
# School of Electrical Engineering and Computer Science

# CptS466: Embedded Systems

# Fall 2021

# Project 1 (P1)

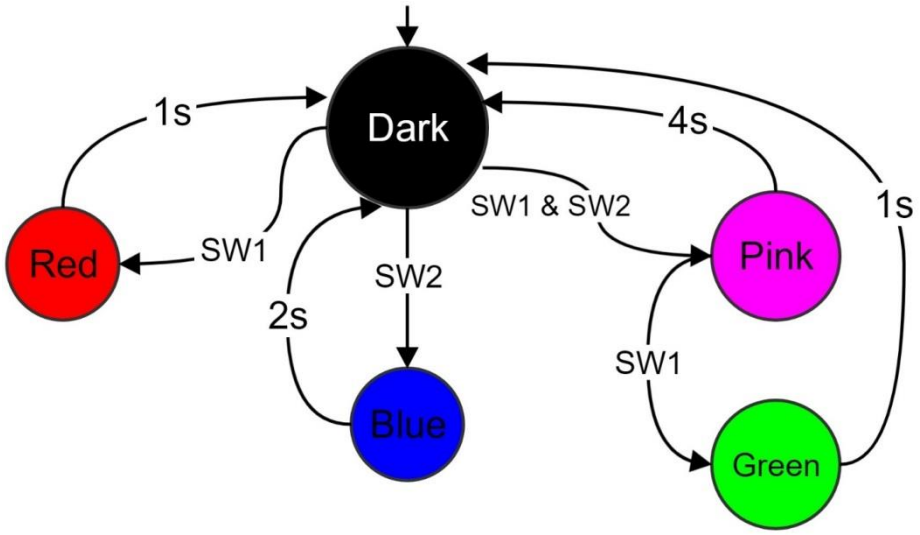
# Code & Report Due: 9/13/2021 @ 11:59pm (Canvas)

# Demo Due: 9/13/2021 @ 11:10am (Classroom)

## 1. Project Description

In this project you will use Keil and Cortex-M C libraries to implement a simple sate machine. Each state in your state machine is associated with a unique color on the LED that exists on the Launchpad. Also, you will use the on-board switches to allow for transition from one state to another. Thus, you will use the on-board switches and multicolor LED of your Launchpad for I/O purposes. You may use the provided demo code to get started on this project (although you can write your code from scratch). A description of how to program your board with the demo code is provided in the Appendix section of this document. The figure below illustrates the state transition diagram which provides an overview of the required events and states.

- Initially, the system goes to 'Dark' state (LED off or dark color).
- While in 'Dark' state, the multicolor LED turns red when you push SW1 and it turns dark (goes back to 'Dark' state) after 1 second delay.
- While in 'Dark' state, the multicolor LED turns blue when you push SW2 and it turns dark after 2 seconds delay.
- The multicolor LED turns pink when you push both SW1 and SW2 together and it turns dark after 4 seconds.
- While in state 'Blue' or state 'Red', pressing a button will not result in transitioning to another state.
- While in state 'Pink', the user may press SW1 to transition to state 'Green'. Note that pressing SW2 should not result in transitioning to 'Green' state.




## 2. What to submit?

Submit a .zip file with the following content:

- Your well commented C source code implementing the application described above.
- Your report about how you developed the code and your observations

In your report discuss:

1. Goals of the project in one or two sentences.
2. How you prevented transition from one state to another due to pressing an unexpected button. For example, how did you make sure that pressing SW2 while in state Red does not result in transition to Blue state.
3. What happens if you press SW1 while in Red state? Would you wait for another 1 second or will you ignore SW1? Discuss the behavior of your program.
4. Similarly, what happens if you press SW2 while in Blue state? Would you wait for another 2 seconds before going to Dark state or will you ignore SW2?

Show a demo of your application in the class on the due date.

## 3. Grading

Assume that the whole assignment is worth 100 points.

- 50 pts for well commented and correct C source code satisfying the project requirements.
- 25 pts for report.
- 25 pts for demo.

