EE/CSE371 LabsLand Setup

Introduction

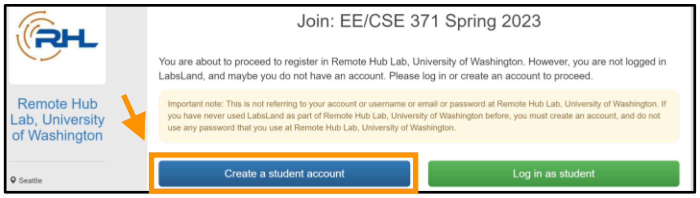
This quarter, we will not be distributing physical lab kits and instead be using *a remote FPGA lab* through the LabsLand web interface. This will avoid the need to carry the hardware around and also means that you don't need to individually procure the necessary peripherals that we will need.

The workflow for the quarter will involve developing, simulating, and debugging the logic of your SystemVerilog code locally in Quartus (as was done in EE 271 / CSE 369), but then instead of generating the bit file locally, you will upload your code to LabsLand to synthesize and run on a DE1-SoC setup.

> ⓘ Because the remote hardware is being shared among all students (and students from other universities!), you will get a limited amount of time per session to use the DE1-SoC, so we will *heavily emphasize* the need to **do thorough testing and debugging before ever using the hardware**.

Creating an Account

1) Go to the following link: https://uw.labsland.com/standalone/join/FKTP5682
2) Select "*Create a Student Account*"

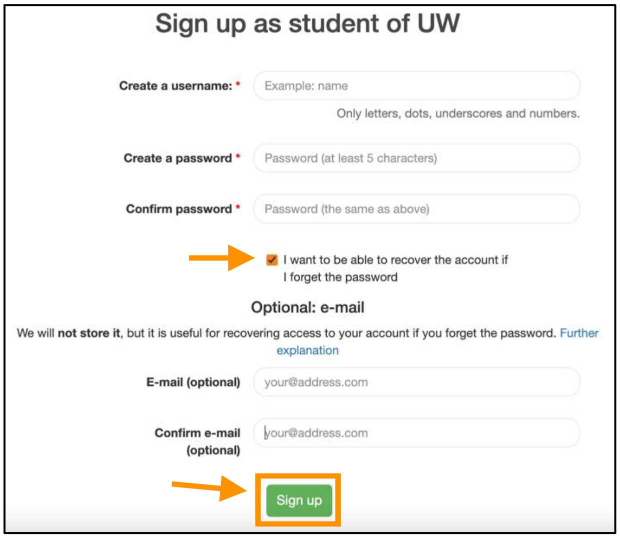


Join: EE/CSE 371 Spring 2023

You are about to proceed to register in Remote Hub Lab, University of Washington. However, you are not logged in LabsLand, and maybe you do not have an account. Please log in or create an account to proceed.

Important note: This is not referring to your account or username or email or password at Remote Hub Lab, University of Washington. If you have never used LabsLand as part of Remote Hub Lab, University of Washington before, you must create an account, and do not use any password that you use at Remote Hub Lab, University of Washington.

3) Sign up using your desired username/password. It is highly recommended that you also select the *"I want to be able to recover the account if I forget the password"* and enter a valid email in case you forget your password. Then select *"Sign up"*.



Sign up as student of UW

Create a username: * Example: name

Only letters, dots, underscores and numbers.

Create a password * Password (at least 5 characters)

Confirm password * Password (the same as above)

☑ I want to be able to recover the account if I forget the password

Optional: e-mail

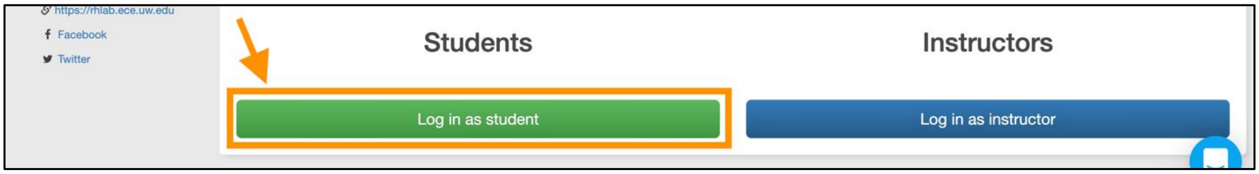
We will **not store it**, but it is useful for recovering access to your account if you forget the password. Further explanation

