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: 12/6/16

CyMote for CprE 185

Project plan

Contents

[1 Introduction 2](#_Toc468747952)

[1.1 Project statement 2](#_Toc468747953)

[1.2 Purpose 2](#_Toc468747954)

[1.3 Goals 2](#_Toc468747955)

[2 Deliverables 3](#_Toc468747956)

[2.1 CyMote (beta model) 3](#_Toc468747957)

[2.2 PC Wrapper 3](#_Toc468747958)

[2.3 Documentation 3](#_Toc468747959)

[3 Design 4](#_Toc468747960)

[3.1 Previous work/literature 4](#_Toc468747961)

[3.2 Proposed System Block diagram 4](#_Toc468747962)

[3.3 High Level Wiring Diagram 5](#_Toc468747963)

[3.4 Validation 5](#_Toc468747964)

[3.5 System Design 6](#_Toc468747965)

[3.6 Detail Design 6](#_Toc468747966)

[3.6.1 Input/Output Specification 6](#_Toc468747967)

[3.6.2 User Interface Specification 6](#_Toc468747968)

[3.6.3 Hardware/Software Specification 6](#_Toc468747969)

[3.6.4 Prototyping Specification 6](#_Toc468747970)

[4 Project Requirements/Specifications 7](#_Toc468747971)

[4.1 Functional 7](#_Toc468747972)

[4.2 Non-functional 7](#_Toc468747973)

[5 Challenges 7](#_Toc468747974)

[6 Timeline 8](#_Toc468747975)

[6.1 Phase 1 – Alpha Model/Planning 8](#_Toc468747976)

[6.2 Phase 2 – Pinciple Project 8](#_Toc468747977)

[6.2.1 Phase 2a – CyMote: Beta Model 9](#_Toc468747978)

[6.2.2 Phase 2b – Project Code and PC Wrapper 9](#_Toc468747979)

[6.2.3 Phase 2c – Presentation 9](#_Toc468747980)

[7 Conclusion 9](#_Toc468747981)

[8 References 10](#_Toc468747982)

# 1 Introduction

## 1.1 Project statement

CprE 185 currently utilizes a device called the Arduino Esplora. The Esplora 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 Esplora 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 and used 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 with 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 allowed 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 understand. 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 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 ambitious, tech savvy young people, our goal is to complete a project that excites both us and other people

# 2 Deliverables

There are three main deliverable items for this project: (1) A physical beta model of the CyMote for display and presentation, (2) a PC wrapper program that will provide students an interface for interacting with the CyMote during class, and (3) a full set of documentation

## 2.1 CyMote (beta model)

1. This project will deliver, at minimum, one beta model of the CyMote design.
2. This device will contain all necessary control, communication, and I/O hardware and software.
3. It will be capable of collecting data from a fall in the Coover Atrium and transmitting it to a PC over Bluetooth.
4. Its power source will be onboard rechargeable batteries.
5. We will explore the possibility of putting the beta model into a controller shell of some kind.

## 2.2 PC Wrapper

1. The PC wrapper will be on a Windows laptop or desktop PC.
2. It will handle all communication through a Bluetooth transceiver on the laptop to the CyMote itself.
3. It will allow the transmission and display of comma-separated values (CSV) representing the one or more of the following:
	1. Accelerometer values (x,y,z)
	2. Button states
	3. Joystick states
	4. LED states
4. It will use a command line interface and user entered flags to determine how much data is shown.
5. It will be capable of recovering from common failure modes.

## 2.3 Documentation

1. User manual for programming the ATSAMB11 software (For professor/TA)
2. User manual for the PC wrapper (for CprE 185 students)
3. Electrical schematic
4. Device datasheets manual

# 3 Design

## 3.1 Previous work/literature

A previous senior design team attempted the same project to little success. There was a small amount of software written, but nothing substantial that we found useful. The previous team acquired parts and managed to assemble a nonfunctional prototype PCB, of which some of the peripherals were recycled for our project.

