CprE 381, Computer Organization and Assembly-Level Programming

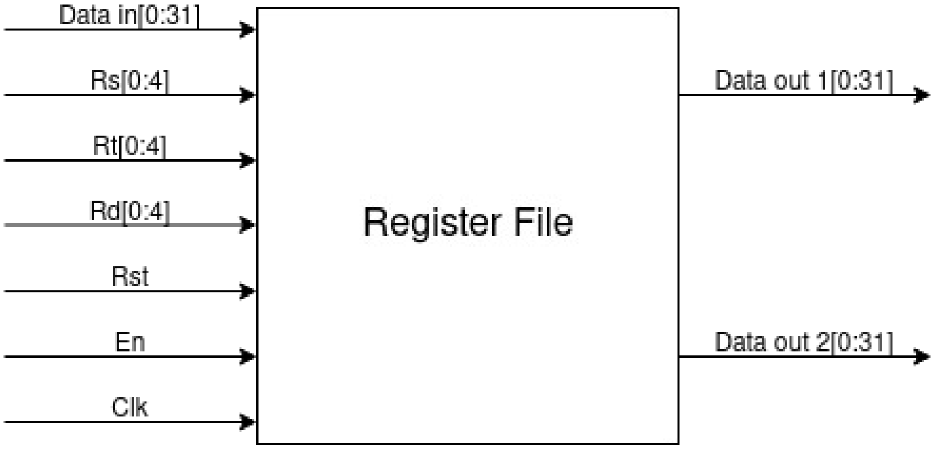
Lab 2 Report

Student Name _Caleb Hemmestad___________________________

***Submit a typeset pdf version of this on Canvas by the due date. Refer to the highlighted language in the lab document for the context of the following questions*.**

[Part 2 (a)] Draw the interface description for the MIPS register file. Which ports do you think are necessary, and how wide (in bits) do they need to be?

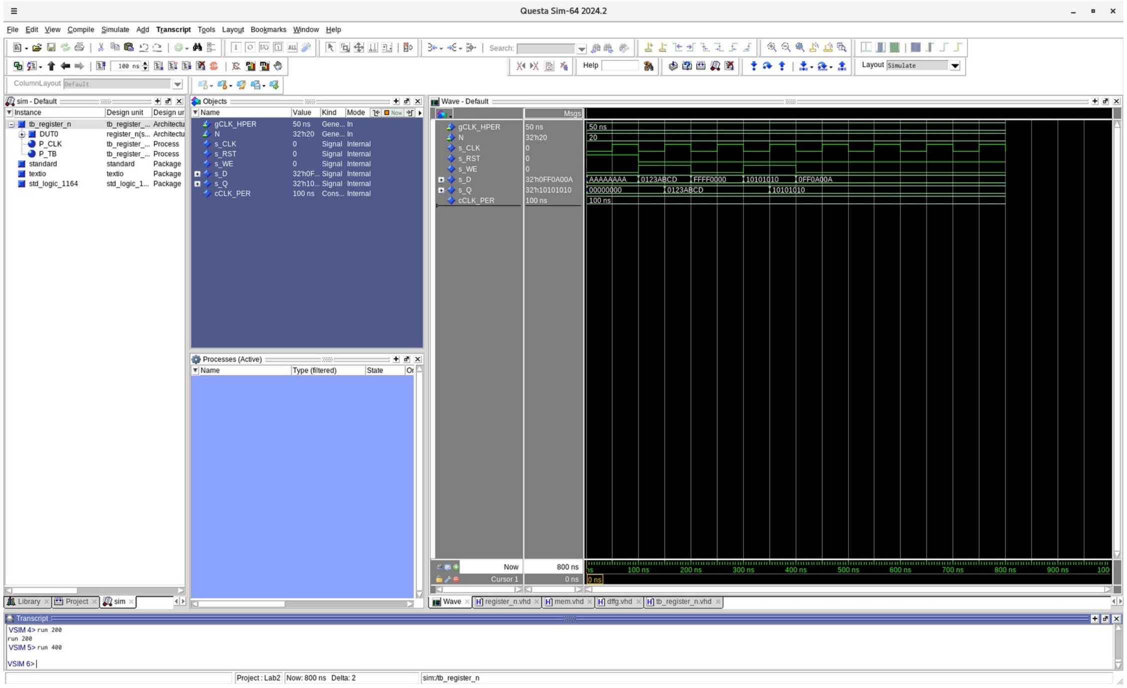
The data in and both the data out's should be 32 bits wide. Rs, rt, and rd should all be five bits wide. Rst, Clk, and En should be only one bit.



[Part 2 (b)] Create an N-bit register using this flip-flop as your basis.

Done.

[Part 2 (c)] Waveform.



Questa Sim-64 2024.2

sim - Default

Instance	Design unit	Design ur
tb_register_n	tb_register_...	Architectu
DUT0	register_n(s...	Architectu
P_CLK	tb_register_...	Process
P_TB	tb_register_...	Process
standard	standard	Package
textio	textio	Package
std_logic_1164	std_logic_1...	Package

Objects

Name	Value	Kind	Mode
gCLK_HPER	50 ns	Gene...	In
N	32'h20	Gene...	In
s_CLK	0	Signal	Internal
s_RST	0	Signal	Internal
s_WE	0	Signal	Internal
s_D	32'h0F...	Signal	Internal
s_Q	32'h10...	Signal	Internal
cCLK_PER	100 ns	Cons...	Internal

Wave - Default

	Msgs
gCLK_HPER	50 ns
N	32'h20
s_CLK	0
s_RST	0
s_WE	0
s_D	32'h0FF0A00A
s_Q	32'h10101010
cCLK_PER	100 ns
Now	800 ns
Cursor 1	0 ns