E-mail (optional) your@address.com

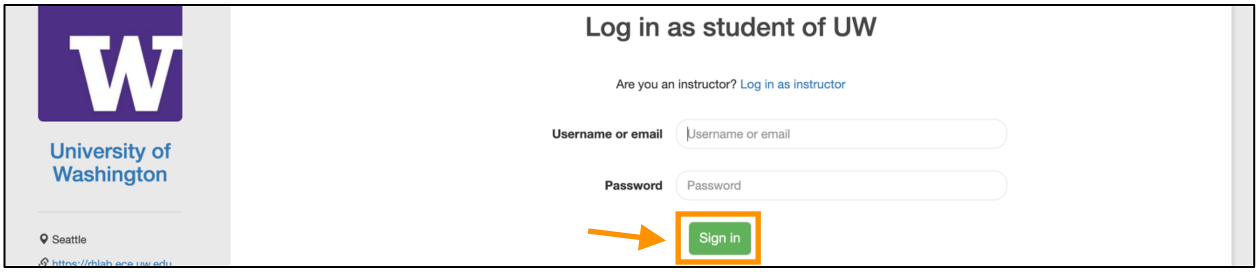
Confirm e-mail (optional) your@address.com

Logging into LabsLand

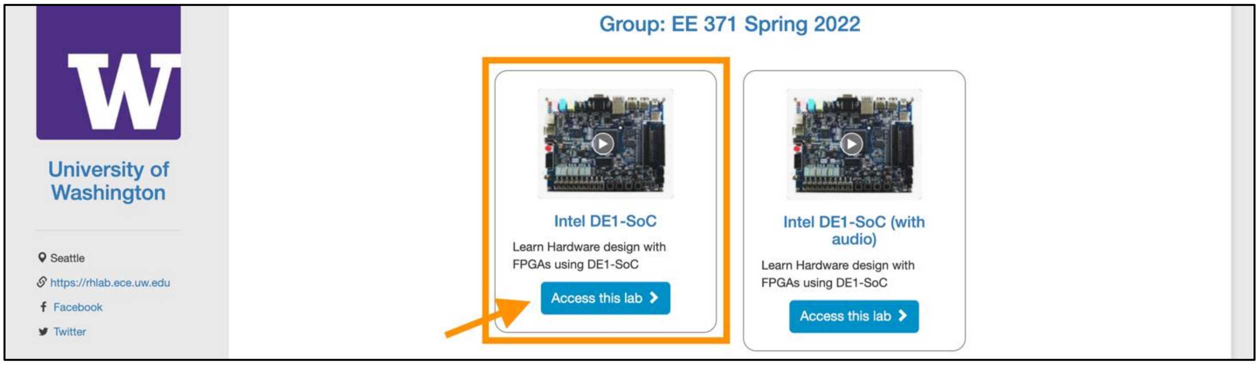
1) Go to the UW LabsLand portal through this link: https://uw.labsland.com/ and click *"Log in as student"*



2) Enter your credentials and click *"Sign in"*.



3) After logging in, you should be able to see the EE/CSE 371's class page on LabsLand, which is shown in the figure below. To access the lab workspace, locate *"Intel DE1-SoC"* and click the *"Access this lab"* below it.


W
University of Washington
Seattle
https://rhlab.ece.uw.edu
Facebook
Twitter
Group: EE 371 Spring 2022
Intel DE1-SoC
Learn Hardware design with FPGAs using DE1-SoC
Access this lab
Intel DE1-SoC (with audio)
Learn Hardware design with FPGAs using DE1-SoC
Access this lab

⚠ Whenever you are working on a lab, please always make sure that the webpage's URL starts with "`uw.labsland.com`"; otherwise, you may be looking at a wrong page and therefore working on incorrect lab materials. To avoid this, please do NOT click the LabsLand logo located on the top-right corner of the page, which will take you back to the generic LabsLand's main page, not the UW LabsLand portal.

Quartus and ModelSim Refresher

Here, we review how to create a Quartus project, write a SystemVerilog program, and simulate your program in ModelSim.

> ⓘ If you don't already have Quartus installed locally, please first refer to the *`Quartus_Install.pdf`* file on the course website before proceeding!
>
> Feel free to skip this task if you feel that you do not need a refresher. Do note that the tutorials below do follow a different workflow than EE 271 and CSE 369.

Watch the series of video tutorials below and follow along. These videos will walk you through the steps of creating a full adder using SystemVerilog and simulating the design in ModelSim. For your reference, the source code used in the tutorials can be found in the Files → Labs → docs → refresher folder on Canvas.

