Deployable Prototype Documentation

April 29, 2019

Data Transmission Infrastructure (Team 7)

Bryan Strong
Cody Rider
David Rabago
Jay Tandel

Professor Neal Levine

Table of Contents

List of Figures

List of Tables

## ELEVATOR PITCH

Develop an infrastructure to transmit large amount of raw sensor data for analysis in order to monitor and improve post-surgical patient recovery.

## EXECUTIVE SUMMARY

Orthopedic surgeons and physical therapists need more data from patient's pre and post-surgery to assist in improving the progression of rehabilitation and to gather insight into the effectiveness of orthopedic surgeries.

Orthopedic Surgeons could benefit from more patient data as it would allow them to monitor the maximum achieved progress of a patient post-surgery. This would allow them to analyze the surgery and implants more effectively.

With specific interest in sports monitoring, gait analysis and inertial sensors are being embedded to form one device with several sensors. These sensors are called IMUs (Inertial Measurement Units). Although IMU's do not have the best accuracy profiles there are several sensor fusion algorithms that increase system reliability.

Consensus Orthopedics is attempting to tackle this problem with their product TracPatch. TracPatch allows a patient's physician to view data, interact with the patient, and update physical therapy routines that better suit the client based on their progress. The entire concept of the Tracpatch plays into the up and coming trend of internet of things (IOT).

What we intend to do as a team is work with Consensus Orthopedics to develop a similar but new product that gathers significantly more data and provides additional functionality. From a hardware aspect we will be working with two IMU's that each have 9 degrees of freedom for a total of 18 degrees of freedom.

We will be using 5.0 Bluetooth modules to assist in the current bottlenecking problem that Consensus is having with some of their current hardware and software.

Our goal is to have two prototypes, each one consisting of one IMU and one 5.0 Bluetooth Module. When we can get the prototypes working our next step will be to incorporate the software we develop into Consensus Orthopedics' current TracPatch hardware.

We will develop the infrastructure to transmit large amounts of sensor data for analysis to monitor and improve post-surgical patient recovery. One deliverable our team is providing is raw data transmitted continuously. With a large volume of raw data transmitted continuously, as opposed to processed data points passed selectively, more metrics about a patient's activities can be developed and provide more insight for practitioners.

Our efforts can offer patients a faster discharge and provide practitioners insight on their patients' recovery from surgery to present day and allow early detection of complications. In addition, practitioners would be able to remotely access patient data for all their patients which saves time and resources.

***Abstract* - What we intend to do as a team is work with Consensus Orthopedics to develop a similar but new product that gathers significantly more data and provides additional functionality. Our goal is to obtain data from the sensors they used, transmit the raw/unprocessed data, transmit over a larger pipeline, and store it in an easily parsed document. This document focuses on the design approach, risk assessment, product status, and future market forecast.**

***Keywords* - Metric, Ambulatory, LMWH, TKA, IMU, ROM, CSV, IOT, BLE**

## I. INTRODUCTION

Orthopedic Surgery is one of the most common surgeries in the United states and it has a huge social and economic impact. Within the discipline of Orthopedics and Physical therapy, monitoring a patient post-surgery is essential to a healthy recovery and a lack thereof could diminish the effectiveness of the surgery itself. Medical complications can occur within days of a patient's release from a hospital which can be days before they are seen by a physician.

Traditionally, the means of acquiring patient data consists of frequent in-hospital visits and monitoring the patient there. This puts limits on both the practitioner and the patient. It is not practical for every patient to be seen every day. Yet within a few days, complications can occur. It is essentially a bottleneck where there is not enough time or doctors to continuously check on every patient. It is not practical either for a doctor to monitor a patient in the event that a complication might occur.

Many practitioners see a much-needed improvement regarding patient monitoring. They all have a mutual consensus that patients need to be monitored more frequently and more effectively. Advances in medical technology are growing, but there is no widespread solution to solving this problem that is accepted by all practitioners.

Consensus Orthopedics is attempting to tackle this problem with their product TracPatch. TracPatch allows a patient's physician to view data, interact with the patient, and update physical therapy routines that better suit the client based on their progress. They are able to measure a patient's range of motion through a serious of repetitious exercises. The device senses movement does a calculation and transmits this data over the internet to the cloud.

The product had increased its IMU's degrees of freedom and Consensus Orthopedics wanted to convert the data transmission from processed to raw. This created a huge stress on the Bluetooth module they have on their current version of TracPatch and noticed bottlenecking. A hardware replacement for the TracPatch may prove to be the best option for the product in order to accommodate the huge increase in data traffic.

The throughput is the main problem Consensus Orthopedics is facing. They are looking to introduce additional feature to the services they provide their practitioners and patients. To do so, they need to collect more data over the course of a patients' day. They were not able to transmit raw sensor data continuously. After meeting with Consensus Orthopedics and discussing the limitations of their product and the contributions we could offer, our sponsorship was established.

What we intend to do as a team is work with Consensus Orthopedics to develop a similar but new product that gathers significantly more data and provides additional functionality.

We're providing Orthopedic Surgeons and Physical Therapists more metrics about a patient's post-surgery behavior to improve patient care. To do this, we are working with Consensus Orthopedics to improve their existing monitor device-TracPatch-to increase the volume of data transmitted.

One deliverable our team is providing is raw data transmitted continuously. With a large volume of raw data transmitted continuously, as opposed to processed data points passed selectively, more metrics about a patient's activities can be developed and provide more insight for practitioners.

We will develop the infrastructure to transmit large amounts of sensor data for analysis in order to monitor and improve post-surgical patient recovery.

Our efforts can offer patients a faster discharge and provide practitioners insight on their patients' recovery from surgery to present day and allow early detection of complications.  In addition, practitioners would be able to remotely access patient data for all their patients which saves time and resources.

Our team will develop an infrastructure to transmit large amount of raw sensor data for analysis in order to monitor and improve post-surgical patient recovery. We will tackle this issue by creating two prototypes each consisting an IMU board and BLE module. IMU board we will be incorporating in our prototype is DK-20948, which is a ICM-20948 IMU with a 9-axis motion sensor composed of 3-axis accelerometers, magnetometers and gyroscopes. DK-20948 will be used to collect the data from the patients which then will be transmitted through the Bluetooth 5.0 module. Bluetooth module used in the prototype is Taiyo Yuden's EKSHSNZWZ evaluation kit. This kit uses the EYSHSNZWZ Bluetooth low energy modules consumes 80% less power than the nRF51 series by using a Nordic Semiconductor nRF52832 SoC.

Second side of our design is the receiver side, once the collected data is transmitted it will be stored in a Microsoft surface go (tablet) in a CSV file. On this Surface go tablet we will build a GUI application that will allow us to view the data that we are collecting, transmitting and storing. This received data will be roughly 51 MB of data per file. There will be two CSV files each day, we will sync this 102 MB of data based on their time-stamped data in timely manner.

Once we are successful in creating a working prototype it will enable aggregating this raw data into large pools of data. A machine learning infrastructure can be developed to analyze the specifics. Once patterns can be pinpointed, the user will no longer be required to input what exercise they are doing. Based on their raw data output from the sensors, a profile can be put together for the individual that charts their type of activity, along with time and effort put into that action. This can be analyzed on a larger scale from the practitioner perspective and determine if the patient is performing as expected post-surgery.

Our team's sponsor Consensus Orthopedics is currently only able to utilize their product for the post knee surgery rehabilitation, our prototype will allow them to expand their product use to post shoulder, hip and even spine surgeries.

## II. SOCIETAL PROBLEM

Currently physical therapists create routines for patients to follow but can only monitor the progress of the patients when they physically come to visit the office. Physical therapy could be greatly improved if physical therapists were able to somehow monitor the activities and progress of the patients on a day to day or even minute by minute basis.

Orthopedic Surgeons could also benefit from more data as it would allow them to monitor the maximum achieved progress of a patient post-surgery. This would allow them to analyze the surgery and implants more effectively.

There is an increasing amount of scientific and commercial solutions for health monitoring that have been prompted by new MEMS (Micro-Electro-Mechanical Systems) based sensors. These new sensors are low cost, small and are good for three-dimensional space orientation. With specific interest in sports monitoring, gait analysis and inertial sensors are being embedded to form one device with several sensors. These sensors are called IMUs (Inertial Measurement Units). IMUs are being utilized more frequently even though they tend to be inaccurate at measuring body orientation or movement profiles. Although IMU's do not have the best accuracy profiles there are several sensor fusion algorithms that increase system reliability.

BLE (Bluetooth Low Energy) devices are being used to transfer the acquired data from the IMU sensors to many different platforms including Android tablets and Personal computers.

Consensus Orthopedics is attempting to tackle this problem with their product TracPatch. TracPatch currently works with an IMU that has 6 degrees of freedom and is only used in applications following knee surgeries. TracPatch allows a patient's physician to monitor data such as ROM (range of motion), number of steps, pain levels and inactivity. It also allows the physician to interact with the patient and update physical therapy routines that better suit the client based on their progress. Below is an image [Figure 1 [7]] of Consensus Orthopedics' current product.


*Figure 1 [7]*

The entire concept of the Tracpatch plays into the up and coming trend of internet of things (IOT). We are seeing an exponential increase in the number of devices being connected to the internet. It started with PC's and has worked its way from cellphones and tablets to mini devices that we wear including smart watches and medical devices that monitor us throughout the day.

Reports are showing that there were 22.9 billion devices connected to the internet in 2016 and is expected to reach 50 billion by 2020. Refer to table {Table 1 [2]] for visual.

