

Project Report: 3D Project

Cameron Anderson

May 13, 2019

1 Introduction

The goal of this project is to use the ADXL362 3-axis accelerometer on the Nexys 4 DDR to map a three dimensional environment by moving the board through space. For example, the board could be used to approximately measure three different size boxes. To map the 3D environment, MATLAB would need to be used by taking UART values from the Nexys 4 DDR as it is moved and plotting the mapped values. This and other projects can be seen at the link below.

My Softcore SoC FPGA Github Repository

2 Experimental Plan

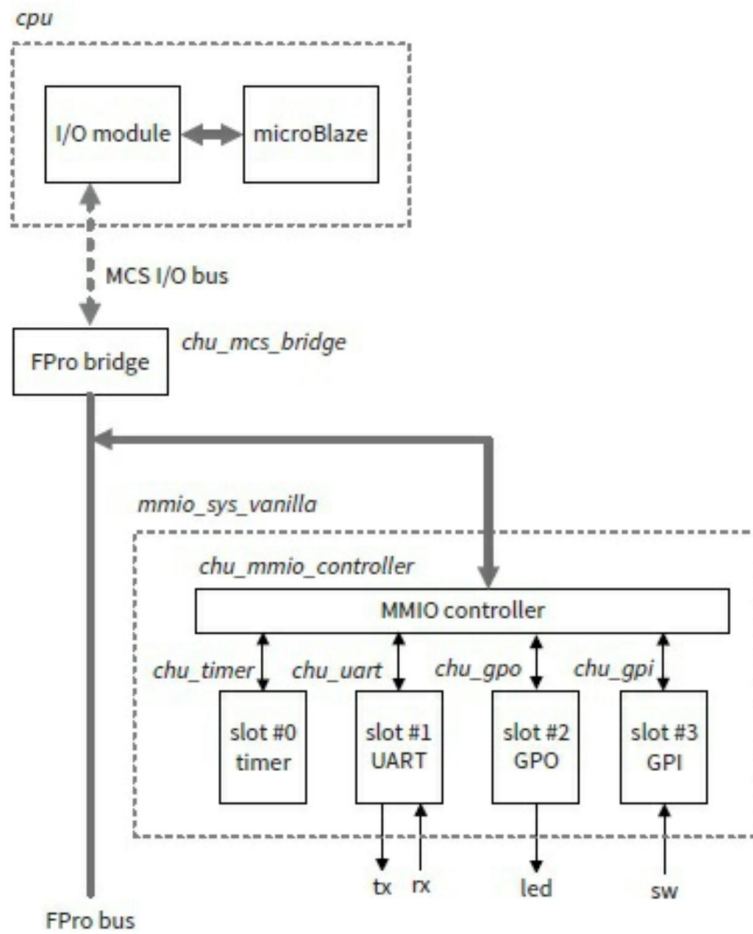
2.1 Microblaze Processor

The first step of the plan is to implement the Microblaze processor in Vivado Design Suite because the processor will be used to ultimately control the required peripherals for the project. The instantiation of the processor is done from the Project Manager window in Vivado Design Suite when IP Catalog is selected. From the IP Catalog Window, Embedded Processing > Processor > Microblaze MCS are selected. Once the core configuration is complete, the `cpu_unit` requires a wrapper to interface with the microprocessor. In this case, the `mcs_top_vanilla.sv` file that Chu provides houses the instantiated microprocessor.

2.2 SPI and UART interfacing

The second part of the plan is to identify and instantiate the necessary modules for the project. The ADXL362 accelerometer operates as a slave module on an SPI bus. To use the accelerometer, an SPI bus module is needed as is a wrapper for the bus that is used to control the bus and slot it into the MMIO unit. Additionally, the data from the accelerometer must be passed to MATLAB via UART protocol. To make a UART bus, Chu provides a UART unit that houses modules for baud rate generation, a receive state machine, a transmit state

Figure 1: Microblaze MCS and MMIO Controller



machine, a fifo rx state machine, and a fifo tx state machine. Chu also provides the wrapper for slotting the UART bus within the MMIO unit. The MMIO unit is housed in the processor block and has universal slots for peripheral units. Chu provides this module as well for the microprocessor. It is worth noting that had Chu not provided these modules, not all of them would be need. For example, only the UART tx functionality is of interest since this project reads from the SPI bus.

2.3 Software Development

Part 3 of the plan is to create the hardware specs that are the foundation for the software development in Xilinx Software Development Kit. The .hdf file is generated by selecting file, export, and export hardware in Vivado Design Suite. Once the .hdf file is generated, the hardware specs are created by selecting File, New, Other, and Hardware Platform Specifications and then uploading the .hdf file. Part 4 of the experimental plan is to incorporate the hardware specs into a board support package that will contain drivers and start-up routines. This is done in Xilinx SDK by selecting File, New, Other, and Board Support Package. Once the target hardware has been selected and a standalone OS is chosen, the BSP can be generated. The BSP must be generated after the Hardware Platform Project. The fifth part of the project plan is to create a software application. In Xilinx SDK, select File, New, and Application Project. From the menu, a C++ project can be created. Part 6 of the project is to write the necessary software to implement our design. The C++ code needs to initialize the SPI and UART buses, then it needs to continuously read from SPI and write to UART. Lastly, MATLAB needs to take the UART values and plot the corresponding 3D space.

3 Analysis

3.1 ADXL362 Accelerometer

To interact with the accelerometer, the SPI bus must be modeled to correctly interface with the device. The ADXL362 Datasheet is easy to find via a Google search.

4 Conclusion

The Fetch stage was successfully implemented. The Fetch stage is the first of five stages that will make up out 64-bit ARM processor. Due to the parameters we used in the Fetch stage, the instruction size and instruction memory could be changed later if need be.

5 Appendix