DECUS

PROGRAM LIBRARY

DECUS NO.	8-546
TITLE	DETEF - DECTAPE FILE-HANDLING SYSTEM
AUTHOR	Dr. Carl Reutersward
COMPANY	Research Institute of National Defense Sundbyberg, Sweden
DATE	April 7, 1972
SOURCE LANGUAGE	

THE DECTAPE FILE-HANDLING SYSTEM "DETEF"

Carl Reuterswärd (Apr. 7, 1972)

- List of Contents -

Summary

DETEF

This DECtape file-handling system provides the user with a keyboard monitor similar to the DISK/DECtape Monitor, though better suited to DECtape operation.

Based on a data-controlled linking loader, this monitor will address any tape unit and any memory field while occupying only the last memory page of field zero. The DECtape hardware is used in continuous mode reading and writing, the core-specifications being word-, not page-wise. Program files as well as data files are relocatable continuous block sequences, easy to create and to retrieve by user programming. Files are addressed by 5-character names and are classified as to type by means of a sixth letter. The system directory will accomodate 63 user files on each tape reel.
A special feature is a system of generating library files, or segmented programs.

Requirements: PDP8/I, 4k, EAE, and TC01 DECtape control.
Optional: Extended Memory, High-Speed Reader and Punch.

System application programs to the DETEF include:-

(1) A 4k Focal version having versatile program and data library capabilities: segments of program and data files may be saved and loaded by programmed instructions, thus virtually eliminating the limits on program size set by core dimension.

(2) An Editor allowing the merging of text from several sources (DECtape files and punched tape) and automatic return of edited text to one of the input files. The listings carry line numbers.

(3) A PAL Assembler with output to DECtape, control of extent of assembly listing, and optional extension of the symbol table by storage on DECtape.

THE DECTAPE FILE-HANDLING SYSTEM, DETEF

USER'S MANUAL

Author: Carl Reuterswärd*

Date: Apr. 14, 1972 (submitted to the DECUS Program Library)

I. System Specifications

1. Purpose

The DETEF System was designed as a general punched-tape and DECtape file monitor for a 4k PDP8/I computer. It permanently occupies only the last 128 locations of this memory field and thus performs the same functions as the DEC Disk/DECtape Monitor. However the design emphasizes a smooth, rapid operation of the DECtape device and supplies the user with a permanently available general DECtape handler. It does not use the instruction, ION.

By Teletype keyboard commands, the DETEF monitor enables the user to load programs by punched tape or similarly structured assembler output files into any memory field. The loaded programs may be saved as core-images in DECtape files and reloaded and started by a call command. DECtape blocks may be allocated for programs and data of several kinds and deleted for retrieving DECtape space. The files are addressed by the DECtape transport unit number, a one-letter type-designation, and a five-character name. The file-directory of each a tape reel accomodates 63 user-files.

The system DECtape handler operates the TCØ1 control in the continuous mode by counting words, not blocks. The transfer to the core memory may be controlled by data stored in the blocks, by which linking of separate block sequences may be achieved. The files generated by the monitor, however, constitute contiguous sequences of blocks, thus ensuring rapid call operations. They are completely relocatable on the tapes.

---

* Address: FOA 1, P.O.Box 416, S-172 04 Sundbyberg, Sweden.

2. General System Lay-out

(1) The minimum system configuration is:
Computer PDP8/I, 4k, with Teletype ASR33,
Extended arithmetic unit (EAE),
DECtape control TC01/08 and one transport unit TU55/56.

(2) Possible extensions are:
Extended memory control MC8/I and up to seven additional 4k units,
Seven additional transports TU55/56,
High speed punched tape reader and punch PT8/I.

(3) The monitor instructions are stored in block 0 and blocks 13 - 31 of the tape on the DECtape transport No. 0/8 (the System Unit). The blocks 1 - 12 are used for temporary storage of user program residing in core locations utilised by the monitor. The System Unit has thus to be operated in the "Write Enabled" mode.

(4) By simple modifications, however, the monitor may be instructed to utilise another transport for the temporary saving of user program. The System Unit may then be operated under "Write Lock", protecting the system monitor from destruction.

(5) Blocks 32 - 34 of each tape store the system directory of files on the tape.

(6) The system loader permanently occupies locations 7600 - 7777 of memory field zero. When the monitor is called into operation, the data of locations 5200 - 7577 are replaced by monitor instructions. During monitor operations, other instructions and the contents of file directories are loaded into this core area. When control is returned to a user program (or the command "EXIT" is given), the saved data are returned (if not intentionally changed by the monitor operation).

(7) The monitor preserves the data stored in the last page of any extended memory unit. The user may profitably store a binary loader in one such page, if available, before loading the DETEF system.

(8) For operation of the system, these programs are essential:

INDEX

PACK

FCOPY

BFILE

They are saved as C-files and their location on the tape is free.

(9) The blocks and locations allocated to the several system segments are presented in Appendix No. 1.

II. System Generation

1. When a System DECtape is available, a new System tape is generated by applying the command "SYSTEM". For further directions, refer to the chapter "Monitor operation".

The freshly generated system tape contains only the monitor routines and a directory. To be fully operable, the system must be complemented by the programs INDEX, PACK, FCOPY and BFILE, copied from the generating system (refer to paragraph FCOPY in the chapter "System Programs").

2. A System DECtape is generated by the DETEF Builder binary punched tape. The procedure is as follows:

(1) Ascertain that there is a Binary Loader in any core field (No. "f").
(2) Mount a correctly formatted DECtape reel (2702_8 blocks of 201_8 word each) on a transport, unit No. 8, with switches set to "Remote" and "Write Enabled".
(3) Put the program tape in the Teletype (L.S.) or the High Speed (H.S.) reader, with leader code (200) in reading position.
(4) Set switches to: I.F. = f, D.F. = 0, S.R. = 7777; press "Load Add" switch.
(5) If H.S. reader be used, set S.R. to 3777.
(6) Press "Start" switch.
(7) If L.S. reader be used, set this to "Start".
(8) Part of the tape is read in; when program halts, check that the AC register is cleared (if not, restart procedure from (1)).
(9) Set I.F. to 0 and S.R. to 200; press "Load Add" and then consol switch "Start".
(10) Program stops at HLT in location 201.
(11) If L.S. reader is used, set bit zero of S.R. to 1; press switch "Cont".
(12) The remainder of the tape is read in, and the system will be stored on the DECtape.

(13) Finally the DETEF Bootstrap Loader is automatically read into locations 7754 - 7766 and started; the monitor will be called from the newly generated system and show its presence by printing a carriage return (C.R.), a line feed (L.F.) and a period (".") at the left margin of the Teletype. It is now awaiting a command from the Teletype keyboard.

(14) Punched tape reading errors lead to JMP . at location 215 with AC cleared. Restart from (1).

(15) DECtape hardware errors during BUILDER operation lead to a HLT at location 127, or a JMP . at 126 with the B Status word in the AC. Check the DECtape switches and restart from (1).

(16) Further programs are added to the system from binary program tapes, one at a time, by applying the commands "LOAD" and "SAVE". Refer to next chapter.

III. Monitor Operation

1. Calling the Monitor

(1) When the monitor exits from core, either by starting a user program or merely by halting after a "EXIT" command, the system loader is left in the last page of core field zero. The monitor is then recalled by starting at 7777 (set I.F. = 0, D.F. = 0, S.R. = 7777; press "Load Add" and "Start").

(2) If the 1-page System Loader is not present, the start may be achieved by toggling in the 11 codes of the DETEF Bootstrap Loader in the core field zero as listed in Appendix 2. This routine is started at 7756.

