Preliminary Project Description

Bluetooth Controlled Locking Mechanism

Adam Berry-Huff

November 26th, 2012

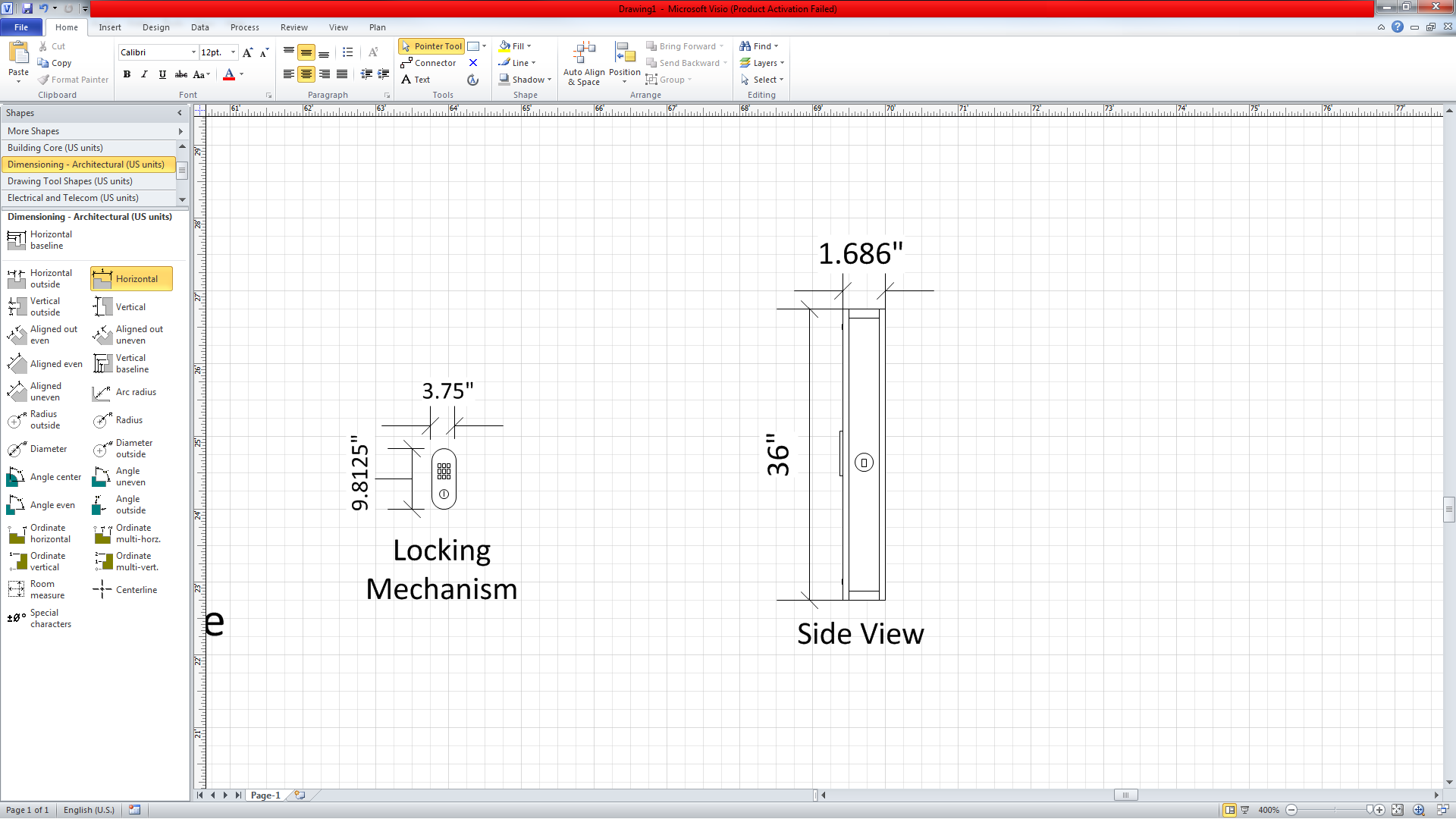
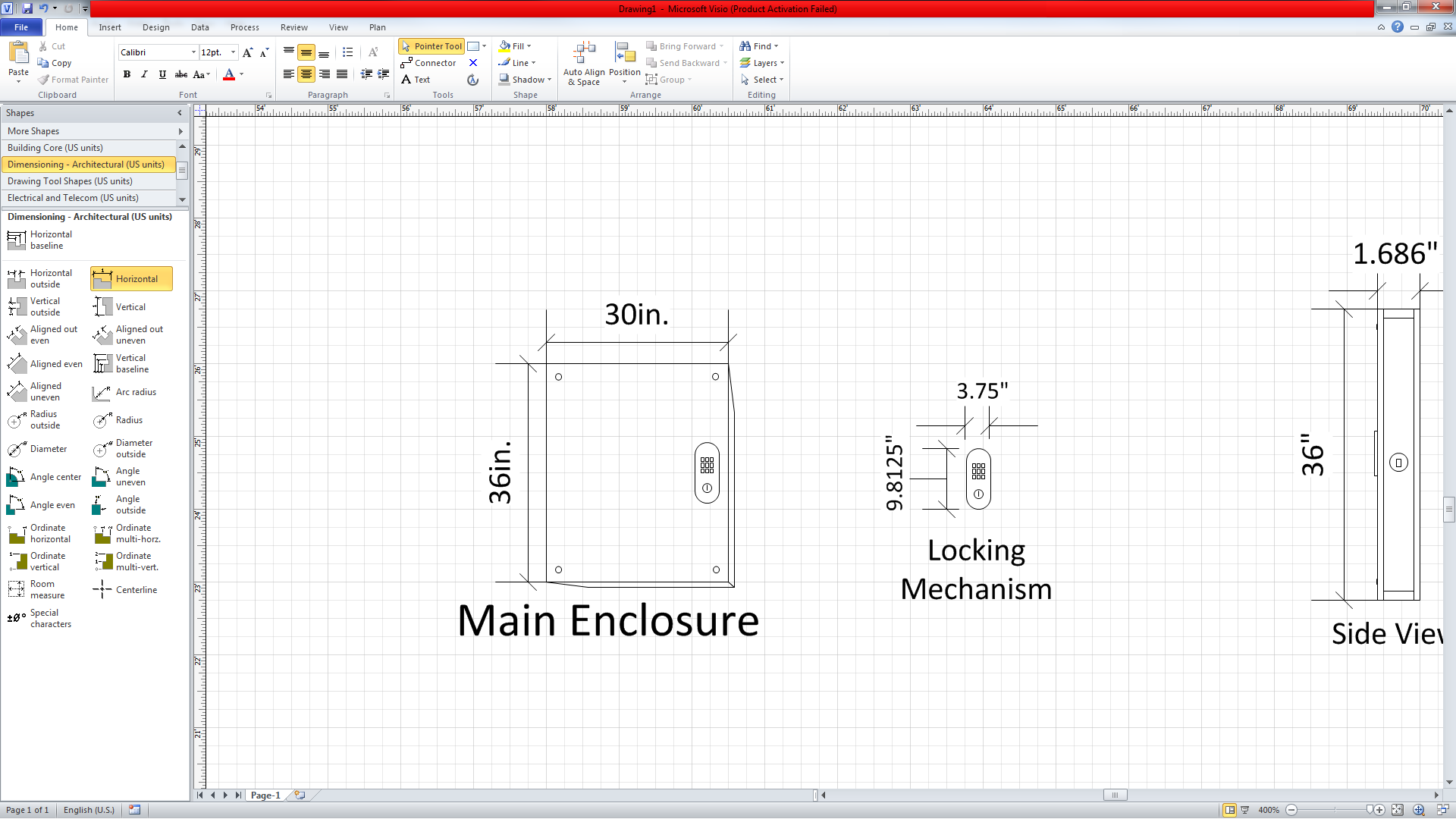
**Functional Description**

