# **CprE 381: Computer Organization and Assembly-Level Programming**

### **Project Part 2 Report**

Team Members: Andrew Wilken

Haris Khan

Wonjun Choi

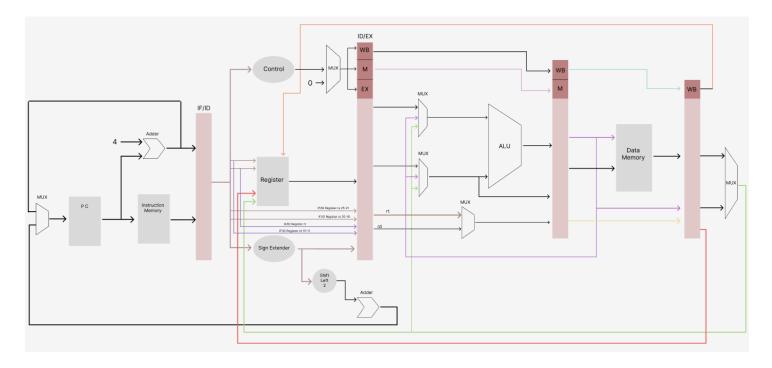
Project Teams Group #: 7-1

#### 1. Software-Scheduled Pipeline.

[1.a] Come up with a global list of the datapath values and control signals that are required during each pipeline stage.

IF	ID	EX	MEM	WB
s_Inst	s_IFIDinst	s_IDEXfunc	s_EXmemMux	s_WBdmem
iCLK	s_IFIDnewPC	s_IDEXwb	s_EXmemRegoMux	s_WBmemALUo
s_PCFour	s_aluControl	s_IDEXmem	s_EXmemNewPC	s_WBmemNewPC
s_IMemAddr	s_Branch	s_IDEXex	s_EXmemWB	s_RegWrAddr
s_NextInstAddr	s_beqS	s_IDEXreg1	s_EXmemMEM	s_WBmemOut
iRST	s_j	s_IDEXreg2	s_EXmemZero	s_Halt
ilnstLd	s_jr	s_IDEXrs	s_EXmemALUo	s_RegWr
ilnstAddr	s_jal	s_IDEXrt	s_EXmemHalt	s_tempram
iInstExt	s_memRead	s_IDEXrd	s_EXmemMux	s_RegWrData
	s_memToReg	s_IDEXnewPC	s_EXmemALUo	
	s_DmemWR_temp	s_IDEXextend	iCLK	
	s_aluSrc	s_IDEXinst	iRST	
	s_RegWr_temp	s_IDEXhalt		
	s_signExtend	s_RegWrA_temp		
	s_regDst	s_aluout		
	s_Halt_temp	s_ZERO		
	s_sub	s_AS		
	s_ignrOvfl	s_mux2out		
	s_movn			
	s_regout1			
	s_regout2			
	s_extended			
	s_WRmuxO			
	iCLK			
	iRST			
	s_RegValue			
	s_BaddO			
	s_BmuxO			
	s_JshiftO			
	s_JmuxO			
	s_JRmuxO			
	s_Carry			
	s_extended			
	s_WRmuxO			
	s_WRtemp			
	s_RegWrData			

#### [1.b.ii] high-level schematic drawing of the interconnection between components.



#### [1.c.i] include an annotated waveform in your writeup and provide a short discussion of result correctness.

```
bash-4.2$ ./381_tf.sh test proj/mips/Inst_With_Nop.s
Using LAB Python Environment
Testing
All VHDL src files compiled successfully
Testing file: proj/mips/Inst_With_Nop.s
Mars simulation: pass
Modelsim simulation: pass
Test Result: pass
Mars Instructions: 73
Processor Cycles: 80
CPI: 1.1
Results in: output/Inst_With_Nop.s
bash-4.2$
```

<u>→</u> s\_WBmemNewPC

We created a test that works with the pipeline processor because the base test of project part 1 lacked NOP instructions. For illustrational purposes, we have attached a waveform that just shows the pipeline stages rather than the whole waveform. If you look at the list of signals, we organized the pipeline stage in chronological order. The PC+4 value is passed from one pipeline step to the next, while the PC+8 value is passed from one pipeline step to the preceding. The whole waveform shown below includes more specific data and shows that the test was error-free. Signals

0040000C

00400008

00400010

00400000

00400014

00400010

that are undefined all the way through were unused for this portion, and since removed (no error).

00000000

00000000

32'h00400158

#### 

#### [Pipeline stages in our waveform ]

00400004

00400008

00400004

### [ Waveform: Base Tests with NOPs]

CLK	1																				
RST InstLd	0 0																				
InstAddr InstExt	32'hXXXXXXXX 32'hXXXXXXXX	XXXXXXXX																			
ALUOut _DMemWr	32'h00000064 0	00000000			7FFF00	00	00000000					7FFFE	FFC	00000	000					7FFFEFF	C
_DMemAddr s_DMemData	32'h0000000A 32'h0000000A	00000000					7FFF0000	00000	000							00000					
_DMemOut IMemAddr	32'hXXXXXXXX 32'h0040015C	XXXXXXXX 00400000		004	004000	08	0040000C	00400	010	004000	14	00400	018	00400	01C	00400	020	00400	024	0040002	8
 S_NextInstAddr s Inst	32'h0040015C 32'hXXXXXXXX	00400000 3C1D7FFF	004000	004	004000		0040000C	00400	010	004000	14	00400			01C	00400	020	00400	024	0040002 2009000	8
 _RegWr	0 5'h00	00	000000	,,,,				1D			,00					1D	020	100	000	2003000	
=RegWrAddr =RegWrData	32'h00000000	00000000						7FFF0	000	000000	000						FFC	00000	000		
_regWriteData _muxMemToRe	32'hXXXXXXXX 32'hXXXXXXXX	XXXXXXXX																			
s_mux1out s_movnOut	U U																				
_mux2out _mux3out	32'h00000064 32'hXXXXXXXX	00000000 XXXXXXXX			00007F	FF .	00000000					0000E	FFC	00000	000					7FFFEFF	·C
_Halt Ovfl	1 0							+-		$\vdash$											
 s_beqS s ZERO	0																				
	0																				
s_signExtend s_regDst	0																				
s_aluSrc s_memToReg	0																				
_memRead _Branch	0																				
	0																				
_ s_jr s_jal	0																				
s_movn s_aluControl	0 4'h0	E	3		E					Iв		E						0			
	32'h00000064	00000000			7FFF00		00000000	100400				7FFFE		00000		00400			020	7FFFEFF 0040002	
_PCFour _tempram	32'h00400160 32'h000000000	00000000	004000	,00	,004000	UC .	00400010	7FFF0	000	000000	000			00400		7FFFE				10040002	
regout1 regout2	32'h00400140 32'h00000000	00000000								7FFF00 7FFF00	000	00000	000							0000000	
_extended _jshifto	32'h00000008 32'h0F800020	00000000			000000					I 0000EF I 0EF7BF		00000						00004 00750		0000000	
_baddro _ALUWB	32'hXXXXXXXX 32'hXXXXXXXX	XXXXXXXX																			
_writePass _wo	5'hXX 5'hXX	XX																			
 S_JALmuxo S_JRmuxo	32'hXXXXXXXX 32'h00400140	XXXXXXXX 00000000	004000	004	004000	08	0040000C	00400	010	004000	14	00400	018	00400	01C	100400	020	00400	024	0040002	8
	0		00.000		00.000		00.00000	100.00		200,000				00.00		00.00	-	00.00		10010002	
_ ignrOvfl	0																				
_RegWr_temp _DmemWR_temp	0									Ш											
_IDEXhalt _EXmemZero	1																				
_EXmemHalt _RegValue	0 0																				
_Halt_temp _IDEXwb	0 3'h2	0	2																		
_IDEXmem s_EXmemWB	3'h0 3'h2	0			2																
_WBmemOut EXmemMEM	3'h0 3'h0	0					2														
 S_WRmuxO S_IDEXrs	5'h00 5'h00		1D		00					1D		00 1D		00				08		08	
IDEXrt	5'h09 5'h09	00			1D 1D		00					1D 1D		00						1D 08	
= RegWrA_temp = EXmemRegoM	5'h09 5'h08	00			1D		00 1D	00				1D		00 1D		00				08	
	5'h00	00	1D		00			,00		1D		00				.00		1D		08	
_IDEXfunc _IDEXex	5'h00 6'h10	00	02 2E		13		2E					00 1B		02 2E						20	
_IDEXreg1 _IDEXreg2	32'h00000000 32'h00000064	00000000										7FFF0 7FFF0	000	00000	000					7FFFEFF	
_IDEXextend _IDEXinst	32'h00000064 32'h20090064	00000000			3C1D7F	FFF	00000000					37BDE	FFC	00000	000					0000402 001D402	20
_IDEXnewPC _EXmemMux	32'h00400158 32'h0000000A	00000000			004000	04	00400008	00400	00C	004000	10	00400	014			00400 00000		00400	020	0040002	4
 _IFIDinst s IFIDnewPC	32'h03E00008 32'h0040015C	00000000	3C1D7		000000		0040000C	00400	010	37BDE		00000 00400				00400		001D4 00400		2008000	5
 _EXmemF EXmemALUo	32'hXXXXXXXX 32'h0000000A	XXXXXXXX 00000000					7FFF0000									00000					
EXmemNewPC WBdmem	32'h00400154 32'h00000000	00000000	XXXXX	XXX			00400004	00400	_	004000	00C	00400	010	00400				00400	01C	0040002	0
 _WBmemF	32'hXXXXXXXX	00000000	XXXXX	XXX	004000	00	00400010	00400	014	004000	110	00400	016	00400	020	00400	024	00400	020	0040002	<u> </u>
S_PCend S_WBmemALUo	32'h00400140 32'h000000000	00000000		,00	004000	00	50400010	7FFF0	000	000000	000					7FFFE	FFC		000		
s_WBmemNewPC s JmuxO	32'h00400150 32'h0040015C	00000000		h04	1004000	108	100400000	00400		004000				00400		00400		00400		10040001	
s_BaddO	32'h0040017C	00000000	00420	000	004000	800				0043C	004	00400	018					00410	0A4	0040003	C
s BshiftO	32'h00000020 32'h0040015C	00000000					0040000C	00400	010	0003B				00400	010	00400	020	100010		0000001	

[1.c.ii] Include an annotated waveform in your writeup of two iterations or recursions of these programs executing correctly and provide a short discussion of result correctness. In your waveform and annotation, provide 3 different examples (at least one data-flow and one control-flow) of where you did not have to use the maximum number of NOPs.

```
bash-4.2$ ./381_tf.sh test proj/mips/Bubblesort_With_Nop.s
Using LAB Python Environment
Testing
All VHDL src files compiled successfully
Testing file: proj/mips/Bubblesort_With_Nop.s
Mars simulation: pass
Modelsim simulation: pass
Test Result: pass
Mars Instructions: 2078
Processor Cycles: 2256
CPI: 1.09
Results in: output/Bubblesort_With_Nop.s
```

In contrast to project part 1, we used bubble sort mips code that included nop to test the pipeline.

In order to prevent errors at each step of the pipeline, we decided how many nops to use and placed them where they needed to be. As can be seen in the complete waveform screenshot below, all values are being located and transferred in the proper manner as a result.

#### [3 different examples in our waveform]



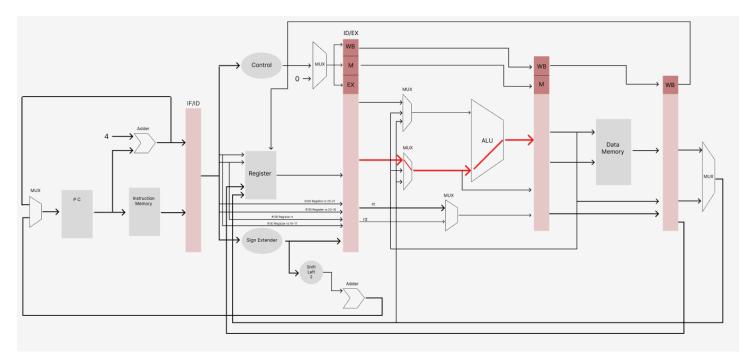
In this waveform you can see that it uses addi in succession without nops which is data flow and then it does jal which is control flow and these instructions are all used with either no nops or minimal nops.