## 4. Appendix

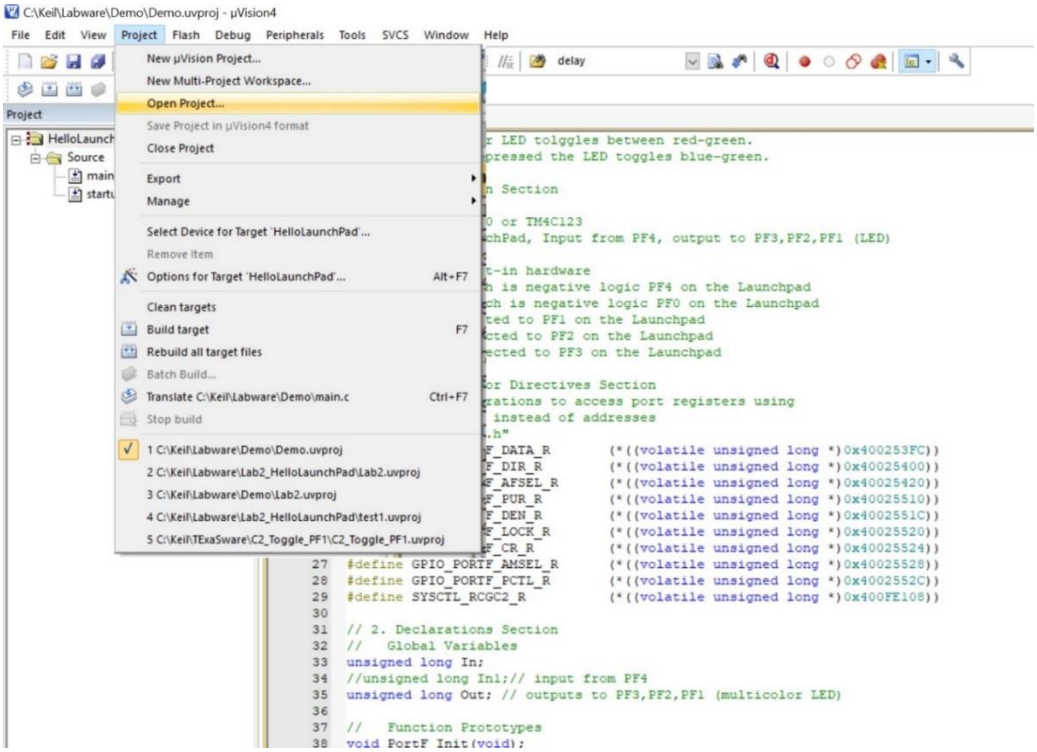
In this section, we discuss how you can program the Launchpad with the provided demo code. You can find the demo software under ‘Assignments’ section in Canvas.

Let’s look at few pages from the TM4C123 microcontroller datasheet. You can download the datasheet for the TM4C123 microcontroller following the link provided under ‘Assignments’ section of the course web page.

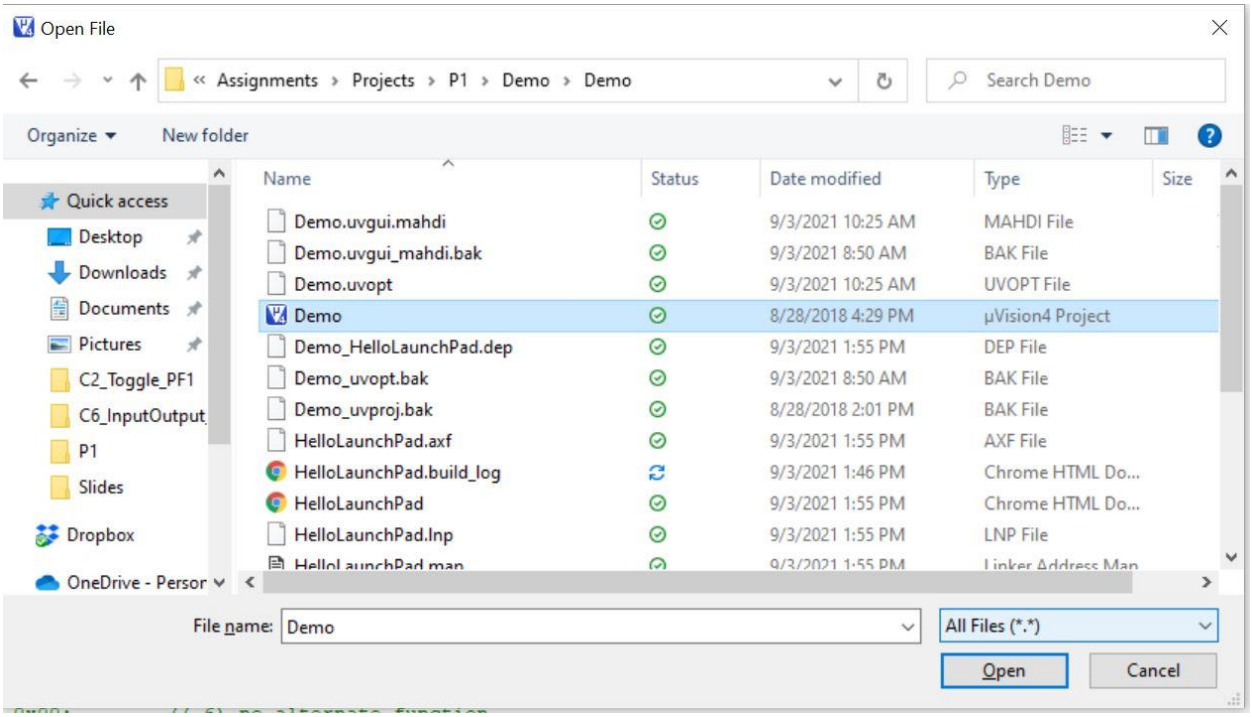
- Look at the block diagram on page 46 to see the amounts of RAM and ROM memories that this microcontroller has.
- Look at page 649 to see how many I/O pins exist.