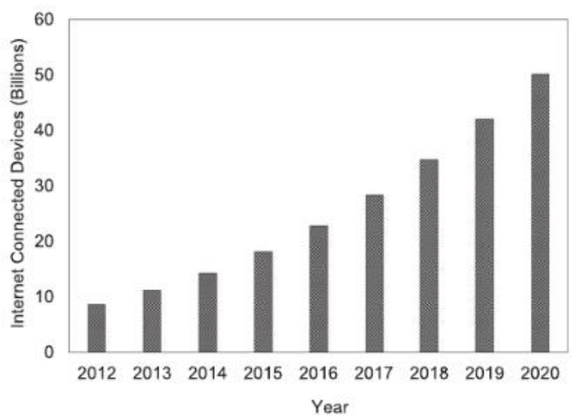


*Table 1 [2]*

This comes hand in hand with the increase in data that is being transmitted from devices. In the table [Table 2 [3]] we can see how explosive the data traffic is becoming now and in the near future.

| | # Devices | Data Traffic |
|---|---|---|
| Year | Billion | Exabytes |
| 2014 | 20 | 44 |
| 2016 | 28 | 300 |
| 2018 | 36 | 789 |
| 2020 | 50 | 1029 |

*Table 2 [3]*

A large portion of the data being collected is from sensors. This is no exception for the TracPatch. Like every other data collection, the data can be used for improvements for the product and inevitably the consumer. For Consensus Orthopedics, this data can be aggregated into large pools of data. A machine learning infrastructure can be developed to analyze the specifics. Once patterns can be pinpointed, the user will no longer be required to input what exercise they are doing. Based on their raw data output from the sensors, a profile can be put together for the individual that charts their type of activity, along with time and effort put into that action. This can be analyzed on a larger scale from the practitioner perspective and determine if the patient is performing as expected post-surgery. This information can also be expanded to understand therapy routines and different tactics for surgery. Practitioners can analyze data with filters, comparing patient results based on routines and surgery. This will give a better understanding of how patients and their affected areas are responding to their plans. They can adjust the prescribed plan in order to increase patient recovery.

For this infrastructure to be put in place, we need to extract the data. Our unique approach will allow such an infrastructure to be functional. With more sensors being placed on devices and requirements of precision and fast sampling, we run the risk of clogging up data transmission. For the TracPatch, this issue hits home. The product had increased its IMU's degrees of freedom and Consensus Orthopedics wanted to convert the data transmission from processed to raw. This created a huge stress on the Bluetooth module they have on their current version of TrackPatch and noticed bottlenecking. The Bluetooth module as of now implements the BLE 4 standard which has a transmit rate of 1Mbps (megabit per second). Fortunately, a new variation, BLE 5, has come to the market with the ability to double the transmission rate. As we intend to move towards this version of Bluetooth, we provide a much-needed change to increase performance. Refer to table [Table 3 [4]] for version comparison.

| Specifications or features | Bluetooth 5 | Bluetooth 4.2 |
|---|---|---|
| Speed | 2x the data rate, Supports about 2 Mbps | Supports about 1 Mbps |
| Range | 4x the range, Supports 40 meters in indoor environment | Supports 10 meters in indoor |
| Power Requirement | Low | High |
| Message Capacity | Large message capacity, About 255 bytes | Small message capacity, About 31 bytes |
| Robustness to operate in congested environment | More | Less |
| Battery Life | Longer | Smaller |
| Security Control | Better | Less secure |
| Theoretical Data Throughput | 2 Mbps, gives about 1.6 Mbps with overhead | 1 Mbps |
| Reliability | High | Low |
| Digital Life | Better | Less better compare to Bluetooth 5 |
| Support for IoT devices | Yes | No |
| Bluetooth Beacon | Beacons have become more popular due to increased range and speed. | Beacons were less popular due to less speed/range. |

*Table 3 [4]*

A hardware replacement for the TracPatch may prove to be the best option for the product in order to accommodate the huge increase in data traffic. This of course comes with its own obstacles as it's a rather new technology with limited resources and compatible devices.

## III. DESIGN IDEA CONTRACT

### *A. Design Approach*

What we intend to do as a team is work with Consensus Orthopedics to develop a similar but new product that gathers significantly more data and provides additional functionality. From a hardware aspect we will be working with two IMU's that each have 9 degrees of freedom for a total of 18 degrees of freedom. These IMU's are each composed of 3-axis accelerometers, magnetometers and gyroscopes.

In conjunction with the IMU's we will be using 5.0 Bluetooth modules that will double the data transfer speed of the previous 4.0 Bluetooth modules. This will assist in the current bottlenecking problem that Consensus is having with some of their current hardware and software.

Our goal is to have two prototypes, each one consisting of one IMU and one 5.0 Bluetooth Module. One will be the slave that collects and transmits data to the master, which in turn collects its' own data, receives the data from the slave and transmits the total amount of data via Bluetooth to an Android tablet where the data will then be synced and formatted into a CSV file. An application can then be built to monitor the patient data.

When we can get the prototypes working our next step will be to incorporate the software, we develop into Consensus Orthopedics' current TracPatch hardware which also consists of two IMU's and two 5.0 Bluetooth modules. Below is an image [8, Figure 1] of Consensus Orthopedics' current TracPatch hardware for their future applications.


*Figure 2 [8]*

As you can see in the above image [Figure 2 [8]], Consensus Orthopedics has moved from one device to two devices. This will not only allow them to collect more data from post knee surgeries which is what they are currently doing but will also allow them to place the sensors on a patient's hip, shoulder and spine to collect even more data from different surgeries and further progress physical rehabilitation techniques.

*B. Resources Needed*

There will be two sides of the project, the transmission side and the receiving side. Both will require very different resources and implementation.

On the transmission side we will be working primarily with the hardware with software mixed in. We will use the IMU dev board to read position data which will sent to the BLE 5.0 dev board where it will be packaged and transmitted. There will be two groups made up of each development board to simulate having two finished products. We will write in C to interface with the IO pins. A J-link cable will be used to load the programs onto the devices for testing. We will work with datasheets and technicians from both dev board companies on the specifics of the devices.

The signal receiving side will have development done on a Microsoft surface go. This tablet provides satisfactory performance for our requirements and comes with a USB connection. The USB connection is important for our design as devices with BLE 5.0 built in are extremely limited. A wifi dongle meant for development and testing will be obtained. An app will be developed that will allow users to input data. We will be reading in CSV files and synchronizing the data. We will be developing both of these with the most recent iteration of scripting language, python (3.7).

*C. Feature Set*

**Feature 1: Data Collection**

To collect data from patients using the TracPatch prototype we will be using two DK-20948 TDK InvenSense Smart Motion Platforms.

The DK-20948 is a comprehensive platform that uses the ICM-20948 IMU with a 9-axis motion sensor comprised of a 3-axis accelerometer, gyroscope, magnetometer and digital motion processor. Below is an image [Figure 3 [6]] of the DK-20948 platform.

*Figure 3 [6]*

The platform is built around a G55 MCU which can be easily used for rapid evaluation and development of projects using the ICM-20948. The DK-20948 also uses an embedded debugger for the G55 MCU.

The DK-20948 comes with InvenSense MotionLink software that includes a GUI for using the embedded Motion Drivers of the ICM-20948.

The embedded Motion Drivers can be used to configure sensor parameters such as full-scale range (FSR), output data rate (ODR), low-power or low-noise mode, and sensor interface to host (I2C, SPI).

The InvenSense MotionLink GUI can be used to capture and visualize the sensor data from the ICM-20948 IMU.

**Feature 2: Data Transmission**

To transmit the data collected from patients using the two DK-20948 platforms we will be using two Taiyo Yuden EKSHSNZWZ Bluetooth low energy module evaluation kits. Below is an image [Figure 4 [5]] of the evaluation kit.

*Figure 4 [5]*

Taiyo Yuden's EKSHSNZWZ evaluation kits use the EYSHSNZWZ Bluetooth low energy modules which use a Nordic Semiconductor nRF52832 SoC that consumes 80% less power than the nRF51 series.

The EYSHSNZWZ is ultra-small (3.25 mm x 8.55 mm x 0.9 mm) which allows it to be used in a wide range of applications with small space requirements.

The EYSHSNZWZ features 512 kB Flash memory, 64 kB RAM, and supports NFC-A Type-2 for touch-to-pair application. Faster time-to-market is made possible for engineers with FCC/ISED/MIC (Japan) certifications.

**Feature 3: Data Storage**

Once that data collected from the DK-20948 development board it will be transmitted to a Microsoft Surface GO platform where it will be stored in a CSV file.

**Feature 4: GUI Application**

We will build a GUI application that will allow us to view the data that we are collecting, transmitting and storing.

GUI application will also allow user to input various parameters such as height, weight, age, gender, ethnicity, etc. to help make the data more accurate and unique to each person.

*D. Task Assessment*

We will break down the four features into smaller milestones that can be set as a standard for determining a features completion.

1) Raw sensor data of motion:

Reading 16-bit raw data from the 6 sensors. Communicate the data over SPI that has 7 MHz capabilities (I2C as alternative) to be received by transmitting device.

2) Variable sampling rates:

We will develop a method of selecting the sampling rate. This will allow the sampling rate to take into consideration the test period whether it's a day of semi precise vs 30 min of very precise data. Sampling rates will be in the 50-100 Hz range.

3) High transmission rate:

For Bluetooth module to tablet we will use Bluetooth throughput equations.

Bluetooth throughput:

$$\text{Packets Per Second} = \frac{\text{1000 mSecs} * \text{Number of Packets in a Connection Interval}}{\text{Connection Interval mSecs}}$$

Equation 1 [13]

Packets per second with 6 packets per connection interval at 7.5 mSecs will be 800 packets per second.

$$\text{Throughput} = \text{Packets Per Second} * \text{Data Per Packet}$$

Equation 2 [14]

Packets should hold at least 20 Mb per packet so 16Kb per second.

4) Daily data collection:

multiply how much data we obtain per minute by each day to find the total amount of data we can achieve.

Data collection intervals should be about 20 minutes. If we sample at 50 Hz with a total of 6 sensors with 16-bit output the equation below should be the amount of data collected.

4800 bits/second x 60 seconds/minute x 20 minutes = 5760000 bits (720kB)

5) Tablet app:

User input such as height, weight, age, gender, race etc. Different exercises for range of data to be collected.

6) Storage format:

We want a format that is easily readable and parsed. The need is for machine learning and data synching. Comma-separated values will store each value received in column format attached with timestamps. User information will be appended to the file.

7) Data synchronization:

A day's collection will see roughly 51 MB of data collected. With two CSV files, we will sync 102 MB of data based on their time-stamped data. This will be done in a timely manner (less than 1 min).

## IV. FUNDING

We started with a very small and limited budget of $200 of personal contribution and we did not spend any of it. After we acquired our sponsor, we were given a budget of $2000 to spend. The budget was estimated to be larger than the funds used, which does occur rarely in industry. We roughly spent $700 on hardware.

