

PDP-8 Simulator Usage

01-May-2021

COPYRIGHT NOTICE

The following copyright notice applies to the SIMH source, binary, and documentation:

Original code published in 1993-2021, written by Robert M Supnik
Copyright (c) 1993-2021, Robert M Supnik

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL ROBERT M SUPNIK BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Robert M Supnik shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from Robert M Supnik.

1	Simulator Files	3
2	PDP-8 Features	3
2.1	CPU	5
2.2	TSC8-75 ETOS Timeshare Control (TSC).....	6
2.3	FPP8A Floating Point Unit (FPP).....	6
2.4	Programmed I/O Devices	7
2.4.1	PC8E Paper Tape Reader (PTR).....	7
2.4.2	PC8E Paper Tape Punch (PTP).....	7
2.4.3	KL8E Terminal Input (TTI).....	8
2.4.4	KL8E Terminal Output (TTO)	8
2.4.5	LE8E Line Printer (LPT)	9
2.4.6	DK8E Line-Frequency Clock (CLK).....	9
2.4.7	KL8JA Additional Terminals (TTIX, TTOX).....	10
2.4.8	TD8E/TU56 DEctape (TD).....	11
2.4.9	TA8E/TA60 Cassette Tape (CT)	12
2.5	Moving Head Disks.....	13
2.5.1	RK8E Cartridge Disk (RK).....	13
2.5.2	RL8A Cartridge Disk (RL).....	13
2.6	RX8E/RX01, RX28/RX02 Floppy Disk (RX)	14
2.7	Fixed Head Disks.....	15
2.7.1	RF08/RS08 Fixed Head Disk (RF)	15
2.7.2	DF32/DS32 Fixed Head Disk (RF)	16
2.8	TC08/TU56 DEctape (DT)	17
2.9	TM8E Magnetic Tape (MT).....	17
3	Symbolic Display and Input.....	18

This memorandum documents the PDP-8 simulator.

1 Simulator Files

```
sim/          scp.h
              sim_console.h
              sim_defs.h
              sim_fio.h
              sim_rev.h
              sim_sock.h
              sim_tape.h
              sim_timer.h
              sim_tmxr.h
              scp.c
              sim_console.c
              sim_fio.c
              sim_sock.c
              sim_tape.c
              sim_timer.c
              sim_tmxr.c

sim/pdp8/     pdp8_defs.h
              pdp8_cpu.c
              pdp8_ct.c
              pdp8_df.c
              pdp8_dt.c
              pdp8_fpp.c
              pdp8_lp.c
              pdp8_mt.c
              pdp8_pt.c
              pdp8_rf.c
              pdp8_rk.c
              pdp8_rl.c
              pdp8_rx.c
              pdp8_sys.c
              pdp8_td.c
              pdp8_tsc.c
              pdp8_tt.c
              pdp8_ttx.c
```

2 PDP-8 Features

The PDP-8 simulator is configured as follows:

device names(s)	simulates
CPU	PDP-8/E CPU with 4KW-32KW of memory
-	KE8E extended arithmetic element (EAE)
-	KM8E memory management and timeshare control
TSC	TSC8-75 ETOS operating system timeshare control
FPP	FPP8A floating point unit
PTR, PTP	PC8E paper tape reader/punch

TTI, TTO	KL8E console terminal
TTIX, TTOX	KL8JA additional terminals, up to 16
LPT	LE8E line printer
CLK	DK8E line frequency clock (also PDP-8/A compatible)
RK	RK8E/RK05 cartridge disk controller with four drives
RF	RF08/RS08 fixed head disk controller with 1-4 platters
DF	DF32/DS32 fixed head disk controller with 1-4 platters
RL	RL8A/RL01 cartridge disk controller with four drives
RX	RX8E/RX01, RX28/RX02 floppy disk controller with two drives
DT	TC08/TU56 DECTape controller with eight drives
TD	TD8E/TU56 DECTape controller with two drives
MT	TM8E/TU10 magnetic tape controller with eight drives
CT	TA8E/TU60 cassette tape controller with two drives

Most devices can be disabled or enabled, by the commands:

```
SET <dev> DISABLED
SET <dev> ENABLED
```

The simulator allows most device numbers to be changed, by the command:

```
SET <dev> DEV=<number>
```