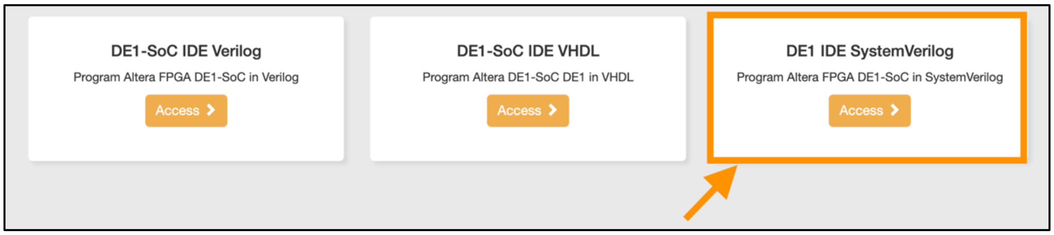
> You will also be performing steps from the tutorial MANY times in subsequent labs, so taking notes on how to do important tasks will save you lots of time later.

1) Launch the Quartus Prime software
2) Create a project from scratch: https://youtu.be/iLbmSTG7bpA
   - Please use the same project name as in the video (`DE1_SoC`).
3) Implement and simulate the full adder: https://youtu.be/BcvclrqZ2fc
   - Note that you may run into a compilation error before you set your top-level module to `fullAdder.sv`.
4) Mapping a SystemVerilog design to an FPGA: https://youtu.be/mnZt2iNNfp4
5) Loading the design onto the FPGA: https://youtu.be/uMxA3VcS3f8
   - Please test your full adder on the DE1-SoC board and verify that it matches the truth table.

You should create future projects in a similar fashion and use SystemVerilog and ModelSim accordingly.

Uploading Code to LabsLand FPGA

1) Locate *"DE1 IDE SystemVerilog"* and click the *"Access"* button below it. You will be directed to a new page called "SystemVerilog IDE for DE1-SoC".

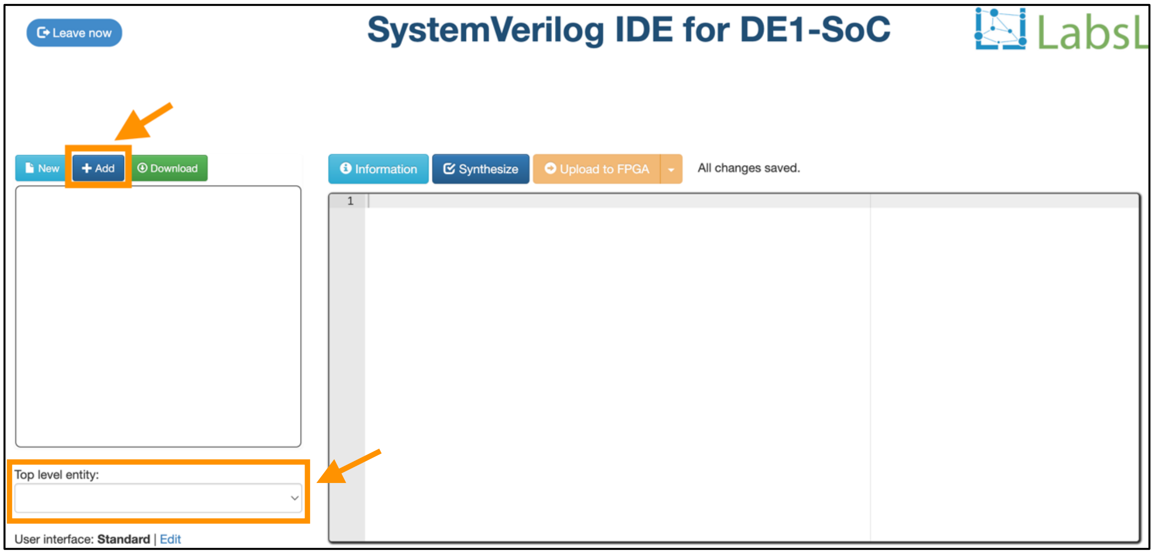


DE1-SoC IDE Verilog
Program Altera FPGA DE1-SoC in Verilog

DE1-SoC IDE VHDL
Program Altera DE1-SoC DE1 in VHDL

DE1 IDE SystemVerilog
Program Altera FPGA DE1-SoC in SystemVerilog

2) In the following page, select the "Add" button to import the top-module "`DE1_SoC.sv`" and file "`fullAdder.sv`" that you created from the Refresher section earlier into this system. You can choose the top-level module using the dropdown menu under "Top level entity". Make sure you select "`DE1_SoC.sv`" as the top module.



