Julian Nguyen ***Edward Ryan*** ***Derek Tan***

Team Watchmen

Mid Term Executive Summary of Progress

Our project consists of making a simple wearable device that is completely enclosed and can act as a motion controller. The watch will contain an MSP430F1611 and be able to connect to either an iOS or Android device via Bluetooth. The main functionality of the watch will be gesture-based controls, a feedback system that notifies the user when an action is registered, and a rechargeable battery that allows the watch to last throughout the day. To achieve this, we will require 4 main component subsystems, the MCU/communications module (i.e. MSP430 and Bluetooth adapter), the feedback system (i.e. vibration motor and LED display), an actuator input (pushbutton), and a power module (i.e. battery and charging module). This watch will also be designed in a form factor small enough to fit comfortably on a user's arm during use.

Our project is a wearable that has the aforementioned functionality of having thorough motion/orientation tracking, and the ability to implement software defined gesture controls by processing the sensor data. Our project will work by integrating the various subsystems listed below and soldering all of those subsystems onto a specialized small PCB that we design. The PCB will be powered by a lithium ion battery which is chargeable by wireless inductive charging. The software/firmware will be developed to allow the user to pair their personal iOS or Android device to the watch. The figure below is a preliminary block diagram for all of the subsystems and constituent parts of the watch, including the specific components we plan to use.