### [ Waveform: Bubble sort with NOPs]

	0												
	0												
ilnstAddr	32'hXXXXXXXX	XXXXXXXX											
iInstExt	32'hXXXXXXXX	XXXXXXXX											
oALUOut	32'h00000000	(00000000					0000000A	00000000			00000028	00000000	
s_DMemWr	0												
s_DMemAddr	32'h0000000A	(00000000						0000000A	00000000			00000028	
s_DMemData	32'h00000001	(00000000										0000000A	1
s_DMemOut	32'h00000004	00000019						00000004				00000000	
s_IMemAddr	32'h004000FC	(00400000	00400004										00400
s_NextInstAddr	32'h004000FC	(00400000	00400004				004000C0	004000C4					
s_Inst	32'hXXXXXXXX	0810002D	00000000	20110000	2008000A	00000000			00082080	0C100002	00000000	20080004	20090
s_RegWr	0												
s_RegWrAddr	5'h00	(00				10	00	11	08	00			04
s_RegWrData	32'h00000000	(00000000							0000000A	00000000			00000
s_regWriteData	32'hXXXXXXXX	XXXXXXX											
s_muxMemToRe	32'hXXXXXXXX	XXXXXXX											
s_mux1out	U												
s_movnOut	U												
s_mux2out	32'h00000000	(00000000					0000000A	00000000			0000000A	00000000	
s_mux3out	32'hXXXXXXXX	XXXXXXX											
s_Halt	1												
s_Ovfl	0												
s_beqS	0												
s_ZERO	0												
s_Carry	0												
s_signExtend	0												
	0												
s_aluSrc	0												
	0												
s_memRead	0												
_	0												
s_j	0												
_	0												
	0												
_	0												
_	0												
s_aluControl	4'h0	0 (E	0	E	0		E				0		0
s_aluout	32'h00000000	(00000000						00000000			00000028		
s_PCFour	32'h00400100	(00400004	00400008	004000B8	004000BC	004000C0	004000C4	004000C8				0040000C	
s_tempram	32'h00000000	(00000000							0000000A	00000000			0000
s_regout1	32'h00000000	00000000											
s_regout2	32'h00000000	00000000								0000000A	00000000		0000
s_extended	32'h0000XXXX	(00000000	0000002D	00000000		0000000A	00000000			00002080	00000002	00000000	0000
s_jshifto	32'h0XXXXXXX	(00000000	004000B4	00000000	00440000	00200028	00000000			00208200	00400008	00000000	0020
s_baddro	32'hXXXXXXXX	XXXXXXXX											
s_ALUWB	32'hXXXXXXXX	XXXXXXXX											
s_writePass	5'hXX	XX											
s_wo	5'hXX	XX											
s_JALmuxo	32'hXXXXXXXX	XXXXXXX											
s_JRmuxo	32'h004000FC	(00000000	004000B4	00400008	004000B8	004000BC	004000C0	004000C4	004000C8	004000CC	00400008	004000D4	0040
s_AS	0												
s_sub	0												
s_ignrOvfl	0												
s_RegWr_temp	0												
s_DmemWR_temp	0												
s_IDEXhalt	0												
s_EXmemZero	1												
s_EXmemHalt	0												
s_RegValue	0												
s_Halt_temp	0												
s_IDEXwb	3'h0	(0	2	0	2							6	2
s_IDEXmem	3'h0	(0											
s_EXmemWB	3'h2	(0		2	0	2							6
s_WBmemOut	3'h0	(0					2						
s_EXmemMEM	3'h0	(0											
s_WRmuxO	5'hXX	(00	10	00	11	08	00			04	1F	00	08
s_IDEXrs	5'h00	(00											
s_IDEXrt	5'h00	(00		10	00	11	08	00			08	10	00
s_IDEXrd	5'h00	(00						00					00
s_RegWrA_temp	5'h00	(00		10				00					00
	5'h02	(00					11	08	00				1F
s_WRtemp	5'hXX	(00	10				00			08	1F		08
s_IDEXfunc	5'h00	(00				00		02					02
s_IDEXex	6'h00	(00				10		2E					2E
s_IDEXreg1	32'h00000000	(00000000											
s_IDEXreg2	32'h00000000	(00000000									0000000A	00000000	
s_IDEXextend	32'h0000000C	(00000000		0000002D	00000000		0000000A	00000000				00000002	
s_IDEXinst	32'h0000000C	(00000000			00000000							OC100002	
s_IDEXnewPC	32'h004000F8	(00000000		00400004	00400008	004000B8	004000BC	004000C0	004000C4			004000D0	
s_EXmemMux	32'h00000001	(00000000										0000000A	
s_IFIDinst	32'hXXXXXXXX	(00000000	0810002D	00000000	20110000	2008000A	00000000			00082080		00000000	
s IFIDnewPC	32'h004000FC	(00000000										004000D4	
s EXmemF	32'hXXXXXXXX	XXXXXXXX											
s EXmemALUo	32'h0000000A	(00000000						A000000A	00000000			00000028	0000
	32'h004000F4	(0000000			00400004	00400008	004000B8			004000C4	004000C8	004000CC	
s EXmemNewPC	32'h00000000	(00000000	00000019						00000004				0000
s_EXmemNewPC s_WBdmem	32'hXXXXXXXXX	(0000000	XXXXXXXX										
s_WBdmem		(00400004			004000BC	00400000	00400004	0040000	00400000	00400000	00400008	0040000C	0040
s_WBdmem s_WBmemF			00400004	00-100000	554000BC	30-1000000	30-1000C4	00 1000C8	0000000A				0000
s_WBdmem s_WBmemF s_PCend	32'h00400100	nnnnnnn							- AUTOMOBION A				1
s_WBdmem s_WBmemF s_PCend s_WBmemALUo	32'h00000000	(00000000				0040004	00400000	00400000					00.40
s_WBdmem s_WBmemF s_PCend s_WBmemALUo s_WBmemNewPC	32'h00000000 32'h004000F0	(00000000							004000BC	004000C0	004000C4	004000C8	
s_WBdmem s_WBmemF s_PCend s_WBmemALUo s_WBmemNewPC	32'h00000000 32'h004000F0 32'h004000FC	(00000000	004000B4		004000B8	004000BC	004000C0	004000C4	004000BC 004000C8	004000C0 004000CC	004000C4 00400008	004000C8 004000D4	004
s_WBdmem s_WBmemF s_PCend s_WBmemALUo s_WBmemNewPC s_JmuxO	32'h00000000 32'h004000F0	(00000000	004000B4 004000B8	00400008	004000B8 004000B8	004000BC	004000C0 004000C0	004000C4 004000C4	004000BC 004000C8	004000C0 004000CC 004082CC	004000C4 00400008 004000D8	004000C8 004000D4	004

[1.d] report the maximum frequency your software-scheduled pipelined processor can run at and determine what your critical path is (specify each module/entity/component that this path goes through).

Our software-scheduled pipelined processor can run at a maximum frequency of 53.58 MHz in 18.625 nanoseconds. Red lines in the diagram below denote critical routes. Specifically, starting with the ID/EX register, moving on to the ALU, EX/MEM register, and concluding.

