104 CLL PAL

3654 20771 7006 RTL

3655 20772 1110 TAD TEM2

3656 20773 3107 DCA TEM1 /ALL OUTPUTS: K1'

3657 20774 5750 JMP I CHANEL /RETURN

3658

3659 20775 0010 00010, 0010

3660 20776 7761 M17, -17

3661

3662

3663 1000 PAGE

3664

3665

3666 1000 CONTI2=.

```

3667
3668
3669
3670 / TAPE - OVERLAY TO PS/3 FOCAL.
3671
3672 / SUBROUTINE BY LAWRENCE MOSS
3673 / FX FUNCTION FOR RANDOM ACCESS FILES
3674 / DECUS PROGRAM LIBRARY FOCAL3-263
3675
3676
3677 / TO STORE A VALUE INTO THE BUFFER USE
3678 /      S D=FX(I,A)
3679 / WHERE "D" IS A DUMMY VARIABLE, "I" IS THE
3680 / MATRIX LOCATION TO BE USED, AND "A" IS THE
3681 / VARIABLE TO BE STORED INTO LOCATION "I".
3682 /
3683 / TO RETRIEVE A VALUE FROM THE BUFFER USE
3684 /      S X=FX(I)
3685 / WHERE VARIABLE "X" IS ASSIGNED THE VALUE
3686 / OF ARRAY ELEMENT "I".
3687
3688
3689
3690 / FOURWD=0 ASSEMBLES FOR 3 WORD VARIABLES AND
3691 /          34 VARIABLES PER BLOCK
3692 / FOURWD=1 ASSEMBLES FOR 4 WORD VARIABLES AND
3693 /          63 VARIABLES PER BLOCK
3694
3695
3696
3697      ZZ01  FIELD 1
3698
3699
3700      3267  *FOLLOW2
3701
3702
3703
3704      13267  4453  XFX,  INTEGER
3705      13270  6222      CIF 20
3706      13271  5672      JMP I .+1      / CONTINUE IN FIELD 1
3707      13272  1000      LMFY1

```

3708					
3709					
3710					
3711					
3712	13273	6203	LMDEPR,	CDF CIF 00	/LINKAGE TO FIELD 2
3713	13274	5675		JMP I .+1	
3714	13275	7304		7304	/DEVICE ERROR
3715					/ROUTINE ADDRESS
3716					
3717					
3718					
3719					
3720					
3721					
3722	13276	7402	LMTAPE,	HLT	
3723	13277	3312		DOA LMARG1	
3724	13300	7501		MCA	
3725	13301	0012		DOA LMARG3	
3726	13302	1721		TAL I OUTHPI	
3727	13303	3302		DOA DEVADR	
3728	13304	6211		CIF 10	
3729	13305	6202		CIF 0	
3730	13306	6002		IOF	/HANG THE DO-NOT-DISTURB
3731	13307	4720		JMS I DEVADR	/ SIGN
3732	13310	0000	LMARG1,	0	
3733	13311	4400	LMARG2,	4400	/OUTPUT BUFFER ADDRESS
3734	13312	0000	LMARG3,	0	
3735	13313	7720		SMA CLA	/ONLY BOTHER WITH
3736	13314	7610		SKP CLA	/FATAL ERROR
3737	13315	5273		JMP LMDERR	/DEVICE ERROR
3738	13316	6222		CIF 20	
3739	13317	5676		JMP I LMTAPE	
3740					
3741					
3742					
3743					
3744					
3745	13320	0000	DEVADR,	0	
3746	13321	0100	OUTHPI,	100	/POINTER TO OUTHND
3747					
3748					
3749					
3750					
3751					
	3322		FOLOW3=.		

3752

3753

3754

3755

0002 FIELD 2

3756

3757

3758

3759

1000 *CONT12

3760

3761

3762

3763

3764

/ TAPE OVERLAY FOR PS/3- OR OS/3-DECTAPE-SYSTEM

3765

3766

3767

3768

3769

21000

7450

LMFX1,

SNA

3770

21001

5777

JMP I IINOWLY

/ZERO: INITIALIZE

3771

/FOR INPUT

3772

21002

7510

SPA

3773

21003

5774

JMP I IERR2

/MATRIX SIZE EXCEEDED

3774

21004

3354

DCA MATLOC

/SAVE LOCATION

3775

21005

3355

DCA MATBLK

/NIGHT AS WELL ZAP COUNTER

3776

21006

1472

TAD I ICHAP

3777

21007

1356

TAD MCOMMA

/TEST FOR COMMA

3778

21010

7640

SZA CLA

/PUT FLAG=0

3779

21011

7140

CLL CMA

/GET FLAG=-1

3780

21012

3364

DCA LMOP

/SAVE OPERATION FLAG

3781

3782

3783

/ NOW CALCULATE BLOCK ADDRESS

3784

3785

21013

1354

TAD MATLOC

3786

21014

1373

LMFXA,

TAD M34

3787

21015

7510

SPA

3788

21016

5223

JMP LMFKE

/NEGATIVE: SEE IF BLOCK

3789

/IS IN CORE

3790

21017

2355

ISZ MATBLK

/INCREMENT BLOCK

3791

21020

3354

DCA MATLOC

3792

21021

1354

TAD MATLOC

3793

21022

5214

JMP LMFXA

3794

3795

3796

3797

/ SEE IF BLOCK IS IN COPE

3798	21023	6221	LMFXE,	CDF 2	/SWITCH FIELDS
3799	21024	7300		CLA CLL	
3800	21025	1765	TAD I LMOBLK		/CURRENT BLOCK
3801	21026	7041		CIA	
3802	21027	1766	TAD I LMSTK		/SUBTRACT STARTING BLOCK
3803	21030	1355	TAD MATELK		/SUBTRACT RELATIVE BLOCK
3804	21031	7650		SNA CLA	/DO I HAVE IT?
3805	21032	5260	JMP LMFXC		/YES; LET'S DO OUR THING
3806	21033	1355	TAD MATELK		/RELATIVE BLOCK, REMEMBER?
3807	21034	1772		TAD I FLNGTH	
3808	21035	7740		SMA SZA CLA	
3809	21036	5774	JMP I IEPP2		/YOU DON'T HAVE THAT
3810					/MANY, DUMMY
3811	21037	2372	ISE LMSAVE		/SHIT WRITE IF NOTHING TO
3812	21040	5243	JMP .+3		/ WRITE
3813	21041	1765	TAD I LMOBLK		/NO; FIRST WRITE
3814					/CURRENT BUFFER
3815	21042	4776	JMS I ILMWPI		
3816	21043	1766	TAD I LMSTK		/NOW GET DESIRED BLOCK
3817	21044	1355	TAD MATELK		
3818	21045	3765	DCA I LMOBLK		/UPDATE CURRENT BLOCK
3819	21046	1765	TAD I LMOBLK		
3820	21047	4775	JMS I ILMPEAD		
3821	21050	3370	DCA LMSAVE		/FLICK THE SWITCH OFF

/ MAKE SURE BLKCNT CONTAINS MAXIMUM FILE SIZE

3822					
3823					
3824					
3825					
3826	21051	1767	TAD I BLKCNT		/CURRENT MAXIMUM
3827	21052	7041		CIA	
3828	21053	1355	TAD MATELK		/NEW BLOCK LARGER?
3829	21054	7750	SFA SNA CLA		
3830	21055	5260	JMP LMFXC		/NO
3831	21056	1355	TAD MATELK		/UPDATE MAXIMUM BLOCK
3832	21057	3767	DCA I BLKCNT		

3333

3334

3335

3336

/ NOW GENERATE SLOT IN MATRIX

3337 21260 1354 LMFXC, TAD MATLOC / POSITION NUMBER

3338 IFZERO FOURWD <

3339 EX10, CLL PAL /COOL WAY TO MULTIPLY

3340 /BY THREE

3341 TAD MATLOC>

3342 IFNZRO FOURWD <

3343 21261 7126 EX10, CLL RTL /COOL WAY TO MULTIPLY

3344 /BY FOUR

3345 21262 7000 NOP>

3346 21263 1371 TAD LM4400 /BUFFER STARTING ADDRESS

3347 21264 3354 DCA MATLOC

3348 21265 2364 ISZ LMOP

3349 21266 5313 JMP LMFWD /PUT OPERATION

3350

3351

3352

/ GET

3353

3354 21267 1754 TAD I MATLOC

3355 21270 6211 CDF 10

3356 21271 3473 DCA I IFLAC

3357 21272 6201 CDF 00

3358 21273 2354 ISZ MATLOC

3359 21274 1754 TAD I MATLOC

3360 21275 6211 CDF 10

3361 21276 3474 DCA I IFLAC1

3362 21277 6201 CDF 00

3363 21100 2354 ISZ MATLOC

3364 21101 1754 TAD I MATLOC

3365 21102 6211 CDF 10

3366 21103 3475 DCA I IFLAC2

3367 21104 6201 CDF 00

3368 IFNZRO FOURWD <

3369 21105 2354 EX11, ISZ MATLOC

3370 21106 1754 TAD I MATLOC >

3371 IFZERO FOURWD <

3372 EX11, SKP

3373 NOP>

3374 21107 6211 CDF 10

3375 21110 3476 DCA I IFLAC3 /NOT USED IN 3 WORD

3376 21111 6201 CDF 00

3377 21112 5342 JMP LMEXIT

3873					
3879					
3880			/ PUT		
3881					
3882	21113	6213	LMFXD,	CDF CIF 12	
3883	21114	4523		JMS I IGET1	/EVALUATE SECOND ARGUMENT
3884	21115	6211		CDF 12	
3885	21116	1473		TAD I IFLAC	
3886	21117	6201		CDF 22	
3887	21120	3754		DCA I MATLOC	
3888	21121	2354		ISZ MATLOC	
3889	21122	6211		CDF 12	
3890	21123	1474		TAD I IFLAC1	
3891	21124	6201		CDF 22	
3892	21125	3754		DCA I MATLOC	
3893	21126	2354		ISZ MATLOC	
3894	21127	6211		CDF 12	
3895	21130	1475		TAD I IFLAC2	
3896	21131	6201		CDF 22	
3897	21132	3754		DCA I MATLOC	
3898				IFNZRO FOURWD <	
3899	21133	2354	EX12,	ISZ MATLOC	
3900	21134	6211		CDF 12	
3901	21135	1476		TAD I IFLAC3	
3902	21136	6201		CDF 22	
3903	21137	3754		DCA I MATLOC >	
3904				IFZERO FOURWD <	
3905			EX12,	JMP .+5	
3906				NOP	
3907				NOP	
3908				NOP	
3909				NOP>	
3910	21140	7142		CMA CLL	
3911	21141	3370		DCA LMSAVE	/TURN ON SAVE SWITCH
3912					
3913					
3914					
3915	21142	1357	LMEXIT,	TAD LMLAST	/GOOSE OUTPUT BUFFER
3916					/POINTER TO END
3917	21143	3761		DCA I LMOPT1	/OF BUFFER
3918	21144	7040		CMA	
3919	21145	1357		TAD LMLAST	
3920	21146	3762		DCA I LMOPT2	
3921	21147	7040		CMA	
3922	21150	3760		DCA I LMO3	/AND FUDGE WORD-OF-3
3923					/SWITCH
3924	21151	7142		CLL CMA	/GOOSE BUFFER CHARACTER
3925	21152	3763		DCA I LMOCHT	/COUNT
3926	21153	5005		JMP PFOCAL	/GO TO PFOCAL

39 27					
39 23					
39 29					
39 30				/LITERALS, POINTERS, CONSTANTS	
39 31	21154	0000	MATLOC, 0		/POSITION IN MATRIX
39 32	21155	0000	MATELK, 0		/BLOCK COUNTER
39 33	21156	7524	MCOMMA, -254		
39 34	21157	4777	LMLAST, 4777		/LAST SPOT IN BUFFER
39 35	21160	3734	LM03, 3734		/POINTER TO PACK SWITCH
39 36	21161	3772	LMOPT1, 3772		/POINTER TO REAL BUFFER
39 37	21162	3773	LMOPT2, 3773		/POINTER TO OCHCT
39 38	21163	3775	LMOCHT, 3775		/POINTER TO FILE LENGTH
39 39	21164	0000	LMOP, 0		/OPERATION FLAG
39 40	21165	3721	LMOBLK, 3721		/POINTER TO OBLK
39 41	21166	0054	LMSTBK, 0054		/POINTER TO STBLK
39 42	21167	3656	BLKCNT, 3656		/POINTER TO FILE
39 43					/LENGTH WORD
39 44	21170	0000	LMSAVE, 0		/SAVE SWITCH: -1=MUST SAVE
39 45	21171	4400	LM4400, 4400		
39 46	21172	0053	FLNGTH, 0053		/POINTER TO FILE LENGTH
39 47					
39 48				DECIMAL	
39 49				IFZERO FOURWD <	
39 50			M34, -34		/3 WORD 34 WORDS
39 51					/PER BLOCK>
39 52	21173	7701	M34, -63		/4 WORD 63 WORDS
39 53					/PER BLOCK>
39 54				OCTAL	
39 55	21174	1231	IERR2, ERROR2		
39 56	21175	1205	ILMREA, LMREAD		
39 57	21176	1200	ILMWRI, LMWRIT		
39 58	21177	1220	IINONL, INONLY		

3959				
3960				
3961	21200	7402	LMWRIT, HLT	
3962	21201	7421	MGL	/PUT IN TAPE BLOCK
3963	21202	1236	TAD LMWROP	
3964	21203	4212	JMS TAPE	
3965	21204	5600	JMP I LMWRIT	
3966				
3967				
3968				
3969				
3970				
3971	21205	7402	LMREAD, HLT	
3972	21206	7421	MGL	
3973	21207	1237	TAD LMPROP	
3974	21210	4212	JMS TAPE	
3975	21211	5605	JMP I LMREAD	
3976				
3977				
3978				
3979				
3980				
3981	21212	7402	TAPE, HLT	
3982	21213	6212	CIF 10	
3983	21214	4635	JMS I ILMTAPE	
3984	21215	6201	CDF 20	
3985	21216	6001	ION	/ TAKE DOWN THE SIGN
3986	21217	5612	JMP I TAPE	
3987				
3988				
3989				
3990				
3991				
3992	21220	7300	INONLY, CLA CLL	
3993	21221	6201	CDF 2	
3994	21222	1640	TAD I INHPT	/ADJUST HANDLER FOR INPUT
3995	21223	3641	DCA I OUTHPT	/ONLY
3996	21224	3642	DCA I POBLK	/NOW KLUDGE THE ADDRESS
3997				/POINTERS SO I DON'T THINK
3998				/I HAVE PAGE IN CORE
3999	21225	3643	DCA I PELKCT	
4000	21226	6221	CDF 20	
4001	21227	3644	DCA I FLMSAV	
4002	21230	5025	JMP PFOCAL	/THAT'S ALL SHE WROTE
4003				/FOR THIS CALL

4004
 4005
 4006 21231 6213 ERPOP2, CDF CIF 12
 4007 21232 4634 JMS I ERP2
 4008 21233 7402 HLT
 4009 21234 2726 ERR2, 2726

4010

4011

4012

4013

4014

4015

/ POINTERS ETC.