(3) The DETEF System may be loaded and started by the Disk/DECtape monitor or the PS/8 Programming System by a routine also listed in Appendix 2.

2. Monitor Commands*

(1) When the monitor has printed "." at the left margin, it is ready to receive a command.

(2) ALLOCATE

This command reserves a sequence of blocks for a file. The format of the command is this (↵ represents a carriage return):

.ALLOCATE t u name!n;a↵ (for C- and L-files)

.ALLOCATE t u name!n↵ (for other file-types)

u = DECtape transport unit No. (0 to 7; 0 stands for 8).

t = file-type (one alphabetic letter)

name: 1 to 5 alphanumeric characters

n = octal number of blocks

a = starting address

By this command the following entries are written into the system directory of the specified unit: name, type, 1:st block no., no. of blocks, starting address (when applicable).
When this has been effected, the monitor reports:

DONE

.

(3) DELETE

When a file is to be scratched from a system directory, the format of the command is:

.DELETE t u name↵

The monitor eliminates the allocation entries pertaining to the file specified, and then reports:

DONE

.

The blocks thus set free may be utilized for further allocations. No other file-blocks are changed. If gaps between existing files are to be eliminated, use the program PACK.

---

* In the following examples of TTY operations, underlined characters are printed under program control.

(4) LOAD

Binary program tapes and files of similar structure (B-files) are loaded by means of the following dialogue with the monitor:

```
.LOAD                    /terminate by a space, not "↓"
H:R,T,Du:name,....↓      /specify 0 to 8 inputs
L:....↓                  /same
*a↓
```

a = starting address in field zero

R stands for the H.S. reader

T stands for the Teletype reader

D stands for DECtape (a B-file of name specified)

u = unit No.

H asks for inputs to high addresses (5200 - 7577) in field zero

L asks same for low addresses (0 - 5177)

Addresses in extended memory fields are treated as low.

Then the inputs are read into core in the order specified. The reading of each punched tape is initialised by the print-out of the character "↑"; the user puts the tape into the reader and answers by typing a CTRL/P character. HLTs (7402) will be deposited in all locations of field zero which are not used.

When loading is complete, a "↑" appears, again to be answered by a CTRL/P. The monitor then exits and transfers control to the address given ("a") in field zero. If a = Ø, the monitor remains and enters Command Mode.

(5) SAVE

A program residing in the core may be saved on a DECtape unit by this command format:

```
.SAVE u f name!h-k,l-m,....;a↓
DONE
.
```

u = DECtape transport unit no. (0 to 7)

f = core field no.

h-k, etc., are core space specifications given in ascending order

a = starting address in core field zero

A maximum number of 6 core specifications may be given, all in one core field.

The SAVE routine operates in two passes: first are those instructions saved, which belong to the "low" part (0 - 5177) of core field zero and those in extended core fields, secondly come those in the "high" part (5200 - 7577).

The files generated are C-files and automatically allocated by the SAVE command. They are core images and will be loaded by the "CALL" command, which also automatically start execution at location a.

(6) CALL

The user types:

.CALL u f name

u = DECtape unit no. (0 to 7)

f = core field no.

The monitor reads the contents of blocks 1 to 12 of the system device into the field zero locations 5200 to 7577, by which the monitor ceases to exist in core. The loader in 7600 - 7777 then reads the called program into the field specified in the call and finally transmits control to the field zero address given at save-time.

If the user intends to immediately continue monitor operation, the starting address should be given as 7777. The monitor will at once be reloaded, and several call-commands may be given, e.g. for loading into several core fields.

Stating 7745 as the starting address will leave the system at a HLT in this location.

(7) LIBRARY

Large programs and program systems may be segmented and conveniently created and called by this command. The files addressed in this way must be allocated as L-files. A L-file may be described as a collection of C-files. Of these, the only one to be started by the monitor is that one beginning at the first block of the L-file (relative block no. zero). The other parts of the program are called through program operation in any way the programmer chooses.

(a) A L-file is created in the following way.

The program segments which are to be the components of the L-file shall exist saved as C-files. The allocation of blocks to the L-file is done with a dummy starting address, e.g. 7777. Then:

```
.LIBRARY u name)        /name of the L-file
#ADD u:name!b)          /name of a C-file
#                       /etc.
```

u = DECtape transport unit no. (0-7)

b = relative 1st-block no. within the L-file

The monitor copies each C-file into the specified place (b) within the L-file. The relative 1st-block numbers must be chosen with due regard to the size of each C-file so that these do not overlap.

The C-file specified with b = 0 will have its starting address imposed as the entry point of the L-file. The system will no longer recognize the interior programs as C-files.

(b) A L-file is called by unit no. and name as above, and then:

```
#GO)
```

The code u100 is deposited at location 7775 and the starting block-number of the L-file at 7776 of field zero. Then the first program in the file is loaded and started.

(8) SYSTEM

This command represents the easiest way of generating a new system DECtape.

Mount a correctly formatted tape on a transport with any number from

1 to 7, set switches to "Remote", "Write Enabled", and type:

.SYSTEM ONTO UNIT:u)
TITLE:name)
↑ (check the switches and type a CTRL/P)
DONE
.

The title should be given by 1 - 5 alphanumeric characters. It will appear in the directory print-out obtained from the program INDEX.

(9) EXIT

This command (terminated by a space!) effects the loading of the contents of blocks 1 to 12, thereby restoring the user program saved at the previous loading of the monitor (if not destroyed by operation of the LOAD command) and finishing with HLT in 7744.

The system Loader remains in the last page of field zero. This loader is not usable for loading punched tapes. The common PDP8 RIM and Binary Loaders may be loaded only by calling a special program, which loads the instructions into the last page after monitor exit.

(10) General rules for typing monitor commands:

(a) Command words may be limited to two characters; always terminate with a space.

(b) Typing errors are amended by typing a Line Feed and restart from begin.

(c) u = 0 or f = 0 need not be typed; if f ≠ 0, however, also a u = 0 must be typed. Thus:

CALL INDEX instead of CALL Ø Ø INDEX;
CALL 7 PALD instead of CALL 7 Ø PALD;
but not CALL 1 SABR1 for CALL Ø 1 SABR1.

(11) General remarks on handling of the linking loader.

Since the linked loading is governed by the control words in the first block of each file segment, files can cause trouble when not properly formatted by a saving or initializing procedure, which should be performed right after allocating.

IV. Programming with DETEF

1. Utilizing the System DECtape handler

The System Loader in the last page of field zero comprises a subroutine for effecting DECtape transfers which can be utilised by the user program. The calling sequence in PAL-D notation is:

```
   :
   JMS I (7626
      PAR
   JMP DTERR /Error exit; Status B in the AC
   :
PAR, ulfc    /u = unit no.; f = field no.; c = 0 for read, = 1 for
              write
     block   /1st-block no.
     wc      /- (no. of words)
     ca      /(1st address)-1
     link
   :
```

The parameter l (bit 6 of parameter word ulfc) controls the automatic linking facility:

l = 0 gives normal transfer with 201_8 word per block:
read and write starts at begin (word Ø) of 1st block;
wc determines number of words in the transfer:
ca determines begin of core segment involved
(the parameter link is not used, buffer location not needed)

l = 1 gives linked loading. When writing, the words wc, ca, and link, will be saved as the 3 first word of transfer (not counted in wc), the others following as in the case of l = 0. The link gives the distance in block numbers to the next segment of the file.
When reading a segmented file with l = 1, only ulfc and block need be specified. The loader first reads the three words wc, ca, and link, into their locations, and then uses wc and ca as parameters for the transfer. The user program can utilize the link for coding the transfer of next segment.