Follow these steps to program your Launchpad.

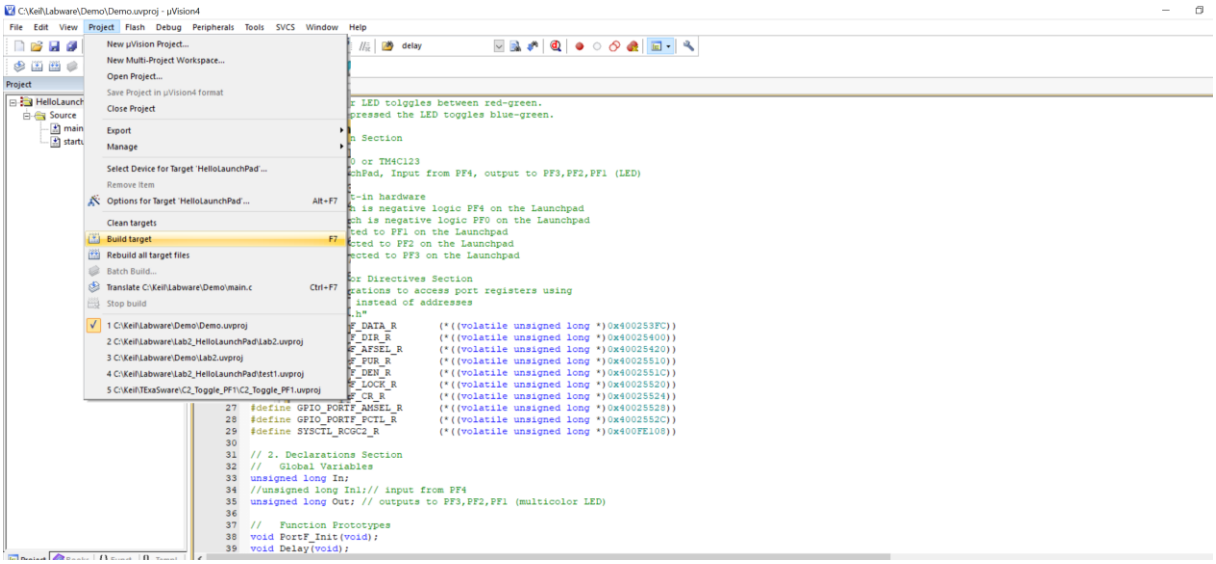
1. Open the software and go to “Project” tab and “Open Project”



