

XDS Sigma 32b Simulator Usage

27-Mar-2016

COPYRIGHT NOTICE

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

Original code published in 1993-2016, written by Robert M Supnik
Copyright (c) 1993-2016, 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	XDS Sigma 32b Features	3
2.1	Central Processor (CPU)	4
2.2	Memory Map (MAP).....	5
2.3	Interrupt System (INT)	6
2.4	Channels (CHANA .. CHANH)	6
2.5	7012 Console Teletype (TT)	7
2.6	7060 Paper Tape Reader (PT)	8
2.7	7440 or 7450 Line Printer (LP).....	8
2.8	Real-Time Clock (RTC)	9
2.9	Character-Oriented Communications Subsystem (MUX)	10
2.10	7240, 7270, 7260, 7265, 7275, and T3281 Moving Head Disk Controllers (DPA, DPB)	11
2.11	7250 Cartridge Disk Controller (DK)	12
2.12	7211/7212 or 7231/7232 Fixed Head Disk (RAD).....	13
2.13	723X Nine-Track Magnetic Tape (MT).....	13
3	Symbolic Display and Input.....	14

This memorandum documents the SDS 940 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/sds/      sigma_defs.h
              sigma_io_defs.h
              sigma_cis.c
              sigma_coc.c
              sigma_cpu.c
              sigma_dk.c
              sigma_dp.c
              sigma_fp.c
              sigma_io.c
              sigma_lp.c
              sigma_map.c
              sigma_mt.c
              sigma_pt.c
              sigma_rad.c
              sigma_rtc.c
              sigma_sys.c
              sigma_tt.c
```

2 XDS Sigma 32b Features

The XDS Sigma 32b simulator is configured as follows:

device name(s)	simulates
CPU	XDS Sigma 5, 6, 7, 7 "big mem", 8, 9, or XDS 550, 560
CHANA..CHANH	I/O channels, up to 8
PT	7060 paper tape reader/punch
TT	7012 console Teletype
LP	7440 or 7450 line printer
RTC	real-time clock
MUX	character-oriented communications subsystem

DK	7250 cartridge disk subsystem with up to 8 drives
DPA	disk pack subsystem with up to 15 drives
DPB	disk pack subsystem with up to 15 drives
RAD	7211/7212 or 7231/7232 fixed head disk
MT	732X 9-track magnetic tape with up to 8 drives

Most devices can be disabled or enabled with the SET <dev> DISABLED and SET <dev> ENABLED commands, respectively.

The LOAD and DUMP commands are not implemented.

2.1 Central Processor (CPU)

Central processor options include the CPU model, the CPU features, and the size of main memory.

SET CPU SIGMA5	set CPU model to Sigma 5
SET CPU SIGMA6	set CPU model to Sigma 6
SET CPU SIGMA7	set CPU model to Sigma 7
SET CPU SIGMA7B	set CPU model to Sigma 7 "big" memory
SET CPU SIGMA8	set CPU Model to Sigma 8
SET CPU SIGMA9	set CPU model to Sigma 9
SET CPU 550	set CPU model to 550
SET CPU 560	set CPU model to 560
SET CPU FP	enable floating point, if available
SET CPU NOFP	disable floating point, if optional
SET CPU DECIMAL	enable decimal, if available
SET CPU NODECIMAL	disable decimal, if optional
SET CPU LASLAM	enable LAS/LAM instructions (6/7 only)
SET CPU NOLASLAM	disable LAS/LAM instructions
SET CPU MAP	enable memory map, if available
SET CPU NOMAP	disable memory map, if optional
SET CPU WRITELOCK	enable write locks, if available
SET CPU NOWRITELOCK	disable write locks, if optional
SET CPU RBLKS=n	set number of register blocks
SET CPU CHAN=n	set number of channels
SET CPU 32K	set memory size = 32KW
SET CPU 64K	set memory size = 64KW
SET CPU 128K	set memory size = 64KW
SET CPU 256K	set memory size = 256KW
SET CPU 512K	set memory size = 512KW
SET CPU 1M	set memory size = 1024KW

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 configuration is Sigma 7 CPU, 4 channels, 128KW of memory, floating point, decimal, map and writelocks options enabled.

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

name	size	comments
PC	17	program counter
R0..R15	32	active general registers
PSW1	32	processor status word 1
PSW2	32	processor status word 2
PSW4	32	processor status word 4 (5x0 only)

CC	4	condition codes
RP	5	register block selector
SSW1..SSW4	1	sense switches
PDF	1	processor fault flag
ALARM	1	console alarm
ALENB	1	console alarm enable
PCF	1	console controlled flop
EXULIM	8	limit for nested EXU's
STOP_ILL	1	if 1, stop on undefined instruction
REG	512	register blocks, 32 x 16
WRU	8	interrupt character

The CPU provides an address converter to display the byte (halfword, word, or doubleword) address equivalent of a byte (halfword, word, or doubleword) input address. Optionally, the input address can be run through memory relocation:

