# EE 316 Computer Engineering Junior Lab Design Project 3

**Spring 2023**

**Specification: Signal Generation using sensors with I2C interface Due Date: Thursday, March 2 - lab demo**

**Tuesday, March 4, written report due.**

**Parts list:**

* Trenz ZynqBerry board with GPIO Expansion board
* One function generator
* One 16x2 character I2C LCD display module
* A PCF8591 ADC-DAC module with I2C interface
* A couple of Op-AMPs (LM741/LM358 compatible), capacitors, resistors, etc.

Design a system to sample, store, display and (re)create analog waveforms. Reset uses an onboard button BTN0. In this project, use Vivado 2022.2 instead of Vivado 2019.1 with the board files (already installed on AppsAnywhere).

**Op-Amp adder circuit:** Periodic sinusoidal waveforms will be generated by function generators. The input waveform should have peak-to-peak amplitude of 3.3 Volts. **Note:** The ADC’s should sample positive analog voltages; thus, the input signal will need to be added to 1.67 volts using an op-amp adder with a gain of 1.

**ADC:** The PCF8591 module has 4 input channels, AIN0, AIN1, AIN2 and AIN3. The output of the Op-Amp Circuit (**ANALOG\_INPUT**) should be connected to AIN2. The other three Inputs can be connected to three sensors using 3 jumpers as shown below:

* AIN0 – Jumper P5 – Light Dependent Resistor (**LDR**)
* AIN1 – Jumper P4 – Thermistor (**TEMP**)
* AIN3 – Jumper P6 – Potentiometer (**POT**)

**PWM:** The system should have a PWM output. The duty cycle of the PWM signal will be proportional to the 8 bit value of the data The user will be able to select the between the 4 inputs to determine the PWM duty-cycle using one of the onboard button (BTN1). The PWM output from when AIN2 is use should pass through a low-pass filter to reconstruct the input sinusoidal waveform and displayed on the oscilloscope along with the original input. In your design, what is the highest input frequency the system can reconstruct?

**Clock Generation:** The system will also be able to generate a clock output. The frequency of the clock will vary between 500 to 1500 Hz and is controlled by the Potentiometer, the Thermistor and the LDR. This module can be enabled or disabled using the same button (BTN1). You can use on-board LEDs to show the states.

**LCD**: There will be an LCD connected to the Zynq board. Use the 4-bit data option for the LCD. The LCD will display the selected source for that makes the PWM signal on the first line of the LCD. The second line of the LCD display “**Clock Output**” should indicate if the system is currently generating a clock.

# Other objective are:

* to maximize the bandwidth of analog signals that can be sampled, stored and analyzed without aliasing
* to make sure the error for the desired frequency is minimized.

# Useful Links:

* [P CF8591 – 8-bit ADC and DAC converter](https://www.nxp.com/docs/en/data-sheet/PCF8591.pdf)
* [A rduino PCF8591 Digital to Analog Tutorial](http://henrysbench.capnfatz.com/henrys-bench/arduino-voltage-measurements/arduino-pcf8591-digital-to-analog-tutorial/)
* [L esson 12 PCF8591 AD Converter](https://www.sunfounder.com/learn/lesson-13-pcf8591-ad-converter-sensor-kit-v2-0-for-b.html)

# Teams:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Team 1 | Team 2 | Team 3 | Team 4 | Team 5 | Team 6 | Writer |
| Kubicka | LaBue | Morris | Skinner | Ernesto | Mathew |  |
| Kaili | Isabelle | Graham | Pamela | Cameron | Shawn |  |
| Nathan | Zander | Keith | Ella | Jacob | Macy |  |
|  |  |  |  |  | Nelson |  |