Digital Systems

Prof. James Clark

ECSE 325

Lab #3: Timing constraint specification and timing analysis using TimeQuest

Winter 2018

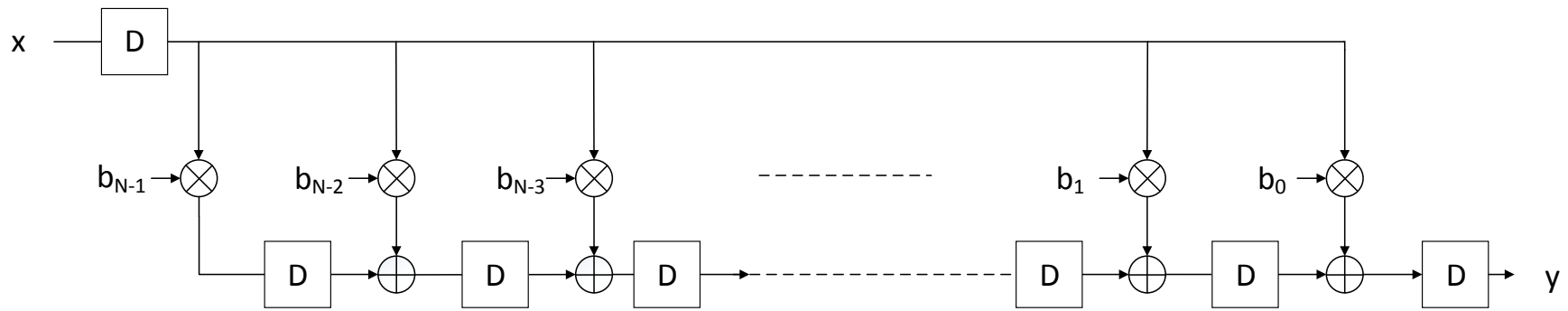
Introduction

In this lab, you will learn how to specify timing constraints and perform static timing analysis of the synthesized circuit using the TimeQuest timing analyzer. You will also learn two techniques to reach timing closure for a time-critical circuit.

FIR Filters–Implementation

In this lab, you will implement a different type of FIR filter, in broadcasting form. You will use the same input files from Lab 3 to verify your design. Quantization for the ilter's input, output signals and weights in the fixed-point representations (1,15), (2,15) and (1,15), respectively.

Note: Use constant parameters to represent the weights.



FIR Filters–Simulation

Once you have implemented your design in VHDL, you need to verify its functionality by writing a testbech code. In the testbench code, you will read the given test vector in your testbench and obtain an output test vector. The obtained output signal must match with the given output signal.

When you have finished the VHDL codes, show them to the TA.

Using the TimeQuest Timing Analyzer

To ensure a properly working circuit, the designer must take into consideration various timing constraints. In class we saw that for a register to correctly store an input value, the input must be held stable for a period (called the setup time) before the clock edge, and also for a period (called the hold time) after the clock edge.

Whether a circuit meets these timing constraints can only be known after the circuit is synthesized. After synthesis is done one can analyze the circuit to see if the timing constraints (setup and hold times) are satisfied using the term slack. Slack is the margin by which a timing requirement is met or not met. It is the difference between the required time and the arrival time. A positive slack value indicates the margin by which a requirement was met. A negative slack value indicates the margin by which a requirement was not met.