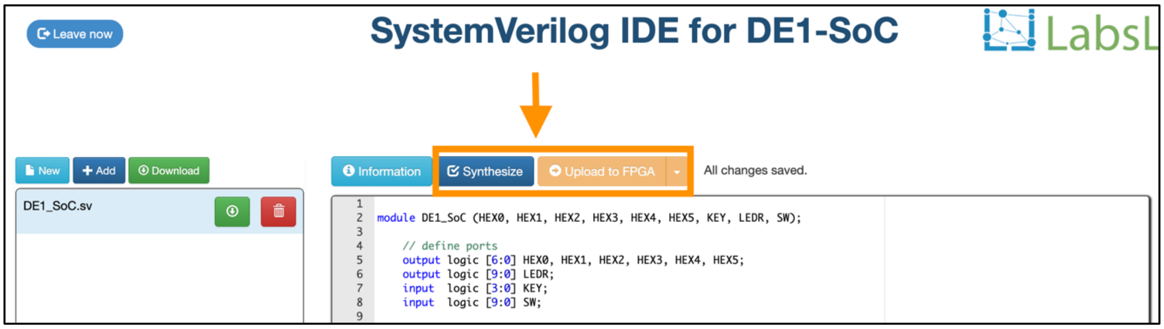
SystemVerilog IDE for DE1-SoC

All changes saved.

Top level entity:

User interface: **Standard** | Edit

3) You will then be able to synthesize the code using the "Synthesize" button. Once the synthesis is complete and succeeds without errors, you can click on "Upload to FPGA" to load your design onto an FPGA.



SystemVerilog IDE for DE1-SoC

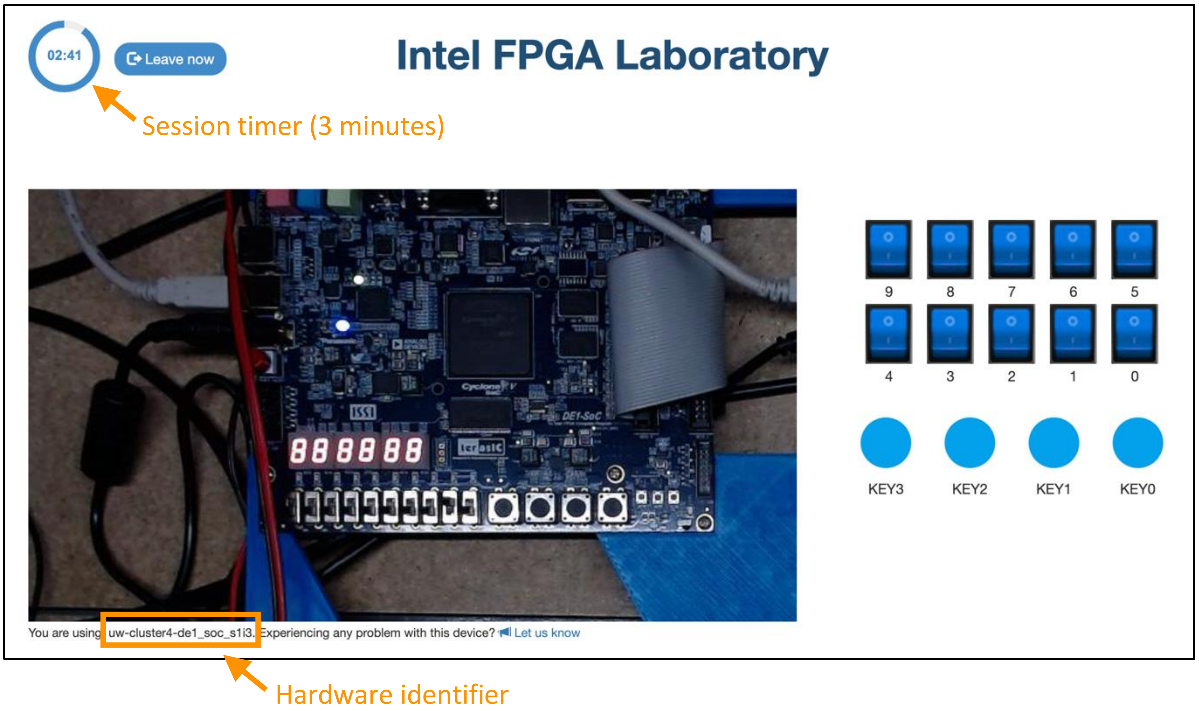
All changes saved.

DE1_SoC.sv

```
module DE1_SoC (HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, KEY, LEDR, SW);

    // define ports
    output logic [6:0] HEX0, HEX1, HEX2, HEX3, HEX4, HEX5;
    output logic [9:0] LEDR;
    input  logic [3:0] KEY;
    input  logic [9:0] SW;
```

4) After waiting for the remote FPGA to connect, you will see the webpage shown below. The right part of the page shows the buttons and keys of the FPGA. You can click on the buttons and keys accordingly as inputs. It is important to note that 'KEYS' need to be held down, as they do not function like switches.



Intel FPGA Laboratory

Session timer (3 minutes)

You are using uw-cluster4-de1_soc_s1i3. Experiencing any problem with this device? Let us know

Hardware identifier