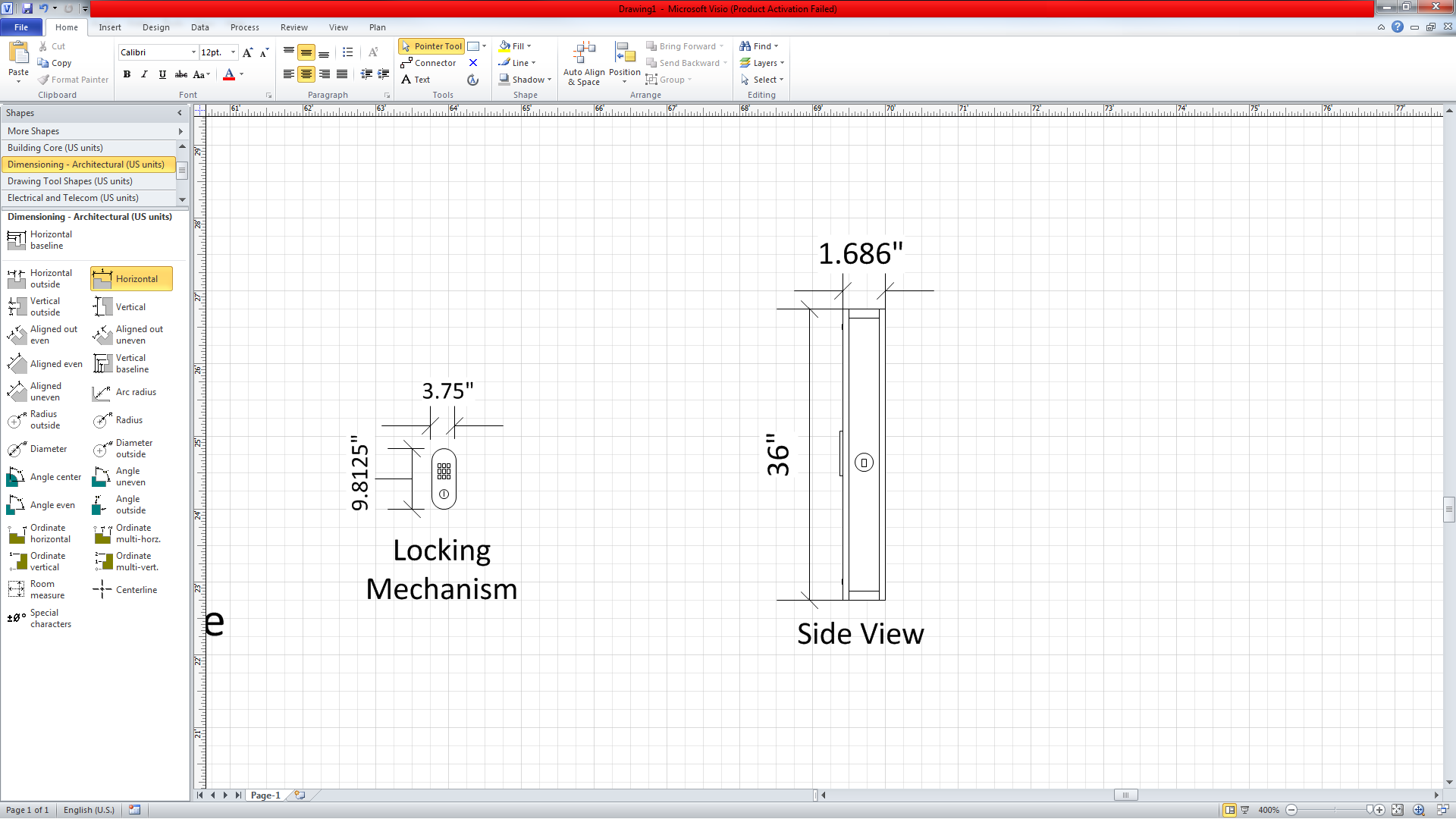
**Introduction**

I’m building a locking mechanism that is controlled by a microcontroller. The microcontroller will be interfaced via Bluetooth by a mobile Android application using Androids native Bluetooth API. Mechanically, when the microcontroller receives the signal from the Bluetooth transmitter, the microcontroller will send an appropriate voltage to the motor that is built into the keyless locking device I am using. This will bypass the logic device built into the keyless entry device. Essentially, I want to build a Bluetooth wireless locking system.

