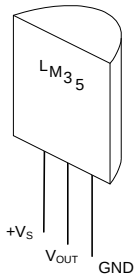
Assignment #2 – Temperature Monitor

In this assignment you will use the DigitalIn, AnalogIn, and BusOut interfaces on the mbed microcontroller board to implement a simple temperature monitor.



The pinout of the LM35 temperature sensor is shown to the right. Connect the LM35's +Vs pin to the mbed's VOUT pin, the LM35's Vout pin to the mbed's p20 pin, and the LM35's GND pin the mbed's GND pin. *Do not confuse the LM35's Vout pin with the mbed's VOUT pin.* The voltage on the LM35's Vout pin is proportional to the temperature and is approximately 10 mV per degree Celsius (thus at 25 C, the LM35 outputs 0.25 V).

Connect one of the pushbutton switches between p28 and VOUT and the other switch between p27 and VOUT (this is the same as in assignment #1). Connect the 7-segment LED display to the mbed as described in section 3.5.4 of the textbook.

Write a program for the mbed that will repetively read the temperature from the LM35 and display the temperature on the 7-segment LED display in either Celsius or Fahrenheit, depending on the currently selected mode. Your program should use the mean average the voltage from the LM35 (computed over at least 1000 samples) so that the displayed temperature does not fluctuate when the temperature of the LM35 is stable.

Since the display can only display one digit, you must display the temperature one digit at a time. Display each digit for 0.5 – 1 seconds, then blank the display for 0.1 seconds (this way the user can distinguish a digit shown once from two digits with the same value shown consecutively). After showing the temperature, the display should show "C" or "F" (depending on the currently selected mode) for 0.5 – 1 seconds and then blank the display for 0.1 seconds.

Pressing the switch connected to p28 selects displaying the temperature in Fahrenheit. Pressing the switch connected to p27 selects displaying the temperature in Celsius. If neither switch is pressed, the temperature should be displayed in whatever was the last selected mode. When your program starts, it may default to whichever mode you prefer. Your program should check the state of the switches at least once per temperature displayed (so the user might have to hold down the switch for an entire display cycle to switch modes). If you want to, you can detect mode changes more often, but you should completely display the temperature in one mode or the other (for example, don't display the first digit from the Celsius temperature and the second digit from the Fahrenheit temperature).

Hints: For debugging purposes, it might be useful to use the Serial object to display intermediate results on your computer, although this is not a requirement for the assignment. Sections 6.2 and 6.4 of the textbook may also be useful.

Submit your "main.cpp" to the appropriate dropbox on http://learn.ou.edu by 11:59 PM of September 24th. (In the Program Workspace view of the mbed compiler, right-click on the file to save and select Export; this will save the file to your computer. You can then upload this file from your computer to the dropbox.)