

Using XSpim

What is XSpim?

- It's a GUI version of the Spim simulator for MIPS
- Free and available for Win, Mac and Linux/Unix

The screenshot shows the XSpim GUI window titled "xspim (on mango)". It displays the following information:

- PC = 00000000**, **EPC = 00000000**, **Cause = 00000000**, **BadVAddr = 00000000**
- Status = 00000000**, **HI = 00000000**, **LO = 00000000**
- General Registers:** R0-R7, R8-R15, R16-R23, R24-R31 with their respective values.
- Double Floating Point Registers:** FP0-FP15 with values 0.00000.
- Single Floating Point Registers:** FP0-FP15 with values 0.00000.
- Control Buttons:** quit, load, reload, run, step, clear, set value, print, breakpoints, help, terminal, mode.
- Text Segments:** A list of memory addresses and instructions, such as:
 - [0x00400000] 0x8fa40000 lw \$4, 0(\$29) ; 140: lw \$a0, 0(\$sp)# ar
 - [0x00400004] 0x27a50004 addiu \$5, \$29, 4 ; 141: addiu \$a1, \$sp, 4#
 - [0x00400008] 0x24a60004 addiu \$6, \$5, 4 ; 142: addiu \$a2, \$a1, 4#
 - [0x0040000c] 0x00041080 sll \$2, \$4, 2 ; 143: sll \$v0, \$a0, 2
 - [0x00400010] 0x00c23021 addu \$6, \$6, \$2 ; 144: addu \$a2, \$a2, \$v0
 - [0x00400014] 0x0c100009 jal 0x00400024 [main] ; 145: jal main
 - [0x00400018] 0x00000000 nop ; 146: nop
 - [0x0040001c] 0x3402000a ori \$2, \$0, 10 ; 148: li \$v0 10
 - [0x00400020] 0x0000000c syscall ; 149: syscall # sys
 - [0x00400024] 0x20190001 addi \$25, \$0, 1 ; 3: addi \$t9, \$zero, 1
- Data Segments:** A list of memory addresses and data values, such as:
 - [0x7fffec90] 0x00000001 0x7fffed21 0x00000000 0x7fffefea
 - [0x7fffec94] 0x7fffedc0 0x7fffeb6 0x7fffeb86 0x7fffeb69
 - [0x7fffec98] 0x7fffeb2c 0x7fffeb1e 0x7fffeb0d 0x7fffebfb
 - [0x7fffec9c] 0x7fffeeee 0x7fffeed6 0x7fffeb5 0x7fffeea
 - [0x7fffec00] 0x7fffee9d 0x7fffee7b 0x7fffee64 0x7fffee54
- Memory and registers cleared**
- Loaded:** /opt/spim-6.5/share/spim-6.5/trap.handler
- SPIM Version 6.5 of January 4, 2003**
- Copyright 1990-2003 by James R. Larus (larus@cs.wisc.edu).**
- All Rights Reserved.**
- See the file README for a full copyright notice.**

Accessing XSpim

- XSpim is available for use on the CUNIX system with the command `xspim`
 - Need to use Xwindows when connecting
 - See <http://www.columbia.edu/~lgw23/cs1004/> for info on connecting to CUNIX and using Xwindows
- Can download to your own system but grading will be done on CUNIX

Creating a File

- Before running XSpim you need to create an assembly file
- File should be plain text, typically with .asm extension

Example Program:

```
.text # signals start of code
main: # signals start of program

    addi $t9, $zero, 1
    beq $t0, $zero, end

loop:

    and $t2, $t0, $t9
    sll $t9, $t9, 1
    beq $t2, $zero, loop
    add $t0, $t2, $zero

end:
```

Loading File

xspim (on mango)

PC = 00000000 EPC = 00000000 Cause = 00000000 BadVAddr= 00000000
Status = 00000000 HI = 00000000 LO = 00000000

