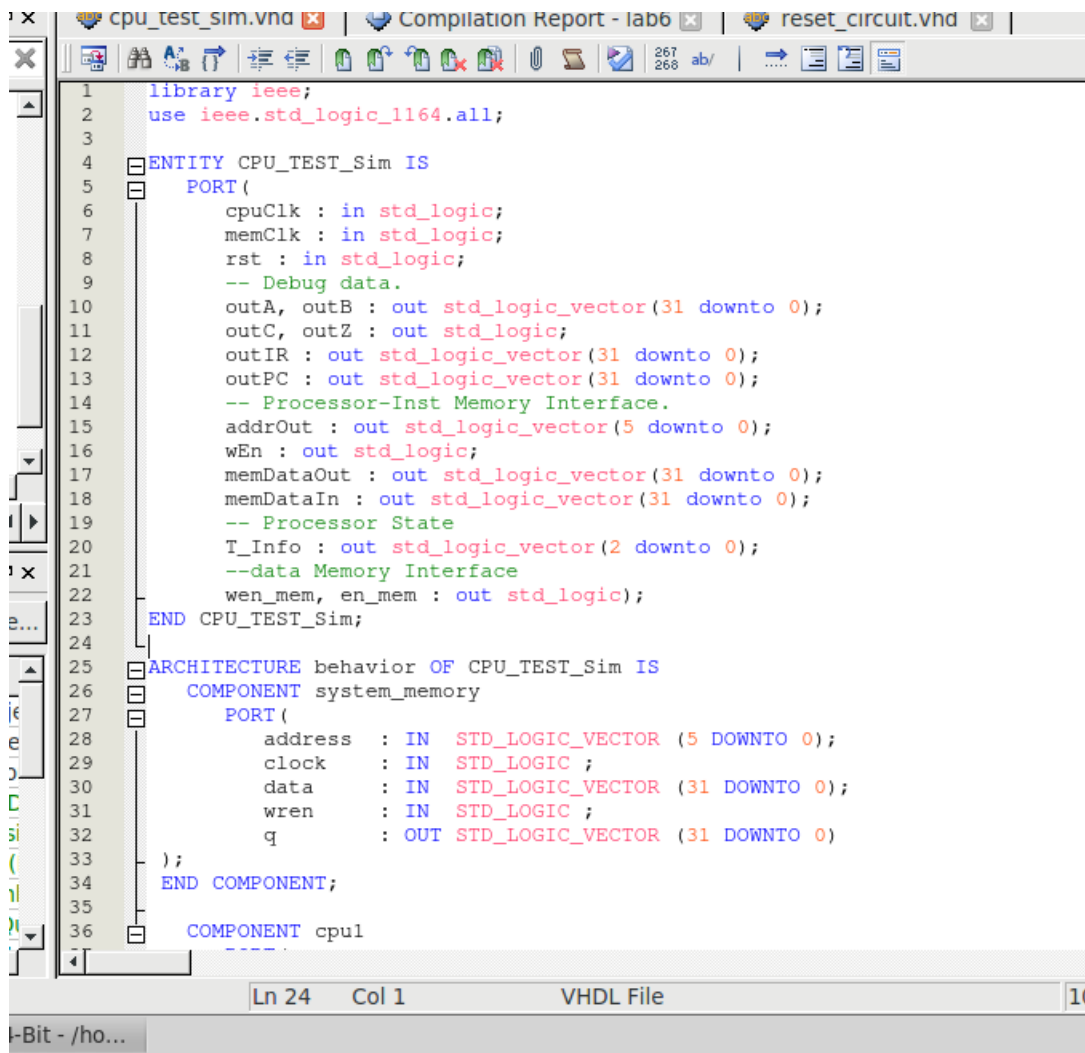


In this lab, we will combine data path and control from the previous labs with the reset circuit and form a complete CPU. We will use the complete CPU to perform the features described in the CPU specifics document.



```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  ENTITY CPU_TEST_Sim IS
5  PORT(
6      cpuClk : in std_logic;
7      memClk : in std_logic;
8      rst : in std_logic;
9      -- Debug data.
10     outA, outB : out std_logic_vector(31 downto 0);
11     outC, outZ : out std_logic;
12     outIR : out std_logic_vector(31 downto 0);
13     outPC : out std_logic_vector(31 downto 0);
14     -- Processor-Inst Memory Interface.
15     addrOut : out std_logic_vector(5 downto 0);
16     wEn : out std_logic;
17     memDataOut : out std_logic_vector(31 downto 0);
18     memDataIn : out std_logic_vector(31 downto 0);
19     -- Processor State
20     T_Info : out std_logic_vector(2 downto 0);
21     --data Memory Interface
22     wen_mem, en_mem : out std_logic);
23 END CPU_TEST_Sim;
24
25 ARCHITECTURE behavior OF CPU_TEST_Sim IS
26     COMPONENT system_memory
27     PORT(
28         address : IN  STD_LOGIC_VECTOR (5 DOWNTO 0);
29         clock : IN  STD_LOGIC ;
30         data : IN  STD_LOGIC_VECTOR (31 DOWNTO 0);
31         wren : IN  STD_LOGIC ;
32         q : OUT STD_LOGIC_VECTOR (31 DOWNTO 0)
33     );
34     END COMPONENT;
35
36     COMPONENT cpul
```

Ln 24 Col 1 VHDL File 10

I-Bit - /ho...

cpu\_test\_sim.vhd x | Compilation Report - lab6 x | reset\_circuit.vhd x

267  
268  
ab/

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

```
COMPONENT cpu1
PORT (
    clk      : in std_logic;
    mem_clk  : in std_logic;
    rst      : in std_logic;
    dataIn   : in std_logic_vector(31 downto 0);
    dataOut  : out std_logic_vector(31 downto 0);
    addrOut  : out std_logic_vector(31 downto 0);
    wEn      : out std_logic;
    dOutA, dOutB : out std_logic_vector(31 downto 0);
    dOutC, dOutZ : out std_logic;
    dOutIR   : out std_logic_vector(31 downto 0);
    dOutPC   : out std_logic_vector(31 downto 0);
    outT     : out std_logic_vector(2 downto 0);
    wen_mem, en_mem : out std_logic);
END COMPONENT;

signal cpu_to_mem: std_logic_vector(31 downto 0);
signal mem_to_cpu: std_logic_vector(31 downto 0);
signal add_from_cpu: std_logic_vector(31 downto 0);
signal wen_from_cpu: std_logic;

BEGIN
-- Component instantiations.
main_memory : system_memory
PORT MAP (
    address => add_from_cpu(5 downto 0),
    clock => memClk,
    data => cpu_to_mem,
    wren => wen_from_cpu,
    q => mem_to_cpu
);
main_processor : cpu1
PORT MAP (
    clk => cpuClk,
```