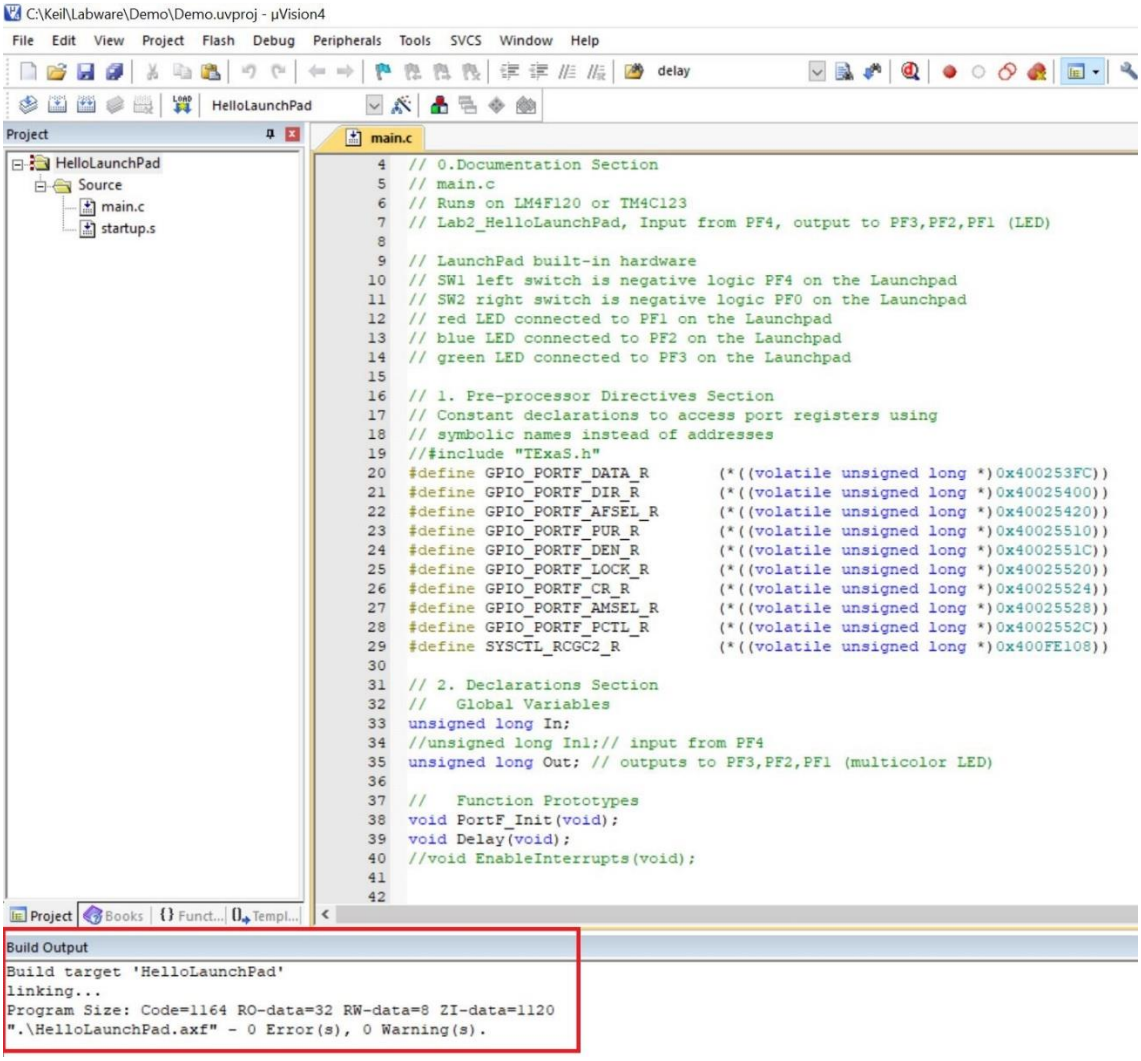
2. Go to where you downloaded the demo and open the project.



