Date: March 19th, 2019

From: Adam Kwok

To: Doctor Kaputa and anyone else interested in my work

Subject: PyQT4 Running PWM with AXI on a Snickerdoodle

Introduction

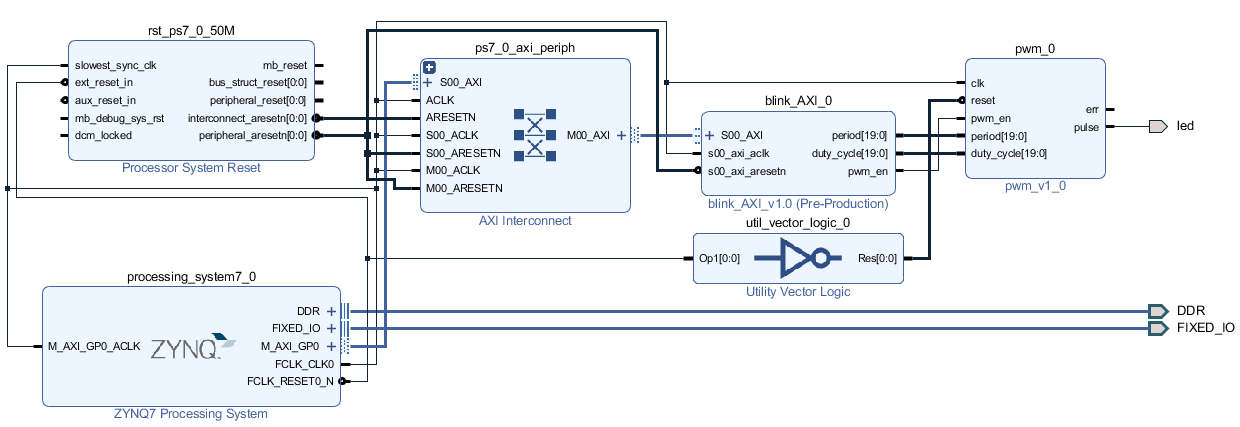
For the implementation of a PWM module designed in VHDL, Xilinx Vivado was utilized to design the AXI interface. The important factors in this design were that the PWM required an enable, and control of both the duty cycle and period controllable from the processor. Additionally, the graphics for the project were to include a enable indicator and have an output signal verifiable by oscilloscope. The reason for these requirements was to ensure the following:

* Establish and demonstrate full communication and interaction with the Snickerdoodle
* Modify pins on the Snickerdoodle through the PyQT GUI.
* Provide alert signals for the user via the user interface

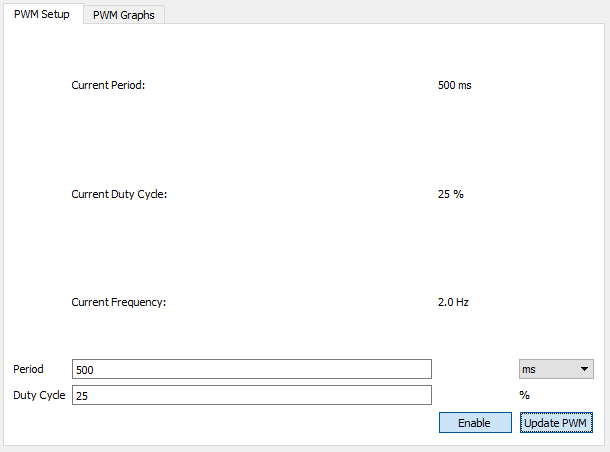
The below analysis investigates the application of the PWM module and also considers how other AXI interface may require relatively similar processes.

Analysis

With the PWM VHDL module having been developed in the first lab, all that was needed to do was to develop the AXI interface, create the GUI, and work through the communication paths necessary to get the necessary resources onto the Snickerdoodle. Upon creating the AXI interface for the PWM module within Xilinx Vivado, connections were established resulting in the following block diagram:



On completion of creating the block diagram and establishing the connection with the PWM signal, the final step was simply to develop the GUI and demonstrate functionality. The GUI was developed such that values that were to be loaded on “Enable” were displayed via labels and these labels would be updated with updated when correct inputs were entered for the period and duty cycle and the “Update PWM” button was selected.



Conclusion

* The user can enter a value of 0 or greater for the period entering a negative value will result in an error message.
* The user can enter a value from 0-100 for duty cycle. Entering a value outside of this range will result in an error message.
* Should the snickerdoodle lose connection with the controller or computer, ideally a motors off, land, or hover should be put in place as to mitigate potential harm.
* Since the period and duty cycle modifiable lines are not processed until the “Update PWM” button is hit, ensure that button is hit whenever a modification is made before enabling (if you want the old values to be replaced).