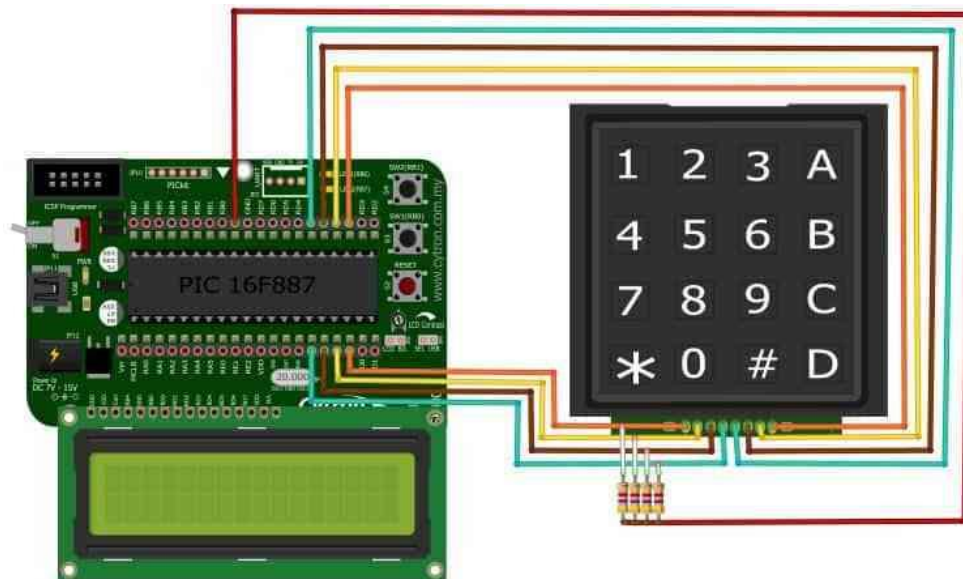


Microcontrollers Lab

Week 7 – LCD, Keyboard and timer integration

April 27th – Due May 6th 2020

In this lab the idea is to integrate aspects of the three modules we have seen so far: LCD display, 4x4 matrix keyboard and timers. Thus, before proceeding, be sure that your code for the LCD and the keyboard works properly. You can use either the 8- or 4-bit option for the LCD code.



The lab will be divided into two parts, described as follows:

Part 1. Menu and output management. Write a very simple program that displays the following message in the LCD screen

PRESS BUTTON
R: 1 B: 2 G: 3

Then write a function that recovers the data from the `get_key()` function (plus a decoder) and switch on a led depending on which key was pressed (1: Red, 2: Blue, 3: Green) and display

RED/BLEUE/GREEN

LED IS ON!

The led should remain on for a few seconds, then go off and display the initial menu again.

Part 2. Ascending Timer: The goal of the program is to implement a simple timer, as the ones used in sports or music. The code should proceed as follows:

1. It should print a hello and stay there for 5 seconds (you can use simple delay for this), then a second message should appear in the first line asking for the user to introduce the number of seconds.
2. To make this interesting, the code should be able to accept at least two digits, so you need to introduce the `get_key`(function) into a while loop that only stops when you press another not numerical key (for instance the * or # key.)
3. The introduced value should set the `TMP_MODULO` register in one timer and start it, either immediately after the termination key was selected or by pressing the same key again.
4. Then the LCD should show a message Counting and showing the current count value.
5. Once the timer has elapsed, you can use a buzzer (if you have it) or a led to mark that the timer went to zero.

Requirements for the report

1. Include the code for each of the functions of your code. The code should be commented and the report should include a short description of each function and how it works, including images of the registers that have been configured for enabling the different functionalities in your code.
2. Provide a state machine or a flow diagram for the entire code.
3. Make some research in how to connect the board to the external components. You can use a virtual program to schematize the connection between the board and the external components (LCD screen and keyboard, external components). An example is Fritzing, but there are others
4. Attach a short video demonstrating the system working. Alternatively, you can share the link to the video in your google drive for me to evaluate it.