4016

4017 21235 3276 ILMTAP, LMTAPE

4018 21236 4200 LMRPOP, 4200

/TAPE OUTPUT

4019 21237 0200 LMRPOP, 0200

/TAPE INPUT

4020 21240 0273 INHPT, 73

/POINTER TO INHPT

4021 21241 0100 OUTHPT, 100

/POINTER TO OUTHPT

4022 21242 3721 POBLK, 3721

/POINTER TO CURRENT

4023 /BLOCK NUMBER

4024 21243 3656 PBLKCT, 3656

/FILE LENGTH (BLOCKS)

4025 21244 1170 FLMSAV, LMSAVE

4026

4027

4028

4029 1245 CONT13=.

```

4030
4031
4032
4033      2001  FIELD 1
4034
4035
4036
4037      3322  *FOLLOW3
4038
4039
4040
4041
4042      / INTERRUPT ROUTINE FOR THE DK3-EP
4043      / PROGRAMMABLE REAL TIME CLOCK.
4044
4045
4046
4047
4048      13322  6131  CLEARPC, CLSK      / CLOCK FLAG ?
4049      13323  5332      JMP INTERC    / NO: RETURN TO FOCAL
4050      13324  6135      CLSA           / YES: READ STATUS,
4051                                     / CLEAR FLAGS
4052      13325  7700      SMA CLA
4053      13326  5332      JMP INTERC    / IGNORE OTHER CLOCK
4054                                     / INTERRUPTS
4055      13327  2335      ISZ HIGHLOW
4056      13330  7410      SKP
4057      13331  2336      ISZ HIGHHIGH
4058      13332  7300  INTERC, CLA CLL
4059      13333  5734      JMP I .+1      / RETURN TO FOCAL
4060      13334  2650      2650
4061
4062      13335  0000  HIGHLO, 0
4063      13336  0000  HIGHHI, 0
4064
4065
4066
4067      3337  FOL03A=.

```


4068

4069

4070

4071

4072

4073

4074

4075

4076

4077

4078

4079

4080

4081

4082

4083

4084

4085

4086

4087

4088

4089

4090

4091

4092

4093

4094

4095

4096

4097

4098

4099

4100

4101

4102

4103

4104

4105

4106

4107

4108

4109

4110

4111

/ OVERLAY FOR KV31-OSMI FOCAL 71

/ SUBROUTINE BY MICHAEL B. ERICKSON

/ OVERLAY FOR KV31-OSMI FOCAL 71

/ DECUS PROGRAM LIBRARY FOCAL3-312

/ THIS PATCH PROVIDES OSMI FOCAL
/ THE ABILITY TO GENERATE
/ CHARACTERS ON A KV31 SCREEN,
/ BY USING THE VSO. THE
/ ROUTINE ARE ALSO INSERTED
/ INTO THE FUNCTION TABLE TO BE
/ USE ON A KV31 SCREEN.

/ SCOPE FUNCTIONS

/ FD(I,X,Y)-PEN DISPLAY

/ I-PEN VALUE

/ 1 : DRAW A VISIBLE VECTOR
/ TO LOCATION X & Y/ 0 : DRAW AN INVISIBLE VECTOR
/ TO LOCATION X & Y

/ -1 : DRAW A POINT AT LOCATION X & Y

/ -N : DRAW A VISIBLE CIRCLE AROUND
/ LOCATION X & Y

/ WITH RADIUS N,

/ N < -1

/ X- X COORDINATE (HORIZONTAL)

/ Y- Y COORDINATE (VERTICAL)

4112

4113

4114

4115

/ SCOPE COMMANDS:

4116

4117

4118

4119

4120

/ * WORKS ONLY IN SCOPE MODE

4121

4122

4123

4124

4125

4126

/ CTPL/A- HOME(UPPER LEFT HAND CORNER)

4127

/ CTPL/B- * LINE FEED

4128

/ CTPL/E- * CAUSES END OF PAGE CONDITION

4129

/ CTPL/H- * BACK SPACE

4130

/ CTPL/N- * UP SPACE

4131

/ CTPL/O- * CARRIAGE RETURN

4132

/ CTPL/C- INCREMENT SIZE

4133

/ CTPL/R- RESET SIZE

4134

/ CTPL/S- CHANGE TO SCOPE MODE

4135

/ CTPL/T- CHANGE TO TTY MODE

4136

/ CTPL/U- ERASE SCREEN & RETURN HOME

4137

/ CTPL/X- ERASE SCREEN

4138

4139

4140

4141

4142

/ +A - FOUT(01) - HOME

4143

/ +B - FOUT(02) - * LINE FEED

4144

/ +E - FOUT(05) - * END OF PAGE

4145

/ +H - FOUT(03) - * BACK SPACE

4146

/ +N - FOUT(14) - * UP SPACE

4147

/ +O - FOUT(15) - * CARRIAGE RETURN

4148

/ +Q - FOUT(17) - INCREMENT SIZE

4149

/ +R - FOUT(13) - RESET SIZE

4150

/ +S - FOUT(19) - SCOPE MODE

4151

/ +T - FOUT(20) - TTY MODE

4152

/ +U - FOUT(21) - ERASE & HOME

4153

/ +X - FOUT(24) - ERASE

4154

4155

4156

4157

/PATCHES TO OSMI FOCAL ITSELF

4158

4159

4160

4161

4162

4163

0001 FIELD 1

4164

4165

4166

4167

2615 *2615

4168 12615 5616

JMP I .+1

/HULGE THIS TYPING ROUTINE

4169 12616 3342

OUT1

4170 2627 *2627

4171 12627 5630

JMP I .+1

/HULGE INPUT ROUTINE

4172 12630 3371

STRAP

4173 2646 *2646

4174 12646 5647

JMP I .+1

/ADD ON SCOPE CLEARING

4175 12647 3362

CLEARPC

/ROUTINE

4176 2711 *2711

4177 12711 5712

JMP I .+1

/HULGE THIS TYPING ROUTINE

4178 12712 3351

OUT2

/ALSO

4179

4180

4181

4182

4183

4184

/NOW DOWN TO THE ACTUAL ROUTINES

4185

4186

4187

4188

4189

4190

3337 *FOLO3A

4191

4192

4193 13337 4453

XFD,

INTEGER

/GET 1. ARGUMENT

4194 13340 6223

CDF CIF 20

/LET FIELD 2 DO THE WORK

4195 13341 5742

JMP I .+1

4196 13342 1245

FDIS20

```

4197
4198
4199      13343  6223  OUT1,  CDF CIF 20      /AC CONTAINS ASCII CODE
4200      13344  4757          JMS I OUT10     /TO BE TYPED OUT
4201      13345  6044          TFC
4202      13346  3016          DCA 16           /RETURN CHAR TO FOCAL
4203      13347  5750          JMP I .+1        /RETURN TO FOCAL
4204      13350  2617          2617
4205
4206
4207
4208
4209
4210      13351  6223  OUT2,  CDF CIF 20      /WHAT A WAST
4211      13352  4757          JMS I OUT10     /CHECK CHAR IN OUTPUT
4212      13353  6046          TLS             /ONLY DIFFERENCE
4213      13354  3016          DCA 16
4214      13355  5756          JMP I .+1        /RETURN TO FOCAL
4215      13356  2713          2713
4216      13357  1400  OUT10, OUTPUT
4217
4218
4219
4220
4221
4222      13360  6052  CLEARC, CCF              /ROUTINE TO CLEAR
4223                                          /INTERRUPT BAR
4224      13361  6072          CRF              /AND READY FLAG
4225      13362  6022          PCF              /REPLACE CLEAR FUNCTIONS
4226      13363  6012          RRB              /THAT WERE DESTROYED
4227      13364  6562          VPD              /CLEAR AF04-FLAG
4228      13365  6442          VICC
4229      13366  6452          VCCF
4230      13367  5770          JMP I .+1        /RETURN TO FOCAL
4231      13370  3322          CLEARC
4232
4233
4234
4235
4236
4237      13371  6223  PTRAP,  CDF CIF 20      /A CHARACTER WAS TYPED,
4238      13372  5773          JMP I .+1        /INTERRUPTED
4239      13373  1445          TRAP              /IS OFF SO CHECK IT
4240
4241
4242
4243      3374  FOLLOW4=.

```



```

4244
4245
4246
4247          /SCOPE FUNCTIONS FD
4248          /THE HARD WORK IS DONE HERE IN FIELD 2
4249
4250
4251          0002  FIELD 2
4252
4253
4254
4255          1245  *CONTI3
4256
4257
4258
4259
4260          21245  3043  FDIS22, DCA STORE          /SAVE 1. APC.
4261          21246  3024  JMS ARGUN                /
4262          21247  3035  DCA XTEMP                /SAVE 2. APC. X
4263          21250  4024  JMS ARGUN                /
4264          21251  3036  DCA YTEMP                /SAVE 3. APC. Y
4265          21252  6002  IOF                      /INTERPRET MUST BE OFF
4266          21253  1043  TAD STORE                /PEN VALUE IN MC
4267          21254  7710  SPA CLA                  /IS IT A "-1"
4268          21255  4275  JMS DEF                    /YES JMP FOR CORRECTIONS
4269          21256  3041  DCA IPEN                  /DRAW INVISIBLE VECTOR TO
4270          21257  4711  JMS I ISCP                 /CORRECT DRIFT
4271          21260  4275  JMS DEF
4272          21261  1043  TAD STORE                /GET PEN VALUE IN MC
4273          21262  3041  DCA IPEN
4274          21263  4711  JMS I ISCP                /DRAW IT
4275          21264  6052  CCF                      /CLEAR ALL FLAGS
4276          21265  6072  CRF
4277          21266  6211  CDF 10
4278          21267  3474  DCA I IFLAC1
4279          21270  1712  TAD I IP13
4280          21271  3473  DCA I IFLAC
4281          21272  3475  DCA I IFLAC2
4282          21273  6001  ION
4283          21274  5005  JMP PFOCAL                /RETURN TO FOCAL WITH
4284                                     /INTERUPT ON

```

```
4235
4236
4237
4233 21273 7402 DEP, HLT
4239 21276 1235 TAD XTEMP
4290 21277 1310 TAD M1000
4291 21300 3031 DCA X0
4292 21301 1236 TAD YTEMP
4293 21302 1310 TAD M1000
4294 21303 3032 DCA Y0
4295 21304 5675 JMP I DEP
4296
4297
4293
4299
4300
4301 / ERROR ROUTINE
4302 / PRODUCE A FOCAL ERROR: 231.<7
4303 / "UNAVAILABLE COMMAND OR FUNCTION"
4304
4305
4306 21305 6213 ERROR, CDF CIF 10
4307 21306 5707 JMP I .+1
4308 21307 2725 ERRORS
4309
4310 1305 NOFUNC=ERROR
4311
4312
4313
4314
4315
4316 /CONSTANTS, POINTERS
4317
4318 21310 7000 M1000, -1000
4319 21311 1716 ISCP, SCP
4320 21312 0005 IP13, P13
4321
4322
4323
4324 1313 CONT3A=.
4325
4326
4327 1400 PAGE
```

```

4328
4329          /FOCAL COMES TO HERE IF ANY OUTPUT IS TO BE DONE
4330
4331
4332 21400 7402 OUTPUT, HLT          /ENTER WITH AC=CHAR
4333 21401 7410 OUTP1, SKP          // ONLY FOR USE WITH THE
4334 21402 5644          JMP I ITEST // KL3-JA INTERFACE
4335 21403 0242          AND C177    /GET LOWER BITS ONLY
4336 21404 3237          DCA CHARE
4337 21405 1237          TAD CHARE
4338 21406 4643 DELKV1, JMS I ISEARCH /CHECK FOR CONTROL CHAR
4339 21407 7640          SZA CLA     /IF AC=2 IGNORE CHAR
4340 21410 5213          JMP .+3
4341 21411 2200 OUTP3, ISZ OUTPUT
4342 21412 5231 DELKV2, JMP TP-4
4343 21413 1042          TAD SCOPE   /SCOPE MODE?
4344 21414 7700          SNA CLA
4345 21415 5235          JMP TP      /NO TYPE CHAR
4346 21416 2200          ISZ OUTPUT  /YES GENERATE CHAR ON
4347                                     /THE SCREEN
4348 21417 1031          TAD XC       /TEST FOR LINES TOO LONG
4349 21420 1237          TAD MARGIN
4350 21421 7710          SPA CLA
4351 21422 5227          JMP OUTP2   /LINE OK
4352 21423 1241          TAD C15     /TOO LONG DO DUMMY C.R.
4353 21424 4253          JMS DSFY
4354 21425 1240          TAD C12     /AND DUMMY LMSF
4355 21426 4253          JMS DSFY
4356 21427 1237 OUTP2, TAD CHARE    /GET CHAR
4357 21430 4253          JMS DSFY    /DISPLAY IT
4358 21431 6052          CCF         /CLEAR SCOPE FLAG
4359 21432 6072          CRF         /AND INTERRUPT BAP,
4360                                     /YOU NEVER KNOW
4361 21433 6213          CDF CIF 10  /RETURN TO FOCAL
4362 21434 5600          JMP I OUTPUT
4363 21435 1237 TP,      TAD CHARE
4364 21436 5231          JMP TP-4
4365 21437 0000 CHARE, 0000
4366
4367
4368
4369
4370          /CONSTANTS, POINTERS
4371
4372
4373 21440 0012 C12,      12
4374 21441 0015 C15,      15
4375 21442 0177 C177,    177
4376 21443 2027 ISEARC,  SEARCH
4377 21444 2366 ITEST,   TEST

```

4378

4379

4380

/FOCAL COMES TO HERE IF A KEY HAS BEEN TYPED

4381

4382

4383

4384

21445 6036

TRAP,

KRB

/ENTER WITH AC=0

4385

21446 0242

AND C177

/GET LOWER BITS ONLY

4386

21447 4643

DELKV3,

JMS I ISEARCH

/CHECK IF IT IS A

4387

/CTPL CHARACTER

4388

21450 6213

CDF CIF 10

4389

21451 5652

JMP I .+1

4390

21452 2631

2631

4391

4392

4393

4394

4395

4396

/THIS SUBROUTINE ACTUALLY DISPLAYS THE CHARACTER

4397

4398

4399

4400

21453 7422

DSFY,

HLT

4401

21454 3043

DCA STORE

/SAVE CHAP IN NO

4402

21455 1355

TAD ILLIS2

/USE SPECIAL ELIST

4403

/IN SEARCH

4404

21456 3760

DCA I ISEARCH

4405

21457 1043

TAD STORE

4406

21460 4643

JMS I ISEARCH

/CHECK FOR CONTROL

4407

/CHARACTER

4408

21461 7450

SNA

/CTPL CHAR FOUND?