## V. MILESTONES

In the course of our work, we completed several milestones before the overall completion of our project.

### *A. Data from IMU*

A significant amount of time for the first semester on the hardware side was getting the IMU to generate data. At first, we were

trying to communicate with the IMU over SPI. It came to our attention after struggling and researching, that the device was going to need I2C in order to communicate with the magnetometer.
This was discovered in the schematics. We noticed that the sensors communication was not direct for SPI. We needed to switch over to I2C. Instead of traditional I2C, we moved over to the two-wire version, TWI. After this point, we were able to finally communicate with all sensors and retrieve the data.

This change in communications did reduce our transmission time between the sensors and BLE module. SPI was at 7MHz while TWI would operate at 400kHz. Fortunately, the communication between the devices was not the bottleneck. We were able to safely make the change over and reach the desired sampling rates.

*B. Broadcasting data from BLE board*

After acquiring the data from the IMU, the next big milestone was broadcasting the data for the mobile side to connect to. This milestone required an understanding of the BLE architecture which translated to hours of research. We decided to create three characteristics that act as channels. Each sensor would be assigned one of these characteristics. They would each transmit their associated data. The modularity allows subscribing to each sensor individually enabling custom sensor reading.

After we acquired the data, we package them in BLE packets and awaited a connection. For this milestone, we used a third party BLE app to read the data being transmitted. Here we were able to verify that 2B of data per sensor axis * 3 axis = 6B per sensor. We were able to subscribe to each sensor individually.

*C. Variable Sampling Rate*

After the data was being collected and sent over BLE, we needed a way for the BLE module to accept commands for changing the sampling rate. For this we added in a 4$^{th}$ characteristic that would wait for any commands sent from the device it was connected to. We decided on ASCII values for ease of human interaction. Once an interface was agreed on, we could start accepting commands. It was easy to change the sampling rate once this feature was in place.

A significant milestone was getting the Android application to connect to the BLE device. Accomplishing this allowed us to further development and testing. Without this, there would be no way to develop methods of data acquisition, formatting or storage. After this connection was made, previous assumptions were proved incorrect and entire methods of programming had to be re-analyzed or scrapped. This milestone showed us an accomplishment that lead to further work to be done.

Establishing notifications and receiving data continuously was another major milestone. After a connection was established with the device and program application, several days were spent on this task alone. The intricacies of this method took time to understand and establish. No data formatting or storage could be done properly if the notifications weren’t sent up correctly. Upon completion of this milestone, data was passing from the device to the program application in real-time and made the rest of the work needed to be done tedious in comparison.

## VI. WORK BREAKDOWN STRUCTURE

Project work breakdown structure (WBS) and Task assignments per team member including hours per task and hours per team member [Project work breakdown structure (WBS) and Task assignments per team member ]. We broke down the sections for the project into 4 main areas; Data collection, BLE transmission, data storage and GUI application. These seemed to be to most fluid separations that allowed the team to work in parallel. The jobs were also distributed in accordance to each member's strengths. Both Cody and David had the most experience in software so they would work with the BLE and data storage. Then for hardware, we gave the assignments to Bryan and Jay due to their coursework background. We also assigned estimated time requirements for each breakdown with BLE and BLE integration taking the bulk.

| Feature | Sub Feature | Metric to be used to determine completion | Assignee | Estimated Hours (hrs) Feature and Testing |
|---|---|---|---|---|
| Data Collection (IMU) | Accelerometer | 6 Byte readings | Bryan | 10 |
| | Gyroscope | 6 Byte readings | Bryan | 10 |
| | Compass | 6 Byte readings | Bryan | 10 |
| | TWI TX | 400KHz | Bryan | 40 |
| IMU/BLE Integration | TWICommunication | TX/RX 400KHz | Cody + Bryan | 10 + 30 = 40 |
| Data Transmission (BLE) | TWI RX | 400KHz | Cody | 30 |
| | Connect BLE module to tablet | module discoverable and connection made | Cody | 20 |
| | Assemble data packets | 6 bytes data per packet | Cody | 30 |
| | Throughput | 4.8kbs | Cody | 10 |
| | at least 3 sample rates | Sample rates between 1, 5, 10 | Cody | 10 |
| | Sample rate RX | RX sample rate from mobile device and send to IMU | Cody | 10 |
| BLE / Android Device Connection | BLE connection | A connection Between BLE device and written application is established. | Cody + David | 56 + 60 = 116 |
| | | Transmit Data from BLE Board | Cody | See Data Transmission (100) |
| | | Receive Data from BLE board | David | See Android Reciever (88) |
| | | Transmit Data Reate from App | Cody + David | 4 + 4 = 8 |
| | | Receive Data Rate from App | Cody | 20 |
| Android Recieiver for BLE data | Check / Establish Permissions | All necessary permissions are established or requested before program execution | David | 8 |
| | Scan for BLE Devices | | David | 30 |
| | Search for BLE Service | | David | 10 |
| | Search for BLE Characteristic | | David | 10 |
| | Establish Characteristic Notifications | | David | 30 |
| Data Storage & Synchronization (Sample) | Create Sample Data | Data corresponds to sa real sample of data from a static instance | David | 4 |
| | Get Sample Data | Data is Aggregated from creation method | David | 1 |
| | Translate Sample Data | Data is translated from Bytes to readable data. | David | 10 |
| | Format Sample Data | Data is saved and opened as a Comma Serperated Value file | David | 12 |
| | Append Sample Data | User and device info is appended to data | David | 1 |
| | Save Sample Data | Data is stored in the file correspondind to date created. | David | 16 |
| | Synchronize Sample Data | Each Data segment has a corresponding timestamp | David | 2 |
| Data Storage & Synchronization | Establish Permission | Device requests permissions | David | 10 |
| | | Device will not crash with incorrect permisions | David | 2 |
| | Receive Data | For every method call, data is accepted and handled | David | 6 |
| | Translate Data | Data is translated from Bytes to readable data. | David | 4 |
| | Format Data | Data is saved and opened as a Comma Serperated Value file | David | 3 |
| | Append Data | User and device info is appended to data | David | 4 |
| | Save Data | Data is stored in the file correspondind to date created. | David | 1 |
| | Synchronize Data | Each Data segment has a corresponding timestamp | David | 3 |
| Data Storage and GUI Integration (Sample) | Function Integration (Sample) | A successful call is made to display data | David | 18 |
| Data Storage and GUI Integration | Function Integration | A successful call is made to access stored, synchronized data. | David + Jay | 2 + 3 = 5 |
| | | Functions present in GUI application display correct data. | David + Jay | 2 + 3 = 5 |
| Gui Application (Android Tablet) | User Input | A number representing age can be accepted and displayed back to user | Jay | 15 |
| | | A number representing weight can be accepted and displayed back to user | Jay | 15 |
| | | A number representing height can be accepted and displayed back to user | Jay | 15 |
| | Data Rate | Choose from three different data transmission rates | Jay | 50 |
| | Browse and View Data | Browse through menu to view pre-synched data in the CSV file | Jay | 45 |

*Figure 5 [12]*

## VII. RISK ASSESSMENT

Risk management will be imperative for accounting for events that cause large impacts to the project. These risks can come in many forms ranging from hardware, software, experience, and any number of external events.

Analyzing our work breakdown structure, we developed a critical path. This will help us locate periods in development that can hinder progress for the project. Our project is modular in nature and allows us to work in parallel for large portions but there are points that require work to be completed before other modules can proceed. Our critical path consists of receiving sensor data over SPI, transmitting that data over BLE, storing that data along with user info into CSV and calling the storage/sync functions from the GUI.

### *A. Perceived Critical Path*

From the critical path figure, we have identified key areas that may hinder our projects progress. Accounting these risky periods and devising ways to mitigate them will better prepare our team to finish the product on time.

The first path is SPI communication. The whole purpose of the project is to receive the data off the IMU. The first communication to be traversed by the data is SPI. We must communicate this data over to the BLE. The IMU must also receive sample set commands over SPI sent from the GUI. Technical difficulties will lie mostly on our inexperience with the technology. Incorrect wiring and connections can result in the board being damaged, requiring us to increase our budget and wait for a replacement.

Second path is BLE communication. Once the data has been collected within a sample range, it must be sent to the tablet. Here yet another choke point occurs. If this feature halts, the data can no longer make progress.

Third path is the storage and syncing of the data. Permissions need to be established to read and write data with the chosen tablet. Setting up permissions inaccurately will delay any further progress. Aggregated sensor data within a certain time frame needs to be correlated or synchronized

**Risk Matrix**

| Probability | | | *Minimum or No Impact* | *Impact Can Be Tolerated* | *Limited Impact* | *May Jeopardize Project* | *Will Jeopardize Project* |
|---|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 | 5 |
| | *Not Likely* | 1 | | Gui Input | | | Sensor Data |
| | *Low Likelihood* | 2 | | Gui Data Visualization | Variable Sample Rate | SPI Transmission | BLE Connection |
| | | | | | Data Format | SPI Reception | Data Storage |
| | | | | | Data Synchronization | | |
| | *Likely* | 3 | | | BLE Throuput | | |
| | | | | | Variable Data Rate | | |
| | *Highly Likely* | 4 | | | | | |
| | *Near Certainty* | 5 | | | | | |
| | | | **Impact** | | | | |

*Figure 6 [11]*

in some way. If we make the time slice too large, the data will be less accurate.

The last path is the GUI. This module is needed for the user to call the storage and sync functions. It will start the data collection and will also send a command over BLE and SPI to set the sampling rate.

*B. Risk Assessment Chart*

This is a Risk assessment chart that details the potential impact on our products timeline vs the probability it will occur [**Error! Reference source not found.**].

*C. Mitigation for Critical Path*

To mitigate risk, deadlines need to be set earlier, and ample time needs to be set aside for unexpected complications. If a situation occurs where more knowledge is needed or more testing is necessary, there will be a delay on the entire project. Without extra time in the schedule, assignments should be aimed to be completed sooner to help complications arise sooner.

To mitigate risks, tasks will be completed early and verified by team members to allow complications to arise sooner. Reference material, useful links, standards, and communication lines will be established prior to them being needed. Also, in the event a task exceeds the allotted time given for completion, communication will be established with Consensus Orthopedics to discuss how they are structuring their code or designing their devices.

