Embedded Board Interface Labs for Android Mobile Embedded Systems class- CPRE 388

Design Document - May 15-20

Version 2

Angela Borgman, Ben Turbes, Alex Jensen, Ryan Dincher

**Index**

1. Project Definition
2. Goals
3. Deliverables
4. System Level Design
	1. System Components
	2. System Requirements
	3. System Analysis
5. Functional Decomposition
6. Detailed Descriptions
	1. Hardware Specifications
	2. Software Specifications
	3. Lab Specifications
	4. Implementation Problems
	5. Testing

Project Definition

This project was designed to add embedded systems programming to the CprE 388 Android Programming class. As the class currently stands, there is only one lab that uses hardware and it does not require students to do any programming directly to or from this hardware. The goal of this project is to use the Arduino Mega ADK board and some form of attachable hardware in addition to the currently used Nexus 7 tablets to create labs involving embedded systems programming. We are planning on creating 4 labs to be added at the end of the current 388 lab curriculum. These labs will use a 32 x 32 LED board that will be attached to the Arduino Mega ADK and students will be required to manipulate and control this LED board using the Nexus 7 tablet.

Goals

1. We will have a setup design that can be easily replicated so that it can be setup at every lab station easily and replicated in the event to needing to be setup again.
2. The 4 labs we develop will be difficult enough to be beneficial, but will not be more work than the average student can handle in approximately 2-6 hours.
3. The labs created will be interesting and show a practical need for this type of programming.
4. We use hardware that is interesting to students and will be flexible enough that there is room for several labs to be done.

Deliverables

1. A lab setup that can be easily replicated
	1. Arduino Mega ADK and LED board simple to replace if necessary
	2. A setup guide that can be used by either CSG or students to install all necessary software and drivers.
	3. Shield for the Arduino Mega ADK that will allow simple connection between Arduino and LED board.
2. Labs designed for the appropriate amount of time using the Nexus 7 tablet and extra hardware
	1. Labs should take the average student 2-6 hours to complete
	2. Must use both Arduino and some other form of hardware (LED board and heart rate monitor)
	3. Should challenge students each week with new material
	4. Lab write-ups should be clear, easy to follow and give students a good understanding of what they are required to do.

System Level Design

**System Components**

1. Nexus 7 Tablet
	1. Used for all the UI controlling
	2. Sends signals to the Arduino in order to power correct pins for lighting up specific pieces of the LED board.
2. Arduino Board
	1. Connected to the Nexus 7 using a USB to MicroUSB
	2. Connected to LED board using ribbon cable (eventually will be connected with a shield)
	3. Controls signals to LED board and other hardware
3. LED Board
	1. Receives signals from the Arduino board in order to light up correct LEDs with correct colors
4. Heart-Rate Monitor
	1. Currently used in the existing 388 course
	2. Sends the heart rate of the user to the Nexus 7 tablet via bluetooth.

**System Requirements**

* Functional Requirements
	+ Labs must help the student to learn concepts related to embedded systems
	+ Must use the Arduino in some form to communicate with outside hardware
	+ Should require students to learn and use new concepts
	+ See Lab Specifications section for specific specifications
* Non-Functional Requirements
	+ Labs are fun and interesting
	+ Labs are relatable to real world problems
	+ Should take approximately 2-4 hours on average to complete
	+ Labs should be easy to set up

**System Analysis**

1. Scope Definition
	1. LED Board is used to aid in the viewing of lab results for the various labs created. Some viewing of results is also done on the Nexus 7 tablet.
	2. The arduino and Nexus provide a way to examine different aspects of embedded systems
	3. The Goals for the System are that it helps students to learn more about embedded components and smart devices
	4. Expected outcome of the system is that the system will aid in the understanding of low level embedded system concepts
2. Problem Analysis
	1. There are many different problems that may arise as we continue to work on this project, The first problem we encountered was achieving communication between the Nexus and the Arduino. We solved this problem through various tutorials online
	2. Another problem we have encountered was powering the LED board, this problem is being temporarily solved by using the power supplies in the TLA and Senior design lab while we look for a cheaper more portable alternative.
3. Decision Analysis
	1. We have had to make a few decisions about our project at this point, the first being the Board we were going to use. We ended up going with the Arduino due to company reputation, amount of online examples and community support.
	2. Another decision made was what other hardware we wanted to get, we chose an LED board since it provided more opportunities over a simple LED and worked with all of our proposed lab ideas
	3. upcoming decisions will be a power supply for the LED board, connection for the LED board to the Arduino and if we implement a lab using a household appliance. These decisions will affect the ease of building the labs as well as the ability to demonstrate the usefulness of embedded devices in real world applications.