**Description**

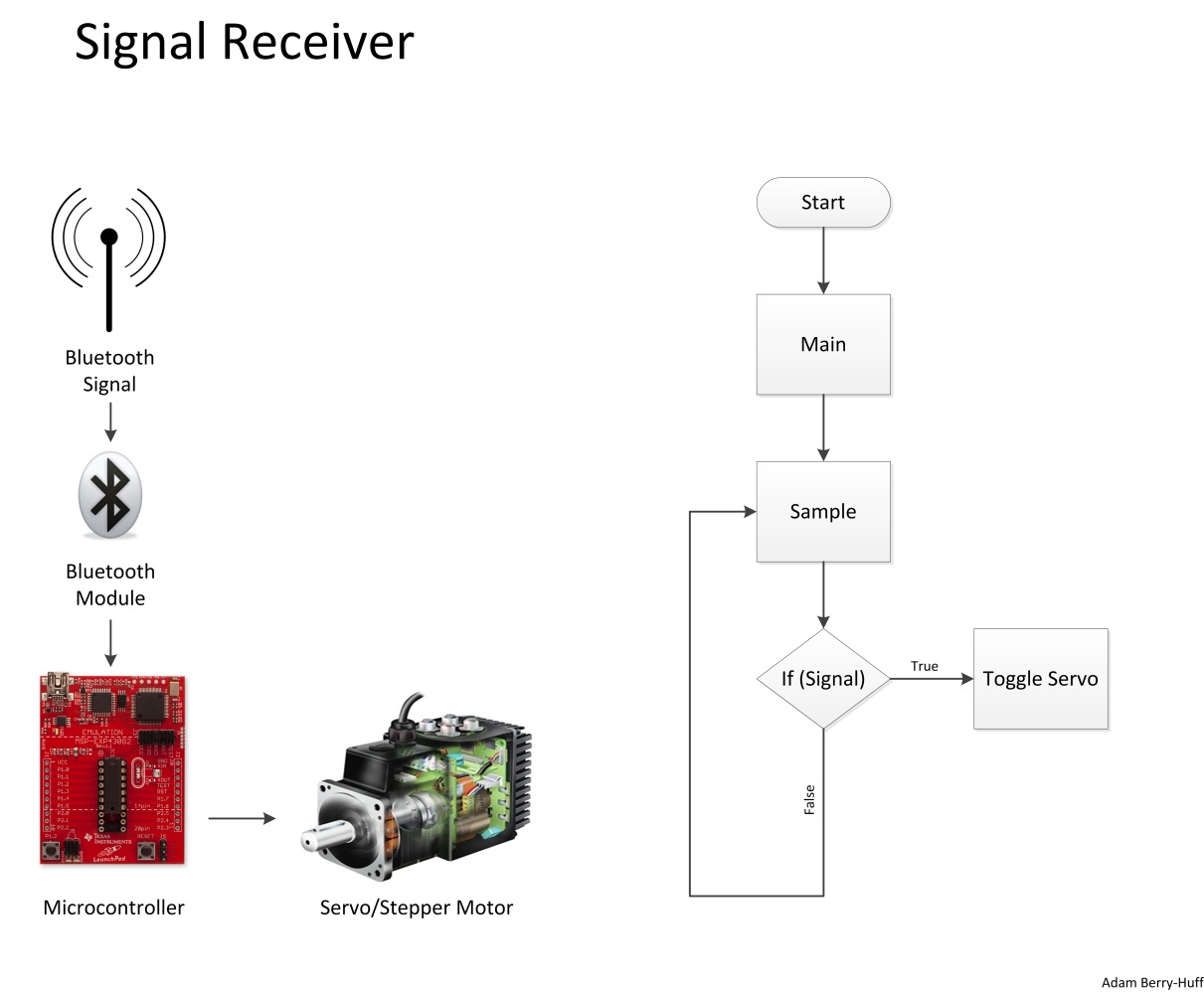
In order to effectively display the functionality of this project, I decided to make a miniature door with a deadbolt but no handle. To display the mechanics of the project as well as create a place to house my microcontroller, I decided to use an acrylic display compiled of two 36” X 30” sheets of acrylic. I am going to add the depth of the door using a wooden frame structure. From the specifications of my locking mechanism, I found that I needed the depth of the display to be anywhere from 1 3/8” to 1 ¾”. By using a 1 ½” board for the frame, I would have a 1.686” total width after adding twice the width of the acrylic sheets (0.093”). This will leave me with a total display depth exactly in the target region for the locking mechanism to fully function. For the wood, I managed to find a 1 ½” by 1 ½” red oak square board that would work perfect for this without needing to be trimmed. The basic dimensions are depicted below.



