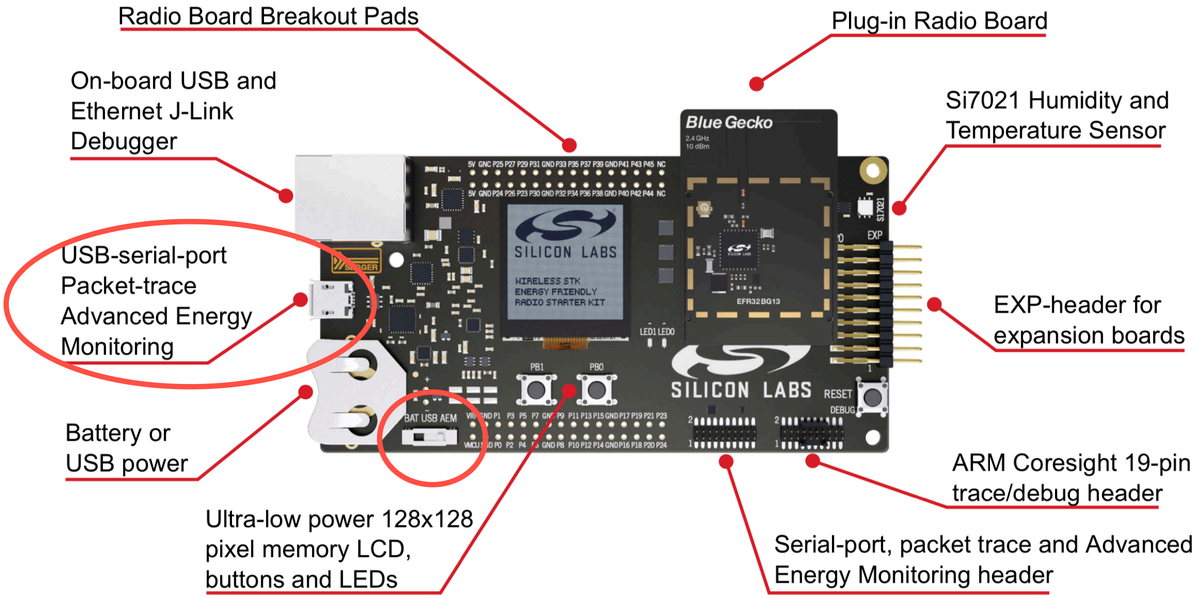
ECEN 5823 Fall 2020
Internet of Things Embedded Firmware
Assignment #1 - Simplicity Exercise

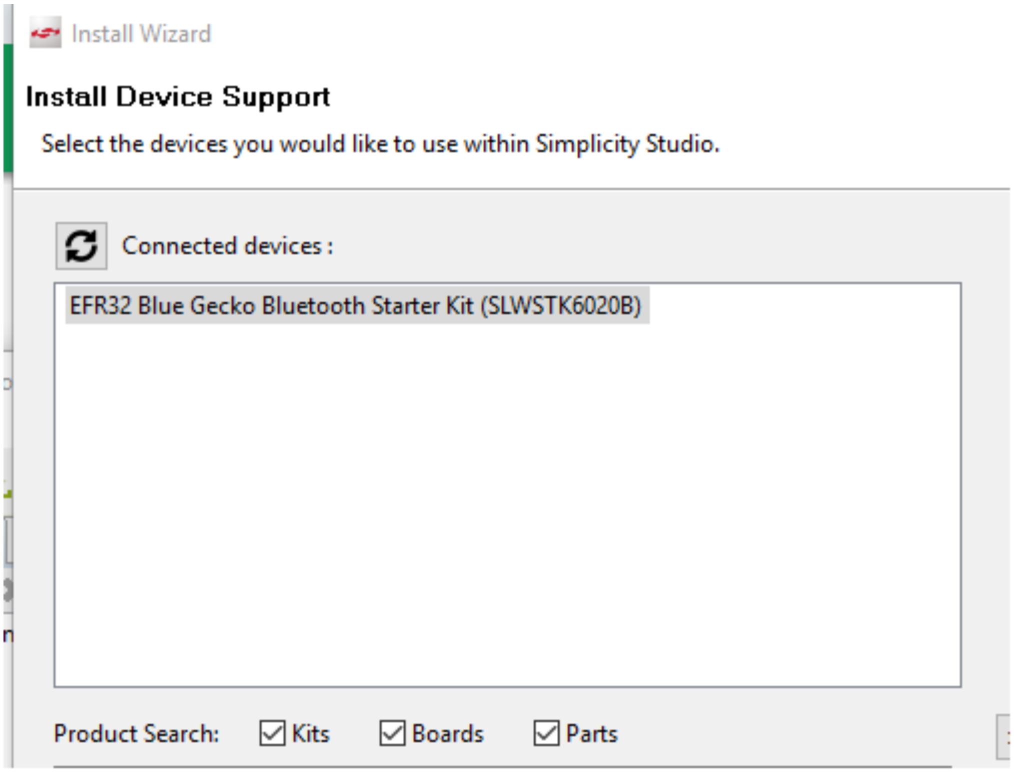
Objective: Install and become familiarized with the Silicon Labs' Simplicity Studio development environment. Complete the setup of Github Classroom used for submission of classroom exercises. Revisions after the initial release of this document are highlighted in yellow.