Transcript

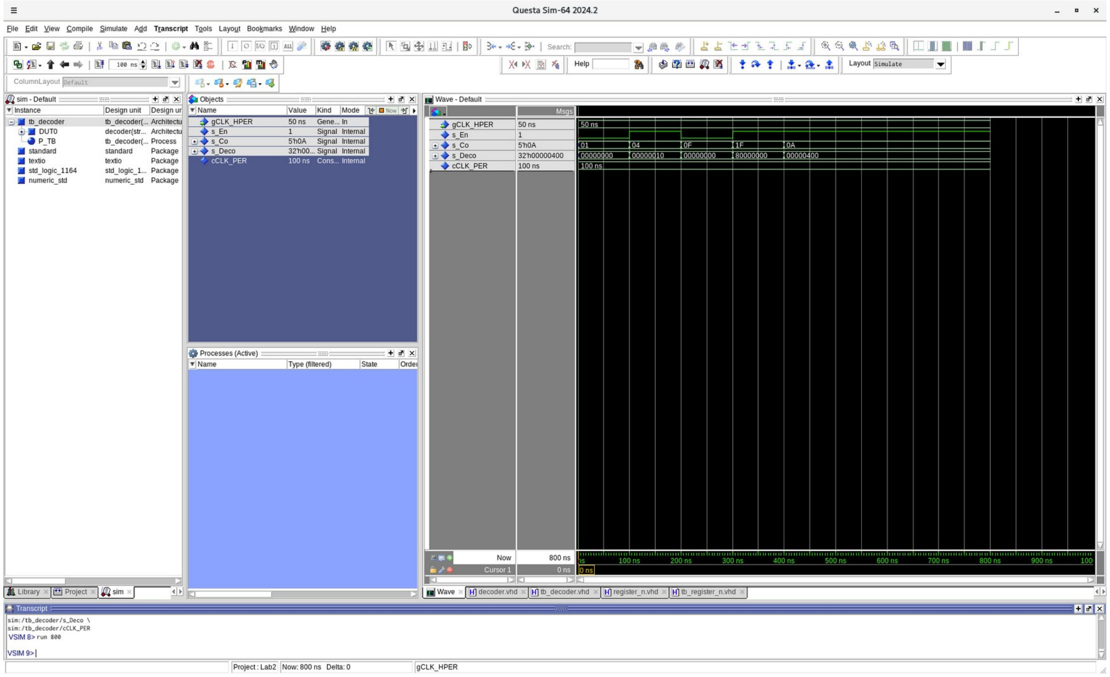
```
VSIM 4> run 200
run 200
VSIM 5> run 400

VSIM 6>
```

Project : Lab2 | Now: 800 ns Delta: 2 | sim:/tb_register_n

[Part 2 (d)] What type of decoder would be required by the MIPS register file and why?

It would need to be a 5:32 decoder because there will be 32 registers to select from and log2(32) = 5 which means you would need five select bits for that many options.

[Part 2 (e)] Waveform.



Questa Sim-64 2024.2

sim - Default

Instance	Design unit	Design ur
tb_decoder	tb_decoder(...	Architectu
DUT0	decoder(str...	Architectu
P_TB	tb_decoder(...	Process
standard	standard	Package
textio	textio	Package
std_logic_1164	std_logic_1...	Package
numeric_std	numeric_std	Package

Objects

Name	Value	Kind	Mode
gCLK_HPER	50 ns	Gene...	In
s_En	1	Signal	Internal
s_Co	5'h0A	Signal	Internal
s_Deco	32'h00...	Signal	Internal
cCLK_PER	100 ns	Cons...	Internal

Wave - Default

	Msgs
gCLK_HPER	50 ns
s_En	1
s_Co	5'h0A
s_Deco	32'h00000400
cCLK_PER	100 ns
Now	800 ns
Cursor 1	0 ns

Transcript

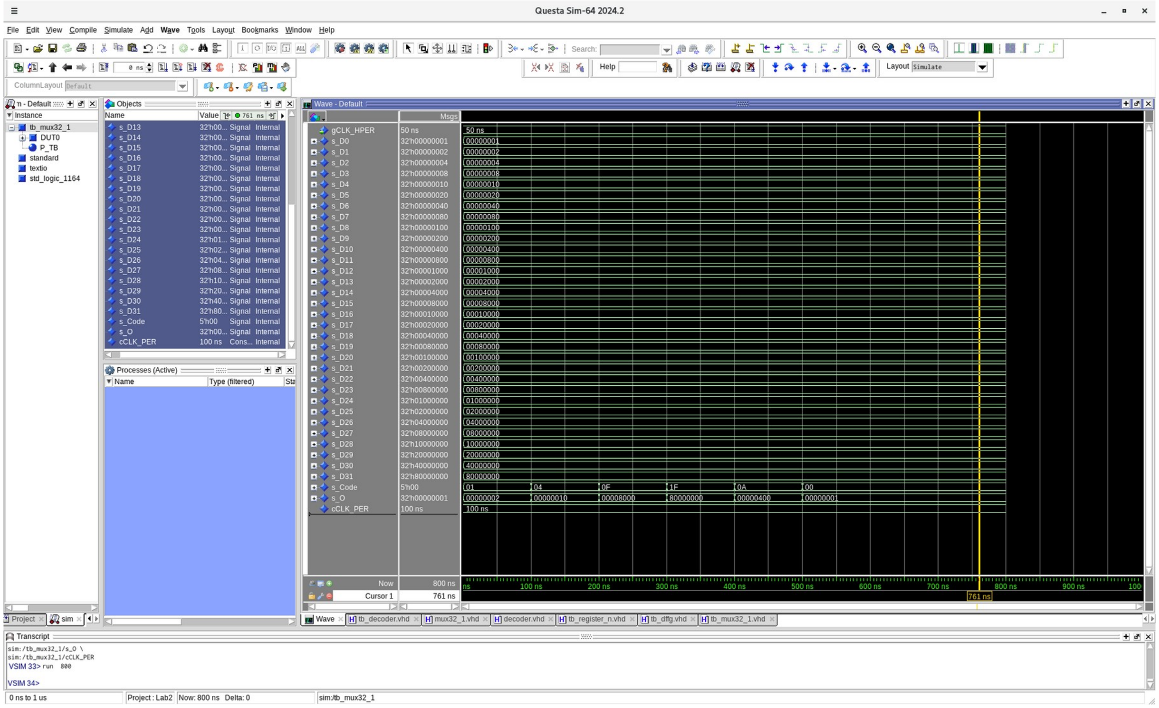
```
sim:/tb_decoder/s_Deco \
sim:/tb_decoder/cCLK_PER
VSIM 8> run 800

VSIM 9>
```

Project : Lab2 | Now: 800 ns Delta: 0 | gCLK_HPER

[Part 2 (f)] In your write-up, describe and defend the design you intend on implementing for the next part.