Ln 24 Col 1 VHDL File

```
57
58 BEGIN
59   -- Component instantiations.
60   main_memory : system_memory
61     PORT MAP (
62       address => add_from_cpu(5 downto 0),
63       clock => memClk,
64       data => cpu_to_mem,
65       wren => wen_from_cpu,
66       q => mem_to_cpu
67     );
68   main_processor : cpu1
69     PORT MAP (
70       clk => cpuClk,
71       mem_clk => memClk,
72       rst => rst,
73       dataIn => mem_to_cpu,
74       dataOut => cpu_to_mem,
75       addrOut => add_from_cpu,
76       wEn => wen_from_cpu,
77       dOutA => outA,
78       dOutB => outB,
79       dOutC => outC,
80       dOutZ => outZ,
81       dOutIR => outIR,
82       dOutPC => outPC,
83       outI => T_Info,
84       wen_mem => wen_mem,
85       en_mem => en_mem
86     );
87
88   addrOut <= add_from_cpu(5 downto 0);
89   wEn <= wen_from_cpu;
90   memDataOut <= mem_to_cpu;
91   memDataIn <= cpu_to_mem;
92 END behavior;
```

Ln 24 Col 1 VHDL File 100%

CPU\_TEST\_Sim compiles successfully:

Applications Places System

Quartus II 64-Bit - /home/student1/mcoskun/Documents/coe608/lab6/lab6 - lab6

File Edit View Project Assignments Processing Tools Window Help

lab6

Project Navigator

- Files
  - UZE.vhd
  - system\_memory.vhd
  - system\_memory.qip
  - system\_memory.mif
  - reset\_circuit.vwf
  - reset\_circuit.vhd.bak
  - reset\_circuit.vhd
  - register32.vhd
  - RED.vhd
  - pc.vhd
  - mux4to1.vhd
  - mux2to1.vhd
  - LZE.vhd
  - lab6\_description.txt
  - lab6.qsf
  - lab6.qpf
  - fulladd.vhd
  - data\_path.vhd
  - data\_path.qws
- Hierarchy
- Files
- Design Units

Flow Summary

Flow Status	Successful - Fri Mar 25 04:40:15 2022
Quartus II 64-Bit Version	14.0.2 Build 209 09/17/2014 SJ Full Version
Revision Name	lab6
Top-level Entity Name	CPU_TEST_Sim
Family	Cyclone IV E
Device	EP4CE75F29C7
Timing Models	Final
Total logic elements	714 / 75,408 ( < 1 % )
Total combinational functions	708 / 75,408 ( < 1 % )
Dedicated logic registers	169 / 75,408 ( < 1 % )
Total registers	169
Total pins	209 / 427 ( 49 % )
Total virtual pins	0
Total memory bits	10,240 / 2,810,880 ( < 1 % )
Embedded Multiplier 9-bit elements	0 / 400 ( 0 % )
Total PLLs	0 / 4 ( 0 % )

Flow: Full Design

Task	Time
Compile Design	00:01:08
Analysis & Synthesis	00:00:14
Fitter (Place & Route)	00:00:32
Assembler (Generate programming files)	00:00:07

100% 00:01:08

Quartus II 64-Bit - /ho... [lab6]

## Reset\_circuit VHDL code:

```

1  Library ieee;
2  USE ieee.std_logic_1164.ALL;
3  USE ieee.std_logic_arith.ALL;
4  USE ieee.std_logic_unsigned.ALL;
5
6  ENTITY reset_circuit IS
7      PORT (
8          Reset: IN STD_LOGIC;
9          Clk: IN STD_LOGIC;
10         Enable_PD: OUT STD_LOGIC := '1';
11         Clr_PC: OUT STD_LOGIC
12     );
13 END reset_circuit;
14 ARCHITECTURE Behavior OF reset_circuit IS
15     TYPE clkNum IS (clk0, clk1, clk2, clk3);
16     SIGNAL present_clk: clkNum;
17 BEGIN
18     process(Clk)begin
19         if rising_edge(Clk) then
20             if Reset = '1' then
21                 Clr_PC <= '1';
22                 Enable_PD <= '0';
23                 present_clk <= clk0;
24             elsif present_clk <= clk0 then
25                 present_clk <= clk1;
26             elsif present_clk <= clk1 then
27                 present_clk <= clk2;
28             elsif present_clk <= clk2 then
29                 present_clk <= clk3;
30             elsif present_clk <= clk3 then
31                 Clr_PC <= '0';
32                 Enable_PD <= '1';
33             end if;
34         end if;
35     end process;
36 END Behavior;

```

We compile reset\_circuit VHDL code successfully.

