FANSHAWE
School of Applied
Science & Technology

Lab 4

Modify the existing program form Lab 3. This exercise will create a local control interface for the alarm and limits on each of the sensor channels previously developed.

Hardware push buttons will allow for control of selectable channel's limits. Linear equation will be applied to the ADC average samples for comparison in real world terms.

Operational Characteristics:

Hardware

1. Update and modify Lab 3's schematic. Add four (4) pushbuttons and three (3) LEDs to the system circuit, all connected in an active low configuration. Ensure you do not consume the PIC18F45K22's serial port pins when selecting hardware pins to connect.

2. Add the four (4) push button circuits to the breadboard of the existing system developed up to Lab 3.

   Label each of the buttons as follows:

PB1 -	Increment
PB2 -	Decrement
PB3 -	Limit Mode
PB4 -	Channel / Select

3. Add three (3) LEDs to your system. Each of these LEDs will represent an alarm state of a sensor channel. With three sensor channels, three LEDs are required.

   Label each of the LEDs on your breadboard setup. If the schematic does not match the physical breadboard, it will be considered an academic offence.

Software

Create a new folder directory and an MPLab project using the C18 tool suite. Name the project ELNC6007(your initials)Lab4, without the brackets.

Do not continue without saving a copy of the complete Lab 3 source file. Maintain completed versions for each exercise and continue from copies that have been renamed and assigned new project folders.

1. Define the values below. Within the sensor channel object's initialization, assign the base limit values below to each of the limit variables within the sensor channel object array.

   HIGHLIMIT – 85 // Degrees C
   LOWLIMIT – 15 // Degrees C

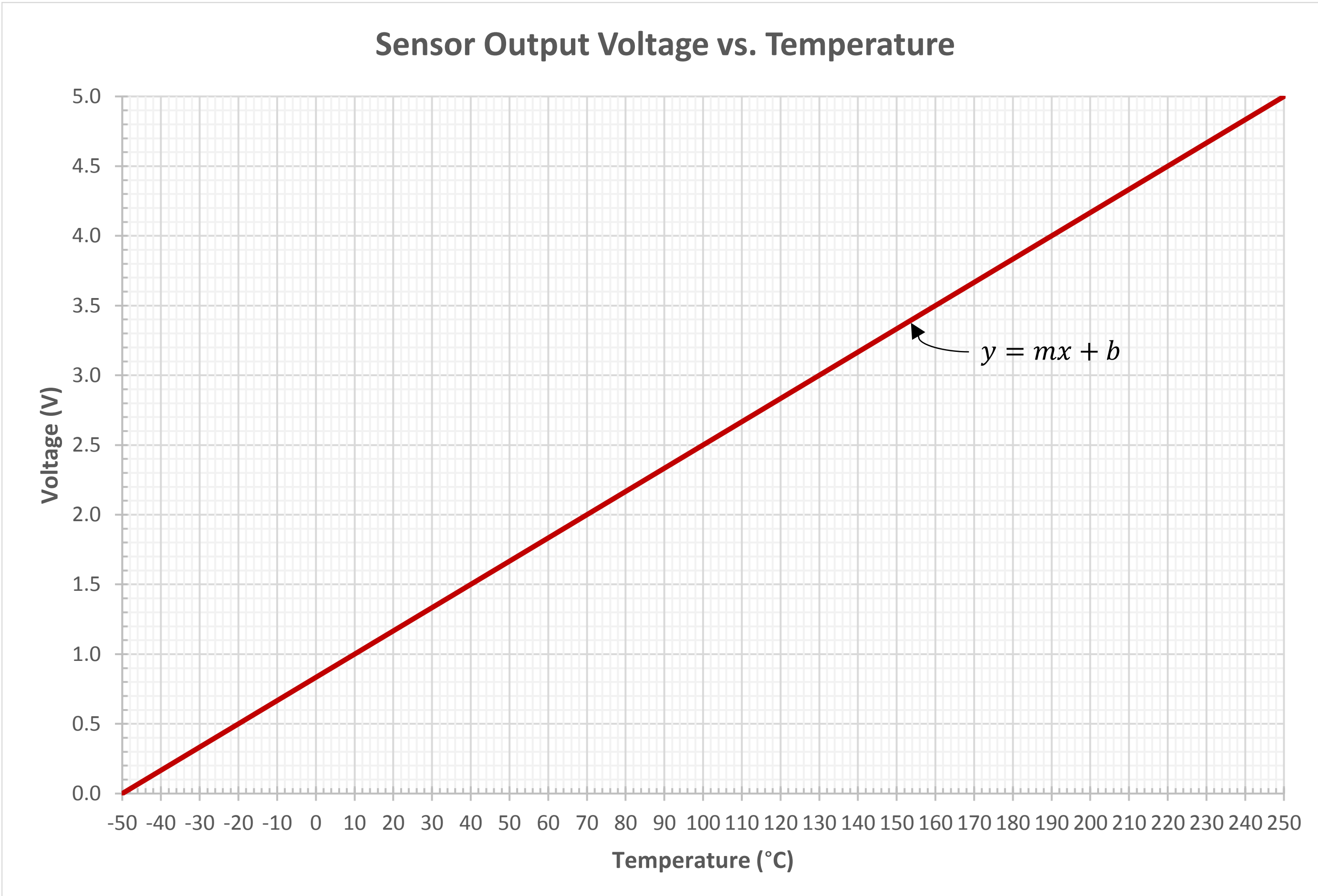
2. Create global control variables for the sensor system. A control variable for channel selection and limit mode can both be small variables that represent the number of channels in the system and either high or low for limit mode selection. Name them "channelSelect" and "mode". Initialize these variables to 0 and FALSE, respectively.
   - channelSelect will hold an absolute value that represents the sensor channel which the user wishes to change a setting.
   - mode will select either the high limit value or the low limit value being changed by use of a TRUE or FALSE value assigned to it.