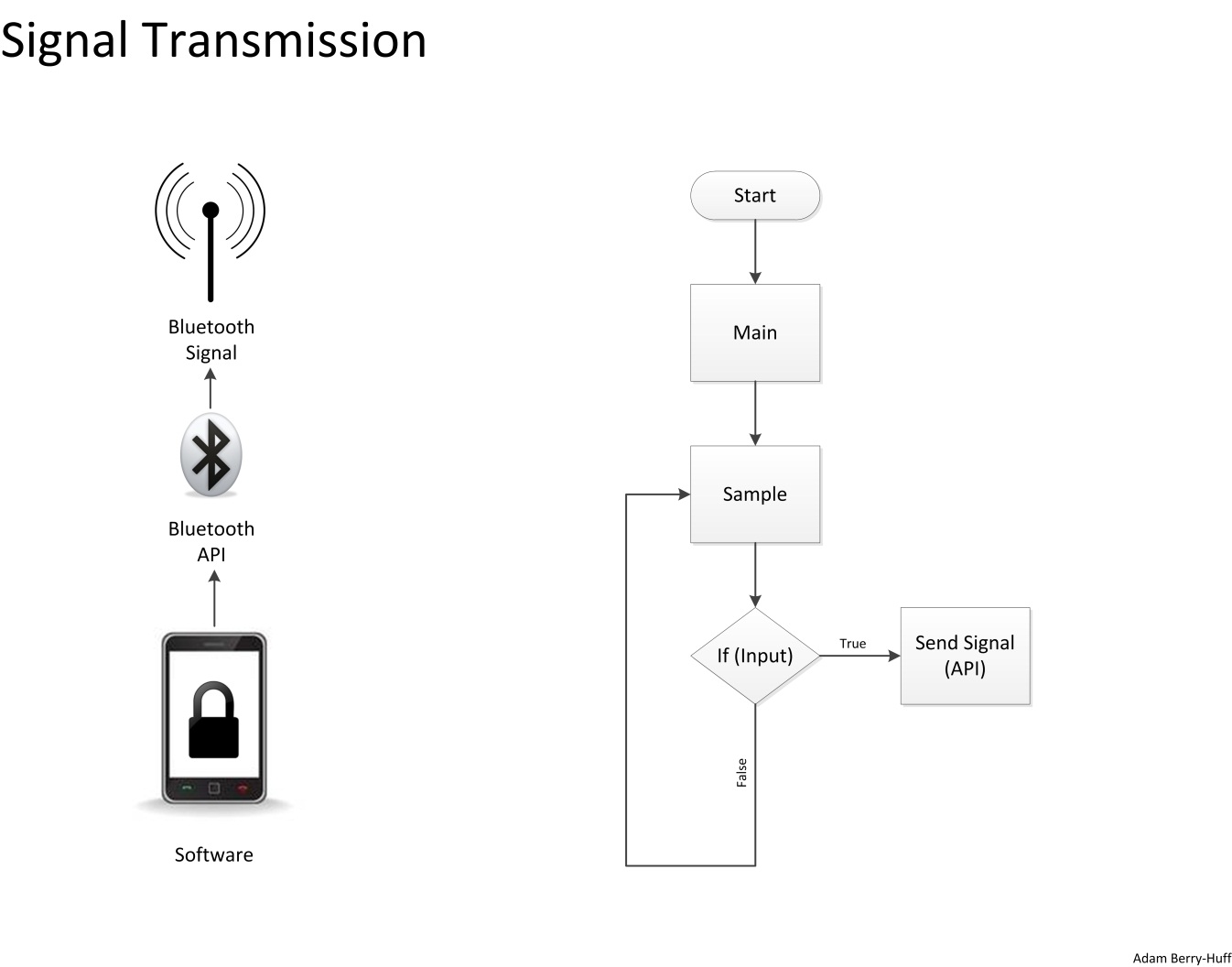


Next, the issue of how the locking mechanism would mount onto the display needed to be addressed. A 1 ½” spade bit was needed for the deadbolt hole. This hole was the smallest concern since it is going to be through wood. Further research reveals that conventional high speed drill bits do not work well on plastics and would probably crack the display. After learning this, a bit needed to be obtained that would work on acrylic. A hole saw bit was needed for the locking mechanism mount and a 3/16” drill bit for the screws to hold the display in place. Some 3/16” screws were needed to accompany this bit and some rubber washers were needed to be sure the screws would not crack the acrylic.

The microcontroller will sit in the bottom of the enclosure and be wired into the motor of the locking mechanism. By using a locking mechanism with a preinstalled motor, the issue of mounting the motor to the deadbolt was resolved. In addition to this, the motor hooked up to the controller can remain installed in the locking mechanism as well so the keypad that is built into the locking mechanism will still work. Even though the PSoC 1 is a large microcontroller with more functionality that needed for this project, I decided to use it since I am very familiar with using the I2C protocol of the PSoC 1 microcontroller. An I2C serial protocol will be used to interface the Bluetooth Module.

****

To transmit the Bluetooth signal for this project, an Android based device with Bluetooth capabilities will be used. This is because a device that will work for this has already been obtained and the development tools needed for transmission will be free. In addition to this, the Java programming language has a larger ease of use and more support over Objective C (used for iOS devices).

