Introduction to Quartus Development Environment
Due date: January 21, 2005.

1 Objectives

In this exercise, we will be working on an incorrect design of 16x16 register file written in VHDL. The goals of this lab include:

1. Get familiar with Altera Quartus.
2. Get familiar with including user's design components in a large design.
3. Perform debugging through waveform simulation.

2 Overview

Register file is a major component in most microprocessors. It is used to stored data that can be quickly accessed by the functional units (e.g. Arithmetic Logic Units). It is commonly presented as an array of registers. In this lab, we will be working with a register file that contains sixteen registers. Each register can store 16-bit wide data. The overview of the register file is presented in Figure 1.

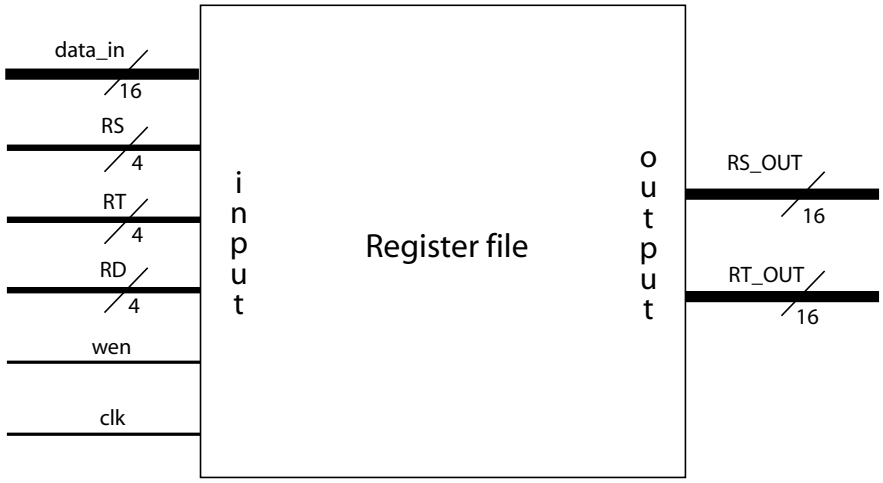

Figure 1: 16 x 16 register file

When the input values are provided through RS and RT inputs, the RS_OUT and RT_OUT would have the values stored in the corresponding registers specified by RS and RT. For example (Figure 2), if RS is set to 4 and RT is set to 6, the RS_OUT and RT_OUT would have the values of 0x1104 and 0x1106, respectively.

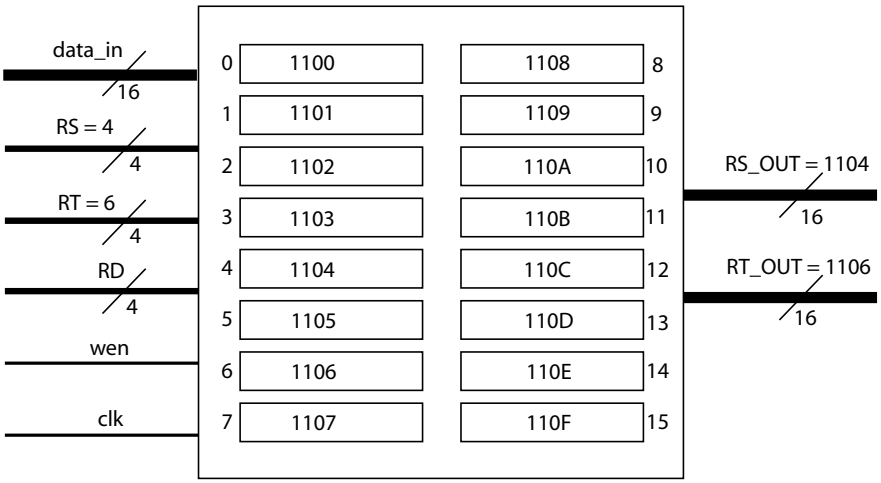

Figure 2: Providing values for RS and RT

Input RD is used to specified the destination register. The value to be written is specified by input *data_in*. In order for a register to be written, we also need to have the write-enable (*wen*) set to '1' and a rising edge on the clock (*clk*) input. In Figure 3, the data_in input is set to 0x1210 and the RD input is set to 6. However, the *wen* input is not set and thus, register 4 will never be written regardless of the clock signal (*clk* input)