3. Create the follow functions with the features as defined below:
   a. **Increment Limit** – when the increment button is pressed, the channel select and mode values will be used to increment the limit (either high or low) within the channel selected.
      ➔ Example: if channelSelect is 0 and mode is TRUE, the high limit value on channel 0 will be incremented.
   b. **Decrement Limit** – same as increment but will reduce the value by 1.
   c. **Channel Select** – this function will execute if the channel / select button is pressed. The channelSelect variable will be increased by 1 with each press. If the channelSelect value is greater than or equal to the maximum channels in the system, the channelSelect variable will be reset to 0.
   d. **Mode Select** – this function will toggle the mode variable value from TRUE to FALSE or vice-versa.

4. Within the indefinite loop of the program, if the pushbuttons are not equal to an UNPRESSED state, read the PBs with a MASK value and take the appropriate action, based on the functions defined above. This should only happen once per press.

5. Apply the linear equation below to the average sample value for each of the sensor channels. We are assuming all three sensor channels are for temperature sensors of the same type.

   Apply the transformation of 5V / 1024 bit value within a floating point type variable. That will transform the average sample value into a voltage. Save the completed calculation into the average sample variable when done. Only whole values of °C.



Graph 1: Sensor Output Characteristics – Voltage vs. Temperature

$$m = \frac{(range\ of\ total\ voltage\ output)V}{(Range\ of\ total\ Temperatures)°C} = m\ value\,\frac{V}{°C}$$

$$b = (Absolute\ value\ of\ total\ negative\ temperature\ °C * m)\ V = b\ value\ V$$

$$y = ADC\ result\ (average\ sample)$$

$$x = temperature\ sensed\ in\ °C$$

Average Sample $x = \frac{(y-b)V}{m\frac{V}{°C}}$ = The end result should be in °C.

6. After an average is calculated, compare the average of a sensor channel to the limits within the sensor channel. The average sample must be between the low limit value within the channel and the high limit value within the channel.
   If the average sample is outside of the limits, light the channel's LED. The terminal must also display a status of "Safe" or "ALARM!" Example display below.

7. Continue to display live, most recent raw ADC samples but for the average sample and the limits values, they should now be displayed as temperature in °C. Channel Status must also be displayed as prescribed.

   Example Display:

ELNC6007CRT**Lab4**

Channel Select: 0 **Mode: High**

Sensor 0		**Sensor 1**		**Sensor 2**	
Current:	452	Current:	1023	Current:	561
Average:	82°C	Average:	249°C	Average:	114°C
HighLimit:	85°C	HighLimit:	85°C	HighLimit:	85°C
LowLimt:	15°C	LowLimt:	15°C	LowLimt:	15°C
Status:	Safe	Status:	ALARM!	Status:	ALARM!

Blue text are specific elements that have changed since last exercise.

This exercise will be expanded in future exercises. Do not leave your work to the last minute. Keep on top of each exercise as they are deployed.

Demonstration due the week of March 15th, in your regularly scheduled lab session. Schematics must be shown prior to demonstrating a working exercise.