The Graphical User Interface will be the only control a user has over the device, turning on and off data collection and setting the variable sampling rate will be done through the GUI. Other features such as viewing the stored data and taking a user input and storing it will not jeopardize the project, it certainly will affect it. To mitigate this part of our project we will set an earlier timeline and have multiple tests after each feature to be sure of the reliability of the GUI.

*D. Mitigation for Sub-Tasks*

1) IMU Risk Assessment

a. Data Collection
Accomplishing sensor data collection should be very straight forward. The only cause that would prevent data acquirement would be hardware failure that would be easily replaced in a short amount of time as there are plenty of parts ready to ship and close enough to be shipped in under a week's time maximum.

b. Transmit Data Over SPI
Transmitting data over a SPI connection to the BLE module could prove to be complicated as our team has limited experience with such communications. Working directly with Consensus Orthopedics and their embedded system designers as well as TDK Invensense applications engineers will help mitigate the fact that we have limited experience as well as working with drivers that have SPI examples that we can modify to work with our project.

2) Bluetooth Risk Assessment

a. JTAG connection
The JTAG connection is used for serial interfacing with the flash memory. Through this we upload new code to be ran on the hardware. The connection however can be easily reversed on the connector. If this is done, the device could short out, damaging

our board. Each board costs $150, resulting in a hefty penalty for failure. Requiring reference documentation to be used when making connection should drastically lower chances of this occurring.

b. SPI connection
It's imperative to have communication with the IMU in order to transmit the data over BLE to the tablet. Different issues can arise with this such as not being able to sync the devices, handle the required throughput, and inexperience with the technology. Thorough research in event handlers should help mitigate this risk and streamline the development.

c. Assembling BLE packets
This portion will be the defining moment for our data throughput. We must find a manner to package the raw data from the IMU and transmit it. This has been a bottleneck before for Consensus Orthopedics and surpassing it may prove difficult due to our inexperience with the technology.

d. Connection intervals
Another key factor for throughput will be sending packets given intervals. We will have to set the tablet up to accept the connection up per interval and accept the data that the BLE has connected. This will take trial and error and will most likely take the longest amount of time resulting in it being the largest risk for BLE.

3) Storage and Synchronization Risk Assessment

a. Storage Permission
When reading or writing files on a device, permissions need to be established. This is the precipice of this section of the project. All other tasks are mute and cannot be verified if permissions are not established. Complications could occur when transferring from an emulated (used for scripting and testing prior to uploading on physical device) to a physical device. The differing environments might require different storage routes or different permission checks.

b. Data Correlation
When synchronizing data, a correlation between data elements needs to be established. Additional fields need to be appended to each data element to indicate a correlation. Editing existing data elements is focal to the latter half of this portion of the project as most other tasks should be completed prior to this one. To minimize complications, a graphical representation of this section will be developed and checked by team members prior to any code development. This will eliminate code regression caused by faultable approaches. An algorithm needs to be written to assign a correlation between data elements. Timing set on BLE device and timing set on mobile device might differ and cause inaccurate correlations.

4) GUI Risk Assessment

a. User Input
GUI will be taking user demographic inputs and store it, this will be done through dropdown menu with all the necessary options, user will select their choice and those choices will be stored. If we come across an issue with the code or difficulties storing through this design, we will change the input process. We will have user type the inputs rather than selecting through dropdown menu.

b. Turn on Data Collection
This feature will simply be an on/off button on the GUI. Which will allow the user to start and stop the data collection. If we aren’t able to facilitate this option, we will

remove this feature and have the user manually turn the collection on and off rather than through the GUI.

c. Setting Sampling Rate
Mitigating the setting sampling rate feature of GUI will be a very important aspect of our project, because this is one of the features that our sponsor CO has required. Complication for setting the sampling rate through the GUI will solely depend on the complexity of the JAVA code and proper connection through BLE and SPI.

d. Viewing Stored Data
Mitigating viewing of the data will require us to use an alternate route for this feature. In a case that the feature doesn't behave the way we intended, we will use an alternate, simpler code and even consider using a different medium for viewing the stored file. If all alternatives fail, we will take this feature out of the project and access the file manually on the tablet and not through the GUI.

Refer to product flow diagram for path clarification [**Error! Reference source not found.**].

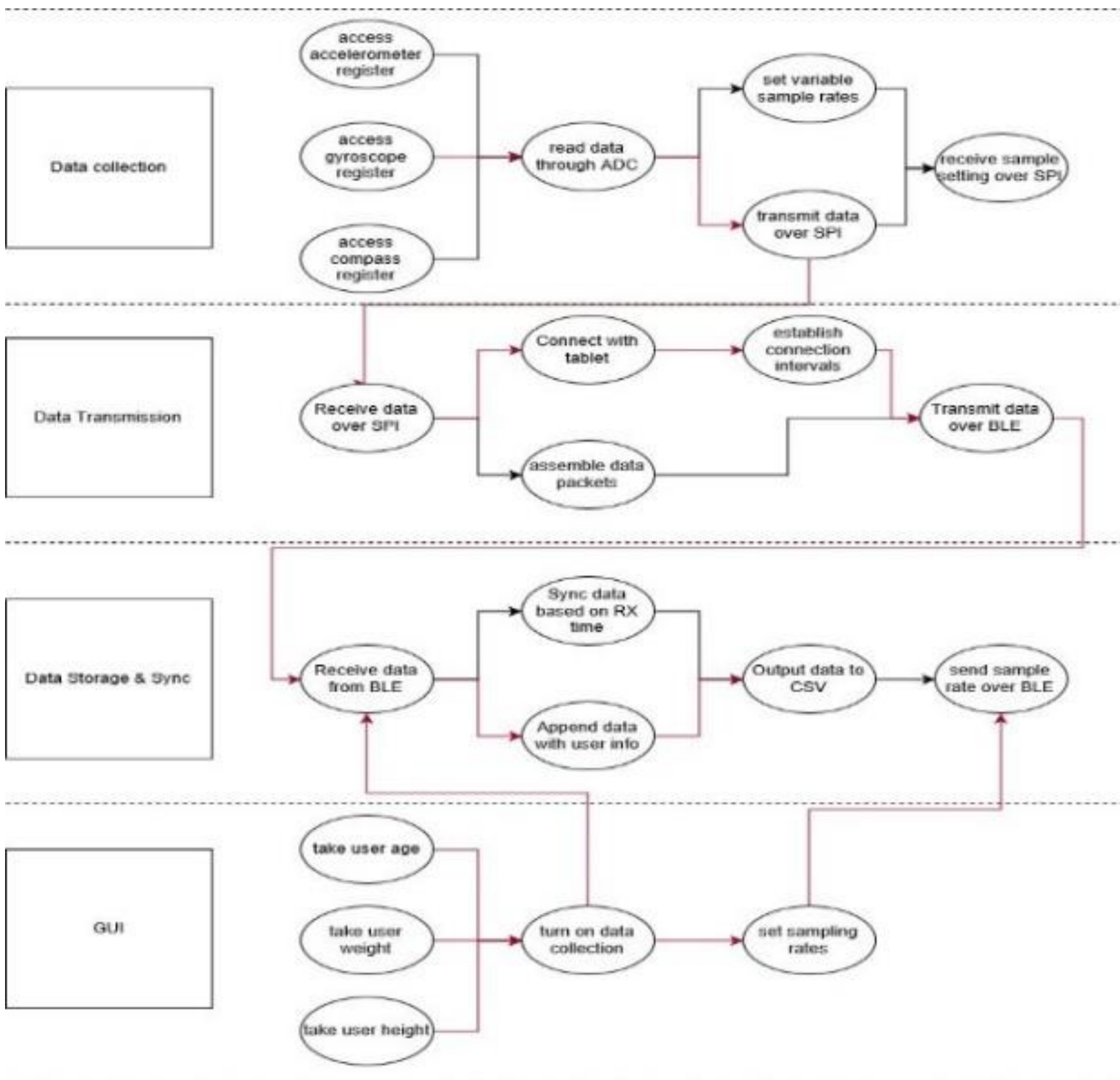


Figure 5 [10]

## VIII. DESIGN OVERVIEW

Our goal was to have a prototype consisting of one IMU and one 5.0 Bluetooth Module. It collects and transmits data via Bluetooth to an Android tablet where the data is then synced and formatted into a CSV file. An application was built to monitor the patient data.

### *A. Bluetooth Feature*

The Bluetooth section is the middle portion between the IMU and the mobile device. The Bluetooth module is using TWI (two wire interface) to communicate with the IMU board and transmits the sensor data over Bluetooth. In the middle of these two endpoints are various key points for setup.

1) BLE stack initialization

This initialization sets up the various layers to implement the Bluetooth protocol [1]. This phase is mostly black box, but we set the BLE event handler here. The BLE event handler is a function that handles all incoming and outgoing BLE messages. This event handler will be a layer below the ble_mpu events and act as a middle man for these events.

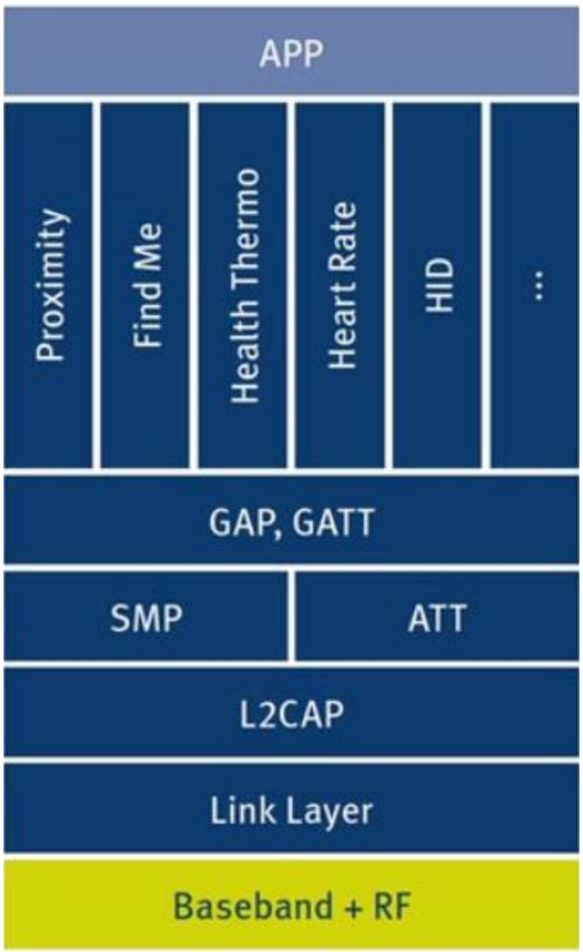


*Figure 7 [19]*

2) GAP parameters initialization

GAP stands for Generic Access Profile. Here is where the general topology is defined. We set up the connection intervals, connection timeouts, device name, and security. We also determine if the system topology will be connection or broadcasting. In our set up we set the connection interval to range from 7.5 to 400ms, the recommended range. This setting is more like a suggestion for the connection. It will attempt to reach somewhere within the range, but the ultimate decision is made behind the scenes for optimization [2].