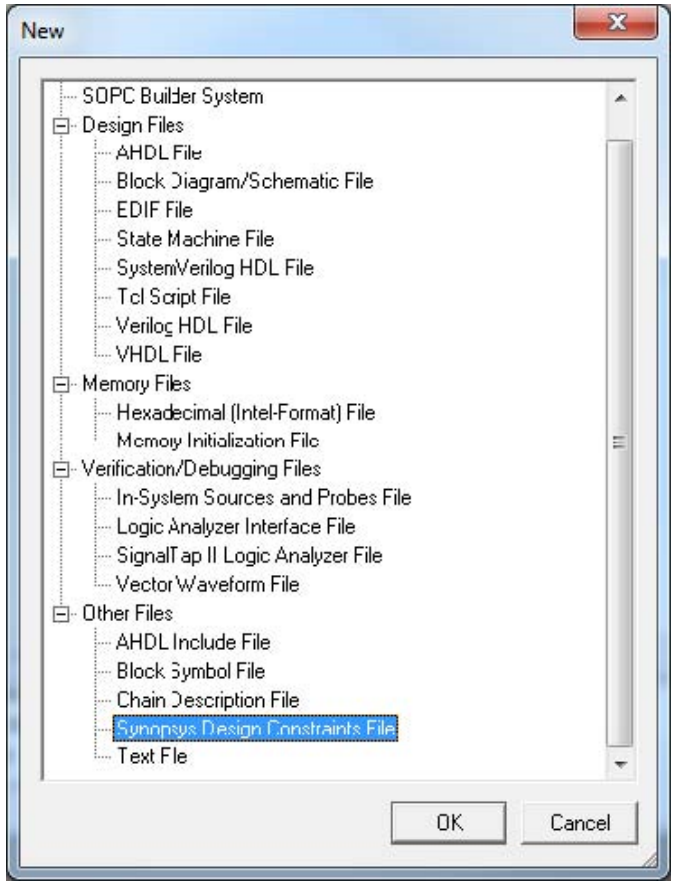
In Quartus II, timing analysis can be done using the TimeQuest Timing Analyzer. Please read pages 7-6 through 7-12 of the document "Altera TimeQuest Timing Analyzer.pdf" before proceeding to the next part of the lab.