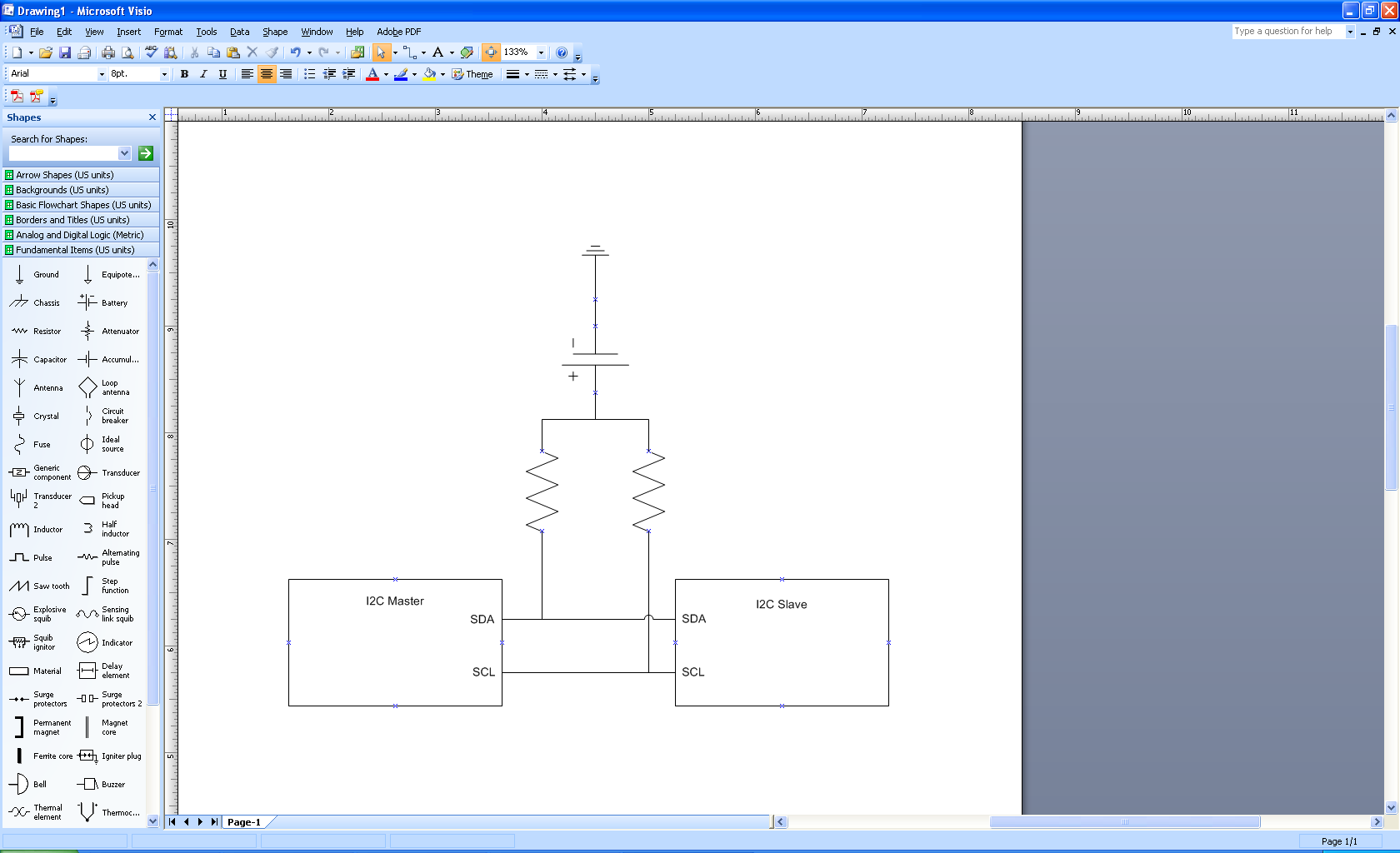
****

**Hardware Description**

|  |  |  |  |
| --- | --- | --- | --- |
| Part | Quantity | Weight (Pounds) | Price |
| PsoCEval CY3214 | 1 | 0.36729 | **$0** |
| Kwikset 907 Powerbolt | 1 | 3.35 | **$69.97** |
| Optix 36”X30” Acryclic Sheet | 2 | 10.24 | **$46.96** |
| Red Oak Square Board | 4 | 9.91 | **$31.28** |
| Energizer MAX AA 4-Pack | 1 | 0.32 | **$3.78** |
| Parani-ESD1000 Bluetooth Module | 1 | 0.013 | **$44.95** |
| Vermont American Hole Saw Set | 1 | \* | **$12.97** |
| Bosch DareDevil Spade Bit | 1 | \* | **$6.17** |
| Milescraft Steel Stubby Bit | 1 | \* | **$3.59** |
| KwikTap Concrete Screws(25) | 1 | 0.08 | **$4.97** |
| Crown Bolt Neoprene Washer | 4 | 0.016 | **$3.32** |

**\*Does not contribute to weight of project.**

The first piece of hardware I picked out was the locking mechanism because the rest of the project is based around a quality functioning locking mechanism. As previously stated, I chose one with an integrated motor for adjusting the deadbolt. This way, I would not have to risk mounting a stepper motor being a serious issue in the well-being of the project. Following this, I decided to use the PSoC 1 development board because I had experience with the board. After doing some research on the Cyprus website, I found that the PSoC does not support USB Bluetooth modules. I knew the PSoC supports serial communication so then decided to use a serial Bluetooth module for my Bluetooth receiver. I then decided to interface my Bluetooth module via I2C as opposed to SPI because I have more experience with the I2C protocol, the I2C protocol required less pins, and my project does not exceed the data transfer limitation of the I2C protocol. Once my parts arrive, I will need to disassemble the locking mechanism and observe the voltages and polarities going through the motor when the lock is activated. Following this, I will wire my PSoC to the motor and practice locking and unlocking the deadbolt using these measured voltages and the voltage divider GPIO on the PSoC Eval Board. Once I have this part of the project completed, I will need to successfully interface the PSoC with the Bluetooth module and attempt to sync the Bluetooth module with the Bluetooth transmitter of my Android Device. I will need to instantiate an infinite loop on the microcontroller to sample the I2C port and trigger the voltage divider to the motor when a signal is received from the serial Bluetooth module. The last thing I will need to do is to develop a user interface Java application and sync it with the Bluetooth transmitter of the Android device using the Java Bluetooth API.

