

Getting Started with Interdata OS/32

March 28, 2016

Copyright ©2016 Don Stalkowski. Portions may be Copyright by The Perkin-Elmer Corporation.

The author of, and contributors to this document provide NO WARRANTY; not even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

My thanks to Davis Johnson for his invaluable help in getting OS/32 and MTM running.

Contents

1	Installing OS/32 8.1	2
2	Usage	6
3	Sysgen	8
4	Installing MTM	13
5	CAL, Fortran, and Pascal	14
5.1	“HELLO WORLD” in CAL32	15
5.2	Fortran “HELLO WORLD”	16
5.3	Pascal “Hello World”	18
6	Notes	19
6.1	Disks	19
6.2	Memory	19
6.3	Transferring Files to id32	20

This guide shows the user how to install OS/32, MTM, Fortran, and Pascal on an Interdata id32 simh system. Examples of programming in CAL, Fortran, and Pascal are also given. The guide is based on a Linux host system.

1 Installing OS/32 8.1

1. These tapes are needed:

```
04-082M71R16S_OS32_starter.tap.gz
04-082M71R16_OS32_8.1.tap.gz
```

They are found in:

```
http://www.bitsavers.org/bits/Interdata/32bit/os32/OS32\_8.1/
```

2. Installation instructions are in:

```
04-082M95R16_OS32v8.1Installation_May85.pdf
```

The operator manual will explain most of the commands used here. *Read it and learn the commands.*

```
48-030F00R03_8.1_Operator_1985.pdf
```

3. gunzip both tapes. In the following it is assumed the tapes are in the same directory as id32 is run in.
4. Boot the starter tape. See section 7.2 and 7.3 of the installation manual. Section 7.3.1 gives a boot sequence which doesn't work with simh. However, if X'78' and X'7A' contain the proper device information for the magtape, then "boot mt0" will work.

We'll be using a 67MB disk which is known to simh as an msm80.

```
./id32
```

```
Interdata 32b simulator V4.0-0 Beta          git commit id: fa407e67
sim> set cpu 832
sim> set dm0 msm80
sim> att dm0 dm0.dsk
DM: creating new file
sim> att mt0 04-082M71R16S_OS32_starter.tap
```

```
sim> d 78 85a18540
sim> boot mt0
```

```
OS32MT08-01.OS32MTS1    LICENSE
MEMORY TEST IN PROGRESS
END OF MEMORY TEST
UNAVAILABLE MEMORY BLOCKS =      0
ENTER DATE AND TIME
*SET TIME 01/28/15,09:29:00
*
```

5. The next step is to format the disk. The format program, FASTCHEK, must be loaded from the starter tape. The layout of the tape is show in section 7.1.2 of the manual. After the boot, the tape is positioned just before the first file mark ("-FM-") so we must skip over 5 file marks before we can read FASTCHEK.TSK.

```
*FF MAG1:
*FF MAG1:
*FF MAG1:
*FF MAG1:
*FF MAG1:
*LOAD F,MAG1:,300
TSKID = F
```

6. If we look at Appendix F we see that the 67 MB disk is DSC4. Format it with the following command:

```
*ST ,INIT=DSC4:,VOL=SYS,LI=CON:
*09:30:06  F:PERKIN-ELMER OS/32 FASTCHEK 03-344 R00-04
*09:30:06  F:
*09:30:06  F:INITIALIZE DSC4:  MODE = NOREADCHECK WITH WRITERECOVERY
*09:30:06  F:    VOLUME SYS    DIRECTORY FOR 640 FILES AT CYLINDER 1 REQ
*09:30:06  F:
*09:30:06  F:VOLUME DESCRIPTOR ERROR
*09:30:06  F:    INVALID VOLUME NAME 00000000
*09:30:06  F:**** SWITCHING TO READCHECK MODE ****
*09:30:06  F:
*09:30:06  F:WARNING: PACK ADMINISTRATION FILE (PACKINFO.DIR) NOT FOUND
*09:30:06  F:
*09:30:06  F:0 DEFECTIVE SECTORS FOUND
*09:30:06  F:
*09:30:06  F:PACK INITIALIZED - PREALLOCATED DIRECTORY AT 000140, BIT MA
```

```
*09:30:06    F:END OF TASK      0
*
```