```
SHOW {-flags} CPU {BA,HA,WA,DA}=address
```

BA, HA, WA, DA specify that the input address is a byte, halfword, word, or doubleword address, respectively. The flags are:

```
-v          input address is virtual
-b          output is byte address
-h          output is halfword address
-w          output is word address (default)
-d          output is doubleword address
```

For example:

```
SHOW -B CPU WA=100
Physical word 100: physical byte 400
```

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 1M entries.

2.2 Memory Map (MAP)

The memory map implements two distinct forms of protection: memory mapping and access protection on virtual addresses; and write lock protection on physical addresses. It also includes a skeleton implementation of the memory status registers from the Sigma 8 and 9, and the XDS 550 and 560. It implements these registers:

name	size	comments
REL[0:511]	11	relocation registers
ACC[0:511]	2	access controls
WLK[0:2047]	4	write locks

SR0[32]	32	memory status register 0
SR1[32]	32	memory status register 1

2.3 Interrupt System (INT)

The Sigma series implements a complex, multi-level interrupts system, with a minimum of 32 distinct interrupts. The interrupt system can be expanded to up to 224 interrupt levels. It implements these registers:

name	size	comments
IHIACT	9	highest active interrupt level
IHIREQ	9	highest outstanding interrupt request
IREQ[0:15]	16	interrupt requests
IENB[0:15]	16	interrupt enabled flags
IARM[0:15]	16	interrupt armed flags
S9_SNAP	32	Sigma 9 snapshot register
S9_MARG	32	Sigma 9 margins registers
S5X0_IREG[0:31]	32	5X0 internal registers (unused)

The simulator supports a variable number of external interrupt blocks. The minimum number is one, the maximum is four (5X0) to thirty-two (Sigma 7). The user can change the number of external interrupt blocks with the SET INT EIBLKS command:

```
SET INT EIBLKS=4           configure four external interrupt blocks
```

Although the Sigma series supports configurable interrupt group priorities, the simulator uses a fixed priority arrangement, high to low as follows:

- counters
- counter overflow
- I/O and panel interrupt
- external group 2 (there is no external group 1)
- external group 3
- etc.

2.4 Channels (CHANA .. CHANH)

A Sigma 32b system has up to eight I/O channels, designated A, B, C, D, E, F, G, and H. The association between a device and a channel is displayed by the SHOW <dev> CHAN command and SHOW <dev> DVA command:

```
SHOW MT CHAN
channel=A
SHOW MT DVA
address=00
```

The user can change the association with the SET <dev> CHAN=<chan> command, where <chan> is a channel letter, and with the SET <dev> DVA=<addr> command, where <addr> is a legal device address:

```
SET MT CHAN=C
SET MT DVA=4
SHOW MT CHAN, DVA
channel=B
address=04
```

The default channel assignments for the simulator are:

Device	Channel	Address
MT	chan A	address 0
TT	chan A	address 1
LP	chan A	address 2
PT	chan A	address 5
COC	chan A	address 6
RAD	chan B	address 0
DK	chan B	address 1
DPA	chan C	address 0
DPB	chan D	address 1

The user can display all the channel registers associated with a device with the `SHOW <dev> CSTATE` command:

```
SHOW MT CSTATE
```

Each channel has eight registers. The registers are arrays, with entry 1 for device [0], entry [1] for device 1, etc. A channel supports a maximum of 32 devices.

name	size	comments
CLC[0:31]	16	channel location counter (doubleword)
CMD[0:31]	8	current channel command
CMF[0:31]	8	channel command flags
BA[0:31]	24	byte address (byte)
BC[0:31]	16	byte count
CMF[0:31]	16	channel status flags
CHI[0:31]	8	channel interrupt flags
CHSF[0:31]	8	channel simulator flags

2.5 7012 Console Teletype (TT)

The console Teletype (TT) consists of two units. Unit 0 is for console input, unit 1 for console output. The console implements these registers:

name	size	comments
KPOS	32	number of characters input
TPOS	32	number of characters input
TTIME	24	time from I/O initiation to interrupt
PANEL	8	character code for panel interrupt

The PANEL variable defaults to control-P. If the user types the control panel character, it is not echoed; instead, a control panel interrupt is generated.

The console can be set to one of two modes, 7P or UC:

mode	input characters	output characters
UC	lower case converted to upper case	lower case converted to upper case
7P		non-printing characters suppressed

The default mode is UC. The console character set is a subset of EBCDIC. By default, the console is assigned to channel A as device 1.

2.6 7060 Paper Tape Reader (PT)

The paper tape controller implements two units. Unit 0 (PT0) is for paper tape input, unit 1 (PT1) for paper tape output. Paper tapes are simulated as disk files. For the reader, register RPOS specifies the number of the next data item to be read. For the punch, register PPOS specifies the number of the next data item to be written. Thus, by changing RPOS or PPOS, the user can backspace or advance the reader or punch.

The paper tape controller implements these registers:

name	size	comments
CMD	9	current command or state
NZC	1	non-zero character seen since ATTACH
RPOS	32	position in the reader input file
RTIME	24	time from reader I/O initiation to response
RSTOP_IOE	1	stop on reader I/O error
PPOS	32	position in the punch output file
PTIME	24	time from punch I/O initiation to response
PSTOP_IOE	1	punch stop on I/O error

The paper-tape reader supports the `BOOT` command. `BOOT PT0` simulates the standard console fill sequence.

Reader error handling is as follows:

error	RSTOP_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

Punch error handling is as follows:

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

By default, the paper tape reader is assigned to channel A as device 5.

2.7 7440 or 7450 Line Printer (LP)

The line printer (LPT) 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 printer.

The line printer implements these registers:

name	size	comments
CMD	9	current command or state
BUF[0:131]	7	data buffer
PASS	1	printing pass (7440 only)
INH	1	inhibit spacing flag
RUNAWAY	1	runaway carriage-control tape flag
CCT[0:131]	8	carriage control tape
CCTP	8	pointer into carriage control tape
CCTL	8	length of carriage control tape
POS	32	position in the output file
TIME	24	time from I/O initiation to response
STOP_IOE	1	stop on I/O error

The line printer model can be set to either the 7440 or the 7450:

```
SET LP 7440          set model to 7440
SET LP 7450          set model to 7450
```

The default model is the 7450.

A carriage control tape can be loaded with the SET LP CCT command:

```
SET LP CCT=<file>          load carriage-control tape from <file>
```

The format of a carriage control tape consists of multiple lines. Each line contains an optional repeat count, enclosed in parentheses, optionally followed by a series of column numbers separated by commas. Column numbers must be between 0 and 7. The following are all legal carriage control specifications:

```
<blank line>          no punch
(5)                   5 lines with no punches
1,5,7                 columns 1, 5, 78 punched
(10)2                 10 lines with column 2 punched
1                     column 1 punched
```

The default form is 1 line long, with every column punched.

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

By default, the line printer is assigned to channel A as device 2.

2.8 Real-Time Clock (RTC)

The Sigma 32b series implements four real-time clocks. Three of them can be set to a variety of frequencies, including off, 50Hz, 60Hz, 100Hz, and 500Hz. The fourth always runs at 500Hz. The frequency of each clock can be adjusted as follows:

```
SET RTC {C1,C2,C3}=freq      set clock 1/2/3 to specified frequency
```

The frequency of each clock can be display as follows:

```
SHOW RTC {C1,C2,C3,C4}      show clock 1/2/3/4 frequency
```

Clocks 1 and 2 default to off, clocks 3 and 4 to 500Mhz.

The RTC can also show the state of all real-time events in the simulator:

```
SHOW RTC EVENTS
```

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

2.9 Character-Oriented Communications Subsystem (MUX)

The character-oriented communications subsystem implements up to 64 asynchronous interfaces, with modem control. The subsystem has two controllers: MUX for the scanner, and MUXL for the individual lines. The terminal multiplexer performs input and output through Telnet sessions connected to a user-specified port. The ATTACH command specifies the port to be used:

```
ATTACH MUX <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.

Unlike the console, the MUX operates in ASCII. Each line (each unit of MUXL) supports four character processing modes: UC, 7P, 7B, and 8B.

mode	input characters	output characters
UC	high-order bit cleared, lower case converted to upper case	high-order bit cleared, lower case converted to upper case
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 is UC. In addition, each line supports output logging. The SET MUXLn LOG command enables logging on a line:

```
SET MUXLn filename          log output of line n to filename
```

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

Once MUX is attached and the simulator is running, the multiplexer listens for connections on the specified port. It assumes that the incoming connections are Telnet connections. The connections remain open until disconnected either by the Telnet client, a SET MUX DISCONNECT command, or a DETACH MUX command.

Other special multiplexer commands:

```
SHOW MUX CONNECTIONS      show current connections
```

```

SHOW MUX STATISTICS          show statistics for active connections
SET MUXLn DISCONNECT        disconnects the specified line.

```

The controller (MUX) implements these registers:

name	size	comments
STA[0:63]	8	status, lines 0 to 63
RBUF[0:63]	8	receive buffer, lines 0 to 63
XBUF[0:63]	8	transmit buffer, lines 0 to 63
SCAN	6	current scanner line
SLCK	1	scanner locked flag
CMD	2	channel command or state

The lines (MUXL) implements these registers:

name	size	comments
TIME[0:63]	24	transmit time, lines 0 to 31

The terminal multiplexer does not support save and restore. All open connections are lost when the simulator shuts down or MUX is detached. By default, the multiplexer is assigned to channel A as device 6.

2.107240, 7270, 7260, 7265, 7275, and T3281 Moving Head Disk Controllers (DPA, DPB)

The Sigma 32b series supports two moving head disk controllers (DPA, DPB). Each can be set to model one of six controllers (7240, 7270, 7260, 7265, 7275, or Telefile 3281):

```

SET DP{A,B} 7240          set DPA (or DPB) to 7240
SET DP{A,B} 7270          set DPA (or DPB) to 7270
SET DP{A,B} 7260          set DPA (or DPB) to 7260
SET DP{A,B} 7265          set DPA (or DPB) to 7265
SET DP{A,B} 7275          set DPA (or DPB) to 7275
SET DP{A,B} T3281        set DPA (or DPB) to Telefile 3281

```

The default for DPA is the 7270, for DPB, the 3T281.

The 7240 and 7270 support up to 8 7242 and 7271 drives, respectively. The 7260, 7265, and 7275 support up to 15 7261, 7266, and 7276 drives, respectively. The T3281 supports up to 15 drives of three different types, which can be mixed:

Model	Cylinders	Heads	Sectors
3288	17	5	823
3282	11	19	815
3283	17	19	815

The command to set a drive to a particular model is:

```

SET DPn <drive_type>          SET unit n to the specified drive type

```

The T3281 can also be set to autosize on ATTACH:

```

SET DPn AUTO                    SET unit n to autosize on ATTACH

```

Units can be set ENABLED or DISABLED. The DP controller supports the BOOT command.

The DP controllers implements the registers listed below. Registers suffixed with [0:14] are replicated per drive.

name	size	comments
FLAGS	8	controller flags
DIFF	16	cylinder difference
SKI	16	queued seek interrupts
TEST	16	test mode flags
ADDR[0:14]	32	current disk address, drives 0 to 14
CMD[0:14]	32	current disk command, drives 0 to 14
TIME	24	time between word transfers
STIME	24	seek time
WLK	16	write lock switches
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

By default, DPA is assigned to channel C as device 0, and DPB is assigned to channel D as device 0.