Instructions:

1. Connect your Silicon Labs' Blue Gecko SLWSTK6020B starter kit to your computer via the supplied USB cable, with the BRD4104A radio board module connected to it. Nothing else should be connected to the base board. **Ensure that the Power Source Select is to the AEM position**.


Radio Board Breakout Pads
Plug-in Radio Board
On-board USB and Ethernet J-Link Debugger
Si7021 Humidity and Temperature Sensor
USB-serial-port Packet-trace Advanced Energy Monitoring
EXP-header for expansion boards
Battery or USB power
ARM Coresight 19-pin trace/debug header
Ultra-low power 128x128 pixel memory LCD, buttons and LEDs
Serial-port, packet trace and Advanced Energy Monitoring header
BAT USB AEM
Blue Gecko
SILICON LABS
RESET
PB1
PB0

2. Install Silicon Labs' Simplicity Studio Version 4 development environment. You can download the software from the following site:
   a. https://www.silabs.com/products/development-tools/software/simplicity-studio
   b. Sign in to Silicon labs, creating an account if necessary. This is required to pull in device libraries.
   c. **Ensure the device is connected to the computer when installing, this should automatically pull in device support:**



Install Wizard

Install Device Support

Select the devices you would like to use within Simplicity Studio.

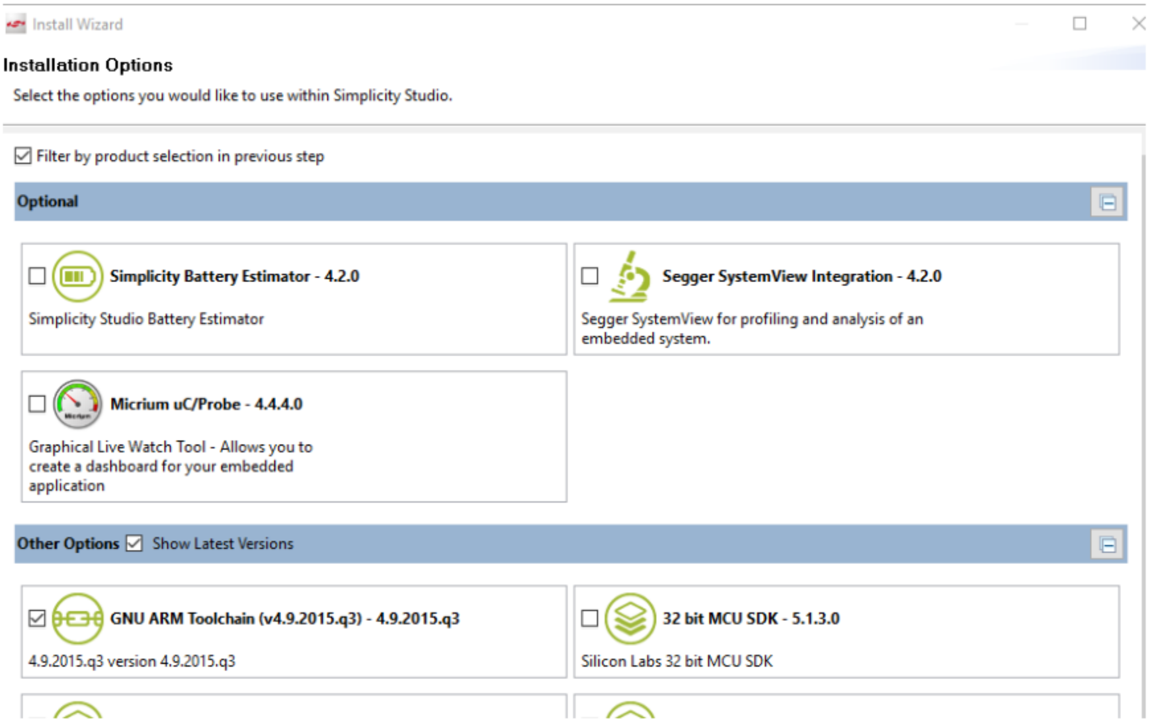
Connected devices :

EFR32 Blue Gecko Bluetooth Starter Kit (SLWSTK6020B)

Product Search: ☑ Kits ☑ Boards ☑ Parts

d. Ensure that you select all EFR32 files as a minimum

e. Within the installation options, install **GNU ARM v7.2.1** compilers, and **32 bit MCU SDK version 5.9.6.0**. (you may need to unclick "show latest versions")



Install Wizard

Installation Options

Select the options you would like to use within Simplicity Studio.

☑ Filter by product selection in previous step

Optional

☐ **Simplicity Battery Estimator - 4.2.0**
Simplicity Studio Battery Estimator

☐ **Segger SystemView Integration - 4.2.0**
Segger SystemView for profiling and analysis of an embedded system.

