

SIM11 NOTES

COMMANDS

The following is a list of the SIM11 commands that may be entered from the keyboard. This file may be viewed from within the simulator with the F5 command. A list of the commands is available with the F10 command from the debug window.

Note the following:

n,m any number from 0 to 0FFFF (hex). The default base is hex. To enter numbers in another base use the suffixes 'T' for base ten, 'O' for base eight or 'Q' for base two. You may also use the prefixes '!' for base ten, '@' for base 8 and '%' for base two. Numbers must start with either one of these suffixes or a numeric character.
 Ex : 0FF = 255T = 11111111Q = !255 = %11111111
 add any valid ROM address (default hex).
 [] optional parameter.

@M [add] Causes a break in execution to occur when a write to the specified location is attempted. If no address is given, the breakpoint is cleared.

ANALOG Displays the current analog inputs on the screen.
 ANALOGn m Sets analog input n to the value m (0<=m<=0ffh) and then display the current analog inputs.

ASM [add] Assemble directly to ROM at address given or starting at last address used by this command.

BR [add] Sets a breakpoint at the indicated ROM address. If no address is given, the breakpoint is cleared.

BW Fixes display for black and white monitor.

CAPTURE fname Saves all hardware outputs in the file specified.

CAPTUREOFF Stops capture and closes capture file. File may be viewed after it is closed with the F5 command.

CY n Sets cycle counter.

DASM [add] Disassembles memory to the screen starting at address given or last location disassembled.

DISKIN Prompts the user for an input file name and an address. While data bytes (2 hex digits per line) are available in the file, all program data reads which references the given address are taken from the file.
Note: the address given in the processor space must be initialized. This command is used to facilitate the debug process by allowing the user to specify which values are read from a location. Only data reads from simulated executing code are taken from the file, code fetches are not. Using this command with a null file name terminates the datain process.

DISKOUT Prompts the user for an output file name and an address. During program execution, data writes to the given address are also logged to the given file along with the PC (program counter) and the cycle count at the beginning of execution for the current instruction. This command is used to facilitate the debug process by allowing the user to observe a sequence of data writes. Using this command with a null file name terminates the logging process.

EVAL n [op] [n] Evaluates input in decimal, hex, and binary. Can do simple calculations where op may be +,-/, or *.

EXIT Return to DOS.

G [add] [add] Set breakpoint at second address. Go from first address. If only one address given, it is the start address. If no stop address is given, simulator will run forever or until an error condition (breakpoint, uninitialized ram, etc.). If no address is given the command is a "Go forever" command. No history is saved.

GO [add] [add] same as G

GONEXT Go from the current PC until the next instruction in memory is fetched. This is commonly used in debugging when the PC is pointed to a JSR instruction. A break would occur after the return from the subroutine.

GOTIL add Go from the current PC until the PC = add. No history is saved.

HELP Shows help screen (same as F10).

INPUTC n Defines the potential input pins of port C to be n. Initial value = \$ff.

INPUTD n Defines the potential input pins of port D to be n. Initial value = \$3f.

LISTON Turns on the screen listing of the step by step information during stepping. This is the default.

LISTOFF Turns off the screen listing of the step by step information during stepping.

LOAD filename Loads the file (in S19 format) into simulator.

LOADH filename Loads the file (in Intel hex format) into simulator.

LOADMAP Loads the symbol table output of an IASM Integrated Assembler into the simulator to facilitate symbolic debugging. An appropriate file name is requested.

MD add Sets the memory window to a specific address.

MEM add [n] Sets ram location at address given to value given. If no value is specified, the user is prompted. Consecutive locations will be prompted for until an invalid number is input.

MM add [n] Same as MEM

PRCL This command writes directly to the port C input latch. Using the MM command or program writes to this register actually writes to the C port output latch.

PRINTON Sends all debug output to the printer. Default is off.

PRINTOFF Stops sending debug output to the printer. Default.

REG Redisplays status line in debug window.

RESET Simulates a hardware reset.

RESTORE fname Restores simulator state from a file created with the save command. File must be in the current directory. This does not restore symbols, use LOADMAP or SYMBOL commands to restore them.

SCI Shows the contents of the SCI.

SCIINPUT n Places n into the Serial Receive Data Register and if 9 bit transmission into R8 of SCCR1 and also sets the error flags and the RDRF bit.

Note that SRDR is not shown in the memory display which shows the Serial Transmit Data Register. To see the SRDR read the SCSR with RDRF=1 followed by a read to SRDR or use the SCI command.

SCILOOPON Loops back the SCI transmitter output into the receiver input by setting the receive bit in INPUTD.

SCILOOPOFF Turn off the SCI loop back.