For my design I am using a dataflow approach because it will be easier and faster to implement for a multiplexer this large.

[Part 2 (g)] Waveform.

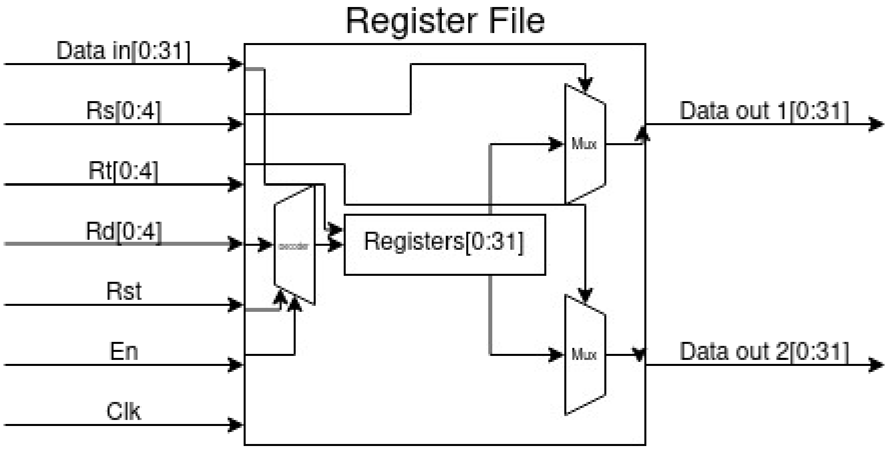


```
                Msgs
gCLK_HPER       50 ns           50 ns
s_D0            32'h00000001    00000001
s_D1            32'h00000002    00000002
s_D2            32'h00000004    00000004
s_D3            32'h00000008    00000008
s_D4            32'h00000010    00000010
s_D5            32'h00000020    00000020
s_D6            32'h00000040    00000040
s_D7            32'h00000080    00000080
s_D8            32'h00000100    00000100
s_D9            32'h00000200    00000200
s_D10           32'h00000400    00000400
s_D11           32'h00000800    00000800
s_D12           32'h00001000    00001000
s_D13           32'h00002000    00002000
s_D14           32'h00004000    00004000
s_D15           32'h00008000    00008000
s_D16           32'h00010000    00010000
s_D17           32'h00020000    00020000
s_D18           32'h00040000    00040000
s_D19           32'h00080000    00080000
s_D20           32'h00100000    00100000
s_D21           32'h00200000    00200000
s_D22           32'h00400000    00400000
s_D23           32'h00800000    00800000
s_D24           32'h01000000    01000000
s_D25           32'h02000000    02000000
s_D26           32'h04000000    04000000
s_D27           32'h08000000    08000000
s_D28           32'h10000000    10000000
s_D29           32'h20000000    20000000
s_D30           32'h40000000    40000000
s_D31           32'h80000000    80000000
s_Code          5'h00           01        04        0F        1F        0A        00
s_O             32'h00000001    00000002  00000010  00008000  80000000  00000400  00000001
cCLK_PER        100 ns          100 ns

Now            800 ns   ns   100 ns   200 ns   300 ns   400 ns   500 ns   600 ns   700 ns   800 ns   900 ns
Cursor 1       761 ns                                                                    761 ns
```

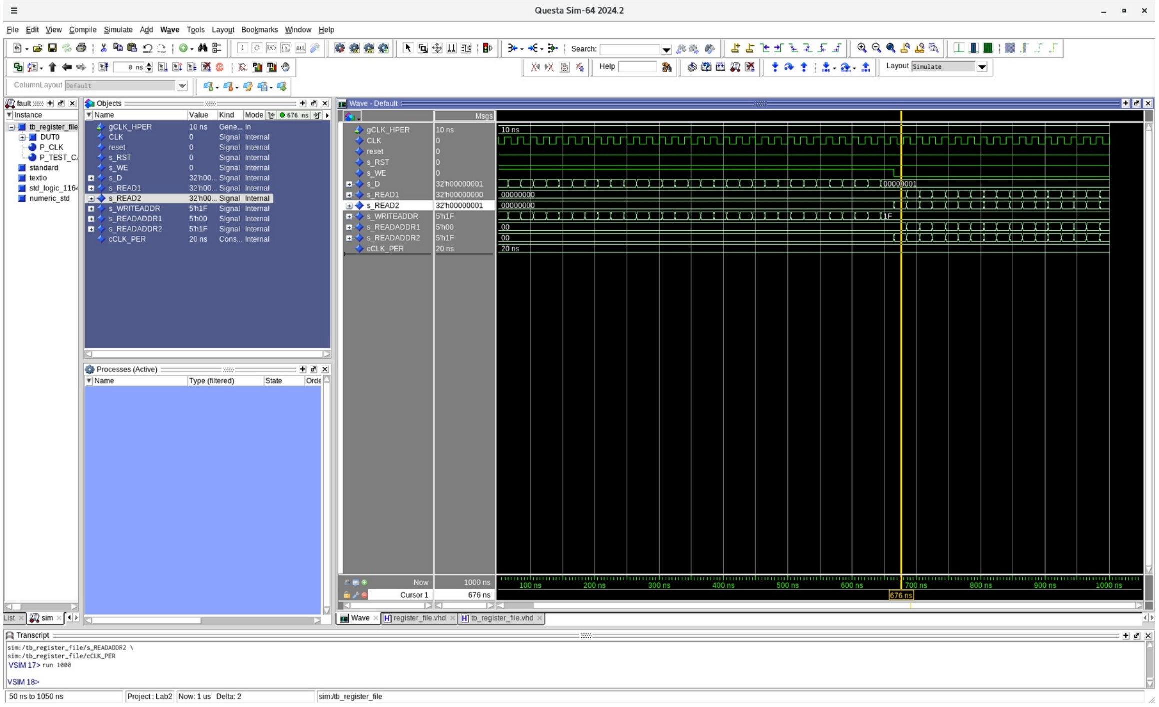
```
Transcript
sim:/tb_mux32_1/s_O \
sim:/tb_mux32_1/cCLK_PER
VSIM 33> run  800

VSIM 34>
```

```
0 ns to 1 us    Project : Lab2    Now: 800 ns  Delta: 0    sim:/tb_mux32_1
```

[Part 2 (h)] Draw a (simplified) schematic for the MIPS register file, using the same top-level interface ports as in your solution describe above and using only the register, decoder, and mux VHDL components you have created.