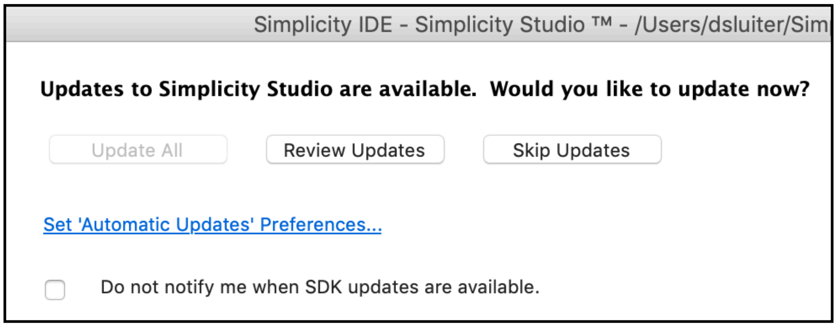
☐ **Micrium uC/Probe - 4.4.4.0**
Graphical Live Watch Tool - Allows you to create a dashboard for your embedded application

Other Options ☑ Show Latest Versions

☑ **GNU ARM Toolchain (v4.9.2015.q3) - 4.9.2015.q3**
4.9.2015.q3 version 4.9.2015.q3

☐ **32 bit MCU SDK - 5.1.3.0**
Silicon Labs 32 bit MCU SDK

IMPORTANT: Every time you run Simplicity Studio it asks you to Update.



Simplicity IDE - Simplicity Studio ™ - /Users/dsluiter/Sim

Updates to Simplicity Studio are available. Would you like to update now?

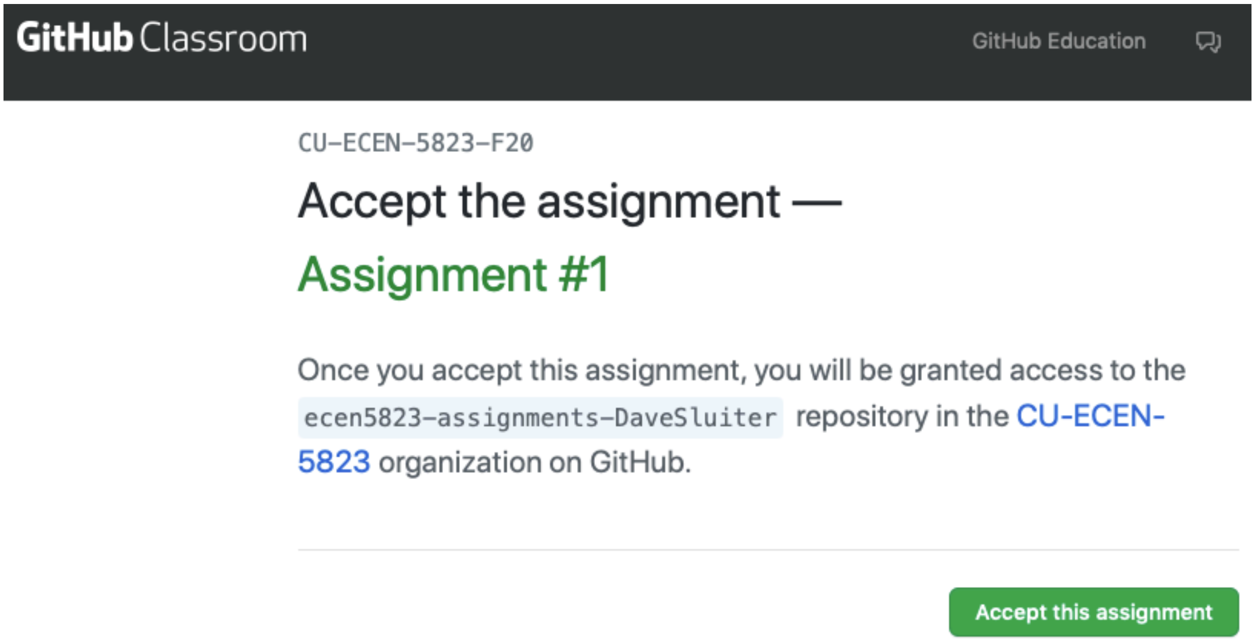
□ Do not notify me when SDK updates are available.

Always always always select "Skip Updates". This will keep all of us on the same version of Simplicity Studio, GCC and the SDKs. The versions we are using this semester are:

Simplicity Studio	4.1.13.6
GCC	7.2.1
Gecko SDK Suite	2.7.6
Bluetooth	2.13.5.0
MCU	5.9.6.0

3. We will use Github Classroom to create a unique repository for each student based on a common starter code repository. If you don't have a GitHub account, please create one (the free account type) prior to clicking the link below. You will then clone your newly created repository from the Cloud to your own machine, do your homework by modifying this repository, commit your changes, and push your changes to Github.
   a. Access the assignment starter code for the Github Classroom assignment at https://classroom.github.com/a/VXcFT7nJ

You should see a page like this:



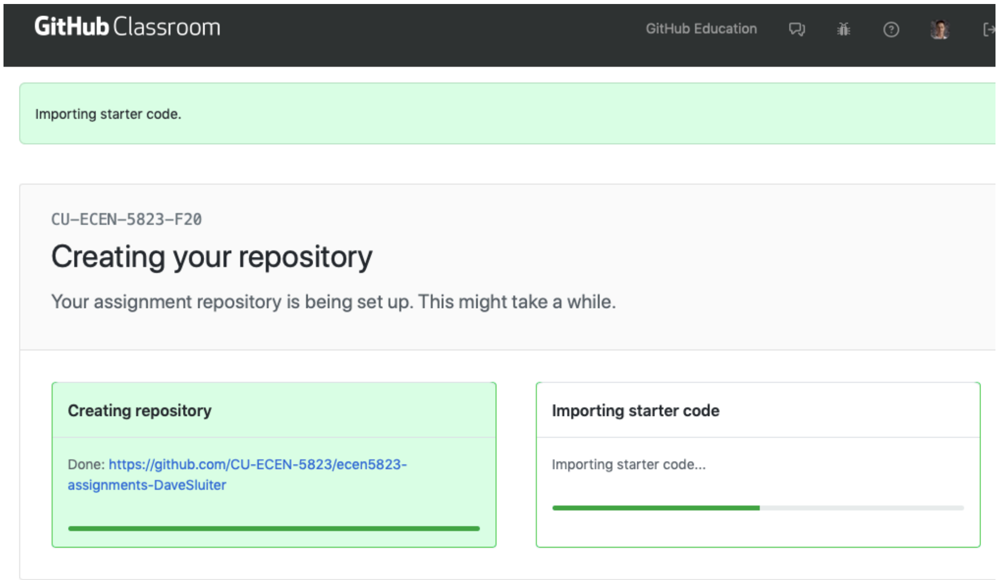
CU-ECEN-5823-F20

Accept the assignment —

Assignment #1

Once you accept this assignment, you will be granted access to the `ecen5823-assignments-DaveSluiter` repository in the CU-ECEN-5823 organization on GitHub.

click **Accept this assignment**. You should see:



Importing starter code.

CU-ECEN-5823-F20

Creating your repository

Your assignment repository is being set up. This might take a while.

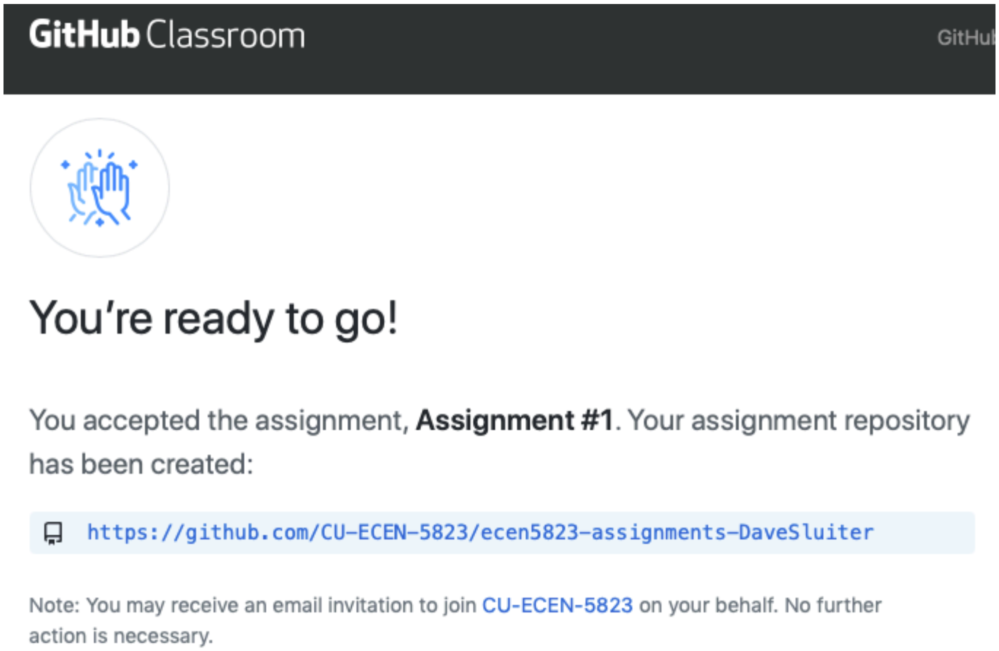
Creating repository

Done: https://github.com/CU-ECEN-5823/ecen5823-assignments-DaveSluiter

Importing starter code

Importing starter code...