7. Now BACKUP is used to restore the OS/32 tape to the disk.

```
*MARK DSC4:,ON
DSC4:  SYS
*FF MAG1:
*L BACKUP,MAG1:
TSKID = BACKUP
```

Type control-E (^E) to return to simh.

```
*Simulation stopped, PC: 22B1A (BAL R12,18EF4)
sim> att mt0 04-082M71R16_OS32_8.1.tap
```

Typing “c” (continue) will return you to OS/32.

```
sim> c
ST ,IN=MAG1:,OUT=DSC4:,LIST=CON:,VERIFY
```

; lots of output...

```
*10:00:52    BACKUP:   130 FILES TRANSFERRED
*10:00:52    BACKUP:   130 FILES TRANSFERRED
*10:00:52    BACKUP:OPTION VERIFY
*10:00:52    BACKUP:OPTION VERIFY
*10:00:52    BACKUP:END OF TASK      0
*
```

8. MARK the disk OFF (very important) and stop simh.

```
*MARK DSC4:,OFF
*^E
Simulation stopped, PC: 22B1A (BAL R12,18EF4)
sim> quit
```

9. Section 2.3.3 of the operator manual describes how to boot from disk. Again, this sequence doesn't work for simh but if step 7 (in §2.3.3) is followed, the boot command may be used. The extension is the “001” from “OS32MTS1.001.”

```
./id32
```

```
Interdata 32b simulator V4.0-0 Beta
```

```
sim> set cpu 832
sim> att -e dm0 dm0.dsk
sim> d 7c 00000001
sim> boot dm0
```

```
OS32MT08-01.OS32MTS1    LICENSE C-838-WB
MEMORY TEST IN PROGRESS
END OF MEMORY TEST
UNAVAILABLE MEMORY BLOCKS =      0
ENTER DATE AND TIME
*SET TIME 01/29/16,10:00:00
*
```

Running From Disk

1. Attach the OS/32 disk.

```
./id32
```

```
Interdata 32b simulator V4.0-0 Beta
```

```
sim> set cpu 832
sim> att -e dm0 dm0.dsk
sim> d 7c 00000001
sim> boot dm0
```

For convenience, these commands can be put in id32.ini.

2. Enter the date and time.

```
OS32MT08-01.OS32MTS1    LICENSE C-838-WB
MEMORY TEST IN PROGRESS
END OF MEMORY TEST
UNAVAILABLE MEMORY BLOCKS =      0
ENTER DATE AND TIME
*SET TIME 01/29/16,10:00:00
```

3. MARK the disk ON and set it up as the default volume.

```
*MARK DSC4:,ON
DSC4:  SYS
*V SYS/SY
*V SYS/TE
*
```

4. Before stopping simh, be sure to MARK the disk OFF.

```
*MARK DSC4:,OFF
*^E
Simulation stopped, PC: 22B1A (BAL R12,18EF4)
sim>quit
#
```

5. If you forget to MARK a disk OFF before exiting simh, it may be repaired with the following:

```
*MARK DSC4:,ON,P
DSC4:  SYS  PROT
*L SYS:FASTCHEK
TSKID = FASTCHEK
*MARK DSC4:,OFF
*T FASTCHEK
*ST ,CH=DSC4:,NOR,EXT,LI=CON:
*12:38:58 FASTCHEK:PERKIN-ELMER OS/32 FASTCHEK 03-344 R00-04
*12:38:58 FASTCHEK:
*12:38:58 FASTCHEK:CHECK DSC4:      MODE = NOREADCHECK
*12:38:58 FASTCHEK:      WITH WRITERECOVERY EXTENDALLOWED
*12:38:58 FASTCHEK:
*12:38:58 FASTCHEK:VOLUME NAME IS SYS
*12:38:58 FASTCHEK:
*12:38:58 FASTCHEK:0 DEFECTIVE SECTORS RECORDED
*12:38:58 FASTCHEK:
*12:38:58 FASTCHEK:
*12:38:58 FASTCHEK:CHECK COMPLETE - VOLUME SYS  READY TO BE MARKED ON
*12:38:58 FASTCHEK:END OF TASK      0
*
```

Now continue at step 3.

2 Usage

OS/32 console commands are documented in the operator manual, 48-030F00R03.8.1_Operator_1985.pdf. Some OS/32 programs have builtin help. These are: LINK, BACKUP, CLU, COPY32, EDIT32, and PATCH.

