1) From mycourses download this pdf.
2) The **objective** of this lab is to extend the functionality of the code developed in lab 10, by transmitting the acquired value via UART to a Terminal program. The lab consists of one part only.

3) For this lab maintain the same board jumper configurations as in labs 10 & 11.

4) **Part 1: Design, code, build, run, demonstrate, and debug fmlxxxx_Lab12_a1.asm or fmlxxxx_Lab12_c1.c code.**

5) You can implement this lab assignment in assembly only, in C only, or a combination of the two. It is highly recommended that you split the code in multiple files.

6) Below is the description of the Lab10 program. I added in **bold** the functionality that you have to add in this lab.

7) The functionality of the program is described below:

    a. After initial configurations **(including now those of the USI)**, the program queries in a loop all five-touch sensors. The result of the query is captured in one value, as you have done in a previous lab. Every time it detects the touching of a sensor it gives the user feedback by turning ON some LEDs until it senses another sensor. We assume an educated and polite user.

    b. It is expected that the user touches the center sensor first (start). When the uC determines that it has been touched, it continues to look for a touch of the wheel top or wheel down sensors. Change the LEDs to let the user know the application is now looking for up/down.

    c. If the wheel top is touched, the acquisition will consist of 20 samples at a rate of ~100 Hz.

    d. If the wheel down is touched, the acquisition will consist of 20 samples at a rate of ~200 Hz.

    e. Next, the uC is checking if the wheel left or right is touched. Change the LEDs to let the user know the application is now looking for left/right.

    f. If the wheel left is touched, it will display each sample value for ~1 second.

    g. If the wheel right is touched, it will display each sample value for ~2 seconds.

    h. Next, the program is performing the acquisitions, with the parameters as described and selected above.

    i. Once the acquisition is complete, it will convert all samples to an 8-bit

"leading 1, non-linear scale" just like in Lab 10.

j. After converting all samples to an 8-bit "leading 1, non-linear scale", you display each sample value based on the parameters described and selected above on the LED's. **Each time a new sample is displayed on the LED's, send it out over TXD (every 1 or 2 seconds). You don't need a transmit or receive ISR. Just check that the TXD-Buffer is ready and write the value to it. Then return to the next iteration of the loop.**

k. The program continues to display in a loop until the center sensor is touched again and the whole loop repeats.

8) Before using the eclipse built-in terminal client, you can choose to receive via loopback and ensure that your transmission works. Check the receive buffer.

9) Once the code compiles, apply a sinusoidal signal with a peak-to-peak amplitude of 1 V, a DC offset of +0.5 V, and a frequency of 10 Hz at the input. Make sure your signal does NOT go over 1V or below 0V!!! Check with an oscilloscope first, before connecting it to your MSP430.

10) Once the code is debugged, try three DC voltage values at the input of the ADC, again making sure your signal does NOT go over 1V or below 0V.

11) **I will be available for help during regular office hours. The TA's help ends this week.** Make sure you write the final report and upload it along with your project archive on mycourses before the dropbox due date!

12) **<u>Grading</u>**:

Working demo (50%) - DUE NEXT WEEK, code should be in the dropbox
Check overall functionality.
Check that touch sensors are working properly, including being debounced - no glitches.
Check correct use of timers for sampling and display.
Check LEDs used for displaying 20 samples - must use multiplexing to turn on both columns.
Check UART output of the 20 samples - must match with LEDs display

Q&A (25%) - DUE LAST DAY OF CLASS, can be done anytime before that. You must schedule a time for this, if outside of LAB.
3-4 questions specific to your code.
Q's about code flow (flow chart will help)
Q's about how/where you used timers
Q's about how/where you used interrupts
If we point to any part of the code, you should be able to describe what it does
Check that the code is well commented - not line by line, but in a more general sense.

Final report (25%) - DUE Saturday AFTER LAST DAY OF CLASS at 6pm, in the dropbox.
Flow chart!!!!! It HAS to be included!
If you have the flow chart ready before the Demo, it will make your Q&A a lot easier.
This report is not a 1-page document like the past ones.
It has to be a proper IEEE report around 3-4 pages, with everything you would expect from a final design report, as described in page 2 of the lab syllabus. Make sure you submit in PDF format. PDF!!! NOT WORD!!!