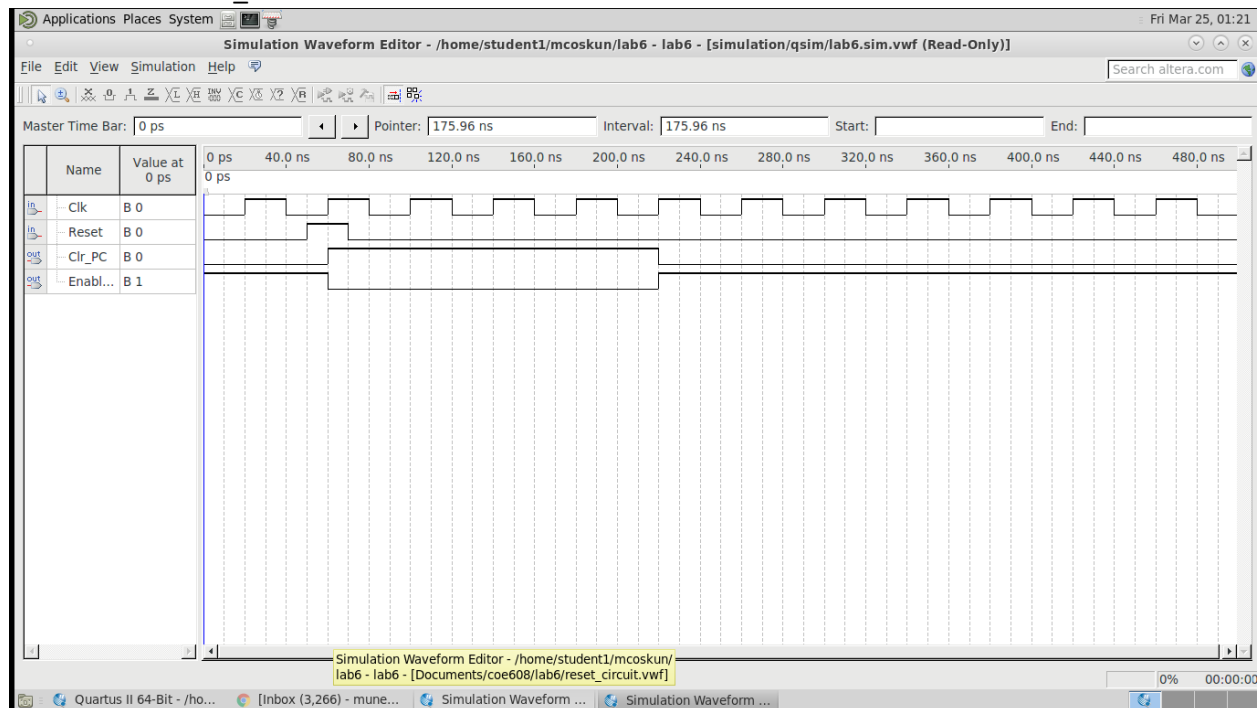
**Flow Summary**

Flow Status	Successful - Fri Mar 25 01:34:54 2022
Quartus II 64-Bit Version	14.0.2 Build 209 09/17/2014 SJ Full Ver
Revision Name	lab6
Top-level Entity Name	CPU_TEST_Sim
Family	Cyclone IV E
Device	EP4CE75F29C7
Timing Models	Final
Total logic elements	714 / 75,408 ( < 1 % )
Total combinational functions	708 / 75,408 ( < 1 % )
Dedicated logic registers	169 / 75,408 ( < 1 % )
Total registers	169
Total pins	209 / 427 ( 49 % )
Total virtual pins	0
Total memory bits	10,240 / 2,810,880 ( < 1 % )
Embedded Multiplier 9-bit elements	0 / 400 ( 0 % )
Total PLLs	0 / 4 ( 0 % )

**Messages**

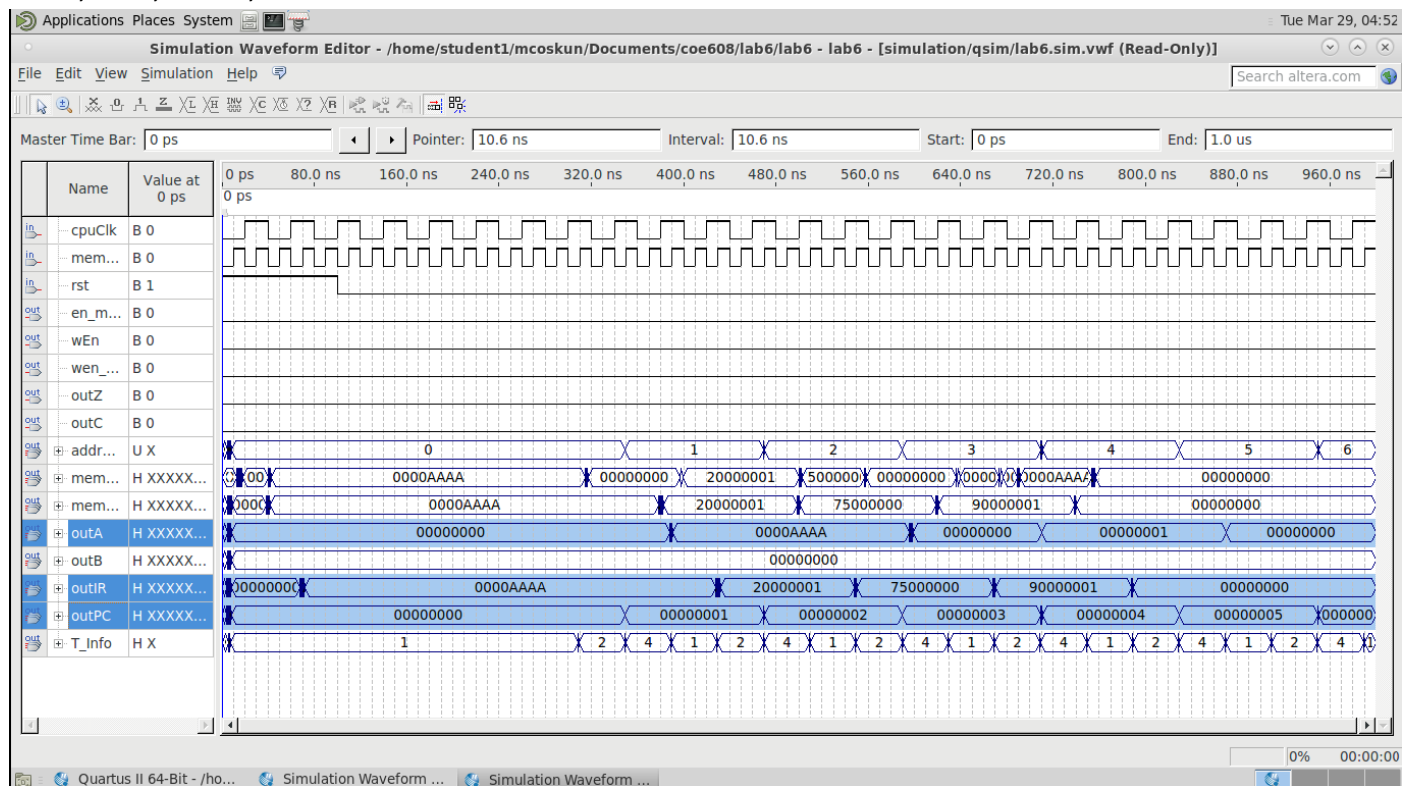
Type	ID	Message
Warning	332102	Design is not fully constrained for setup requirements
Warning	332102	Design is not fully constrained for hold requirements
Success		Quartus II 64-Bit TimeQuest Timing Analyzer was successful. 0 errors, 5 warnings
Success		Running Quartus II 64-Bit EDA Netlist Writer
Success		Command: quartus_eda --read_settings_files=off --write_settings_files=off lab6
Success	204019	Generated file lab6.vo in folder "/home/student1/mcoskun/simulation/qsim/"
Success		Quartus II 64-Bit EDA Netlist Writer was successful. 0 errors, 0 warnings
Success	293000	Quartus II Full Compilation was successful. 0 errors, 90 warnings