4409

21462 5272

JMP .+10

/YES, IGNORE CHARACTER

4410

21463 3043

DCA STORE

/CHECK IF IT IS A NON

4411

21464 1043

TAD STORE

/ACTIVE CONTROL CHARACTER

4412

21465 1070

TAD M40

4413

21466 7710

SPA CLA

/IS IT A CONTROL CHARACTER

4414

21467 5272

JMP .+3

/YES, IGNORE IT

4415

21470 1043

TAD STORE

4416

21471 4757

JMS I IOUTCHM

/NO, DISPLAY CHAP

4417

21472 1354

TAD ILLIS

/RETURN ELIST INTO SRCH

4418

21473 3760

DCA I ISEARCH

4419

21474 5653

JMP I DSFY

/RETURN

4420

4421

4422

/SPECIAL PURPOSE ROUTINES

4423

4424

4425

4426	21475	7327	RESET,	LW6	/CTRL/U INCOUNTED
4427	21476	6066		EXC	/EXECUTE TO ERASE SCREEN
4428	21477	4500		JMS I ISFLAG	/WAIT FOR SCOPE FLAG
4429	21500	1040	HOME,	TAD TOP	/CTRL/A INCOUNTED
4430	21501	3032		DCA Y0	/RETURN TO HOME
4431	21502	1353		TAD M14	
4432	21503	4477		JMS I ISCALE	
4433	21504	1032		TAD Y0	
4434	21505	3032		DCA Y0	
4435	21506	1037	CR,	TAD MARGIN	/CTRL/C INCOUNTED
4436					/(RETURN KEY)
4437	21507	3031		DCA X0	/DO A CARRIAGE RETURN
4438	21510	1347		TAD C240	/RESET PEN LOCATION
4439					/AND X-Y LAST
4440	21511	4757		JMS I IOUTCHM	
4441	21512	5501		JMP I IDSFYI	
4442					
4443					
4444					
4445					
4446					
4447	21513	1346	VSPACE,	TAD C32	/CTRL/W INCOUNTED
4448					/ADD 14+14 FOR VERTICAL
4449					/SPACING
4450	21514	1353	LNDF,	TAD M14	/CTRL/B INCOUNTED
4451					/(LINE FEED KEY)
4452					/ADD VALUE FOR
4453					/DOWN SPACING
4454	21515	4477		JMS I ISCALE	/CONVERT TO SCALE
4455	21516	1032		TAD Y0	/PLACE IN Y LOCATION
4456	21517	3032		DCA Y0	
4457	21520	1032		TAD Y0	/CHECK IF AT BOTTOM
4458					/OF PAGE
4459	21521	1040		TAD TOP	
4460	21522	7740		SMA SZA CLA	/ARE WE AT THE BOTTOM?
4461	21523	5501		JMP I IDSFYI	/RETURN

4462					
4463					
4464	21524	1351	FMFD,	TAD C5252	/CTPL/E INCOUNTERED
4465					/DO AC AND LINK DISPLAY
4466	21525	6072		CPF	
4467	21526	6071	STLP,	SRF	
4468	21527	7410		SKP	
4469	21528	5337		JMP LFMFD	
4470	21531	2044		ISZ LINKL	
4471	21532	5326		JMP STLP	
4472	21533	2044		ISZ LINKL	
4473	21534	5333		JMP .-1	
4474	21535	7264		CMA CML RAL	
4475	21536	5326		JMP STLP	
4476	21537	7300	LFMFD,	CLA CLL	
4477	21540	5300		JMP HOME	
4478					
4479					
4480					
4481	21541	1352	BNSP,	TAD M7	/CTPL/H INCOUNTERED
4482					/VALUE FOR A BAKSPACE
4483	21542	4477		JMS I ISCALE	/CONVERT TO SCALE
4484	21543	1031		TAD X0	/PLACE IN X LOCATION
4485	21544	3031		DCA X0	
4486	21545	5501		JMP I IDSPYI	/RETURN
4487					
4488					
4489					
4490					
4491			/CONSTANTS, POINTERS		
4492					
4493	21546	0030	C30,	30	
4494	21547	0240	C240,	240	
4495	21550	1000	C1000,	1000	
4496	21551	5252	C5252,	5252	
4497	21552	7771	M7,	-7	
4498	21553	7764	M14,	-14	
4499	21554	2055	IDLIS,	DLIST-1	
4500	21555	2074	IDLIS2,	DLIST2-1	
4501	21556	1754	IEXECU,	EXECUTE	
4502	21557	1615	IOUTCH,	OUTCHM	
4503	21560	2252	ISFCH,	SRCH	

```

4524
4525
4526
4527 21561 1031 CIRCLE, TAD X2
4528 21562 3031 DCA X0
4529 21563 4756 JMS I IEXECUTE
4530 21564 1041 TAD IPEN
4531 21565 7041 CIA
4532 21566 1031 TAD X0
4533 21567 3031 DCA X0
4534 21570 1041 TAD IPEN
4535 21571 7001 IAC
4536 21572 7640 SZA CLA / -1 : POINT PLOT
4537 21573 1350 TAD C1000 /<-1 : CIRCLE
4538 21574 1046 TAD C00
4539 21575 5776 JMP I .+1
4540 21576 1742 EXME

```

/DEFINE SOME MORE VARIABLES

```

4527 1600 PAGE
4528
4529 21600 0000 A, 0000 /CONTROL WORD
4530 21601 0000 B, 0000 /MASK WORD
4531 21602 0000 C, 0000 /MASK POINTER
4532 21603 0000 H, 0000 /HALFWORD SWITCH

```

/CONSTANTS

```

4533
4539 21604 2166 SAPI, SAR
4540 21605 2113 MASKS, MASK0-1

```

```

4541
4542 /COPYRIGHT BY DIGITAL EQUIPMENT CORPORATION 1969
4543 /BASIC CHARACTER GENERATOR
4544 /WRITTEN BY MURRAY RUBEN AS PART OF THE
4545 /KUB/I SOFTWARE.
4546
4547 /THIS IS THE ROUTINE WHICH DEALS WITH THE "MASK"
4548 /AND "DISPATCH" CONTROL WORDS FOR THE ACTUAL
4549 /DETERMINATION AND EXECUTION OF THE APPROPRIATE
4550 /VECTOR STROKES TO BE DISPLAYED
4551 /ON THE SCOPE.....
4552
4553
4554
4555 21606 1055 OUTDEP, TAD C7
4556 21607 4477 JMS I ISCALE
4557 21610 1035 TAD XTEMP
4558 21611 3031 DCA X2
4559 21612 1236 TAD YTEMP
4560 21613 3732 DCA Y2
4561 21614 5615 JMP I OUTCHM
4562
4563 21615 0020 OUTCHM, 0020 /DISPLAY CHARACTER.
4564 /ENTER WITH ASCII
4565 21616 0045 AND C77 /MASK TO 6 BITS
4566 21617 7124 CLL PAL /*2
4567 21620 1224 TAD SAPI /ADD DISPATCH
4568 21621 3200 DCA A
4569 21622 1600 TAD I A /THIS IS MASK WORD
4570 21623 0055 AND C7 /MASK OFF MASK BITS
4571 21624 7124 CLL PAL /*2
4572 21625 3201 DCA B
4573 21626 1201 TAD B
4574 21627 7124 CLL PAL /*4
4575 21630 1201 TAD B /*6 NOW
4576 21631 1205 TAD MASKS /6 TIMES MASK+MASK
4577 /HEAD POINTER
4578 21632 3202 DCA C /PTR FOR MASK ADDRESSES
4579 21633 1600 TAD I A
4580 21634 7110 CLL PAR
4581 21635 3201 DCA B /SETS INTENSIFY
4582 /BLANKING BITS,
4583 21636 2200 ISZ A /FIRST ALWAYS BLANKED
4584 21637 1600 TAD I A
4585 21640 3200 DCA A /RETRIEVE CONTROL WORD
4586 21641 3203 DCA H /SETS LEFT HALF
4587 21642 1031 TAD X2
4588 21643 3035 DCA XTEMP /CHAR ADVANCE
4589 21644 1032 TAD Y2
4590 21645 3036 DCA YTEMP /SAVE Y2

```


4591

4592

4593

4594

4595

4596

4597

/START LOOP FOR DRAWING VECTORS

4598 21646 1203 VA,

TAD H

4599 21647 7041

CIA

4600 21650 3203

DCA H

/RESET HALFWORD SWITCH

4601 21651 2203

ISZ H

4602 21652 2202

ISZ C

/ADVANCE POINTER ON ZERO H

4603 21653 1200

TAD A

4604 21654 7450

SNA

/TEST NEXT CONTROL BIT

4605 21655 5206

JMP OUTDEF

/ZERO MEANS ALL VECTORS

4606 21656 7124

CLL RAL

/WERE EXECUTED, SO EXIT

4607 21657 3207

DCA A

/NEXT CONTROL BIT IN LINK

4608 21660 7420

SNL

4609 21661 5246

JMP VA

/NOT AN EXECUTION

4610 21662 1203

TAD H

/2 IF RIGHT HALF, 1 IF

4611

/LEFT HALF

4612 21663 7110

CLL RAR

/INTO LINK

4613 21664 1602

TAD I C

/GET MASK ADDRESS

4614 21665 7420

SNL

4615 21666 5272

JMP .+4

/R.H.

4616 21667 7112

CLL RTR

4617 21670 7012

PTR

4618 21671 7012

RTR

/L.H.

4619 21672 3043

DCA STORE

/SAVE IN MULT. QUOTIENT

4620 21673 1043

TAD STORE

4621 21674 7012

RTR

4622 21675 7010

RAR

4623 21676 0055

AND C7

/MASK FOR X VALUE

4624 21677 4477

JMS I ISCALE

4625 21700 1035

TAD XTEMP

/UPDATE TO CURRENT

4626

/POSITION

4627 21701 3031

DCA X0

/LOAD FOR EXECUTION

4628 21702 1043

TAD STORE

4629 21703 0055

AND C7

/GET Y VALUE

4630 21704 4477

JMS I ISCALE

4631 21705 1036

TAD YTEMP

4632 21706 3032

DCA Y0

/LOAD FOR EXECUTION

4633 21707 1201

TAD B

/GET PEN VALUE

4634 21710 7104

CLL RAL

/PUT IT INTO THE LINK

4635 21711 3201

DCA B

/UPDATE B

4636 21712 7004

RAL

4637 21713 3041

DCA IPEN

/PUT IT INTO IPEN

4638 21714 4316

JMS SCP

/DRAW IT

4639 21715 5246

JMP VA

/RETURN FOR MORE VECTORS

```

4640
4641
4642
4643
4644
4645
4646      21716  7402  SCP,    HLT
4647      21717  1041          TAD IPEN      /GET PEN VALUE
4648      21720  7750          SPA SNA CLA    /IS IT A POINT OR
4649                                /INVISIBLE VECTOR?
4650      21721  5335          JMP IPOINT    /YES SKIP CALCULATIONS
4651      21722  1031          TAD X0       /GET NEW X COORDINATE
4652      21723  7041          CIA
4653      21724  1033          TAD XLAST    /SUBTRACT FROM CURRENT
4654                                /LOCATION
4655      21725  4345          JMS SUB1      /IS IT LESS THAN 32 UNITS
4656      21726  5335          JMP IPOINT    /NO TOO LONG FOR
4657                                /SHORT VECTOR?
4658      21727  1032          TAD Y0       /YES CHECK Y DISTANCE
4659      21730  7041          CIA
4660      21731  1034          TAD YLAST    /AGAINST CURRENT LOCATION
4661      21732  4345          JMS SUB1      /IS IT TOO LONG?
4662      21733  5335          JMP IPOINT    /YES
4663      21734  1065          TAD C0140    /NO SET SHORT VECTOR MODE
4664      21735  3375  IPOINT, DCA FORMAT  /SAVE IT
4665      21736  1041          TAD IPEN      /GET PEN VALUE
4666      21737  7510          SPA          /IS IT POINT MODE?
4667      21740  5777          JMP I ICIRCLE /YES
4668      21741  7640          SZA CLA       /IS IT VISIBLE?
4669      21742  7001  EXXE,  IAC          /YES SET VISIBLE MODE
4670      21743  4354          JMS EXECUTE  /EXECUTE VECTOR?
4671      21744  5716          JMP I SCP     /RETURN
4672
4673
4674
4675
4676      /ROUTINE TO CHECK IF VECTOR TOO SHORT
4677
4678
4679
4680      21745  7402  SUB1,    HLT
4681      21746  7510          SPA          /TAKE ABSOLUTE VALUE
4682      21747  7041          CIA
4683      21750  1070          TAD M40
4684      21751  7750          SNA SPA CLA    /IS IT TOO LONG
4685      21752  2345          ISZ SUB1      /NO CHANGE RETURN LOCATION
4686      21753  5745          JMP I SUB1

```

4637					
4638					
4639					
4690	21754	7402	EXECUT, HLT		
4691	21755	1376	TAD C0440	/VALUE FOR LONG	
4692				/INVISIBLE VECTOR	
4693	21756	1375	TAD FOPMAT	/MAKE NECESSARY CHANGES	
4694	21757	6063	LDF	/LOAD FORMAT	
4695	21760	3375	DCA FOPMAT	/SAVE FORMAT	
4696	21761	1031	TAD X0	/GET X	
4697	21762	6064	LDF	/LOAD IT	
4698	21763	3033	DCA XLAST	/UPDATE PEN LOCATION	
4699	21764	4500	JMS I ISFLAG	/WAIT FOR READY FLAG	
4700	21765	1032	TAD Y0	/GET Y	
4701	21766	6065	LDF		
4702	21767	3034	DCA YLAST	/UPDATE PEN LOCATION	
4703	21770	4500	JMS I ISFLAG	/WAIT FOR READY FLAG	
4704	21771	1375	TAD FOPMAT	/GET FOPMAT	
4705	21772	6066	EMC	/EXECUTE VECTOR	
4706	21773	4500	JMS I ISFLAG	/ALWAYS EXIT WITH FLAG SET	
4707	21774	5754	JMP I EXECUTE	/RETURN	
4708					
4709					
4710					
4711			/CONSTANTS, VARIABLE		
4712					
4713	21775	0200	FORMAT, 0200		
4714	21776	0440	C2440, 0440		
4715	21777	1561	ICIRCL, CIRCLE		