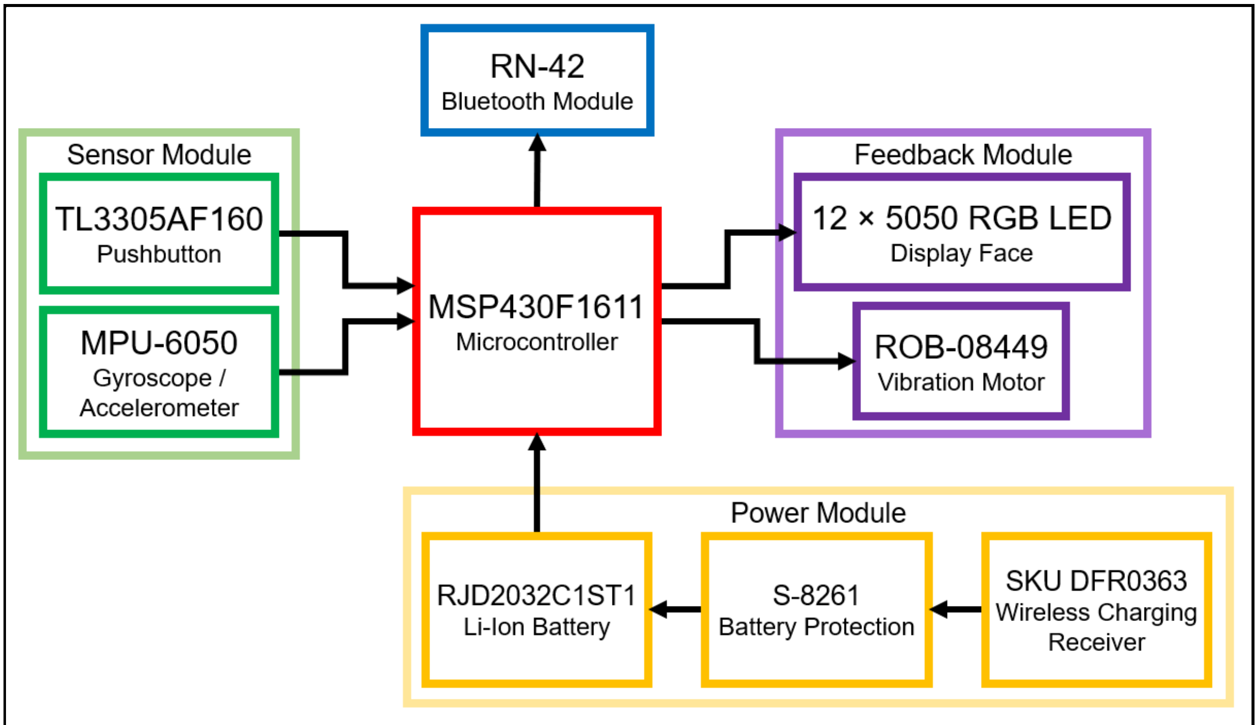

Figure 1.1: Systems and Subsystems

One of the biggest problems that we face is the power module system as well as the ability to fit all of the components into a wearable form factor. We expect to create a functional wearable with the ability to connect to a smartphone via Bluetooth. It should have enough battery power to last throughout a regular day's use. We do not anticipate using any external materials beyond those provided by the ECE lab. The project's main subsystems are laid out in Figure 1.1 above. Those 4 systems include the sensors (green), processor and communications (red/blue), power (yellow), and feedback module (purple).

This project will call upon our knowledge of all the ECE Fundamentals classes, mainly in the form of circuits and PCB design. It will also call upon all our knowledge from our previous coding classes, mainly CS 2150 and ECE 3430 for their teachings in C/C++ and lower/embedded systems.

This project is complementing our other classes in a few ways. Our entire capstone team is actually also a team for the eCommerce class project. In eCommerce, we decided to make our web project into a luxury watch rental business. This was done partially because the early research we did into smart watches for capstone meant that we already had the basic ideas fresh in memory. Besides that, the capstone project requires a lot of general problem solving, systematic thinking, and programming. These skills are used most -if not all- of our classes at this point.

While all team members will likely find themselves doing a bit of everything, our specialization is as follows:

Edward is the primary programmer in the project, and the secondary hardware debugger/tester.

Julian is the primary person working on the Multisim schematic/simulations and Ultiboard/PCB layout. He will primarily focus on anything pertaining to hardware. He will also be the secondary 3D modeler.

Derek is the primary 3D modeler, and the secondary programmer in the project.