[Part 2 (i)] Waveform.



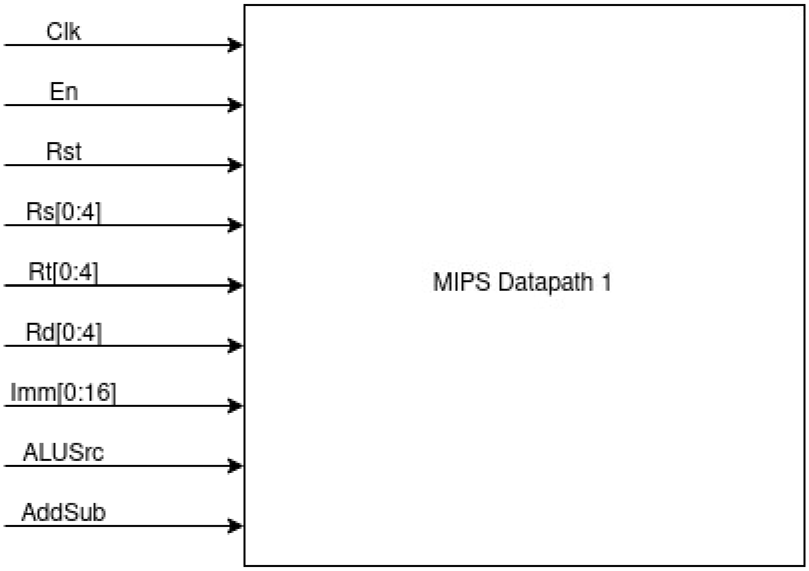
Name	Value	Kind	Mode
gCLK_HPER	10 ns	Gene...	In
CLK	0	Signal	Internal
reset	0	Signal	Internal
s_RST	0	Signal	Internal
s_WE	0	Signal	Internal
s_D	32'h00...	Signal	Internal
s_READ1	32'h00...	Signal	Internal
s_READ2	32'h00...	Signal	Internal
s_WRITEADDR	5'h1F	Signal	Internal
s_READADDR1	5'h00	Signal	Internal
s_READADDR2	5'h1F	Signal	Internal
cCLK_PER	20 ns	Cons...	Internal

	Msgs
gCLK_HPER	10 ns
CLK	0
reset	0
s_RST	0
s_WE	0
s_D	32'h00000001
s_READ1	32'h00000000
s_READ2	32'h00000001
s_WRITEADDR	5'h1F
s_READADDR1	5'h00
s_READADDR2	5'h1F
cCLK_PER	20 ns
Now	1000 ns
Cursor 1	676 ns

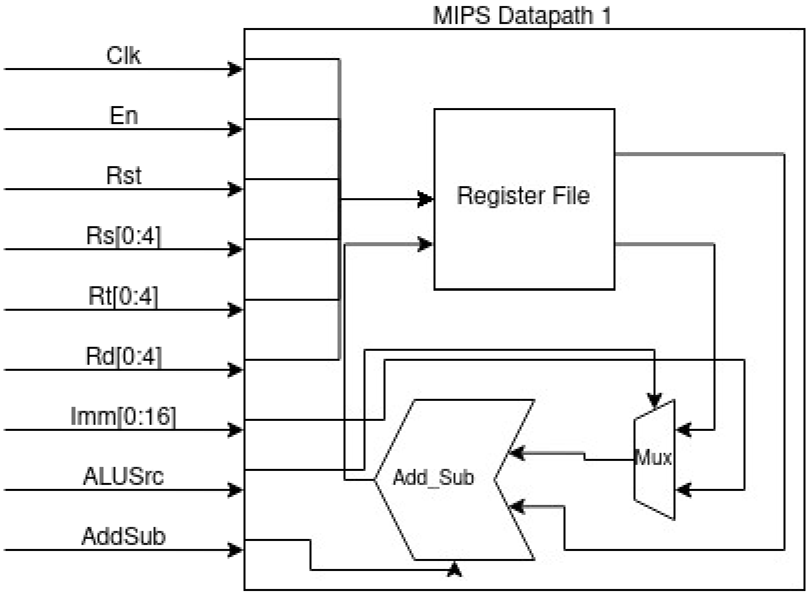
```
sim:/tb_register_file/s_READADDR2 \
sim:/tb_register_file/cCLK_PER
VSIM 17> run 1000

VSIM 18>
```

```
50 ns to 1050 ns    Project : Lab2   Now: 1 us  Delta: 2    sim:/tb_register_file
```