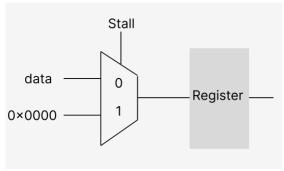


#### [Synthesis sw: timing.txt]

```
*timing.txt
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              ≣ _ ¤ ×
               Open ▼
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  Save
    # CprE 381 toolflow Timing dump
   FMax: 53.69mhz Clk Constraint: 20.00ns Slack: 1.37ns
The path is given below
      From Node
                                                                                                 : IDEX:id ex|dffg N:RegOut2|s Q[1]
                                                                                                               EXmem:ex mem|dffg N:RegALU|s Q[31]
         To Node
      Launch Clock : iCLK
         Latch Clock
      Data Arrival Path:
       Total (ns) Incr (ns)
                                                                                                                                                                                                                                 Element
                                                                                                                                                                                           Type
                                                                                                                                            == ==
                                         0.000
                                                                                                                      0.000
                                                                                                                                                                                                                                       launch edge time
                                                                                                                      3.090
                                                                                                                                                                                                                                         clock network delay
                                          3.090
                                                                                                                                                                                                                                    IDEX:id_ex|dffg_N:RegOut2|s_Q[1]
id ex|RegOut2|s Q[1]|q
                                         3.322
                                                                                                                      0.232
                                                                                                                                                                                           uTco
                                          3.322
                                                                                                                      0.000 FF
                                                                                                                                                                                          CELL
                                                                                                                                                                                                                                   5.078
                                                                                                                      1.756 FF
                                                                                                                      0.424 FF
                                            5.502
                                                                                                                                                                                          CELL
                                             5.752
                                                                                                                      0.250 FF
                                          5.877
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                            6.135
                                                                                                                      0.258
                                                                                                                                                                                                                                    \label{eq:mainLu} $$ \text{mainALU} = \text{addSub} = \text{fulladder} \\ $ \text{MainALU} = \text{addSub} = \text{fulladder} \\ $ \text{mainALU} = \text{addSub} \\ $ \text{mainALU} = \text{addSub}
                                         6.416
6.671
                                                                                                                      0.281 FF
                                                                                                                                                                                           CELL
                                                                                                                      0.255 FF
                                                                                                                                                                                                         IC
                                            6.952
                                                                                                                      0.281 FF
                                                                                                                                                                                                                                 mainALU|g_addSub|g_fulladder|\G_NBit_MUX:3:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:4:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:4:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:5:MUXI|g_or|o_F~0|datac mainALU|g_addSub|g_fulladder|\G_NBit_MUX:5:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:6:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:6:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:6:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:6:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:6:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:6:MUXI|g_or|o_F~0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|combout mainALU|g_addSub|g_fulladder|G_NBit_MUX:0|c
                                            7.202
                                                                                                                      0.250 FF
                                                                                                                                                                                                         IC
                                             7.327
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                            7.585
                                                                                                                      0.258 FF
                                             7.866
                                                                                                                      0.281 FF
                                                                                                                                                                                          CELL
                                            8.114
                                                                                                                      0.248 FF
                                                                                                                                                                                          CELL
                                          8.239
                                                                                                                      0.125 FF
                                            8.488
                                                                                                                      0.249 FF
                                                                                                                                                                                                                                 mainALU|g addSub|g fulladder|\G NBit MUX: 7:MUXI|g or|o F-0|datad mainALU|g addSub|g fulladder|\G NBit MUX: 8:MUXI|g or|o F-0|combout mainALU|g addSub|g fulladder|\G NBit MUX: 8:MUXI|g or|o F-0|combout mainALU|g addSub|g fulladder|\G NBit MUX: 8:MUXI|g or|o F-0|combout mainALU|g addSub|g fulladder|\G NBit MUX: 9:MUXI|g or|o F-0|combout mainALU|g addSub|g fulladder|\G NBit MUX: 9:MUXI|g or|o F-0|combout mainALU|g addSub|g fulladder|\G NBit MUX: 10:MUXI|g or|o F-0|combout mainALU|g addSub|g fulladder|G fulladder|g MUX: 10:MUXI|g or|o F-0|combout mainALU|g addSub|g fulladder|g fulladder|g fulladder|g fulladder|g fulladder|g fulladder|g fulladder|g fulladder|g full
                                          8.613
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                            8.863
                                                                                                                      0.250 FF
                                                                                                                                                                                                         IC
                                            8.988
                                                                                                                      0.125 FF
                                          9.383
                                                                                                                      0.395 FF
                                                                                                                                                                                                         IC
                                            9.508
                                          9.759
                                                                                                                      0.251 FF
                                                                                                                                                                                                         TC
                                          9.884
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                                                                                                                                                                                                                    \label{eq:mainalu} \begin{array}{ll} \text{mainalu} | g & \text{addSub} | g & \text{fulladder} | \langle G & \text{NBit} & \text{MUX}: 11: \text{MUXI} | g & \text{or} | \text{o} & \text{F} \sim \theta | \text{datad} \\ \text{mainalu} | g & \text{addSub} | g & \text{fulladder} | \langle G & \text{NBit} & \text{MUX}: 11: \text{MUXI} | g & \text{or} | \text{o} & \text{F} \sim \theta | \text{combout} \\ \text{mainalu} | g & \text{addSub} | g & \text{fulladder} | \langle G & \text{NBit} & \text{MUX}: 12: \text{MUXI} | g & \text{or} | \text{o} & \text{F} \sim \theta | \text{datad} \\ \end{array}
                                   10.135
                                                                                                                      0.251 FF
                                   10.260
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                    10.511
                                                                                                                      0.251 FF
                                                                                                                                                                                                                                 mainALU|g_addSub|g_fulladder|\G_NBit_MUX:12:MUXI|g_or|o_F-0|datad mainALU|g_addSub|g_fulladder|\G_NBit_MUX:12:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:13:MUXI|g_or|o_F-0|datad mainALU|g_addSub|g_fulladder|\G_NBit_MUX:13:MUXI|g_or|o_F-0|datad mainALU|g_addSub|g_fulladder|\G_NBit_MUX:14:MUXI|g_or|o_F-0|datac mainALU|g_addSub|g_fulladder|\G_NBit_MUX:15:MUXI|g_or|o_F-0|datad mainALU|g_addSub|g_fulladder|\G_NBit_MUX:15:MUXI|g_or|o_F-0|datad mainALU|g_addSub|g_fulladder|\G_NBit_MUX:15:MUXI|g_or|o_F-0|datad mainALU|g_addSub|g_fulladder|\G_NBit_MUX:15:MUXI|g_or|o_F-0|datad mainALU|g_addSub|g_fulladder|\G_NBit_MUX:15:MUXI|g_or|o_F-0|datad
                                   10.636
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                    10.885
                                                                                                                      0.249 FF
                                                                                                                     0.125 FF
0.255 FF
                                   11.010
                                                                                                                                                                                           CELL
                                   11.265
                                                                                                                                                                                                         IC
                                    11.546
                                                                                                                      0.281 FF
                                   11.794
                                                                                                                      0.248 FF
                                   11.919
                                                                                                                      0.125 FF
                                                                                                                                                                                                                                    \label{eq:mainalu} \begin{array}{ll} \texttt{mainalu} | \texttt{g} = \texttt{addSub} | \texttt{g} = \texttt{fulladder} | \texttt{G} = \texttt{NBit} = \texttt{MUX:} 16 : \texttt{MUXI} | \texttt{g} = \texttt{or} | \texttt{o} = \texttt{F-o} | \texttt{datac} \\ \texttt{mainalu} | \texttt{g} = \texttt{addSub} | \texttt{g} = \texttt{fulladder} | \texttt{G} = \texttt{NBit} = \texttt{MUX:} 16 : \texttt{MUXI} | \texttt{g} = \texttt{or} | \texttt{o} = \texttt{F-o} | \texttt{datac} \\ \texttt{mainalu} | \texttt{g} = \texttt{addSub} | \texttt{g} = \texttt{fulladder} | \texttt{G} = \texttt{NBit} = \texttt{MUX:} 17 : \texttt{MUXI} | \texttt{g} = \texttt{or} | \texttt{o} = \texttt{F-o} | \texttt{datac} \\ \end{bmatrix}
                                   12.176
                                                                                                                      0.257 FF
                                   12.457
                                                                                                                      0.281 FF
                                                                                                                                                                                           CELL
                                    12.712
                                                                                                                      0.255 FF
                                                                                                                                                                                                                                   mainALU|g_addSub|g_fulladder|\G_NBit_MUX:17:MUXI|g_or|o_F-0|combout
mainALU|g_addSub|g_fulladder|\G_NBit_MUX:18:MUXI|g_or|o_F-0|datad
mainALU|g_addSub|g_fulladder|\G_NBit_MUX:18:MUXI|g_or|o_F-0|combout
mainALU|g_addSub|g_fulladder|\G_NBit_MUX:19:MUXI|g_or|o_F-0|datad
                                   12.993
                                                                                                                      0.281 FF
                                                                                                                                                                                        CELL
                                   13.368
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                   13.618
                                                                                                                      0.250 FF
                                                                                                                                                                                                                                    13.743
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                   13.995
                                                                                                                      0.252 FF
                                                                                                                                                                                                         IC
                                    14.120
                                                                                                                      0.125 FF
                                                                                                                                                                                                                                   mainALU|g_addSub|g_fulladder|\G_NBit_MUX:20:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:21:MUXI|g_or|o_F-0|datac mainALU|g_addSub|g_fulladder|\G_NBit_MUX:21:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:22:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:22:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:23:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:23:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:23:MUXI|g_or|o_F-0|combout mainALU|g_addSub|g_fulladder|\G_NBit_MUX:24:MUXI|g_or|o_F-0|combout
                                   14.380
                                                                                                                      0.260 FF
                                                                                                                                                                                                         IC
                                    14.661
                                                                                                                      0.281 FF
                                                                                                                                                                                           CELL
                                   14.914
                                                                                                                      0.253 FF
                                   15.039
                                                                                                                      0.125 FF
                                                                                                                                                                                          CELL
                                                                                                                      0.298 FF
                                    15.337
                                   15.761
                                                                                                                      0.424 FF
                                                                                                                                                                                          CELL
                                                                                                                      0.250 FF
                                   16.011
                                                                                                                                                                                                         IC
                                                                                                                                                                                                                                    16.136
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                   16.559
                                                                                                                      0.423 FF
                                                                                                                                                                                                         IC
                                    16.840
                                                                                                                      0.281 FF
                                                                                                                                                                                                                                   \label{eq:mainLU} \begin{array}{ll} \text{mainALU} | g \text{ addsub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 26: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$d$atad} \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 26: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 27: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 27: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 27: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 27: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 27: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{MUX}: 27: \text{MUXI} | g \text{ or} \mid o \text{ } \text{$F$\sim$0} \mid \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \setminus G \text{ NBit } \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{mainALU} | g \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} | g \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} \mid G \text{ fulladder} \mid G \text{ NBit } \text{$c$ombout } \\ \text{ addSub} \mid G \text{ fulladd
                                   17.090
                                                                                                                      0.250 FF
                                                                                                                                                                                                         IC
                                    17.215
                                                                                                                      0.125 FF
                                   17.465
17.590
                                                                                                                      0.250 FF
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                    17.828
                                                                                                                      0.238 FF
                                                                                                                                                                                                                                      mainALU|g_addSub|g_fulladder|\G_NBit_MUX:28:MUXI|g_or|o_F~0|datad
                                                                                                                                                                                                                                    \label{eq:mainalu} \begin{array}{ll} \text{mainalu} \mid g \quad \text{addSub} \mid g \quad \text{fulladder} \mid \backslash G \quad \text{NBit} \quad \text{MUX}: 28: \text{MUXI} \mid g \quad \text{or} \mid \text{o} \quad \text{F} \sim \theta \mid \text{combout} \\ \text{mainalu} \mid g \quad \text{addSub} \mid g \quad \text{fulladder} \mid \backslash G \quad \text{NBit} \quad \text{MUX}: 29: \text{MUXI} \mid g \quad \text{or} \mid \sigma \mid \text{F} \sim \theta \mid \text{datac} \\ \end{array}
                                   17.953
                                                                                                                      0.125 FF
                                                                                                                                                                                           CELL
                                   18.211
                                                                                                                      0.258
                                                                                                                                                                                                                                    \label{eq:mainalu} \begin{array}{ll} \text{mainalu} \mid g \mid \text{addSub} \mid g \mid \text{fulladder} \mid \backslash G \mid \text{NBit} \mid \text{MUX}: 29: \text{MUXI} \mid g \mid \text{or} \mid \text{o} \mid \text{F-o} \mid \text{combout} \\ \text{mainalu} \mid g \mid g \mid \text{dot} \mid \text{dot} \mid \text{dot} \mid \text{MUX}: 30: \text{MUXI} \mid g \mid \text{or} \mid \text{o} \mid \text{F-o} \mid \text{combout} \\ \text{mainalu} \mid g \mid g \mid \text{dot} \mid \text{dot
                                   18.492
                                                                                                                      0.281 FF
                                                                                                                                                                                           CELL
                                   18.727
                                                                                                                      0.235 FF
                                                                                                                                                                                                         IC
                                   18.852
                                                                                                                      0.125 FF
                                                                                                                                                                                                                                    \label{eq:mainALU} $$ {\bf mainALU} = {\bf addSub} = {\bf fulladder} = {\bf NBitMUX:31:MUXI} = {\bf xor2} = {\bf fulladder} = {\bf MUX:31:MUXI} = {\bf xor2} = {\bf fulladder} = {\bf fulladder}
                                   19.082
                                                                                                                      0.230 FF
                                                                                                                      0.150 FR
                                                                                                                                                                                        CELL
                                                                                                                                                                                                                                    mainALU|g_mux9|Mux0~5|datad
mainALU|g mux9|Mux0~5|combout
                                   19.458
                                                                                                                      0.226 RR
                                                                                                                                                                                                         IC
                                   19.613
                                                                                                                      0.155 RR
                                                                                                                                                                                          CELL
                                                                                                                                                                                                                                    mainALU|g_mux9|Mux0~6|datac
mainALU|g_mux9|Mux0~6|combout
mainALU|g_mux9|Mux0~7|datad
                                   20.878
                                                                                                                      1.265 RR
                                                                                                                                                                                                         IC
                                                                                                                      0.287 RR
                                                                                                                                                                                        CELL
                                   21.165
                                   21.367
                                                                                                                      0.202 RR
                                                                                                                                                                                                                                   mainALU|g_mux9|Mux0~7|combout
ex_mem|RegALU|s_Q[31]|d
                                   21.506
                                                                                                                      0.139 RF
                                                                                                                                                                                          CELL
                                   21.506
                                                                                                                      0.000 FF
                                   21.610
                                                                                                                      0.104 FF
                                                                                                                                                                                          CELL
                                                                                                                                                                                                                                      EXmem:ex_mem|dffg_N:RegALU|s_Q[31]
      Data Required Path:
          Total (ns)
                                                                                                                                                                                              Type
                                   20.000
                                                                                                               20.000
                                                                                                                                                                                                                                       latch edge time
                                   22.977
                                                                                                                      2.977
                                                                                                                                                                                                                                       clock network delay
                                   22.985
                                                                                                                      0.008
                                                                                                                                                                                                                                       clock pessimism removed
                                   22.965
                                                                                                                     0.020
                                                                                                                                                                                                                                    EXmem:ex mem|dffg N:RegALU|s Q[31]
                                   22.983
                                                                                                                     0.018
                                                                                                                                                                                           uTsu
       Data Arrival Time
                                                                                                                                                                               21.610
       Data Required Time :
                                                                                                                                                                               22.983
       Slack
                                                                                                                                                                                     1.373
```

#### 2. Hardware-Scheduled Pipeline.

[2.a.ii] Draw a simple schematic showing how you could implement stalling and flushing operations given an ideal N-bit register.

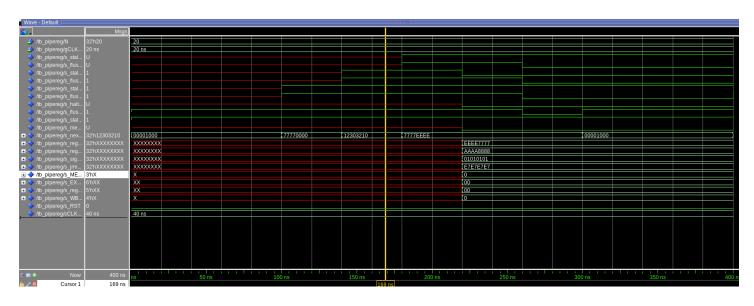


[2.a.iii] Create a testbench that instantiates all four of the registers in a single design. Show that values that are stored in the initial IF/ID register are available as expected four cycles later, and that new values can be inserted into the pipeline every single cycle. Most importantly, this testbench should also test that each pipeline register can be individually stalled or flushed.

```
stallIF => s_stallIF,
stallID => s_IDEXstall,
stallEX => s_EXmemstall,
stallWB => s_memWBstall,
flushIF => s_IFIDflush,
flushID => s_IDEXflush,
flushEX => s_EXmemflush,
flushWB => s_memWBflush,
flushtemp => s_flush2
```

A testbench file exists but we included additional waveforms to show correctness. Below are the four register values when running grendel.s as well as the testbench (tb is first). As you can see, the values get passed down the pipeline as expected through all stages. Each stage can be stalled or flushed with the following code in the hazard unit. Since each stage has its own stall and flush, this is possible.

