

CS447 Lab #9 (Week 10)

Introduction

For this lab, you will use Logisim to build a register file, and you will use control signals to selectively update and read the registers' contents.

Note: This Lab must be submitted online by 11:59 pm on Monday, November 7th. Please see the instructions on the final page.

Objective: Build and use a register file

This lab will take advantage of the skills you learned for Lab #8, and will also make use of some of the new control components that were covered in lecture this week. You might find it helpful to consult the Lab #8 instructions if you have forgotten how to perform some of the steps described for this lab.

Set up the components for the Register File as a separate circuit

As the canvas for your register file, create a new circuit in Logisim. For simplicity, we will create a four-register file, with each register holding the state of 8 bits. We will use Logisim's built-in registers – you don't need to construct your own out of D flip-flops.

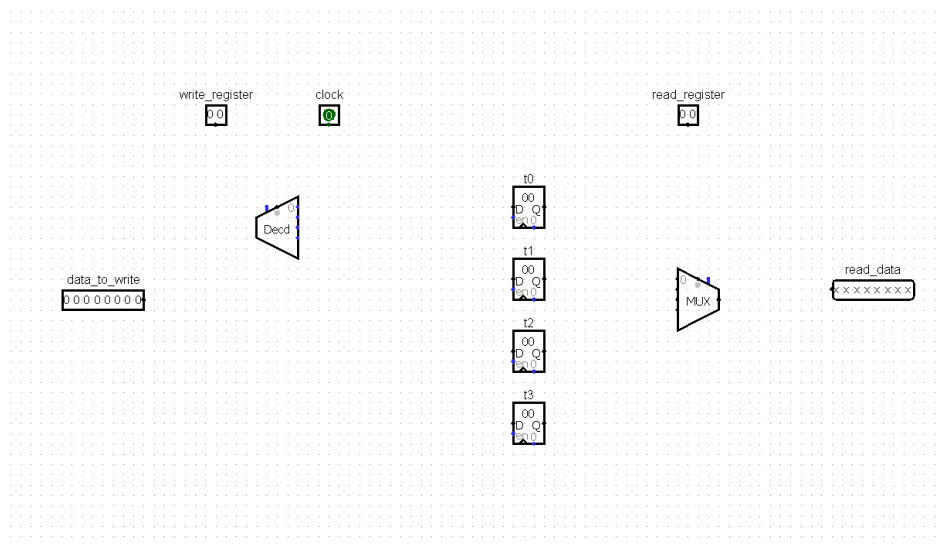
Your register file will be equipped to write to any single register on a clock cycle, and to read from any single register. Your circuit will include components that will allow you to select which registers to read and write.

Steps to create the register file:

1. Create 4 registers. The **register** component may be found under the **Memory** folder.
 - Be sure to adjust the bit width!
 - For the trigger, choose either Rising or Falling Edge. Just remember to be consistent across registers!
 - You may find it helpful to give your registers labels (for example, t0, t1,..etc as in MIPS)
2. Create an input pin for the Clock.
3. Create an input pin for the Data that may be written to the registers.
 - The bit width for this pin should match the bit width of the Register contents.
4. Create an output pin for the Data that you read out of the registers.
 - Again, adjust the bit width.
 - You may find it helpful to adjust "Three-state?" to no.
5. Create an input pin to select which of the four registers should be written.
 - Bit width = 2 ($2^2 = 4$ unique register specifiers)
6. Create an input pin to select which of the four registers should be read.

- Again, adjust the bit width.
7. Create a decoder to selectively enable a register for write access
 - The **decoder** component is located within the **Plexers** folder
 - Select bits = 2
 8. Create a multiplexer to selectively route one register's contents to be read
 - The **multiplexer** component is also located within **Plexers**
 - Again, adjust the select bits
 - Data bits = Register bits

At this stage, your canvas may look something like this:



Wire together the components of the Register File

The details of this step are left as a problem-solving exercise. However, here are some hints:

1. In Lab #8, each of the D flip-flops inside your register required inputs to three of its pins, and to read the contents, we attached a wire to a fourth pin. Since each register can be composed of an array of D flip-flops, the same input and output sites will be used on each of your registers. (There is also one register pin that you will not use.)
2. As stated above, the decoder plays an important role in choosing which register to write. Therefore, it will need to receive input from the appropriate selection bits, and implement some form of control to enable a write to only the one selected register. (Check out the Library Reference or the 11/1 slides for a refresher on Decoders).
3. The data to write will need to be somehow routed to the registers.
4. As was also stated above, the multiplexer plays an important role in choosing which register's contents will be read. Again, you will need to make use of some selection bits,

and implement some form of control to allow only one register's contents to pass through to the output pins.

Use the Register File in the main circuit

In this final part of the lab, you will set up a Main circuit that will allow for very simple testing of your register file. Again, this step will be described at a high level, and you are encouraged to consult the previous lab if you have forgotten how to set up multiple circuits within a single Logisim file.

1. In the Main canvas, place a single instance of your Register File.
2. Create an input pin that will allow you to manually enter a data value to write to the Register File.
3. Create an output pin that will allow you to view a value read from the register file.
4. Create two sets of selection inputs: one to choose the register to write, and one to choose the register to read.
5. Create a clock.
6. Wire these new components to the appropriate sites on your Register File component.

Test the Register File

To test your register file, try writing different values to different registers. Subsequently, ensure that you can read back the different values that you wrote. Remember that the clock must tick (automatically or manually) in order for the registers to update.

Submission:

If you finish during recitation, please notify the TA, so that your credit for this assignment can be confirmed right away.

Regardless of whether you finish during recitation or not, you must submit your work online. Save the file as **YourPittUserName_lab9.circ** and submit the file via the appropriate link in Courseweb (see Course Documents/Week 10 Lab). The lab must be submitted by **Monday, November 7th, by 11:59 pm.**