The error exit will normally be achieved when a hardware error is encountered. However, a write-lock condition when using the DECtape handler for writing in the linking mode will lead to a closed loop at 7715-7720 with 7775 in the AC and the transport halted.

2. DECtape Errors

(1) When using the Bootstrap Loader, a tight loop at 7761 - 7762 may occur. The loading of the System Loader was unsuccessful. Correct erroneous switches on the DECtape transport and restart the Bootstrap at 7756.

(2) A HLT in the System Loader location 7745 indicates an error before loading of the monitor; Status B is in the AC. Correct DECtape switches and restart at 7777.

(3) When the Monitor is in operation, a DECtape error will lead to the diagnose: '?$' and a tight loop at 7412; Status B is in the AC. Strike the Space key for continuation.

---

3. B-files

(1) The structure of these files is similar to those of the binary program tapes. They represent a means of saving on DECtape the data of such tapes having a complicated loading pattern, since the SAVE command can deal with maximally only 6 origin settings. They are generated by the system program BFILE, or could be the output of a suitably modified PAL assembler.

The DETEF B-file conforms to the pattern of binary files of the Disk/DECtape Monitor System by packing three 8-bit codes in two 12-bit words. Thus three program instructions occupy four data words in the B-file. Except origins, field designations, checksum, and program data, the B-file comprises start and end codes corresponding to the leader/trailer codes of the binary tape, and, additionally, the control words for operating the linking loader.

The LOAD command expects the file segments to be loaded within the core section, 6200 - 6777.

4. Ending user programs

If the user programs ends up with a JMP I (7777, the monitor will automatically be started. Alternatively, the user program may end with a JMP I (7745, resulting in a HLT, from which the monitor can be called by pressing the switch "Cont".

5. Write-Lock on the system unit

In order to eliminate the need for operating the system device in the "Write Enabled" mode the following patches have to be made in the system tape:

Block	0, loc.	14:	change 1 to 1001
16	11	0	1000
16	15	0	1000
20	165	0	1000
24	122	0	1000

Now the user core saving routine will utilize unit no. 1, which must always be present and operated in the "Write Enabled" mode. The unit no. 0/8 can be "Write Locked".

6. Reading the Directory

The directory located in blocks 32 - 34 of each system tape can be read and searched by DISUB, three subroutines listed in Appendix 3. They occupy 145_8 words in one memory page and may be assembled together with a user program after specifying the three constants:

CODSTA = 1st address of DISUB

BUFSTA = 1st address of a 3-page buffer

FILTYP = the ASCII code of the file-type character.

The App. 3 gives further details on the structure of the directory and a sample program for utilising the DISUB-routines.

V. System and Ancillary Programs

A list of core allocations of these programs is given in App. 4.

1. INDEX

This program will produce a directory of files allocated on a DETEF system tape. User intervention is called twice:

(1) to give the DECtape unit number and add a string of commenting text (such as the calendar date, etc.).

(2) to expand the listing to include numeric file parameters, such as first block number (FBLK), the number of blocks (BLKS), and the program entry point (in the case of 'C' and 'L' files).

Refer to Appendix 5 for a sample listing.

2. PACK

The DELETE command leaves unused gaps in the block sequence of a tape. These are eliminated by the PACK routine. The operation is straight-forward as in this example:

```
.CALL PACK↵
PACK UNIT 2 ON TO UNIT 5↵
↑                          /check DECtape unit switches, then hit CTRL/P
DONE
PACK UNIT                  /CTRL/C for return to monitor
```

If DECtape error occurs, the program will type out '?$XXXX', where XXXX represents the B Status word. If the error occurred during a file transfer operation, there will, in addition, appear four octal words, giving the parameters of the IODT call: ulfc, block, wc, ca (cf., paragraph IV.1).

3. FCOPY

Copies the contents of one file into another file, which must have been allocated as the same type and at least the same number of blocks as the original. This is a sample dialogue:

```
.CALL FCOPY↵
TYPE-C                     /File-type
OUT-D0:CFIL↵               /Receiving file
IN-D7:XFIL↵                /Original file
↑                          /Check DECtape unit switches, then hit CTRL/P
TYPE-
```

DECtape errors will be diagnosed as in the program PACK.

4. BFILE

This program converts the data of a saved program (a core image) to the format which is required by the LOAD command for use in loading simultaneously with punched tape binary program data as needed for complex overlay operations. The contents of three program locations will be packed into four B-file data words.
The B-file is structured into 3-block segments, having a capacity of 440 program instructions (including origin settings). The start and end of program data will be marked by one leader/trailer code (200) in each case. No field-setting is given.
Before calling the program BFILE, the user must have allocated a B-file of sufficient capacity (one-third larger than the C-file in consideration).
In executing BFILE, the user answers the questions "OUT-D" and "IN-D" by typing a DECtape unit number, a colon, a file-name, and a carriage return. The program will then start operating. When finished, it will type out "$" and restart the initializing dialogue.
The diagnosis, "?FE", will indicate that the B-file is too small. A DECtape error will be signalled by "?$" and the B Status word.

5. OEDT

This Octal Editor of DECtape loads one block into the core memory for the purpose of inspection and changing single words or listing in entirety. Then the updated data may be saved in the same block. The dialogue runs thus:

U: The user types the DECtape unit number, and a "⤸"

B: The user types the octal block number, and a "⤸"

L: A "⤸" given here will result in a 8-column listing of the 129 words in the block

L: The user may inspect and update the contents of the buffer in the ODT manner by typing a "/" and an octal location number (Ø - 2ØØ), reading the contents of the location and changing these by typing a 12-bit word. Terminating the line with a line-feed will open the following location for inspection; a "⤸" will put an end to this procedure.

L: By typing "S", the user will save the updated data into the old block and restart the procedure from "U:". (The command S will not be accepted if the unit number stated is Ø or 8.) An "U" or a "B" will end the operations on the current block without saving the buffer contents.

U: The program is interrupted by typing CTRL/C.

6. TDUPL

This program utilizes the System Loader subroutines in the last page of field zero for duplicating an entire DECtape reel. The commands are given as in this example:

```
.CALL TDUPL↵
I:Ø↵                    /The user types the DECtape unit number
                        /of the original
O:7↵                    /The copy appears on the unit specified
                        /here
↑                       /Check the commands and switches; then
                        /hit CTRL/P
$                       /Duplicating is done
I:                      /Go next duplicating operation or hit
                        /CTRL/C for returning to the system mon-
                        /itor
```

DECtape error diagnoses are like those of the program PACK.

7. PTCOR

This program may assist in giving a correct command for saving programs loaded as binary tapes. It will read such a tape and type out a core-list, showing into which core locations the program is loaded.

When first started, the program will ask:

R/T?

The user indicates which reader is to be used.