#### [ Pipeline Register Waveform ]



Wave - Default												<b>+</b>
<b>S</b>	Msgs											
	32'hXXXXXXXX	08100016	20080004	20110000	2008000A	00082080	0C1000	20490000	20080004			2009
/tb/MyMips/if_id/i_CLK		$\Rightarrow$ $\vdash$										┨
/tb/MyMips/if_id/i_flush /tb/MyMips/if_id/i_stall												
+ // /tb/MyMips/if_id/i_newPC	32'h0040008C	00400004	00400008	004000	00400060	00400064	00400068	004000	00400004		00400008	0040
** /tb//MyMips/if id/o inst	32'hXXXXXXXXX		08100016								00400008	2008
/tb/MyMips/if_id/o_newPC	32'h00400088		00400004							00400004		0040
	32'hXXXXXXXX	08100016	00000000									2009
	32'hXXXXXXXX	00000000	08100016	00000000	20110000	200800	00082080	0C1000	00000000			2008
/tb/MyMips/if_id/s_tempWE												
		2	0	2				6	2			
	3'h0	0										
	6'h00	2E	00	2E	10		2E	00	2E			10
7												
/tb/MyMips/id_ex/i_flush												
/tb/MyMips/id_ex/i_stall // dy/MyMips/id_ex/i_reg1out	32'h00000000	00000000										
+ 4 /tb/MyMips/id_ex/i_reg2out	32'h00000000	00000000										0000
# 4 /tb/MyMips/id ex/i_NewPC	32'h00400088	00000000	00400004	00400008	004000	00400060	00400064	00400068	004000	00400004		0040
	32'hXXXXXXXX	00000000						0C1000				2008
	32'h0000XXXX	00000000	00000016					00000001				0000
	5'hXX	00	10	00	11	08	04	1F	00			08
	32'h00400088	00000000	00400058	00400008	004000	00400060	00400064	00400004	004000	00400004		0040
/tb/MyMips/id_ex/i_memre												
/tb/MyMips/id_ex/i_CLK												
/tb/MyMips/id_ex/i_RST	0				00							00
	5'h00				00		02		02	02		00
	5'h00 4'h0	00	02 2		02 2	00				02 2		
	4'h0 3'h0	0	<u>د</u>	. 0								
** /tb/MyMips/id_ex/o_MEMSig	6'h00	00	2E	00	2E	10		2E	00	2E		
	32'h00000000	00000000										
	32'h00000000	00000000										
	5'hXX	00										
	32'h00400084	00000000		00400004	00400008	004000	00400060	00400064	00400068	004000	00400004	
/tb/MyMips/id_ex/o_RegRt  tb/MyMips/id_ex/o_RegRt	5'hXX	00		10	00	11	08		10	00		
#	32'h0000XXXX	00000000		00000016	00000000		0000000A	00002080	00000001	00000000		
	5'hXX	00						1	00			
_ ,	5'hXX	00				11	08	1	1F	00		
+ /tb/MyMips/id_ex/o_inst	32'hXXXXXXXX	00000000						00082080				
/	32'h00400084	00000000		00400058	00400008	004000	00400060	00400064	00400004	004000	00400004	
· · · · · · · · · · · · · · · · · · ·												
· · · · · · · · · · · · · · · · · · ·												
/tb/MyMips/id_ex/tmp_we /tb/MyMips/id_ex/tmp_halt												
/tb/MyMips/id_ex/tmp_man												
	32'hXXXXXXXX	00000000	08100016	00000000	20110000	200800	00082080	0C1000	00000000			2008
	32'h0000XXXX		00000016					00000001				0000
# // /tb/MyMips/id_ex/tmp_reg1	32'h00000000	00000000										
	32'h00000000	00000000										0000
	32'h00400088	00000000	00400004	00400008	004000	00400060	00400064	00400068	004000	00400004		0040
	32'h00400088	00000000	00400058	00400008	004000	00400060	00400064	00400004	004000	00400004		0040
	5'hXX	00										
	5'hXX		10	00	11	08			00			08
	5'hXX	00	10	00	11	00	04	00	00			V 00
	5'hXX 4'h0			00 2	11	08			00 2			08
	6'h00		1——		10				2E			10
	3'h0	0										
tb/MyMips/ex_mem/i_alu	32'h00000000	00000000					0000000A	00000000				
/tb/MyMips/ex_mem/i_CLK												
	4'h0	0	2	0	2				6	2		
	3'h0	0										
	32'h00000000	00000000						00000000				
	5'hXX	00								00	004000	
	32'h00400084 32'h00400084	00000000			00400008			00400064 00400064				
	32'h00400084 0	00000000		00400058	00400008	004000	,00400060	,00400064	00400004	004000	00400004	
	5'hXX	00						04	00			
/tb/MyMips/ex_mem/i_flush												
/tb/MyMips/ex_mem/i_stall												
	4'h0	0		2	0	2				6	2	
		0										
· · · ·												
	32'h00000000	00000000						0000000A				
** /tb/MyMips/ex_mem/o_Ex	32'h00000000	00000000				_		0000000A				
	5'hXX	00			10	00	11	08			00	0040
	32'h00400080	00000000			00400004	00400008	,004000	00400060	00400064	00400068	,004000	,0040
	5'hXX	00			00400058	0040000	004000	00400060	00400064	00400004	004000	0040
	32'h00400000											10040
** /tb/MyMips/ex_mem/o_jrm	32'h00400080 0	00000000			00400000	00400000	004000	,00400000	00400004	00400004	004000	
/tb/MyMips/ex_mem/o_jrm /tb/MyMips/ex_mem/o_me	32'h00400080 0 0	00000000			,00400036			0040000	00400004	00400004	, 004000	

/tb/MyMips/mem_wb/i_flush	1												
/tb/MyMips/mem_wb/i_stall	1												
/tb/MyMips/mem_wb/i_halt	0												
/tb/MyMips/mem_wb/i_CLK	0												
/tb/MyMips/mem_wb/i_RST	0												
	4'h0	0			2	0	2				6	2	
	5'h00	00											
	32'h00000000	00000000				00400058	00400008	004000	00400060	00400064	00400004	004000	0040.
/tb/MyMips/mem_wb/i_me	0												
★ /tb/MyMips/mem_wb/o_dm	32'h00000000	00000000		00000019						00000004	00000019		
🕳 💠 /tb/MyMips/mem_wb/o_alu	32'h00000000	00000000								0000000A	00000000		
	32'h00000000	00000000					00400004	00400008	004000	00400060	00400064	00400068	0040.
	5'h00	00					10	00	11	08	04	1F	00
	5'h00	00											
	4'h0	0				2	0	2				6	2
	32'h00000000	00000000					00400058	00400008	004000	00400060	00400064	00400004	0040.
/tb/MyMips/mem_wb/o_me	0												
/tb/MyMips/mem_wb/o_halt	0												
/tb/MyMips/mem_wb/tmp_we	1												
/tb/MyMips/mem_wb/tmp_h	0												
/tb/MyMips/mem_wb/tmp	0												
	32'h00000000	00000000							0000000A				
	32'h00000019	00000019							00000004				
	32'h00000000	00000000				00400004	00400008				00400068		0040.
_ , , , , , , , , , , , , , , , , , , ,	32'h00000000	00000000				00400058	00400008	004000	00400060	00400064	00400004	004000	0040.
	5'h00	00				10	00	11	08	04	1F	00	
	5'h00	00											
+ / /tb/MyMips/mem_wb/tmp	4'h0	0			2	0	2				6	2	
△ 🛒 🏵 Now	980 ns	1 1 1	1	50 ns	1 1	100	l i i i ) ns	1 1 1	150 ns	1 1 1	200	lııı Ons	1   1

#### [2.b.i] list which instructions produce values, and what signals (i.e., bus names) in the pipeline these correspond to.

Instr	Signals
add, addu, sub, subu, sra, srl, sll, and, or, nor, xor, slt, addi, addiu, jr, andi, ori, xori, lui, slti	s_RegWrA_temp, s_writeAddrEX, s_RegWrAddr, s_EXmemReadWR, s_memWBReadWR
j, jr, jal	s_JRmuxO, s_IDEXJRmuxO, s_EXmemJRmuxO, s_JALmuxo, s_JRmuxo
bne, beq	s_exmembranchaddr

#### [2.b.ii] List which of these same instructions consume values, and what signals in the pipeline these correspond to.

Instr	Signals
add, addu, sub, subu, sra, srl, sll, and, or, nor, xor, slt, addi, addiu, jr, andi, ori, xori, lui, slti	i_alu_A, i_alu_B

## [2.b.iii] generalized list of potential data dependencies. From this generalized list, select those dependencies that can be forwarded (write down the corresponding pipeline stages that will be forwarding and receiving the data), and those dependencies that will require hazard stalls.

forwarded	require hazard stalls
s_RegWrA_temp, s_writeAddrEX, s_RegWrAddr,	s_EXmemReadWR, s_memWBReadWR

### [2.b.iv] global list of the datapath values and control signals that are required during each pipeline stage

IF	ID	EX	MEM	WB
s_Inst	s_IFIDinst	s_IDEXfunc	s_EXmemMux	s_WBdmem
iCLK	s_IFIDnewPC	s_IDEXwb	s_EXmemRegoMux	s_WBmemALUo
s_PCFour	s_aluControl	s_IDEXmem	s_EXmemNewPC	s_WBmemNewPC
s_IMemAddr	s_Branch	s_IDEXex	s_EXmemWB	s_RegWrAddr
s_NextInstAddr	s_beqS	s_IDEXreg1	s_EXmemMEM	s_WBmemOut
iRST	s_j	s_IDEXreg2	s_EXmemZero	s_Halt
ilnstLd	s_jr	s_IDEXrs	s_EXmemALUo	s_RegWr
ilnstAddr	s_jal	s_IDEXrt	s_EXmemHalt	s_tempram
iInstExt	s_memRead	s_IDEXrd	s_EXmemMux	s_RegWrData
s_IFIDflush	s_memToReg	s_IDEXnewPC	s_EXmemALUo	s_memWBjrmuxO
s_stallIF	s_DmemWR_temp	s_IDEXextend	iCLK	s_memWBReadWR
s_PCstallAddr	s_aluSrc	s_IDEXinst	iRST	s_memwbrd
	s_RegWr_temp	s_IDEXhalt	s_EXmemReadWR	
	s_signExtend	s_RegWrA_temp	s_EXmemrd	
	s_regDst	s_aluout	s_tempForwardRegO	
	s_Halt_temp	s_ZERO	s_memwbflush	
	s_sub	s_AS	s_memwbstall	
	s_ignrOvfl	s_mux2out	s_DMemWr	
	s_movn	s_IDEXJRmuxO	s_DMemData	
	s_regout1	s_IDEXReadWRmen	s_DMemAddr	
	s_regout2	s_IDEXhalt		
	s_extended	s_EXmemFlush		
	s_WRmuxO	s_flush2		
	iCLK	s_EXmemStall		
	iRST	s_IDEXReadWRmen	n	
	s_RegValue	s_forwardBMuxOut		
	s_BaddO	s_forwardB		
	s_BmuxO	s_forwardA		
	s_JshiftO	s_forwardAMuxOut		
	s_JmuxO	s_forwardBranch2		
	s_JRmuxO	s_forwardBranch		
	s_Carry			
	s_extended			
	s_WRmuxO			
	s_WRtemp			
	s_RegWrData			
	s_IDEXflush			
	s_IDEXstall			
	s_ReadWRmem			
	s_regequal			
	s_temp_regout2			
	s_temp_regout1			

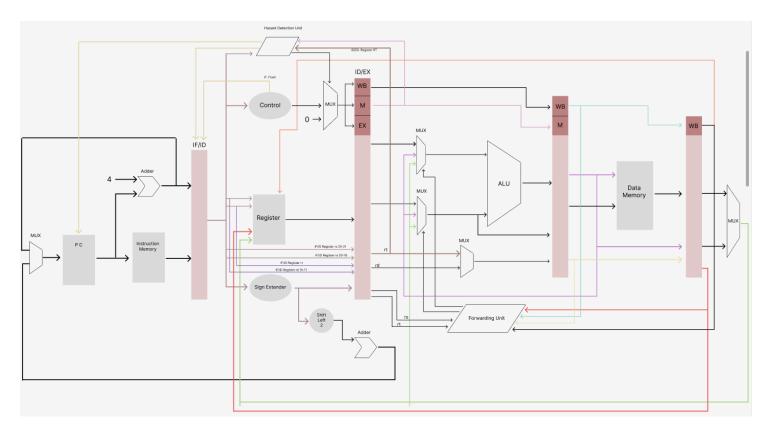
#### [2.c.i] list all instructions that may result in a non-sequential PC update and in which pipeline stage that update occurs.

instr	pipeline stage
bne, beq, j, jr, jal	wb

[2.c.ii] For these instructions, list which stages need to be stalled and which stages need to be squashed/flushed relative to the stage each of these instructions is in.

