EECS 3201 - LABS 8-9
Project - Function Generator

Objective

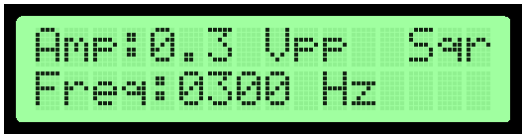
Create a fully functional function generator in Verilog.

Specifications

In this lab you will create a function generator with the following features:

- Capable of generating Sine, Square, Triangular and Sawtooth waveforms
- Variable frequency between 0 and 1KHz (user selectable by cycling though at least 10 fixed values in this range)
- Variable amplitude between 0 to 3.3 Vpp, in 0.1 steps
- Waveform definition of at least 512 points per cycle

The output is provided through the DAC (digital to analog converter) Pmod. The DAC is controlled through a serial-type communication, so you will need to send data to it together with a clock. Each time a new value is transmitted to the DAC, its output voltage is updated. Therefore to create a signal output that varies in time, the DAC output value needs to be updated at a predetermined rate. The value for the output voltage is encoded into a 16-bit long command (first 2 bits are don't cares, see datasheet). You will also have to implement an LCD module that shows the characteristics of the wave being generated.



Example of how the LCD module will display the current settings

You are free to determine which buttons or switches will control the parameters and how. It is suggested that the external pushbutton Pmod is used rather than the on-board pushbuttons, as the external board uses de-bouncing filters and there is a lower risk of glitches.

Here are some further tips and guidelines for your design:

- Make a modular design for simplicity. For example, have a separate module to send data to the DAC.
- Waveforms such as sine, triangle and sawtooth are trickier to implement than a square wave. A tip is to use lookup tables to generate the shape of the wave. You can easily create a list of values for the waveforms (at least 512 values) in a tool like Excel. Then you can simply append or prepend to the list of values the syntax needed for Verilog, easily generating the list of Verilog lines to copy and paste in your code. You can finally control the amplitude of the waveform by scaling this list of values by a multiplier, depending on the user's selection.
- Since the DAC is unipolar (it can only output positive voltages 0-3.3 V), we suggest having a constant DC output on top of which to superimpose the desired waveform.
- The DAC module should not be interrupted during the transmission of its data. It is therefore recommended that you use a 1-bit register as a semaphore to insure that no new transmission of the data is initiated until the previous one as completed.
- You can achieve variable frequency output by setting the rate at which you communicate the waveform data to the DAC.
- You can use the example LCD module below to implement your version.

Procedure

For this project, you can come to the lab to program and test your design. You will also be allowed to use the board and Pmods outside EECS 3201 lab hours by signing them off at the lab monitor station. A TA will be available during normal EECS 3201 lab hours.

Show your design to the TA when done – the project is due by the end of your lab 9 session.

Resources

DAC PMod Datasheet LCD Sample Module