IDENTIFICATION

Product Code:	DEC-10-SBAB-D
Product Title:	Creating a PDP-10 Time-Sharing Monitor
Date Created:	March 8, 1968
Maintainer:	Software Services

The process of creating a working time-sharing Monitor from separate source files involves four basic steps:

1. Assembling the source files (Chapter 1).
2. Combining the resulting relocatable binary files into one Monitor Library file (Chapter 2).
3. Building a customized Monitor with System Builder (BUILD) and saving the core image of the Monitor (Chapter 3).
4. Exec mode loading the Monitor built and saved in step 3 (Chapter 4).

Normally, a user who wants only to create a system that matches his own hardware configuration need only perform steps 3 and 4. However, any user who wishes to rewrite any portion of the Monitor, or to add any sections of code to it, must be familiar with all four steps.

The Monitor can also be debugged and/or patched by use of the Executive Dynamic Debugging Technique (Exec DDT) program.

CHAPTER 1
ASSEMBLING OF MONITOR SOURCE FILES

The following discussion assumes that correctly edited source files of the Monitor components have been prepared and are ready for assembling. The assumption is also made that the user wishes to assemble all source files comprising the Monitor (not only those source files which he may have modified).

1.1 USE OF THE SYSTEM PARAMETER TAPE IN MONITOR ASSEMBLING

Every source file of the Monitor (except those described under "Channel Interrupt and Null Channel Routines", below) must be assembled with the System Parameter Tape file, a source file with the filename "S". This file must be assembled prior to any Monitor routine since it defines symbols used by that routine. For example, the Macro command string for assembling the Monitor routine APRSER would be:

*DTA3:APRSER,DSK:APRSER/C←DSK:S,APRSER.MAC↵

1.2 FEATURE SWITCHES

Among the symbols defined in S are a series of feature switches, which are assigned mnemonic tags beginning with the letters FT....and are defined to be either zero or nonzero (the nonzero setting is normally equal to -1). These feature switches are found at the end of the S file.

In addition to being used at System Build time, these switches also serve to define assembly parameters to the Macro Assembler; that is, in many cases, the values of these switches will cause a given source file to assemble as one of several possible versions of a routine. For instance, the setting of FTLOGIN will determine, in part, the relocatable binary version of APRSER produced. If FTLOGIN is set to zero, those portions of APRSER dealing with the interpretation and execution of the LOGIN command will not be assembled, and as a result the LOGIN feature will not be a part of the Monitor finally produced. In the version of the S file distributed by Digital, most feature switches are initially set to nonzero to indicate that the associated features are to be included.

At this point, the importance of checking the settings of the feature switches before assembling should be evident. There are three particularly important feature switches which determine the overall nature of the system: 1) FTDISK, 2) FTLOGIN, and 3) FTSWAP. The settings of these switches as related to the type of Monitor produced are given in the following table:

Feature Switch	Multiprogramming Nondisk System	Multiprogramming Disk System Nonswapping	Multiprogramming Disk Swapping System
FTDISK	0	-1	-1
FTLOGIN	0	-1	-1
FTSWAP	0	0	-1

If any of these switches is set to nonzero, the user must refer to the comments related to the switch on the source listing of the S file to determine which specific routines must be assembled when the particular switch is set. Generally, the comments given with each feature switch tell the user how to assemble the needed routines.

1.3 ASSEMBLING CHANNEL INTERRUPT AND NULL CHANNEL ROUTINES

In the rare instance when the user has to reassemble the channel interrupt or null channel routines, he must adhere to the procedure described below.

Seven routines in each category (CHAN1.REL through CHAN7.REL, and NULL1.REL through NULL7.REL) must be assembled. Since the coding for all seven routines in each category is the same (only the symbols are different), only two source files are maintained: CHAN.MAC and NULL.MAC. These two files are each in the form of a macro definition, "C" for the channel routine, and "N" for the null routine. The calling of either of these macros with a numeric argument (channel number) produces the assembled routine for that channel. The simplest way to assemble these routines is to specify the Teletype as an input device (note that the S file is not used in assembling these routines).

```
.R MACRO
*DTA2:CHAN1,LPT:←DTA3:CHAN.MAC,TTY:
LOAD NEXT FILE
C 1                                   User types macro C with the
                                      numeric argument 1 (channel
                                      number).
END
↑Z                                    ↑Z (CTRL Z) ends Teletype
                                      input.
```

(End of pass 1 of Macro Assembler)

LOAD NEXT FILE	Beginning of pass 2.
C 1↵	User repeats macro and END statements.
END↵	
↑Z↵	
THERE ARE NO ERRORS	
etc.	

Note that the Teletype input must be repeated for pass 2. The above sequence is performed from the beginning for each of the remaining six CHANn.REL routines, then for each of the seven NULLn.REL routines (this time typing the "N" macro with a numeric argument). The end result is 14 relocatable binary files.