4716

4717

4718

4719

/ROUTINE TO WAIT FOR READY FLAG

4720

4721

4722 22000 7402 SFLAG, HLT

4723 22001 7300 CLA CLL

4724 22002 6071 SRF

4725 22003 5202 JMP .-1

4726 22004 5600 JMP I SFLAG

4727

4728

4729

4730

4731

/SCALING ROUTINE

4732

4733

4734 22005 7402 SCALE, HLT

4735 22006 3217 DCA TEMS

4736 22007 1220 TAD SCALEF

4737 22008 7041 CIA

4738 22009 3216 DCA COUNT1

4739 22010 1217 TAD TEMS

/ADD N TIMES

4740 22011 2216 ISZ COUNT1

4741 22012 5212 JMP .-2

4742 22013 5605 JMP I SCALE

4743 22014 0000 COUNT1, 0000

4744 22015 0000 TEMS, 0000

4745 22016 0001 SCALEF, 0001

4746

4747

4748

4749

4750 22021 3220 RESETS, DCA SCALEF

/CTPL/R INCOUNTERED

4751 22022 2220 INC, ISZ SCALEF

/RESET SCALE

4752 22023 5501 JMP I IDSPYI

/CTPL/G INCOUNTERED

4753 22024 7330 SETSC, CLA STL RAR

/INCPMENT SCALE

4754 22025 3042 CLEAR, DCA SCOPE

/CTPL/S INCOUNTERED

4755 22026 5501 JMP I IDSPYI

/AC=4200=SCOPE MODE

4756 22027 5202 JMP .-1

/CTPL/T INCOUNTERED

4757 22028 5605 JMP I SCALE

/NON SCOPE MODE

4758 22029 3217 TAD TEMS

/RETURN

4759 22030 7402 SFLAG, HLT


```

4760
4761
4762
4763      /THE FOLLOWING SEARCH ROUTINE CHECKS THE INPUT
4764      /ASCII CODE AGAINST THE ACTIVE CONTROL CHARACTERS
4765
4766
4767
4768
4769      22027  7402  SEARCH, HLT      /ENTER WITH ASCII CODE
4770                                     /IN AC
4771      22030  3251      DCA SAVE      /SAVE INPUT
4772                                     /ASCII CHARACTER
4773      22031  1252      TAD SRCH
4774      22032  3351      DCA PTR
4775
4776      22030  2351  SRCH2,  ISZ PTR      /FOLLOW ALONG CHAR TABLE
4777      22034  1751      TAD I PTR      /GET LIST ELEMENT
4778      22035  7457      SNA
4779      22036  5247      JMP SREXIT      /END OF TABLE AND NO MATCH
4780      22037  7041      CIA      /COMPLEMENT TO TEST
4781      22040  2351      ISZ PTR
4782      22041  1251      TAD SAVE      /NOW TEST AGAINST CHAR
4783      22042  7640      SZA CLA
4784      22043  5233      JMP SRCH2      /NO MATCH FOUND TRY AGAIN
4785      22044  1751      TAD I PTR      /MATCH FOUND!!
4786      22045  3351      DCA PTR      /DO AN INDIRECT JMP
4787      22046  5751      JMP I PTR
4788      22047  1251  SREXIT, TAD SAVE      /RETURN TO CALLER
4789      22050  5627  DSFYI, JMP I SEARCH /WITH CHAR IN AC IF NO
4790                                     /MATCH FOUND
4791
4792      22051  0000  SAVE,  0000
4793      22052  2055  SRCH,  DLIST-1
4794
4795
4796
4797
4798
4799      22053  7327  ERASE,  LW6      /CTRL/X
4800      22054  6066      EXC
4801      22055  5501      JMP I DSFYI

```

4302

4303

4304

4305

4306

4307

4308

4309

/TABLE OF ACTIVE CONTROL CHAPACTERS.

4310	22056	0001	01	/CTRL/A RETURN TO HOME
4311	22057	1500	HOME	
4312	22060	0021	21	/CTRL/G INCREMENT SIZE
4313	22061	2022	INC	
4314	22062	0022	22	/CTRL/R RESET SIZE
4315	22063	2021	RESETS	
4316	22064	0023	23	/CTRL/S SCOPE MODE
4317	22065	2024	SETSC	
4318	22066	0024	24	/CTRL/T TTY MODE
4319	22067	2025	CLEAR	
4320	22070	0025	25	/CTRL/U ERASE SCREEN AND
4321	22071	1475	RESET	/RETURN HOME
4322	22072	0030	30	/CTRL/X ERASE SCREEN
4323	22073	2053	ERASE	
4324	22074	0000	0000	/TERMINATES LIST

4325

4326

4327

4328

4329

/TABLE OF SPECIAL CHARACTERS FOR SCOPE MODE ONLY

4330

4331

4332

4333

4334

4335

4336

4337

4338

4339

4340

4341

4342

4343

4344

4345

4346

4347

4348

ELIST2=.

4334	22075	0002	02	/CTRL/B LINE FEED
4335	22076	1514	LNDF	
4336	22077	0012	12	
4337	22100	1514	LNDF	
4338	22101	0005	05	/CTRL/E WAIT FOR ERASE BUTTON
4339	22102	1524	FMFD	
4340	22103	0010	10	/CTRL/H BACK SPACE
4341	22104	1541	BKSP	
4342	22105	0016	16	/CTRL/N UP SPACE
4343	22106	1513	VSPACE	
4344	22107	0017	17	/CTRL/O RETURN KEY
4345	22110	1506	CR	
4346	22111	0015	15	
4347	22112	1506	CR	
4348	22113	0000	0000	/TERMINATES LIST

4849

4850

4851

4852

4853

4854

4855

4856

4857

4858

4859

4860

4861

4862

4863

4864

4865

4866

4867

4868

4869

4870

4871

4872

4873

4874

4875

4876

4877

4878

4879

4880

4881

4882

4883

4884

4885

4886

4887

4888

4889

4890

4891

4892

4893

4894

4895

4896

4897

4898

/THE FOLLOWING PARTS ARE THE CHARACTER TABLES
/FOR THE CHARACTER GENERATOR .

/CHARACTER MASK COORDINATES

22114	0301	MASK2,	0301	/D,J,S,Z,X,I
22115	2143		2143	
22116	4145		4145	
22117	2705		2705	
22120	0747		0747	
22121	0301		0301	
22122	0701	MASK1,	0701	/L,U,V,W,X,Y,I,T,N,M,C,D,G),I
22123	0706		0706	
22124	2447		2447	
22125	2724		2724	
22126	2141		2141	
22127	0147		0147	
22130	0141	MASK2,	0141	/2,S,4,9,7,Z,S,/, \,<,>
22131	0747		0747	
22132	4404		4404	
22133	0747		0747	
22134	0141		0141	
22135	2127		2127	
22136	0424	MASK3,	0424	/A,C,E,F,G,H,N,O,Q,P,R,S,6,3,L,E
22137	3444		3444	
22140	4147		4147	
22141	0701		0701	
22142	4144		4144	
22143	2404		2404	
22144	2622	MASK4,	2622	/I,E,A,.,-,+,*
22145	4305		4305	
22146	4503		4503	
22147	0426		0426	
22150	4404		4404	
22151	0000	PTP,	0000	
22152	4525	MASK5,	4525	/E, #, =
22153	0503		0503	
22154	2343		2343	
22155	4721		4721	
22156	2707		2707	
22157	0141		0141	
22160	0627	MASK6,	0627	/",',;,:;?,!,,,
22161	2525		2525	
22162	4524		4524	
22163	2323		2323	
22164	1121		1121	
22165	2147		2147	

4399				
4900				
4901	2166	SAP=.	/CHARACTER DISPATCH TABLE	
4902				
4903				
4904				
4905	22166	7745	7745	/Q
4906	22167	6347	6347	
4907	22170	5603	5603	/A
4908	22171	4760	4760	
4909	22172	7703	7703	/B
4910	22173	2175	2175	
4911	22174	7003	7003	/C
4912	22175	0170	0170	
4913	22176	7700	7700	/D
4914	22177	3551	3551	
4915	22200	5603	5603	/E
4916	22201	5100	5100	
4917	22202	5403	5403	/F
4918	22203	5160	5160	
4919	22204	7603	7603	/G
4920	22205	0176	0176	
4921	22206	5203	5203	/H
4922	22207	4760	4760	
4923	22210	5201	5201	/I
4924	22211	1156	1156	
4925	22212	7000	7000	/J
4926	22213	5404	5404	
4927	22214	6503	6503	/K
4928	22215	6172	6172	
4929	22216	6003	6003	/L
4930	22217	0070	0070	
4931	22220	7401	7401	/M
4932	22221	3304	3304	
4933	22222	7001	7001	/N
4934	22223	3005	3005	
4935	22224	7403	7403	/O
4936	22225	0370	0370	
4937	22226	7403	7403	/P
4938	22227	4560	4560	
4939	22230	7603	7603	/Q
4940	22231	2370	2370	
4941	22232	7503	7503	/R
4942	22233	4572	4572	
4943	22234	7602	7602	/S
4944	22235	6360	6360	
4945	22236	5001	5001	/T
4946	22237	1150	1150	
4947	22240	7001	7001	/U
4948	22241	6025	6025	

4949				
4950				
4951	22242	6001	6001	/V
4952	22243	4011	4011	
4953	22244	7401	7401	/W
4954	22245	6205	6205	
4955	22246	5001	5001	/X
4956	22247	1007	1007	
4957	22250	6401	6401	/Y
4958	22251	1330	1330	
4959	22252	7002	7002	/Z
4960	22253	0074	0074	
4961	22254	7001	7001	/C
4962	22255	0154	0154	
4963	22256	4002	4002	/^
4964	22257	0044	0044	
4965	22260	7000	7000	/I
4966	22261	0050	0050	
4967	22262	5404	5404	/t
4968	22263	6070	6070	
4969	22264	6404	6404	/E.A.
4970	22265	2074	2074	

4971				
4972				
4973	22266	0000	0000	/SP
4974	22267	7000	7000	/EXECUTE 3 "INVISIBLE" VECTOR STROKES.
4975	22270	5006	5006	/!
4976	22271	2046	2046	
4977	22272	5006	5006	/"
4978	22273	3201	3201	
4979	22274	5245	5245	/#
4980	22275	5572	5572	
4931	22276	7642	7642	/S
4932	22277	6363	6363	
4933	22300	6740	6740	/%
4934	22301	1675	1675	
4935	22302	6760	6760	/2
4936	22303	3637	3637	
4937	22304	4006	4006	/'
4938	22305	3000	3000	
4939	22306	6001	6001	/C
4990	22307	0124	0124	
4991	22310	6001	6001	/)
4992	22311	1202	1202	
4993	22312	5224	5224	/*
4994	22313	7700	7700	
4995	22314	5004	5004	/+
4996	22315	6014	6014	
4997	22316	4006	4006	/,
4998	22317	0030	0030	
4999	22320	4004	4004	/-
5000	22321	0014	0014	
5001	22322	4006	4006	/.
5002	22323	0060	0060	
5003	22324	4002	4002	//
5004	22325	4400	4400	

5005				
5006				
5007	22326	7601	7601	/0
5008	22327	3107	3107	
5009	22330	6401	6401	/1
5010	22331	0456	0456	
5011	22332	7602	7602	/2
5012	22333	1714	1714	
5013	22334	5503	5503	/3
5014	22335	4770	4770	
5015	22336	6402	6402	/4
5016	22337	0364	0364	
5017	22340	7600	7600	/5
5018	22341	3434	3434	
5019	22342	7603	7603	/6
5020	22343	0175	0175	
5021	22344	6002	6002	/7
5022	22345	0070	0070	
5023	22346	5703	5703	/8
5024	22347	4770	4770	
5025	22350	7402	7402	/9
5026	22351	0364	0364	
5027	22352	5006	5006	/:
5028	22353	1460	1460	
5029	22354	5006	5006	/;
5030	22355	1430	1430	
5031	22356	6002	6002	/<
5032	22357	2120	2120	
5033	22360	5005	5005	/=
5034	22361	5500	5500	
5035	22362	6002	6002	/>
5036	22363	4240	4240	
5037	22364	7506	7506	/?
5038	22365	6346	6346	

5039

5040

5041

5042

5043

5044

5045

5046

/THIS IS THE LAST OF THE CHARACTER DISPATCH TABLE.

2366 CONTI4=.

5047

5048

5049

5050

/ FOCAL FUNCTION FIC(.....)

5051

5052

5053

5054

5055

5056

0001 FIELD 1

5057

5058

5059

3374 *FOLLOW4

5060

5061

10374

4453

KFIC,

INTEGER

/ GET FIRST ARGUMENT

5062

10375

7421

MCL

/ SAVE IT

5063

10376

4245

JMS GETARG

/ EVALUATE NEXT ARGUMENT

5064

10377

6223

OLF OIF 20

5065

10400

5601

CMP I .+1

/ CONTINUE IN FIELD 2

5066

10401

2557

MAY

5067

5068

5069

3402 FOLLOWS=.

5070

5071

5072

5073

5074

0002 FIELD 2

5075

5076

1313 *CONT3A


```

5277
5273
5279
5280 / ROUTINE FOR INPUT DATA DURING FOCAL VIA THE
5281 / KL3-JA INTERFACE: RECEIVE DATA.
5282 /
5283 / FOCAL FUNCTION:
5284 /
5285 /     SET D=FIO(1,1)
5286 /
5287 /     I > 2
5288 /
5289 / AFTER INPUT THE NUMBER OF DATA CONTAINS
5290 / VARIABLE "D"; THAT IS:  "I"  OF THE MAXIMUM
5291 / POSSIBLE NUMBER, I(MAX)=1199.
5292
5293
5294
5295 21310 6202 INPUT,  IOF
5296 21314 1105 TAD ARG / GET SECOND ARGUMENT
5297 21315 7550 SPA SNA
5298 21316 5512 JMP I IERPOP / ERPOP: INPUT < 1 !!
5299 21317 7041 CIA
5300 21320 3104 DCA COUNT
5301 21321 1513 TAD I IPOIN
5302 21322 3212 DCA AUTO / SET POINTER.
5303 21323 6442 VICC
5304 21324 1212 INLOOP, TAD AUTO
5305 21325 1264 TAD C205
5306 21326 7450 SNA
5307 21327 5515 JMP I IILoop
5308 21330 7021 IAC
5309 21331 7650 SNA CLA
5310 21332 5515 JMP I IILoop
5311 21333 4524 JMS I IDICONV
5312 21334 0102 HIGH
5313 21335 1102 TAD HIGH
5314 21336 3410 DCA I AUTO
5315 21337 1103 TAD LOW
5316 21340 3410 DCA I AUTO
5317 21341 2104 ISZ COUNT / ENOUGH ?
5318 21342 5324 JMP INLOOP / NO. CONTINUE.
5319 21343 7300 ILOOP, CLA CLL / YES.
5320 21344 7421 MCL
5321 21345 1105 TAD ARG
5322 21346 1104 TAD COUNT
5323 21347 6001 ION
5324 21350 5362 JMP FLOAT2 / EXIT.

```

```

5125
5126
5127      / ROUTINE TO GET AND FLOAT A DATA OF THE
5128      / INPUT-ROUTINE.
5129      /
5130      / FOCAL FUNCTION:
5131      /
5132      /      SET A=FIO(2,1)
5133      /                               I > 2
5134      /
5135      / THE VARIABLE "A" CONTAINS THE DATA.
5136
5137
5138
5139
5140      21351  1105  FLOAT,  TAD ARG
5141      21352  7550          SPA SWA
5142      21353  5512          JMP I IEPFOR      / IEPFOR: INPUT < 1 !!
5143      21354  7104          CLL PAL
5144      21355  1114          TAD POINT      / SET DATA POINTER.
5145      21356  3010          DCA AUTO
5146      21357  1410          TAD I AUTO      / GET HIGH
5147      21360  7421          MQL
5148      21361  1410          TAD I AUTO      / GET LOW
5149      21362  6211  FLOAT2, CDF 10
5150      21363  3475          DCA I IFLAC2
5151      21364  7501          MQA
5152      21365  3474          DCA I IFLAC1
5153      21366  3476          DCA I IFLAC3
5154      21367  6221          CDF 20
5155      21370  5025          JMP PFOCAL      / EXIT.
5156
5157
5158
5159
5160
5161
5162      / SUBROUTINE TO TRANSMIT DATA VIA
5163      / THE KL3-JA INTERFACE.
5164
5165
5166      21371  7402  TRANSM, HLT
5167      21372  6002          IOF
5168      21373  6456          VOLS      / TRANSMIT
5169      21374  6451          VOSF      / READY?
5170      21375  5374          JMP .-1      / NO
5171      21376  6452          VOCF
5172      21377  5771          JMP I TRANSMIT / YES, EXIT

```

5173

5174

5175

5176

2366 *CONT14

5177

5178

5179

5180

5181

/ ROUTINE FOR OUTPUT TO KLS-JA INTERFACE

5182

/ CHECK FOR TERMINATOR, TRANSMIT DATA.