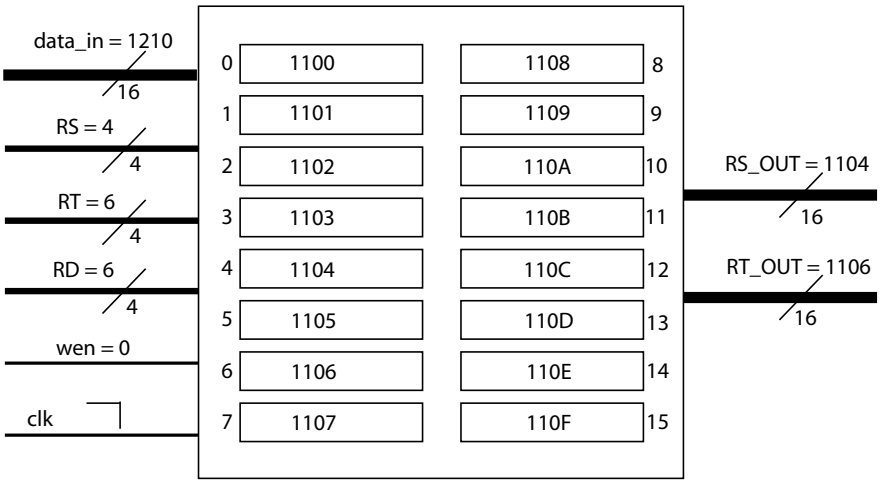

Figure 3: Providing values for *data_in* and RD but wen is not set

In Figure 4-left, the *wen* input is set to '1'; however, the rising edge of the clock have yet to arrive. Thus, register 6 is not written. On the other hand, Figure 4-right indicates that the rising edge has just arrived. Thus, register 6 is written with the new value 0x1210. Notice, since we never change the values of RT, RT_OUT now has the most recent value of register 6.

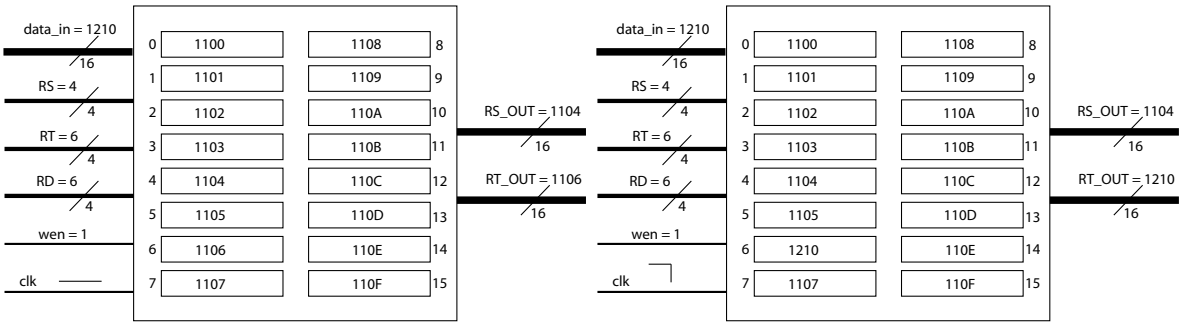

Figure 4: *wen* is now set to '1'

3 Inlab

1. Download sample component file from the course website (under assignment page). If you are using the lab machine, save the file into c:/csce351_lab directory. Unzip the downloaded file in the same directory.

2. Start Quartus and choose "New Project Wizard". When the wizard asks for the working directory, choose the recently unzipped directory. You should set the top-level design entry and project name to "regfile" then choose "Next".

3. On the next page, choose "add all". Make sure that the order is "project_const.vhd", "mux16.vhd", and "regfile.vhd". Use up and down buttons to adjust the order. Then choose "Next".

4. Choose "Next".

5. Choose "Stratix" device family. Also choose "Yes, I want to specify a specific device". Choose "Next".

6. Choose "EP1S10F780C5" then choose "Finish".

We will go over the steps to perform simulation in class. After you are done with simulation, you should notice that the design is incorrect. Based on the simulation result, fix "regfile.vhd" so that the simulation result is correct.

4 Submission Procedure

Submit the correct "regfile.vhd" through handin by noon on Friday January 21, 2005.