**ET420 Microcontroller Applications**

Matt Alanskas

**Assignment #2**

**25 points**

**Another Dice simulation:** Create a C application using a 16F1823 to provide a dice simulation. The clock should run at 1MHz – to make sure you can’t cheat. The dice should be “rolled” by a push button – which triggers an interrupt to assign values for the two dice. This should be done by reading the clock, take the clock reading mod 6 – add 1 – that’s your first die value. Shift the original clock reading to the right two places (divide by 4) and take mod 6 of that value and add 1 – that’s your second die value.

Display the two values on one 7 segment display – alternate every two seconds. Do this by using timer1 driven by the clock – use an overflow interrupt to trigger the switch between values.

Display should start with all seven segs illuminated to ensure everything is working.

You will need to turn in this cover sheet, your design, your properly commented code, a CAD-drawn schematic diagram, test plan, and test data. I will check your functioning systems off in lab. Your design should consist of a paragraph describing your approach to the problem, and a flowchart or pseudocode. Your test plan should consist of how you will test your program, the sample data used, and the results obtained.

|  |  |  |
| --- | --- | --- |
| Functionality: |  | /10 |
| Documentation: |  | /10 |
| Code Analysis: |  | /5 |

**Design Approach:**

For the hardware side of the project, I used the exact same setup as assignment one, so I kept everything wired the same as assignment one. Approaching the software side of the project, I plan on using the clock, an interrupt function, and timer function to create a dice simulation that will be displayed on one seven segment display that switches back and forth between the two “rolled” die values.

**Pseudo Code:**

Setup libraries, arrays, and pin setups

Start main function

Set ports according to pins

Set clock to 1MHz

Enable interrupt on change for RA2 negative edge

Enable interrupt for timer 1

Enable global interrupt

Start while loop

Call seg display function with die values

Delay

End while loop

End main function

Start seg display function

Write arrays to set up PORTA and PORTC for blank, 1-6, and 8.

End seg display function

Start interrupt function

Start first if statement

Turn off timer 1 to set values

Set first die to timer 1 mod 6

Add 1 to first die value

Shift the reading over two positions right

Set second die is timer 1 mod 6 shifted two positions right

Add 1 to second die value

Turn timer 1 back on

Reset the timer flag

End first if statement

Start second if statement

Turn off timer 1 to set values

Set timer one to 34,286

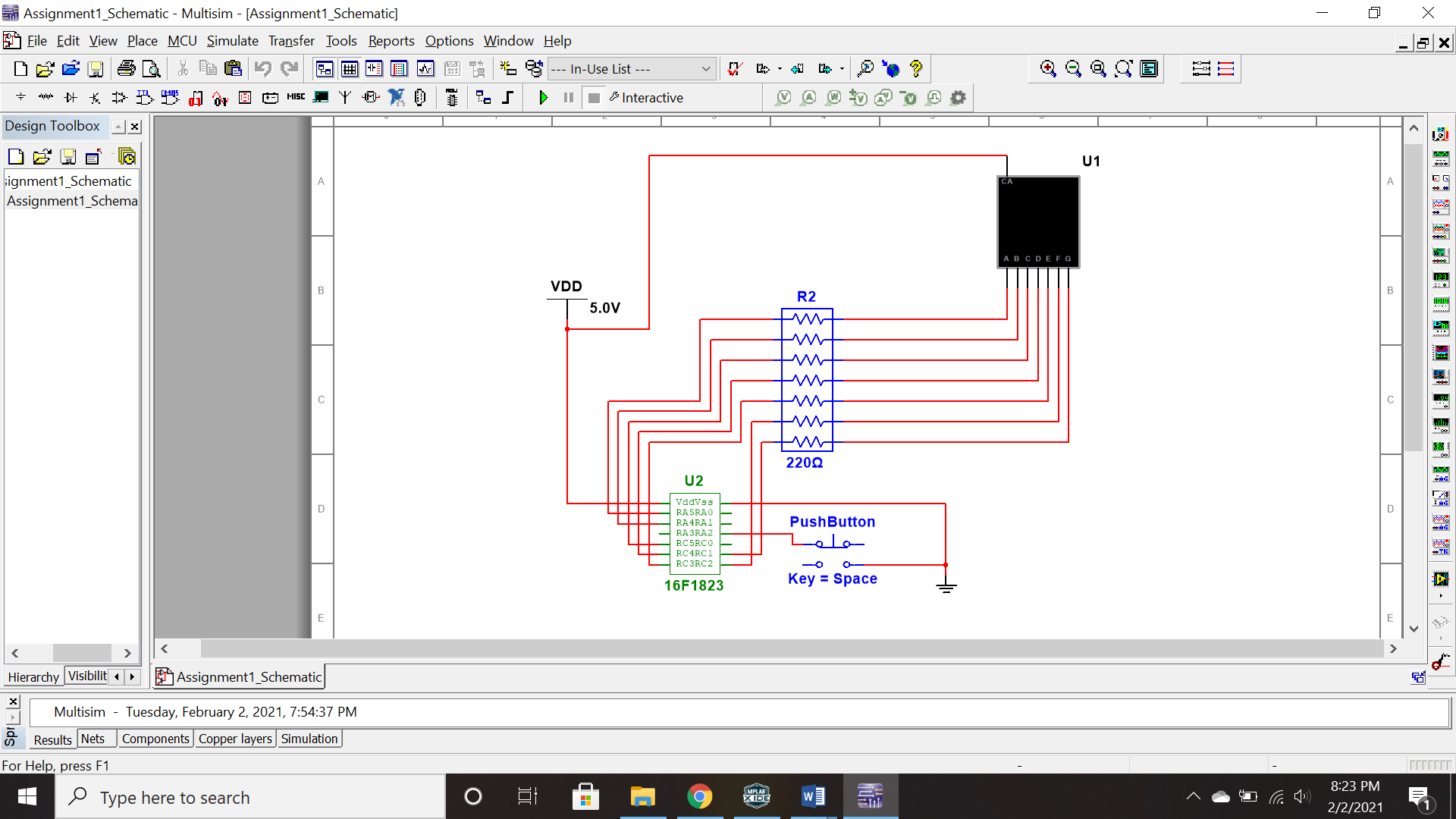
Turn timer 1 back on

Resetting the timer flag

End second if statement

End interrupt function

**Schematic:**



**Test Plan:**

I was unable to get my program to switch my die back and forth between the two different die values, but it was outputting random numbers every time the push button was engaged, so I ran a series of ten tests to ensure that random numbers were being produced.

|  |  |  |
| --- | --- | --- |
| Test Number | Die One | Die Two |
| 1 | 4 | 3 |
| 2 | 2 | 2 |
| 3 | 5 | 3 |
| 4 | 4 | 1 |
| 5 | 3 | 1 |
| 6 | 5 | 3 |
| 7 | 1 | 4 |
| 8 | 3 | 2 |
| 9 | 1 | 5 |
| 10 | 2 | 2 |