Embedded Systems Lab 12 Report

Cameron Neville

***Abstract*—This project demonstrates many of the key features of the TI Launchpad. The project requires the use timers, capacitive touch sensors, LED modulation, ADC, and UART communication. The project can only function properly if all of these tools are implemented properly.**

I. Requirements

The project was to be completed using TI Code Composer Studio, using one or more C and/or assembly programs. The code was to run on a TI Launchpad development board with an MSP430G2553 microcontroller.

When run, the board will indicate that it is awaiting input. The user presses the center button, then presses up or down, then presses left or right. The board will use the on-board LEDs to indicate which set of inputs it is waiting for.

The board will then sample an analog signal at one of the input pins using the 10-bit ADC. The board will always take 20 consecutive samples, but the sampling frequency is dependent upon whether up or down was selected. If up was pressed, the sample frequency is 100 Hz; if down was pressed, the sample frequency is 200 Hz. The samples are stored in an array for later use.

The board will then begin displaying the sampled values consecutively. The values are displayed on the eight LEDs using a logarithmic scale. The number of LEDs turned on for each sample is equal to $log_2(N) - 2$, where N is the digital sampled value (for values less than 8, no LEDs are turned on). If left was pressed during configuration, one sample is displayed every second. If right was pressed, one sample is displayed every two seconds. In order to use all eight LEDs at once, they must be pulsed faster than the human eye can see, so they all appear to be turned on simultaneously.

When each sample is displayed, it is simultaneously transmitted to a connected PC via on-chip UART hardware. A baud rate of 9600 is used, which must be calibrated using an oscilloscope or logic analyzer connected to the TX pin.

The board will continue to loop through all twenty sampled values until the center button is pressed again. It will then return to waiting for up or down to be pressed, and begin the process again.

II. Implementation

The capacitive touch sensors were implemented using internal oscillators on the digital I/O pins P2. To test whether a sensor is being touch, the oscillator is started and used as the clock for an internal timer. The oscillator is allowed to operate for a set amount of time, and the timer records the number of oscillations. This is then compared to a baseline determined for each sensor at the start of operation. When additional capacitance is attached to the oscillator, the frequency decreases measurably. When the difference between the measured value and the baseline is above a specified offset, the sensor is determined to have been pressed. The offset can be changed to adjust the sensitivity of the sensors.

The LEDs are controlled by pins P1.3 through P1.7. The eight LEDs are divided into two sets of four. One set uses P1.3 as a common anode, while the other uses it as a common cathode. Each pin P1.4 through P1.7 controls one LED from each of the two sets. Only LEDs from a single set can be turned on at a time. To create the illusion of having LEDs from both sets turned on at the same time, the LEDs are flashed faster than the human eye can distinguish. While a frequency of about 100 Hz is typically sufficient, a frequency of about 1000 Hz was used to ensure the pulsing was unnoticeable.

The ADC used a reference voltage of 1.5 V and ran off the system clock, meaning conversions were completed quickly and did not bottleneck processing. Inputs were limited to 0 V through 1 V, which converts to 0 through 682 (0x2AA).