CSS files contain scripts and are run just like a regular command. See the operator manual for details.

In general, a task is executed by first LOADING it and then STARTing it. A “paused” task may be stopped with the CANCEL command or continued with the CONTINUE command. The START command may also accept options for the task.

Typing mistakes may be corrected with control-h (terminal dependent).

Example:

```
*L EDIT32
TSKID = EDIT32
*ST
*12:44:43 EDIT32:PERKIN-ELMER OS/32 EDIT 03-145 R04-01
*EDIT32>HELP
*12:47:19 EDIT32:FOR A LIST OF COMMANDS TYPE HELP *
*12:47:19 EDIT32:FOR HELP ON ANY COMMAND MNEMONIC, TYPE HELP <MNEMONIC>
*EDIT32>HELP *
*12:47:45 EDIT32:AF(TER)          AG(AIN)          AL(TER)
*12:47:45 EDIT32:A(PPEND)        BE(FORE)         B(OTTOM)
*12:47:45 EDIT32:CH(ANGE)        CO(LUMN)        DE(LETE)
*12:47:45 EDIT32:DO(NE)          END             F(IND)
*12:47:45 EDIT32:G(ET)           H(ELP)          INC(LUDE)
*12:47:45 EDIT32:INS(ERT)        M(OVE)          N(OFOUND)
*12:47:45 EDIT32:NOT(RUNCATE)    O(PTION)        P(AUSE)
*12:47:45 EDIT32:RE(PLACE)       RU(LER)         S(AVE)
*12:47:45 EDIT32:SC(REEN)        SU(BSTITUTE)    TO(P)
*12:47:45 EDIT32:TR(UNCATE)      T(YPE)          -
*12:47:45 EDIT32:+
*12:47:45 EDIT32:FOR HELP ON ANY OF THE ABOVE COMMAND MNEMONICS,
*12:47:45 EDIT32: TYPE HELP <MNEMONIC>
*EDIT32>HELP GET
*12:48:19 EDIT32:GET: THIS COMMAND ASSIGNS A FILE FROM DISK AND PREPARE
*12:48:19 EDIT32:EDITING.
*12:48:19 EDIT32:
*12:48:19 EDIT32:SYNTAX: (G)ET FD
*12:48:19 EDIT32:
*12:48:19 EDIT32:FUNCTION: ASSIGNS THE FILE SPECIFIED BY FD AND READIES
*12:48:19 EDIT32:EDITING.
*EDIT32>GET INIT.CSS
*EDIT32>T 1-5
*12:44:06 EDIT32: 1 **INIT.CSS [FD] , [MEMORY]
*12:44:06 EDIT32: 2 **03-217
*12:44:06 EDIT32: 3 **
*12:44:06 EDIT32: 4 ** ZERO AN ALLOCATED CONTIGUOUS FILE
*12:44:06 EDIT32: 5 **
*EDIT32>END
```

```
*12:49:12  EDIT32:END OF TASK      0
*
```

3 Sysgen

At this point it's probably a good idea to reconfigure the system and do a sysgen. The sysgen process is documented in 48-037F00R02_SYSGEN32_7.2_1984.pdf (in bitsavers os32/7.2_1984 directory).

The default configuration file is OS32MTS1.SYS. We'll do the following:

- enable 3 msm80 drives
- configure MTM terminals
- set SYS as the default device
- other minor changes

You'll need to know how to use the editor for this. The "CH", "INS", and line number commands should suffice but see the console log if you have problems.

A copy of the new configuration file is on tape "new02.tap" for those who do not want to learn to use EDIT32.

Here's the original OS32MTS1.SYS and the changes that have to be made:

```
1 COPY
2      MCALL DCBI,CCBI,CONVNUM,EVNGEN,DMT,SLABEL
3      MCALL $DCB$, $CCB, $EVN, $TABL$
4      MCALL MMDGEN, $MMDDCB, DCB53, DCB54
5      MCALL MTPI, $MTP, DCB64, DCB65
6      MCALL BIOCGEN, BIOCD CB, BIOCEQU, $VFDCB, DCB39
7 ENDCOPY
8 VERSION OS32MTS1          *OS32MTS1 SYSTEM
9 CPU      8/32,8           *CPU = 7/32 OR 8/3
10 MEMORY  1024             *MEMORY = 1024-KB
```

Increase the system space to 100.

```
11 DSYS      50             *SYSTEM SPACE = 50
12 MEMCHECK                   *MEMORY DIAGNOSTIC
13 DEVADS     3             *MAX OF 1023 DEVIC
14 FLOAT      S,S           *FLOATING POINT SO
15 CLOCK 60,6C,6D,D         *60-HZ, LFC=6C, PI
16 BACKGROUND 16,200        *BACKGROUND TASK'S
```

Set the default VOLUME and TEMP to SYS.