and then



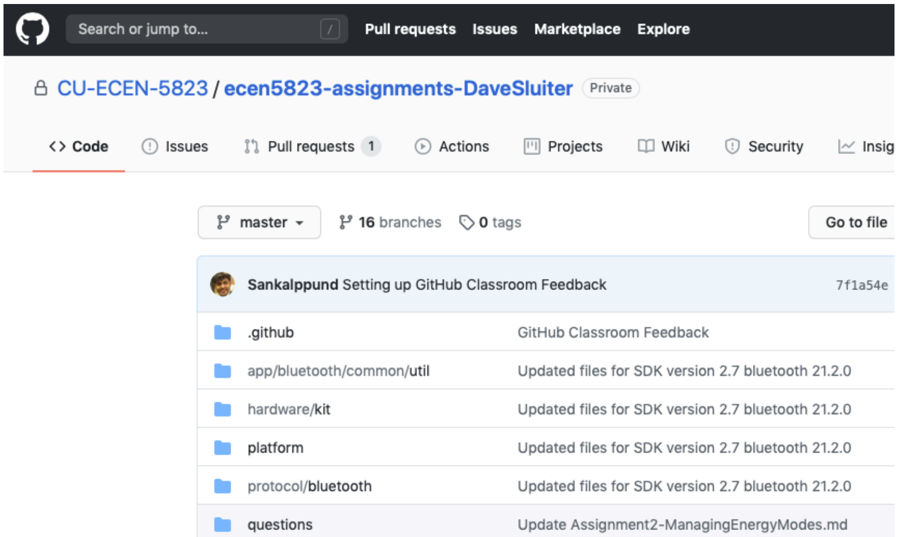
You're ready to go!

You accepted the assignment, **Assignment #1**. Your assignment repository has been created:

https://github.com/CU-ECEN-5823/ecen5823-assignments-DaveSluiter

Note: You may receive an email invitation to join CU-ECEN-5823 on your behalf. No further action is necessary.

Click the link to go to your newly created repository. You should see a new repository setup for you like this (this is just a screen capture of the top portion of the newly created repository):



CU-ECEN-5823 / ecen5823-assignments-DaveSluiter Private

Sankalppund Setting up GitHub Classroom Feedback 7f1a54e

.github	GitHub Classroom Feedback
app/bluetooth/common/util	Updated files for SDK version 2.7 bluetooth 21.2.0
hardware/kit	Updated files for SDK version 2.7 bluetooth 21.2.0
platform	Updated files for SDK version 2.7 bluetooth 21.2.0
protocol/bluetooth	Updated files for SDK version 2.7 bluetooth 21.2.0
questions	Update Assignment2-ManagingEnergyModes.md

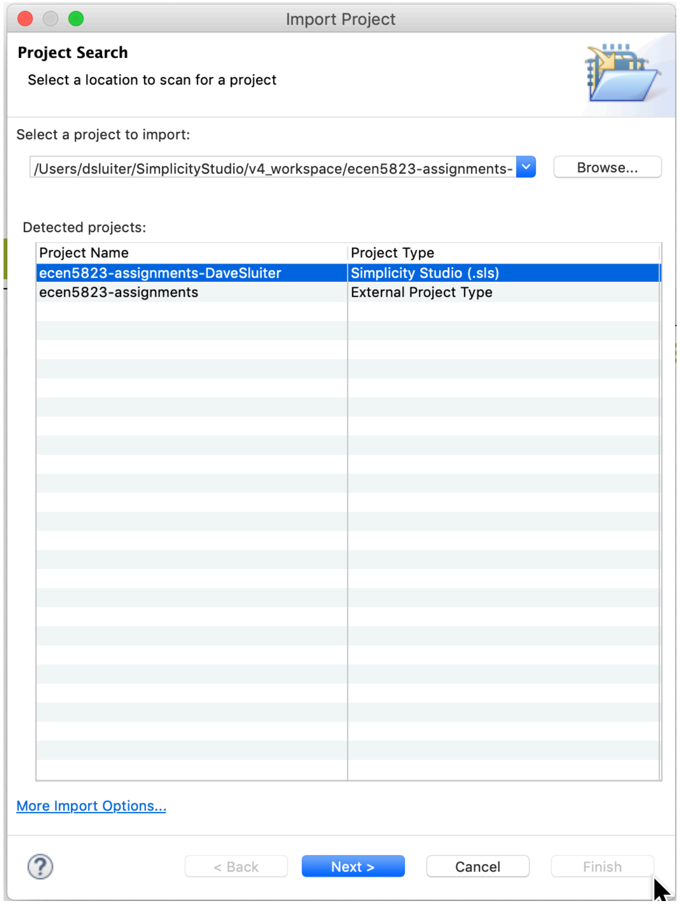
5. Clone into your simplicity studio workspace directory (~/SimplicityStudio/v4_workspace/ where ~ is your home directory or C:\Users\username on windows).
   a. Use the simplicity studio workspace directory since when located outside this directory some features **such as project rename** are not supported in the IDE.
   b. See https://help.github.com/articles/cloning-a-repository/ for clone instructions from github.