The program then types:

↑

and the user responds with CTRL/P after loading the reader with the tape (leader code in reading position). The pro-

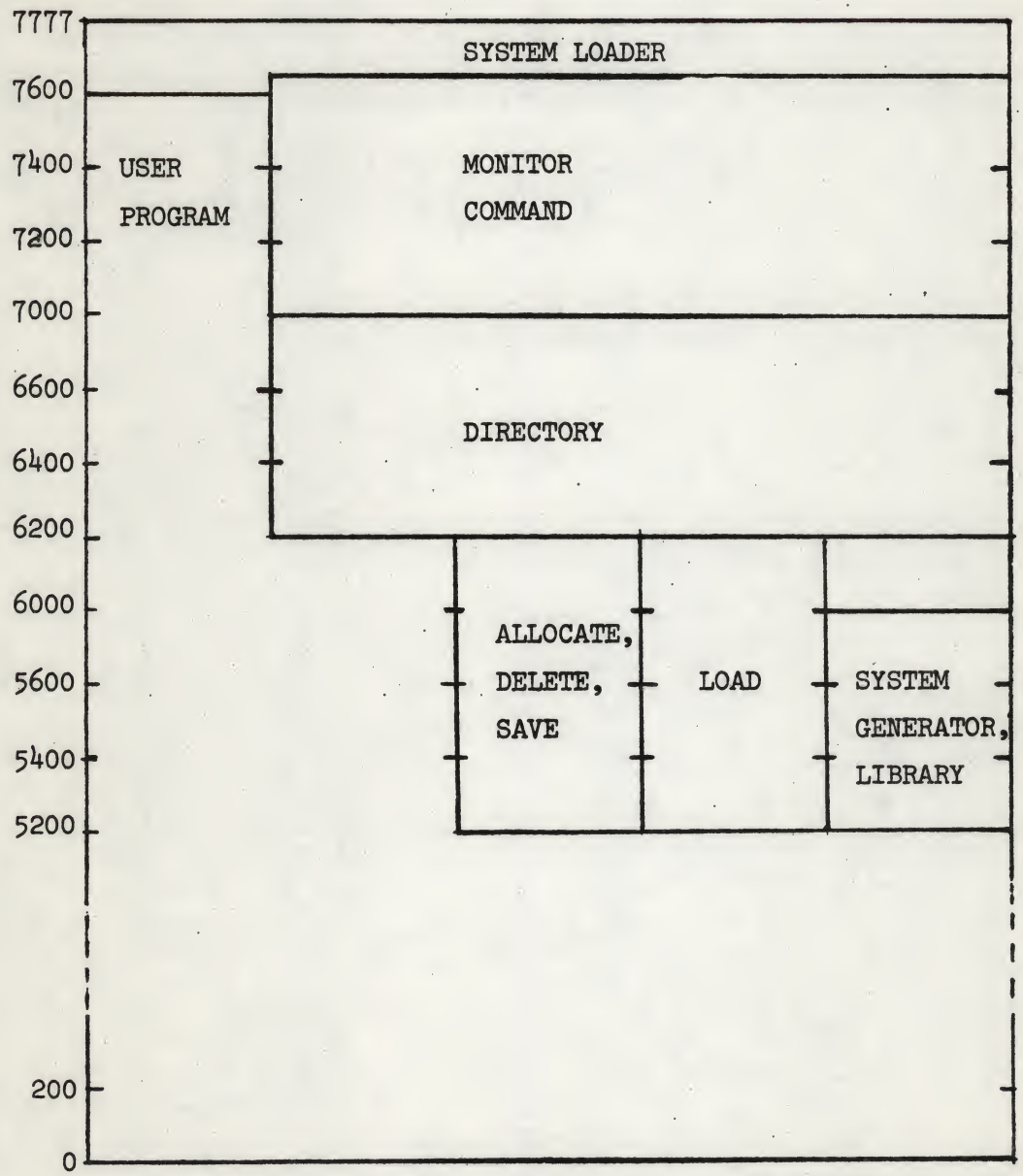
gram starts reading the tape and types origins as "*nnnn", field designations as "#n", and core-specifications as "nnnn - nnnn", five in each line.
The operation ends by the output "↑": the user may start another run, or finish by hitting CTRL/C, which calls the system monitor.

8. CFPT

This "C-File to Punched Tape" copying program converts a C-file to a binary program tape. When called, CFPT will ask:
R/T?
The user states whether the high-speed punch (R) or the teleprinter (T) should be used. Then the question:
IN-D
is answered by typing the DECtape unit no., a colon, the name of a C-file and a carriage return. The program types:
↑
and the user puts the punch "on", and hits CTRL/P.
The program will then start punching, ending by again typing:
↑
The user puts the punch "off" and hits CTRL/P. The program should then restart by asking "R/T?". At this point the diagnose:
XXXX BLOCKS
will indicate, that the copied program did not use all the blocks of the file.
Other messages output by CFPT are the following:
?F = Format error: an illegal file-name was typed
?N = Name error: no such file in the directory
?D = Data error: the program in the file cannot be properly read
?$XXXX = DECtape error: XXXX = B Status word.

Appendix 1:1

Core allocations of Monitor instructions



BLOCK ALLOCATIONS OF DETEF SYSTEM

0	SYSTEM LOADER
1	
2	SAVE
3	BUFFER
4	FOR
5	USER
6	PROGRAM
7	LOCATIONS
10	5200-7577
11	
12	
13	
14	MONITOR COMMAND
15	
16	
17	
20	ALLOCATE
21	DELETE, SAVE
22	
23	
24	LOAD
25	
26	
27	
30	SYSTEM GENERATOR
31	LIBRARY
32	
33	DIRECTORY
34	
35-	RELOCATABLE SYSTEM & USER PROGRAMS, DATA FILES
-2701	

DETEF System Starters

```
                    DETEF SYSTEM
                    BOOTSTRAP LOADER

                    7754  0000
                    7755  7577
ENTRY POINT-----    7756  6774
                    7757  1365
                    7760  6766
                    7761  6771
                    7762  5361
                    7763  1366
                    7764  5360
                    7765  0600
                    7766  0220
```

```
                ///DETEF SYSTEM STARTER
                /(DISKSYS)
                *200
0200  7602      DTFBS,  CLA HLT  /CHANGE SYSTEM TAPE REEL
0201  6774        DTLB
0202  3633        DCA I WC
0203  1227        TAD MOVB
0204  4222        JMS DWAIT
0205  1230        TAD K7577
0206  3634        DCA I CA
0207  1220        TAD M200
0210  3633        DCA I WC
0211  1231        TAD REDF
0212  4222        JMS DWAIT
0213  1232        TAD K200
0214  6764        DTXA
0215  6772        DTRB
0216  7510        SPA
0217  7402        HLT
0220  7600      M200,  7600      /CLA
0221  5620        JMP I .-1
0222  0000      DWAIT,  00
0223  6766        DTCA DTXA
0224  6771        DTSF
0225  5224        JMP .-1
0226  5622        JMP I DWAIT
0227  0600      MOVB,  600
0230  7577      K7577,  7577
0231  0220      REDF,  220
0232  0200      K200,  200
0233  7754      WC,  7754
0234  7755      CA,  7755
                /
```