****

I2C Communication

**Software Description**

|  |  |  |
| --- | --- | --- |
| Application | Use | Price |
| Android SDK | Required For Development | **$0** |
| Eclipse (Included in SDK) | Coding Environment | **$0** |
| Java Bluetooth API | Bluetooth Transmission Syncing | **$0** |
| PSoC Designer 4.6 | I2C Sampling & Motor Toggling | **$0** |
|  |  |  |
|  |  |  |

The first code I will write for this will involve toggling the motor using the GPIO and voltage divider on the PSoC board. After this, I will create an infinite loop to sample the I2C port of the PSoC and toggle the motor when a signal is received from the Bluetooth module through the serial communication. Once all this is in working order, I can begin developing from the transmission end. I can develop a user interface on my Android device and trigger the Bluetooth signal transmission using the Java Bluetooth API.

**User Interface Requirements**

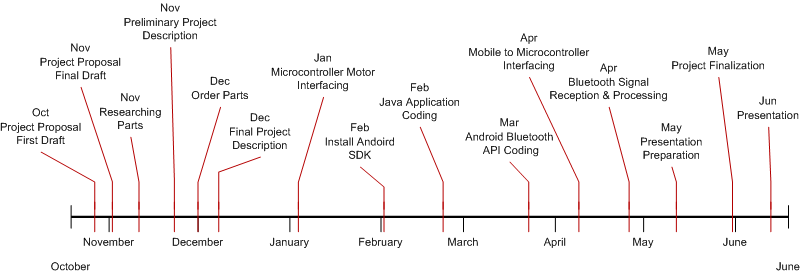
The user interface of the project will come into development last. This will be a simple Java application with an image of a lock. When the user clicks the lock, it will send the Bluetooth signal, unlock the deadbolt, and the image of the lock will become an image of an unlocked lock. After 30 seconds, the lock image will return to the locked position and the deadbolt will automatically relock. If time permits, I may make this auto-lock feature an optional feature.

**Sustainable Design**

This design is an efficient one. It uses very little power and wastes very little with the exception of the fact that the microcontroller is always running. One way to make this an even more efficient would be to use a smaller microcontroller that utilizes even less power. Since this is a prototype, this was not a primary concern.

**Development Plan**

**Development Tasks**

****

**Development Tools**

To accomplish this project successfully, there are quite a few tools I will need. When it comes to software development tools, I will need the Android SDK, the Android Bluetooth API, Eclipse, and PSoC Designer. For hardware development, I will need a multi-meter, a soldering iron, solder, soldering flux, wire, the PSoC mini-prog and USB cable. The receiving unit (microcontroller) will be coded with the min-prog and USB and then powered with a 9-volt AC adapter via the J13 power supply connector on the PSoC unit. For more information on hardware and software development tools, reference the hardware and software development sections.

**Prototype & Materials**

To connect the PSoC microcontroller to the motor, I am going to use heat-shrinking wire wrap, solder, and some wire from the lab. The motor and microcontroller are going to be enclosed in the acrylic mock door described in the first part of the description. My prototype will be on the development board at the time of the presentation. I will have a poster board displaying information about the development process of my project and a monitor displaying my web site and resume. I will also bring in an Android powered tablet computer with Bluetooth capabilities for people to test my project on.

**References**

**Android SDK** - http://developer.android.com/sdk/index.html

**Bluetooth API** - http://developer.android.com/guide/topics/connectivity/bluetooth. html

**PSoC User Guide** - http://www.cypress.com/?rID=37815

**Serial Bluetooth Module** -http://www.lemosint.com/bluetooth/b luetooth\_serial\_adapter\_details.php?itemID=55.

**Locking Mechanism** - http://www.homedepot.com/h\_d1/N-25ecodZ5yc1vZ12l2/R-202290200/h\_d2/ProductDisplay?catalogId=10053&langId=-1&keyword=keyless&storeId=100 51&supe rSkuId=203322569#specifications

**Using Bluetooth With PSoC** - http://www.cypress.com/?id=4&rID=37688

**I2C vs. SPI Pros and Cons** - http://dev.emcelettronica.com/i2c-or-spi-serial-communication-which-one-to-go

**Acrylic Sheets** - http://www.homedepot.com/h\_d1/N-5yc1v/R-202038044/h\_d2/P roductDisplay?catalogId=10053&langId=-1&keyword=plexiglass&storeId=100 51&supe rSkuId=202939033#specifications

## Drilling Into Acrylic - http://www.bertram31.com/proj/tips/drill\_acrylic.htm