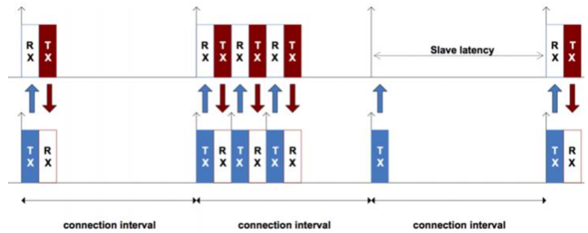


*Figure 8 [20]*

The device name is set here. This name appears in the discoverable devices window. Our project set this as MPU BLE.
Security when connecting to devices is important, so we enabled the encryption on the device that requires a 128-bit encryption key.

For this setup, we went with connection topology over broadcasting. Connection topology allows for higher throughput to one device. This is more beneficial with just the two edge devices.

3) GATT initialization

GATT stands for Generic Attribute Profile. This section lays out the details how attributes (data) is transferred once devices have made a dedicated connection. It also sets up the advertising scheme. All standard BLE profiles are based on GATT and must comply with it to operate correctly.
Our BLE module sets up as the server while the phone acts as a client. The BLE module will wait for a BLE event that establishes a connection before transmitting any data. The BLE module will advertise in intervals of 180ms. This interval is chosen as a midpoint in power consumption and responsiveness.

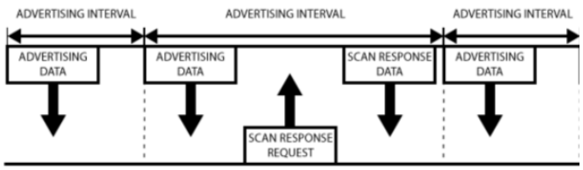


*Figure 9 [21]*

4) APP timer's initialization

3 application timers are setup to run as timed interrupts. There is a timer associated with each sensor on the IMU. They are setup to repeat every time the timer expires and they each have their own event handlers. The event handlers set a notification to true, signifying a timer has expired and its sensor must be read.

5) Services initialization

A service represents conceptually similar data that are grouped together. These services are defined under a profile. Services are then made up of characteristics that can transmit various data [4]. Each of these services and characteristics will have their own UUID (universally unique identifier).

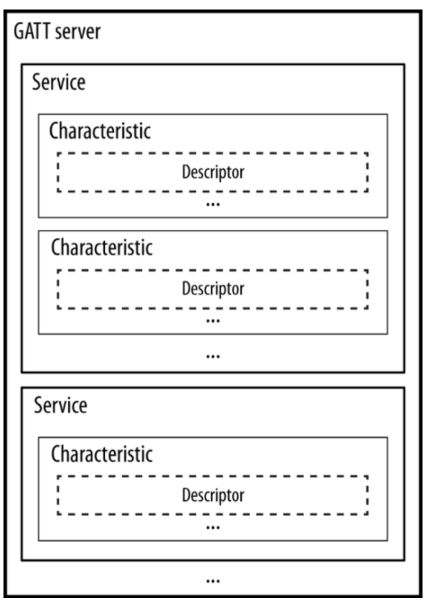


*Figure 10 [22]*

Our application has one service. This service has four characteristics. Each sensors data is transmitted over a respective characteristic. The fourth characteristic is for commands. Currently only one command is available, to change sample rates.

6) MPU initialization

We communicate with IMU through TWI. We start the IMU by resetting the IMU and assigning an event handler to deal with all events leaving and enter the BLE board. We send various commands to the IMU for setup such as the accelerometer range and internal sampling rates.

7) Start app timers

We start the timers for all 3 applications. They are initialized to 1Hz at the start of the program

8) Start advertising

Advertising begins, waiting for client to make connection and request data transfer.

9) Enter power management

Power management puts the device in low power state while waiting for events to occur. This resides in an infinite loop and acts as a blocking statement. It will unblock once an event occurs.

10) Update characteristic on event

Once a connection is established, the client can subscribe to a characteristic for updates. A notification will be flagged on the BLE module through ble events. This will enable the app_event_handlers to trigger characteristic updates.

When the application timers expire, the app_event_handlers set a notification flag for the infinite loop to check once power management is exited. Each application is set with the same sampling rate resulting in each device requesting data from the IMU via TWI.

When the IMU responds with the data for each sensor, TWI event handlers will store the data. Each sensor has three axis's and must transmit each one. These values are combined into 1 packet of 6 Bytes (2 Bytes for each axis).

The characteristic connections are checked and if they are, the corresponding sensor data is sent over its characteristic. The notification is reset, and power management is started until another event occurs.

At any point, the mobile device can send a sample rate command. This command comes in the form of 1 Byte over its own characteristic. In the occurrence of an event, it will modify the application timers to match the desired sampling rate. Its range has a programmed limitation of 1-100Hz (Figure 22 [9]).

*B. Data Storage*

1) Get Data

In regard to getting data, there are many things to consider. After the Bluetooth communication is established, there needs a way for the data to be aggregated in the code. Methods are established to activate when a connection is established and when data is received.

2) Translate Data

Once the data is received, the data needs to be changed to a different data type. Two approaches are common. Using input streams and output streams or using a buffer. Either way, the data is turned into a String data type. This way the data can read and manipulated properly.

3) Format Data

Once the data is understandable and translated into a String, it can be manipulated more easily. The data is modified to include spacing, indents, and commas to be consistent with CSV format: comma separated values.

4) Append Data

Once the data has been formatted, information regarding the user, device, and time need to be correlated with the data. When looking at data, this information is relevant, especially when trying to correlate data from several devices over some time.

5) Save Data

After the data is appended with the proper information, it needs to be saved. The data is translated again to a CSV file. Permissions need to be established to access the phone memory. A path to the file location needs to be specified and saved in the code.

6) Synchronize Data

A CSV file is saved per device per day. This will create unique, very specific and clear segments of code. To correlate the data, the use of timestamps and user information is used to establish relationships and data links between the files

*C. Data Generation*

The BLE module interfaces with the IMU via TWI. TWI allows minimal wire connections in a compact device. Our connection runs on 400KHz. This provides fast communication between the devices, accelerating data transmission via Bluetooth.

Mobile Device The mobile device subscribes to the BLE service MPU BLE. It can then subscribe to each/all the characteristics, receiving three6 byte sensor data. The mobile device can select the sampling rate via user interface. It transmits the sampling rate over the characteristic.

## IX. DEPLOYABLE PROTOTYPE STATUS

*Figure 11 [23]*

Our project is mostly split into two categories; hardware and software. It's these categories that we will explain in detail regarding the project's current status.

### A. Hardware

Our product is currently communicating with TWI between the IMU and BLE module. This was one of the first features in our pipeline. We wanted the devices to communicate fast enough to accommodate the data coming from IMU and the variable sampling rates.

Our packet bursts consisted of three write commands with three read commands [Figure 12 [25]]. With each clock, the value on the data line is read.

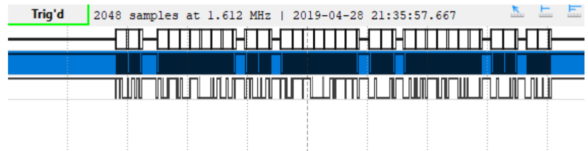


*Figure 12 [25]*

The transmission speed of the communication was meant to be 400kHz. This speed was achieved [Figure 13 [26]].

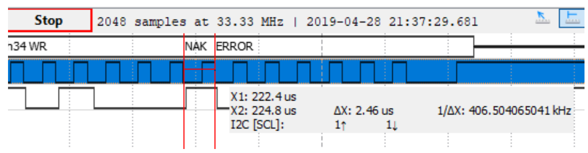


*Figure 13 [26]*

When analyzing the packets send and received, we can see the values being transmitted over TWI. The first portion is the write command. The register that we wish to read is placed in the write register. Then the appropriate sensor responds with the sensor values. 6B can be seen in the transmission. This group represents one sensor and its 3 axes'. Each sensor can be further broken down to the 2B for each axis [Figure 14 [27]].

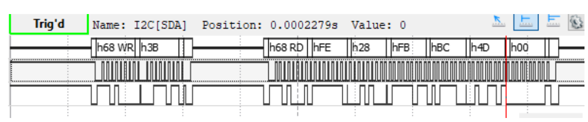


*Figure 14 [27]*

### B. Software

We programmed the BLE module to broadcast with a predefined UUID. Devices could search for the device and connect to it [Figure 15 [28]].