Timeline

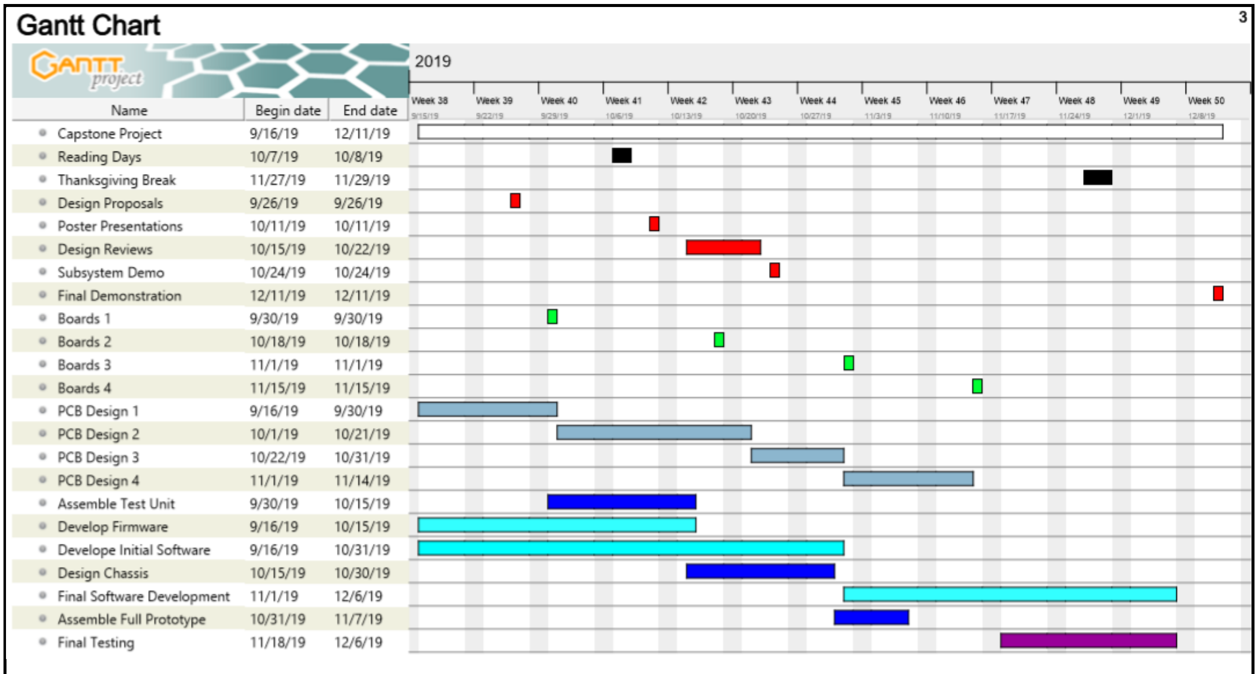

Figure 1.2: Gantt Chart

Schematic and PCB

The first draft circuit schematic and PCB layout (Figure 1.3) have been created for preliminary testing and analysis. The PCB layout was decided to be 40 mm in diameter to fit within a typical watch face form factor for comfortable usage. The manufactured PCB provided a size reference to test the sample watch chassis. Routing for the MSP430 microcontroller and related components have been designed to optimize space for the chosen form factor. There is confidence/finality in most if not all digital pin connections between integrated chips, as many interfaces involve the MSP430 microcontroller's generic I/O pins which can be adapted to different needs through software. For the purpose of testing and development, we have ordered testing/header boards for the MSP430 and gyroscope/accelerometer chips until they are ready to be mounted to the final PCB alone. This has required a special JTAG interface with the development MSP430 in order to flash programs for preliminary software testing. We are considering adding dedicated testing pins/throughhole connections to the PCB in order to more easily load our program onto the final project board.

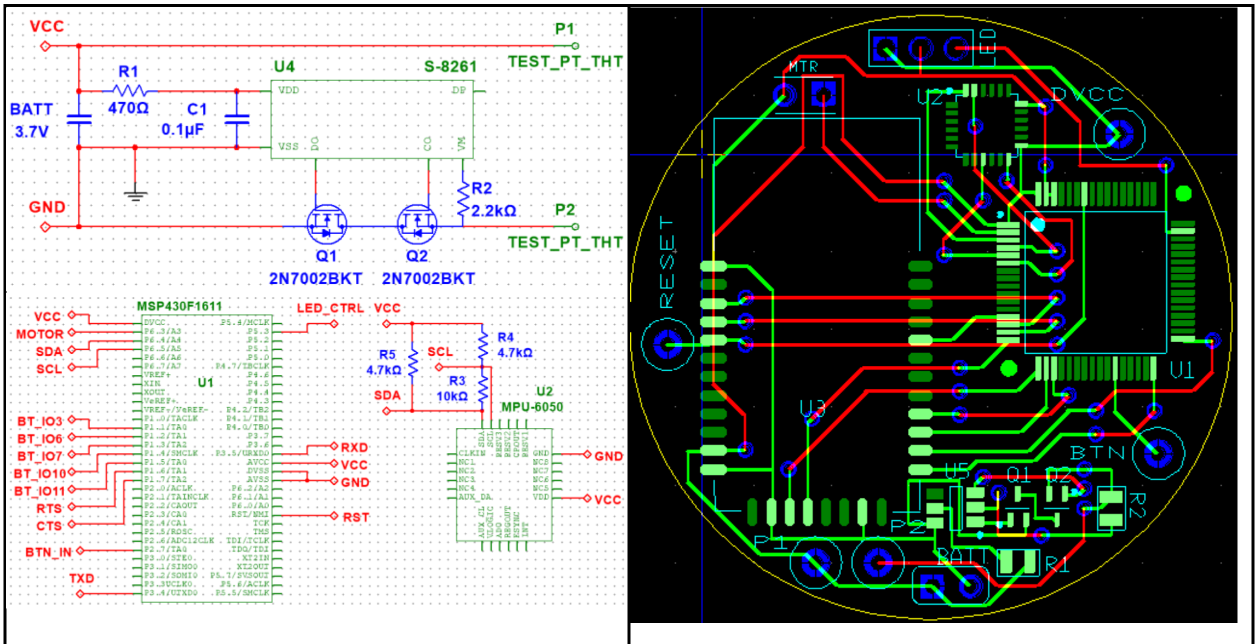

Figure 1.3: Partial Circuit Schematic (left) and PCB Layout (right)