SAVE filename Saves the simulator state in the specified file. This command is useful when at a point in the code where there are several branches to be tested. Returning to the top of the decision tree is easy if it is saved first. Does not save symbols.

SCRIPT fname Executes a script file. This file will usually be used to simulate hardware inputs but may be used to run any predetermined command sequence. The file must reside in the current directory.

SPI Shows the contents of the SPI.

ST [n] Single step. Simulate the execution of n instructions. If no number is given, one instruction will be executed. History will be saved.

STEP [n] Same as ST.

STEPTIL add Repeatedly single step until the program counter is equal to the given address. History will be saved.

STRA n Sets the STRA input pin to n (0 or 1). The STRA input pin is displayed in the PORTS window as the left character under STAB.

STRB n Sets the STRB flip flop to n (0 or 1). Note that the flip flop and not the port pin is set. The port pin is displayed in the PORTS window as the right character under STAB. The STRB flip flop and the STRB pin may not be the same.

T [n] Same as ST.

QUIT Return to DOS.

LOADMAP Loads the symbol table output of an IASM Integrated Assembler into the simulator to facilitate symbolic debugging. An appropriate file name is requested.

SYMBOL lab val Adds the given label to the symbol table with the value given.

WHEREIS symbol Gives the value for the given symbol.

Any on screen location with an alphabetic name (SP, PC, etc.) may be changed by entering the name of the location followed by a value such as:

SP 100 PC 2000 A 37 etc.

FUNCTION KEYS

- F1 Go to the debug window.
- F2 Go to the code window. Once in this window, the user may scroll and see either history or future code.
- F3 Go to the memory window. Once in this window, the user may scroll and set the window at any location within the memory space.
- F4 Go to the log window. The user may log notes to a file to be retrieved externally to the simulator.
- F5 Go to the file view window. The user may call any ASCII file and view it. Files longer than 2048 lines will be truncated.
- F6 Toggle the memory window between memory, expanded timer and misc. register windows.
- F9 Repeat last command.
- F10 Help window.

SCRIPT FILES

Script files are files of commands that the user wants to run on an automatic and timely basis. The typical use for these files is to simulate hardware inputs and interrupts to the device being simulated.

Any regular simulator command (see above) may be included in a script file plus one additional command, WAIT. The wait command is followed by a parameter that is from 0 to 7FFF and represents a number of simulated machine cycles to delay before continuing with the script file.

When the user enters the command "SCRIPT filename" the simulator opens the file specified and starts executing commands from it until either the end of the file is reached or a wait command is encountered. If the end of the file is reached, control passes back to the keyboard. If a wait command is encountered, control is passed back to the keyboard until the specified number of cycles have been executed. Then commands from the script file are once again executed until one of the two mentioned conditions happens again.

Example 1:
prtb 00000000q wait 25 prtb 00000001q wait 10 prtb 00000000q

where q is the suffix for binary numbers.

This simple script file would cause the simulator to see a 10 cycle long pulse on input pin 0 of port b. To simulate a continual wave form, the last four lines could be copied and repeated as many times as desired.

Note that since any value may be altered with a script file, such a file can greatly enhance your testing and at the same time really make a mess of things unless you are careful.

Example 2:
load myfile.s19 prtb 06 prta 03

This script file will set up the environment to a simple starting position and eliminate some typing. Obviously this can be made as elaborate as you like.

Script file commands can also contain comments if they are delineated by a semi-colon such as :

wait 32 ;delay prta 03 ;switch 2 closing

If the simulator is running from a GO or GOTIL command when a script file wait command times out, the script commands will be executed and then the GO or GOTIL will continue. If the simulator is running a ST, STEP, or STEPTIL command, the command will be terminated by the script file's execution. Reaching the end of a script file will terminate any running command.

Note that the commands GO and GOTIL will run slower when a script file is interacting with them.

CAPTURE FILES

The capture file simply records any changes to an output port caused by the program under test or from keyboard commands. In other words, any time any output pin is changed, the capture file is updated. The current cycle count is also recorded. The capture file may be closed with a CAPTUREOFF command and then viewed with the F5 command. The capture file is a simple ASCII file.

LOG FILES (F4)

Log files are files created by the user during a debug session. By hitting F4 and entering a file name, the user creates a simple ASCII file. This file may be added to anytime by re-entering the window. The use of log files is totally up to the user and it only logs notes typed in by the user, no simulator state is saved.

I/O PORTS

For a good explanation of the 68HC11 I/O port operations, refer to:

"M68HC11 REFERENCE MANUAL" number M68HC11RM/AD,
by Prentice Hall, Englewood Cliffs, NJ 07632, copyright Motorola 1989.