General Registers

R0 (r0) = 00000000	R8 (t0) = 00000000	R16 (s0) = 00000000	R24 (t8) = 00000000
R1 (at) = 00000000	R9 (t1) = 00000000	R17 (s1) = 00000000	R25 (t9) = 00000001
R2 (v0) = 00000004	R10 (t2) = 00000000	R18 (s2) = 00000000	R26 (k0) = 00000000
R3 (v1) = 00000000	R11 (t3) = 00000000	R19 (s3) = 00000000	R27 (k1) = 00000000
R4 (a0) = 00000001	R12 (t4) = 00000000	R20 (s4) = 00000000	R28 (gp) = 10008000
R5 (a1) = 7fffec94	R13 (t5) = 00000000	R21 (s5) = 00000000	R29 (sp) = 7fffec90
R6 (a2) = 7fffec9c	R14 (t6) = 00000000	R22 (s6) = 00000000	R30 (s8) = 00000000
R7 (a3) = 00000000	R15 (t7) = 00000000	R23 (s7) = 00000000	R31 (ra) = 00400018

Double Floating Point Registers

FP0 = 0.00000	FP8 = 0.00000	FP16 = 0.00000	FP24 = 0.00000
FP2 = 0.00000	FP10 = 0.00000	FP18 = 0.00000	FP26 = 0.00000
FP4 = 0.00000	FP12 = 0.00000	FP20 = 0.00000	FP28 = 0.00000
FP6 = 0.00000	FP14 = 0.00000	FP22 = 0.00000	FP30 = 0.00000

Single Floating Point Registers

FP0 = 0.00000	FP8 = 0.00000	FP16 = 0.00000	FP24 = 0.00000
---------------	---------------	----------------	----------------

Buttons: quit, **load**, reload, run, step, clear, set value, print, breakpoints, help, terminal, mode

Text Segments

```
[0x00400000] 0x8fa40000 lw $4, 0($29) ; 140: lw $a0, 0($sp)# ar
[0x00400004] 0x27a50004 addiu $5, $29, 4 ; 141: addiu $a1, $sp, 4#
[0x00400008] 0x24a60004 addiu $6, $5, 4 ; 142: addiu $a2, $a1, 4#
[0x0040000c] 0x00041080 sll $2, $4, 2 ; 143: sll $v0, $a0, 2
[0x00400010] 0x00c23021 addu $6, $6, $2 ; 144: addu $a2, $a2, $v0
[0x00400014] 0x0c100009 jal 0x00400024 [main] ; 145: jal main
[0x00400018] 0x00000000 nop ; 146: nop
[0x0040001c] 0x3402000a ori $2, $0, 10 ; 148: li $v0 10
[0x00400020] 0x0000000c syscall ; 149: syscall # sys
[0x00400024] 0x20190001 addi $25, $0, 1 ; 3: addi $t9, $zero, 1
```

Data Segments

DATA
[0x10000000]... [0x10020000] 0x00000000

STACK

[0x7fffec90]	0x00000001	0x7fffed21	0x00000000	0x7fffefea
[0x7fffec80]	0x7fffedc0	0x7fffeb60	0x7fffeb86	0x7fffeb69
[0x7fffeb00]	0x7fffeb2c	0x7fffeb1e	0x7fffeb0d	0x7fffebfb
[0x7fffecc0]	0x7fffeeee	0x7fffed66	0x7fffeb55	0x7fffeea5
[0x7fffed00]	0x7fffee9d	0x7fffee7b	0x7fffee64	0x7fffee54

Memory and registers cleared

Loaded: /opt/spim-6.5/share/spim-6.5/trap.handler
SPIM Version 6.5 of January 4, 2003
Copyright 1990-2003 by James R. Larus (larus@cs.wisc.edu).
All Rights Reserved.
See the file README for a full copyright notice.

xspim (on mango)

PC = 00000000 EPC = 00000000 Cause = 00000000 BadVAddr= 00000000
Status = 00000000 HI = 00000000 LO = 00000000