The PDP-8 can support only one of the set {DF32, RF08, RL8A} using the default device numbers, since they all use device numbers 60-61. The default is the RF08. To change the disk at device numbers 60-61:

```
SET RF DISABLED           disable RF08
SET DF ENABLED, or       enable DF32
SET RL ENABLED           enable RL8A
```

The PDP-8 can only support one of the set {TC08, TD8E} using the default device numbers, since both use device number 77. The default is the TC08. To change the DECTape controller to the TD8E:

```
SET DT DISABLED           disable TC08
SET TD ENABLED           enable TD8E
```

The PDP-8 can only support one of the set {TM8E, TA8E} using the default device numbers, since both use device number 70. The default is the TM8E. To change the device at device number 70:

```
SET MT DISABLED           disable TM8E
SET CT ENABLED           enable TA8E
```

Alternately, the device conflict can be eliminated by changing device numbers:

```
SET RL DEV=50
SET RL ENA
SET TD DEV=74
SET TD ENA
SET CT DEV=73
SET CT ENA
```

However, devices can only be BOOTed with their default device numbers.

The PDP-8 simulator implements several unique stop conditions:

- If an undefined instruction (unimplemented IOT or OPR) is decoded, and STOP_INST is set
- If a simulated DECTape runs off the end of its reel

The LOAD command supports both RIM format and BIN format tapes. If the file extension is .RIM, or the r switch is specified with LOAD, the file is assumed to be RIM format; if the file extension is not .RIM, or the -b switch is specified, the file is assumed to be BIN format.

2.1 CPU

The only CPU options are the presence of the EAE and the size of main memory; the memory extension and time-share control is always included, even if memory size is 4K.

```

SET CPU EAE           enable EAE
SET CPU NOEAE        disable EAE
SET CPU 4K           set memory size = 4K
SET CPU 8K           set memory size = 8K
SET CPU 12K          set memory size = 12K
SET CPU 16K          set memory size = 16K
SET CPU 20K          set memory size = 20K
SET CPU 24K          set memory size = 24K
SET CPU 28K          set memory size = 28K
SET CPU 32K          set memory size = 32K

```

If memory size is being reduced, and the memory being truncated contains non-zero data, the simulator asks for confirmation. Data in the truncated portion of memory is lost. Initial memory size is 32K.

CPU registers include the visible state of the processor as well as the control registers for the interrupt system.

name	size	comments
PC	15	program counter, including IF as high 3 bits
AC	12	accumulator
MQ	12	multiplier-quotient
L	1	link
SR	12	front panel switches
IF	3	instruction field
DF	3	data field
IB	3	instruction field buffer
SF	7	save field
UF	1	user mode flag
UB	1	user mode buffer
SC	5	EAE shift counter
GTF	1	EAE greater than flag
EMODE	1	EAE mode (0 = A, 1 = B)
ION	1	interrupt enable
ION_DELAY	1	interrupt enable delay for ION
CIF_DELAY	1	interrupt enable delay for CIF
PWR_INT	1	power fail interrupt
UF_INT	1	user mode violation interrupt
INT	15	interrupt pending flags
DONE	15	device done flags
ENABLE	15	device interrupt enable flags
PCQ[0:63]	15	PC prior to last JMP, JMS, or interrupt; most recent PC change first
STOP_INST	1	stop on undefined instruction

The CPU attempts to detect when the simulator is idle. When idle, the simulator does not use any resources on the host system. Idle detection is controlled by the `SET IDLE` and `SET NOIDLE` commands:

```
SET CPU IDLE           enable idle detection
SET CPU NOIDLE        disable idle detection
```

Idle detection is disabled by default. At present, the CPU is considered idle if it is executing a `KSF/JMP *-1` loop with interrupts disabled (OS/8, DMS-8) or a `JMP *` loop (TSS/8).

The CPU can maintain a history of the most recently executed instructions. This is controlled by the `SET CPU HISTORY` and `SHOW CPU HISTORY` commands:

```
SET CPU HISTORY       clear history buffer
SET CPU HISTORY=0     disable history
SET CPU HISTORY=n     enable history, length = n
SHOW CPU HISTORY      print CPU history
SHOW CPU HISTORY=n    print first n entries of CPU history
```

The maximum length for the history is 65536 entries.