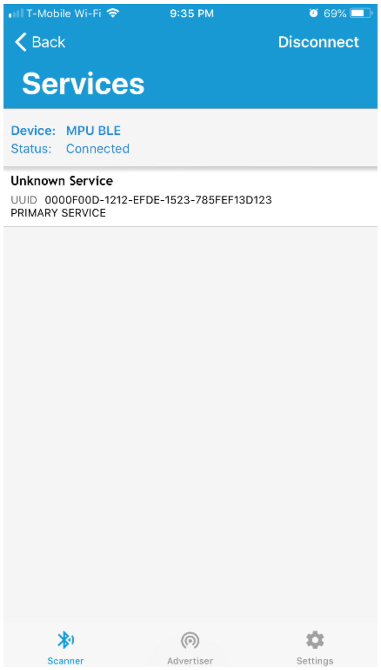


*Figure 15 [28]*

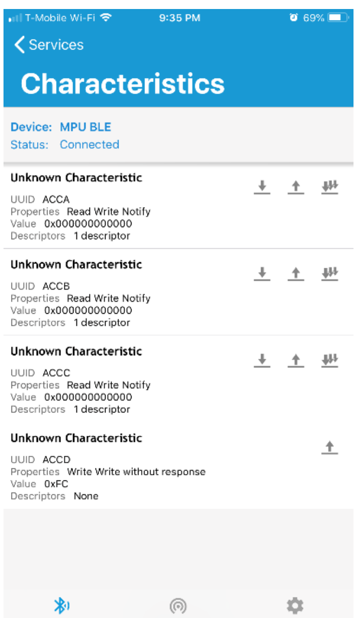


*Figure 16 [29]*

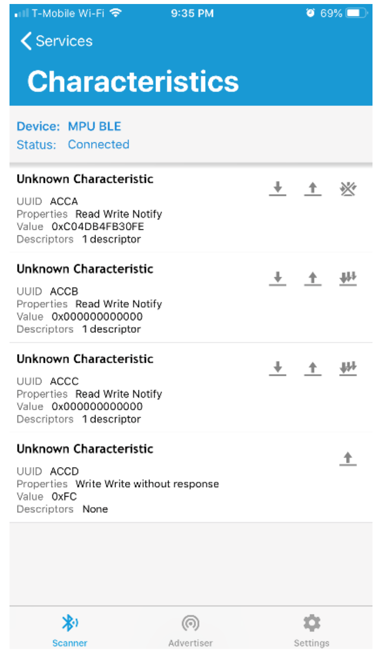


*Figure 17 [30]*

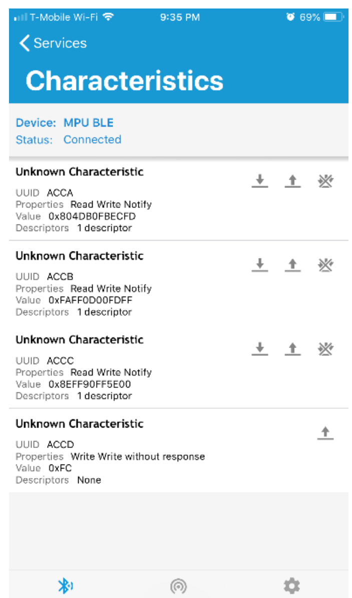


*Figure 18 [31]*

Once you select the service, the characteristics are seen [Figure 16 [29]]. Each characteristic UUID can also be seen. The first three are assigned to transmit data for a corresponding sensor. Then there is the fourth one for sampling rate commands. These characteristics are turned off initially. They can each be selected which will start the data transmission [Figure 17 [30]].

All the sensors can select which will maximize the throughput at the starting sampling rate of 1Hz [Figure 18 [31]].

During runtime, the sampling rate can be changed. While monitoring through UART, we can see the events occurring [Figure 19 [32]]. When an EVENT occurs, that means a BLE packet was received. We can see the characteristics being subscribed to when the throughput is increasing from 384 to 762 to 1152. Then we change the sampling rate to double and the throughput increases to 2304.

```
<info> app: EVENT!!
<info> app: Throughput: 00384

<info> app: Throughput: 00384

<info> app: Throughput: 00384

<info> app: Throughput: 00384

<info> app: EVENT!!
<info> app: EVENT!!
<info> app: EVENT!!
<info> app: Throughput: 00768

<info> app: Throughput: 00768

<info> app: EVENT!!
<info> app: Throughput: 01152

<info> app: Throughput: 01152

<info> app: Throughput: 01152

<info> app: Throughput: 01152

<info> app: EVENT!!
<info> app: Changing sample rate to 2

<info> app: Throughput: 02304

<info> app: Throughput: 02304

<info> app: Throughput: 02304

<info> app: Throughput: 02304

<info> app: Throughput: 02304

<info> app: Throughput: 02304

<info> app: Throughput: 02304

<info> app: EVENT!!
<info> app: Changing sample rate to 1

<info> app: Throughput: 01152
```

*Figure 19 [32]*

## X. MARKETABILITY FORECAST

The product we are developing falls under wearable technology. Wearables, as the name suggests, are gadgets that can be worn. Wearable technology achieved mainstream popularity with the Bluetooth headset in 2002. Between 2006 and 2013, iconic wearable technology devices Nike+, Fitbit and Google Glass were released. In 2014, dubbed “The Year of Wearable Technology” by several media outlets, activity trackers grew in popularity and the Apple Watch was introduced.
“Wearables will become the world’s best-selling consumer electronics product after smartphones,” CNBC reports. Autonomous or smart wearables are projected to exceed 305 million units in 2020, with a compound annual growth rate of 55 percent. Projected sales figures for autonomous wearables are well above laptops and televisions. Smartphones will remain dominant in the consumer market, with projected sales to top 1.6 billion units in 2020.

Our product is more focused in the industry of healthcare and fitness of the wearable devices. Fitness devices have had more sales than the smartwatches, and the sales are projected to triple by the end of 2020.

The initial cost of manufacturing the wearable devices is very high leading to high prices for the ultimate consumers. The producers of the wearable devices have to spend a lot of money on research, marketing & promotion, shipping, licensing, software, and development of wearables due to which the cost incurred is substantially high. For example, our sponsor’s current product TracPatch with all the accessories is about 700 dollars. The manufacturing cost in this industry may decrease with the growth in technological advancements and awareness in healthcare.

products with growing markets to address another growing market, remote patient monitoring.

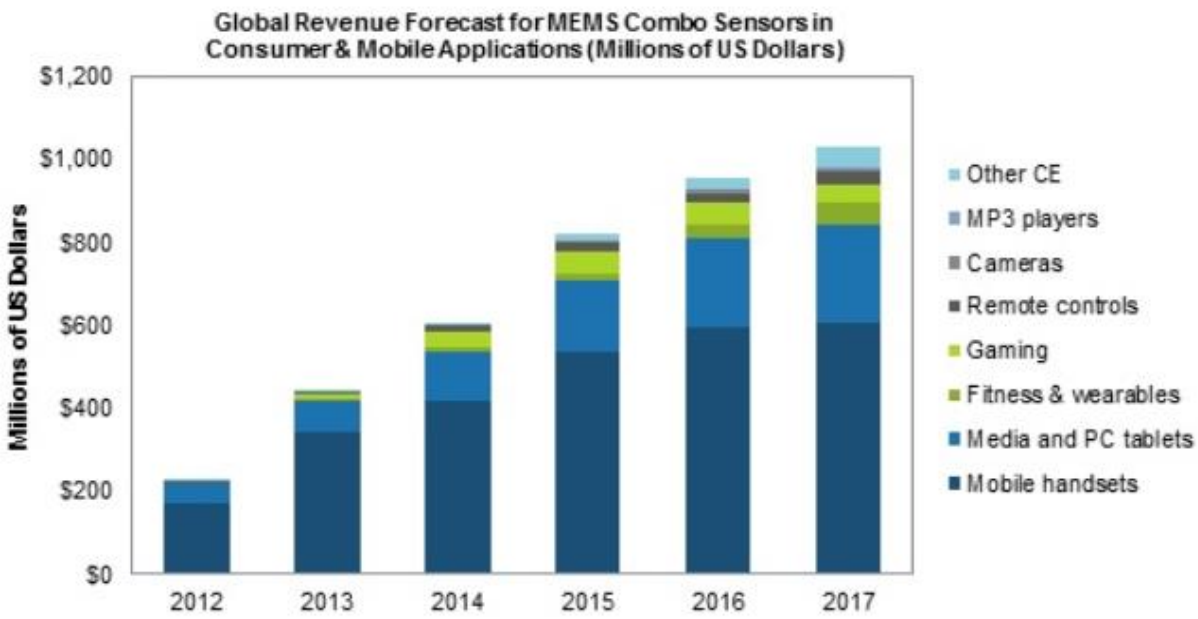


*Table 4 [17]*

There are large numbers of companies that are our competitors, ranging from Fitbits to big companies like Apple. This can be an issue for us if we were solely focused on marketing our product directly to consumers, but our major focus is towards the hospitals and healthcare providers.

*A. Outlook*

The market for measuring movement is continuously growing. Inertial measurement units are being placed in millions of devices. The higher the sampling rate, the more precise and relevant the data becomes. This however runs into the issue of tying up the data pipeline. New and upgraded technologies open the door for the data to transmitted to devices. Giving companies access to this data allows them to serve their customer base more efficiently. These particular features are important for companies such as our sponsor, Consensus Orthopedics. They are combining these two

Global microelectromechanical systems (MEMS) sensor market has seen revenue increases as high as 37% from 2012-2014. Consumer and mobile applications were roughly $608.2 million in 2014, up from $443 million the prior year. In 2017, combo sensors passed the $1 billion threshold. Combo sensors have risen to 50% of total revenue from MEMS in 2016. Currently 6-axis IMU's dominate the market at 79% of combo sensor revenue. They will continue to expand with 29% compound annual growth rate (CAGR) [1]. 9-axis IMU have yet to pick up in popularity compared to 6-axis. In 2013, production was limited to 5 million units. This growth in sensors on products creates a nest of data to be harvested [Table 4 [17]].

All this data is waiting for a method of collection. This issue is becoming more prevalent with the advent of internet of things (IOT). More companies are using

Bluetooth as their preferred method of connection. There has been a 12% CAGR over the past 10 years [Table 5 [18]].

