ECEN3763 – Homework Week 9

Spring, 2022

Due Monday March 14th, end of day – 15 Points

This assignment provides an introduction to Signal Tap. You will use Homework Week 8 as the starting point for this assignment.

After completing this assignment, you will submit a .qar file for your completed project.

Part 1: Begin with your Homework Week 8 project and add a SignalTap core

1) Open a functioning version of the Week 8 homework assignment. You may either use the existing project or create a new project based on the project. Compile your project.

2) Add a Signal Tap instance to your project by going to File > New > Signal Tap Logic Analyzer File. The SignalTap GUI will open.

3) Begin in the Signal Configuration pane. The first step is to select the sample clock that will be used as the sample clock. Click the search box to the right of the Clock entry, and the Node Finder window will appear.

4) A small box with two arrows is located to the right of the List button. If the arrows are pointing down, click on the box. You need to set Options that are hidden if the arrows are pointing downward. In the Filter: drop down, select Signal Tap: pre-synthesis, and confirm the Look In: entry is pointing to your project.

5) Press the List button, and select CLOCK_50 as the sample clock. You can leave all other Signal Configuration settings as the default settings.

6) Double click in the blank area where it says “Double-click to add nodes”. Click the list button and expand the entry in the list of nodes that matches the name of your Platform Designer component. Find the entry with the _master appended to the component name, and add these signals:
    master_read
    master_readdatavalid
    master_reset_reset
    master_waitrequest
    master_write

Now expand the entry for your 8 bit PIO, and add these signal:

chipselect
write_n
address
data_out
out_port
readdata
writedata

What did you just create?

Nothing yet. You configured the settings for a SignalTap core, and now you must recompile your project so that Quartus can create the SignalTap core, and make the requested connections.

The signals you added to the SignalTap core will be captured on the rising edge of CLOCK_50. You have a set of Avalon Master signals, as well as Avalon Slave signals for your PIO core. It is often tricky to identify the exact signals you want to capture, it takes practice with the SignalTap tool, familiarity with the Avalon signal naming, and trial and error.

Part 2: Test your SignalTap Core

1) In the SignalTap GUI, Processing > Start Rapid Recompile. Rapid Recompile is a special flow where Quartus attempts to use the previous version of your routed design, and simply adds the SignalTap core. This should speed up the compilation.

2) When the Assembler step completes in Quartus, program your board using the programming pane in the SignalTap GUI. While you are waiting, review the Quartus Flow Summary. Notice the large number of registers and the memory blocks now being used. Much of this is due to the addition of the SignalTap core.

3) Back in the SignalTap GUI, click on your SignalTap core in the Instance Manager pane, and click the Run Analysis icon (first icon following Instance Manager:) or press F5. Since there is no trigger setup, hit Escape (on your keyboard) to stop the acquisition.

Part 3: Test your design

You will setup up a trigger condition in SignalTap, then use commands in System Console to start a transaction.

1) In the SignalTap GUI (click on the Setup tab to get to the proper view), select the PIO write_n signal, right click in the Trigger Conditions column, and select falling edge. Start an acquisition with Run Analysis. Note: When I added the trigger condition, SignalTap indicated a recompile was necessary. I ignored this and started the acquisition, and SignalTap worked without the recompile.

   It should not be necessary to recompile for simple trigger changes.

2) Open System Console with Tools > System Debugging Tools > System Console.

3) Setup the master paths as you did in Homework 8, and then perform a write to the 8 bit PIO. This write will trigger data capture in the SignalTap core. Confirm that you see the Avalon transactions captured in Signal Tap. Do the data values match the pattern you wrote to the PIO?