5183

5184

5185 22366 7421 TEST, MQL / SAVE OUTPUT

5186 22367 2127 ISZ TESTE

5187 22372 7000 NOP

5188 22371 1127 TAD TESTE

5189 22372 7712 SPA CLA

5190 22373 5377 JMT TEST1 / JUMP

5191 22374 3107 DCA TESTE

5192 22375 7521 MCA

5193 22376 4516 JMS I ITRANSMIT / TRANSMIT

5194 22377 7000 TEST1, CLA CLL

5195 22402 5621 JMP I .+1 / RETURN TO FOCAL

5196 22401 1411 OUTP3

```

5197
5198      /DOUBLE PRECISION DECIMAL-TO-BINARY CONVERSION
5199      /AND INPUT
5200      /CALLING SEQUENCE:      /AC IGNORED
5201      /      JMS DICONV /SUBROUTINE CALLED
5202      /      ADDRPS      /ADDRESS TO STORE
5203      /      /HIGH-ORDER WORD
5204      /      /LOW -ORDER WORD
5205      /      /IN ADDRESS+1
5206
5207
5208      22402  7402  DICONV, HLT
5209      22403  7300  CLA CLL      /INITIALIZE PROGRAM
5210      /SWITCHES
5211      22404  1323  TAD DISET1+1
5212      22405  3234  DCA DICTPL
5213      22406  1323  TAD DISET1+1
5214      22407  3226  DCA DIXSW1
5215      22410  1602  TAD I DICONV      /PICK UP ADDRESS TO
5216      /STORE HIGH-ORD. WORD
5217      22411  3110  DCA DIGET
5218      22412  3344  DCA DINIME      /CLEAR LOCATIONS USED TO
5219      /HOLD INCOMING
5220      22413  3345  DCA DILOHD      /NUMBER
5221      22414  3107  DCA DINEGI      /CLEAR NEGATIVE SWITCH
5222      22415  5266  JMP DIIN
5223      22416  7501  DIFROC, MCA      / GET CHARACTER
5224      22417  1340  TAD DIFBUT
5225      22420  1341  TAD DIM260
5226      22421  7510  SPA      /IS IT LESS THAN 260
5227      / (IE. "0")?
5228      22422  5234  JMP DICTPL      /YES, TRANSFER TO SEE
5229      /WHAT CHARACTER IT IS
5230      22423  1342  TAD DIM271
5231      22424  7740  SMA SZA CLA      /IS IT GREATER THAN 271
5232      / (IE. "9")?
5233      22425  5234  JMP DICTPL      /YES, TRANSFER TO SEE
5234      /WHAT CHARACTER IT IS
5235      22426  7300  DIXSW1, CLA CLL      /NO, FIRST CHARACTER WAS
5236      /A DECIMAL DIGIT
5237      22427  1233  TAD .+4      /CLOSE SWITCH TO GO TO
5238      /"DINMBP" NEXT
5239      22430  3226  DCA .-2
5240      22431  1240  TAD DINMBR-2      /SET SWITCH TO SENSE
5241      /TERMINATING CHAR.
5242      22432  3234  DCA DICTPL
5243      22433  5242  JMP DINMBR

```


5244					
5245	22434	7200	DICTPL,	CLA	/CONTINUE CHECKING TO
5246					/DETERMINE CHAP.
5247	22435	7501		MCA	
5248	22436	1343		TAD DIMSPC	
5249	22437	7650		SNA CLA	/IS IT A "SPACE"?
5250	22440	5301		JMP DIEND	/YES
5251	22441	5266		JMP DIIN	/NO, IT WAS A TERMINATING
5252					/CHARACTER
5253	22442	1345	DINMBR,	TAD DILOHD	/STORE ASSEMBLED NUMBER
5254					/TEMPORARILY
5255	22443	3346		DCA DIXTM1	
5256	22444	1344		TAD DIHIND	
5257	22445	3347		DCA DIXTM2	
5258	22446	4327		JMS DI DSPL	/MULTIPLY CURRENT BY "16"
5259	22447	4327		JMS DI DSPL	
5260	22450	1345		TAD DILOHD	
5261	22451	1346		TAD DIXTM1	
5262	22452	3345		DCA DILOHD	
5263	22453	7004		PAL	
5264	22454	1344		TAD DIHIND	
5265	22455	1347		TAD DIXTM2	
5266	22456	3344		DCA DIHIND	
5267	22457	4327		JMS DI DSPL	
5268	22460	7501		MCA	/PICK UP CURRENT DIGIT
5269	22461	0337		AND DIXMSK	/MASK OFF HIGH-ORDER BITS
5270	22462	1345		TAD DILOHD	/ADD REMAINDER TO CURRENT
5271					/NUMBER
5272	22463	3345		DCA DILOHD	
5273	22464	7430		SZL	/DID IT OVERFLOW?
5274	22465	2344		ISZ DIHIND	/YES, CORRECT HIGH-ORDER
5275					/WORD
5276					
5277					
5278					
5279					
5280					
5281					
5282					
5283	22466	6441	DIIN,	VISF	
5284	22467	5266		JMP .-1	
5285	22470	6446		VIRE	
5286	22471	0045		AND C77	/COMPLETE 3 BIT CODE
5287	22472	1063		TAD C200	
5288	22473	7421		MGL	
5289	22474	7501		MCA	
5290	22475	1343		TAD DIMSPC	
5291	22476	7650		SNA CLA	/WAS INPUT A SPACE?
5292	22477	5266		JMP DIIN	/YES, IGNORE INPUT
5293	22500	5216		JMP DI PROC	/NO

/INPUT ROUTINE

5294

5295

5296

5297

5298

5299

5300

/TERMINATING ROUTINE

5301

22501 7200

DIEND,

CLA

5302

22502 1107

TAD DINEG1

/PICK UP NEGATIVE NUMBER

5303

22503 7112

CLL PAR

/PUT IT INTO LINK. ("1"

5304

/IF NEGATIVE)

5305

22504 1344

TAD DIHND

/PICK UP HIGH ORDER

5306

/PORTION

5307

22505 7430

SZL

/IS LINK "1"?

5308

22506 7040

CMA

/YES, NUMBER NEGATIVE.

5309

/COMPLEMENT IT

5310

22507 3510

DCA I DIGET

/STORE IT

5311

22508 1345

TAD DILOD

/PICK UP LOW-ORDER PORTION

5312

22509 7430

SZL

/IS LINK "1"?

5313

22510 7141

CLL CMA IAC

/YES, TWO'S COMP.IT. IF

5314

/OVERFLOW, LINK=1

5315

22511 7430

SZL

/IS LINK "1"?

5316

22512 2512

ISZ I DIGET

/INDEX HIGH-ORDER PORTION

5317

22513 7000

NOP

/TAKES CARE WHEN HIGH-

5318

/ORDER PORTION = 0

5319

22514 2110

ISZ DIGET

/INDEX POINTER FOR LOW-

5320

/ORDER PORTION

5321

22515 3510

DCA I DIGET

/STORE LOW-ORDER PORTION

5322

/OF NUMBER

5323

22516 2202

ISZ DICONV

/INDEX FOR CORRECT RETURN

5324

22517 5602

JMP I DICONV

/RETURN

5325

22518 2107

DISET1,

ISZ DINEG1

/SET NEGATIVE SWITCH

5326

22519 7300

CLA CLL

/CLOSE SWITCH TO TRANSFER

5327

/TO TERMINATION

5328

22520 1241

TAD DINMER-1

5329

22521 3234

DCA DICTRL

5330

22522 5266

JMP DIIN

/JUMP TO WAIT FOR NEXT

5331

/CHARACTER

5332

5333

5334

5335

5336

/DOUBLE PRECISION LEFT SHIFT (X2)

5337	22527	0000	DIDSPL,	0
5338	22530	1345	TAD DILOND	
5339	22531	7104	CLL RAL	
5340	22532	3345	DCA DILOND	
5341	22533	1344	TAD DIHIND	
5342	22534	7004	RAL	
5343	22535	3344	DCA DIHIND	
5344	22536	5727	JMP I DIDSPL	

5345

5346

5347

5348

5349

5350

/CONSTANTS AND VARIABLES

5351

5352	22537	0017	DIMSK,	17	/MASK FOR LAST FOUR BITS
5353	22540	7441	DIBUT,	-337	/CODE FOR ERASE
5354	22541	0057	DIM260,	57	/NUMBER USED TO GENERATE
5355					/CODE "260"
5356	22542	7767	DIM271,	-11	/NUMBER USED TO GENERATE
5357					/CODE "271"
5358	22543	7540	DIMSPC,	-240	/CODE FOR SPACE
5359		0107	DINEG1=TIMER		
5360		0110	DIGET=TIMER+1		
5361	22544	0000	DIHIND,	0	
5362	22545	0000	DILOND,	0	
5363	22546	0000	DIXTM1,	0	
5364	22547	0000	DIXTM2,	0	

5365

5366

5367

5368

5369

/ CHECK WHICH SUBROUTINE: -1 OR 0 OR +1 ?

5370

5371

5372 22550 3105 WAY, DCA ARG / SAVE SECOND ARGUMENT

5373 22551 7501 MGA

5374 22552 1367 TAD MMS / GET FIRST ARGUMENT

5375 22553 7700 SMA CLA / > +4 ?

5376 22554 5512 JMP I IERROR / YES: ERROR

5377 22555 7501 MGA

5378 22556 7001 IAC

5379 22557 7710 SPA CLA / < -1 ?

5380 22558 5512 JMP I IERROR / YES: ERROR

5381 22559 7501 MGA

5382 22560 1111 TAD WHICH

5383 22561 3107 WAYOUT, DCA TIMEP

5384 22562 1507 TAD I TIMEP

5385 22563 3107 DCA TIMEP

5386 22564 5507 JMP I TIMEP / GO TO FUNCTION

5387

5388

5389

5390

5391

/ CONSTANTS AND POINTERS

5392

5393 22567 7773 MMS, -5

5394 22570 2576 OUTPU /FOR DATA TRANSMIT

5395 22571 1351 WHICH, FLOAT /GET DATA FROM DATA BUFFER

5396 22572 1313 INPUT /RECEIVE DATA

5397 22573 1305 NEWFIO, NOFUNC /POINTERS FOR NEW FUNCTIONS

5398 22574 1305 NOFUNC

5399 22575 1305 NOFUNC


```

5422
5421
5422      / ROUTINE FOR OUTPUT DURING FOCAL:
5423      /
5424      / FOCAL FUNCTION:
5425      /
5426      / SET DUMMY=FIO(-1,A)
5427      /
5428      /           A = 0   OUTPUT ONTO TTY
5429      /           A > 0   OUTPUT TO KL3-JA
5430      /           TRANSMIT DATA
5431
5432
5433
5434
5435      22576   1186   OUTPUT,   TAD COUNT2      / FOR DELAY
5436      22577   3127   DCA TIMER
5437      22600   2112   DELAY,   ISZ TIMER+1      / DELAY
5438      22601   5200   JMP DELAY
5439      22602   2127   ISZ TIMER
5440      22603   5200   JMP DELAY
5441      22604   1125   TAD AFG      / WHAT MODE?
5442      22605   7510   SPA      / NEGATIVE MODE?
5443      22606   5512   JMP I IEPROP / YES: EPROP !!
5444      22607   7650   SNA CLA      / NO: MODE 2?
5445      22612   5220   JMP MODE2    / YES: TTY
5446      22611   1231   TAD CNOP      / V24 - OUTPUT
5447      22612   3626   DCA I IOUTPI
5448      22613   1230   TAD C212      / START V24-TRANSMIT
5449      22614   4516   JMS I ITRANSMIT / TRANSMIT
5450      22615   7346   LWM3
5451      22616   3127   DCA TESTE
5452      22617   5225   JMP PFOCAL
5453      22620   1232   MODE2,   TAD CSKP      / TTY - OUTPUT
5454      22621   3626   DCA I IOUTPI
5455      22622   1227   TAD C173      / END OF V24-TRANSMIT
5456      22623   4516   JMS I ITRANSMIT / TRANSMIT
5457      22624   7300   CLA CLL
5458      22625   5225   JMP PFOCAL      / EXIT
5459
5460
5461      / CONSTANTS
5462
5463      22626   1401   IOUTPI, OUTPI
5464      22627   0173   C173,   173
5465      22630   0212   C212,   212
5466      22631   7000   CNOP,   NOP
5467      22632   7410   CSKP,   SKP
5468
5469      2633   CONTIS=.

```

```

5450
5451
5452
5453
5454          / FOCAL FUNCTION          FADC(...)
5455
5456
5457
5458
5459          2001  FIELD 1
5460
5461
5462          1343  *1343
5463
5464          11343  4453  XADC,  INTEGER      / GET 1. ARGUMENT
5465          11344  6223                CDF CIF 20  / CONTINUE FUNCTION FADC
5466          11345  5746                JMP I .+1    / IN FIELD 2.
5467          11346  2633                ADC
5468
5469
5470
5471
5472          11347  6220                FLINTP
5473          11350  7402  FLINT,  HLT
5474          11351  4747                JMS I FLINT-1  / DO FL.F.INPUT
5475          11352  6222                CIF 20
5476          11353  5750                JMP I FLINT
5477
5478
5479
5480
5481
5482          3402  *FOLLOWS
5483
5484
5485          13402  7402  FLINF,  HLT
5486          13403  6223                CDF CIF 20
5487          13404  4606                JMS I IAFPUT    / SIMULATE AN INPUT FOR
5488          13405  5602                JMP I FLINF     / THE FLOATING-POINT-
5489                                     / ROUTINE XINPUT AND XIN.
5490          13406  0150  IAFPUT,  AFFUT
5491
5492
5493
5494
5495          3407  FOLLOW6=.
5496
5497          0002  FIELD 2
5498
5499          2633  *CONTIS

```

```

5500
5501
5502
5503 / ROUTINE FOR ANALOG TO DIGITAL CONVERSION VIA
5504 / THE AX03 A/D-CONVERTER.
5505 /
5506 / FOCAL FUNCTION:
5507 /
5508 / 1.1 SET A=FADC(K)
5509 / 1.2 IF (A-256) 1.3,1.3; SET A=A-4096
5510 / 1.3 SET A=A-256
5511 /
5512 / K = CHANNEL-NUMBER
5513 /
5514 / CHANNEL-NUMBER "K": 0 - 31
5515 /
5516 / THE VARIABLE "A" CONTAINS THE DECIMAL
5517 / DATA
5518
5519
5520
5521
5522
5523
5524 22633 3103 ADC, DCA LOW / SAVE 1. ARG.
5525 22634 4156 JMS CHECK / NEXT ARGUMENT ?
5526 22635 5246 NOAF04, JMP AF04 / YES
5527 22636 1103 NOAX03, TAD LOW / NO: AX03
5528 22637 6375 ACMX ADCV / SELECT CHANNEL, CONVERT
5529 22640 6332 SKAD
5530 22641 5240 JMP .-1 / WAIT FOR DONE
5531 22642 6362 PADC
5532 22643 6211 CDF 12
5533 22644 3475 DCA I IFLAC2
5534 22645 5004 JMP PFOCAL-1 / RETURN TO FOCAL.

