

Sandeep Sankar

3 February 2026

ECE554 Minilab1a Report

A. Github Repository Info

Link: https://github.com/ssankar25/ECE554_Minilabs/tree/main/1a

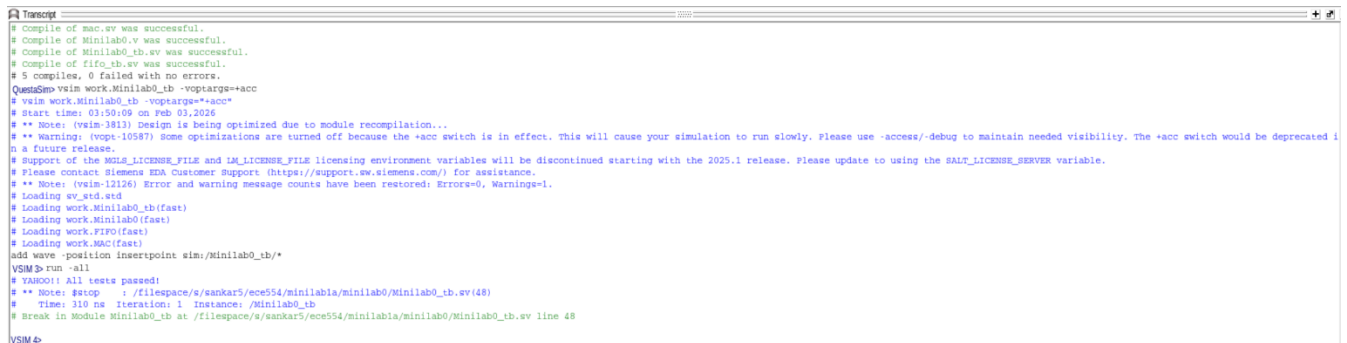
The .sv files for the first part of the project without any IP files are in the root of the repository. The .sv files that contain the implementation with the Quartus-provided IP files are in the “ip” folder of the repository. The waveforms and simulation logs, along with the resource utilization summaries, are located in the “sim” directory.

I created this repository by first creating a new repository from the GitHub homepage. I then used the “add files” and “upload files” features directly from the website to upload these files. This allowed me to easily structure the repository even though the files were in different places on my local machine.

B. Simulation Logs and Waveforms

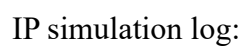
Screenshots for the Minilab0_tb simulation for both the IP and non-IP implementations are shown below.

Non-IP simulation log:



```
Transcript
# Compile of mac.sv was successful.
# Compile of Minilab0.v was successful.
# Compile of Minilab0_tb.sv was successful.
# Compile of fifo_tb.sv was successful.
# 5 compiles, 0 failed with no errors.
QuartusSim veim work.Minilab0_tb -voptargs=+acc
# vsim work.Minilab0_tb -voptargs=+acc
# Start time: 03:50:09 on Feb 03, 2026
# ** Note: (vsim-3813) Design is being optimized due to module recompilation...
# ** Warning: (vsim-10587) Some optimizations are turned off because the +acc switch is in effect. This will cause your simulation to run slowly. Please use -access/-debug to maintain needed visibility. The +acc switch would be deprecated in a future release.
# Support of the MOLA_LICENSE_FILE and LM_LICENSE_FILE licensing environment variables will be discontinued starting with the 2025.1 release. Please update to using the SALT_LICENSE_SERVER variable.
# Please contact Siemens EDA Customer Support (https://support.sw.siemens.com/) for assistance.
# ** Note: (vsim-12126) Error and warning message counts have been restored: Errors=0, Warnings=1.
# Loading sv_std.sv
# Loading work.Minilab0_tb(fast)
# Loading work.Minilab0(fast)
# Loading work.FIFO(fast)
# Loading work.MAC(fast)
add wave -position insertpoint sim:/Minilab0_tb*
VSIModel>run -all
# VANDU: All tests passed:
# ** Note: $stop : /filepace/s/sankar5/ece554/minilab1a/minilab0/Minilab0_tb.sv(48)
# Time: 310 ns Iteration: 1 Instance: /Minilab0_tb
# Break in Module Minilab0_tb at /filepace/s/sankar5/ece554/minilab1a/minilab0/Minilab0_tb.sv line 48
VSIModel>
```

Non-IP waveforms:



IP waveforms:

C. Resource Utilization

The resource utilization in the IP implementation in the second summary report file shows that this implementation uses less estimated logic ALMs than our implementation. This is likely because the IP implementations are more optimized for synthesis, which is why it requires less dedicated logic registers and functions. This is also why the IP implementation has less fan-out than our implementation.