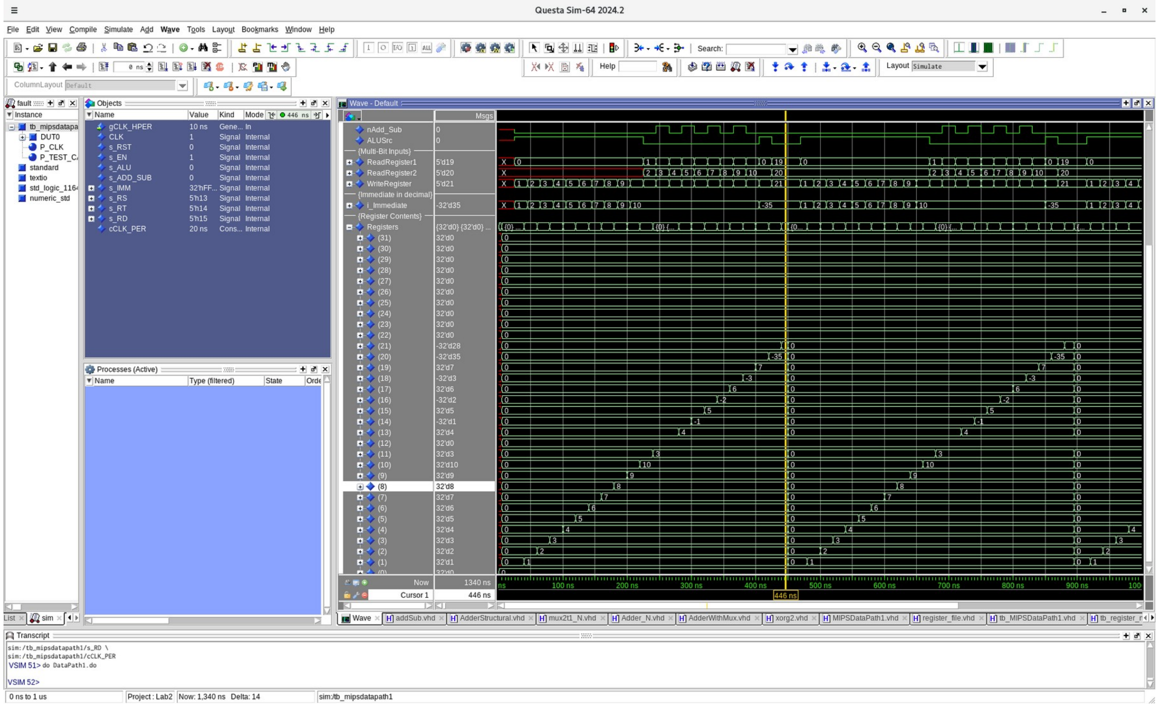
[Part 3 (b)] Draw a symbol for this MIPS-like datapath.



[Part 3 (c)] Draw a schematic of the simplified MIPS processor datapath consisting only of the component described in part (a) and the register file from problem (1).



[Part 3 (d)] Include in your report waveform screenshots that demonstrate your properly functioning design. Annotate what the final register file state should be.



Name	Value	Kind	Mode
gCLK_HPER	10 ns	Gene...	In
CLK	1	Signal	Internal
s_RST	0	Signal	Internal
s_EN	1	Signal	Internal
s_ALU	0	Signal	Internal
s_ADD_SUB	0	Signal	Internal
s_IMM	32'hFF...	Signal	Internal
s_RS	5'h13	Signal	Internal
s_RT	5'h14	Signal	Internal
s_RD	5'h15	Signal	Internal
cCLK_PER	20 ns	Cons...	Internal

	Msgs
nAdd_Sub	0
ALUSrc	0
{Multi-Bit Inputs}	
ReadRegister1	5'd19
ReadRegister2	5'd20
WriteRegister	5'd21
{Immediate in decimal}	
i_Immediate	-32'd35
{Register Contents}	
Registers	{32'd0} {32'd0} ...
(31)	32'd0
(30)	32'd0
(29)	32'd0
(28)	32'd0
(27)	32'd0
(26)	32'd0
(25)	32'd0
(24)	32'd0
(23)	32'd0
(22)	32'd0
(21)	-32'd28
(20)	-32'd35
(19)	32'd7
(18)	-32'd3
(17)	32'd6
(16)	-32'd2
(15)	32'd5
(14)	-32'd1
(13)	32'd4
(12)	32'd0
(11)	32'd3
(10)	32'd10
(9)	32'd9
(8)	32'd8
(7)	32'd7
(6)	32'd6
(5)	32'd5
(4)	32'd4
(3)	32'd3
(2)	32'd2
(1)	32'd1
(0)	32'd0
Now	1340 ns
Cursor 1	446 ns

```
sim:/tb_mipsdatapath1/s_RD \
sim:/tb_mipsdatapath1/cCLK_PER
VSIM 51> do DataPath1.do

VSIM 52>
```

0 ns to 1 us | Project : Lab2 | Now: 1,340 ns Delta: 14 | sim:/tb_mipsdatapath1

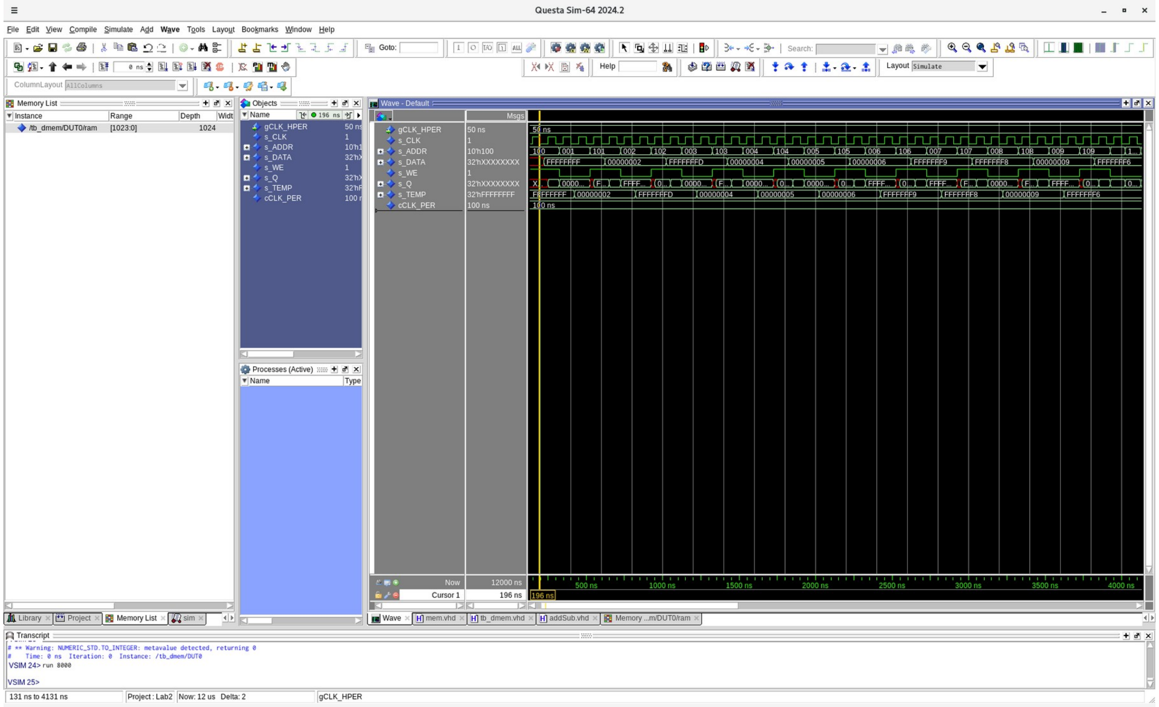
The file registers should be as follows:
$0 = 0
$1 = 1
$2 = 2