CHAPTER 2
CREATING THE MONITOR LIBRARY FILE

The current version of Build expects all relocatable binary files that comprise a version of the Monitor to be combined into a Library file. This library file should be named as follows.

Multiprogramming, Nondisk System	SYS40N.REL
Multiprogramming Disk System, Nonswapping	SYS40D.REL
Multiprogramming Disk Swapping System	SYS50.REL

The procedure for obtaining each of these three versions of Monitor is described under Feature Switches in the previous chapter.

The Library file is created by the File Update Generator (FUDGE2) program. If a Library file is to be created from scratch, the Append (/A) switch can be used.

```
.R FUDGE2
*DTA1:SYS40N←DTA3:FIRST,DTA2:CHAN1,CHAN2,
CHAN3, CHAN4, CHAN5, CHAN6, CHAN7, LOWCOR,
NULL1,.....NOTDDT, ONCE/A($)

*                                   FUDGE 2 is completed.
```

All files must be specified in the proper order so that the resultant Library file will be properly formatted for Build. For example, the order of subfiles on the Library file that Build expects for a SYS40N.REL file is as follows.

NOTE

A listing of the subfiles of any of the three Monitor Library files can be obtained by use of the List option (/L) in FUDGE2.

```
.R FUDGE2
*TTY:/L←DTAL:SYS40N($)

FIRST
CHAN1
CHAN2
CHAN3
CHAN4
CHAN5
CHAN6
CHAN7
```

```
LOWCOR
NULL1
NULL2
NULL3
NULL4
NULL5
NULL6
NULL7
SYSINI
   .
   .
(all other Monitor subfiles in any order)
   .
   .
UUOCON
PATCH
SYSMAK
EDDT
NOTDDT
ONCE
*
```

NOTE

The subfiles listed by name must be in the order shown.

The procedure is much simpler if the user has merely created new files with which he wishes to replace older subfiles. He can then use the Replace option (/R) in FUDGE2, since the original file is assumed to be in the proper order.

```
.R FUDGE2
*DTA2:SYS40N←DTA3:SYS40N<SCNSER>,DTA4:SCNSER/R($)
*
```

If the user wishes only to add additional routines to his Monitor, he does not insert the new routines into the Library file; instead, he incorporates them into his customized Monitor at Build time by typing their names in answer to the query.

```
TYPE "DEVICE:NAME:" FOR ANY SPECIAL ROUTINES TO BE LOADED.
TYPE ALT-MODE WHEN THROUGH.
*
```

It should be noted that certain subfiles must be included in certain Library files but not in others.

a. SYS40N.REL should contain the same subfiles as SYS40D.REL but in addition exclude DSKER and DSKINT.

b. SYS40D.REL should contain all subfiles except CLKCSW, QCSS, SWAP, and SWAPSER.

c. SYS50.REL should contain all subfiles except CLKCSS.

The user must make certain that the properly assembled version of each routine is placed in the appropriate Library file.

CHAPTER 3
BUILDING A CUSTOMIZED MONITOR

The procedures detailed in this chapter assume the existence of the Monitor Library files named SYS40N.REL, SYS40D.REL, and SYS50.REL, all complete, properly assembled and ordered.

The System Builder (BUILD) program will run under time sharing in the same manner as any other CUSP program.

3.1 SAMPLE DIALOGUE OF SYSTEM BUILDER

Mount the DECtape containing SYS40N.REL, SYS40D.REL, and SYS50.REL on drive n and type:

.ASSIGN DTAn: DTA1↵

Run Build in sufficient core to allow for the Build program (3K) plus the estimated size of the symbol table plus the estimated size of the loaded system. Build uses as much core as it is given.

.RUN DTA3: BUILD 18

At this point, Build starts asking specific questions and indicates to the user how to answer them. In the dialogue given below, user responses are underlined; the remaining lines are Build type-outs.

a. IS EITHER A 10, 20, OR 30 SYSTEM TO BE BUILT (TYPE Y OR N)?

   N↵

b. IS A 10/40 SYSTEM TO BE BUILT (Y OR N)?

   Y↵

c. DO YOU HAVE A DISK?

   Y↵

d. DO YOU WISH TO HAVE EXEC DDT LOADED (Y OR N)?

   Y↵

e. DO YOU WISH TO HAVE LOCAL SYMBOLS LOADED (Y OR N)?

   Y↵

   [EXEC DDT and local symbols need be loaded only for debugging the Monitor (for example, if the user has modified or added to it). Both of these take up large amounts of core at Build time and may cause symbol table overflow.]

f. TYPE NAME OF THIS SYSTEM (10 CHARACTERS OR LESS)

   ABCD1234↵

   [This name is typed when the Monitor is loaded.]

g. TYPE NAME OF SYSTEM DEVICE

   DSK↵

h. DO YOU HAVE ANY OF THE FOLLOWING? TYPE Y OR N

PT READER?