```

```

5535
5536
5537
5538      / ROUTINE FOR ANALOG TO DIGITAL CONVERSION VIA
5539      / THE AX23 A/D-CONVERTER.
5540      /
5541      / FOCAL FUNCTION:
5542      /
5543      /      SET A=FADC(K,F,R,S)
5544      /                      K = CHANNEL-NUMBER
5545      /                      F = FUNCTION
5546      /                      R = RANGE
5547      /                      S = RESOLUTION
5548      /
5549      /      CHANNEL-NUMBER "K":          0 - 999
5550      /
5551      /                      (3) (12)
5552      /      FUNCTION "F":  AC-VOLTAGE      = 400  056
5553      /                      DC-VOLTAGE     = 200  123
5554      /                      RESISTANCE     = 100  64
5555      /                      FREQUENCY     = 40  00
5556      /                      PERIOD        = 20  16
5557      /                      SCAN RESET    = 10  3
5558      /                      SWITCH CLOSURE = 4  4
5559      /
5560      /      RANGE "R":          10 MV      = 1
5561      /                      100 MV       = 2
5562      /                      1000 MV      = 3
5563      /                      10 V / 10K   = 4
5564      /                      100 V / 100K = 5
5565      /                      1000 V      = 6
5566      /                      AUTO RANGE   = 7
5567      /
5568      /      RESOLUTION "S":  0.1 %      = 1
5569      /                      0.01 %     = 2
5570      /                      0.001 %    = 3
5571      /
5572      /      THE VARIABLE "A" CONTAINS THE DECIMAL
5573      /      DATA.
5574
5575
5576
5577
5578      22646  4024  AF04,  JMS ARGUM      / GET 2. ARG.: FUNCTION
5579      22647  0067      AND C774
5580      22650  7450      SNA
5581      22651  5512      JMP I IERROR    / ZERO ILLEGAL.
5582      22652  7106      CLL PTL
5583      22653  7024      RAL
5584      22654  3125      DCA ARG

```


5535				
5536				
5537	22655	4024	JMS ARGUM	/ GET 3. ARG.: RANGE
5538	22656	0055	AND C7	
5539	22657	7450	SNA	
5590	22660	5512	JMP I IERROR	/ ZERO ILLEGAL
5591	22661	7106	CLL RTL	
5592	22662	1105	TAD ARG	
5593	22663	3105	DCA ARG	
5594	22664	4024	JMS ARGUM	/ GET 4. ARG.: RESOLUTION
5595	22665	0053	AND C3	
5596	22666	7450	SNA	
5597	22667	5512	JMP I IERROR	/ ZERO ILLEGAL.
5598	22670	1105	TAD ARG	
5599	22671	6002	IOF	
5600	22672	6542	USEL	/ APP4: SELECT FUNCTION,
5601	22673	7000	CLA CLL	/ RANGE AND RESOLUTION.
5602	22674	1100	TAD LUT	
5603	22675	4505	JMS I IBCD	/ CALCULATE CHANNEL IN BCD
5604	22676	6541	CONV	/ SELECT CHANNEL, CONVERT
5605	22677	6562	UPD	/ CLEAR ENTROPY FLAG
5606	22700	7000	CLA CLL	
5607				
5608	22701	6561	VSDR	
5609	22702	5301	JMP --1	
5610	22703	6562	UPD	/ READ 1. BYTE
5611	22704	3125	DCA ARG	
5612				
5613	22705	6564	VBA	
5614	22706	6561	VSDR	
5615	22707	5306	JMP --1	
5616	22710	6562	VRD	/ READ 2. BYTE
5617	22711	3117	DCA ARG2	
5618				
5619	22712	6564	VBA	
5620	22713	6561	VSDR	
5621	22714	5313	JMP --1	
5622	22715	6562	VRD	/ READ 3. BYTE
5623	22716	3120	DCA ARG3	
5624				
5625	22717	1105	TAD ARG	
5626	22720	7710	SPA CLA	/ CHECK SIGN.
5627	22721	7344	LWM2	
5628	22722	1145	TAD C255	
5629	22723	3135	DCA AFSIGN	/ AND SET IT.

5630				
5631				
5632				
5633	22724	1105	TAD ARG	
5634	22725	0062	AND C60	
5635	22726	7112	CLL RTR	
5636	22727	7012	RTR	
5637	22730	4020	JMS AFOUT	
5638	22731	3136	DCA AFSIGN+1	/ 1. CHAP.
5639				
5640	22732	1105	TAD ARG	
5641	22733	4020	JMS AFOUT	
5642	22734	3137	DCA AFSIGN+2	/ 2. CHAP.
5643				
5644	22735	1117	TAD APG2	
5645	22736	7106	CLL RTL	
5646	22737	7006	RTL	
5647	22740	7024	RAL	
5648	22741	4020	JMS AFOUT	
5649	22742	3140	DCA AFSIGN+3	/ 3. CHAP.
5650				
5651	22743	1117	TAD APG2	
5652	22744	7112	CLL PTP	
5653	22745	7012	PTP	
5654	22746	4020	JMS AFOUT	
5655	22747	3141	DCA AFSIGN+4	/ 4. CHAP.
5656				
5657	22750	1117	TAD APG2	
5658	22751	4020	JMS AFOUT	
5659	22752	3142	DCA AFSIGN+5	/ 5. CHAP.
5660				
5661	22753	1120	TAD APG3	
5662	22754	7106	CLL RTL	
5663	22755	7006	RTL	
5664	22756	7004	RAL	
5665	22757	4020	JMS AFOUT	
5666	22760	3143	DCA AFSIGN+6	/ 6. CHAP.
5667				
5668	22761	1105	TAD ARG	
5669	22762	7106	CLL RTL	
5670	22763	7006	RTL	
5671	22764	7004	RAL	
5672	22765	0055	AND C7	
5673	22766	4020	JMS AFOUT	
5674	22767	3146	DCA AFEXP	/ EXPONENT = RANGE

5675			
5676			
5677			
5678			
5679	22770	6211	CDF 10
5680	22771	1126	TAD NEWIN
5681	22772	3530	DCA I AFLOP1 / SET NEW FL.P.INPUT
5682	22773	1126	TAD NEWIN
5683	22774	3531	DCA I AFLOP2 / SET NEW FL.P.XIN
5684	22775	1132	TAD AFINPUT
5685	22776	3124	DCA COUNT / SET DATA FOINTER
5686	22777	7240	LWM1
5687	23000	6213	CDF CIF 10
5688	23001	4533	JMS I IFLINT / FL.P.INPUT
5689	23002	1056	TAD C11
5690	23003	3524	DCA I ISORTON
5691	23004	1127	TAD OLDIN
5692	23005	3530	DCA I AFLOP1 / SET OLD FL.P.INPUT
5693	23006	1127	TAD OLDIN
5694	23007	3531	DCA I AFLOP2 / SET OLD FL.P.XIN
5695	23010	5804	JMP PFOCAL-1 / RETURN TO FOCAL

```

5696
5697
5698
5699      / ROUTINE TO SET THE PROGRAMMABLE REAL TIME
5700      / CLOCK DK3-EP TO ZERO AND TO START IT.
5701      /
5702      / FOCAL FUNCTION:
5703      /
5704      /      SET DUMMY=FW(3,C)
5705      /
5706      /
5707      /      C = CLOCK TIME FLAG AT
5708      /      "C" MILLISECONDS.
5709      /      C = 1, ..., 4096
5710
5711
5712
5713      23011      4024      CLOCKZ, JMS ARGUM      / SET R. ARG.
5714      23012      7041      CIA
5715      23013      6100      CLAB      / - ... MS INTO
5716      23014      7300      CLA CLL      / CLOCK COUNTER
5717      23015      6211      CDF 10
5718      23016      3634      DCA I INIGHX
5719      23017      3633      DCA I INIGHL      / SET COUNTER TO ZERO
5720      23020      1232      TAD ENABLE      / OVERFLOW + MORE 01 +
5721      / + RATE (1 MS) +
5722      / + INTERRUPT
5723      23021      6102      CLDE      / SET ENABLE REGISTER
5724      23022      5005      JMP PFOCAL      / RETURN TO FOCAL

```



```

5725
5726
5727
5728 / ROUTINE TO READ THE NUMBER OF CLOCK FLAGS
5729 / OF THE DK3-EP PROGRAMMABLE REAL TIME CLOCK
5730 /
5731 / FOCAL FUNCTION
5732 /
5733 /     SET TI=FN(4)
5734 /
5735 / THE VARIABLE "TI" CONTAINS THE NUMBER
5736 / OF CLOCK TIME FLAGS.
5737 /
5738 / REAL TIME = TI * C (MILLISECONDS)
5739 /
5740 / WITH E.G. C=100 YOU GET A CLOCK FLAG
5741 / AT EVERY 100 MILLISECOND; THE
5742 / MAXIMUM TIME WILL BE NOW
5743 / 30336.23 SECONDS OR
5744 / 20 HOURS 13 MINUTES AND 6.23 SECONDS.
5745
5746
5747
5748
5749 23023 6211 CLOCKP, CDF 10
5750 23024 1634 TAD I IHIGHH / GET THE TIME
5751 23025 7421 MCL
5752 23026 1633 TAD I IHIGHL
5753 23027 6221 CDF 20
5754 23030 5631 JMP I .+1 / GO TO FOCAL
5755 23031 1362 FLOAT2
5756
5757
5758
5759
5760
5761 / CONSTANT, POINTERS.
5762
5763 23032 5310 ENABLE, 5310
5764 23033 3335 IHIGHL, HIGHLOW
5765 23034 3336 IHIGHH, HIGHHIGH

```

5766
5767
5763
5769
5770
5771
5772
5773
5774
5775
5776
5777
5773
5779
5730
5731
5732
5733
5734
5735
5736
5737
5733
5739
5790
5791
5792
5793
5794
5795
5796
5797
5798
5799
5300
5301

23035 3035 POINTER, POINTER

3036 CONTIG=.

0001 FIELD 1

3407 *FOLLOW6

3407 STVAR=.

3407 FOLLOW=STVAR

/ END OF COMPLETE OVERLAY !!!!

\$

A	1620	CC240	4137	CSKRN	2713	DEVADR	3020
AA	0045	CC260	0066	C0010	2775	DICONV	2402
ACMX	6371	CC2725	4471	C0140	2065	DICTPL	2434
ACIH	0041	CD	7051	C0440	1776	DIDSPL	2527
ACIL	0042	CE	7031	C10	4365	DIENE	2521
AD	7067	CF	4771	C100	4134	DIGET	0110
ADC	2633	CHANEL	0750	C1000	1550	DIHND	2544
ADCV	6364	CHANGE	4207	C105	4347	DIIN	2466
ADDRES	0011	CHAR	0066	C11	0056	DILOHD	2545
AE	7204	CHARE	1437	C12	1440	DIMSPC	2543
AF	7042	CHECK	0156	C13	0632	DIM260	2541
AFEXP	0146	CHECKL	4403	C15	1441	DIM271	2542
AFINPU	0132	CIRCLE	1561	C17	0057	DINEG1	0107
AFLOP1	0130	CLAB	6133	C173	2627	DINMBP	2442
AFLOP2	0131	CLBA	6136	C177	1442	DIPROC	2416
AFOUT	0020	CLCA	6137	C20	0060	DIFBUT	2540
AFFUT	0150	CLDE	6102	C200	0063	DIS	0004
AFSIGN	0105	CLFAP	0005	C205	0064	DISET1	0502
AF04	0646	CLEARC	0002	C210	2600	DINKSH	0507
ANEMIN	4652	CLEARC	0060	C215	4106	DIKSV1	0406
AP0	0105	CLEN	6134	C240	1547	DIXTH1	0546
APGMA	4363	CLER	6351	C255	0145	DIXTH2	0547
APGMAX	0121	CLOCK	0550	C260	0113	DLIST	0056
ARGUM	0024	CLOCKF	0023	C2725	4361	DLIST2	0075
ARG2	0117	CLOCKZ	0011	C3	0053	DMULT	7004
ARG3	0120	CLOCK1	0562	C30	1546	DSB	0004
ASR	7415	CLOCK2	0563	C400	4350	DSFY	1453
AUTO	0010	CLOCK3	0576	C4002	0047	DSFYI	0050
A1201	4503	CLOCK4	0603	C412	4470	DVI	7407
A134	4670	CLR	0505	C413	4467	EXC	6301
A175	4671	CLPK	6354	C414	4362	EXL	6302
A2616	4656	CLSA	6135	C5	0054	DYC	6311
A2630	4660	CLSK	6131	C500	4351	DYL	6312
A2647	4662	CLXK	6352	C5252	1551	E	0042
A2712	4664	CLZE	6130	C60	0062	EAE	4600
A31	4667	CNOP	2631	C7	0055	EAEI	4370
A411	4504	CONTEN	0012	C77	0045	EAEISTO	4673
A413	4654	CONTIN	4370	C7700	4140	EAEIST1	4712
A6137	4653	CONTI0	4064	C774	0067	EAEI	4716
B	1601	CONTI1	0455	D	0041	EAE2	4741
BB	0046	CONTI2	1000	DEL	0662	EAE3	4770
B0D	0400	CONTI3	1245	DELAFC	4344	EAE4	4770
BD	7213	CONTI4	2366	DELAY	2600	EAE5	5057
BE	7060	CONTI5	2633	DELDK	4500	EAFBUT	2464
BF	7020	CONTI6	3036	DELDK1	4501	E0LEAP	7224
EKSP	1541	CONTRL	0166	DELDK2	4502	EFLINF	2460
ELXCNT	1167	CONT3A	1313	DELDK3	4462	EFOLOW	4666
C	1602	COREX	4373	DELFI0	4444	EFUN3	2017
CAM	7621	COUNT	0104	DELKV	4451	EFUN3I	0136
CC	0047	COUNT1	2016	DELDV1	1406	ENAEI	4067
CCDF	4223	COUNT2	0106	DELDV2	1412	ENABLE	3032
CCF	6052	CR	1506	DELDV3	1447	EOUT1	7236
CCLPK	0714	CRF	6272	DELLAB	4335	EOUT10	7252
CCNOP	4473	CPLF	4116	DELOUT	0707	EOUT2	7244
CC212	4135	CSKP	2632	DEP	1275	EPASE	2053