$3 = 3
$4 = 4
$5 = 5
$6 = 6
$7 = 7
$8 = 8
$9 = 9
$10 = 10
$11 = 3
$12 = 0
$13 = 4
$14 = -1
$15 = 5
$16 = -2
$17 = 6
$18 = -3
$19 = 7
$20 = -35
$21 = -28
$22 = 0
$23 = 0
$24 = 0
$25 = 0
$26 = 0
$27 = 0
$28 = 0
$29 = 0
$30 = 0
$31 = 0

[Part 4 (a)] Read through the mem.vhd file, and based on your understanding of the VHDL implementation, provide a 2-3 sentence description of each of the individual ports (both generic and regular).

Data width is for how wide the data that is being used is and then Addr width is for how wide the memory address is. Clk is for the clock to load and store data. Addr is for the address value to know where to load and read the data from. We is for enabling read or write. Q is for the output when reading the data.

[Part 4 (c)] Waveforms.

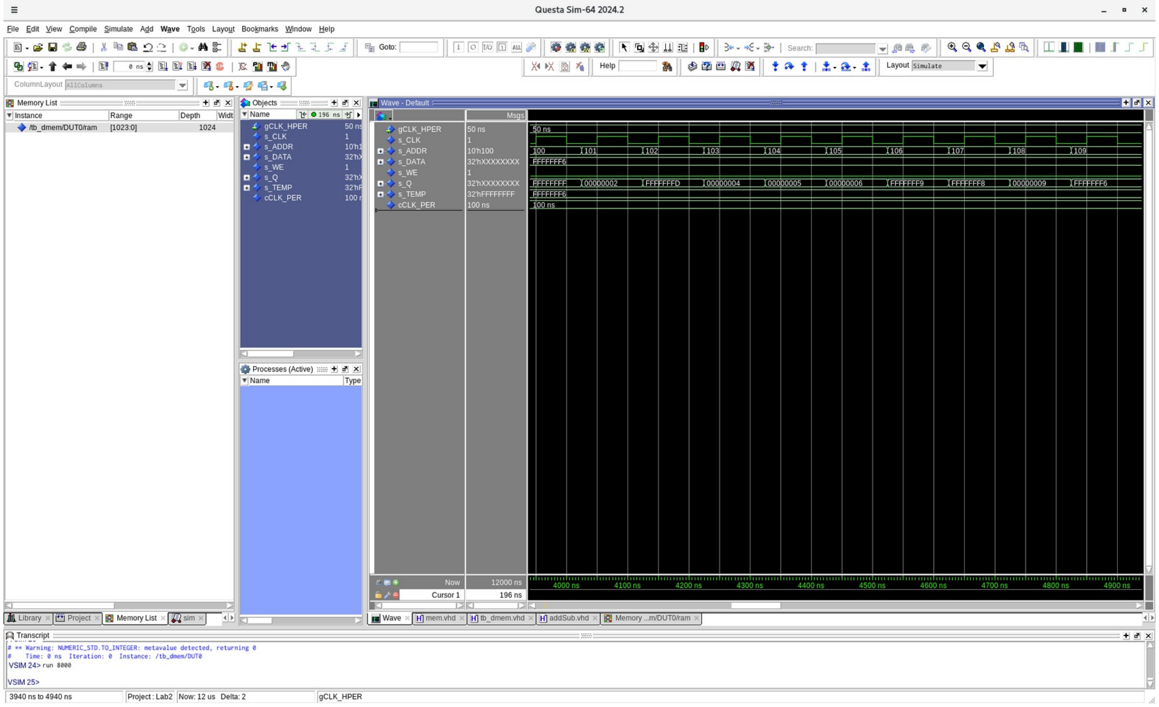

Questa Sim-64 2024.2
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Goto:
Search:
Help
Layout Simulate
ColumnLayout AllColumns
Memory List
Instance Range Depth Width
/tb_dmem/DUT0/ram [1023:0] 1024
Objects
Processes (Active)
Name Type
Wave - Default
Msgs
100 001 101 002 102 003 103 004 104 005 105 006 106 007 107 008 108 009 109
FFFFFFFF 00000002 FFFFFFFD 00000004 00000005 00000006 FFFFFFF9 FFFFFFF8 00000009 FFFFFFF6
500 ns 1000 ns 1500 ns 2000 ns 2500 ns 3000 ns 3500 ns 4000 ns
Library Project Memory List sim
Wave mem.vhd tb_dmem.vhd addSub.vhd Memory ...m/DUT0/ram
Transcript

Name	Msgs
gCLK_HPER	50 ns
s_CLK	1
s_ADDR	10'h100
s_DATA	32'hXXXXXXXX
s_WE	1
s_Q	32'hXXXXXXXX
s_TEMP	32'hFFFFFFFF
cCLK_PER	100 ns

Now 12000 ns
Cursor 1 196 ns

```
# ** Warning: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
#    Time: 0 ns  Iteration: 0  Instance: /tb_dmem/DUT0
VSIM 24> run 8000

VSIM 25>
```

131 ns to 4131 ns | Project : Lab2 | Now: 12 us Delta: 2 | gCLK_HPER


Questa Sim-64 2024.2
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Goto:
Search:
Help
Layout Simulate
ColumnLayout AllColumns
Memory List
Instance Range Depth Width
/tb_dmem/DUT0/ram [1023:0] 1024
Objects
Processes (Active)
Name Type
Wave - Default
Msgs
100 101 102 103 104 105 106 107 108 109
FFFFFFF6
FFFFFFFF 00000002 FFFFFFFD 00000004 00000005 00000006 FFFFFFF9 FFFFFFF8 00000009 FFFFFFF6
4000 ns 4100 ns 4200 ns 4300 ns 4400 ns 4500 ns 4600 ns 4700 ns 4800 ns 4900 ns
Library Project Memory List sim
Wave mem.vhd tb_dmem.vhd addSub.vhd Memory ...m/DUT0/ram
Transcript