Y↵

PT PUNCH?

Y↵

LINE PRINTER?

Y↵

CARD READER?

Y↵

DISPLAY?

N↵

HOW MANY (DECIMAL) OF EACH OF THE FOLLOWING DO YOU HAVE

DEC TAPES?

5↵

DO YOU HAVE TD 10 DECTAPE CONTROL (Y OR N)?

Y↵

MAG TAPES?

0↵

NOTE

Type "0", not "N".

i. JOBS TO RUN AT ONE TIME(BOTH ATTACHED AND DETACHED)?

15↵

j. PSEUDO TELETYPES?

3↵

k. TYPE "DEVICE-MNEMONIC:CHANNEL" FOR ANY NON-STANDARD DEVICES
TYPE ALT-MODE WHEN THROUGH

*XYZ:6↵

*$

[Enter the device mnemonics and priority channels for any of the user's own device service routines. See User Monitor Routines.]

l. TYPE "SYMBOL:VALUE" FOR ANY CHANGES (VALUE IN DECIMAL)
TYPE ALT-MODE WHEN THROUGH

*NSPMEM:5000↵

*$↵

[The value of any global may be changed (see Symbol Modification.)]

m. TYPE "DEVICE:NAME: FOR ANY SPECIAL ROUTINES TO BE LOADED.
TYPE ALT-MODE WHEN THROUGH

*DTA1:XYZSER ↵

* $↵

[Any routine to be included in the Monitor, but not linked with the rest of the Monitor, should be specified here. In addition, any device specified in response to question k whose binary coding is not in the appropriate file of the Build tape should be specified. After ALTMODE is struck, there is a long delay as most of the loading is performed at this point.]

n. DO YOU WANT A STORAGE MAP?

↑C↵

.ASSIGN LPT ↵

.LPT ASSIGNED

.CONT ↵

Y↵

[If a storage map is requested, an INIT is performed by Build on the line printer; if Build finds that the line printer is not available, it types the storage map on the user's Teletype. Thus, the typeins above to assign the line printer are optional.]

o. EXIT

↑C

[Build overlays itself with the new Monitor and exits.]

p. .SAVE DTA2:SYS001

JOB SAVED

↑C

The Build process has been completed and the new Monitor is in the user's core area. The user now saves this core image of the new Monitor on a DECtape (DECtape 2, in this example) and assigns it a filename. Loading this file in exec mode will give the user a working Monitor (see the next chapter).

3.2 SYMBOL MODIFICATION

The user can change the value of any global symbol in the Monitor at Build time. Some symbols that might commonly be changed and their standard values (in decimal) are:

a)	JIFSEC = 60	The frequency of the power source. Users who have 50 cycle power must change this value.
b)	NSPMEM = 2000	The number of microseconds in 1000 memory cycles (i.e. memory cycle time in microseconds multiplied by 1000). Users having other than 2μs memory cycle times must change this value.

c) DTTRY = 4

The number of retries performed upon sensing a DECtape error.

d) MTSIZ = 128

Size of magnetic tape records (i.e., number of data words in a buffer).

e) LPTSIZ = 24

Size of line printer buffer (in number of data words).

f) DETDDB = 0

Maximum number of detached jobs. If a user wants to run 15 jobs but has only 12 Teletypes, then DETDDB must be set to 3 at Build time.

g) STDENS = 2

Magnetic tape density and parity. It is computed by the formula

STDENS = d + p

STDENS VALUES

p (parity) VALUES	d (density)		
	1 200 BPI	2 556 BPI	3 800 BPI
0 = Odd, binary	1	2	3
4 = Even, BCD	5	7	8

h) devCHN = n

Priority Interrupt channel assignments. The standard PI channels and the name of the associated global are given in the table below. If, for example, a 10/50 user wishes to change the assignment of his paper tape punch from PI 5 to PI 3, he types

PTPCHN = 3

PI Channel	Global Symbol	Standard Values	
		10/40 Nondisk System	10/40 Disk, 10/50 Systems
Paper tape reader	PTRCHN	3	4
Paper tape punch	PTPCHN	4	5
Line printer	LPTCHN	3	4
Card reader	CDRCHN	2	3
Display	DISCHN	5	6
Scanner	SCNCHN	3	4
Console Teletype	CTYCHN	3	4

PI Channel	Global Symbol	Standard Values	
		10/40 Nondisk System	10/40 Disk, 10/50 Systems
Arithmetic Processor	APRCHN	2	3
Light Pen	PENCHN	4	5
Clock	CLKCHN	7	7
Data control for DECtape and mag tape	DCTCHN	1	2
Data control for disk	DCBCHN	-	1
DECtape	DTCCHN	4	5
Magnetic tape	MTCCHN	3	4
Disk	DSKCHN	-	5

3.3 USER MONITOR ROUTINES

The user can add routines of his own to his customized Monitor. These routines can be of three types: 1) nondevice service routines, 2) device service routines for special devices, and 3) replacements to standard device service routines.