## Waveform of reset\_circuit



Timing waveforms for the rest of the lab manual as directed:

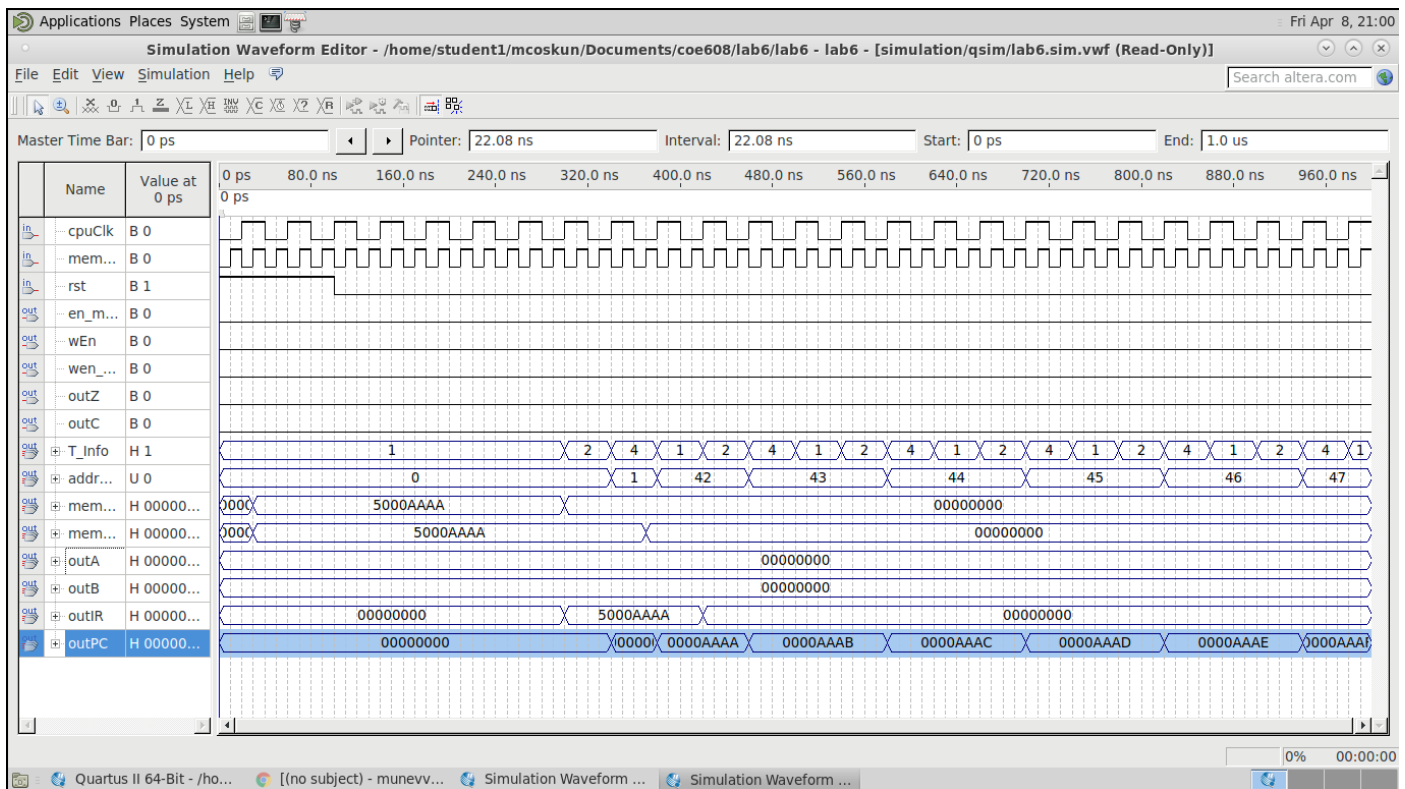
## LDAI, STA, CLRA, LDA



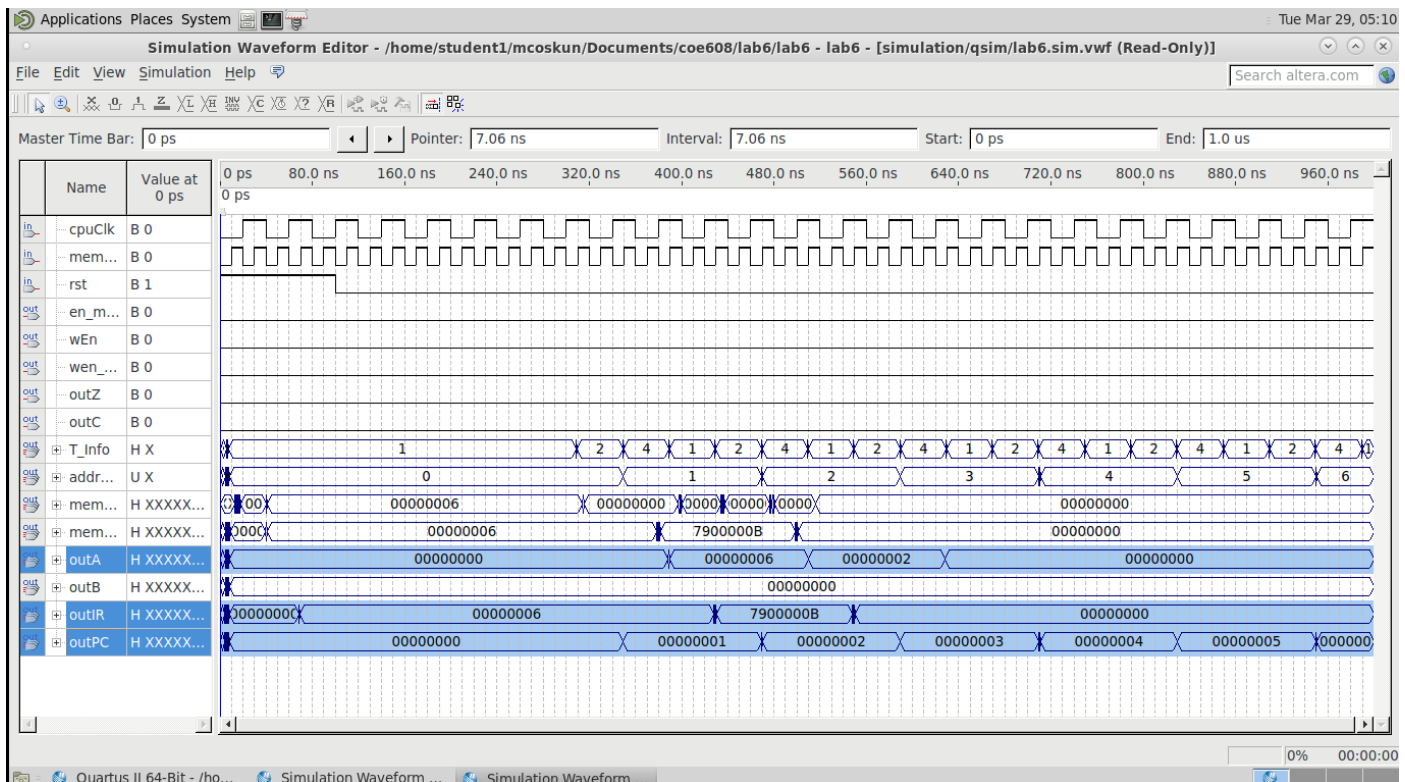
## LDBI, STB, CLRB, LDB



