Lab #5: Data Register and Instruction Register

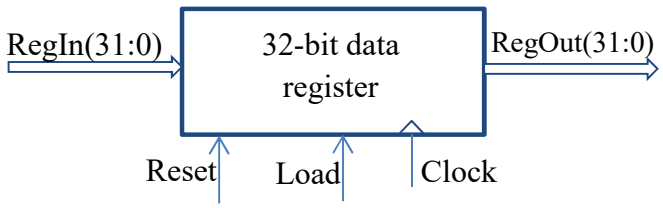
I. Data Register

1. Requirement

Design a 32-bit register with synchronous reset that has the following entity declaration:

```
entity Reg32bits is
Port(
      RegIn     : IN  std_logic_vector(31 downto 0); -- Input
      Load      : IN  std_logic; -- Load the input value when Load='1'
      Clock     : IN  std_logic;
      Reset     : IN  std_logic; -- Reset the value of the register
      RegOut    : OUT std_logic_vector (31 downto 0));
end Reg32bits;
```

The block diagram of this data register is as following:



In your test bench, it should show that your register is loading the input value only at Load = 1 and not at Load = 0; your register is storing the value inside, when loaded, without changing it; and your register value is reset to "0000_0000" whenever the reset is set to 1.

2. Pre-lab

- Study the processor registers (what it is, its purposes, where it is in the memory hierarchy, etc.)

3. Lab

- Write the VHDL code for 32-bit data register
- Simulate using Xilinx ISE simulator

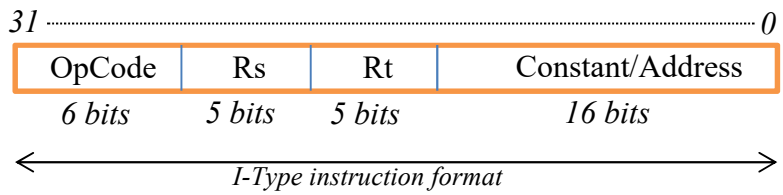
4. Deliverables

- VHDL program
- VHDL test bench with enough test cases
- Waveform with comments

II. I-type Instruction Format Register

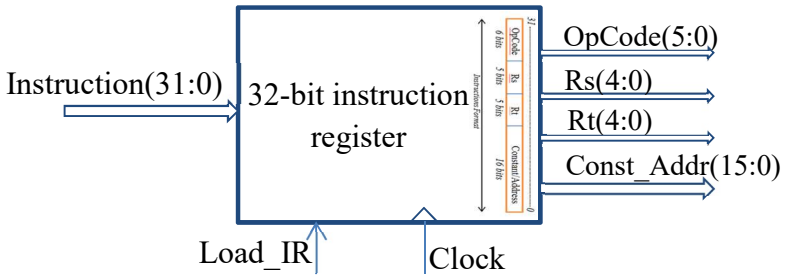
1. Requirement

Design a 32-bit instruction register which has the data format similar to MIPS **I-format** instruction:



Opcode: Operation code
Rs: first source register number
Rt: second source register number

The block diagram of this register is as following:



In your test bench, it should show that your register is loading the input value only at Load_IR = 1 and not at Load_IR = 0; your register is storing the value inside, when loaded, without changing it. This register has 4 outputs: Opcode, Rs, Rt and Const_Addr. In your test bench, you should include enough number of test cases to show the correctness of your design.

2. Pre-lab

- Study MIPS instructions formats in lecture notes chapter 2 of Computer Architecture class (R-Type, I-Type).
- Which instructions use I-Type format? Give some examples.

3. Lab

- Write the VHDL code for I-Type instruction format register
- Simulate using Xilinx ISE simulator

4. Deliverables

- VHDL program
- VHDL test bench with enough test cases
- Waveform with comments