17	VOLUME	MT8A	*DEFAULT SYSTEM VO
18	TEMP	MT8A	*DEFAULT TEMPORARY
19	DIRECTORY		*CORE DIRECTORY IN
20	DISCBLOCK	32,IN=8/1	*DISKFILE BLOCKING
21	CSS	8	*CSS NESTING DEPTH
22	CMDLEN	100	*COMMAND BUFFER LE
23	LOGLEN	100	*LOG BUFFER LENGTH
24	DEVICES		

Configure MTM terminals into your system. From Davis Johnson: "I prefer using the RS232 interface and driver ("PASLA" and "39 driver"). It also gets you lower case, recall and edit previous line and a bunch of other good stuff."

Replace line 25 with:

24.01	CON:	,10,39,XDCOD=X2A0D,RECL=120,CONS	
24.02	T020:	,20,39,XDCOD=X2A0D	*PE 550/PE 1100
24.03	T022:	,22,39,XDCOD=X2A0D	*PE 550/PE 1100
24.04	T024:	,24,39,XDCOD=X2A0D	*PE 550/PE 1100
24.05	T026:	,26,39,XDCOD=X2A0D	*PE 550/PE 1100
24.06	T028:	,28,39,XDCOD=X2A0D	*PE 550/PE 1100
24.07	T02A:	,2A,39,XDCOD=X2A0D	*PE 550/PE 1100
24.08	T02C:	,2C,39,XDCOD=X2A0D	*PE 550/PE 1100
24.09	T02E:	,2E,39,XDCOD=X2A0D	*PE 550/PE 1100
25	CON:	, 2,16,CONS	*CONSOLE - M33 TTY
26	CRT:	,12,39,XDCOD=X2A0D	*PE 550/PE 1100
27	CR:	, 4,96	*CARD READER
28	PR:	,62,113	*300-LPM LINE PRIN
29	CAS1:	,45,66	*CASSETTE NO 1
30	CAS2:	,55,66	*CASSETTE NO 2
31	FLP1:	,C1,55,SPINDLE=0	*FLOPPY DISK-1
32	FLP2:	,C1,55,SPINDLE=1	*FLOPPY DISK-2
33	MAG1:	,85,64,SELCH=F0,CONTR=0	* 800-BPI TAPE
34	MAG2:	,C5,65,SELCH=F0,CONTR=1	*1600-BPI TAPE
35	MAG3:	,85,68,SELCH=F0,CONTR=2	*6250-BPI TAPE HPT
36	MAG4:	,85,69,SELCH=F0,CONTR=3	*6250-BPI TAPE STC
37	MAG5:	,85,70,SELCH=F0,CONTR=4	*6250-BPI TAPE TEL
38	DSC0:	,C8,49,SELCH=F0,CONTR=B6	*2.5-MB DISK (REMO
39	DSC1:	,C6,51,SELCH=F0,CONTR=B6	* 5-MB DISK (REMO
40	DSC2:	,C7,50,SELCH=F0,CONTR=B6	* 5-MB DISK (FIXE

Change the following 3 disks to type "53" (msm80) and use separate addresses. Set DSC3 to address FD and DSC5 to address FE.

41	DSC3:	,FC,52,SELCH=F0,CONTR=FB	* 40-MB DISK (REMO
----	-------	--------------------------	--------------------

```

42 DSC4: ,FC,53,SELCH=F0,CONTR=FB      * 67-MB DISK (REMO
43 DSC5: ,FC,54,SELCH=F0,CONTR=FB      *256-MB DISK (REMO
44 ENDD
45 ENDC

```

Save the new configuration as NEW02.SYS.