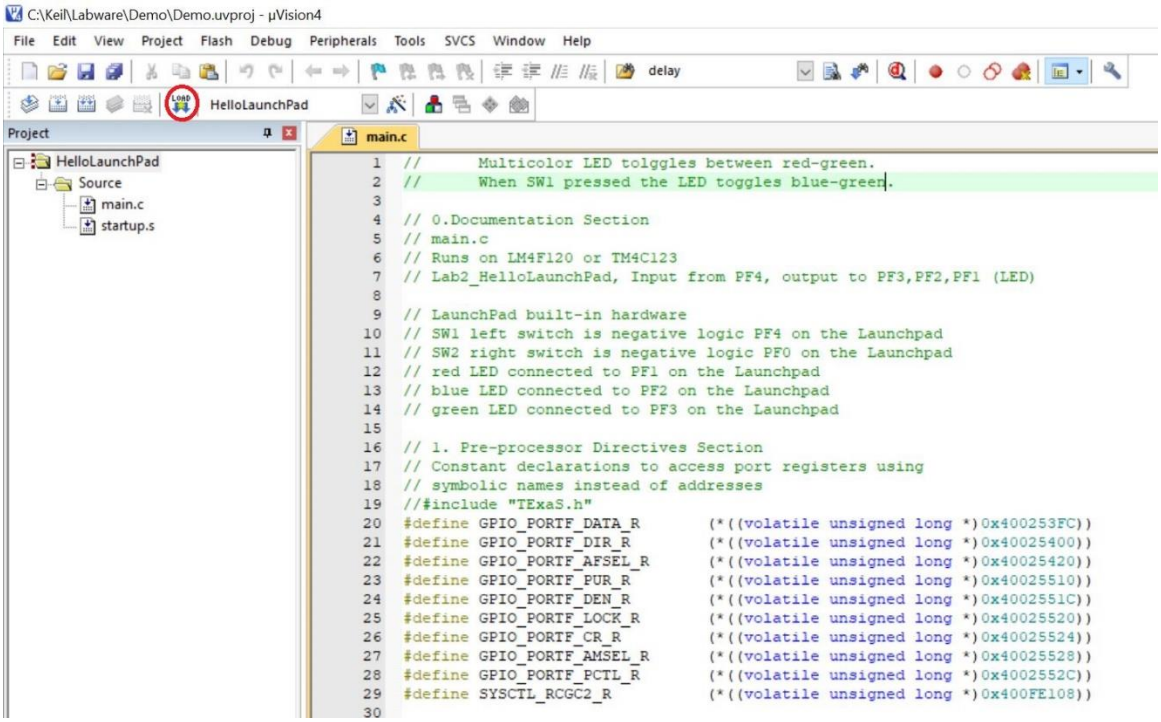

3. Go to Project tab and Build Target.



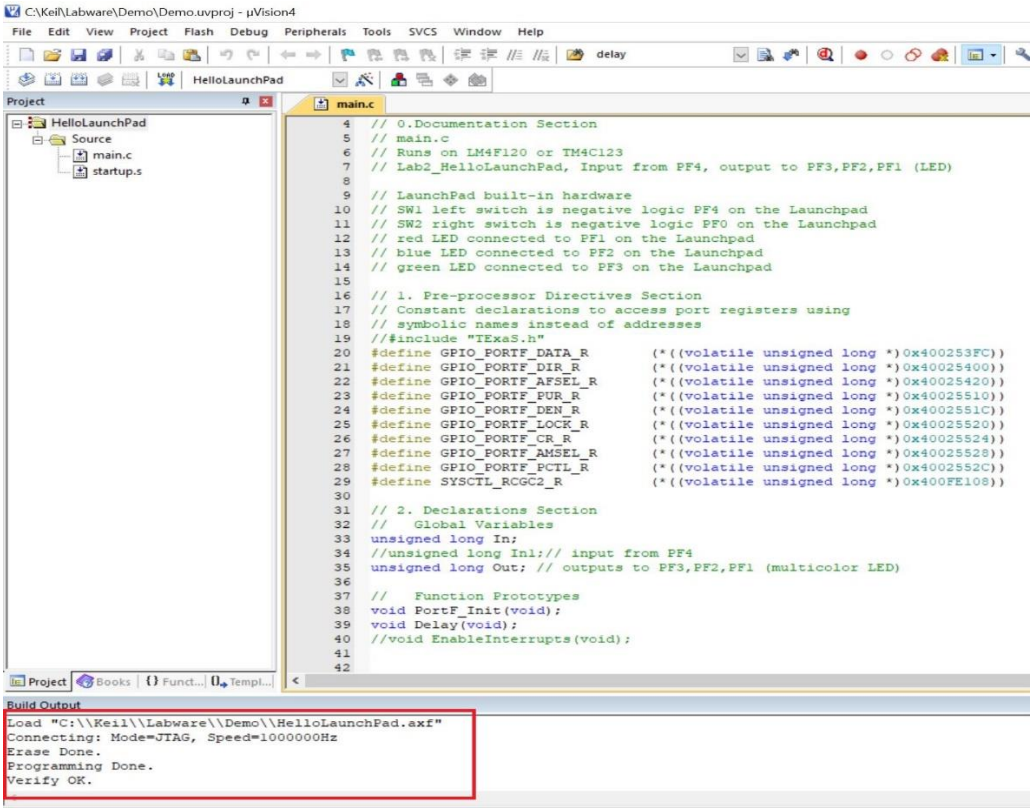
4. In Build Output section you would observe that the target has been built without any error and warning.



5. Click on the download button to load the program into your launchpad.




6. In Build Output section you will observe that the programming is done successfully.



7. Press RESET button on your launchpad then your program is running on the board!