Linux example:

```
% cd ~/SimplicityStudio/v4_workspace
% git clone https://github.com/CU-ECEN-5823/ecen5823-assignments-DaveSluiter.git
```

DaveSluiter is my GitHub username. A **new directory** will be created in ~/SimplicityStudio/v4_workspace/ called ecen5823-assignments-DaveSluiter/

6. In Simplicity Studio, select File->Import and browse to the ecen5823-assignments-<username> directory that was created in the previous step. You should see a screen similar to this:



Project Search

Select a location to scan for a project

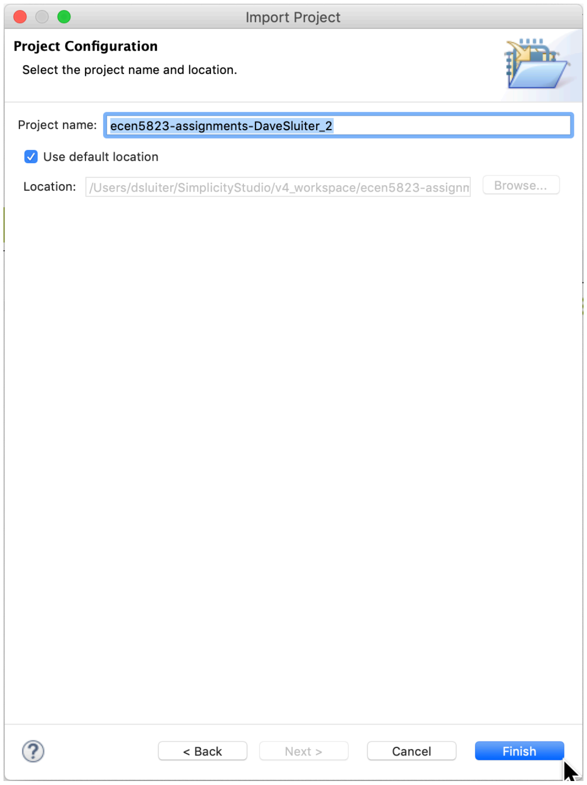
Select a project to import:

/Users/dsluiter/SimplicityStudio/v4_workspace/ecen5823-assignments-

Detected projects:

Project Name	Project Type
ecen5823-assignments-DaveSluiter	Simplicity Studio (.sls)
ecen5823-assignments	External Project Type

Select the “Simplicity Studio.sls” project type and click “Next” through the next two screens, ensuring that the project name is “ecen5823-assignments-<username>”. **Here you see it added an “_2”, remove the “_2” before continuing**.



Import Project

Project Configuration

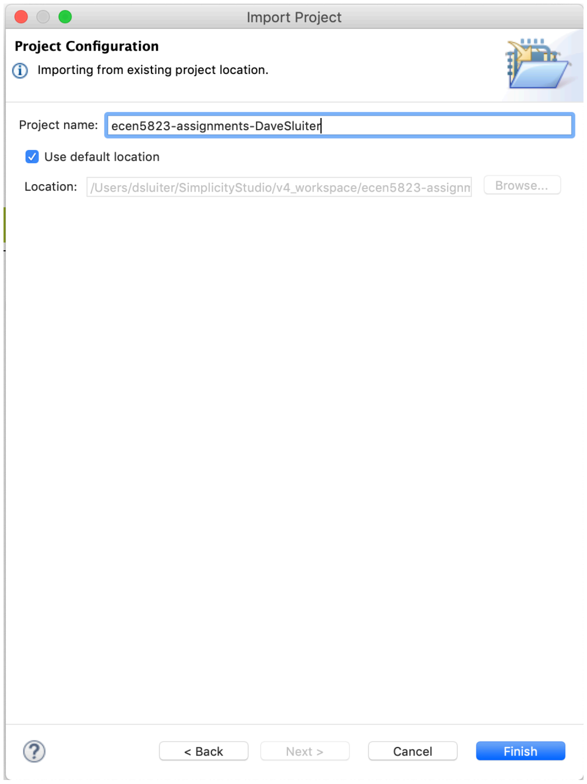
Select the project name and location.

Project name: ecen5823-assignments-DaveSluiter_2

☑ Use default location

Location: /Users/dsluiter/SimplicityStudio/v4_workspace/ecen5823-assignm

_2 removed:



Import Project

Project Configuration

Importing from existing project location.

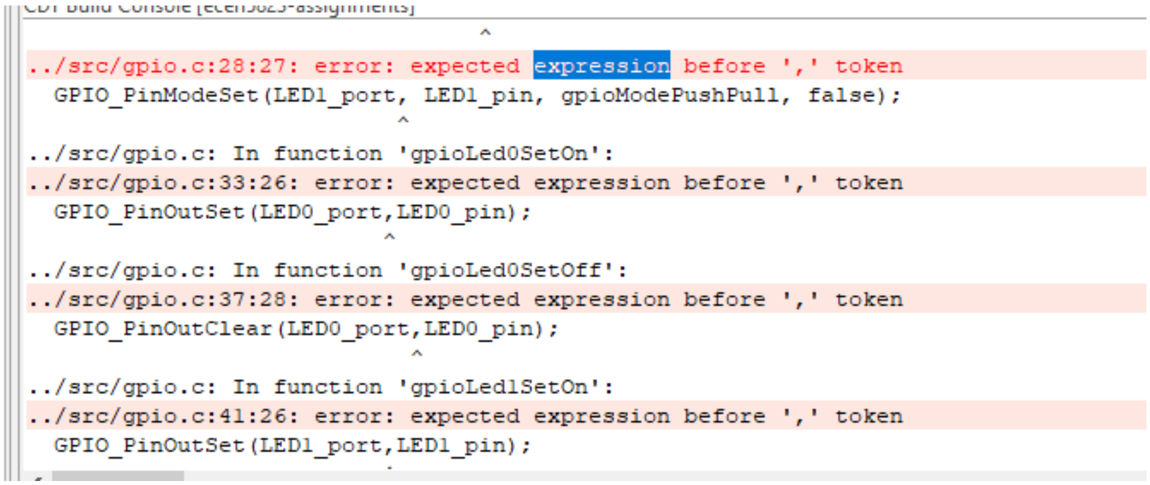
Project name: ecen5823-assignments-DaveSluiter