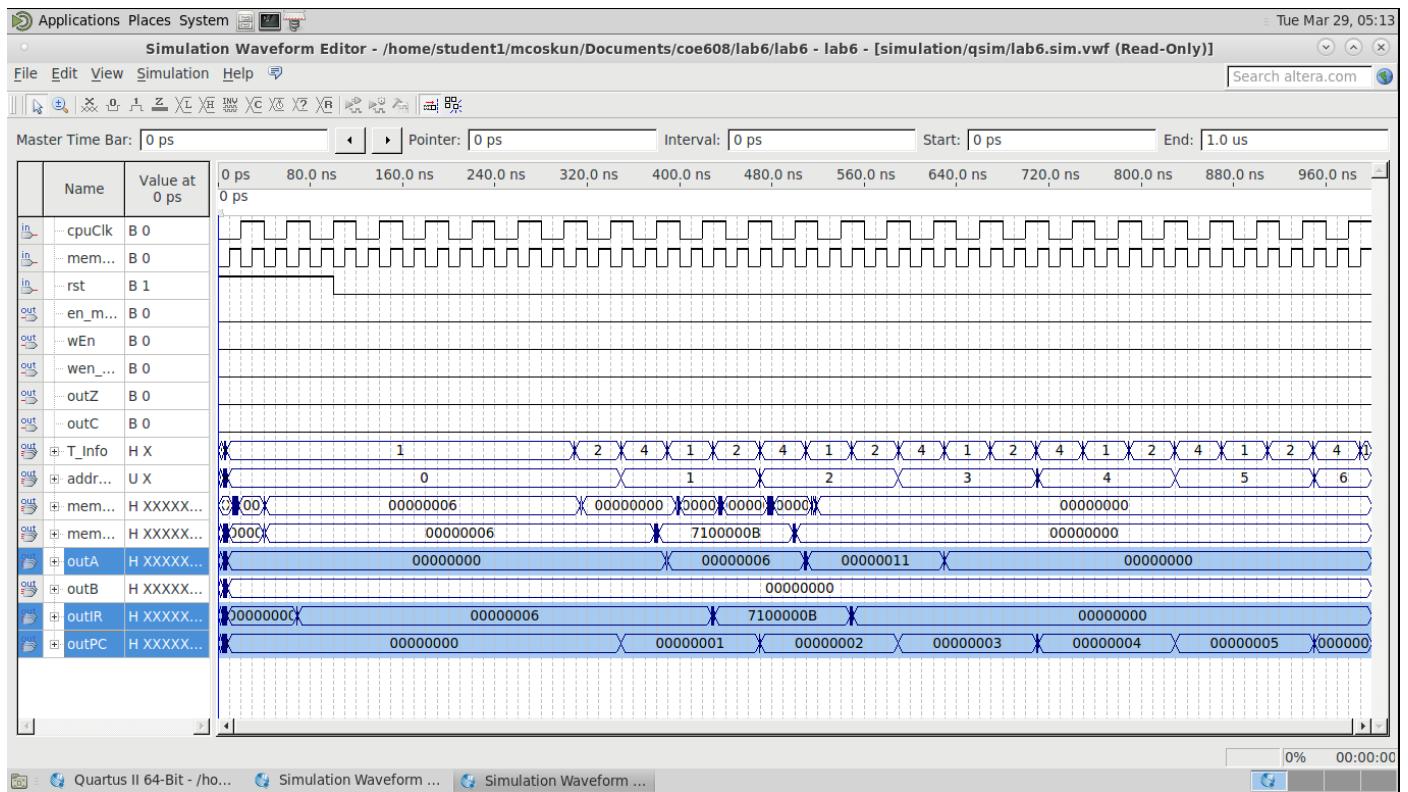




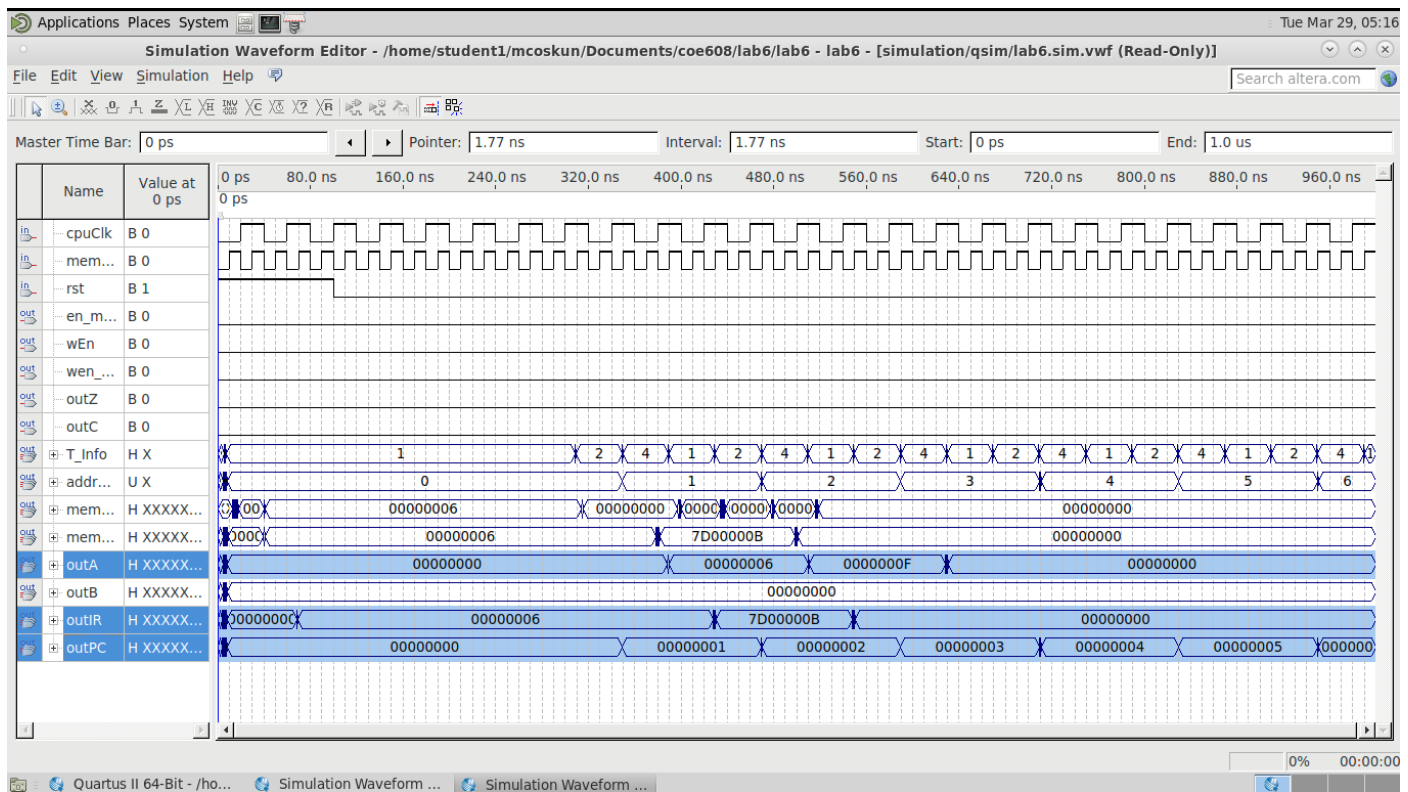