Name	Msgs
gCLK_HPER	50 ns
s_CLK	1
s_ADDR	10'h100
s_DATA	32'hXXXXXXXX
s_WE	1
s_Q	32'hXXXXXXXX
s_TEMP	32'hFFFFFFFF
cCLK_PER	100 ns

Now 12000 ns
Cursor 1 196 ns

```
# ** Warning: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
#    Time: 0 ns  Iteration: 0  Instance: /tb_dmem/DUT0
VSIM 24> run 8000

VSIM 25>
```

3940 ns to 4940 ns | Project : Lab2 | Now: 12 us Delta: 2 | gCLK_HPER

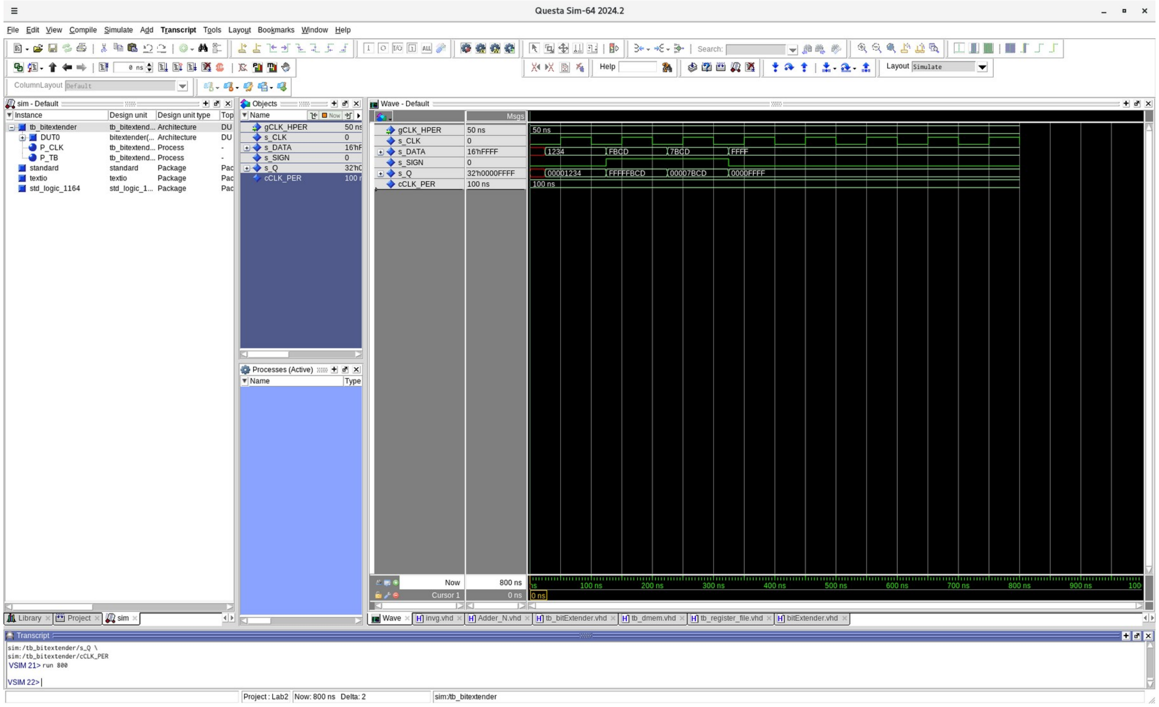
[Part 5 (a)] What are the MIPS instructions that require some value to be sign extended? What are the MIPS instructions that require some value to be zero extended?

Some instructions that require it to be extended are sltiu, stli, addiu, and addi and then some instructions that would be required to be zero extended are andi and ori.

[Part 5 (b)] what are the different 16-bit to 32-bit “extender” components that would be required by a MIPS processor implementation?

We would need one for the two add instructions and another for the less than instructions. Then you would need one for the load word instruction and another for the store word instruction.

[Part 5 (d)] Waveform.



Questa Sim-64 2024.2

Instance	Design unit	Design unit type	Top
tb_bitextender	tb_bitextend...	Architecture	DU
DUT0	bitextender(...	Architecture	DU
P_CLK	tb_bitextend...	Process	-
P_TB	tb_bitextend...	Process	-
standard	standard	Package	Pac
textio	textio	Package	Pac
std_logic_1164	std_logic_1...	Package	Pac

Name	Msgs	Wave
gCLK_HPER	50 ns	50 ns
s_CLK	0	
s_DATA	16'hFFFF	1234, FBCD, 7BCD, FFFF
s_SIGN	0	
s_Q	32'h0000FFFF	00001234, FFFFFBCD, 00007BCD, 0000FFFF
cCLK_PER	100 ns	100 ns
Now	800 ns	
Cursor 1	0 ns	0 ns