Using the TimeQuest Timing Analyzer

From the "File/New" menu item, select "Synopsys Design Constraints File".

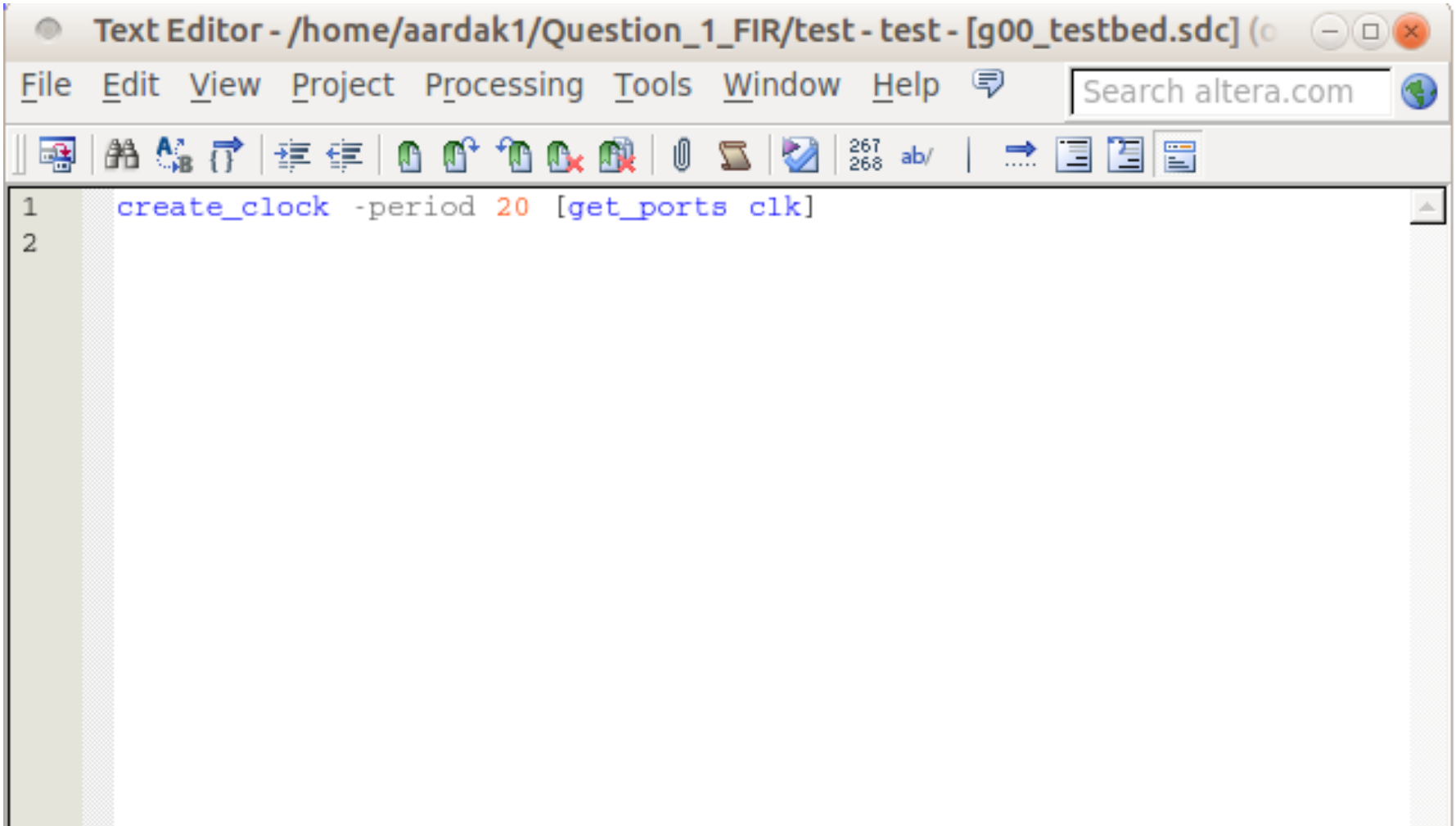
This ".sdc" file is where we can specify various timing constraints for our design.