2.2 TSC8-75 ETOS Timeshare Control (TSC)

ETOS is a timeshared operating system for the PDP-8, providing multiple virtual OS/8 environments for up to 32 users. It requires a special timeshare control option, the TSC8-75. The TSC8-75 is normally disabled; to run ETOS, it must be enabled with the command:

```
SET TSC ENABLED
```

The TSC8-75 implements these registers:

name	size	comments
IR	12	most recently trapped instruction
PC	12	PC of most recently trapped instruction
CDF	1	1 if trapped instruction is CDF, 0 otherwise
ENB	1	interrupt enable flag
INT	1	interrupt pending flag

Except for operation of ETOS, the TSC8-75 should be left disabled.

2.3 FPP8A Floating Point Unit (FPP)

The floating point unit (FPP) is an add-on device that provides floating point capabilities. It operates as a coprocessor to the main CPU, with its own program counter and instruction set. The FPP8A is normally disabled; to use it, it must be enabled with the command:

```
SET FPP ENABLED
```

The FPP8A implements these registers:

name	size	comments
------	------	----------

FSPACE	12	floating AC exponent
FPAC0..4	12	floating AC fraction words 0..4
CMD	12	FPP command register
STA	12	FPP status register
APTA	15	APT address pointer
APTSVF	3	APT save field
FPC	15	FPP program counter
BRA	15	base register pointer
XRA	15	index register pointer
OPA	15	operand pointer
SSF	12	single step flag
LASTLOCK	12	last lock bit
FLAG	1	FPP flag

Except for environments that explicitly support it, the FPP8A should be left disabled.

2.4 Programmed I/O Devices

2.4.1 PC8E Paper Tape Reader (PTR)

The paper tape reader (PTR) reads data from a disk file. The POS register specifies the number of the next data item to be read. Thus, by changing POS, the user can backspace or advance the reader.

The paper tape reader supports the `BOOT` command. `BOOT PTR` copies the RIM loader into memory and starts it running.

The paper tape reader implements these registers:

name	size	comments
BUF	8	last data item processed
DONE	1	device done flag
ENABLE	1	interrupt enable flag
INT	1	interrupt pending flag
POS	32	position in the input file
TIME	24	time from I/O initiation to interrupt
STOP_IOE	1	stop on I/O error

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	out of tape
end of file	1	report error and stop
	0	out of tape
OS I/O error	x	report error and stop

2.4.2 PC8E Paper Tape Punch (PTP)

The paper tape punch (PTP) writes data to a disk file. The POS register specifies the number of the next data item to be written. Thus, by changing POS, the user can backspace or advance the punch.

The paper tape punch implements these registers:

name	size	comments
BUF	8	last data item processed
DONE	1	device done flag
ENABLE	1	interrupt enable flag
INT	1	interrupt pending flag
POS	32	position in the output file
TIME	24	time from I/O initiation to interrupt
STOP_IOE	1	stop on I/O error

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	out of tape
OS I/O error	x	report error and stop

2.4.3 KL8E Terminal Input (TTI)

The terminal interfaces (TTI, TTO) can be set to one of four modes, KSR, 7B, 7P, or 8B:

mode	input characters	output characters
KSR	lower case converted to upper case, high-order bit set	lower case converted to upper case, high-order bit cleared, non-printing characters suppressed
7P	high-order bit cleared	high-order bit cleared, non-printing characters suppressed
7B	high-order bit cleared	high-order bit cleared
8B	no changes	no changes

The default mode is KSR.

The terminal input (TTI) polls the console keyboard for input. It implements these registers:

name	size	comments
BUF	8	last data item processed
DONE	1	device done flag
ENABLE	1	interrupt enable flag
INT	1	interrupt pending flag
POS	32	number of characters input
TIME	24	input polling interval

The terminal input is normally polled synchronously with the real-time clock. To avoid data loss, a poll is scheduled 'TIME' instructions after the CPU reads a character.

2.4.4 KL8E Terminal Output (TTO)

The terminal output (TTO) writes to the simulator console window. It implements these registers:

name	size	comments
BUF	8	last data item processed
DONE	1	device done flag
ENABLE	1	interrupt enable flag
INT	1	interrupt pending flag
POS	32	number of characters output
TIME	24	time from I/O initiation to interrupt

2.4.5 LE8E Line Printer (LPT)

The line printer (LPT) writes data to a disk file. The POS register specifies the number of the next data item to be read or written. Thus, by changing POS, the user can backspace or advance the printer.