## ANDI



# ADDI



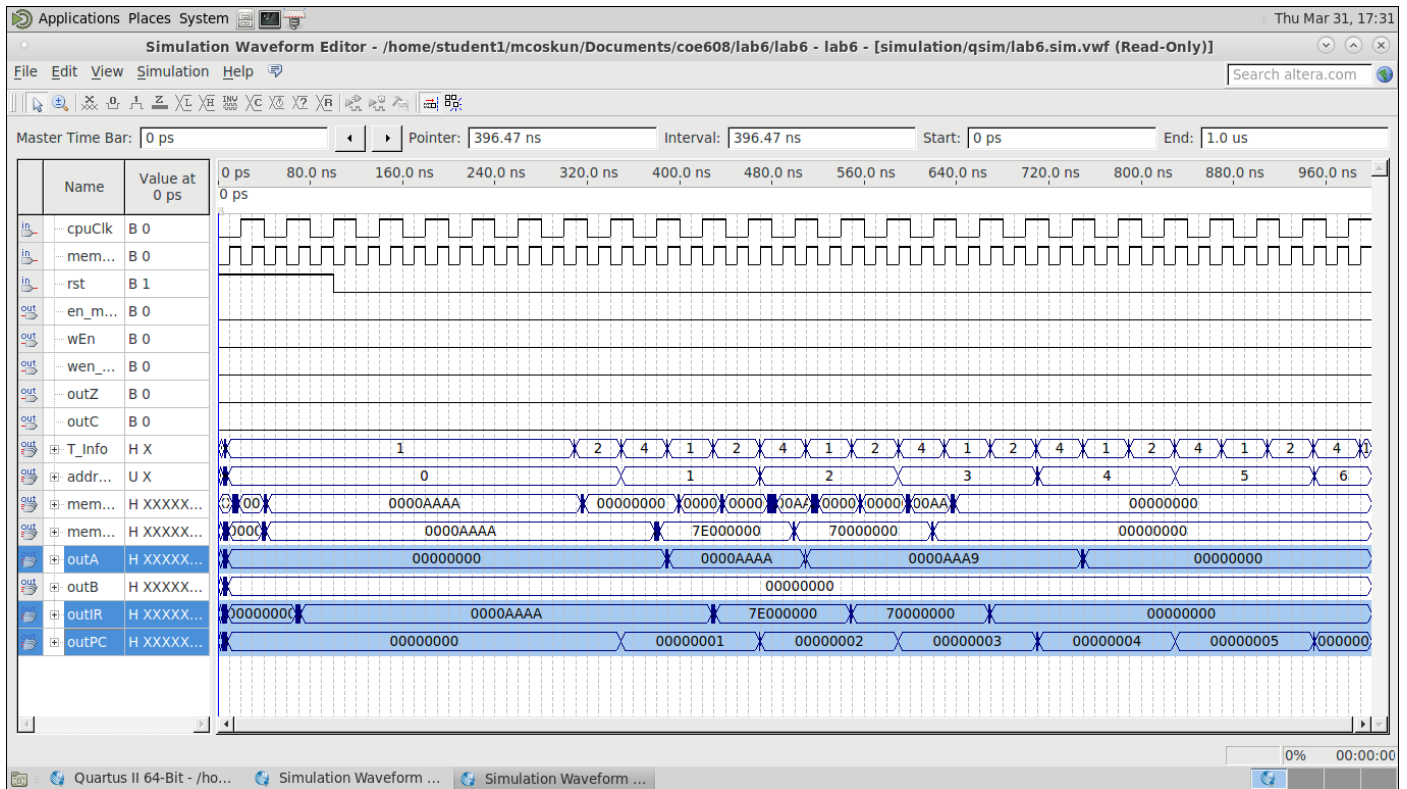
ORI



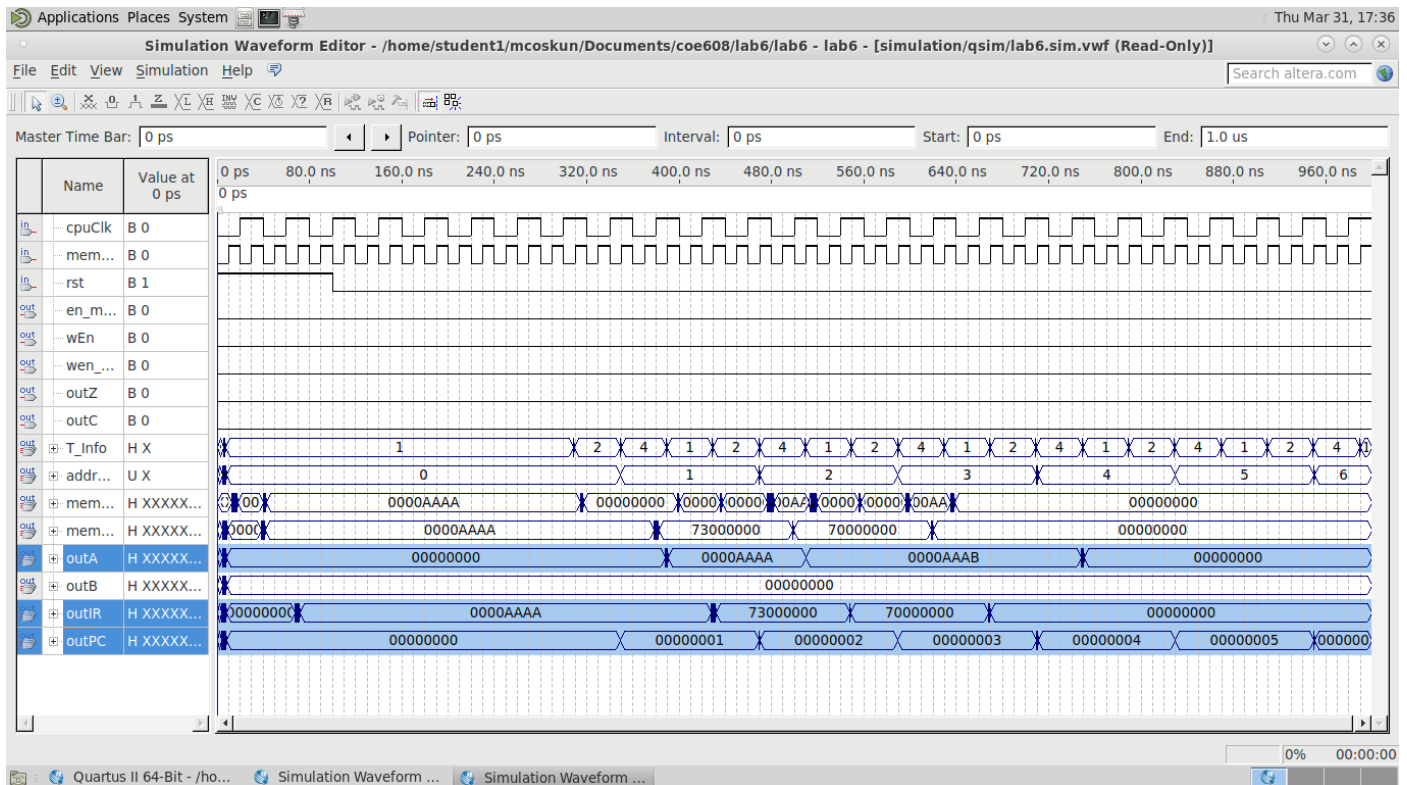
ADD



