Chapter 1

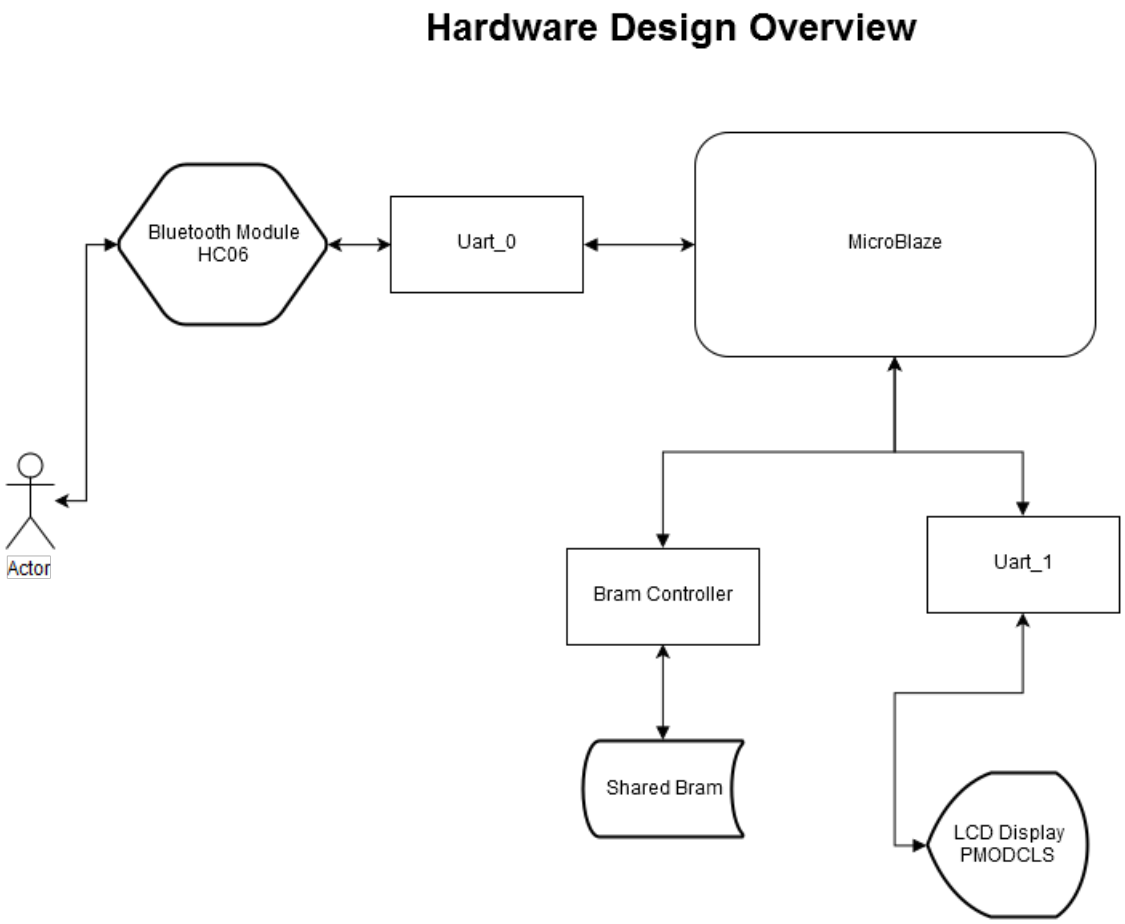
Hardware Design

1.1 Objective

The aim of this project is to build a system that is capable of receiving data via Bluetooth, saving those data and displaying them on a LCD.

1.2 Overview

The input interface of system is a Bluetooth module, with this module the system can communicate with the outside world. The Bluetooth module sends and receives serial data and is connected to processor via an UART module. This module is controlled by the processor. The processor saves the data on a BRAM block using a Bram Controller. In order to provide a user feedback a LCD is used. This LCD is connected to the processor via another UART module. In a typical use case scenario, data is received via Bluetooth, processed by the processor, saved and then displayed on LCD.



1.3 Processor

As processing unit a MicroBlaze soft core processor implemented on the FPGA.

1.3.1 MicroblazeâĎć

MicroBlazeâĎć is a key element of XilinxâĂŹs Embedded Product Portfolio. As a full-featured, FPGA optimized 32-bit Reduced Instruction Set Computer (RISC) soft processor, Microblaze meet requirements for diverse applications such as industrial, medical, automotive, consumer, and communication infrastructure markets among others. MicroBlaze is a highly configurable and easy to use processor and can be used across FPGAs and All Programmable (AP) SoC families. It is included free with VivadoÂő Design and System Edition and Vivado Webpack Edition. It is also

available as part of legacy IDS embedded edition for older FPGA device families like SpartanÂő-6, VirtexÂő-6 etc.

MicroBlaze is highly configurable IP core supporting 70+ configuration options. Some of the key configuration options are Instruction/Data Cache, Floating Point unit, Memory Management Unit etc. With highly flexible and configurable core, user can implement virtually any processor use case, from a very-small-footprint state machine or micro controller to a high-performance, compute-intensive microprocessor-based system running Linux. The IP can be configured to operate in either a three-stage pipeline mode (to optimize for size), or in a five-stage pipeline mode (to optimize for speed)âĂŤthus delivering faster DMIPs performance than any other FPGA-based soft processing solution.

1.3.2 Processor Design

During the design process two UART IP modules added to the design in order to utilize the Bluetooth module and LCD. A Bram block and a Bram Controller is added to design for the data storage. The MicroBlaze processor and the modules mentioned above are connected over the AXI bus.

1.4 Bluetooth Connection

As a Bluetooth module, HC-06 is used. It has its own buffer which it can use to save and receive data. The TX and RX pins are connected to the UART module. Any received data over the Bluetooth comes to the FIFO receive register of the UART module. Similarly, any data to be send first written to FIFO send register of the UART module and then sent via the Bluetooth module. UART Controller is programmed and utilized by the software written on the processor. Interrupt mode of the Bluetooth controller is used for receiving data, on the other hand for sending data Polling mode is used.

1.5 Data Storage

Storing data in a memory unit outside the memory map of the processor is one of the key aspects of the project. This way the system can share data with other systems without corrupting the memory of the processor. In order to do this a Bram block and a Bram controller are used.

The system accepts 8-bit data from user however Bram block has 32-bit registers. In order to utilize the memory more efficiently an algorithm is developed to write four 8-bit unsigned integers on a single register and keep the track of the memory addresses. This algorithm also provides the necessary information to read a specific data from a specific register.

1.5.1 Functional Description

At runtime processor receives data via its Bluetooth interface and keeps it on its own memory. After initial processing of the data it is saved on the Bram block by utilizing Bram Controller. Bram Controller has a vendor library which provides a vast number operations on the Bram block. With this library writing data on Bram becomes a simple task of writing data on a memory address provided by the Bram Controller.

1.6 Display

In order to provide a useful feedback to user a LCD module is used. It is connected to a the processor via UART interface. It allows user to see what data is saved on BRAM block. The module chosen for this project is PmodCLS. It supports the UART protocol and some vendor provided basic commands.

1.7 Software

All of the software is written in C language using the tools provided by the vendor. Software is divided in subsystems corresponding to the hardware modules. One of the

biggest challenge was to make sure all the components work as intended in cooperation with each other. Therefore software needed to be designed with extra care and detail.