We will add a single constraint specifying the clock period.



Using the TimeQuest Timing Analyzer

Type the following text into the window, and save the file with the name "gNN_testbed.sdc" where NN is your group number. This will tell the timing analyzer that your clock period is 20 ns.



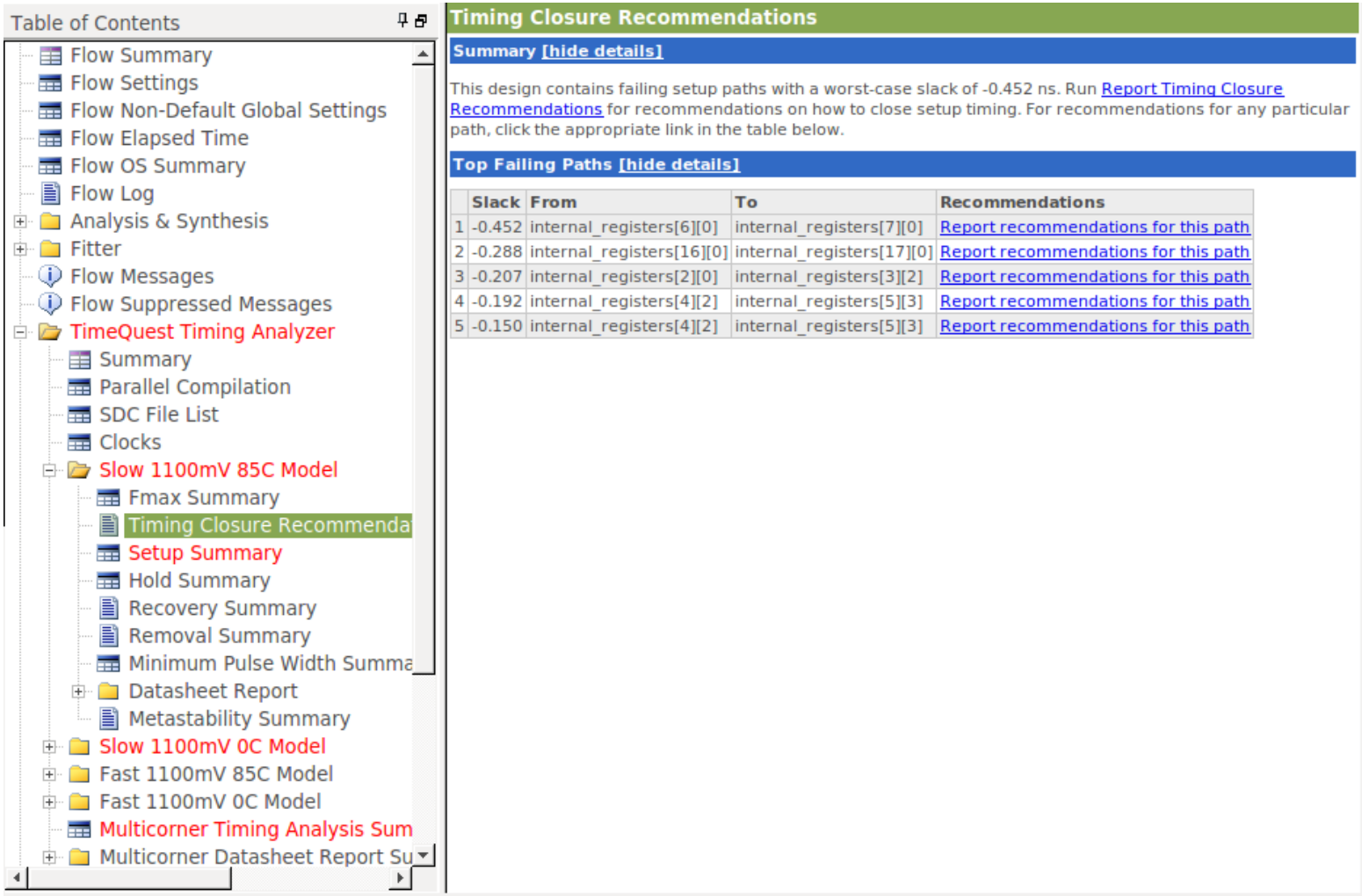
Using the TimeQuest Timing Analyzer

Recompile your design. The Timing Analyzer will read the .sdc file and use the constraint information when doing its analysis.

After the compilation is finished, look at the timing summaries for the **Slow Model** and the **Fast Model**. Pay attention to the **Fmax Summary** (which gives the maximum clock speed) and the **Setup** and **Hold** summaries (which give the slack amounts for the setup and hold constraints). Find the maximum achievable frequency of your design and show it to the TA. Does your design meet the specified timing constraint (i.e., the clock period of 20 ns)?