General Registers

R0 (r0) = 00000000	R8 (t0) = 00000000	R16 (s0) = 00000000	R24 (t8) = 00000000
R1 (at) = 00000000	R9 (t1) = 00000000	R17 (s1) = 00000000	R25 (t9) = 00000001
R2 (v0) = 00000004	R10 (t2) = 00000000	R18 (s2) = 00000000	R26 (k0) = 00000000
R3 (v1) = 00000000	R11 (t3) = 00000000	R19 (s3) = 00000000	R27 (k1) = 00000000
R4 (a0) = 00000001	R12 (t4) = 00000000	R20 (s4) = 00000000	R28 (gp) = 10008000
R5 (a1) = 7fffec94	R13 (t5) = 00000000	R21 (s5) = 00000000	R29 (sp) = 7fffec90
R6 (a2) = 7fffec9c	R14 (t6) = 00000000	R22 (s6) = 00000000	R30 (s8) = 00000000
R7 (a3) = 00000000	R15 (t7) = 00000000	R23 (s7) = 00000000	R31 (ra) = 00400018

Double Floating Point Registers

FP0 = 0.00000	FP8 = 0.00000	FP16 = 0.00000	FP24 = 0.00000
FP2 = 0.00000	FP10 = 0.00000	FP18 = 0.00000	FP26 = 0.00000
FP4 = 0.00000	FP12 = 0.00000	FP20 = 0.00000	FP28 = 0.00000
FP6 = 0.00000	FP14 = 0.00000	FP22 = 0.00000	FP30 = 0.00000

Single Floating Point Registers

FP0 = 0.00000	FP8 = 0.00000	FP16 = 0.00000	FP24 = 0.00000
---------------	---------------	----------------	----------------

Buttons: quit, load, reload, run, step, clear, set value, print, breakpoints, help, terminal, mode

popup (on mango)

input filename:
test.asm
assembly file abort command

Data Segments

DATA
[0x10000000]... [0x10020000] 0x00000000

STACK

[0x7fffec90]	0x00000001	0x7fffed21	0x00000000	0x7fffefea
[0x7fffec80]	0x7fffedc0	0x7fffeb60	0x7fffeb86	0x7fffeb69
[0x7fffeb00]	0x7fffeb2c	0x7fffeb1e	0x7fffeb0d	0x7fffebfb
[0x7fffecc0]	0x7fffeeee	0x7fffed66	0x7fffeb55	0x7fffeea5
[0x7fffed00]	0x7fffee9d	0x7fffee7b	0x7fffee64	0x7fffee54

Memory and registers cleared

Loaded: /opt/spim-6.5/share/spim-6.5/trap.handler
SPIM Version 6.5 of January 4, 2003
Copyright 1990-2003 by James R. Larus (larus@cs.wisc.edu).
All Rights Reserved.
See the file README for a full copyright notice.

Running Program

The screenshot shows the xspim (on mango) interface. The top section displays system information: PC = 00000000, EPC = 00000000, Cause = 00000000, BadVAddr = 00000000, Status = 00000000, HI = 00000000, LO = 00000000. Below this are the General Registers (R0-R31) and Double Floating Point Registers (FP0-FP31). The 'run' button in the control panel is circled in red. The Text Segments section shows assembly code starting at 0x00400000, including instructions like lw, addiu, sll, addu, jal, nop, ori, syscall, and addi. The Data Segments section shows memory addresses from 0x10000000 to 0x10020000 with a value of 0x00000000. The STACK section shows memory addresses from 0x7fffec90 to 0x7fffee00 with various values. The bottom section contains copyright information for SPIM Version 6.5 of January 4, 2003, by James R. Larus.

The screenshot shows the xspim (on mango) interface with a 'prompt (on mango)' dialog box open. The dialog box has a title bar 'prompt (on mango) X' and contains the text 'run program'. It has two input fields: 'starting address:' with the value '0x00400000' and 'args:' with the value 'test.asm'. There are 'ok' and 'abort command' buttons. The background shows the same xspim interface as the first screenshot, but the 'run' button is not circled. The Text Segments, Data Segments, and STACK sections are visible and contain the same information as in the first screenshot.