Your new configuration file should look like:

COPY

```

        MCALL DCBI,CCBI,CONVNUM,EVNGEN,DMT,SLABEL
        MCALL $DCB$, $CCB, $EVN, $TABL$
        MCALL MMDGEN, $MMDDCB,DCB53,DCB54
        MCALL MTPI, $MTP,DCB64,DCB65
        MCALL BIOCGEN,BIOCD CB,BIOCEQU,$VFDCB,DCB39

```

ENDCOPY

```

VERSION OS32MTS1          *OS32MTS1 SYSTEM
CPU      8/32,8           *CPU = 7/32 OR 8/3
MEMORY   1024             *MEMORY = 1024-KB
DSYS      100             *SYSTEM SPACE = 1
MEMCHECK                *MEMORY DIAGNOSTIC
DEVADS     3              *MAX OF 1023 DEVIC
FLOAT      S,S            *FLOATING POINT SO
CLOCK 60,6C,6D,D         *60-HZ, LFC=6C, PI
BACKGROUND 16,200        *BACKGROUND TASK'S
VOLUME     SYS            *DEFAULT SYSTEM VOL
TEMP       SYS            *DEFAULT TEMPORARY
DIRECTORY                *CORE DIRECTORY IN
DISCBLOCK  32,IN=8/1     *DISKFILE BLOCKING
CSS        8              *CSS NESTING DEPTH
CMDLEN     100            *COMMAND BUFFER LE
LOGLEN     100            *LOG BUFFER LENGTH

```

DEVICES

```

CON: ,10,39,XDCOD=X2A0D,RECL=120,CONS
T020: ,20,39,XDCOD=X2A0D      *PE 550/PE 1100
T022: ,22,39,XDCOD=X2A0D      *PE 550/PE 110
T024: ,24,39,XDCOD=X2A0D      *PE 550/PE 1100
T026: ,26,39,XDCOD=X2A0D      *PE 550/PE 1100
T028: ,28,39,XDCOD=X2A0D      *PE 550/PE 1100
T02A: ,2A,39,XDCOD=X2A0D      *PE 550/PE 1100
T02C: ,2C,39,XDCOD=X2A0D      *PE 550/PE 1100
T02E: ,2E,39,XDCOD=X2A0D      *PE 550/PE 1100
CRT: ,12,39,XDCOD=X2A0D      *PE 550/PE 1100
CR: , 4,96                  *CARD READER
PR: ,62,113                 *300-LPM LINE PRIN

```

```

CAS1: ,45,66          *CASSETTE NO 1
CAS2: ,55,66          *CASSETTE NO 2
FLP1: ,C1,55,SPINDLE=0 *FLOPPY DISK-1
FLP2: ,C1,55,SPINDLE=1 *FLOPPY DISK-2
MAG1: ,85,64,SELCH=F0,CONTR=0 * 800-BPI TAPE
MAG2: ,C5,65,SELCH=F0,CONTR=1 *1600-BPI TAPE
MAG3: ,85,68,SELCH=F0,CONTR=2 *6250-BPI TAPE HPT
MAG4: ,85,69,SELCH=F0,CONTR=3 *6250-BPI TAPE STC
MAG5: ,85,70,SELCH=F0,CONTR=4 *6250-BPI TAPE TEL
DSC0: ,C8,49,SELCH=F0,CONTR=B6 *2.5-MB DISK (REMO
DSC1: ,C6,51,SELCH=F0,CONTR=B6 * 5-MB DISK (REMO
DSC2: ,C7,50,SELCH=F0,CONTR=B6 * 5-MB DISK (FIXE
DSC3: ,FD,53,SELCH=F0,CONTR=FB * 67-MB DISK (REMO
DSC4: ,FC,53,SELCH=F0,CONTR=FB * 67-MB DISK (REMO
DSC5: ,FE,53,SELCH=F0,CONTR=FB *67-MB DISK (REMOV
ENDD
ENDC

```

Build the SYSGEN command files. See section 7.6.1.

```

*OSGEN
****  OSGEN R08-01 STARTED  ***
*SYSGEN.CSS      CREATED
*SYSGEN32.CSS    CREATED
*SYSMACRO.CSS    CREATED
*SYSLINK.CSS     CREATED
*MLBCK.CSS       CREATED
*SYSCHECK.CSS    CREATED
****  OSGEN COMPLETED  ***