The line printer implements these registers:

name	size	comments
BUF	8	last data item processed
ERR	1	error status flag
DONE	1	device done flag
ENABLE	1	interrupt enable flag
INT	1	interrupt pending flag
POS	32	position in the output file
TIME	24	time from I/O initiation to interrupt

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	out of paper
OS I/O error	x	report error and stop

2.4.6 DK8E Line-Frequency Clock (CLK)

The real-time clock (CLK) frequency can be adjusted as follows:

SET CLK 60HZ	set frequency to 60Hz
SET CLK 50HZ	set frequency to 50Hz
SET CLK NORMAL	set clock to calibrated timing
SET CLK DIAG	set clock to fixed timing

The defaults are 60Hz, calibrated timing.

The clock implements these registers:

name	size	comments
DONE	1	device done flag
ENABLE	1	interrupt enable flag
INT	1	interrupt pending flag
TIME	24	clock interval

The real-time clock autocalibrates; the clock interval is adjusted up or down so that the clock tracks actual elapsed time.

2.4.7 KL8JA Additional Terminals (TTIX, TTOX)

The simulator supports 1 to 16 additional terminals, with an initial default of 4 lines. The additional terminals consist of two independent devices, TTIX and TTOX. The entire set is modeled as a terminal multiplexer, with TTIX as the master controller. The number of lines is specified with a `SET` command:

```
SET TTIX LINES=n          set number of additional lines to n [1-16]
```

The `ATTACH` command specifies the port to be used:

```
ATTACH TTIX <port>      set up listening port
```

where port is a decimal number between 1 and 65535 that is not being used for other TCP/IP activities. The additional terminals are disabled by default.

The additional terminals can be set to one of four modes: UC, 7P, 7B, or 8B.

mode	input characters	output characters
UC	lower case converted to upper case, high-order bit cleared	lower case converted to upper case, high-order bit cleared, non-printing characters suppressed
7P	high-order bit cleared	high-order bit cleared, non-printing characters suppressed
7B	high-order bit cleared	high-order bit cleared
8B	no changes	no changes

The default mode is UC. Finally, each line supports output logging. The `SET TTOXn LOG` command enables logging on a line:

```
SET TTOXn LOG=filename   log output of line n to filename
```

The `SET TTOXn NOLOG` command disables logging and closes the open log file, if any.

Once TTIX is attached and the simulator is running, the terminals listen for connections on the specified port. They assume that the incoming connections are Telnet connections. The connections remain open until disconnected either by the Telnet client, a `SET TTIX DISCONNECT` command, or a `DETACH TTIX` command.

Other special commands:

```
SHOW TTIX CONNECTIONS   show current connections
SHOW TTIX STATISTICS    show statistics for active connections
SET TTOXn DISCONNECT    disconnects the specified line.
```

The input device (TTIX) implements these registers:

name	size	comments
BUF[0:15]	8	input buffer, lines 0 to 15
DONE	16	device done flags (line 0 rightmost)
ENABLE	16	interrupt enable flag

TIME	24	initial polling interval
------	----	--------------------------

The input device is normally polled synchronously with the real-time clock. To avoid data loss, a poll is scheduled 'TIME' instructions after the CPU reads a character from any line.

The output device (TTOX) implements these registers:

name	size	comments
BUF[0:15]	8	last data item processed, lines 0-15
DONE	16	device done flag (line 0 rightmost)
ENABLE	16	interrupt enable flag
TIME[0:16]	24	time from I/O initiation to interrupt, lines 0-15

The additional terminals do not support save and restore. All open connections are lost when the simulator shuts down or TTX is detached.

2.4.8 TD8E/TU56 DECtape (TD)

The TD8E is a programmed I/O, non-interrupt controller, supporting two DECtape drives (0 and 1). The TD8E simulator puts a high burden on the host processor, because tape activity is simulated a line (3b) at a time. Unless the PDP-8 software requires the TD8E, the TC08 should be used to simulate DECtapes. The TD8E is disabled by default.

TD8E options include the ability to make units write enabled or write locked.

SET TDn LOCKED	set unit n write locked
SET TDn WRITEENABLED	set unit n write enabled

Units can also be set `ENABLED` or `DISABLED`. The TD8E supports the `BOOT` command, but only for unit 0.