3D Modeling

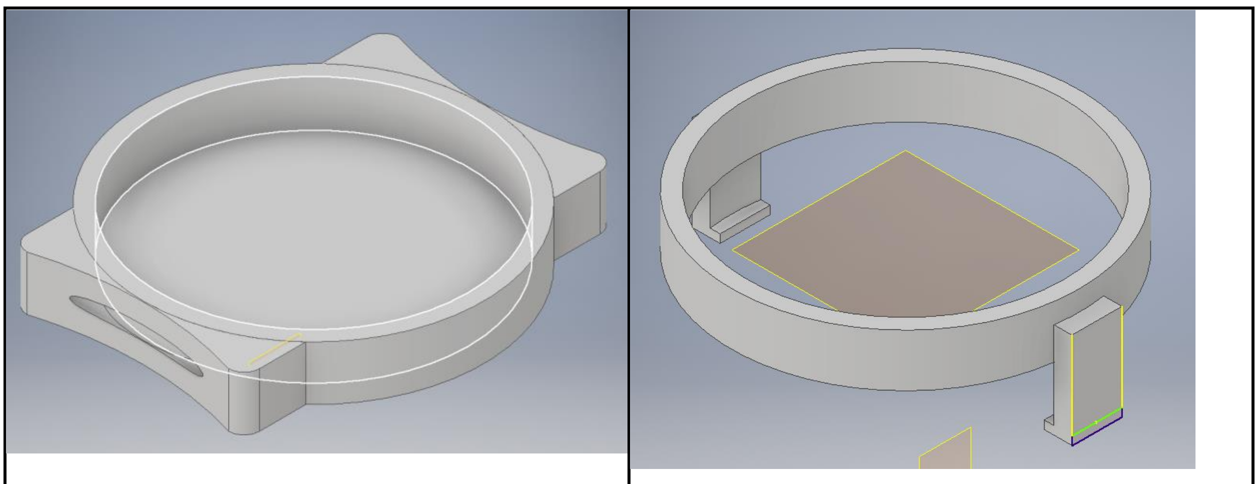

Figure 1.4: Watch Bottom (left) and Top (right) Chassis Version 1

The first iteration of the chassis (Figure 1.4) proved somewhat erroneous as the top chassis of the display did not conform to the size of the LED display and hence did not fit snugly into the chassis. Also the design of the clip mechanism was very flimsy and the mechanism broke off. The design also did not encompass all the open components and hence some of the circuitry was exposed and not completely enclosed. The bottom part of the bottom chassis was designed so that it would be 1 mm thick with print density at 15%. This design decision proved to be correct when it came to testing the wireless charging module. The 1 mm thick plastic was tested and confirmed to be thin enough between the receiver and transmitter of the charger. Virtual bench results displayed that the voltage increased to 5 V and the LED turned on in our test circuit (see Figure 1.6).

