A blue background with white letters

Description automatically generated

Lab Project Notes

ELEX 7660

Bryce Reid & Parry Zhuo

2025

# Modules

## enc2sel

* input left encoder directional output
* Cycle through parameters
* Update selected parameter output
* sel = 0 -> select wave shape
* sel = 1 - > select rate
* sel = 2 -> select wave depth

### Module Declaration

module enc2sel ( input logic cw, ccw, output logic [1:0] sel, input logic clk, reset\_n ) ;

## enc2param

* input right encoder directional output
* Increment/decrement selected parameter value to output
* There are 6 waveform shapes to cycle through (sine, square, triangle, ramp up, ramp down, & pseudo-random)
* I haven’t determined what range of rate and depth outputs will be most suitable yet

### Module Declaration

module enc2param ( input logic cw, ccw, input logic [1:0] sel, output logic [2:0] wave, output logic [#:0] depth, output logic [#:0] rate, input logic clk, reset\_n ) ;

## freqGen

* Input desired rate from enc2param
* Generate wclk with frequency require to produce desired output rate ( similar to tonegen in lab 3 )
* Output wclk

### Module Declaration

module #( parameter fclk ) freqGen ( input logic [#:0] rate, output logic wclk, input logic clk, reset\_n ) ;

## waveGen

* Use LUT's to define each wave shape
* Use wave input to select which LUT to use
* Use depth input to scale LUT indices
* Use rate input as clock

## DAC\_interface

* SPI master to interface with the DAC

A diagram of a diagram

AI-generated content may be incorrect.

CLR & LDAC pins were tied to VDDIO (pulled high) for testing, as they are unnecessary for normal SPI operation.

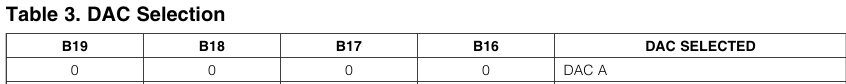
A screenshot of a computer

AI-generated content may be incorrect.



A table with black text

AI-generated content may be incorrect.





Thus, to code\_load DAC A my command signal is: **0011\_0000\_D[11:0]\_0000**

**TEST SIMULATION**

A green lines on a black background

AI-generated content may be incorrect.

*Ignoring the “data\_req” and “data\_ack” signals for now as they were intended for interacting with another module and are not necessary for SPI interface testing.*

**From Datasheet:** “The serial input register transfers its contents to the destination registers after loading 24 bits of data on the ***24th SCLK falling edge***.”

I tried a version where SCLK is continuously running since the datasheet says “During CSB high periods, SCLK is ignored”. This didn’t work.  
I tried a range of SCLK frequencies from **97.65625 kHz to 12.5 MHz**. The datasheet says it should work between 0-50 MHz and I found a given microcontroller example for the DAC which uses 12 MHz.

## LCD\_interface()

* Interface module for the LCD Display

# Top-Level Flow Chart

A diagram of a computer

AI-generated content may be incorrect.