All the pipeline stages must stall while these instructions are in WB stage. The processor won't load an instruction if there are existing branching or jumping instructions in the pipeline because we positioned our Hazard Detection module inside the Fetch stage. Therefore, there is no need to flush stages, and the processor will stall itself while it waits.

[2.d] implement the hardware-scheduled pipeline using only structural VHDL. As with the previous processors that you have implemented, start with a high-level schematic drawing of the interconnection between components.



[2.e-i, ii, and iii] In your writeup, show the Modelsim output for each of the following tests, and provide a discussion of result correctness. It may be helpful to also annotate the waveforms directly.

[Grendel Test] + 31 XXXXXX 10010000 10011000 00000000 00400000 00400018 00000000 00400018 10010FD8 10010FFC 10010... 0000000 10010000 10011000 00000000 00400000 00400018 00000000 00400018 10010FD8 10010... 00000000 00001001 00001000 00000000 00000040 00000018 00000000 00400018 FFFFFD8 00000. 00400000 100000000 1FFFFFFF 100000000 00400004 00400008 0040000C 00400010 00400014 00400018 0040001C 00400020 00400024 343D1000 241E0000 3C010040 343F0018 008100007 50000000 27BDFF... AFBF0024 AFBE0020 0040000 3C01100 00 10010000 10011000 00000000 00400000 00400018 00000000 00400018 10010... 00000000 00000000 T00001001 T00001000 T00000000 T00000040 T00000018 T00000000 IFFFFFFD8 I00000024 I00000... 101 11D 100 11F 01 1D 01 1F 10 00 1D 00 00 11D ITE 11F 100 11D 13 1B 10 13 00000000 10010000 000000000 10011000 10010000 00000000 10011000 00400018 00000 00001001 00001000 00000000 00000040 00000018 00000007 00000000 FFFFFFD8 00000024 00000 0000000 00400004 00400008 0040000C 00400010 00400014 0040018 0040001C 00400020 00400024 00001001 00001001 00000000 00000004 00000018 00000000 00400018 FFFFFFB 00000... 0000000 00000000 00000000 00000000 | 00400004 | 00400008 | 00400004 | 00400008 | 00400005 | 00400010 | 00400014 | 00400018 | 00400015 | 00400 | 00400004 | 00400008 | 00400005 | 00400010 | 00400014 | 00400015 | 00400015 | 00400024 | 00400024 | 00400024 | 00400024 | 00400024 | 00400024 | 00400040 | 00400005 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 00400010 | 0

> s_WRtemp	5'h00	00	01	1D	↓1E	01	Ĭ1F	10	00	1D	1F	00	
s IDEXfunc	5'h00	00	02		00	02		00	08	02		00	02
s IDEXex	6'h00	00	2E	13	1B	10	13	1B	00	2E	10		2E
s IDEXreg1	32'	00000000							00000000		10011000		000
s IDEXreg2	32'	00000000					10010000	00000000	1		10011000	00400018	7
s IDEXextend	32'	00000000		00001001	00001000	00000000		00000018					
s IDEXinst	32'	00000000						343F0018					
s IDEXnewPC	32'	00000000		1	1	1	1	00400014	1	1		1	$\overline{}$
s EXmemMux	32'	00000000			1	1		00000040		1			
s IFIDinst	32'	00000000	3C011001	343D1000	î .	1		08100007		1		1	
s IFIDnewPC	32'	00000000						00400018					
s EXmemALUo	32'	00000000	00400004	00400000				00400000				10010ED8	10
s EXmemNewPC	32'	00000000			1	1		00400010		1			1
s WBdmem	32'	00000000	FFFFFFF		00400004	00400008	0040000C	100400010	00400014		FFFFFFF		100
s_wbuillelli s PCend	32'	00400004		00400000	00400010	00400014	00400010	10040001C	00400000	1		00000000	
s_PCend s WBmemALUo	32'	00000000	00400008	,0040000C	00400010	1		1000000000	1	1		00400018	10
	32'	00000000						10040000C					
s_WBmemNewPC s JmuxO	32'	00000000	00400004	00400000	00400000			10040000C				,0040001C	UU
				1	1	1	1	1	1	1		00400004	
s_BaddO	32'	00000000						00400034					
s_BshiftO	32'							0000001C				00000000	
s_BmuxO	32'	00000000	00400004	00400008	0040000C	00400010	00400014	00400018	.0040001C	00400020	00400024		
s_stallIF	1												
s_IFIDflush	1												
s_readwritemem	0												
s_EXmemstall	1												
s_EXmemflush	1												
s_IDEXstall	1												
s_IDEXflush	1												
s_memWBflush	1												
s_memWBstall	1												
s_forwardA	2'h0	0			2	0		2	0	1	0	2	1
s_forwardB	2'h0	0								1	0		1
s_forwardBranch	2'h0	0		1	0		1	0			1	0	
s_forwardBranch2	2'h0	0		0	0	0	0	lo			0		
s_IDEXhalt	0												
s_PCstallAddr	32'	00000004									00000000		
s memWBrd	5'h00	00				02	00						
s_EXmemRD	5'h00	00			02	00							
s idexmemreadw	0												
s exmemreadwrite	0												
s memwbreadwrite													
s idexjrmuxout	32'	00000000		00400004	00400008	0040000C	00400010	00400014	0040001C		00400020	00400024	
s exmemjrmuxout	32'	00000000						00400010				00400020	
s memwbjrmuxout	32'	00000000			33100007			0040001C					00
s_temp_regout2	32'	00000000				1	00000000	1	1	10011000			_
s_temp_regout1	32'	00000000				, 1U		100000000		10011000	00400018	00000000	
	32'	00000000			10010000	100000000	10010000			00400018	10011000		
<ul><li>s_forwardAMuxOut</li><li>s_forwardBMuxOut</li></ul>	32'	00000000		00001001	1		000000040						
				TOOTOOT	00001000	100000000	100000040	00000018	100000000	100400018	FFFFFD8	00000024	LU.

#### [ Bubble Sort without Nops Test ]

Wave - Default ====			[ D	ubbie	SUIT W	ithout .	Moh2 1	est j				<b>+</b> e
wave - Delauit	Msgs											
	1											
→ iRST	0											
	0											
	32'	XXXXXXXX										
	32'	0000000	n					0000000A	00000000			
→ s_DMemWr	0	10000000						1000000A	,00000000			
	32'	0000000	0						0000000A	00000000		
<u>■</u> ◆ s_DMemData	32'	0000000	0							00000000		
	32'	00000019							00000004	00000019		
s_IMemAddr	32'	(0040000				0040005C			00400068	00400004		
s_NextInstAd     s_Inst	32'	08100016				2008000A			1			
s_RegWr	0	00100010		20000001	20110000	20000000	000002000	00100001	20100000	20000001		
	5'h00	00					10	00	11	08	04	1F
★ s_RegWrData	32'	0000000	0							0000000A	00000000	0040.
	32'	XXXXXXXX										
s_muxMemT	32' U	XXXXXXXX										
s_mux1out s_movnOut	U											
	32'	0000000	0					0000000A	00000000			
	32'	XXXXXXXX										
s_Halt	1											
s_Ovfl	0											
s_beqS	0											
	0											
s_signExtend	0											
s_regDst	0											
s_aluSrc	0											
→ s_memToReg	0											
s_memRead	0											
s_Branch	0				+							
s_ANDo	0											
→ s_jr	0											
s_jal	0											
s_movn	0											
s_aluControl	4'h0	0 (E		0	ΪE	0		E	0	ΪΕ		_
	32'	(0000000		00400008	00400050	00400060	00400064		00000000 0040006C	00400004		0040
± ♦ s_tempram	32'	0000000		100400000	100400000	100400000	100400004	100400000	, 004000C		00000000	
	32'	00000000										
★ s_regout2	32'	00000000										
	32'	0000000			00000000		0000000A		00000001	00000000		
	32'	0000000	0	00400058	00000000	00440000	00200028	00208200	00400004	00000000		
	32'	XXXXXXXX										
	5'hXX	XX										
± ♦ s_wo	5'hXX	XX										
★ s_JALmuxo	32'	XXXXXXXX										
★ s_JRmuxo	32'	0000000	0	00400058	00400008	0040005C	00400060	00400064	00400004	0040006C	00400004	
→ s_AS	0											
s_sub	0					+						
s_ignrOvfl s_RegWr_te	0											_
s_NegWi_te	0											
◆ s_EXmemZero												
s_regequal	0											
s_EXmemHalt												
s_RegValue s Halt temp	0											
s_Hait_temp	1											
	3'h0	X (0										
	3'h0	X (0										
<u>★</u> ◆ s_EXmemWB	4'h0	X (0			2	0	2				6	2
± ♦ s_IDEXwb	4'h0	X (0		2	0	2				6	2	
		X (0		10	00	2	0	04	115	100		6
	5'hXX 5'hXX	(00		10	00	11	08	104	1F	00		
	5'hXX	(00			10	00	11	08		10	00	
	5'hXX	00							04	00		
★ s_RegWrA_t	5'hXX	00			10		11	08	04	1F	00	
s_tempForw	5'hXX	(00				10	00	11	08	04	1F	00
± ♦ s_EXmemRe	5'hXX 5'hXX	(00		110	00	11	00	11	08 1F	04	1F	00
	5'h00	(00		02	08	02	00		02	08	02	
	6'h00	(00		2E	100	2E	110		2E	00	2E	
	32'	(0000000	0									
<u>⊕</u> ♦ s_IDEXreg2	32'	0000000	0									
		(0000000				00000000		0000000A	i e	00000001		
	32'	0000000				00000000						
s_IDEXnewPC		0000000			00400004	00400008	0040005C	00400060		00400068	,0040006C	,0040
	32'	(0000000		08100016	00000000	20110000	20080004	00082080	0000000A 0C100001	00000000		
	32'	(0000000				0040005C					00400004	
	32'	XXXXXXXX										
	_											