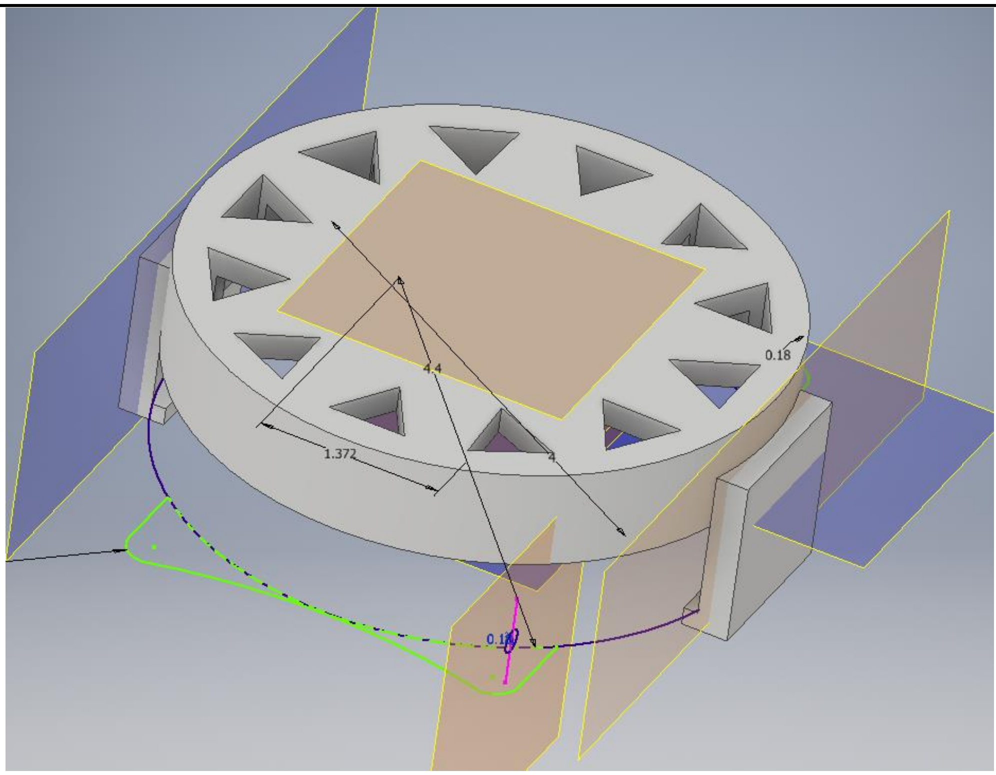

Figure 1.5: Top Chassis Version 2

The second version (Figure 1.5) of the top chassis sought to remedy this and the watch is now completely enclosed when the top and bottom chassis are clipped to each other. The clips were made thicker to make it sturdier and cut outs were made so that LED light can show through the plastic. The tolerances of the 3D printing (+/–.03 mm) were taken into account the second time around. The top chassis diameter was also designed to be 37.2 mm so that the LED display fit snugly.

Power Module

Basic testing was done to verify the wireless charging module. Powering the transceiver with a 5 V source resulted in successful transfer to the receiver with contact/close proximity, even with the test chassis cover between. This was confirmed first by powering an LED (Figure 1.6) from the receiver end and later through connecting with the greater power module on the

PCB. This ensures that the inductive charging device is a viable method for delivering power to the battery, allowing continued testing for over/under-voltage and over-discharge of the battery through these means.