The TD8E supports PDP-8 format, PDP-11 format, and 18b format DECtape images. `ATTACH` assumes the image is in PDP-8 format; the user can force other choices with switches:

-s	PDP-11 format
-f	18b format
-a	autoselect based on file on file size

The TD8E controller is a data-only simulator; the timing and mark track, and block header and trailer, are not stored. Thus, read always produces standard values for mark track, header, and trailer words, and write throws mark track, header, and trailer words into the bit bucket.

The TD8E controller implements these registers:

name	size	comments
TDCMD	4	command register
TDDAT	12	data register
TDMTK	6	mark track register
TDSLF	1	single line flag
TDQLF	1	quad line flag
TDTME	1	timing error flag
TDQL	2	quad line counter
LTIME	31	time between lines

DCTIME	31	time to decelerate to a full stop
POS[0:1]	32	position, in lines, units 0 and 1
STATT[0:1]	18	unit state, units 0 and 1
STOP_OFFR	1	stop on off-reel error

The LTIME parameter should not be changed, or OS/8 may fail to run correctly. The DCTIME parameter should always be at least 100 times greater than LTIME. Acceleration time is 75% of deceleration time.

2.4.9 TA8E/TA60 Cassette Tape (CT)

The TA8E is a programmed I/O controller supporting two cassette drives (0 and 1). The TA8E can be used with the MCP/IP program under OS/8, and with the CAPS-8 operating system. Cassettes are simulated as magnetic tapes with a fixed capacity (93,000 characters). The tape format is always SimH standard. The TA8E is disabled by default.

TA8E options include the ability to make units write enabled or write locked.

SET CTn LOCKED	set unit n write locked
SET CTn WRITEENABLED	set unit n write enabled

Units cannot be set ENABLED or DISABLED. The TA8E supports the BOOT command, but only for CAPS-8, and only for unit 0.

The TA8E controller implements these registers:

name	size	comments
CTSRA	8	status register A
CTSRB	8	status register B
CTDB	8	data buffer
CTDF	1	data flag
RDY	1	ready flag
WLE	1	write lock error
WRITE	1	TA60 write operation flag
INT	1	interrupt request
BPTR	17	buffer pointer
BLNT	17	buffer length
STIME	24	operation start time
CTIME	24	character latency
STOP_IOE	1	stop on I/O errors flag
POS[0:1]	32	position, units 0-1

Error handling is as follows:

error	processed as
not attached	tape not ready; if STOP_IOE, stop
end of file	bad tape
OS I/O error	CRC error; if STOP_IOE, stop

2.5 Moving Head Disks

2.5.1 RK8E Cartridge Disk (RK)

RK8E options include the ability to make units write enabled or write locked:

```
SET RKn LOCKED           set unit n write locked
SET RKn WRITEENABLED     set unit n write enabled
```

Units can also be set ENABLED or DISABLED. The RK8E supports the BOOT command.

The RK8E implements these registers:

name	size	comments
RKSTA	12	status
RKCMD	12	disk command
RKDA	12	disk address
RKMA	12	current memory address
BUSY	1	control busy flag
INT	1	interrupt pending flag
STIME	24	seek time, per cylinder
RTIME	24	rotational delay
STOP_IOE	1	stop on I/O error

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	disk not ready
end of file	x	assume rest of disk is zero
OS I/O error	x	report error and stop

2.5.2 RL8A Cartridge Disk (RL)

RL8A options include the ability to make units write enabled or write locked:

```
SET RLn LOCKED           set unit n write locked
SET RLn WRITEENABLED     set unit n write enabled
```

Units can also be set ENABLED or DISABLED. The RL8A supports the BOOT command, but only for unit 0.

The RL8A implements these registers:

name	size	comments
RLCSA	12	control/status A
RLCSB	12	control/status B
RLMA	12	memory address
RLWC	12	word count
RLSA	6	sector address
RLER	12	error flags

RLSI	16	silos top word
RLSI1	16	silos second word
RLSI2	16	silos third word
RLSIL	1	silos read left/right flag
INT	1	interrupt request
DONE	1	done flag
ERR	1	composite error flag
STIME	1	seek time, per cylinder
RTIME	1	rotational delay
STOP_IOE	1	stop on I/O error

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	disk not ready
end of file	x	assume rest of disk is zero
OS I/O error	x	report error and stop

2.6 RX8E/RX01, RX28/RX02 Floppy Disk (RX)

The RX can be configured as an RX8E with two RX01 drives, or an RX28 with two RX02 drives:

```
SET RX RX8E           set controller to RX8E/RX01
SET RX RX28           set controller to RX28/RX02
```

The controller is set to the RX8E by default. The RX28 is not backwards-compatible with the RX8E and will not work with the standard OS/8 V3D floppy disk driver.