## 3.2 Proposed System Block diagram

A basic block diagram of the system can be found in Figure 1 below. The CyMote will talk to the PC over BLE. The PC will have a corresponding BLE dongle, and every PC will have a matching CyMote. The CyMote will keep itself online and able to connect, and it’s on-board software will manage peripherals. When the PC wrapper requests, the CyMote will transmit the requested information: accelerometer data, button data, or joystick data. The connection will also expose the CyMote LEDs to the students if they want to use them.

**Figure 1: Basic System Architecture**

## 3.3 High Level Wiring Diagram

A basic One-Line Diagram of the electrical connections:



**Figure 2: One-Line Diagram**

## 3.4 Validation

The CyMote will need to meet certain benchmarks to be considered functionally complete. Namely, the CyMote will need to be able to send at least 30 data transmissions per second, or one data transmission every 33 milliseconds. Interference from multiple CyMotes transmitting data in an enclosed space must not slow the transmission speeds below this point or decrease their accuracy in a significant way.

The CyMote will also need to meet electrical performance specifications. We will need to validate that the CyMote beta can hold a charge for ≥8 hours. Verification of physical robustness will also be needed. We will have to design a test to drop the CyMote several times and ensure that it does not break or disassemble if placed in a padded container.

## 3.5 System Design

The CyMote needs to be:

* Durable
* Hold a program
* Portable

## 3.6 Detail Design

### 3.6.1 Input/Output Specification

Input is taken from various sensors including a 9° of freedom sensor, joystick, and four buttons**.** The sensor data is then outputted in real time through BLE and is displayed on a PC through a command line interface.

### 3.6.2 User Interface Specification

A PC wrapper program is run from the Windows command line that is supplied with various tags indicating which sensors to use. While running, the program will output the data as comma-separated values to the command line. Various LEDs will be present on the board itself to indicate sensor activity, board power, or errors. The user will have to pair the board with their PC prior to transmission.

### 3.6.3 Hardware/Software Specification

The CyMote will use the ATSAMB11 to control various peripherals. The peripherals used are:

* Joystick w/ button
* 4 buttons
* Power Switch
* 9-Degrees of Freedom sensor (LSM9DS0)
* 3.7V Battery
* RGB LED

The software component will be written in C using Atmel Studio and the ASF project structure.

### 3.6.4 Prototyping Specification

The breadboard prototype must be durable enough to withstand transportation and movement to illicit a meaningful response from the accelerometer. The breadboard prototype will contain all sensors connected in a fashion that will be easily portable to a PCB design. This design will have loaded all ‘Hello World’ programs written to test the various peripherals. Various breakout and development boards will be used to ensure easy interaction with the components during development. Atmel’s Xplained Pro development board provides an early interface with the ATSAMB11. The breadboard model can be powered by an outside source, and code will be loaded onto the board through the Xplained Pro’s onboard debugger.

# 4 Project Requirements/Specifications

## 4.1 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

## 4.2 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. Battery life long enough to last a full day (up to 8 hours) of use
4. Beta model built on a handheld size PCB

# 5 Challenges

We are trying to face the challenges for this project head on. One of the early challenges of this project was striking a definition of done for the device that was realistic, manageable, and would still provide a useful product for the client. We spent a significant amount of time discussing what features we would put in scope, what features we were going to scope out, and which features we could make a better decision on later.

Another challenge we face that the device must not only function properly, it must also adhere to certain other guidelines that we have not had to face as students before. Namely, this device is the first step towards another device that will be mass-produced and used in a classroom setting. We have to keep the production model in mind when we design the alpha and beta models. We have to keep power consumption, communication-bus protocol reliability, and individual part support in mind as we pick parts and assemble the device.

 A third challenge some of us face is the apparent gap in knowledge when it comes to some important concepts relevant to this project. PCB design is something that none of the group has worked with before, and it will take a significant amount of research, learning, and reaching out to get to a point where we can make a functional model.

# 6 Timeline

We will accomplish the project in two main phases divided between semesters. Phase 1 will be a planning and testing phase. We will be constructing an alpha model of the device on a breadboard and creating some simple commands to talk with the device. Phase 2 will be three parts: (1) beta model construction, (2) software/PC wrapper composition, and (3) project presentation. The following is a Gantt chart representation of the overall project:



An important note is that we are specifically planning not to work on any school-sanctioned break. We have removed Thanksgiving break, the winter break, and spring break from our scope for work. The emerging exception to this is that the hardware team will work on the PCB during the winter break.

