

ECE-111 Fall-2020

Final Project:

Implement the BitCoin Hashing Algorithm using SystemVerilog.
See Lecture 13 slides for the algorithm and project descriptions.

- Your design(s) must pass the testbench provided in FinalProject.zip: **tb_bitcoin_hash.sv**
- Use the **cycle count** from the **tb_bitcoin_hash.sv transcript** for your delay calculation
- Your sequential design (**Final project 1**) will be relatively graded based on min (**area*delay**)
- Your parallel design (**Final project 2**) will be relatively graded based on min (**delay**) only

Submission instructions: Turn in all of the following files into one compressed file on gradescope, named like (LastName, FirstName)_(LastName, FirstName)_finalproject.zip or (LastName, FirstName)_finalproject.zip if you are working alone.

1. Fill up the finalsummary.xlsx (shared on canvas in final_project.zip)
2. Fill up the Final-Report-Template.docx (shared in final_project.zip)
3. bitcoin_hash1.sv (min area*delay). Add other sv files if you split your designs into different sv files.
4. transcript1.txt (min area*delay)
5. message1.txt (min area*delay)
6. bitcoin_hash1.fit.rpt (min area*delay)
7. bitcoin_hash1.sta.rpt (min area*delay)

Parallel design files (Final project 2) :

8. bitcoin_hash2.sv (min delay). Add other sv files if you split your designs into different sv files.
9. transcript2.sv (min delay)
10. message2.txt (min delay)
11. bitcoin_hash2.fit.rpt (min delay)
12. bitcoin_hash2.sta.rpt (min delay)

Due Date: 12/18/2020, 11:59 pm