RX8E options include the ability to set units write enabled or write locked:

```
SET RXn LOCKED        set unit n write locked
SET RXn WRITEENABLED  set unit n write enabled
```

RX28 options include, in addition, the ability to set the unit density to single density, double density, or autosized; autosizing is the default:

```
SET RXn SINGLE        set unit n single density
SET RXn DOUBLE        set unit n double density
SET RXn AUTOSIZE      set unit n autosize
```

The RX8E and RX28 support the `BOOT` command.

The RX8E and RX28 implement these registers:

name	size	comments
RXCS	12	status
RXDB	12	data buffer
RXES	12	error status
RXTA	8	current track
RXSA	8	current sector
STAPTR	4	controller state

BUFPTR	8	buffer pointer
INT	1	interrupt pending flag
DONE	1	device done flag
ENABLE	1	interrupt enable flag
TR	1	transfer ready flag
ERR	1	error flag
CTIME	24	command completion time
STIME	24	seek time, per track
XTIME	24	transfer ready delay
STOP_IOE	1	stop on I/O error
SBUF[0:255]	8	sector buffer array

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	disk not ready

RX01 and RX02 data files are buffered in memory; therefore, end of file and OS I/O errors cannot occur.

2.7 Fixed Head Disks

With default device addressing, either the RF08 or the DF32 can be present in a configuration, but not both.

2.7.1 RF08/RS08 Fixed Head Disk (RF)

RF08 options include the ability to set the number of platters to a fixed value between 1 and 4, or to autosize the number of platters:

SET RF 1P	one platter (256K)
SET RF 2P	two platters (512K)
SET RF 3P	three platters (768K)
SET RF 4P	four platters (1024K)
SET RF AUTOSIZE	autosized on ATTACH

The default is one platter.

The RF08 implements these registers:

name	size	comments
STA	12	status
DA	20	current disk address
MA	12	memory address (in memory)
WC	12	word count (in memory)
WLK	32	write lock switches
INT	1	interrupt pending flag
DONE	1	device done flag
TIME	24	rotational delay, per word
BURST	1	burst flag
STOP_IOE	1	stop on I/O error

The RF08 supports the `BOOT` command. The default bootstrap is for OS/8. To bootstrap the 4K Disk Monitor, use the `BOOT -D RF` command.

The RF08 is a three-cycle data break device. If BURST = 0, word transfers are scheduled individually; if BURST = 1, the entire transfer occurs in a single data break.

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	disk not ready

RF08 data files are buffered in memory; therefore, end of file and OS I/O errors cannot occur.

2.7.2 DF32/DS32 Fixed Head Disk (RF)

DF32 options include the ability to set the number of platters to a fixed value between 1 and 4, or to autosize the number of platters:

SET DF 1P	one platter (32K)
SET DF 2P	two platters (64K)
SET DF 3P	three platters (98K)
SET DF 4P	four platters (128K)
SET DF AUTOSIZE	autosized on ATTACH

The default is one platter.

The DF32 implements these registers:

name	size	comments
STA	12	status, disk and memory address extension
DA	12	low order disk address
MA	12	memory address (in memory)
WC	12	word count (in memory)
WLK	16	write lock switches
INT	1	interrupt pending flag
DONE	1	device done flag
TIME	24	rotational delay, per word
BURST	1	burst flag
STOP_IOE	1	stop on I/O error

The DF32 supports the `BOOT` command. The default bootstrap is for OS/8. To bootstrap the 4K Disk Monitor, use the `BOOT -D DF` command.

The DF32 is a three-cycle data break device. If BURST = 0, word transfers are scheduled individually; if BURST = 1, the entire transfer occurs in a single data break.

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	disk not ready

DF32 data files are buffered in memory; therefore, end of file and OS I/O errors cannot occur.

2.8 TC08/TU56 DECtape (DT)

DT implements the TC08 DECtape controller and TU56 drives. TC08 options include the ability to make units write enabled or write locked.

```
SET DTn LOCKED           set unit n write locked
SET DTn WRITEENABLED    set unit n write enabled
```

Units can also be set `ENABLED` or `DISABLED`. The TC08 supports the `BOOT` command, but only for unit 0.