```

Build the new system.

```

*SYSGEN32 NEW02
*10:00:31  .BG:
*10:00:31  .BG:PERKIN-ELMER OS/32 SYSGEN32 03-356 R08-01
*10:00:31  .BG:END OF TASK      0
*
*SYSMACRO NEW02
*10:02:38  .BG: CAL MACRO/32 03-339 R00-01
*10:02:38  .BG:  NO ERRORS
*10:02:38  .BG:END OF TASK      0
*10:02:38  .BG:CAL/32 03-338R01-01
*10:02:38  .BG:SPTINIT          NO ERRORS    TABLE SPACE USED :    14K    DI
*10:02:38  .BG:DMTINIT          NO ERRORS    TABLE SPACE USED :     2K    DI

```

```

*10:02:40 .BG:DCBS          NO ERRORS   TABLE SPACE USED :    24K   DI
*10:02:40 .BG:EVNGEN        NO ERRORS   TABLE SPACE USED :     7K   DI
*10:02:40 .BG:END OF TASK    0
*
*SYSLINK NEW02
*10:03:16 .BG:PERKIN-ELMER OS/32 LINKAGE EDITOR 03-242 R01-03
*10:03:16 .BG:END OF TASK    0
*OS MAP ==> NEW02.LST
*OS OBJ ==> NEW02.OBJ
*OS TSK ==> NEW02.OS

```

Copy the new OS to a bootable name.

```

*L COPY32
TSKID = COPY32
*ST
*10:07:26 COPY32:PERKIN-ELMER OS/32 COPY 03-215 R01-03
*COPY32>COPY NEW02.OS,OS32MTS1.002
*10:07:59 COPY32:END OF MEDIUM - 721 RECORD(S) COPIED
*10:07:59 COPY32:COPY COMPLETE - 1 FILE(S) COPIED
*COPY32>END
*10:08:06 COPY32:END OF TASK    0

```

MARK DSC4 off and quit simh.

Modify your id32.ini to reflect the system changes.

When the system is booted, it searches the disk for an OS file ending in the suffix stored in word 7C, so modify your id32.ini to look like this:

```

set cpu 832
set ttp enabled
set pas devno=20
att pas 1026
att -e dm0 dm0.dsk
d 7c 00000002
boot dm0

```

The “set ttp enabled,” “set pas devno=20,” and “att pas 1026” commands are for the new serial lines.

Restart id32.

```
./id32
```

Interdata 32b simulator V4.0-0 Beta
Listening on port 1026

git commit id: fa407e67

```

OS32MT08-01.OS32MTS1    LICENSE C-838-WB
MEMORY TEST IN PROGRESS
END OF MEMORY TEST
UNAVAILABLE MEMORY BLOCKS =    0
ENTER DATE AND TIME
*set time 02/24/16,11:20
*mark dsc4:,on
DSC4:  SYS
*v sys
*v sys/te
*
```

4 Installing MTM

1. Unpack the MTM tape, 04-083M71R10.OS32MTM8.1.tap, onto SYS. You may have to use the DELETE option because of a duplicate INFORM.CSS: “st ,in=mag1:,out=dsc4:,list=con:,ac=0,verify,delete” A safer way is to unpack to a temporary disk and copy the files.
2. Generate MTM with the command “MTMSGN MAC.”
3. Run ACTUTY to add some users. User 255 is the admin user.

```

*l a,actuty
TSKID = A
*st ,com=con:,li=con:,cre=users.auf
*10:21:12  A:  OS/32      MTM ACCOUNT UTILITY 03-150 R02-01
*10:21:12  A:  ACCOUNT LIMIT =      255
*A>list
*10:21:21  A:  ACT    GRP          NAME          TIME LEFT    CPU LEFT
*A>add 255,255,pass1,master,*,*,FFFFFFF0
*10:22:30  A:  255    255    master          **          **
*A>add 25,20,user1,user1,*,*,FFFFFFF0
*10:23:08  A:   25     20    user1          **          **
*A>list
*10:23:11  A:  ACT    GRP          NAME          TIME LEFT    CPU LEFT
*10:23:11  A:   25     20    user1          **          **
*10:23:11  A:  255    255    master          **          **
*A>end
*10:23:20  A:END OF TASK      0
*ren users.auf,users.auf/255
```