★ S. DEXnewPC         32         000000000         004000004         00400008         00400005         00400000         004000000		32'	(00000000			08100016	00000000	20110000	2008000A	00002080	0C100001	00000000	
S		_								1			100
\$   Filbries   22						00400004	00400008	10040003C	00400000			00400000	1
② ◆ S. Finnew PC. 22.		_			09100016	00000000	20110000	20090004	0000000				
\$ S. EXMEMPER   \$2.   \$2.   \$3.										1		00400004	
S   S   S   S   S   S   S   S   S   S					00400004	00400008	0040003C	00400000	00400004	00400008	10040000C	00400004	
S = EXmemNen.         2         000000000         00400004         00400008         00400060         00400060         00400006         00400006         00400006         00400006         00400004         00400006         00400004         00400004         00400004         00400004         00400004         00400004         00400004         00400004         00400004         00400005         00400005         00400005         00400005         00400005         00400005         00400005         00400005         00400005         00400005         00400005         00400005         00400005         00400006         00400006         00400006         00400006         00400006         0040006										00000004	00000000		
(							00400004	00400008	00400050			00400068	100
(					00000019		00400004	100400000	00400030	,00400000			100
(\$\infty\$ \ \circ\$ \ \ \circ\$ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \		_			00000013						.00000004	00000013	
③ S. WBmemh 32					00400058	0040005C	00400060	00400064	00400068	00400004	00400004		100
③ ◆ S. WBmemN 32					00400000	00400030	0040000	100400004	00400000	,00400004		00000000	+
								00400004	00400008	0040005C			100
BaddO         32*         00000000         0040006C         0040006C         0040006C         0040006C         0040006C         00400004         00400008         00400008         00400008         00400008         004000000         00400000         00400000         00400000         00400000         004000000         00400000         00400000         00400000         00400000         004000000         00400000         00400000         00400000         00400000         00400000         00400000         00400000         00400000         00400000         00400000         004000000         004000000         004000000         004000000					00400058	00400008	0040005C						1
S		_											
## \$ BmuxO 32"							00400000					00400004	
\$ s_stall F							0040005C					00400004	
S_FIDflush   1			10000000		00+0000+	00400000	00400000	,00400000	00400004	00400000	100400000	00400004	
S = Freadwrite   O										-			
S   EXmemstall   1						1					1		
S_EXmemflu   1		-											
S   DEXstall   1	<del>- : -</del>												
S_IDEXflush   1		_											
S memWBflu 1													H
★ s_memWBstall         2h0         0         1         0         2         0         1         0         2         0         0         1         0         2         0         0         1         0         2         0         0         1         0         2         0         0         0         1         0         0         1         0         0         0         1         0         0         0         0         0         0         1         0													
★ s forwardA         2h0           ★ s forwardB         2h0           ★ s forwardBr         2h0           ★ s EXTMERND         5h00           ★ s EXTMERND         5h0X           ★ s flowsigner         3cmmwbre         0           ★ s memwbl         3cmmwbre         0           ★ s memwbl         3cmmwbre         0           ★ s flexignext         3cmmwbre         0           ★ s flexignext         3cmmybre         3cmmybre           ★ s flexignext         3cmmybre         3cmmybre         3cmmybre           ★ s flexignex													
★ s forwardB         2h0           ★ s forwardBr         2h0           ★ s forwardBr         2h0           ★ s forwardBr         2h0           ★ s FOstallAddr         32'           ★ s memWBrd         5h00           ★ s ExmemBr         5h2X           ★ s exmemdr         5h2X           ★ s memwbr         0           ★ s memwbr         0           ★ s memwbr         0           ★ s memwbra         0           ★ s memwbra         0           ★ s jdexsignest         32'           ★ s jdexsignest         32'           ★ s jdexignmux         32'           (000000000         00000000           1							1 1	10				<del>                                     </del>	₽
■												2	
★ s forwardBr         2 h0         0         1         0         1         0         1         0							-	1					
S s IDEXhalt         0           ★ S PCstallAddr         32'           ★ S s_memWBrd         5h0X           ★ S s_exmemrdrt         5hXX           ★ S s_exmemrdr         0           ★ S s_exmemrea         0           ★ S s_memwbre         0           ★ S s_membra         32'           ★ S s_dexignext         32'           ★ S s_memwbjr         3								10	11	10			
★ s PCstallAddr 32'         000000004         100000000         00           ★ s s_memWBrd 5h00         5h00         00								1	1				Т
★ s_memWBrd         5h00           ★ s_EXmemRD         5hXX           ★ s_dexmemrt         0           ★ s_memwbre         0           ★ s_memwbm         0           ★ s_memwbm         0           ★ s_idexsignext         32'           ★ s_idexjmux         32'           ★ s_idexjmux         32'           ★ s_idexjmux         32'           ★ s_memwbir         32'           ★ s_memmybir         32'           ★ s_temp_rego         32'		32'	00000004								100000000		00
★ S EXMEMRD       5hXX         ★ S exmemrdt       5hXX         ★ S exmemrea       0         ★ S memwbre       0         ★ S idexsignext       32'         ★ S idexsignext       32'         ★ S idexjrmux       32'         ★ S exmemprr       32'         ★ S exmemprr       32'         ★ S memwbjr       32'         ★ S memwbjr       32'         ★ S temp_rego       32'         ★ S forwardAM       32'         ★ S forwardBM       32' <tb>100000000       10000000A       10000000A         ★ S forwardBM       32'         ★ S forwardBM       32'</tb>	<del>- : -</del>												
★ s exmemrdrt         5hXX           ★ s_idexmemr         0           ★ s_memwbre         0           ★ s_memwbre         0           ★ s_memwbre         0           ★ s_memwbal         32'           ★ s_idexsignext         32'           ★ s_exmembra         32'           ★ s_idexjmux         32'           ★ s_memwbjr         32'           ★ s_memwbjr         32'           ★ s_memwbjr         32'           ★ s_temp_rego         32'													
♦ s Idexmemr         0           ♦ s_exmemrea         0           ♦ s_memwbre         0           ♦ s_memwbre         0           ♦ s_memwbal         32'           ★ \$ s_idexignext         32'           ★ \$ s_idexjrmux         32'           ★ \$ s_idexjrmux         32'           ★ \$ s_memwbjr         32' <td></td>													
♦ S exmemrea       0         ♦ s_memwbre       0         ♦ s_memwbre       0         ♦ s_memwbal       32'         ★ \$_sldexsignext       32'         ★ \$_sidexjrmux       32'         ★ \$_sememjrm       32'         ★ \$_sememjrm.													
♦ s memwbre       0         ♦ s_memwbm       U         ★ s_memwbal       32'         ★ s_idexsignext       32'         ★ s_idexjrmux       32'         ★ s_idexjrmux       32'         ★ s_sexmemjrm       32' <tr< td=""><td><del>-</del></td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr<>	<del>-</del>	0											
♦ s memwbm       U         ★ \$ s_memwbal       32'         ★ \$ s_idexsignext       32'         ★ \$ s_idexjrmux       32'         ★ \$ s_idexjrmux       32'         ★ \$ s_exmemjrm	s memwbre	0											
★ s idexsignext       32'         ★ s_idexjrmux       32'         ★ s_idexjrmux       32'         ★ s_idexjrmux       32'         ★ s_exmemjrm       32'         ★ s_memwbjr       32'         ★ s_temp_rego       32'         ★ s_temp_rego       32'         ★ s_temp_rego       32'         ★ s_temp_rego       32'         ★ s_torwardAM       32'         ★ s_torwardBM       32'		U											
★ \$ s_Idexsignext       32'         ★ \$ s_exmembra       32'         ★ \$ s_idexjrmux       32'         ★ \$ s_exmemjrm       32'         ★ \$ s_memwbjr       32'         ★ \$ s_temp_rego       32'         ★ \$ s_temp_rego       32'         ★ \$ s_temp_rego       32'         ★ \$ s_torwardAM       32'		32'	XXXXXXXX										
★ S exmembra       32'         ★ S_idexjrmux       32'         ★ S_exmemjrm       32'         ★ S_memwbjr       32'         ★ S_temp_rego       32'         ★ S_temp_rego       32'         ★ S_temp_rego       32'         ★ S_torwardAM       32'         ★ S_torwardAM       32'         ★ S_torwardBM       32'		32'	XXXXXXXX										
★ s_idexjrmux       32'       (00000000       00400058       00400005       00400060       00400064       00400064       00400006       00400060       00400060       00400064       00400064       00400064       00400060       00400060       00400060       00400060       00400060       00400060       00400064       00400064       00400064       00400064       00400060       00400064       00400064       00400064       00400064       00400064       00400064       00400064       00400064       00400064       00400066		32'	XXXXXXXX										
★ s_exmemjrm       32'       (00000000       (00400058 0040005C 00400060 00400064 00400004 00400004 00400064 004000064 004000064 00400064 00400064 00400064 00400064 004000664 00400666		32'	00000000			00400058	00400008	0040005C	00400060	00400064	00400004	0040006C	00
★ s_memwbjr       32'         ★ s_temp_rego       32'         ★ s_temp_rego       32'         ★ s_forwardAM       32'         ★ s_forwardBM       32'         Now       180 ns         180 ns       20 ns         40 ns       60 ns         80 ns       120 ns         100 ns       120 ns         120 ns       180 ns		32'	00000000				00400058	00400008	0040005C	00400060	00400064	00400004	00
★ s temp_rego       32'         ★ s temp_rego       32'         ★ s forwardAM       32'         ★ s forwardBM       32'         Now   80 ns       100000000         100000000       100000000         100000000       100000000         100000000       100000000         100000000       100000000         100000000       100000000         100000000       100000000         100000000       100000000		32'	(00000000					00400058	00400008	0040005C	00400060	00400064	00
★ s forwardAM       32'       (00000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       1000000000       100000000       100000000       100000000       100000000       1000000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       1000000000       100000000       100000000       100000000       100000000       1000000000       100000000       100000000       100000000       100000000       1000000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       100000000       10000000       100000000       10000000       10000000       10000000       10000000       10000000       10000000       10000000       100000000       10000000       10000000       10000000       10000000       10000000       10000000       100000000       100000000       100000000       100000000		32'	00000000										
★ s_forwardBM       32'       (000000000       100000000A       10000000A       100000000A       10000000A       100000000A       100000000A       100000000A       100000000A       <		32'	00000000										
★ s_forwardBM       32'       (00000000       00000000       00000000         L E ● Now 180 ns       20 ns       40 ns       60 ns       80 ns       100 ns       120 ns       140 ns       160 ns       180 ns       200 ns	_ : _ :		(00000000										
Now 180 ns 15 20 ns 40 ns 60 ns 80 ns 100 ns 120 ns 140 ns 160 ns 180 ns 200 ns		32'	(00000000						0000000A	00000000			
15 20 HS 40 HS 80 HS 100 HS 120 HS 140 HS 180 HS 200 H		190 ps		1 1 1	1 1 1		1 1 1	1 1 1 1	1 1 1 1	1 1 1	1 1 1	1 1 1	ij
□ ≯ ○ Cursor 1 0 ns   0 ns				ns 40	ns 60	ns 80	ns 10	0 ns 12	0 ns 14	0 ns 16	0 ns 18	0 ns 20	0 ns
	🔟 🧨 😊 Cursor 1	0 ns	0 ns										