3.3.1 Nondevice Service Routines

At Build time, each such routine is a file on any input device. When requested by Build (question m, Sample Dialogue of System Builder), the user keys in the name of the input device and the name of the file.

3.3.2 Device Service Routines for Special Devices

At Build time, these routines could come from one of two sources:

1) As files on any input device. In this case, they are handled as described under Nondevice Service Routines, above.

2) As part of the appropriate Monitor Library file (SYS40N, SYS40D, or SYS50). In this case, they would have been incorporated into the Library file by FUDGE2 (see previous chapter) and must appear after SYSINI.

In either of these cases, the user must type a 3-letter mnemonic and priority channel for the special device when requested by Build (see question k). Build expects any special device to have a service routine written according to the methods discussed in the Real-Time Use of the PDP-10 Monitors Manual DEC-10-WMAA-D, and Digital-supplied device service routines. Build links the device into the interrupt routine for the specified channel and links its device data block into the DDB chain.

3.3.3 Replacement of Standard Monitor Device Service Routines

The user may also replace an existing device service routine for a standard device by one of his own. For example, a user may wish to use his own line printer service routine rather than the one supplied by Digital. He does this by substituting his own routine for the standard routine during the creation of a Monitor Library file by FUDGE2 (see the previous chapter).

3.4 DIGITAL-SUPPLIED MONITOR ROUTINES

In addition to the standard Monitor routines supplied by Digital, the storage map produced by Build lists routines called BUILT1, BUILT2, and BUILT3; these are artificial routines created by Build to hold the symbols that Build creates.

3.5 BUILD ERROR MESSAGES

Error messages produced by Build are given in the table below.

Table 3-1
Build Error Messages

Message	Meaning
ERROR IN LOADER COMMAND	Indicates either a system or hardware malfunction.
ILL. FORMAT LIBRARY TAPE BAD	The input Library file is in error. Try to rerun Build. If this fails, recreate the Library file.
INPUT ERROR LIBRARY TAPE BAD	Rerun Build. If this fails, recreate the Library file.
symbol old-value new-value MUL.DEF. GLOBAL	Multiple-defined global. The old value is accepted and processing continues. This message may occur if the user has keyed in this symbol; in this case, the keyed-in value is always accepted. Otherwise the input to Build is in error and should be investigated and corrected.
device NOT AVAILABLE	The device which the user indicated his routine was on is unavailable. Try the response again.
NOT ENOUGH CORE	Occurs only when building 10/40 or 10/50 systems under time sharing. Rerun Build and assign it more core. If this fails, try the following: a. Run Build under a Monitor containing only DECtape and line printer service routines. b. Do not load Exec DDT. c. Do not load local symbols.

Table 3-1 (cont)
Build Error Message

Message	Meaning
program-name.REL NOT FOUND	User has requested that one of his own routines be included in Monitor, but the filename of the routine cannot be found on the device specified. Try the request again.
symbol NOT FOUND EXIT	A device data block or device interrupt service routine cannot be found. Try the following: a. If the user is including his own routines in the Monitor, he should check that the tags on his device data blocks and interrupt routines are spelled correctly and are globals. b. Check if a device service routine is missing from the input Library file. c. Restart Build from the beginning.
SYSTEM OR MACHINE ERROR. RESTART PROGRAM	This message should never appear.
UNDEFINED GLOBALS	Appears at the end of the storage map (on either the line printer or Teletype). If the input to Build is supplied solely by Digital, this message should never appear.
WHAT?	The user has made an error in a typein. Repeat the response.

CHAPTER 4
EXEC MODE LOADING OF THE SYSTEM

Exec mode loading of the Monitor provides a working Monitor and must be performed every time a system is loaded into an "empty" machine. The procedure for exec mode loading is as follows.

1. Set the starting address 34000 in the address switches on the console.
2. Load TENDMP V006 into the paper tape reader.
3. Depress the STOP, I/O RESET, and READ IN keys. The paper tape should now read in.
4. Mount the DECtape containing the saved Monitor on DECtape unit n and type n(Ⓢ). The DECtape is positioned by TENDMP at block 100.
5. Two actions are possible at this point and are chosen by the following typeins.

F(Ⓢ)	Lists the file directory of the DECtape.
L(Ⓢ)filename SAV	Loads the Monitor. Filename is the name assigned when the core image of the Monitor was saved.

NOTES

1. TENDMP reads from LINK to LINK.
2. TENDMP won't handle BLOCK 1 or end zone flags.