EPASEN	3222	FOLLOW1	3263	ILMTAP	1235	LMREAD	1205
EPROP	1305	FOLLOW2	3267	ILMWPI	1176	LMSAVE	1172
EPFOR2	1231	FOLLOW3	3322	ILCOF	1343	LMSTEX	1166
EPFOR5	2725	FOLLOW4	3374	IMESA	4356	LMTAPE	3276
ERR2	1234	FOLLOW5	3402	IMESA3	4672	LMWRIT	1200
ERTPAP	2455	FOLLOW6	3407	IMESSA	4472	LMWROP	1236
EVAL	1613	FOLO3A	3337	INC	2222	LM4400	1171
EXC	6066	FOPADD	4222	INHPT	1240	LNDF	1514
EXECUT	1754	FORMAT	1775	INLOOP	1324	LOPD	0046
EXFIO	6123	FOUR	7101	INOAF0	4357	LOW	0103
EXTEN	4013	FOURWD	0001	INOAX0	4360	LSR	7417
EXTEND	4000	GETAR	3252	INONLY	1220	LWM1	7240
EXTEPN	0611	GETARG	3245	INFUT	1313	LWM2	7344
EXT0	4021	GETARI	3256	INFUTT	2756	LWM3	7346
EXKE	1742	H	1600	INTEGE	4453	LV0	7000
EX1	0042	HIGH	0102	INTEFC	0002	LV1	7001
EX10	1061	HIGHHI	0036	IOUCH	1557	LV2	7026
EX11	1125	HIGHLO	0035	IOUTFI	2626	LV2000	7032
EX12	1133	HOME	1500	IPEN	2041	LV3	7025
EX2	5101	HORD	0045	IPOIN	0113	LV3777	7050
EX3	2232	IAFFUT	3406	IPOINT	1735	LV4	7007
EX4	2274	IBCD	0125	IP13	1312	LV4002	7000
EX5	0300	ICAR	0072	ISCALE	0077	LV5777	7052
EX6	0343	ICIRCL	1777	ISCP	1311	LV6	7027
EX7	0453	ICMX	6361	ISEARC	1443	LV6002	7000
F	0043	ICPLF	4355	ISFLAG	0122	MARGIN	0037
FDIS20	1245	IDICON	0124	ISORTC	0134	MASKS	1605
FELD2	0224	IDLIS	1554	ISRCH	1560	MASK0	2114
FELD20	0253	IDLIS2	1555	ITEST	1444	MASK1	2122
FELD3	0256	IDSFYI	0101	ITRANS	0116	MASK2	2130
FELD3N	0367	IELKV1	4474	JMPERR	4352	MASK3	2136
FELD30	0315	IELKV2	4475	LDA	6074	MASK4	2144
FFT	0200	IELKV3	4476	LDF	6063	MASK5	2152
FILECO	4436	IEPL	3232	LDX	6064	MASK6	2160
FLAB	0455	IERFOR	0112	LDY	6065	MATELK	1155
FLASN	0472	IERR2	1174	LFMFD	1537	MATLOC	1154
FLAC	0044	IERR3	3233	LINKL	0044	MCOMMA	1156
FLINF	3402	IERVX	3231	LMARG1	3310	MESSAGE	4064
FLINT	1350	IEXECU	1556	LMARG2	3311	MESAG1	4070
FLINTP	6200	IFLASN	0470	LMARG3	3312	MESLAM	4105
FLNGTH	1172	IFLAC	0073	LMERR	3273	MES1	4525
FLOAT	1351	IFLAC1	0074	LMEXIT	1142	MES113	4353
FLOAT2	1362	IFLAC2	0075	LMFXA	1014	MES2	4511
FLOOP	0212	IFLAC3	0076	LMFXB	1023	MES212	4354
FLOUTP	6000	IFLINT	0133	LMFXC	1060	MES3	4516
FMFD	1524	IFNSTO	0322	LMFXD	1113	MES4	4522
FNEW	0351	IFNUMB	0122	LMFX1	1000	MES5	4527
FNGET	0323	IFUNC	4371	LMLAST	1157	MES6	4534
FNGET1	0330	IGET1	0123	LMOELK	1165	MES7	4542
FNGET2	0340	IHIGHH	3034	LMOCHT	1163	MES3	4547
FNSTO	0425	IHIGHL	3033	LMOP	1164	MM5	2567
FNSTO1	0440	IICPLF	4477	LMOPT1	1161	MODE0	2620
FNSTO2	0450	IILLOOP	0115	LMOPT2	1162	MP4	7200
FNUMBE	0365	IINONL	1177	LMO3	1160	MULDIV	7121
FOLLOW	3407	ILMREA	1175	LMRDOP	1237	MUME	0364

MUY	7405	PULSE	2633	STAR11	4422	XFN	3241
M1022	1310	PULSOF	0736	STAR6	4245	XFX	3267
M12	0471	PUSHJ	4540	STLP	1526	XIN	6306
M14	1553	P13	0005	STORE	0043	XINFUT	5666
M17	0776	P177	0106	STVAR	3407	KLAST	2033
M20	0061	P4000	0124	SUB1	1745	XPCL	6334
M21	4012	RADC	6362	TABLE	0167	XPIN	6331
M254	0071	RCEN	0715	TAPE	1212	XRTL6	3204
M31	4362	RCLOCK	0652	TEMS	2017	XTEMP	0035
M40	0070	RCWAIT	0513	TEM1	0107	XZ	0031
M5	4366	RELAY	0641	TEM2	0110	YLAST	0034
M7	1552	PELOUT	0650	TEST	0066	YTEMP	0036
M34	1173	RESET	1475	TESTC	4564	YC	0030
NEWPIO	0570	RESETS	0001	TESTE	0100	YTH	6040
NEWFN	0070	PTRAP	3371	TESTI	2077		
NEWIN	0126	SAC	6062	THREE	7124		
NEWLAB	0500	SAR	2166	TIMER	0107		
NFELDC	4372	SAP1	1604	TOP	0040		
NMI	7411	SAVE	2051	TP	1435		
NOAF04	2635	SCA	7441	TRANSM	1071		
NOAX03	2636	SCALE	2005	TRAP	1445		
NOFUNC	1305	SCALEF	2020	TWO	0047		
NUM0	0046	SCL	7403	T3	0033		
OLDIN	0127	SCOPE	0042	VA	1646		
OTEN	6344	SCP	1716	VBA	6564		
OUTCHM	1615	SDA	6073	VCONV	6541		
OUTDEP	1606	SEARCH	2027	VICC	6442		
OUTDG	6154	SETSC	2024	VICF	6440		
OUTHPT	1241	SFLAG	2000	VIE	6445		
OUTHPI	3321	SHL	7413	VINX	6544		
OUTPU	2576	SIGN	7124	VIRB	6446		
OUTPUT	1400	SKAD	6332	VIRS	6444		
OUTPI	1401	SKER	6322	WISE	6445		
OUTP2	1427	SKPK	6341	VISF	6441		
OUTP3	1411	SKXX	6321	VOCF	6452		
OUT1	3343	SMIN	6136	VOFL	6450		
OUT10	3357	SNC	6051	VOLS	6456		
OUT2	3351	SORTCN	0054	VOFC	6454		
OVER1	0043	SFL	6102	VOPI	6455		
OVER2	0047	SRCH	2052	VO SF	6451		
PELKCT	1243	SRCH2	2033	VRD	6562		
PFOCAL	0005	SREXIT	2047	VSCC	6571		
FLMSAV	1244	SRF	6071	VSDR	6561		
FLO	0517	STARTE	4364	VSEL	6542		
FLOTN	0531	START0	4200	VSPACE	1513		
FOELK	1242	START1	4215	WAY	2550		
POINT	0114	START2	4220	WAYOUT	2563		
POINTE	3035	START3	4232	WHATC	0702		
FOINTR	0410	START4	4237	WHATS	0703		
PRINT	4126	START5	4247	WHICH	2571		
PRINTC	4551	START6	4252	WHICH0	0111		
PPNT	2442	START7	4324	XADC	1343		
PPNT1	6137	START8	4331	XFD	3337		
PSCONT	4363	START9	4407	XFIO	3374		
PTR	2151	STAR10	4414	XFLAB	3263		

ERRORS DETECTED: 0
LINKS GENERATED: 0

A	4529#	4563	4569	4579	4533	4534	4535	4603	4627
AA	2241#	2332	2409	2434					
ACMX	1101#	5523							
AC1H	1237#	2245							
AC1L	1233#	2246							
AD	2409#								
ADC	5467	5524#							
ADCV	1104#	5523							
ADDRES	2009	2014	2022	2709#					
AE	2434#								
AF	2332#								
AFEXP	2333#	5674							
AFINPU	2310#	5634							
AFLOP1	2308#	5631	5692						
AFLOP2	2309#	5633	5694						
AFOUT	2719#	2722	5637	5641	5643	5654	5653	5665	5670
AFPUT	2749	2431	2343#	2343	5490				
AFSIGN	2310	2304#	5609	5603	5642	5649	5655	5659	5666
AF04	5526	5573#							
ANEVIN	1937	1976#							
APG	2736#	5096	5121	5140	5372	5421	5534	5592	5590 5593
	5611	5625	5633	5640	5663				
APGMA	1367	1539	1726	1723#					
APGMAX	1592	2301#	3013						
ARGUM	2730#	2733	2913	2915	2956	3014	3044	3035	3403 3501
	3531	3559	3594	3633	4261	4260	5573	5537	5594 5710
APG2	1036	1033	2799#	2914	2920	2957	2953	2964	2015 3017
	3023	3029	5617	5644	5651	5657			
APG3	2300#	5623	5661						
ASP	1041#								
AUTO	1334	1356	1353	2703#	2969	2972	2977	3050	3073 3030
	3086	3091	3194	3196	3202	3227	5102	5104	5114 5116
	5145	5146	5143						
A1201	1733	1363#							
A134	1955	1990#							
A175	1957	1991#							
A2616	1944	1945	1930#						
A2630	1946	1947	1932#						
A2647	1948	1949	1934#						
A2712	1950	1951	1936#						
A31	1953	1939#							
A411	1739	1364#							
A413	1942	1943	1973#						
A6137	1941	1977#							
B	4530#	4572	4573	4575	4531	4633	4635		
BB	2242#	2360	2393	2442					
BCD	2305	3152#	3172						
BD	2442#								
BE	2354	2093#							
BF	2360#								
EKSP	4431#	4341							
ELKONT	3326	3332	3942#						
C	4531#	4573	4602	4613					
CAM	1043#								
CC	2243#	2350	2371	2391					
CCDF	1567	1574#							
CCF	1121#	1625	1765	2165	2501	4222	4275	4353	
CCLPK	3570	3535#							
CCNOP	1327	1355#							

CC212	1523	1510	1533#						
CC240	1495	1540#							
CC260	2721	2770#							
CC2725	1735	1733	1303	1321	1353#				
CD	2391#								
CE	2371#								
CF	1373	1375	2034#						
CHANEL	3472	3500	3637#	3657					
CHANGE	1423#	1427							
CHAR	1256#	2626	2774						
CHAPE	4336	4337	4356	4363	4365#				
CHECK	2362#	2363	2369	2930	3051	5525			
CHECKL	1676	1751#							
CIPCLE	4507#	4715							
CLAB	1059#	1636	5715						
CLBA	1064#								
CLCA	1065#								
CLDE	1061#	1639	5700						
CLDAP	4757#	4319							
CLDAPC	2174	2510	4043#	4201					
CLDAPC	4175	4222#							
CLEN	1062#	1632	1774						
CLSP	1093#	3293							
CLOCK	3275	3335#							
CLOCKP	3124	5749#							
CLOCKZ	3123	5713#							
CLOCK1	3395#	3403							
CLOCK2	3396#	3403	3405						
CLOCK3	3399	3407#							
CLOCK4	3412#	3454							
CLR	3263	3294#							
CLRK	1100#	3293	3313	3351	3356	3392	3395	3506	3535
CLSA	1063#	1634	1776	4050					
CLSK	1053#	1770	4043						
CLXX	1099#	1606	1664	3293	3396				
CLZE	1060#	1633	1775						
CNOP	5426	5446#							
CONTEN	2012	2020	2710#						
CONTIN	1433#	1439	1443						
CONTI0	1414#	1454							
CONTI1	3212#	3247							
CONTI2	3666#	3759							
CONTI3	4029#	4255							
CONTI4	5046#	5176							
CONTI5	5449#	5499							
CONTI6	5776#								
CONTEL	2376#	3154							
CONT3A	4324#	5076							
COREX	1599	1737#	1737						
COUNT	1336	1338	1466	1467	1473	1473	1563	1573	1593
	1601	2013	2024	2734#	2844	2845	2919	2932	3153
	3389	3407	5100	5117	5122	5635			3164
COUNT1	4733	4740	4743#						
COUNT2	2787#	5415							
CR	4435#	4345	4347						
CRF	1135#	1624	1764	2166	2502	4224	4276	4359	4466
CPLF	1505#	1512	1713	1359					
CSKP	5433	5447#							

0300 0304#
 00010 3647 3659#

C0142	2769#	4663							
C0440	4691	4714#							
C10	1568	1731#							
C100	1494	1537#							
C1000	4495#	4517							
C105	1610	1712#							
C11	2762#	5639							
C12	4354	4373#							
C13	3449	3456#							
C15	4352	4374#							
C17	2720	2763#							
C173	5435	5444#							
C177	4335	4375#	4335						
C20	2764#	3157							
C200	2767#	5237							
C205	2763#	2976	5125						
C212	5423	5445#							
C215	1506	1539#							
C240	4433	4494#							
C255	2332#	5623							
C260	1247#	2274	2277						
C2725	1637	1722#							
C3	2759#	5595							
C30	4447	4493#	4513						
C400	1555	1713#							
C4002	2755#	3349	3364						
C412	1323	1352#							
C413	1310	1351#							
C414	1639	1723#							
C5	2760#	3354	3365						
C500	1556	1714#							
C5252	4464	4496#							
C60	2766#	5634							
C7	2761#	3604	3652	4555	4570	4623	4629	5533	5672
C77	1474	1552	2753#	4565	5236				
C7700	1463	1541#							
C774	2771#	5579							
D	2245#	2393	2411	2444					
DEL	3274	3553#							
DELAFF2	1670	1702#							
DELAY	5417#	5413	5420						
DELDK	1339	1341	1360#						
DELDK1	1340	1361#							
DELDK2	1342	1362#							
DELDK3	1771	1339#							
DELFIO	1752	1303#							
DELKV	1761	1321#							
DELKV1	1356	4333#							
DELKV2	1357	4342#							
DELKV3	1353	4336#							
DELLAB	1661	1637#							
DELOUT	3533	3579#							
DEP	4263	4271	4233#	4295					
DEVADR	3727	3731	3745#						
DICONV	2304	5203#	5215	5323	5324				