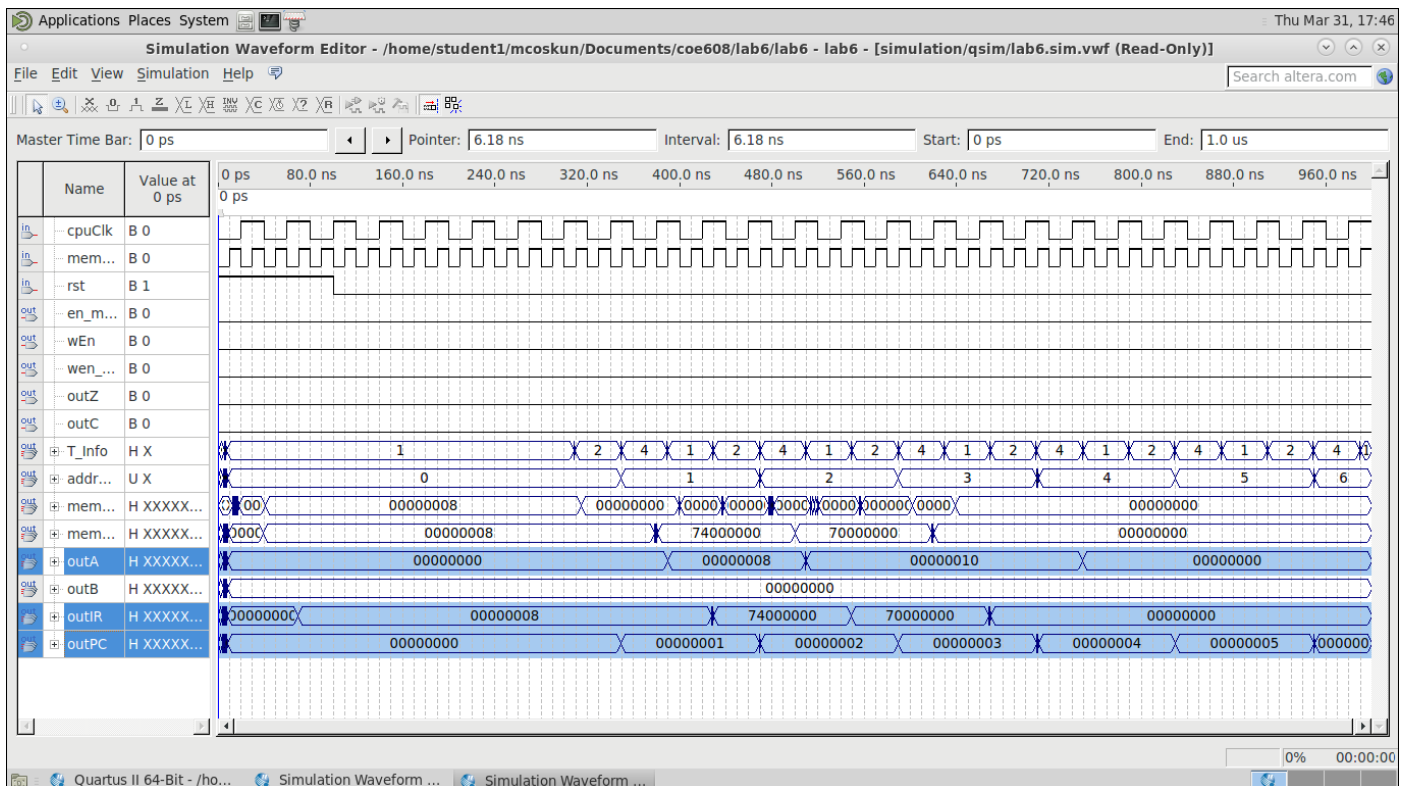




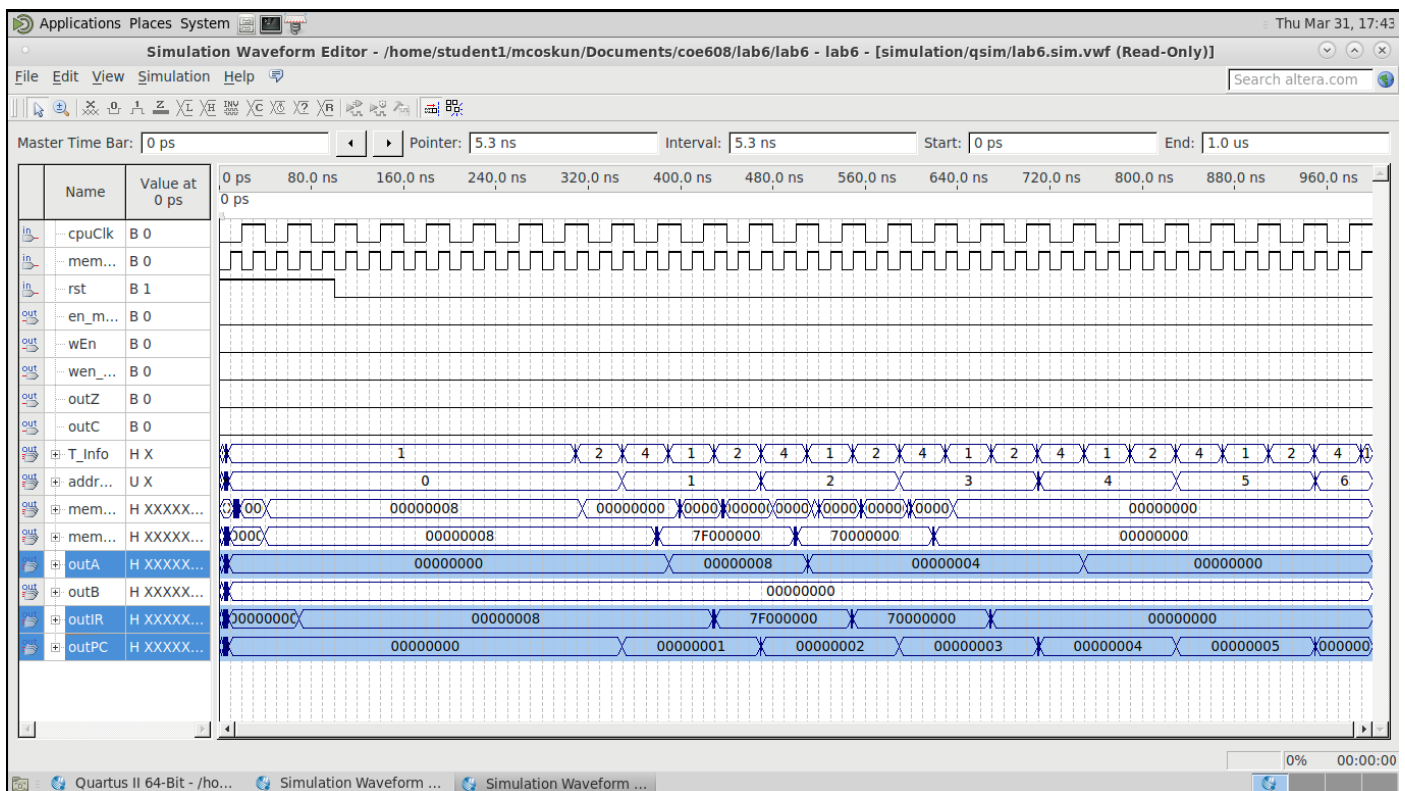
# INCA



## ROL



ROR



BEQ