Functional Decomposition

In labs, students will be creating Android USB accessories using the Arduino Mega 2560 board as a controller for various additional hardware components. The LED board is one such component. The Arduino board and the LED board will together be an Android USB accessory. When connected via USB, the Android device will launch the students’ app on the device, allowing data to be read to/from the accessory controlled by the arduino.



Detailed Design

Since CprE 388 is a hybrid of implementing software into hardware for specific applications, the following specifications for the hardware and software will be utilized per person in a lab:

**Hardware Specifications**

1. A computer with:
	1. Windows, Mac, or Linux
	2. A USB port
		1. To communicate with the Arduino device
2. Arduino Mega 2560
	1. Device to interface with the LED board and receive commands from Nexus 7
	2. Communicates with Nexus 7 and any additional hardware via USB
3. Nexus 7
	1. To be programmed by students to talk to the Arduino for the various labs
	2. Will communicate with the arduino via USB
4. RGB LED Panel - 32x32 (COM-12584)
	1. A great visual device for learning as students can see the commands they send with a Nexus 7 to an Arduino for any sort of project they may encounter
5. Power Supply
	1. Needs to run between 3.3-5vdc @ up to 2.5 amps to power the LED panel
	2. Will create these using a Hard Disk external power supply that can be soldered to the power cord that comes with the LED board. This is a cheap and effective solution.

**Software Specifications**

1. Arduino IDE
	1. Will be used by students to program the Arduino
	2. LED board used provides a library with some functions to aid in the communication between the arduino and the board.
2. Eclipse with Android SDK
	1. To be able to write Android Applications that can be sent to the Nexus 7 to interface with the Arduino

**Lab Specifications**

1. Morse Code Lab
2. Provide code for basic application that toggles an LED on the Arduino board (this will allow students to test their setup before beginning to code).
3. Provide outline for Morse Code application including code which converts a string into an array of bytes representing that string in Morse Code.
4. Students will be responsible for designing an interface for this lab, which allows the user to enter a string, convert it to Morse Code, and then transmit that to the Arduino board.
5. Students will be responsible for writing Arduino code to interpret the Morse Code and toggling the LED for the appropriate lengths of time.
6. SketchPad Lab
7. Provide outline for SketchPad application including an user interface which will be used to control individual LEDs on the LED board (this could be a grid representing all the pixels, or some fields which allow you to enter the coordinates of the pixel to update).
8. Provide libraries which students can use to assist in updating LED board.
9. Students will be responsible for using the provided user interface to send signals to the Arduino telling it how to update the LED board.
10. Students will be responsible for updating the LED board based on the signal they sent.
11. Heart Rate Monitor Lab
12. Provide students with an outline for Heart Rate Monitor Lab including the user interface.
13. Students will be responsible for connecting to heart rate monitor and displaying heart rate on Android device.
14. Students will be responsible for sending that heart rate to the Arduino, which will update the LED board to visually display the heart rate (for example, a bar will grow and turn red, the higher the heart rate).
15. Coffee Pot Lab
16. Provide students with coffee pot converted to interface with the Arduino board.
17. Provide students with a basic outline for this lab.
18. Students will be responsible for creating their own user interface and sending a signal to control the coffee pot to the arduino.

**Implementation Problems**

1. For device safety and ease of use, we will have to come up with a “shield” for the Arduino to connect the necessary pins into a connector that sits on top of the Arduino

**Testing**

1. Initial testing is to have a working version of each of our proposed labs. This requires that we test the communication between the Nexus 7 tablet, the Arduino and any other external hardware.
2. Once we have all of our labs drafted we are planning on offering a short 1-credit lab course aimed at students who already have some experience with Android programming. These students will, over the course of 4 weeks, complete the labs we have developed. Once the lab is complete we will require an evaluation form be filled out. This will happen in late March, early April so that once it is complete we have time to make any additional changes to our final versions.