DIEND	5252	5301#							
DIGET	5217	5310	5316	5319	5321	5362#			
DIHIND	5213	5256	5264	5266	5274	5325	5341	5342	5361#

DIIN	5222	5251	5233#	5292	5330				
DILORD	5220	5253	5260	5262	5270	5272	5311	5338	5340 5362#
DIMSPC	5243	5290	5353#						
DIM260	5225	5354#							
DIM271	5230	5356#							
DINEG1	5221	5302	5325	5359#					
DINMSR	5240	5243	5253#	5328					
DIPROC	5223#	5293							
DIREUT	5224	5353#							
DIS	1239#								
DISET1	5211	5213	5325#						
DIXMSK	5269	5352#							
DIXSW1	5214	5235#							
DIXTM1	5255	5261	5363#						
DIXTM2	5257	5265	5364#						
ELIST	4499	4793	4309#						
ELIST2	4500	4333#							
DMULT	1253#	1960	2344	2413					
DSE	1090#								
DSPY	4353	4355	4357	4400#	4419				
DSPY1	2731	4739#							
DVI	1043#	2271	2304						
DXC	1035#								
DXL	1036#								
DYC	1037#								
DYL	1033#								
E	2246#	2373	2400	2436					
EAE	1734	1923#	1969						
EAE1	1655	1734#							
EAE2	1929	1933	1959	1963	2007#	2003	2010	2011	2012 2216
EAE3	2019	2026							
EAE4	2020#	2025							
EAE5	1931	2030#							
EAE6	1935	2053#							
EAE7	1939	2079#							
EAE8	1961	2033#							
EAE9	1965	2144#							
EAFPUT	2479	2481#							
ECLER	1935	2501#							
EFLINP	1976	2477#	2430						
EFOLLOW	1952	1954	1956	1933#					
EFUN3	1261#	2700							
EFUN31	1265#								
ENABL	1633	1733#							
ENABLE	5720	5763#							
EOUT1	1931	2515#							
EOUT10	2516	2526	2532#						
EOUT2	1937	2525#							
ERASE	4799#	4323							
ERASEN	2614	2621#							
ERROR	2794	4306#	4310						
ERROR2	3955	4006#							
ERROR5	1272#	4303							

EVAL	1262#	2664	2631		
EXC	1129#	1627	4427	4785	4300
EXECUT	4501	4670	4690#	4787	
EXFIO	1979	2433#			
EXTEN	1327	1355#	1360		

EXTEND	1322	1333#			
EXTERN	3276	3433#			
EXT0	1333	1365#	1365		
EXXE	4520	4669#			
EX1	1377	2127#			
EX10	1397	1399	3339	3343#	
EX11	1401	1403	3369	3372#	
EX12	1405	3399	3905#		
EX2	1077	2162#			
EX3	1033	1035	2963	2966#	
EX4	1037	3009	3001#		
EX5	1039	3009	3041#		
EX6	1091	3039	3091#		
EX7	1093	3205	3207#		
F	2247#	2355	2362	2334	
FDIS20	4196	4260#			
FELD2	2956#	3121			
FELD20	2933#	3047			
FELD3	3014#	3122			
FELD3N	1736	1361	1362	3122#	
FELD30	3048	3054#	3055		
FFT	2913#	3120			
FILECO	1787	1790#			
FLAB	3233	3251#			
FLAEN	3266	3263#			
FLAC	1257#	2775	2776	2777	2773
FLINP	2806	5435#	5433		
FLINT	2311	5473#	5474	5476	
FLINTP	1266#	5472			
FLNGTH	3307	3946#			
FLOAT	5140#	5395			
FLOAT2	5124	5149#	5755		
FLOOP	2924#	2933			
FLOWTP	1250#	2311			
FMFD	4464#	4339			
FNEW	2657	3103#			
FNGET	2932	3053	3072#	3073	3075
FNGET1	3074	3077#			
FNGET2	3076	3035#			
FNST0	3061	3132#	3133	3135	
FNST01	3134	3193#			
FNST02	3136	3201#			
FNUMBE	2302	3120#			
FOLLOW	5793#				
FOLOW1	2635#	3225			
FOLOW2	3236#	3700			
FOLOW3	3750#	4037			
FOLOW4	4243#	5059			
FOLOW5	5069#	5431			
FOLOW6	5495#	5787			
FOLO3A	1938	4067#	4190		

FOURAT	2256#	2367	2373							
FOUR	1276#	1725	1727	2349	2352	2421	2423	2443	2452	2962
FOURWD	2965	3223	3232	3233	3242	3233	3292	3224	3226	3333
	3342	3363	3371	3393	3924	3943	3951			
GETAR	2671#	2674	2734							
GETARG	2275	2494	2662#	2666	2672	5263				
GETAP1	2679#	2633	2323							

H	4532#	4536	4593	4602	4621	4612				
HIGH	1564	1577	1531	1535	1592	2732#	3222	3226	3233	3392
	3424	3429	5112	5113						
HIGHMI	4257	4263#	5765							
HIGHLO	4255	4262#	5764							
HOME	4429#	4477	4311							
HORD	1241#	2241	2414							
IAPPUT	5437	5492#								
IBCD	2325#	5623								
ICHAR	2774#	2347	2364	3776						
ICIRCL	4667	4715#								
ICMX	1122#									
ICPLF	1642	1713#								
IDICON	2324#	5111								
IDLIS	4417	4499#								
IDLIS2	4422	4522#								
IDSPY1	2731#	4441	4461	4436	4754	4759	4321			
IELKV1	1326	1356#								
IELKV2	1325	1357#								
IELKV3	1323	1353#								
IEPL	2624	2631#								
IERROR	1715	2794#	2961	2974	2979	3222	3126	3129	3254	3257
	3337	3426	3411	3442	3596	3622	3642	3645	5293	5142
	5376	5332	5423	5531	5592	5597				
IERR2	3773	3329	3955#							
IERR3	2625	2632#								
IERVK	2623	2632#								
IEXECU	4521#	4529								
IFLAEN	2259	3266#								
IFLAC	2775#	2934	3234	3192	3451	3356	3335	4232		
IFLAC1	2776#	2917	2921	2924	2926	3232	3192	3412	3361	3392
	4273	5152								
IFLAC2	2777#	2922	2927	2929	3295	3222	3415	3366	3395	4231
	5152	5533								
IFLAC3	2778#	3293	3193	3416	3375	3921	5153			
IFLINT	2311#	5633								
IFNSTO	2934	3257	3261#							
IFNUM2	2322#	3111								
IFUNC	1595	1735#								
IGET1	2323#	3133	3333							
IHIGHH	5713	5752	5765#							
IHIGHL	5719	5752	5764#							
IICPLF	1794	1795	1359#							
IILoop	2797#	5127	5112							
IINONL	3772	3953#								
ILMREA	3322	3956#								
ILMTAP	3933	4217#								
ILMWPI	3315	3957#								
ILOOP	2797	5119#								

IMESSA	1753	1762	1772	1354#
INC	4752#	4313		
INHFT	3994	4222#		
INLOOP	5124#	5113		
INOAF2	1723	1722#		
INOAX2	1692	1721#		
INONLY	3953	3992#		
INPUT	5295#	5396		
INPUTT	1267#	2327		

INTEGE	1263#	2269	2433	2654	2665	3232	3724	4193	5261	5464
INTEPC	4249	4253	4253#							
IOUTCH	4416	4442	4522#							
IOUTP1	5427	5434	5443#							
IPEN	2749#	4269	4273	4510	4514	4637	4647	4665		
IPFIN	2795#	2972	5101							
IPINT	4652	4656	4662	4664#						
IP13	4279	4322#								
ISCALE	2779#	4432	4454	4433	4556	4624	4632			
ISCP	4270	4274	4319#							
ISEAPC	4333	4376#	4336	4426						
ISFLAG	2732#	4423	4699	4723	4726					
ISOPTC	2312#	5692								
ISPCN	4424	4413	4523#							
ITEST	4334	4377#								
ITRANS	2798#	5193	5429	5436						
JMPERR	1691	1722	1715#							
LDA	1142#									
LEF	1124#	4694								
LEX	1125#	4697								
LEY	1127#	4701								
LFMFD	4469	4476#								
LINKL	2752#	4472	4472							
LMAFG1	3723	3732#								
LMARG2	3733#									
LMAFG3	3725	3734#								
LMDEPR	3712#	3737								
LMEXIT	3877	3915#								
LMFXA	3786#	3793								
LMFXB	3733	3793#								
LMFXC	3825	3832	3337#							
LMFXD	3849	3832#								
LMFXI	3727	3769#								
LMLAST	3915	3919	3934#							
LMOELK	3822	3313	3313	3319	3942#					
LMOCHT	3925	3933#								
LMOP	3732	3843	3939#							
LMOPT1	3917	3936#								
LMOPT2	3922	3937#								
LM03	3922	3935#								
LMRDOP	3973	4219#								
LMREAD	3956	3971#	3975							
LMSAVE	3811	3821	3911	3944#	4225					
LMSTK	3822	3816	3941#							
LMTAPE	3722#	3739	4217							
LMWRIT	3957	3961#	3965							
LMWROP	3963	4213#								

LW3	4252#	4335	4337							
LORD	1242#	2242	2416							
LOW	1357	1359	1566	1536	2733#	3032	3049	3153	3159	3162
	3171	3391	3402	3414	3443	5115	5524	5527	5602	
LSP	1046#									
LWM1	1226#	1635	5636							
LWM2	1227#	5627								
LWM3	1223#	5430								
LWZ	1215#									
LW1	1216#									
LW2	1217#	1562								
LW2000	1221#									

LW3	1213#									
LV3777	1222#	1651								
LV4	1219#	3593								
LV4000	1223#									
LW5777	1224#									
LW6	1220#	1570	1626	4426	4799					
LW6000	1225#									
MARGIN	2747#	4349	4435							
MASKS	4540#	4576								
MASK0	4540	4357#								
MASK1	4363#									
MASK2	4369#									
MASK3	4375#									
MASK4	4331#									
MASK5	4337#									
MASK6	4393#									
MATELK	3775	3790	3303	3306	3317	3323	3331	3932#		
MATLOC	3774	3735	3791	3792	3337	3341	3347	3354	3353	3359
	3363	3364	3369	3370	3337	3333	3392	3393	3397	3399
	3903	3931#								
MCOMMA	3777	3933#								
MESSAGE	1463#	1464	1465	1476	1719	1354	1992			
MESAG1	1467#	1479								
MESLAM	1472	1477	1439#	1491	1497					
MES1	1644	1716	1371#							
MES113	1557	1716#								
MES2	1646	1717	1376#							
MES212	1602	1717#								
MES3	1332#	1963								
MES4	1663	1337#								
MES5	1672	1393#								
MES6	1754	1399#								
MES7	1763	1906#								
MES8	1773	1912#								
MM5	5374	5393#								
MODE0	5425	5433#								
MP4	2429#	2453								
MUL DIV	1962	2256	2417	2419#						
MUMB	3107	3119#								
MLY	1042#	1373	2356	2363	2376	2335	2394	2402	2404	2412
	2437	2445								
M1000	4290	4293	4313#							
M12	3255	3267#								
M14	4431	4450	4493#							
M17	3643	3660#								

M254	2773#	2366		
M31	1425#	1429		
M40	1492	2772#	4412	4633
M5	1565	1732#		
M7	4481	4497#		
M34	1427	3736	3949	3952#
NEWFO	5397#			
NEWFN	3125#			
NEWIN	1933	2306#	5632	5632
NEWLAB	3277#			
NFELD3	1596	1736#		
NMI	1044#			
NOAF04	1720	5526#		

NOAM03	1721	5527#								
NOFLNC	1735	1360	3125	3126	3127	3123	3129	3130	3277	3273
	4310#	5397	5393	5399						
NLM0	2754#	3625								
OLDIN	2307#	5691	5693							
OTEN	1097#	3352	3255	3474	3505	3535	3573			
OUTCHM	4522	4561	4563#							
OUTDEP	4555#	4605								
OUTDG	1252#	2303								
OUTHPT	3995	4021#								
OUTHPI	3726	3746#								
OUTPU	5394	5415#								
OUTPUT	2137	2532	4216	4332#	4341	4346	4362			
OUTP1	4333#	5443								
OUTP2	4351	4356#								
OUTP3	4341#	5196								
OUT1	4169	4199#								
OUT10	4200	4211	4216#							
OUT2	4173	4210#								
OVER1	1239#	2247								
OVER2	1243#	2243	2254	2452						
PBXCT	3999	4024#								
PFOCAL	2693#	2935	3096	3203	3299	3319	3363	3417	3477	3506
	3503	3537	3532	3926	4002	4234	5153	5432	5433	5534
	5695	5724								
FLMSAV	4021	4025#								
FLO	3269	3344#								
FLOTN	3343	3354#								
POELK	3996	4022#								
POINT	2796#	2963	5144							
POINTE	2795	2796	5774#	5774						
POINTR	3155	3160#	3166	3163						
PRINT	1496	1507	1509	1511	1522#	1527				
PRINTC	1253#	2275	2273	2300						
PPNT	2269#	2279	2326							
PPNT1	2310	2326#								
PSCONT	1342	1427#								
PTR	4774	4776	4777	4731	4735	4736	4737	4336#		
FULSE	3270	3472#								
PULSOF	3476	3507	3620#	3629						
PUSHJ	1259#	2663	2630							
P13	1245#	4320								
P177	1246#	2270								

SALC	1113#	3531		
RCEK	3529	3553	3593#	3612
RCLOCK	3272	3529#		
RWAIT	3270	3316#		
RELAY	3271	3520#		
RELOUT	3523	3507#		
RESET	4426#	4321		
RESETS	4750#	4315		
RTRAP	4172	4237#		
SAC	1123#			
SAR	4539	4901#		
SAR1	4539#	4567		
SAVE	4771	4732	4733	4792#
SCA	1047#			
SCALE	2779	4734#	4742	
SCALEF	4736	4745#	4750	4752

SCL	1249#							
SCOPE	2750#	4343	4757					
SCP	4319	4633	4646#	4671				
SDA	1137#							
SEARCH	4376	4769#	4739					
SETSC	4755#	4317						
SFLAG	2730	4722#	4726					
SHL	1045#	1652						
SIGN	1254#	2255						
SKAD	1094#	5529						
SKER	1095#							
SKPK	1091#	3316	3352	3357	3393	3397	3534	
SKXX	1092#	1660	3400					
SMIN	1251#	2299						
SNC	1117#							
SORTCN	1263#	2312						
SPL	1194#							
SRCH	4503	4773	4793#					
SRCH2	4776#	4734						
SPEXIT	4779	4733#						
SRF	1134#	1760	4467	4724				
STARTE	1572	1573	1730#					
START0	1447	1551#						
START1	1563#	1530						
START2	1569	1571#	1579					
START3	1576	1531#						
START4	1536#	1591						
START5	1533	1594#						
START6	1593	1598#						
START7	1664#	1693						
START8	1673#	1704						
START9	1755#	1312						
STAR10	1764#	1329						
STAR11	1774#	1343						
STAR6	1533	1592#						
STLP	4467#	4471	4475					
STORE	2751#	4260	4266	4272	4401	4405	4410	4411
	4620	4623					4415	4619
STVAR	2557	2559	2561	5792#	5793			
SUB1	4655	4661	4630#	4635	4636			
TABLE	2376	2377#	3160					

KTEMP	2745#	4262	4239	4557	4533	4625				
XZ	2739#	4291	4343	4407	4434	4485	4527	4583	4512	4513
	4553	4537	4627	4651	4696					
YLAST	2744#	4662	4702							
YTEMP	2746#	4264	4292	4559	4590	4601				
Y0	2740#	4294	4430	4433	4434	4455	4456	4457	4560	4539
	4632	4653	4700							
ZTEN	1296#	3297	3367	3531	3623					

V3

.Y