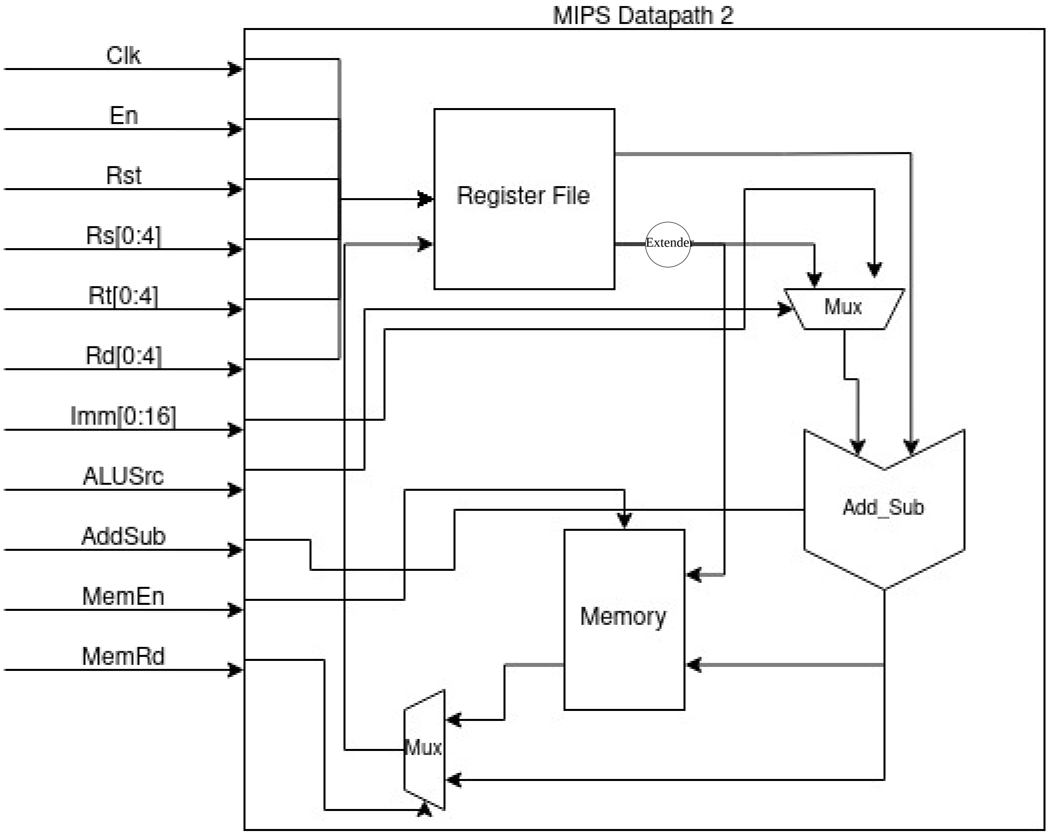
```
sim:/tb_bitextender/s_Q \
sim:/tb_bitextender/cCLK_PER
VSIM 21> run 800

VSIM 22>
```

Project : Lab2 | Now: 800 ns Delta: 2 | sim:/tb_bitextender

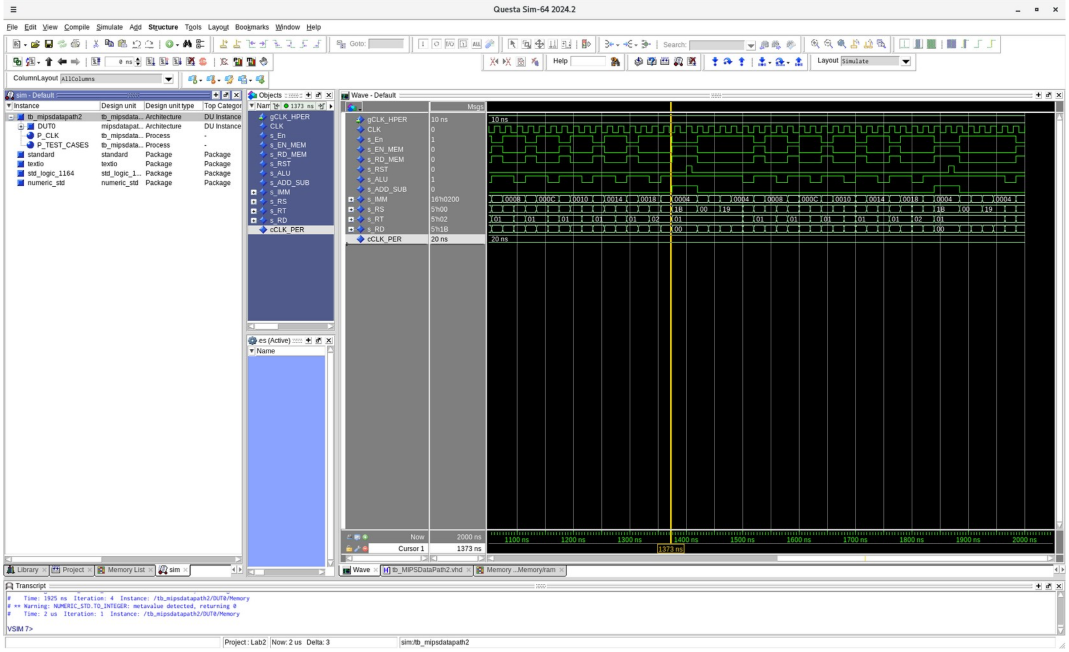
[Part 6 (a)] what control signals will need to be added to the simple processor from part 2? How do these control signals correspond to the ports on the mem.vhd component analyzed in part 3?

There will need to be an enable signal for writing to the memory and then another signal for reading from the memory.

[Part 6 (b)] Draw a schematic of a simplified MIPS processor consisting only of the base components used in part 2, the extender component described in part 4, and the data memory from part 3.



[Part 6 (c)] Waveform.



Instance	Design unit	Design unit type	Top Categor
tb_mipsdatapath2	tb_mipsdata...	Architecture	DU Instance
DUT0	mipsdatapat...	Architecture	DU Instance
P_CLK	tb_mipsdata...	Process	-
P_TEST_CASES	tb_mipsdata...	Process	-
standard	standard	Package	Package
textio	textio	Package	Package
std_logic_1164	std_logic_1...	Package	Package
numeric_std	numeric_std	Package	Package

	Msgs
gCLK_HPER	10 ns
CLK	0
s_En	1
s_EN_MEM	0
s_RD_MEM	0
s_RST	0
s_ALU	1
s_ADD_SUB	0
s_IMM	16'h0200
s_RS	5'h00
s_RT	5'h02
s_RD	5'h1B
cCLK_PER	20 ns
Now	2000 ns
Cursor 1	1373 ns

```
#    Time: 1925 ns  Iteration: 4  Instance: /tb_mipsdatapath2/DUT0/Memory
# ** Warning: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
#    Time: 2 us  Iteration: 1  Instance: /tb_mipsdatapath2/DUT0/Memory

VSIM 7>
```

Project : Lab2 | Now: 2 us Delta: 3 | sim:/tb_mipsdatapath2