## 6.1 Phase 1 – Alpha Model/Planning

Phase 1 work ends in dead week of fall semester when we present our Project Documentation. Phase 1 is comparatively flexible to Phase 2. The main objective of Phase 1 is a breadboard model of the device that we can use to simulate the actual function of the final device. The definition of done for this project is what makes this phase so flexible.

The other main objective of this project is the presentation of the project plan. We will be presenting our project as a concept during dead week of the fall semester. This includes design documentation, a project plan, and all necessary visual aids. However, in order to keep this project moving forward, we are planning on starting Phase 2 before Phase 1 is complete.

## 6.2 Phase 2 – Pinciple Project

We have begun work on the beta model already. This overlapped with the alpha model design and construction before and after Thanksgiving break. Because the alpha model tests were successful, we decided for sure to put the hardware on a PCB in early October. The main part of the hardware team’s work will be in the early spring. The software team will have to begin work on the PC wrapper in tandem with the construction of the PCB. This should give us enough time near the end of the semester to get our documentation in order.

### 6.2.1 Phase 2a – CyMote: Beta Model

We have already begun work on the beta model. Because we decided to put the hardware on a PCB instead of a breadboard, we have begun work on the PCB design. Since Nick is from Ames and doesn’t have winter break plans, he will be working on the PCB design and testing all break. With Lee’s help, the hardware team will try to get a PCB cut by the first or second week of spring semester. Since this part of the project is necessary for the PC wrapper to be fully developed, we set the deadline for the week before spring break.

### 6.2.2 Phase 2b – Project Code and PC Wrapper

This part of the project has begun already as well. The software team decided to plan not to work over the winter break. They will hit the ground running with BLE development and begin the PC wrapper program immediately when we return in January. Because we want to ensure our project presentation has the best chance for success, and because we are all graduating in May and know that our schedules will be very full by the end of the year, the deadline for software is three weeks before our presentation. This should give us plenty of room to make mistakes or to cross the finish line a little early if possible.

### 6.2.3 Phase 2c – Presentation

Our final presentation will be in the first week of May, but we would like to have a lot of supporting visual aids, so we plan on beginning preparation for the presentation in mid-February. With all other phases of the project winding down three weeks before our presentation, we should have enough time to create a well thought-out display.

# 7 Conclusion

This project will have two main phases and two main deliverable items. We will build a test alpha model by the end of the fall semester, and we will be able to communicate with the device on basic levels. As we present the project at the end of the fall semester, we will be waist deep in phase two, which will end with the two project deliverables: a sufficiently reliable, robust, and simple CyMote device, and a PC wrapper that allows communication with the device.

The full intention of the beta model of the device and the software is for another group to take our design in the future and develop the production model of the CyMote for use in CprE185 and EE285.

# 8 References

Datasheet for the LSM9DS0:

[http://www.st.com/content/ccc/resource/technical/document/datasheet/ab/2a/3b/45/f0/92/41/73/DM00087365.pdf/files/DM00087365.pdf/jcr:content/translations/en.DM00087365.pdf](http://www.st.com/content/ccc/resource/technical/document/datasheet/ab/2a/3b/45/f0/92/41/73/DM00087365.pdf/files/DM00087365.pdf/jcr%3Acontent/translations/en.DM00087365.pdf)

Datasheet for the ATSAMB11:

<http://www.atmel.com/Images/Atmel-42426-SmartConnect-SAMB11-SOC_Datasheet.pdf>

Datasheet for the Xplained Pro Development Board:

<http://www.atmel.com/Images/Atmel-42664-ATSAMB11-Xplained-Pro_UserGuide.pdf>

Atmel Software Framework (ASF):

<http://asf.atmel.com/docs/latest/>

SparkFun LSM9DS0 Arduino Library:

<https://github.com/sparkfun/SparkFun_LSM9DS0_Arduino_Library/blob/master/src/SFE_LSM9DS0.cpp>

Official description of CprE 185

<http://catalog.iastate.edu/azcourses/cpr_e/>