Birzeit University - Faculty of Engineering and Technology
Electrical & Computer Engineering Department - ENCS4330
Real-Time Applications & Embedded Systems - 1^{st} semester - 2021/22

Project #4
16F877A PICMicro programming under MPLAB
Due: December 22, 2021

Instructor: Dr. Hanna Bullata

Fun binary counter

We would like to build a fun binary counter that is able to count up from 0 to 31 on a 16F877A PIC microcontroller but not necessarily in a sequential order. The hardware should be composed of 5 LEDs to show the count and 2 push buttons. We'll call the 2 push buttons P_1 and P_2 consecutively. The system should behave as follows:

- When powered up, the default behavior for the counter is to count up sequentially from 0 to 31 with a pause of 1 second between the counts.
- If push button P_1 is clicked, the counter should continue counting from its current value up to 31 but should increment by 2 instead of 1.
- If push button P_1 is clicked again, the counter should continue counting from its current value up to 31 but should increment by 3. Further clickings on P_1 should increment the counter by 4, then by 5 and then back to increment by 1.
- If push button P_2 is clicked during counting, the counter should increase the pause value between increments: The first time P_2 is clicked, the pause between counts should be made 2 seconds. Further clickings on P_2 should make the pause between counts 3 seconds, 4 seconds and 5 seconds consecutively. Clicking on P_2 after the 5-second pause should bring it back to 1 second.

What you should do

- Build the controller described above on a bread board. Remember to add to each LED a 220Ω series resistor to limit the current. Remember as well to add a 10KΩ pull-up resistor to each push button (or use the internal weak pull-up resistors), add a 4MHZ oscillator (with 2 × 22pF capacitors) and a 4.7KΩ pull-up resistor to the MCLR pin.
- Build the PIC assembly code that implements the behavior described above under MPLAB IDE.
- Assemble your code and make sure you get a successful build. Use the simulator if you wish to make sure the behavior is correct.
- Send the zipped folder that contains your source before the deadline. If the deadline is reached and you are still having problems with your code, just send it as is!