### [ Base Test No Nops ]

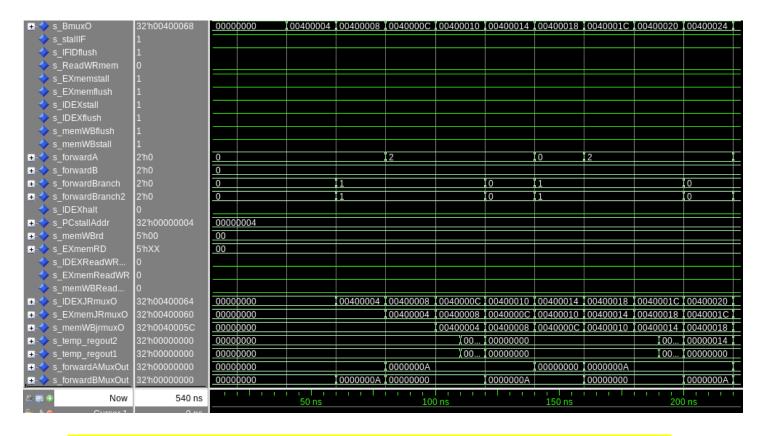
wa Wa	ve - Default ====						*****	, opo j						= + 3
<u> </u>	١	Msgs												
4	iclk irst	1 0												
-	<b>→</b> iInstLd	0												
1	ilnstAddr ilnstExt	32' 32'	XXXXXXXX											
■ 4	o ALUOut	32'	00000000		7FFF0000	7FFFEFFC	00000000	00000005	0000000A	00000000			10010000	100
	> s_DMemWr > s_DMemAddr	U 32'	00000000			7FFF0000	7FFFEFFC	00000000	00000005	0000000A	00000000			100
	> s_DMemData	32'	00000000	00400004	0040000				00000005				000000FF	000
	> s_IMemAddr > s_NextInstAddr	32' 32'	00400000			0040000C 0040000C								
■ 4	s_Inst	32'	3C1D7FFF	37BDEFFC	001D4020	20080005	2009000A	016B6021	01096824	31AE00FF	3C081001	AD0E0000	8D0A0000	
	> s_RegWr > s_RegWrAddr	5'hXX	00				1D		08		09	0C	0D	0E
	s_RegWrData	32' 32'	00000000		00007555	0000EFFC			00000000			00000000 000000FF		000
	> s_mux2out > s_Halt	J2 U	0000000		00007FFF	, OUGUEFFC ,	0000000	00000003	0000000A	, 00000000		000000FF	00001001	, 000
4	s_Ovfl s_beqS	0	-+											
4	s_zeqs s_zero	0												
4	s_Carry s_signExtend	0	$\rightarrow$						-					-
	s_regDst	0	=											
	> s_aluSrc > s_memToReg	0 0	-+											
4	s_memRead	0												
4	> s_Branch > s_j	0												
4	s_ANDo	U												
4	> s_jr > s_jal	0												
4	s_movn	0	_											
	> s_aluControl > s_aluout	4'h0 32'	00000000		7FFF0000	7FFFEFFC	00000000	00000005	0000000A	00000000			10010000	100
	s_PCFour	32' 32'	00400004	00400008	0040000C	00400010			0040001C 00000000					
	> s_tempram > s_regout1	32'	00000000				,/FFF0000 ,	,/FFFEFFC	,00000000		00000000		00000005	000
	s_regout2	32' 32'	00000000	00007555	ODDOEEEC	00004020	00000005	0000000	00006021	00006924	0000000	00000005		
	<ul><li>s_extended</li><li>s_jshifto</li></ul>	32'	00000000			00750080								000
	s_JRmuxo s_AS	32' U	00000000	00400004	00400008	0040000C	00400010	00400014	00400018	0040001C	00400020	00400024	00400028	
4	s_sub	0												
4	s_ignrOvfl s_RegWr_temp	0												
4	s_DmemWR_t	0	$\rightarrow$											
4	s_EXmemZero s_regequal	U 0	-+											
4	s_EXmemHalt	Ů	=											
	> s_Halt_temp > s_flush2	0 1	=	-										
	s_IDEXmem	3'hX	0											2
	s_EXmemMEM s_EXmemWB	3'hX 4'hX	0		2									
	s_IDEXwb	4'hX	0	2										0
	s_WBmemOut s_WRmuxO	4'hX 5'hXX	00	1D		08	08	09	0C	0D	0E	08	0E	00
	s_IDEXrs	5'hXX 5'hXX	00		1D	1D	00	08			08			08 0E
	s_IDEXrt s_IDEXrd	5'hXX	00			1D		00						00
	s_RegWrA_te s_tempForwar	5'hXX 5'hXX	00		1D	, 1D	08	08						0E 08
■ <	s_EXmemReg	5'hXX	00			1D		08				0D	0E	08
	s_WRtemp s_IDEXfunc	5'hXX 5'hXX	00	1D 02		00	08	09			0E 00			00
■ 🕠	s_IDEXex	6'hXX	00		13		20	10		20	21	11	13	10
	s_IDEXreg1 s_IDEXreg2	32' 32'	00000000								00000005		00000005	000
■ 4	s_IDEXextend	32'	0000000			0000EFFC						000000FF	00001001	000
	s_IDEXinst s_IDEXnewPC	32' 32'	00000000			37BDEF 00400008								
#-◆	s_EXmemMux	32'	0000000			00007FFF	0000EFFC	00000000	00000005	0000000A	00000000		000000FF	000
	s_IFIDinst s_IFIDnewPC	32' 32'	00000000			001D4020 0040000C								
■ 4	s_EXmemALUo	32'	00000000			7FFF0000	7FFFEFFC	00000000	00000005	0000000A	00000000			100
	s_EXmemNe s_PCend	32' 32'	00000000 00400004	00400008	0040000C	00400004			00400010 0040001C					,004
# ◀	s_WBmemAL	32'	00000000				7FFF0000	7FFFEFFC	00000000	00000005	0000000A	00000000		004
	s_WBmemNe s_JmuxO	32' 32'	00000000	00400004	00400008	0040000C			0040000C 00400018					
■ 4	s_BaddO	32'	00000000	00420000	0043BFF8	0041008C	00400024	0040003C	0041809C	0041A0AC	0040041C	00404028	00400028	
	s_BshiftO s_BmuxO	32' 32'	00000000			00010080 0040000C								
4	s_stallIF	1												
4	<ul><li>s_IFIDflush</li><li>s_ReadWRme</li></ul>	0												
4	s_EXmemstall	1												

s stallif			00 10000 1	00 100000	00 100000	00 100010	00100011	00 100010	00 100010	00 100020	00 1000L 1	00 100020	
s_stalling 1													
s ReadWRme	,												
s EXmemstall 1	í												
s IDEXstall													
s IDEXflush													
s_memWBflush 1													
s memWBstall 1													
_	2'h0	0			2	10					2	10	2
	2'h0	0											
±♦ s_forwardBran 2	2'h0	0		1	0					1	0	1	O
±♦ s_forwardBran 2	2'h0	0		1		1	0	0	2	0	0	2	O
♦ s_IDEXhalt l  L  L  L  L  L  L  L  L  L  L  L  L  L	J												
± ♦ s_PCstallAddr 3	32'	00000004										00000000	
± ♦ s_memWBrd 5	5'hXX	00				0F	00				0C	0D	00
• → s_EXmemRD 5	5'hXX	00			0F	00				0C	0D	00	
🔷 s_IDEXRead l	J												
🔷 s_EXmemRea l	J												
🔷 s_memWBRe l	J												
	32'	00000000		00400004			00400010		1			1	
	32'	00000000					0040000C		i e			1	
🛂 🔷 s_memWBjrm 🛭 3	32'	00000000				00400004	00400008	0040000C	00400010	00400014	00400018		004
🛂 🔷 s_temp_regout2 🛭		0000000									00000005		
	32'	00000000							00	00000000		00000005	7
- · · -	32'	00000000			7FFF0000					00000005			100
₽~♦ s_forwardBMu 3	32'	00000000		00007FFF	0000EFFC	00000000	00000005	A0000000	00000000		000000FF	00001001	000
S ■ Now 8	80 ns		50 ns	1 1 1	100	l i i i Dns	1 1 1	150 ns	1 1 1 1	20	l i i i Dins	1 1 1	250 1
Cursor 1	0.00												

## [2.e.i] Create a spreadsheet to track these cases and justify the coverage of your testing approach. Include this spreadsheet in your report as a table.

Stage	Description	Examples
IF -> ID	Instruction in IF depends on data ID stage	add \$t0, \$0, \$sp addi \$t0, \$0, 5
IF -> EX	Instruction in IF depends on data EX stage	sw \$t6, 0(\$t0) lw \$t2, 0(\$t0) nor \$t3, \$t2, \$t0
IF -> MEM	Instruction in IF depends on data MEM stage	xor \$t4, \$t2, \$t3 xori \$t5, \$t4, 0x1100 or \$t2, \$t3, \$0 ori \$t4, \$t2, 0x11

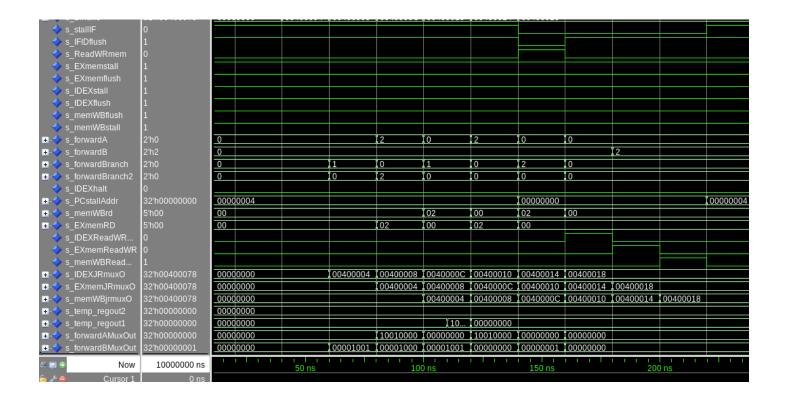
wave - Default :======												
<u></u>	Msgs											
	1											
	0											
ilnstAddr	32'hXXXXXXXXX		XXXX									
	32'hXXXXXXXX 32'h00000000	0000	XXXX 0000		0000000A	0000000A		00000014	0000000A	0000000A		00000014
s_DMemWr	0	2222				*						
<b>±</b> - <b>♦</b> s_DMemAddr <b>±</b> - <b>♦</b> s_DMemData	32'h00000000 32'h00000000	0000				0000000A 0000000A	00000000		00000014 00000000A	0000000A	00000000	
<b> </b>	32'h00400068	0040	0000			0040000C	00400010	00400014	00400018	0040001C	00400020	
	32'h00400068 32'hXXXXXXXX	2008		01084020	00400008		200A000A			0040001C	20080050	
s_RegWr	0											
	5'h00 32'h00000008	0000	0000				08 0000000A			00000014	0A 0000000A	
s_mux2out	32'h00000000	0000			0000000A	00000000		0000000A		00000000		0000000A
	0											
s_beqS	ő											
	0											
s_signExtend	0											
	0											
s_memToReg	ő											
s_memRead	0											
	0											
s_ANDo	0											
◆ s_jr   ◆ s_jal	0											
s_movn	0											
s_aluControl  s_aluout	4'h0 32'h00000000	E 0000		0	0000000A	0000000A		00000014	0000000A	0000000A		00000014
■ → s_PCFour	32'h0040006C	0040	0004	00400008		00400010	00400014	00400018		00400020	00400024	
	32'h00000008 32'h00000000	0000					0000000A	00000000		00000014		00000000
• s_regout2	32'h00000000	0000	0000				(00	00000000			(00	00000014
	32'h0000XXXX 32'h0XXXXXXX	0000		0000000A 00200028					00005020 05294080			00000050
■ → s_JRmuxo	32'h00400068	0000								0040001C		
	0											
s_jgnrOvfl	ő											
s_RegWr_temp s_DmemWR_temp	0											
	0											
s_regequal	0											
	ő											
s_flush2	1											
	3'h0 3'h0	0										
s_EXmemWB	4'h0	0			2							
■  s_IDEXwb ■  s_WBmemOut	4'h0 4'h0	0		2		2						
<b></b> → s_WRmuxO	5'hXX	00		08				0A	V 0.0			08
■  s_IDEXrs ■  s_IDEXrt	5'hXX 5'hXX	00			08	08			00 0A	0A		
■ → s_IDEXrd	5'hXX	00				08			00	0A		
■  s_RegWrA_temp  ■  s_tempForwardR	5'hXX 5'hXX	00			08	08			0A	0A		
<b>±</b> - <b>♦</b> s_EXmemRegoM	5'hXX	00				08				0A		
	5'hXX 5'h00	00		08 02	00			0A				08
<b></b> → s_IDEXex	6'h00	00		2E	10	20			10	20		
■  s_IDEXreg1  ■  s_IDEXreg2	32'h00000000 32'h00000000	0000							00000000			0000000A
<b></b> → s_IDEXextend	32'h0000XXXX	0000	0000			00004020			0000000A	00005020		3071
■  s_IDEXinst ■  s_IDEXnewPC	32'hXXXXXXXX 32'h00400064	0000				01084020				014A5020 00400018		00400020
■ → s_EXmemMux	32'h00000000	0000	0000			0000000A	00000000		0000000A		00000000	
	32'hXXXXXXXX 32'h00400068	0000		2008000A					014A5020	0040001C		20080050
<b>±</b> → s_EXmemALUo	32'h00000000	0000		30400004	30-1000008	0000000A			00000014	0000000A		
	32'h00400060 32'h0040006C	0000		00400000	00400000				1	00400014 00400020		
	32'h00000008	0000		00400008	, 0040000C		00000000A		, 004000IC	00000014		,00400028
s_WBmemNewPC	32'h0040005C	0000		0040000	0040000					00400010 0040001C		
	32'h00400068 32'h004XXXXX	0000								0040001C		
■ → s_BshiftO	32'h000XXXXX	0000		00000028		0040000			00014080			00000140
■ s_BmuxO  s_stallIF	32'h00400068 1	0000	5000	00400004	00400008	0040000C	00400010	00400014	00400018	0040001C	00400020	00400024
s_IFIDflush	1											
	0											



[2.e.ii] Create a spreadsheet to track these cases and justify the coverage of your testing approach. Include this spreadsheet in your report as a table.