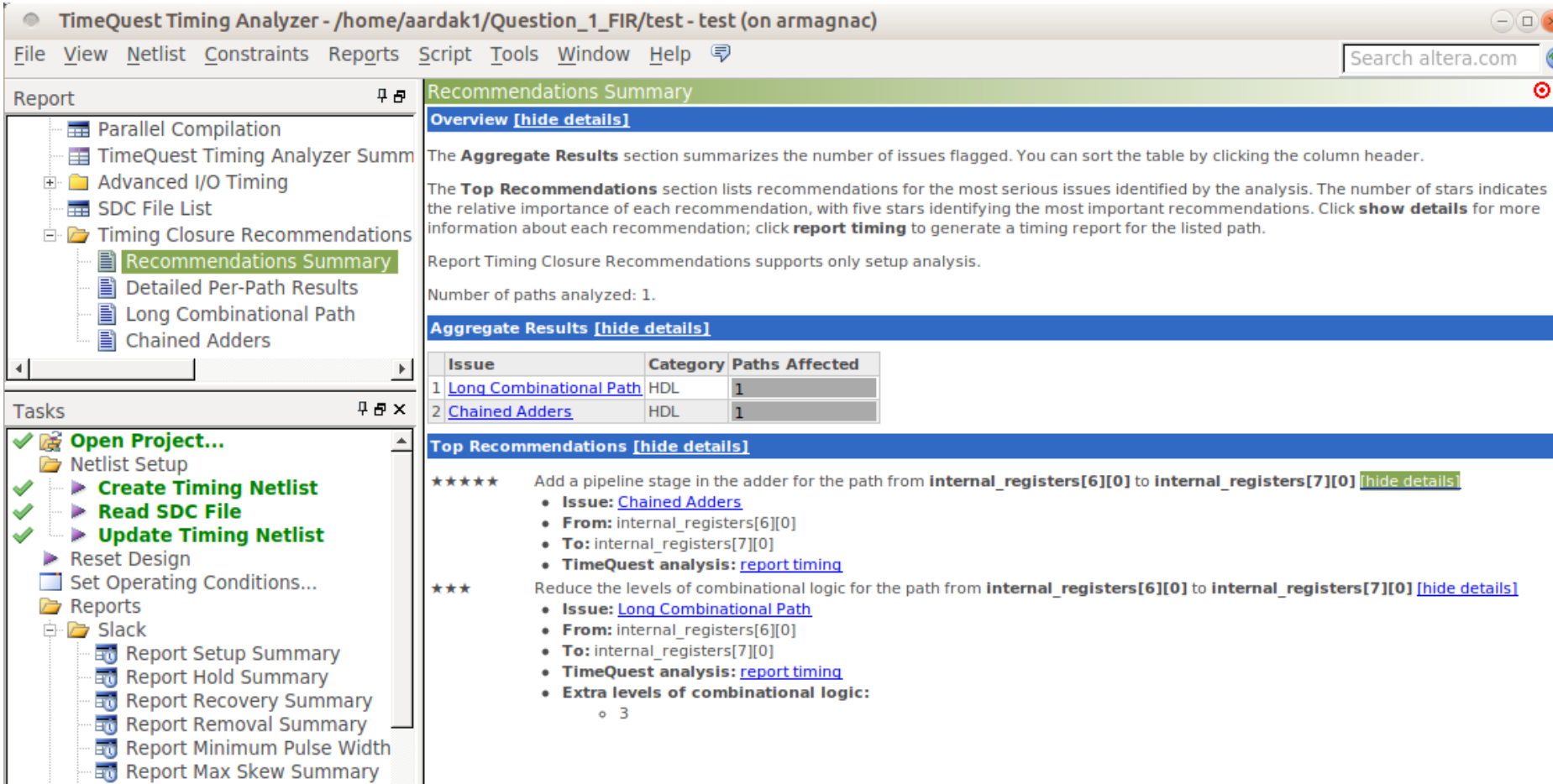
Using the TimeQuest Timing Analyzer-Timing Violation

In case of any timing violation, Quartus recommends solutions for each violated path in your design. Denote violated paths in your design on Fig. 1 and show them to the TA.



Using the TimeQuest Timing Analyzer-Timing Violation

Click on the "Report recommendations for this path" to open up TimeQuest Timing Analyzer and get more information on the violated paths. Under the "Recommendation Summary", you can find possible solutions to fix the timing violations.



Writeup the Lab Report

You are required to submit a written report and your code to myCourses on the first Friday after the Lab3 period at midnight.

- The report in the electronic file should be in PDF format.
- The report should be written in the standard technical report format.
- Document every design choice clearly.
- Everything should be organized for the grader to easily reproduce your results by running your code through the tools.
- The code should be well-documented and easy to read.
- The grader should not have to struggle to understand your design.

Writeup of the Lab Report - cont'd

Please refer to the example lab report from myCourses content for an example lab report template.

Your report must include:

- An introduction in which the objective of the lab is discussed
- For both filter designs:
- The VHDL code you wrote for the broadcast FIR filter
- Discussion on the resource utilization of your design (i.e. how many registers are used and why? What changes would you respect in the resource utilization if you increased the bit size of the counter?)
- Testbench VHDL code to verify your filter design
- Content of SDC file, screenshots of timing violations (if any), reported maximum clock frequency

Grading Sheet

Group Number:
Name 1:
Name 2:

Task	Grade	/Total	Comments
VHDL for broadcast		/40	
Testbench VHDL		/40	
Maximum Frequency		/20	