Listings of the subroutines DISUB

and a sample coding for utilising these for loading and searching the Directory.

```
0001            :
0002            CLA
0003            TAD UCODE
0004            AND (7000
0005            JMS I (DIRED      /LOAD THE DIRECTORY
0006            JMP DTERR         /DECTAPE ERROR; STATUS B IN AC
0007            JMS I (DIGET      /GET THE FILENAME AND CARR.RETURN
0008            JMP GERROR        /TOO MANY CHARACTERS; 0 IN AC
0009            JMS I (DISER      /SEARCH DIRECTORY
0010            JMP SERROR        /FILE NOT IN DIRECTORY; 0 IN AC
0011            DCA STBLA         /ADDRESS OF PARAMETER STBL
0012            TAD STBLA
0013            DCA PTR
0014            TAD I PTR
0015            DCA UCODE+1
0016            ISZ PTR
0017            TAD I PTR
0018            MQL MUY
0019              201
0020            CIA
0021            DCA UCODE+2
0022            JMS I (IODT       /IODT=7626
0023              UCODE
0024            JMP DTERR         /DECTAPE ERROR; STATUS B IN AC
0025            :         /DATA LOADED
0026            :
0027  STBLA,    00
0028  PTR,      00
0029  UCODE,    U0F0      /U=UNIT NO., F=FIELD NO.
0030            00        /STBL=NO. OF FIRST BLOCK
0031            00        /WC=MINUS COUNT OF WORDS IN DATA FILE
0032            DATBUF-1          /DATBUF=FIRST LOCATION OF DATA BUFFER
0033            :
0034  /DIRED=CODSTA, DISER=CODSTA+7, DIGET=CODSTA+52
0035  /FILTYP=304
```

```
                XLIST
                PAUSEC ODSTA=200
                BUFSTA=7000
                FILTYP=304
                /
                ///DISUB. ROUTINES FOR READING AND SEARCHING THE DIRECTORY
                /(DETEF)
                *C0DSTA
                /DIRED SUBROUTINE: READS THE DIRECTORY INTO THE BUFFER
                /CALLING SEQUENCE:
                /  TAD UNIT
                /  CLL RTR
                /  RTR
                /  JMS I (DIRED/
                /  (DECTAPE ERROR RETURN: AC=STATUS B
                /  (RETURN: AC=0)
0200  0000    DIRED.  00
0201  3603      DCA I .+2
0202  4746      JMS I DIODT
0203  0335        FUNC
0204  7410      SKP
0205  2200      ISZ DIRED
0206  5600      JMP I DIRED
                /DISER SUBROUTINE: SEARCHES THE DIRECTORY
                /CALLING SEQUENCE:
                /(FILENAME XXXXXT IN 3-WORD NAME-BUFFER "FILNAM")
                /  JMS I (DISER
                /  (ERROR RETURN: AC=0)
                /  (NORMAL RETURN: AC=ADDR.OF STBL)
0207  0000    DISER.  00
0210  1340      TAD DBUFF
0211  7001      IAC
0212  3330      DCA DPTR1
0213  1730      TAD I DPTR1
0214  7041      CIA
0215  3332      DCA DCNT1   /MINUS COUNT OF ENTRIES
0216  2330      ISZ DPTR1
0217  2330      ISZ DPTR1
0220  1341    NXTENT.  TAD NAMBUF       /ENTRY FORMAT:
0221  3331      DCA DPTR2                /  XX
0222  7346      CLL CLA CMA RTL          /  XX  (ASCII-CODES - 240)
0223  3333      DCA DCNT2                /  XT
0224  1730    NXTWRD.  TAD I DPTR1       /  STBL
0225  7041      CIA                      /  BLKS
0226  1731      TAD I DPTR2              /  E.P.
0227  7640      SZA CLA
0230  5240      JMP NOMATC
0231  2330      ISZ DPTR1
0232  2331      ISZ DPTR2
0233  2333      ISZ DCNT2
0234  5224      JMP NXTWRD
0235  1330      TAD DPTR1   /ENTRY FOUND: GET ADDR. OF STBL
0236  2207      ISZ DISER
0237  5607      JMP I DISER   /DONE
0240  2330    NOMATC.  ISZ DPTR1   /WRONG ENTRY
0241  2331      ISZ DPTR2
```

```
0242  2333    ISZ DCNT2
0243  5240    JMP .-3
0244  2330    ISZ DPTR1  /SKIP STBL
0245  2330    ISZ DPTR1  /SKIP BLKS
0246  2330    ISZ DPTR1  /SKIP E.P.
0247  2332    ISZ DCNT1  /ALL ENTRIES EXAMINED?
0250  5220    JMP NXTENT
0251  5607    JMP I DISER  /YES; ERROR RETURN
              /DIGET SUBROUTINE; READS THE FILENAME FROM TTY
              /CALLING SEQUENCE:
              /  JMS I (DIGET
              /  (ERROR RETURN; AC=0)
              /  (NORMAL RETURN; AC=0; NAME & FILETYPE IN 3-WORD BUFFER)
0252  0000  DIGET, 00
0253  1342    TAD M6
0254  3332    DCA DCNT1
0255  1341    TAD NAMBUF
0256  3330    DCA DPTR1
0257  3334    DCA HALF
0260  6031  INPUT,  KSF  /READ ONE CHARACTER
0261  5260    JMP .-1
0262  6036    KRB
0263  6041    TSF  /PRINT IT
0264  5263    JMP .-1
0265  6046    TLS
0266  1343    TAD MCR  /TEST IT
0267  7450    SNA  /CARRIAGE RETURN?
0270  5276    JMP PACKT
0271  1344    TAD CRMSP  /NO; PACK IT
0272  4307    JMS PACK
0273  2332    ISZ DCNT1
0274  5260    JMP INPUT
0275  5652    JMP I DIGET  /TOO MANY CHARS.; ERROR RETURN
0276  2332  PACKT,  ISZ DCNT1
0277  7410    SKP
0300  5303    JMP .+3
0301  4307    JMS PACK  /FILL OUT WITH ZEROS IF < 5 CHARS.
0302  5276    JMP PACKT
0303  1345    TAD FTYPE  /PACK THE FILETYPE
0304  4307    JMS PACK
0305  2252    ISZ DIGET
0306  5652    JMP I DIGET  /DONE
0307  0000  PACK,  00
0310  2334    ISZ HALF  /WHICH HALF?
0311  5316    JMP LEFT
0312  1730    TAD I DPTR1  /PACK IN RIGHT HALF
0313  3730    DCA I DPTR1
0314  2330    ISZ DPTR1
0315  5707    JMP I PACK
0316  7106  LEFT,  CLL RTL
0317  7006    RTL
0320  7006    RTL
0321  3730    DCA I DPTR1
0322  7040    CMA
0323  3334    DCA HALF
0324  5707    JMP I PACK
```

```
0325  0000  FILNAM,  00
0326  0000    00
0327  0000    00
0330  0000  DPTR1,  00
0331  0000  DPTR2,  00
0332  0000  DCNT1,  00
0333  0000  DCNT2,  00
0334  0000  HALF,  00
0335  0000  FUNC,  00
0336  0032    32
0337  7200    -600
0340  6777  DBUFF,  BUFSTA-1
0341  0325  NAMBUF,  FILNAM
0342  7772  M6,  -6
0343  7563  MCR,  -215
0344  7755  CRMSP,  215-240
0345  0044  FTYPE,  FILTYP-240  /ASCII-CODE - 240
0346  7626  DIODT,  7626

BUFSTA 7000
CODSTA 0200
CRMSP  0344
DBUFF  0340
DCNT1  0332
DCNT2  0333
DIGET  0252
DIODT  0346
DIRED  0200
DISER  0207
DPTR1  0330
DPTR2  0331
FILNAM 0325
FILTYP 0304
FTYPE  0345
FUNC   0335
HALF   0334
INPUT  0260
LEFT   0316
MCR    0343
M6     0342
NAMBUF 0341
NOMATC 0240
NXTENT 0220
NXTWRD 0224
PACK   0307
PACKT  0276
```

Core Allocations of DETEF Ancillary Programs

Name	Save locations	Entry Paint
INDEX	7000 - 7414	7000
PACK	7000 - 7435	7000
BFILE	20 - 1031	200
FCOPY	6720 - 7565	7000
CFPT	20 - 752	200
TDUPL	200 - 464	200
OEDT	6600 - 7377	6600
PTCOR	200 - 564	200

Sample listing of the Directory

(Underlined characters were typed by the user)

```
•CALL INDEX↓
FILES ON UNIT 3↓

0014 FILES. 0210 BLOCKS

NAME TYPE FBLK  BLKS  E.P.

DETEF   *  0000  0035
INDEX   C  0035  0003  7000
PACK    C  0040  0003  7000
BFILE   C  0043  0005  0200
FCOPY   C  0050  0004  7000
CFPT    C  0054  0004  0200
UEDT    C  0060  0003  6600
PTCOR   C  0063  0002  0200
TDUPL   C  0065  0002  0200
FOCAL   L  0067  0040  0200
PALD    L  0127  0041  5000
EDIT    C  0170  0020  3000

FILES ON UNIT
•
```

APPLICATIONS OF THE "DETEF" DECTAPE FILE HANDLING SYSTEM

I. DETEF-FOCAL

Carl Reuterswärd[1)]

Date Apr. 14, 1972

The DETEF-FOCAL is equivalent to the FOCAL version of the TIROS Timesharing and Real-Time Operating system for the PDP8/I[2)]. It constitutes a modification of FOCAL W (DEC 08-AJAC-D) specifically intended for handling large amounts of experimental data, such as are obtained in spectroscopy. It utilizes program library files and data files on DECtape; automatic concatenation of program segments can be used to effectively remove the limits on program size inherent in 4k FOCAL operation.