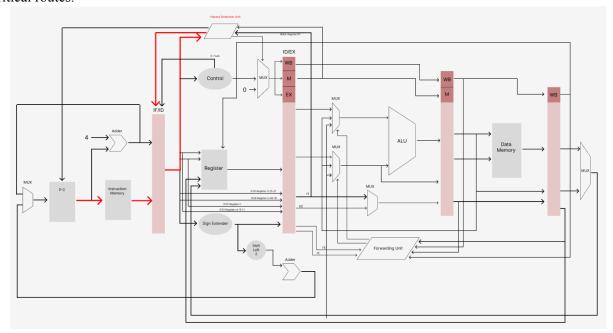
Instruction	Description	Example
Jump	Completes alu instructions and jumps to a new set of instructions that are sw in after, variable change	addi \$t1, \$t1, 0x1 ori \$t7, \$t7, 0x1 j after
Branch	branches after lw, conditional change	lw \$t5, 0x10(\$t0) beq \$t7, 0x1, odd

Columbia		Wa <sup>*</sup>	ve - Default =					*****							
RICT	ľ			Msgs											
The color of the		<b>-</b> <del>4</del> 3		0		┨	┤└─	┤└─							
Part		4		0											
Company   Comp															
Part   Defended   Company   Compan							10010000	10011000	10010000	10010000	00000001	00000000			
Company   Comp				0			10010000								
D	ı		_												
C   2	ı							00001001	00001000	00001001	00000000	00000001	00000000		
	ı	■ 4	s_IMemAddr												
1   Party   1															
D	ı				30011001	1343DI000	30011001	34200000	240F0001	80090000	6DUA0004				
Part	ı	■ ◆	s_RegWrAddr									08			
Septiment   Sept	ı						00001001					00000000	00000001		00000000
Lings				0	0000000		00001001	00001000	00001001	00000000	00000001				
Light		4		0											
Laptic Bear	ı	3		0											
Light   Ligh				0											
	ı		_ •	0											
	ı			1											
Limited   Company   Comp				0											
		4	s_memRead	0											
Part				0											
	ı			0											
Description   Section		4	s_jr	0											
	ı		_	0											
Description   Substitution   Subst				~	E	3	В	3	В	0		E			
Compact   September   Septem			_									00000000			
S   sequentary   Sequence   Seq	ı					00400008	0040000C	00400010					00000001		
S. s. separtal   Stribo0000000   00000000   00000000   00000000	ı		_ '										00000001	,00000000	
S   S   S   S   S   S   S   S   S   S	ı	₽-4	s_regout2		00000000										
S   S   S   S   S   S   S   S   S   S	ı														
S. S. Sub	ı														
S. Sign/CM   Emp   S. Dimem/K   Emp   D   S. Exmem/Zeri   S. Fregequal   D   S. Exmem/Zeri   D   S. Exmem/	ı			0											
S. Regilfy Lemp S. ExmemAtait S. Hall, Eump S. Burner S.				0			-				-				
S. ExmemAzir   S. E				1											
S. F.	ı			0											
S. ExmemHatt D. S. Jinshi Z. D	ı		_	1											
S.   S.   S.   S.   S.   S.   S.   S.	ı	4		0											
Description   State   Description   State   Description   State   Description   Desc	ı			0											
Description   Septemble   Se				1 3'h0	0										
	ı		_		0										
					0		2							2	
S   S   S   S   S   S   S   S   S   S						12		2				3	. 2	3	2
D						01			08	0F	09	00			
D															
C								, IU			, UF	.09	,00		
C   S   SEXMEMREGOM   5h00   00   01   1D   01   08   0F   09   00   00   00   00   00   00		■ 4	s_RegWrA_temp	5'h00	00		01		01	08					
C															
C						01		1					09	00	
Company   Comp		■ 🕠	s_IDEXfunc	5'h02	00	02		00	02	00	02				
S   JDEXreg2   32h0000000   32h000000   32h0000000   32h0000000   32h0000000   32h000000   32h0000000   32h0000000   32h0000000   32h000000   32h0000000   32h0000000   32h0000000   32h000000   32h000000   32h0000000   32h000000   32h0000000   32h000000   32h00000   32h000000   32h00000   32h000000   32h00000   32h000000   32h000000   32h00000   32h000000   32h000000   32h000000   32h00000   32h00000   32h00000   32h00000   32h00000   32h00000   32h00000   32h000000   32h00000   32h00000   32h00000   32h000000   32h00000   32h00000   32h00000   32h00000   32h00000   32h00000   32h000000   32h00000   32h00000   32h000000   32h0000000   32h000000   32h0000000   32h0000000   32h0000000   32h00000000   32h00000000   32h0000000   32h0000000   32h00000000   32h0000000   32h00000000   3						ZE	13	1B					2E		
C										10010000	300000000				
Company   Comp		■ 🕠	s_IDEXextend	32'h00000000	00000000										
S   SEXIMEMMUX   32h00000001   32h00000001   32h00000000   32h0000000   32h0000000   32h00000000   32h0000000   32h000000   32h000000   32h000000   32h0000000   32h000000   32h000000   32h000000   32h000000   32h0000000   32h000000   32h0000000   32h000000   32h0000000   32h000000   32h0000000   32h00000000   32h000000000000   32h00000000   32h00000000   32h0000000   32h00000000   32h0000000   32h00000000   32h00000000   32h00000000   32h0000															
C → S_IFIDINST         32h00000000         32h0000000         300000000         300000000         30000000         3000000							00400004								
Company   Section   Sect		■ <	s_IFIDinst	32'h00000000	00000000			3C011001	34280000	240F0001	8D090000	00000000			
Company   Comp						00400004	00400008						00000000		
Company   Section   Sect			_												
C → S_WBmemALU0         32h10010010         32h10010010         10010000         10010000         10010000         00000000         00000000           C → S_BmemNewPC         32h00400078         32h00400078         32h00400078         0000000         00440004         00400004         00400004         00400001         00400014         00400014         00400014         00400018           C → S_BshiftO         32h00400078         32h00400078         0000000         00404008         00404001         00400010         00400018         00400018           C → S_BmiuxO         32h00400078         32h00400078         0000000         0040004         00040004         00040001         00400014         00400018           → S_StallIIF         0         0000000         0040004         00400008         00400000         00400014         00400018		■ 4	s_WBdmem	32'h00000010	00000000										
t → s_WBmemNewPC         32 h00400078         00000000         00400004         00400004         00400000         00400014         00400014         00400014         00400014         00400014         00400014         00400018           t → s_BaddO         32 h00400078         32 h00400078         0000000         00404008         004040010         00400011         00400018         00400018           t → s_BhilfO         32 h00400078         32 h00400078         0000000         0004004         00400040         0000000         00004004         00400018         00400018           • s_BmuxO         32 h00400078         32 h00400078         0000000         0040004         00400008         00400000         00400014         00400018         00400018           • s_stallIF         0         0000000         00400004         00400008         00400000         00400014         00400018         00400018						00400008	0040000C	00400010					00000000		
C → S JmuxO         32h00400078         0000000         00400004         00400008         00400001         00400014         00400018         00400018           C → S BaddO         32h00400078         32h00400000         00404008         00404008         00404010         00400010         00400018         00400018           C → S BmuxO         32h00400078         0000000         0040004         0040000         0040000         00400010         00400014         00400018           → S stallIF         0         0000000         0040004         00400008         00400001         00400014         00400018           → S _FIDIflush         1         0         00400004         00400001         00400014         00400018														î	
S S BshiftO 32h0000000 00000000 00000000 0000000 000000		■ 🕠	s_JmuxO		00000000				00400010	00400014	00400018				
S S BMIUXO 32'h00400078 00000000 00400004 00400008 0040000C 00400010 00400018 004000018 004000018 004000018 004000018 004000018 004000018 004000018 0040000018 004000018 0040000018 0040000018 0040000018 0040000000000															
◆ S_stalliF          0               ■             ■															
♦ s_IFIDflush 1						00700004	00,00000	30,00000	00.00010	00 /00014	00.00010				
△ □ ○ Now 10000000 ns 50 ns 100 ns 150 ns 200 ns		- :		1											
		₩ 🗊	Now	10000000 ns		50 ns		100	) ns		150 ns		200	0 ns	



## [2.f] report the maximum frequency your hardware-scheduled pipelined processor can run at and determine what your critical path is (specify each module/entity/component that this path goes through).

Our hardware-scheduled pipelined processor can run at a maximum frequency of 43.09 MHz in 23.2 nanoseconds. The critical path starts at the id/ex pipeline register and goes to the alu then ends at the pc reg. Red lines in the diagram below denote critical routes.



## [ Synthesis\_hw: timing.txt ] \*timingHW(1).txt

Open ▼ 🙉 # CprE 381 toolflow Timing dump FMax: 43.09mhz Clk Constraint: 20.00ns Slack: -3.21ns The path is given below IDEX:id\_ex|dffg\_N:RegInst|s\_0[30]
PC:pcReg|s\_OUT[2] From Node To Node Launch Clock : iCLK Latch Clock iCLK Data Arrival Path
Total (ns) Incr Incr (ns) Type Element launch edge time
clock network delay
IDEX:id\_ex|dffg\_N:RegInst|s\_Q[30]
id\_ex|RegInst|s\_Q[30]|q
Mux0-0|dataa 3.258 0.232 0.232 0.000 FF CELL 3.258 3.789 0.531 FF 0.406 FR CELL 4.195 Mux0~0|combout 0.204 RR 0.139 RF 0.486 FF 4.399 Mux0~2|datad CELL 4.538 5.024 0.486 FF 0.150 FR 0.204 RR 0.139 RF 0.237 FF 0.125 FF 0.249 FF 6.128 0.125 FF CELL 6.253 6.879 0.626 FF 0.125 FF 7.256 0.252 7.381 0.125 FF 0.256 FF 0.256 FF 0.281 FF 0.250 FF 0.125 FF 0.251 FF 0.125 FF 8.669 8.919 0.250 FF 0.125 FF 9.044 9.293 0.249 FF 9.418 0.125 FF 9.672 0.254 FF 0.254 FF IC 0.281 FF CELL 0.251 FF IC 0.125 FF CELL 0.250 FF IC 0.125 FF CELL 0.257 FF IC 0.281 FF CELL 0.248 FF IC 9.953 11.242 11.490 0.248 FF IC 0.125 FF CELL 0.415 FF IC 11.615 12.030 0.125 FF CELL 0.252 FF IC 12.155 0.252 FF IC 0.125 FF CELL 0.255 FF IC 0.281 FF CELL 0.250 FF IC 0.125 FF CELL 13.699 0.256 FF 0.281 FF 13.980 14.235 0.255 FF IC 0.281 FF CELL 14.516 0.252 FF 0.252 FF IC
0.125 FF CELL
0.249 FF IC
0.125 FF CELL
0.251 FF IC
0.125 FF CELL
0.258 FF IC
0.281 FF CELL 14.893 15.142 15.267 15.518 15.643 15.901 16.182 16.437 0.255 FF 0.255 FF IC 0.281 FF CELL 0.250 FF IC 16.718 16.968 0.125 FF CELL 0.249 FF TC 17.342 0.249 FF 0.125 FF 0.250 FF 0.125 FF 0.251 FF 0.125 FF 0.227 FF 17.717 17.842 18.093 18.218 18.595 0.150 FR 19.010 0.415 RR 19.297 0.287 RR 19.502 0.205 RR 19.657 0.155 RR 0.155 RR 0.203 RR 0.155 RR 0.897 RR 0.155 RR 1.252 RR 0.386 RF 22.705 0.386 RF 0.228 FF 22.933 0.125 FF CELL 0.227 FF IC CELL s\_regequal~3|combout
IC s\_regequal~4|datad
CELL s\_regequal~4|combout
IC s\_regequal~2|datad 23.058 23.285 0.125 FF CELL 23.410 23.638 0.228 FF 23.788 0.150 FR CELL s regegual~2|combout s\_regequal-2|combout
comb-2|datad
comb-2|combout
prePCmux|\G\_NBit\_MUX:2:MUXI|g\_org1|o\_F-0|datad
prePCmux\\G\_NBit\_MUX:2:MUXI|g\_org1|o\_F-0|combout
prePCmux\\G\_NBit\_MUX:2:MUXI|g\_org1|o\_F-2|datad
prePCmux\\G\_NBit\_MUX:2:MUXI|g\_org1|o\_F-2|combout
prePCmux\\G\_NBit\_MUX:2:MUXI|g\_org1|o\_F-2|combout
pcReg|s\_OUT[2]|d
PC:pcReg|s\_OUT[2] 0.698 RR 0.139 RF 0.940 FF 24.625 25.565 25.715 CELL CELL 0.150 FR 25.919 0.204 RR IC CELL 0.139 RF 26.058 0.000 FF 26.162 0.104 FF Data Required Path: Total (ns) Incomp Type Element latch edge time clock network delay clock pessimism removed clock uncertainty 20.000 20.000 22.924 22.956 22.936 -0.020 uTsu 22.954 0.018 PC:pcReg|s\_OUT[2] Data Arrival Time 26.162 Data Required Time : -3.208 (VIOLATED)