☑ Use default location

Location: /Users/dsluiter/SimplicityStudio/v4_workspace/ecen5823-assignm

7. Ensure the project name is ecen5823-assignments-<username> and click “**Finish**” to complete. Now, the demo / example project should be loaded into your workspace. To reduce the number and types of issues you may see, it is a good practice to perform a **Clean…** operation on your project immediately after importing and prior to attempting to build (compile) the design. Right-click on the project in the Project Explorer and select “**Clean Project”**.

8. Right click on the project in the Project Explorer pane and select “**Build Project**”. You may see errors like these (since we haven’t defined GPIO pins yet).



```
CDT Build Console [ecen5823-assignments]
                                 ^
../src/gpio.c:28:27: error: expected expression before ',' token
  GPIO_PinModeSet(LED1_port, LED1_pin, gpioModePushPull, false);
                           ^
../src/gpio.c: In function 'gpioLed0SetOn':
../src/gpio.c:33:26: error: expected expression before ',' token
  GPIO_PinOutSet(LED0_port,LED0_pin);
                          ^
../src/gpio.c: In function 'gpioLed0SetOff':
../src/gpio.c:37:28: error: expected expression before ',' token
  GPIO_PinOutClear(LED0_port,LED0_pin);
                            ^
../src/gpio.c: In function 'gpioLed1SetOn':
../src/gpio.c:41:26: error: expected expression before ',' token
  GPIO_PinOutSet(LED1_port,LED1_pin);
```

However, you may not see these errors, but may have a run-time error with regard to the gpio. To resolve this, add the function call gpioInit() after gecko_init() in main.c.

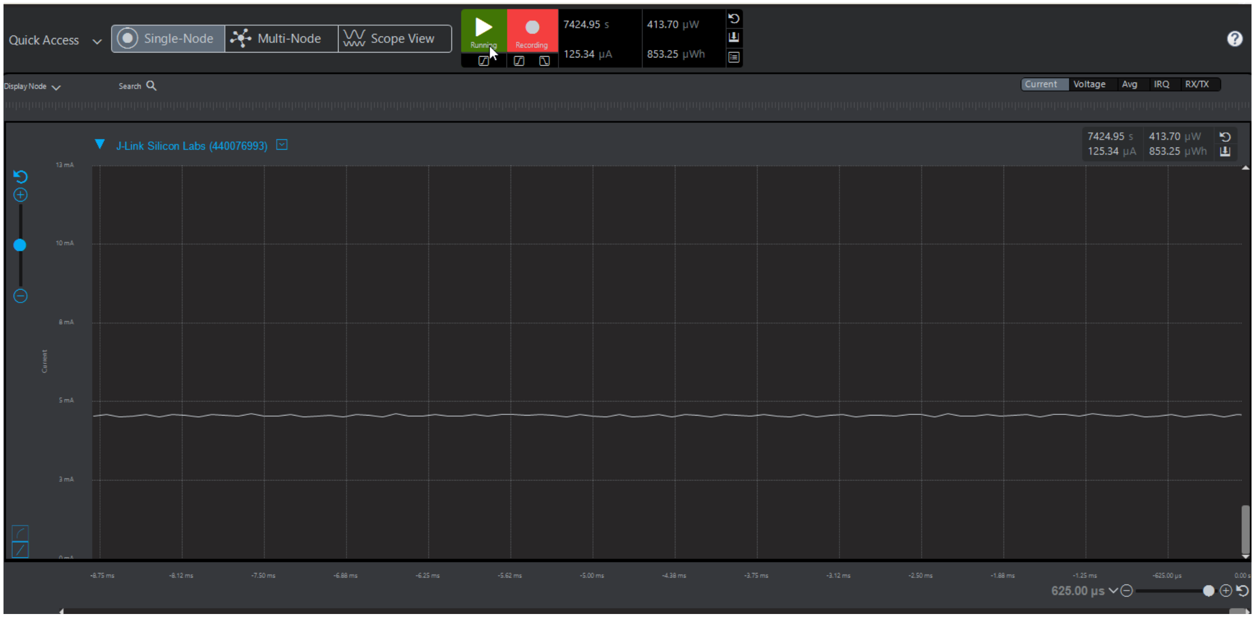
9. Expand the project and then open up the /src folder. Open up the gpio.c file and complete the following #define statements by tracing the trace from the LEDs to the Blue Gecko. See the documentation in the gpio.c file comments for more information.

   a. #define LED0_port -1 //change to correct port and pin
   b. #define LED0_pin -1
   c. #define LED1_port -1
   d. #define LED1_pin -1