The TC08 supports PDP-8 format, PDP-11 format, and 18b format DECtape images. `ATTACH` assumes the image is in PDP-8 format; the user can force other choices with switches:

```
-s           PDP-11 format
-f           18b format
-a           autoselect based on file on file size
```

The TC08 controller is a data-only simulator; the timing and mark track, and block header and trailer, are not stored. Thus, the `WRITE TIMING AND MARK TRACK` function is not supported; the `READ ALL` function always returns the hardware standard block header and trailer; and the `WRITE ALL` function dumps non-data words into the bit bucket.

The DECtape controller implements these registers:

name	size	comments
DTSA	12	status register A
DTSB	12	status register B
INT	1	interrupt pending flag
ENB	1	interrupt enable flag
DTF	1	DECtape flag
ERF	1	error flag
CA	12	current address (memory location 7754)
WC	12	word count (memory location 7755)
LTIME	31	time between lines
DCTIME	31	time to decelerate to a full stop
SUBSTATE	2	read/write command substate
POS[0:7]	32	position, in lines, units 0 to 7
STATT[0:7]	31	unit state, units 0 to 7
STOP_OFFR	1	stop on off-reel error

It is critically important to maintain certain timing relationships among the DECtape parameters, or the DECtape simulator will fail to operate correctly.

- LTIME must be at least 6
- DCTIME needs to be at least 100 times LTIME

Acceleration time is set to 75% of deceleration time.

2.9 TM8E Magnetic Tape (MT)

Magnetic tape options include the ability to make units write enabled or write locked.

```
SET MTn LOCKED           set unit n write locked
SET MTn WRITEENABLED    set unit n write enabled
```

Magnetic tape units can be set to a specific reel capacity in MB, or to unlimited capacity:

```
SET MTn CAPAC=m          set unit n capacity to m MB (0 = unlimited)
SHOW MTn CAPAC           show unit n capacity in MB
```

Units can also be set ENABLED or DISABLED.

The magnetic tape controller implements these registers:

name	size	comments
CMD	12	command
FNC	12	function
CA	12	memory address
WC	12	word count
DB	12	data buffer
STA	12	main status
STA2	6	secondary status
DONE	1	device done flag
INT	1	interrupt pending flag
STOP_IOE	1	stop on I/O error
TIME	24	record delay
UST[0:7]	24	unit status, units 0 to 7
POS[0:7]	32	position, units 0 to 7

Error handling is as follows:

error	processed as
not attached	tape not ready; if STOP_IOE, stop
end of file	bad tape
OS I/O error	parity error; if STOP_IOE, stop

3 Symbolic Display and Input

The PDP-8 simulator implements symbolic display and input. Display is controlled by command line switches:

```
-a          display as ASCII character
-c          display as two packed sixbit characters
-t          display as two packed TSS/8 sixbit characters
-m          display instruction mnemonics
```

Input parsing is controlled by the first character typed in or by command line switches:

' or -a	ASCII character
" or -c	two packed sixbit characters
# or -t	two packed TSS/8 sixbit characters
alphabetic	instruction mnemonic
numeric	octal number

Instruction input uses standard PDP-8 assembler syntax. There are four instruction classes: memory reference, IOT, field change, and operate.

Memory reference instructions have the format

```
memref {I} {C/Z} address
```

where I signifies indirect, C a current page reference, and Z a zero page reference. The address is an octal number in the range 0 - 07777; if C or Z is specified, the address is a page offset in the range 0 - 177. Normally, C is not needed; the simulator figures out from the address what mode to use. However, when referencing memory outside the CPU (eg, disks), there is no valid PC, and C must be used to specify current page addressing.

IOT instructions consist of single mnemonics, eg, KRB, TLS. IOT instructions may be or'd together

```
iot iot iot...
```

The simulator does not check the legality of the proposed combination. IOT's for which there is no opcode may be specified as IOT n, where n is an octal number in the range 0 - 0777.

Field change instructions (CIF, CDF) have the format

```
fldchg field
```

where field is an octal number in the range 0 - 7. Field change instructions may be or'd together.

Operate instructions have the format

```
opr opr opr...
```

The simulator does not check the legality of the proposed combination. EAE mode A and B mnemonics may be specified regardless of the EAE mode. The operands for MUY and DVI must be deposited explicitly.