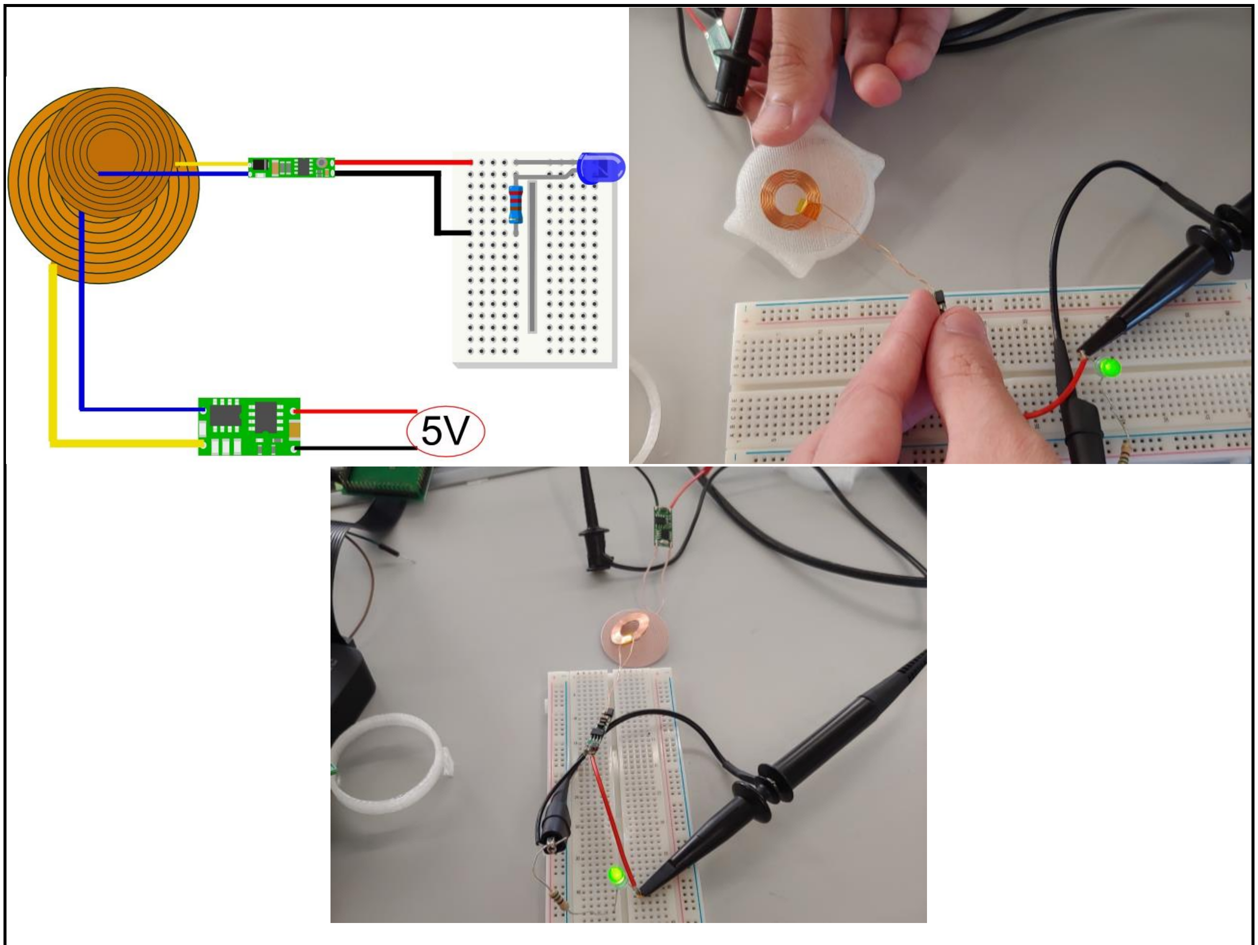

Figure 1.6: Wireless Charging Module Testing

Header Board Testing

Edward wrote some basic code in Code Composer, and loaded it onto the test MSP430F1611 w/header board. This code was simply used to verify that the test MSP430 is functioning normally, that the code composer environment is set up correctly, and that the I/O pins we chose for I2C data receiving are connected properly. The code being run in the screen snip below is just showing that the pin we chose for SCL is outputting a square wave at 3.3 volts and an expected (though currently insignificant) frequency, and that the pin we chose for SDA is outputting an identical but shifted square wave.

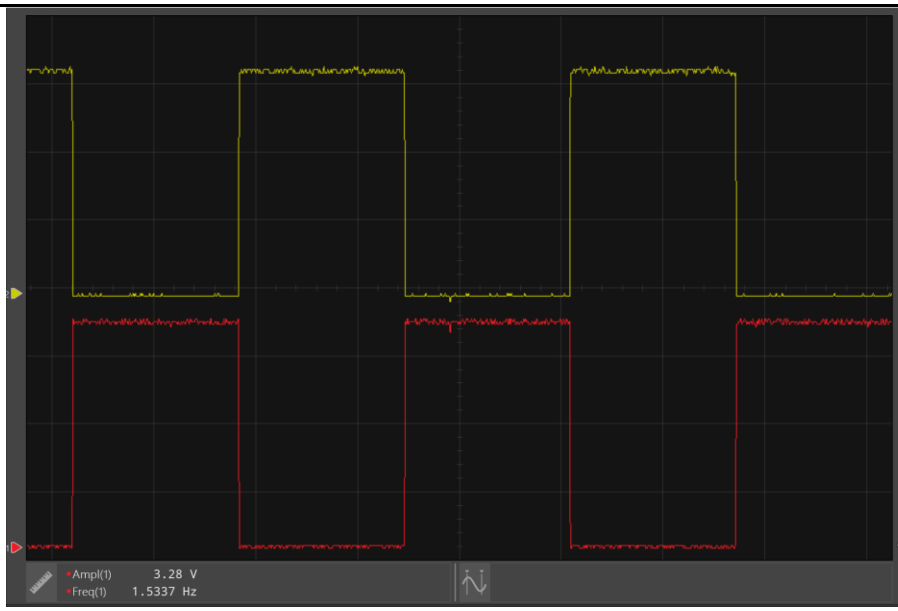

Figure 1.7: Testing MSP430 Header Board Pins

We realized that the MSP430F1611 is much more difficult to use with I2C data transfers than many other MSP430 variants. We briefly considered switching processors, but we've already invested in the F1611, and we think we can write a software implementation of I2C using general I/O pins. We also realized SPI would probably be slightly easier for us than I2C, but we've realized the motion sensor chip we ordered, the MPU 6050, only supports I2C. The MPU 6000 actually supports I2C in addition to SPI, so that may have been a better option. However, we plan to work with what we have.