Other Useful Buttons

- Set value – set the value of a register
- Step – step through the code, line by line
- Clear – reset registers back to zero
- Reload – reload last assembly program

```
xspim (on mango)
PC = 00000000 EPC = 00000000 Cause = 00000000 BadVAddr= 00000000
Status = 00000000 HI = 00000000 Lo = 00000000

General Registers
R0 (r0) = 00000000 R8 (t0) = 00000000 R16 (s0) = 00000000 R24 (t8) = 00000000
R1 (a0) = 00000000 R9 (t1) = 00000000 R17 (s1) = 00000000 R25 (t9) = 00000001
R2 (v0) = 00000004 R10 (t2) = 00000000 R18 (s2) = 00000000 R26 (k0) = 00000000
R3 (v1) = 00000000 R11 (t3) = 00000000 R19 (s3) = 00000000 R27 (k1) = 00000000
R4 (a0) = 00000001 R12 (t4) = 00000000 R20 (s4) = 00000000 R28 (gp) = 10008000
R5 (a1) = 7fffec94 R13 (t5) = 00000000 R21 (s5) = 00000000 R29 (sp) = 7fffec90
R6 (a2) = 7fffec9c R14 (t6) = 00000000 R22 (s6) = 00000000 R30 (s8) = 00000000
R7 (a3) = 00000000 R15 (t7) = 00000000 R23 (s7) = 00000000 R31 (ra) = 00400018

Double Floating Point Registers
FP0 = 0.00000 FP8 = 0.00000 FPI6 = 0.00000 FP24 = 0.00000
FP2 = 0.00000 FP10 = 0.00000 FP18 = 0.00000 FP26 = 0.00000
FP4 = 0.00000 FP12 = 0.00000 FP20 = 0.00000 FP28 = 0.00000
FP6 = 0.00000 FP14 = 0.00000 FP22 = 0.00000 FP30 = 0.00000

Single Floating Point Registers
FP0 = 0.00000 FP8 = 0.00000 FPI6 = 0.00000 FP24 = 0.00000

quit load reload run step clear
set value print breakpoints help terminal mode

Text Segments
[0x00400000] 0x8fa40000 lw $4, 0($29) ; 140: lw $a0, 0($sp)# ar
[0x00400004] 0x27a50004 addiu $5, $29, 4 ; 141: addiu $a1, $sp, 4#
[0x00400008] 0x24a60004 addiu $6, $5, 4 ; 142: addiu $a2, $a1, 4#
[0x0040000c] 0x00041080 sll $2, $4, 2 ; 143: sll $v0, $a0, 2
[0x00400010] 0x00c23021 addu $6, $6, $2 ; 144: addu $a2, $a2, $v0
[0x00400014] 0x0c100009 jal 0x00400024 [main] ; 145: jal main
[0x00400018] 0x00000000 nop ; 146: nop
[0x0040001c] 0x3402000a ori $2, $0, 10 ; 148: li $v0 10
[0x00400020] 0x0000000c syscall ; 149: syscall # sys
[0x00400024] 0x20190001 addi $25, $0, 1 ; 3: addi $t9, $zero, 1

Data Segments
DATA
[0x10000000]... [0x10020000] 0x00000000
STACK
[0x7fffec90] 0x00000001 0x7fffed21 0x00000000 0x7fffefea
[0x7fffec94] 0x7fffedfc 0x7fffeb6 0x7fffe186 0x7fffeb69
[0x7fffec98] 0x7fffeb2c 0x7fffeb1e 0x7fffeb0d 0x7fffebfb
[0x7fffec9c] 0x7fffeeee 0x7fffed6 0x7fffeb5 0x7fffeea
[0x7fffec00] 0x7fffee9d 0x7fffee7b 0x7fffee64 0x7fffee54

Memory and registers cleared
Loaded: /opt/spim-6.5/share/spim-6.5/trap.handler
SPIM Version 6.5 of January 4, 2003
Copyright 1990-2003 by James R. Larus (larus@cs.wisc.edu).
All Rights Reserved.
See the file README for a full copyright notice.
```

Additional Tips

- Avoid clicking the 'x' button on windows to close them as this will crash XSpim
 - Use 'abort command' buttons instead
- When typing data into a field, make sure your mouse cursor is over the field
 - Moving away will require you to select the field again
- A comprehensive discussion of MIPS and XSPIM can be found in the Appendix of P&H