

EE333 Microcontroller Engineering

Oregon Tech Portland, Fall 2013

Lab Assignment #6 - *Touch Switch*
Due December 5

Objectives:

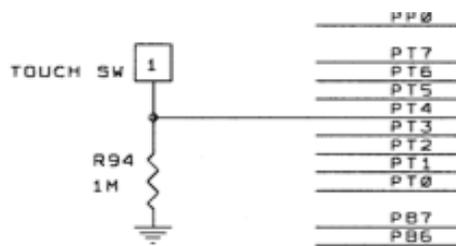
Show the use of the timer module to measure the relative capacitance of the touch switch.

Equipment Required:

Start with your working lab assignment 5.

Background Information:

The Dragon12 board has an area marked as a "Touch Switch". If you look on the schematic you find the very mysterious:



The switch is just a square of insulated copper connecting to Port T pin 4 and a 1 megohm resistor to ground. Its operation relies on the increase in capacitance on pin PT4 to ground when a finger touches the insulation over the copper square.

To measure the capacitance, you need to first drive PT4 to 5 volts and then open the circuit by switching the pin direction to input. The capacitance will then discharge through the 1 megohm resistor. If you measure the time from when the circuit is opened until the level on PT4 reaches the low state, the time will be proportional to the capacitance.

Assignment:

Light the leftmost LED, PB7 while the touch switch is pressed. You will first need to insure that the existing code which drives the row of LEDs (from the push buttons) will not affect PB7, since you will be controlling PB7 from a different location in the code.

Add two word variables, one for the start time of the capacitance measurement, I call mine *timest*, and a second for the elapsed time of the capacitance measurement. I call mine *timets*. For initialization, you will need to set up the interrupt service routine vector for timer channel 4 (much like you did in lab assignment 5), enable the timer module, and configure channel 4 to interrupt on the

falling edge. You also need to configure port T pin 4 as an output pin and drive it high.

Start a new measurement every 1.024 milliseconds at the end of the RTI interrupt routine. Switch the pin direction to an input and immediately save the value of TCNT in the *timest* variable. In the newly added timer channel 4 interrupt service routine you need to get the time of the falling edge (it will be in TC4, not TCNT), subtract *timest* and store the elapsed time in *timets*. Clear the interrupt flag and drive the pin high again for the next measurement.

Somewhere in the code you need to read the value in *timets* and if it above some threshold turn PB7 on (use `bset displ #80`) and if it is below the threshold turn PB7 off. You can try \$200 as the threshold value. It worked well for me.

To turn in:

- Documented program listing.
- Description of how you tested the program.
- Discussion of any problems you had

This should all be placed in a single file (PDF format preferred, Word or Open Office formats also acceptable).