MAY1735

ISU Department of ECprE

Dr. Daniels

Kyle Fischer - Team Lead

Michael Linthicum - Communications Lead

Daniel Shauger - Concept Holder Lead

Sam Neff - Webmaster

Nick Juelsgaard - Schedule and Planning Lead

may1735@iastate.edu

http://may1735.sd.ece.iastate.edu/

Revised: 11/7/16

CyMote for CprE 185

Design Document V1.0

Contents

[1 Introduction 2](#_Toc466370214)

[1.1 Project statement 2](#_Toc466370215)

[1.2 Purpose 2](#_Toc466370216)

[1.3 Goals 2](#_Toc466370217)

[2 Deliverables 2](#_Toc466370218)

[3 Design 3](#_Toc466370219)

[3.1 System specifications 3](#_Toc466370220)

[3.1.1 Non-functional 3](#_Toc466370221)

[3.1.2 Functional 3](#_Toc466370222)

[3.2 PROPOSED DESIGN/METHOD 4](#_Toc466370223)

[3.3 DESIGN ANALYSIS 4](#_Toc466370224)

[4 Testing/Development 4](#_Toc466370225)

[4.1 INTERFACE specifications 4](#_Toc466370226)

[4.2 Process 4](#_Toc466370227)

[6 Conclusions 5](#_Toc466370228)

[7 References 5](#_Toc466370229)

[8 Appendices 6](#_Toc466370230)

# 1 Introduction

## 1.1 Project statement

CprE 185 currently utilizes a device called the Arduino Esplora. The Esplor is a handheld controller with a simple set of I/O on board, including some buttons, a joystick, Bluetooth, and, most importantly, a three dimensional accelerometer. The students in CprE 185 collect real-world data from the Esplora to solve interesting programming and physics-based challenges. Unfortunately, the Eplora has been discontinued.

Our Senior Design group will design and build a beta model for a new device for CprE 185 specifically designed for this class. This new device, called the CyMote, will provide all the functionality previously provided while costing less. The design will be on a PCB with student generated control code.

## 1.2 Purpose

The department has specifically designed CprE 185 to challenge freshmen students to think like engineers. It is important for students to associate the work they do in class to solving problems in the real world. The Esplora was that connection. It allowed both user input in the form of the joystick and buttons, but it also allows for collection of the effect of the laws of physics through the accelerometer.

The device that we develop will allow students a meaningful pathway into solving real-world problems. This device will reliably provide the desired I/O, withstand constant handling, and be easy enough for a freshman student to figure out. The three main tenants of the project design are:

1. Reliability
2. robustness
3. Simplicity for End User

## 1.3 Goals

The goal of the project is to create a beta model of a demonstrably robust CyMote device that will be capable of all required functions. As students, our goal is to learn and grow as technically skilled engineers. As future professionals, our goal is to refine our skills in communication, teamwork, presentation, as well as project and time management. As ambitious, tech savvy young people, our goal is to complete a project that excites both us and other people. As seniors, our goal is to graduate.

# 2 Deliverables

There are two main deliverable items for this project: (1) A physical beta model of the CyMote for display and presentation, and (2) a PC wrapper that will provide a student with information from the CyMote for use in class.

# 3 Design

The CyMote will emulate the Arduino Esplora in all facets that pertain to the CprE 185 class. The CyMote will be loaded into a soft casing and be dropped from the third floor of the Coover Atrium. This is the main use case. The CyMote will be self-powered and will send info back to a computer over BLE. It will send enough data fast enough for the students to get see visibly how the acceleration of the device is affected by the drop and the landing.

## 3.1 System specifications

The beta model of the CyMote will be laid out on a PCB. It will use the ATSAMB11 to control various peripherals. The peripherals used are:

* Joystick w/ button
* 4 buttons
* Power Switch
* 9º of Freedom sensor
* 3.7V Battery
* RGB LED

The software component will be written in C using Atmel Studio and ASF project structure. There will be a wrapper program on the CyMote’s corresponding PC. This program will be a Windows executable. The wrapper will handle communication with the CyMote and will present the desired information to the student in the command line.

### 3.1.1 Non-functional

1. Look like a game controller
2. Stream data in real time – fast enough to register current movement on screen as having no lag
3. Beta model built on a handheld size PCB

### 3.1.2 Functional

1. Droppable – survive a 30’ drop in a soft container of some kind
2. Robust enough to have people plugging/pressing/slamming/charging often
3. Communicate with a partner PC over Bluetooth wirelessly
4. Low power consumption from onboard rechargeable batteries
5. I/O components
   1. 9° of Freedom – 3D accelerometer
   2. Joystick – (x,y analog inputs, button press)
   3. 4 game controller style input buttons
   4. Status LEDs for all the devices
   5. Tricolor LED for display/status/fun
6. Power on/off switch
7. Five hours of battery life
8. Communicates without loss or bugs over USB and BLE
9. Streams >30 accelerometer records per second
10. Permanently named (physical, BLE, code) for paring with computers

## 3.2 PROPOSED DESIGN/METHOD

We have divided this project up into several phases. First we would like to pick devices that we know will work. Then we would like to get them all together and design an Alpha model that will allow us to determine if our pieces will work together. After that we will begin designing the PCB for the beta model. As we develop the hardware for the beta model, we will begin creating the software for the board. Once we know that we can use the board to communicate over BLE and USB to all the on board sensors, then we will begin creating the wrapper program.

## 3.3 DESIGN ANALYSIS

So far we have spent a decent amount of time finding appropriate pieces, defining the scope of our project, and testing our chosen MCU.

We have chosen the Atmel ATSAMB11 as a controller. We purchased three development board that Atmel provides to start working with these controllers. We have used startup documentation provided by Atmel as well as some pre-designed programs to develop our BLE communication and SPI communication. So far we have gotten buttons, LEDs, the joystick, SPI, and BLE to fully function with their “Hello World” programs.

For the hardware, we decided that we will be putting the beta model on a PCB. We have talked to Lee about how to get this done. He had a lot of good ideas, and we’re planning on trying to get our first board cut before Christmas if possible. As for power management, we’ve done some basic research on battery power and had a good conversation with Dr. Tuttle about how to choose the correct power management ICs. We are in the process of ordering some test pieces.

# 4 Testing/Development

## 4.1 INTERFACE specifications

The over hardware is the board. We will be communicating with the MCU over USB and BLE. The MCU will be our entry point to the 9º of Freedom. The board will also charge its batteries over USB.

## 4.2 Process

We will be testing each hardware piece on its own first. We have already tested almost all the pieces alone. The software team can talk to the MCUs, talk to the 9º of Freedom over SPI, and can get feedback from the buttons and the joystick. They have also made the LED light up.

We are developing our first alpha board schematic right now. Once we get all the pieces installed we will be testing the robustness of the design. Once we have a robust design, we will begin design on the PCB. Once we have a PCB cut, we will try out hand at surface mount soldering. We will verify the board and make any necessary changes, and then order our final boards and try to make a solid beta model.

# 5 Conclusions

So far we have developed out scope, spent time specifying our parts, and started making progress with the hardware design. We have already made very good progress with the software. We are on track to have an alpha model done by the end of fall semester, and have a PCB beta model done in May.

# 6 References

There are no official references right now, but our unofficial references are:

* Previous senior design group that did not quite complete the project with an Edison controller
* Atmel Studios template projects for the ATSAMB controller
* Dr. Daniels/EE185 class documentation

# 7 Appendices

If you have any large graphs, tables, or similar that does not directly pertain to the problem but helps support it, include that here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc. PCB testing issues etc. Software bugs etc.