2.11 7250 Cartridge Disk Controller (DK)

The cartridge disk controller (DK) supports up to 8 drives. DK options include the ability to make drives write enabled or write locked:

SET DKn LOCKED	set drive n write locked
SET DKn WRITEENABLED	set drive n write enabled

The cartridge disk controller implements these registers:

name	size	comments
CMD	9	current command or state
FLAGS	8	controller flags
ADDR	8	current disk address
TIME	24	interval between data transfers
STIME	24	seek interval
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 non-existent disk

DK data files are buffered in memory; therefore, end of file and OS I/O errors cannot occur. By default, the cartridge disk controller is assigned to channel B as device 1.

2.127211/7212 or 7231/7232 Fixed Head Disk (RAD)

The Sigma 32b series supports two models of fixed head disk:

7211/7212	1.343MW
7231/7232	1.572MW

The user can select the model as follows:

SET RAD 7211	set model to 7211/7212
SET RAD 7231	set model to 7231/7232

The fixed head disk controller supports four units (drives). Units can be set ENABLED or DISABLED.

The fixed head disk controller implements these registers:

CMD	9	current command or state
FLAGS	8	controller flags
ADDR	15	current disk address
WLK	16	write lock switches
TIME	24	interval between data transfers

Error handling is as follows:

error	STOP_IOE	processed as
not attached	1	report error and stop
	0	non-existent disk

RAD data files are buffered in memory; therefore, end of file and OS I/O errors cannot occur. By default, the fixed head disk is assigned to channel B as device 0.

2.13723X Nine-Track Magnetic Tape (MT)

The magnetic tape controller supports up to eight units. MT 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 supports the BOOT command. BOOT MTn simulates the standard console fill sequence for unit n.

The magnetic tape implements these registers:

name	size	comments
BUF[0:65535]	8	transfer buffer
BPTR	17	buffer pointer
BLNT	17	buffer length
RWINT	8	queued rewind interrupts
TIME	24	interval between character transfers
CTIME	24	channel response time
RWTIME	24	rewind time
UST[0:7]	8	drive status, drives 0 to 7
UCMD[0:7]	8	channel command, drives 0 to 7
POS[0:7]	32	position, drives 0 to 7
STOP_IOE	1	stop on I/O error

Error handling is as follows:

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

By default, the magnetic tape is assigned to channel A as device 0.

3 Symbolic Display and Input

The Sigma 32b simulator implements symbolic display and input. Display is controlled by command line switches:

-a	display as ASCII character (byte addressing)
-b	display as byte (byte addressing)
-e	display as EBCDIC character (byte addressing)
-h	display as halfword (halfword addressing)
-ca	display as four ASCII characters (word addressing)
-ce	display as four EBCDIC characters (word addressing)
-m	display instruction mnemonics (word addressing)

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

# or -a	ASCII character (byte addressing)
' or -e	EBCDIC character (byte addressing)
-b	hexadecimal byte (byte addressing)
-h	hexadecimal halfword (halfword addressing)
" or -ac	four packed ASCII characters (word addressing)
-ae	four packed EBCDIC characters (word addressing)
alphabetic	instruction mnemonic (word addressing)
numeric	hexadecimal word (word addressing)

Instruction input uses (more or less) standard XDS Sigma assembler syntax. All instructions are variants on the same basic form:

```
mnemonic{, reg}    {*{address{, index}}}
```

Mnemonics are symbolic names for instructions. Registers are decimal values between 0 and 15. '*' represents indirect addressing. Addresses are hexadecimal and can be signed if used as literals. Index registers are always less than 8 and thus can be considered decimal.

Examples:

```
LW, 5          *100, 7
AI, 14         -3
LCFI          A
WAIT
BCR, 12        400
BNOV          10FE
SLS, 6         -8
```

The extended branch and shift mnemonics are recognized and decoded properly.