DETEF-FOCAL FEATURES

1. Starting Dialogue

.LI FOCAL↓
#GO
*XFUN-x↓
* (The user types "N", "S" or "A" for "No", "Some" or "All" Extended Functions)
*BUFS-xxx+yyy↓ (or *BUFS-xxxx↓, or *BUFS-↓)
* (The user states size (octal) of "Data" + "Reference" buffers
* (or "Data" buffer only); type ↓ only if no buffers is wanted)
*LIB-u:name↓
(The user indicates an F-file)
*DATA-u:name,u:name↓ (or *DATA-u:name↓)
* (The user indicates one or two (identical or different) D-files
* as store of data)
* (Focal is now in command mode. The user may start calculating, programming, or calling a program from the library.)

In this dialogue, typing a rubout will cause the question to be repeated.

---

1) Address: FOA 1, P.O.Box 416, S-172 04 Sundbyberg, Sweden

2) Presented at DECUS 7th European Seminar, Amsterdam, Sept. 1971.

2. The Data Buffer

This buffer is a core area where data may be stored safe from being destroyed by an Erase command. This area is also directly accessible for transfers between the core and the D-files specified in the initialisation dialogue.
The instruction: SET X=FBUF(I), sets the variable X equal to the 12-bit integer stored at the buffer location I (I=Ø,1,...). The octal word 4ØØØ is rendered as Ø.

The instruction: SET D=FBUF(I,X), places X as a 12-bit integer at the buffer location I: 2047≥X≥-2047. (X-values outside this range are rendered modulo (4096)); in this process the dummy variable D acquires the integer value X.
If the buffer size has been given as x+y, then x pertains to a specific "Data" section and y to a "Reference" section. The numbering of buffer locations is common to both, the "Data" section starting at location Ø, the "Reference" at location x.

Double-precision floating-point numbers may be stored in the buffer area by means of the special variable DB(I). Each number is stored in three consecutive locations: 3I, 3I+1, 3I+2.
Since ordinarily each variable needs 5 core locations, the data buffer feature offers a certain economy of storage.

3. The Library Command

When a library file has been specified in the initialization, the contents of the Focal program buffer can be saved by the command:
LIBRARY SAVE PROGRAM x (or L S P x)
where x is an integer (x=Ø,1,...)
The command considers x to point to the number of an 8-block segment of the library file. These segments are numbered: Ø,1,....
Reversely, the command:
LIBRARY CALL PROGRAM x (or L C P x)
loads the program stored in segment no. x into the program buffer, destroying any program residing there. The variables are destroyed only if necessary because of lack of space; the variables first defined are those first deleted. The buffer area is not effected.

In this connection, the user should be made aware of the importance of filling newly allocated program files with programs (or cleared text buffers) in order to avoid the troubles which accidentally may arise through calling a program which has not been saved.

4. Data File Operations

The Library command also applies to the "Data" and "Reference" buffers, and transfers are effected to D-files specified in the initialization (the first one is associated with the "Data" buffer, the second one to the "Reference" buffer):

LIBRARY CALL DATA x (L C D x)
LIBRARY CALL REFERENCE x (L C R x)

and similarly for saving (L S D x, L S R x)

The integer x (=Ø,1,...) points to the number of a segment of the file: if n is the number of buffer locations, then each segment comprises [n/129] + 1 blocks ([x] meaning the integer part of a number x).

6. Programming library commands

Transfers may be initiated by programmed commands. Thus the programmed instruction:

L C P 1;GOTO 4.3

will call a program from file segment no. 1 and start execution at line no. 4.3. In general, one instruction following the library command on the same line will be saved for execution after loading the new program. The application of this feature is illustrated in Appendix 1.

5. Buffer size considerations

The maximum buffer size is 2155_8 locations. This is reduced stepwise by the various options as follows:

extended functions "S": -400_8 "A": -600_8

program or data library: -173_8

7. Display function

This function operates a storage display tube via the AXØ8 computer/laboratory interface. The command formats are the following two:

FDIS(x,y) - illuminates a point
FDIS(x,y,z) - z=E for Erase, S for Store, N for Non-Store.

The program should allow a delay of 0.1 sec for execution of the E command before ordering a point display, as in the following which stores a rectangular frame display:

```
1.1 S D=FDIS(,,E); S D=FDIS(,,S); F I=Ø,9; S X=Ø
1.2 F Y=-255,510,255; F X=Ø,4,5Ø8; S D=FDIS(X,Y)
1.3 F X=Ø,511,511; F Y=-252,4,252; S D=FDIS(X,Y)
```

The coding of the relevant AXØ8 instructions is listed in Appendix 2.

8. Sundry other features

(1) The DETEF-FOCAL comprises an improved FRAN, which produces pseudo-random numbers in the range (-1, +1) with the period 4,194,304. The initial argument of FRAN( ) is 20485. It can be changed by means of the function FSET:

```
SET D=FSET(x)
```

will set the initial argument of FRAN( ) to x; x should be an odd integer in the range $\pm 2^{23}$, and D is a dummy variable.

(2) The FADC has been eliminated.

(3) The symbol of decimal exponentiation has been changed from "E" to "&". This leaves the character E free for use as a numeral, e.g.:

```
*TYPE ØDEF↲
456.ØØØ*
```

(cf., the demo program of App. 1).

(4) The command Alt. Mode has been replaced by CTRL/P. Typing CTRL/C as an answer will call the System Monitor.

(5) The High Speed Reader input is switched on and off by the command "↑".

(6) DETEF-FOCAL does not utilize the program interrupt system; the Low Speed Reader will never run out of synchronism with the program.

(7) For a list of error messages, refer to Appendix 3.

9 Generation of DETEF-FOCAL

DETEF-FOCAL is generated by an overlay to DEC FOCALW after initialization. The procedure is the following:

(1) Load the FOCALW by paper tape (two binaries in one tape)
(2) Save 46ØØ-4771 as FOC1; S.A. = 7777
(3) Load FOCALW again
(4) Set 74Ø2 at location 45ØØ using the Consol Switches
(5) Start execution at 2ØØ; the program will halt at 45ØØ
(6) Save 1-7577 as FOCØ; S.A. = 7777
(7) Allocate a B-file "FOCØ" of 6Ø blocks
(8) Use the system program "BFILE" for converting the C-file FOCØ into the B-file FOCØ
(9) Load the B-file FOCØ and the DETEF-FOCAL Overlay (4 binaries in one tape)
(10) Save 1-7577 as FOCØ; S.A. = 2ØØ
(11) Allocate L-file FOCAL, 4Ø blocks, S.A. = 7777
(12) Use the Library command to create this file:

```
#ADD u:FOCØ!Ø)
#ADD u:FOC1!37)
```

(13) L-file FOCAL is ready for use.