Team Member Contributions :

Julian Nguyen

Julian has contributed so far by designing the first draft circuit schematic and PCB layout (Figure 1.3), as well as picking the majority of components and I/O pins to be used. He has designated parts/values and design for the battery protection circuit, as referenced for the chosen battery protector IC. Components had to be carefully chosen, such that they were both compatible and small enough to fit within the ~40 mm diameter watch form factor. Two components (gyroscope/accelerometer and Bluetooth module) required manual Ultiboard part layout design, as their chips had non-standard package footprints. This required careful measurement for chip dimensions and pin size/spacing.

Components and pin connections were chosen in order to optimize layout and routing to be compact to fit within the 40 mm diameter PCB. Several components (charging receiver, battery, vibration motor, LED ring, pushbutton) were given generic test pins/throughhole footprints, since they would be mounted elsewhere from the PCB. For example, the battery is intended to rest under the PCB. Preliminary testing was done on the wireless charging module, verifying the flow of charge from the transceiver through the protection circuit.

Derek Tan

Derek has printed out a preliminary 3D chassis for the watch. The chassis consists of a top and bottom enclosure. The top part of the chassis is intended to house the LED display ring (watch face), while the bottom part will house the PCB and all the other free components. The LED display attaches to the top chassis, which is able to attach onto the bottom chassis. The top and bottom chassis are designed in a way such that they can be easily attached/detached by the user. The dimensions of the chassis had to be designed to fit the PCB diameter of 40 mm and LED display of 37 mm.

The height for the entire chassis was calculated to be 7 mm at a minimum by adding the battery thickness (1 mm), PCB thickness (1 mm), and LED Display (5 mm). This fits with the overall flow of the project as the chassis has to be designed so that all the components work and fit with it. For example, the wireless charging receiver was tested with the bottom chassis to make sure that the induction charge could flow through it, as too thick of a chassis would inhibit the charging.

Edward Ryan

Came up with the idea to have the original smart watch design be able to scroll through presentation slides with gestures, though, at the time, it was thought of as a potential extra feature.

Upon input from professor Powell, helped change plans from an all out smartwatch, to a reduced functionality gesture control watch, now focusing in on special gesture features such as being used as a gesture control presentation tool.

Found the RN-42 bluetooth module and MPU-6050 motion sensors to replace the original microprocessor’s embedded bluetooth functionality, and the original 3 DOF only motion sensor. If gestures are going to be our trick, we would need at least a 6 DOF motion sensor.

Started software development on the MSP430 header board and testing on the MPU-6050 gyroscope/accelerometer. After research and testing on both chips, has decided to implement a software solution to I2C data transfer between these devices. The work referenced around figure 1.7.

Summary and Conclusions

The majority of parts needed for the project have arrived, and we have enough components to begin testing and development of major subsystems. We have successfully verified the wireless charging module, confirming our fundamental idea for power delivery to the system. Once the remaining components involved in the power module arrive, we will be able to continue to full battery integration and hopefully finalize the design for power management of our project.

We have started testing the interface with the gyroscope/accelerometer IC for the upcoming subsystem demo. We are currently in the process of developing code to configure generic MSP430F1611 I/O pins for a software-defined I2C protocol. This will provide a continuous two-way stream of data with the gyroscope/accelerometer and the MSP430. Once this serial interface is established, development can continue towards processing key data into motion-control related behavior. We are confident that the schematic is close to final, considering many interfaces are dynamic/weighted towards software-side, and do not plan on dramatically changing the schematic or layout. Therefore, we can assume the same PCB dimensions and thus will likely not need to rethink major design guidelines regarding form factor and build layout. As it stands, the project remains on track to be a portable/wearable device capable of tracking and interpreting motion.