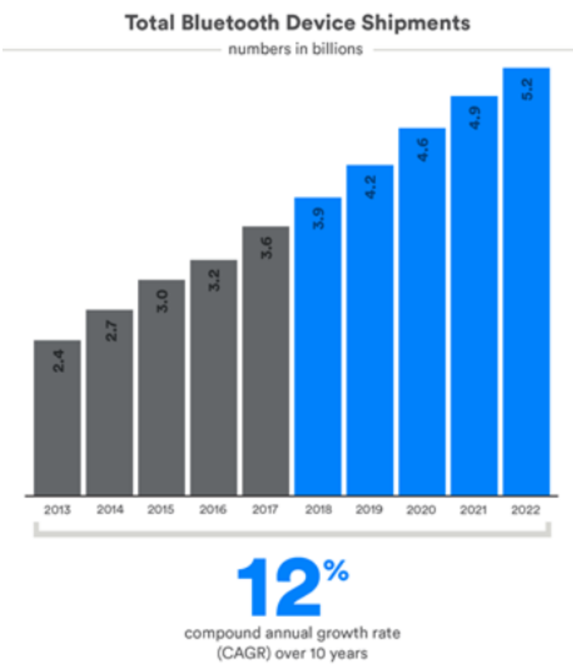


Table 5 [18]

Bluetooth 5.0 in particular is allowing a wider door for this data to pass through allowing more applications and data analysis. Using Bluetooth as the data transmission mode allows for a 24/7 connection, low power consumption, high compatibility, and high throughput. In 2018, 4 billion devices will be shipping with the new 5.0.

Our sponsor and remote patient monitoring are mostly concerned with point-to-point topology. This focus is on data transfer and is highly favored in the sports and health markets. Over 550 million devices were shipped in 2018 with this focus in mind. It's projected to have a 2x growth in data transfer solutions by 2022 and will continue to make up a large portion of Bluetooth topologies [Table 6 [13]][Table 7 [14]].

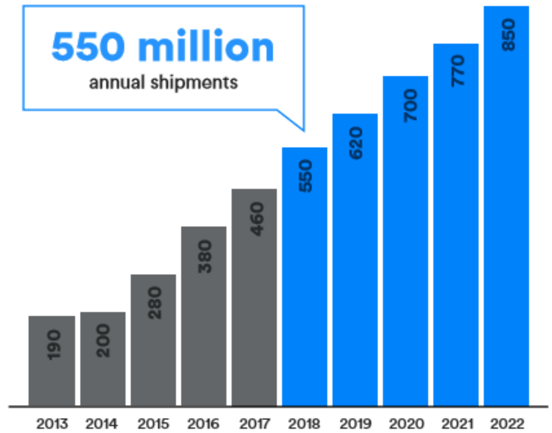


Table 6 [13]

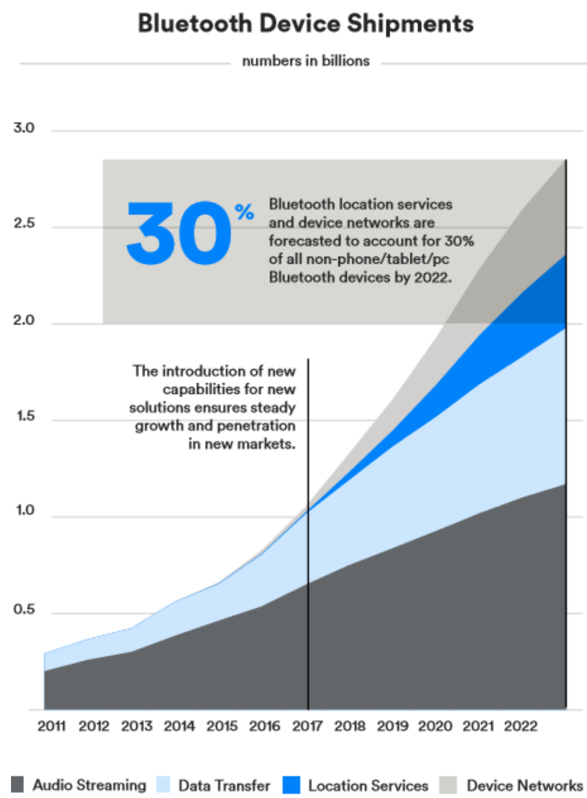


Table 7 [14]

Smartphones, tablets, and laptops have made Bluetooth a base requirement in production with 100% of the 2.05 billion devices in 2018 containing the technology [Table 8 [15]].

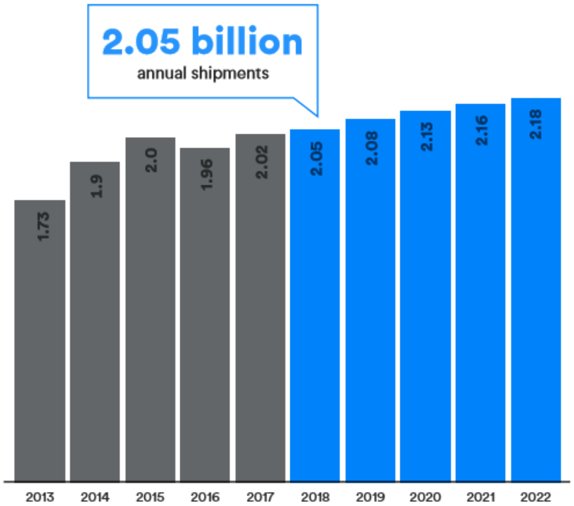


*Table 8 [15]*

Bluetooth 5.0 has been the fastest integrated version to release. It sets the stage for meta-data to be collected in the upcoming IOT era.

The combination of IMU and BLE will be significant in various sectors such as medical and industrial. Healthcare providers are continuing to contribute to the market's growth significantly, driving a 28% CAGR in healthcare wearables in the next 5 years. This extends to healthcare buildings that are projected to have 100 million Bluetooth devices shipping in 2022. Robotics and assembly lines will be able to use these to track products and equipment. Industry will see 7x increase in annual Bluetooth devices and 12x in asset tracking and management by 2022 [Table 9 [16]].

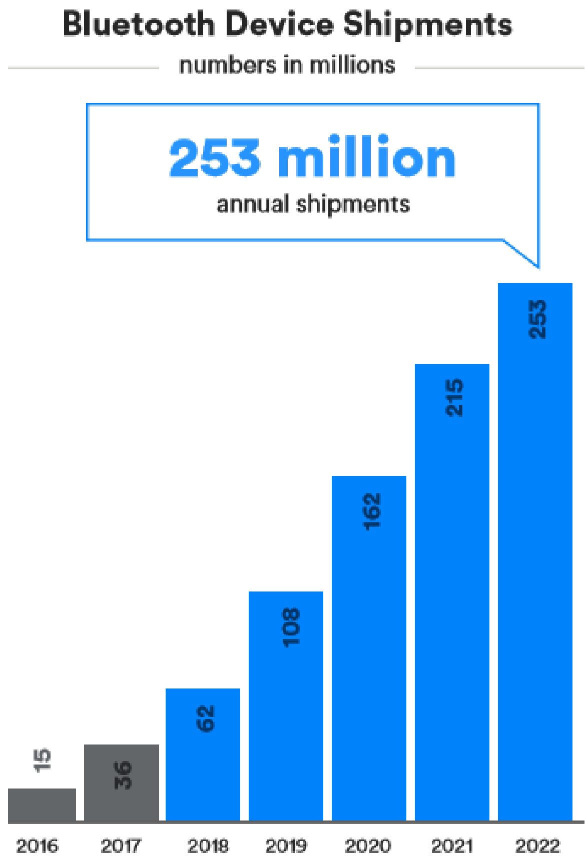


*Table 9 [16]*

*B. Target Market*

We are attempting to transmit large amounts of raw sensor data continuously with Bluetooth 5.

We intend to establish an infrastructure which includes Bluetooth 5 and three movement sensors: accelerometer, gyroscope, magnetometer. Our product will serve as a proof of concept and be used as an intermediary to future product. Our product is not intended to be implemented to the public market as it requires refinement by Consensus Orthopedics.

There exist devices which transmit small amounts of data for short intervals of time with Bluetooth 4; Consensus Orthopedics' TracPatch does just that. There are devices that transmit small amounts of data with Bluetooth 5. There are devices that transmit large amounts of data, but do not include Bluetooth 5. Currently, there does not exist a product in the public market which includes a device with our criteria.

Consensus Orthopedics paid engineers to develop TracPatch and are interested in the product we are developing. Other Surgeons could potentially have interest in measuring patient movement. Other medical professions such as Chiropractors and Physical Therapists could make use of this product. It should be noted that hospitals or practices would be the target of our sales instead of the patients or doctors. Distributors of medical devices would also be a desired target.

The features included in our product could have applications outside the medical field and so it is possible that our target market could indirectly extend.

Our device will be announced during engineering and medical conferences to establish rapport among the industry. Studies will be done with the device to show how much improvement is seen through patients using it. These studies will be published in medical journals and newspapers. Our device will be advertised online through websites selling medical devices and through paper catalogs.

*C. Competitive Analysis*

Our sponsor Consensus Orthopedics has already developed their first iteration of TracPatch to monitor and collect data from post-surgery patient's rehabilitation and is helping improve the connection between surgeons and patients.

Consensus wants to move forward and develop a second iteration of TracPatch to further improve the quality of post-surgery recovery and the connection between surgeons and patients by using the data they collect for rehabilitation and gait analysis.

Gait analysis dates to before Jesus with scientific pioneers such as Aristotle. But it did not make huge leaps and bounds until the early 1900's when cameras were developed and even more so in the late 1900's when video systems became popular.

Consensus and their competitors are attempting to make further progress. Right now, the TracPatch is pretty much on par with their direct competitors such as BTS Bioengineering with their G-Walk, Running Injury Clinic with their KinetiGait and Kinesis Health Technologies with their Kinesis Gait.

What limits Consensus is the amount and type of data they collect because of the types of sensors, the number of sensors and the technology they are using in their sensors. Consensus has already moved beyond their competitors like Running Injury Clinic who use bulky treadmills with cameras and wired systems, but the drawback they have by using wireless compact sensors is the limitations on the data they collect.

This is where the second iteration of TracPatch will improve and stand out amongst competitors. Competitors are currently using cameras, two devices mounted on separate limbs or one device that has limited battery and data logging capabilities.

The second iteration of TracPatch will have two wireless sensors that patients can wear throughout the day without the need of any bulky equipment. These sensors will be applied to either side of the joint instead of being on separate limbs which will increase the types and amount of data about the limb that has undergone orthopedic surgery. Lastly, the technology that will be incorporated into the second iteration of the Tracpatch is cutting edge.

Consensus is using some of the most recently developed inertial measurement units (IMU's) with nine degrees of freedom per sensor (three axis accelerometer, three axis gyroscope and three axis magnetometer) that collect and process more data more quickly in conjunction with cutting edge Bluetooth Low Energy 5.0 (BLE) technology that allows roughly double the speed, distance and throughput of previous Bluetooth technology for data to be transmitted and analyzed.