10 Core utilization of DETEF-FOCAL

The following locations of memory field Ø are left free to the user of DETEF-FOCAL:
0-3, 171-175, 2372-2400, 2754-2762, 6371-6376, 7175-7177, and 7377.

Application of Library calls

The following demonstration program will call a library segment as a subroutine in order to evaluate the answer given by the user to a question put by the program. It will use segment nos. Ø and 1 of a library file specified in the FOCAL initialization dialogue. The program will then be generated, saved and called into operation by typing this:

```
*E A
*1.1 ASK "TWO PLUS THREE MAKE"X;SET Y=Ø
*1.2 L C P 1;GO
*1.3 TYPE "ERROR"!;QUIT
*1.4 TYPE "CORRECT"!;QUIT
*L S P Ø
*E A
*1.1 L C P Y;IF (X-ØFIVE)1.3,1.4,1.3
*L S P 1
*L C P Ø;GO
```

Implementation of FDIS for AXØ8 as interface to a storage display

Function:	Non-Store	Store	Erase
Relay outputs:	R1:= Ø	R1:= Ø	R1= -3V
	R2:= Ø	R2:= -3V	
Program coding:	2713 TAD C7757	2722 TAD C2Ø	2743 TAD C1Ø
	2714 6342	2723 6344	2744 6344
	2715 CLA	2715 CLA	2752 C1Ø, 1Ø
	2716 TAD C7767	2716 TAD C7767	
	2717 6342	2717 6342	
	2734 C7757, 7757	2733 C2Ø, 2Ø	
	2735 C7767, 7767		

8. Summary of error messages

The error message format is ?nn.nn @ mm.mm; where nn.nn is the error code and mm.mm is the line number.

?00.00 Restart via CTRL/C and REFOC
?01.00 Restart via CTRL/P
?01.35 Group zero is an illegal line number
?01.43 Illegal step or line number used
?01.89 GOTO was not used as one word
?01.;2 Line number is too large
?01.;3 Double periods found in a line number
?02.48 Nonexistant line number referenced by DO
?02.63 Nonexistant group referenced by DO
?02.81 Storage was filled by push-down list
?03.09 Nonexistant line number used or a tight loop
?03.31 Illegal command used
?04.07 No space after IF or illegal format
?04.35 Left of = in error in FOR or SET
?04.48 Excess right paranthesis encountered
?04.56 Illegal terminator in FOR command
?05.63 Bad argument to MODIFY
?06.13 Illegal use of function or number
?06.64 Storage filled by variables
?07.14 Operator missing in an expression or double E
?07.34 No operator used before parenthesis
?07.<0 Double operators used
?07.;1 No argument given after function call
?07.;8 Illegal function name given
?08.50 Parenthesis do not match
?09.16 Bad argument in ERASE
?09.50 Maximum group number exceeded
?11.05 Illegal argument in FBUF
?12.83 Storage was filled by program
?13.09 Illegal LIBRARY command
?13.39 Illegal program or buffer number
?13.56 Illegal LIBRARY command
?13.67 Illegal terminator after LIBRARY command
?13.94 DEC-tape error
?20.41 Logarithm of zero requested
?23.35 Literal number is too large
?23.;8 DB subscript out of range
?26.91 Negative exponent used
?26.96 Exponent is too large
?26.<5 DB subscript out of range
?28.58 Division by zero requested
?30.48 Imaginary square roots required

APPLICATIONS OF THE "DETEF" DECTAPE FILE HANDLING SYSTEM

II. DETEF-EDIT

Carl Reuterswärd[1)]

Date Apr. 14, 1972

The DETEF System Editor is a modification of Disk Editor (DEC-D8-ESAB-PB 4/24/68). The changes enable the Editor to utilize files in the DETEF File Handling system[2)] for storage of edited text and, further, add some operational improvements.

1. The Editor Input/Output Files

These are DETEF System files of type 'A' and consist of contiguous sequences of blocks, grouped into segments of max. 3 blocks each. File space must be reserved in advance by means of the ALLOCATE commond. The text is packed with two 6-bit characters to each file location, and the number of blocks required for each editor page is

$$[(c + 2t + 41)/256_{10}] + 1$$

where: c = the number of characters,
t = the number of tabs,
l = the number of lines in the page,
and [x] denotes the integer part of x. For a program with a modest amount of comments this expression will not exceed [1/10] + 1.

2. Initialization of operation

The Editor is called from the DETEF System by the CALL command:

```
.CALL EDIT)
*OUT-
```

[1)] Address: FOA 1, P.O.Box 416, S-172 04 Sundbyberg, Sweden

[2)] Presented at DECUS 7th European Seminar, Amsterdam, Sept. 1971.

The user responds with ↵ for no output, T↵ for output to the Teletype, R↵ for output to the high-speed reader, and Du:xxxx ↵ for output to an allocated A-file of name xxxx on the DECtape unit no. u. Typing errors may be amended after hitting the Rubout key.

The Editor acknowledges a correct answer by typing a '*' and asking for input. The user now specifies from zero up to seven inputs - Teletype reader (T), High-Speed Reader (R), or DECtape files - in the order in which they will be used, e.g.:

*IN-R,D1:ASC1,D1:ASC2,R↵

The Editor acknowledges each input with a '*' and finally asks:

*OPT-

The user answers (n denoting a digit):

B (or nB) meaning: presence blanks, or

T (or nT) meaning: replace multiple blanks with tab/rubout combinations

Instead of 'T', ↵ or any other letter may be used.

The number n points to one of the inputs specified: n=1 to the first, etc. The Editor will finally return the output to this input, if these be DECtape files, using the specified output merely as a buffer.

If n=0, no such returning will take place.

The Editor types a '*' indicating command mode.

3. Editor commands (cf., DEC manuals D8-ASAA-D and 08-ESAB-D)

R command: Two or more Editor pages may be concatenated by using multiple R commands. If the Editor buffer (about 4100_8 locations) then becomes full, the bell will ring and the reading is stopped before all of the last page is read. If DEC-tape is used for both input and output the next R command will not read the rest of the page!

L and S commands: When one of these commands is used, the line number is printed before each line.

X command: This is used to switch the input to the following one as specified in the initialization dialogue; e.g.:

```
*R)          /One page is loaded from input no. 1
*X)          /Switch
*N)          /The loaded page is saved in the output, and
             /the first page of input no. 2 is loaded.
```

G command: This has been deleted.

E command: If input is from the teletype and output is to DECtape, a CNTRL/FORM must be typed following the E command. If a return to an input file has been specified, the Editor will first save the last pages in the output file, then type a '#', and finally transfer the contents of the output to the designated return file.

CTRL/P: This command during output to the teletype causes the Editor to stop typing and return to the command mode.

CTRL/C: This command will cease further Editor operation and call the DETEF System Monitor.

4. Error messages

FE (file exceeded): This message will be written when output is to DECtape and the number of blocks allocated to the file is too small. After writing "FE" the Editor will transfer control to the DETEF System Monitor.

If the output is to the file designated as the output file in the initialization dialogue, this file will be usable (as input) despite the error, but the last part or all of the last page will be lost. If the output is to an input file specified as the return file, i.e. if the message appears after the character '#', then this file is not properly closed and may not be used as input. However, the edited data are available in the buffer file (designated as output file in the dialogue).