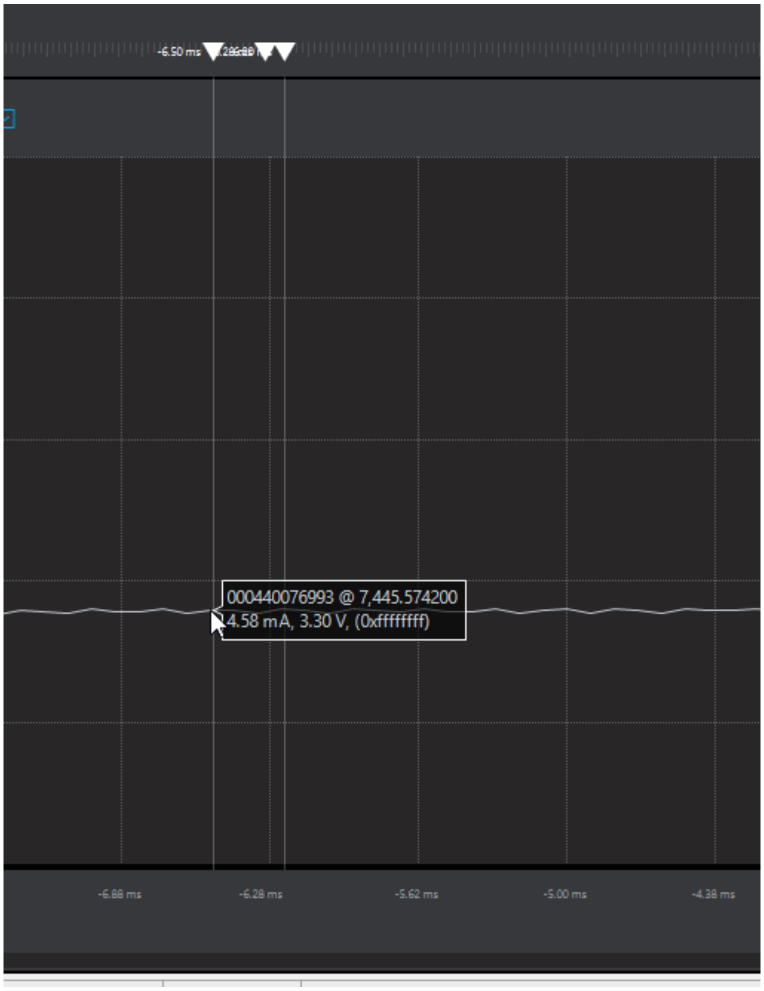
10. Now, build the project again to make sure there are no compilation issues, then under the Run pull down menu, select “Profile As / Simplicity Energy Profiler Target”

    a. Simplicity IDE should begin to compile the project, and then download the code onto Blue Gecko board
    b. After downloading the code to the board, the code should begin to run, and LED1 on the Blue gecko should begin to flash
    c. Simplicity should now open the Energy Profiler

11. In the Energy Profiler, click the “Running” or “Paused” button in the center to switch between running/paused modes. You may have to use the Zoom control at the left to increase the vertical display height of the waveform.



12. Click a point with the mouse to display instantaneous current measurements



000440076993 @ 7,445.574200
4.58 mA, 3.30 V, (0xffffffff)

13. Pausing the Energy Profiler, use the instantaneous current measurement to determine how much current a single LED draws.

14. In gpio.c, the lines of function gpioInit() should look like:

```
GPIO_DriveStrengthSet(LED0_port, gpioDriveStrengthStrongAlternateStrong);
//GPIO_DriveStrengthSet(LED0_port, gpioDriveStrengthWeakAlternateWeak);
GPIO_PinModeSet(LED0_port, LED0_pin, gpioModePushPull, false);

GPIO_DriveStrengthSet(LED1_port, gpioDriveStrengthStrongAlternateStrong);
//GPIO_DriveStrengthSet(LED1_port, gpioDriveStrengthWeakAlternateWeak);
GPIO_PinModeSet(LED1_port, LED1_pin, gpioModePushPull, false);
```

If not, edit the lines of code to match this. We want the “StrongStrong” setting for both LED0 and LED1 for the next step.

15. Run the Energy Profiler, pause it, then use the instantaneous current measurement to determine how much current a single LED draws (LED1) with the “StrongStrong” case.

16. Edit the code (comment/un-comment) to profile the “WeakWeak” case for both LED0 and LED1, and measure the current when the LEDs are on.

Questions:

Answer the questions in the folder questions/Assignment1-SimplicityExerciseQuestions.md within your repository and commit with your final code.

Deliverables:

1. Answers to the 5 assignment questions in the questions/Assignment1-SimplicityExerciseQuestions.md file within your repository
2. Modified program files with all modifications made (including correct GPIO assignments).

How to submit your assignment:

All files should be submitted using Github Classroom and the link at https://classroom.github.com/a/VXcFT7nJ

- Verify your ecen5823-assignments-<username> repository contains your latest code and question responses before the deadline.
- In **Canvas**, submit the URL to your github repository to complete your assignment submission.

Be sure to push your changes back to your repo before the deadline!