4. Start MTM.

```
*l .mtm,mtmmac
TSKID = .MTM
*st ,auf=users.auf,atf=null:,term=(t020:,t022:)
*10:24:14 .MTM:OS/32 TERMINAL MONITOR 08-01
*
```

LICENSE C-838-

5. telnet to your host and signon.

```
#telnet cel2 1026
Connected to cel2.
Escape character is '^]'.
```

Connected to the Interdata 32b simulator PAS device, line 0

```
*signon user1,25,user1
*d f
FILE(S) NOT FOUND ON SYS
*signoff
ELAPSED TIME=20
TIME OFF=02/16/16 10:34:02
OS/32 TERMINAL MONITOR 08-01
*
```

6. For a nice description of OS/32 and MTM commands, see:

http://eprints.whiterose.ac.uk/76197/1/report_180.pdf

At this point the “proper” thing to do would be to install various software packages using MTM but that is beyond the scope of this document.

5 CAL, Fortran, and Pascal

CAL, Fortran, and Pascal can be used without being fully installed with MTM. Here are some examples.

5.1 “HELLO WORLD” in CAL32

The spacing in the .CAL file is important! I don’t know what the exact spacing requirements are, but this worked.

The “os32st” directory on bitsavers contains a “Program Reference” manual that proved helpful in writing this.

Enter the following program into hello.cal:

```

                SVC    1,SAY
                SVC    3,0
                ALIGN  ADC
SAY             DB     X'28'
                DB     X'00'
                DS      2
                DC      A(SAY1)
                DC      A(SAY2)
                DAS     2
SAY1            DC     C'HELLO WORLD '
SAY2            EQU    *-1
                END
```

Assemble and link:

```
*^E
Simulation stopped, PC: 22C9A (BAL R12,19064)
sim> as lpt pr01
LPT: creating new file
sim> c
l cal32
TSKID = CAL32
*as 1,hello.cal
al helloc.obj,in,126
*as 2,helloc.obj
*as 3,pr:
*st
*13:27:19  CAL32:CAL/32 03-338R01-01
*13:27:19  CAL32:          NO ERRORS   TABLE SPACE USED :      1K
*13:27:19  CAL32:END OF TASK      0
*l link
TSKID = LINK
*st
*13:27:32  LINK:PERKIN-ELMER OS/32 LINKAGE EDITOR 03-242 R01-03
*LINK>in helloc.obj
*LINK>bu
```

```

*13:27:40 LINK:ENTER FILE DESCRIPTOR FOR IMAGE*LINK>helloc
*13:27:46 LINK:MAP?*LINK>no
*LINK>end
*13:27:53 LINK:END OF TASK      0
*1 helloc
TSKID = HELLO
*as 0,con:
*st
*13:28:15 HELLO:HELLO WORLD
*13:28:15 HELLO:END OF TASK      0
*

```

5.2 Fortran “HELLO WORLD”

Although the Fortran tape includes an install program, it requires that the “OS/32 R06-02 Package” is installed. Since that is not available, manual installation and usage is necessary.

1. Unpack 04-101M31R09_FortranVII.tap to a temporary disk.
2. Copy to SYS:
F7D51.TSK, F7D51.ERR, F7LIB51.OBJ, F7RTL51.ERR, PEM51.OBJ,
F7O51.TSK, F7ZO51.ERR, F7LIB51A.OBJ, PEM51A.OBJ
F7D51 is the Fortran VII Development Compiler
F7O51 is the Fortran VII Optimizing Compiler
3. Put a program into hello.ftn

```

C      HELLO WORLD
      WRITE(0,1000)
1000   FORMAT(12H HELLO WORLD )
      END

```

4. Compile
Using F7D51

```

*al hellof.obj,in,126
*l f7d51
TSKID = F7D51
*as 1,hello.ftn
*as 2,hellof.obj
*as 3,pr:

```



```

*st
*21:51:15    F7D51:FORTTRAN-VIID R05-01.00
*21:51:15    F7D51:.MAIN          NO  ERROR(S)      TABLE SPACE USED:      1 K
*21:51:15    F7D51:END OF TASK      0
*
```