5. Loading and saving the DETEF Editor

The Editor is generated in file by loading the binary tape and then giving the command:

```
.SAVE EDIT!1-3656;3000)
```

APPLICATIONS OF THE "DETEF" DECTAPE FILE HANDLING SYSTEM

III. DETEF-PALD

Carl Reuterswärd[1)]

Date Apr. 14, 1972

The DETEF System Assembler is a modification of the Disk Assembler (DEC-D8-ASAA-LA 4/25/68) enabling it to utilize files in the DETEF file handling system[2)] for source input and binary output, and in addition to improve the operational qualities in some respects (cf. DEC manual D8-ASAA-D).

1. Input files

There are DETEF files of type A (ASCII-files) which are produced by the DETEF-Editor. They are loaded segment-wise into core by means of the DETEF Linking Loader; the buffer comprises locations 7000-7577. The storage format is one which packs two 6-bit stripped ASCII-codes into each location.
Punched paper tapes read through the Teletype Reader or the High Speed Reader are also accepted as inputs. A maximum of 10 sources are allowed for input into one assembly operation.

2. Output files

The output of the assembler is paper tape punched in the Teletype Punch or the High-Speed Punch, or DETEF files of type B (Binary files). These are made up of concatenated segments of max. 2 block each, which will be loaded into locations 6400 - 6777 by means of the Linking Loader. The binary data are stored packed, three 8-bit codes into two file locations.

3. The symbol table

App. 1 gives a list of the pseudo-instructions and permanent PAL instructions of DETEF-PALD, 96 symbols in all.

---

1)Address: FOA 1, P.O.Box 416, S-17204 Sundbyberg, Sweden

2)Presented at DECUS 7th European Seminar, Amsterdam, Sept. 1971.

In operation, the user may choose between a smaller symbol table with a more speedy assembly, or an extended table with a slower assembly.
The smaller symbol table has room for 223 user-defined symbols with assembly output to DECtape, or 255 with no output, or punched tape output.
The extended symbol table holding a maximum of 409 user-symbols, is partly saved onto DECtape; during assembly these parts are swapped to and from the core memory, thus slowing down the process of assembly. However a maximum of 248 symbols may be assembled without any swapping.
The saved parts of the table are stored in blocks $1\text{-}12_8$ of DECtape unit no. Ø (or any other unit as defined by patching one instruction of the assembler).

3. Assembly operation

DETEF-PALD is a segmented program, which modifies itself by overlays during operation. The initialization dialogue runs as follows:

•LI PALD ⤶
#GO ⤶
*OUT-

The use types T ⤶ for output (of binary) to the Teletype, R ⤶ for the High-Speed Punch, and unit:name ⤶ for output to a DECtape file. Use the Rubout key for correcting a miss-spelling.

*IN-

The user types T, R or unit:name for each of 1 to 10 inputs, separated by commas and finished by ⤶

*

The assembler answers with one asterisk for each legal input, and continues:

*LIST-

The user may respond with a decimal number terminated by T, R or ⤶ , according to the choice of an assembly listing produced on the Teletype, on the High-Speed Punch, or not at all. The number

specifies the number of initial lines of the source program which are excluded from the listing; the user may use this feature in order to avoid unnecessary listing and thus speed-up the assembly operation.
When this option of skipping initial program lines is intended, the pseudo-instruction XLIST should appear at the begin of the source program. In fact, the LIST-option works by issuing this command after completion of the specified count of lines. With this count being zero or not specified, the operation in this respect will be sensed by the user as normal DEC-PALD operation (though listing will be interrupted after 4096 lines).

*SYMT-

The user types X if the extended symbol table is intended, or ↵ if not. The assembler then **starts** the assembly process, asking for punched tape inputs by typing '↑'; to which the user, after loading the current reader, answers by a CTRL/P key-in. For further information on assembler operation, refer to the DEC-PALD manual.

Saving parts of the symbol table on DECtape will particularly impede the producing of an alphabetic list of symbols, since in the sorting process the whole table has to be searched once for each item listed. In order to circumvent this, the symbol list will be split in two, one for those symbols which are in core, and a second one for those saved on DECtape. Thus no time will be lost in swapping during this part of the assembly.
The option of an extended symbol table will also influence the output to DECtape: the produced B-file is built-up of 1-block segments instead of 2-blocks. This may increase the amount of DECtape movements during assembly. In order to avoid this the user should arrange for separate DECtape units being used for storage of symbol table, input files, and output files.

4. Creating the DETEF-PALD file

DETEF-PALD is obtained from a punched tape comprising three binaries by the following procedure:

1) Load binary no. 1
2) Save PALØ!1-6577;5ØØØ
3) Load binary no. 2
4) Save PAL1!24ØØ-2422,4Ø66-45Ø3;7777
5) Load binary no. 3
6) Save PAL2!2411-2761;7777
7) Allocate L PALD!41;7777
8) Add to Library PALD: PALØ!Ø, PAL1!33, PAL2!37

In order that the symbol table extension may be saved on a DECtape unit other than the no. Ø, the following patch has to be made to the C-file PAL1 before adding to the library file (use the program OEDT):

block no. STBL+2, loc.no. 13: change ØØØØ to xØØØ, 'x' denoting an octal numeral. The symbol table will then be saved on DECtape unit no. x.

TABLE OF PAL INSTRUCTIONS AND PSEUDO INSTRUCTIONS

I. INSTRUCTIONS

AND=	0000	SMA=	7500	MQL=	7421	PCF=	6022
TAD=	1000	SZA=	7440	LSR=	7417	PPC=	6024
ISZ=	2000	SPA=	7510	ASR=	7415	PLS=	6026
DCA=	3000	SNA=	7450	SHL=	7413	KSF=	6031
JMS=	4000	SNL=	7420	NMI=	7411	KCC=	6032
JMP=	5000	SZL=	7430	DVI=	7407	KRS=	6034
IOT=	6000	SKP=	7410	DTLB=	6774	KRB=	6036
OPR=	7000	OSR=	7404	DTRB=	6772	TSF=	6041
NOP=	7000	HLT=	7402	DTSF=	6771	TCF=	6042
CLA=	7200	CIA=	7041	DTXA=	6764	TPC=	6044
CLL=	7100	LAS=	7604	DTCA=	6762	TLS=	6046
CMA=	7040	STA=	7240	DTRA=	6761	RDF=	6214
CML=	7020	STL=	7120	ION=	6001	RIF=	6224
RAR=	7010	GLK=	7204	IOF=	6002	RMF=	6244
RTR=	7012	SCL=	7403	RSF=	6011	RIB=	6234
RAL=	7004	SCA=	7441	RRB=	6012	CDF=	6201
RTL=	7006	MQA=	7501	RFC=	6014	CIF=	6202
IAC=	7001	MUY=	7405	PSF=	6021		

II. SPECIAL CHARACTERS AND PSEUDO INSTRUCTIONS

OCTAL
DECIMAL
TEXT
XLIST
FIXTAB
EXPUNG
PAUSE
PAGE
FIELD

" & $; [(! * . , = Z I - + (SPACE)

The DECtape File-handling System "DETEF"

Users Manual of April 7, 1972

Revision of Oct., 1972

ERRATA

Page 13 line 15: change '7744' to '7745'
" 14 " 15: " '6' " '5'
" 16 " 13: " '165' " '164'
" 16 " 14: " '122' " '124'
" 18 " 31-32 should read:
'... by typing an octal location number
(Ø-2ØØ) and a "/", reading the ...'
Page 20 line 1: delete 'origins ...,'
" 28 " 5: change '7435' to '7436'
" 28 " 9: " '464' " '465'
" 34 " 12: append '; command "EXIT "'
" 34 " 14: " '; start System at 7777'