Using F7O51

```

*al hellof.obj,in,126
*l f7o51
TSKID = F7O51
*as 2,hellof.obj
*as 1,hello.ftn
*as 3,pr:
*te 4,in,80/5
*te 0,in,80/5
*te 8,co,2000
*st
*12:19:29    F7O51:FORTTRAN-VIIO R05-01.00
*12:19:29    F7O51:*** INTERNAL TABLES PAGING TO DISC ***
*12:19:29    F7O51:.MAIN          NO ERRORS.      TABLE SPACE USED:      22.75K
*12:19:29    F7O51:END OF TASK      0
*
```

5. Link

```

*l link
TSKID = LINK
*st
*21:51:59    LINK:PERKIN-ELMER OS/32 LINKAGE EDITOR 03-242 R01-03
*LINK>in hellof
*LINK>lib f7lib51, pem51
*LINK>bu
*21:52:34    LINK:ENTER FILE DESCRIPTOR FOR IMAGE*LINK>hellof
*21:52:40    LINK:MAP?*LINK>no
*LINK>end
*21:52:59    LINK:END OF TASK      0
*
```

6. Run. The program needs some memory so add a “segsize increment.”

```

*1 hellof,5
TSKID = HELLOF
*as 0,con:
*st
*22:03:36    HELLOF:HELLO WORLD
*22:03:36    HELLOF:END OF TASK      0
*

```

5.3 Pascal “Hello World”

1. Unpack OS32.pascal.tap onto a temporary disk. There will be errors.
2. Copy PASCAL.TSK, PASRTL.OBJ, PRIMES.PAS to SYS:
3. Enter a Pascal program.

```

program hello(output);
begin
    writeln('Hello World')
end.

```

4. Compile.

```

*1 pascal,64
TSKID = PASCAL
*as 0,con:
*as 1,hello.pas
*as 7,hello.obj
*as 2,pr:          ; list file/device
*as 6,pr:          ; assembly/map file/device
*te 3,in,512
*te 4,in,512
*st
*09:39:33    PASCAL:PERKIN-ELMER PASCAL R01-00          LICENSE C-0428
*09:39:33    PASCAL:END OF TASK      0

```

5. Link.

```

*1 link
TSKID = LINK
*st
*09:39:41    LINK:PERKIN-ELMER OS/32 LINKAGE EDITOR 03-242 R01-03      ^@
*LINK>in hello

```

```

*LINK>lib pasrtl
*LINK>bu
*09:39:55    LINK:ENTER FILE DESCRIPTOR FOR IMAGE*LINK>hellop
*09:40:01    LINK:MAP?*LINK>no
*LINK>end
*09:40:05    LINK:END OF TASK      0

```

6. Run.

```

*l hellop,64
TSKID = HELLOP
*as 0,con:
*09:40:28    HELLOP:Hello World
*09:40:28    HELLOP:END OF TASK    0

```

7. primes.pas needs a bit more memory, so use “l primes,100” to run it.

6 Notes

6.1 Disks

```

DSC4  dm0  msm80 67MB
DSC3  dm1  msm80 67MB
DSC5  dm2  msm80 67MB

```

```

DSC1  dp0  5440 10MB
DSC2  dp0

```

DSC1 (removable) and DSC2 (fixed) are in the same simh file. simh does not allow for a separate removable part.

I’ve never gotten a “2315” disk to work.

The nomenclature “2315” and “5440” is misleading since it refers to the cartridge and not the device.

6.2 Memory

“set cpu 1m” doesn’t work. simh defaults to 1 MB, so unless you want less memory, don’t set it.

6.3 Transferring Files to id32

Linux telnet is not able to “send” files so a cut and paste method must be used. Use the BUILD command to enter the data.

To automate things a little, use netcat. Since id32 is relatively slow, use a delay interval to give it a chance to keep up. Also, the file must use CR/LF to end lines.

For example, this will create the file “file05.ftn” on user1’s account:

```
# cat file05
```

```
signon user1,25,user1
build file05.ftn
C      HELLO WORLD
      WRITE(0,1000)
      1000  FORMAT(12H HELLO WORLD )
      END
endb
signoff
```

```
# cat file05 | todos | nc -t -i 1 cel2 1026
```

```
*
*signon user1,25,user1
*build file05.ftn
B>C      HELLO WORLD
B>      WRITE(0,1000)
B>1000  FORMAT(12H HELLO WORLD )
B>      END
B>endb
*signoff
ELAPSED TIME=7
TIME OFF=02/25/16  11:02:06
OS/32 TERMINAL MONITOR 08-01
*^C
#
```