These cutting-edge low power consumption IMU's and BLE's in conjunction will be able to collect data and be worn by the patient all day while only needing to be removed and charged at night while the patient is inactive.

The one other aspect of the second iteration of the TracPatch that stands out amongst its' competitors is that it is being designed to be placed on multiple joints to assist in other orthopedic surgeries as well, including the shoulder, hip, neck and spine. Therefore, TracPatch will be able to monitor multiple orthopedic surgeries as well as perform gait analysis.

### *D. Final Forecast*

The work we've done for our sponsor will allow them to further their products functionality by providing them a doorway to data. Big data is becoming increasingly important to companies due to the potential for analysis. There is one main area that the product could move into to further its marketability greatly, data upload.

Currently, the patients must visit the medical office for their caregiver to download. If we implemented a method for the data to be uploaded to the cloud, we could eliminate/decrease the need for in person visits. This will increase the willingness for patients to use the product, making it a more formidable opponent to similar products.

Industry, market and competition are all things that must be taken into consideration when working on a new project or developing a new product. The industry we are moving into is the wearable biometrics industry which is growing rapidly and will be an essential technology for society and it's continued healthy recovery from orthopedic surgeries. The market we are targeting is the health industry. This wearable technology will not be for the average joe at home to analyze his own recovery. The data generated by our product is for trained professionals to review and consider improvements for rehabilitation. There is no lack of competition for our product as this market is growing and there is a lot of room for advancement. We have collaborated with Consensus Orthopedics to bring a new and efficient product to the market to better collect and transmit data to better improve post-surgery rehabilitation and surgeon and patient connectivity.

## XI. CONCLUSION

Orthopedic Surgery is one of the most common surgeries in the United states and it has a huge social and economic impact. Within the discipline of Orthopedics and Physical therapy, monitoring a patient post-surgery is essential to a healthy recovery and a lack thereof could diminish the effectiveness of the surgery itself. Medical complications can occur within days of a patient's release from a hospital which can be days before they are seen by a physician.

Traditionally, the means of acquiring patient data consists of frequent in-hospital visits and monitoring the patient there. This puts limits on both the practitioner and the patient. It is not practical for every patient to be seen every day. Yet within a few days, complications can occur. It is essentially a bottleneck where there is not enough time or doctors to continuously check on every patient. It is not practical either for a doctor to monitor a patient in the event that a complication might occur.

Many practitioners see a much-needed improvement regarding patient monitoring. They all have a mutual consensus that patients need to be monitored more frequently and more effectively. Advances in medical technology are growing, but there is no widespread solution to solving this problem that is accepted by all practitioners.

Consensus Orthopedics is attempting to tackle this problem with their product TracPatch. TracPatch allows a patient's physician to view data, interact with the patient, and update physical therapy routines that better suit the client based on their progress. They are able to measure a patient's range of motion through a serious of repetitious exercises. The device senses movement does a calculation and transmits this data over the internet to the cloud.

The product had increased its IMU's degrees of freedom and Consensus Orthopedics wanted to convert the data transmission from processed to raw. This created a huge stress on the Bluetooth module they have on their current version of TracPatch and noticed bottlenecking. A hardware replacement for the TracPatch may prove to be the best option for the product in order to accommodate the huge increase in data traffic.

The throughput is the main problem Consensus Orthopedics is facing. They are looking to introduce additional feature to the services they provide their practitioners and patients. To do so, they need to collect more data over the course of a patients' day. They were not able to transmit raw sensor data continuously. After meeting with Consensus Orthopedics and discussing the limitations of their product and the contributions we could offer, our sponsorship was established.

What we intend to do as a team is work with Consensus Orthopedics to develop a similar but new product that gathers significantly more data and provides additional functionality.

We're providing Orthopedic Surgeons and Physical Therapists more metrics about a patient's post-surgery behavior to improve patient care. To do this, we are working with Consensus Orthopedics to improve their existing monitor device-TracPatch-to increase the volume of data transmitted.

One deliverable our team is providing is raw data transmitted continuously. With a large volume of raw data transmitted continuously, as opposed to processed data points passed selectively, more metrics about a patient's activities can be developed and provide more insight for practitioners.

We will develop the infrastructure to transmit large amounts of sensor data for analysis in order to monitor and improve post-surgical patient recovery.

Our efforts can offer patients a faster discharge and provide practitioners insight on their patients' recovery from surgery to present day and allow early detection of complications.  In addition, practitioners would be able to remotely access patient data for all their patients which saves time and resources.

Our team will develop an infrastructure to transmit large amount of raw sensor data for analysis in order to monitor and improve post-surgical patient recovery. We will tackle this issue by creating two prototypes each consisting an IMU board and BLE module. IMU board we will be incorporating in our prototype is DK-20948, which is an ICM-20948 IMU with a 9-axis motion sensor composed of 3-axis accelerometers, magnetometers and gyroscopes. DK-20948 will be used to collect the data from the patients which then will be transmitted through the Bluetooth 5.0 module. Bluetooth module used in the prototype is Taiyo Yuden's EKSHSNZWZ evaluation kit. This kit uses the EYSHSNZWZ Bluetooth low energy modules consumes 80% less power than the nRF51 series by using a Nordic Semiconductor nRF52832 SoC.

Second side of our design is the receiver side, once the collected data is transmitted it will be stored in a Microsoft surface go (tablet) in a CSV file. On this Surface go tablet we will build a GUI application that will allow us to view the data that we are collecting, transmitting and storing. This received data will be roughly 51 MB of data per file. There will be two CSV files each day, we will sync this 102 MB of data based on their time-stamped data in timely manner.

Once we are successful in creating a working prototype it will enable aggregating this raw data into large pools of data. A machine learning infrastructure can be developed to analyze the specifics. Once patterns can be pinpointed, the user will no longer be required to input what exercise they are doing. Based on their raw data output from the sensors, a profile can be put together for the individual that charts their type of activity, along with time and effort put into that action. This can be analyzed on a larger scale from the practitioner perspective and determine if the patient is performing as expected post-surgery.

Our team's sponsor Consensus Orthopedics is currently only able to utilize their product for the post knee surgery rehabilitation, our prototype will allow them to expand their product use to post shoulder, hip and even spine surgeries.

REFERENCES

[8] Tracpatch.com. (2018). *TracPatch – Wearable Device for Post-op Total Joint Patients. – TracPatch – Wearable Device for Post-op Total Joint Patients.*. [online] Available at: https://www.tracpatch.com/ [Accessed 24 Sep. 2018].

[7] Physical Therapy Products. (2018). *TracPatch Wearable Device Allows Remote Postsurgical Patient Monitoring - Physical Therapy Products*. [online] Available at: http://www.ptproductsonline.com/2017/03/tracpatch-wearable-device-allows-remote-postsurgical-patient-monitoring/ [Accessed 24 Sep. 2018].

[6] Semiconductorstore.com. (2018). *Development Kit for ICM-20948 9-Axis Motion Sensor*. [online] Available at: https://www.semiconductorstore.com/Pages/asp/Item_MoreImages.asp?ItemNumber=DK-20948 [Accessed 24 Sep. 2018].

[5] Semiconductorstore.com. (2018). *Bluetooth 5 Ultra-Small Module, 802.15.1 BLE 5.0 Eval Kit*. [online] Available at: https://www.semiconductorstore.com/Pages/asp/Item_MoreImages.asp?ItemNumber=EKSHSNZWZ [Accessed 24 Sep. 2018].

Amaro, J., Patrao, S., Moita, F. and Roseiro, L. (2017). Bluetooth low energy profile for MPU9150 IMU data transfers. *2017 IEEE 5th Portuguese Meeting on Bioengineering (ENBENG)*.

[1] Maradit Kremers, H., Larson, D., Crowson, C., Kremers, W., Washington, R., Steiner, C., Jiranek, W. and Berry, D. (2015). Prevalence of Total Hip and Knee Replacement in the United States. *The Journal of Bone and Joint Surgery-American Volume*, 97(17), pp.1386-1397.

Weinstein, A., Rome, B., Reichmann, W., Collins, J., Burbine, S., Thornhill, T., Wright, J., Katz, J. and Losina, E. (2013). Estimating the Burden of Total Knee Replacement in the United States. *The Journal of Bone and Joint Surgery-American Volume*, 95(5), pp.385-392.

A Nation in Motion. (2018). *Value of Orthopaedic Treatment: Knee Replacement - A Nation in Motion*. [online] Available at: https://www.anationinmotion.org/value/knee/ [Accessed 23 Sep. 2018].

Imbalzano, Creazzo, Mandraffino, Sardo, & Saitta. (2013). Treatment failure of low molecular weight heparin in post-surgery orthopedic case. *International Journal of Cardiology, 169*(3), E36-E37.

Cremeans-Smith, J., Krupko, T., Greene, K., & Delahanty, D. (2013). Predicting symptoms of post-traumatic stress among patients undergoing orthopedic surgery on the basis of routinely collected cardiovascular data. Journal of Health Psychology, 18(1), 55-64.

Eysenbach, G., Marchal, F., Karunanithi, M., Price, M., Adams, Z., Semple, J., . . . Metcalfe, K. (2015). Using a Mobile App for Monitoring Post-Operative Quality of Recovery of Patients at Home: A Feasibility Study. *JMIR MHealth and UHealth, 3*(1), E18.

[3] R. Math and A. Goje, "Impact of IoT (Internet of Things) on Growth of Cloud Applications in Service Sector with Specific Reference to Western Maharashtra - IEEE Conference Publication", Ieeexplore.ieee.org, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/7819688/. [Accessed: 23- Sep- 2018].

[2] E. Ahmed, I. Yaqoob, I. Hashem and A. Vasilakos, "The role of big data analytics in Internet of Things", Researchgate.net, 2018. [Online]. Available: https://www.researchgate.net/publication/317617290_The_role_of_big_data_analytics_in_Internet_of_Things. [Accessed: 23- Sep- 2018].

M. Praveen Kumar, S. Santoshkumar, T. Gowdhaman and S. Shajahaan, "A SURVEY ON IoT PERFORMANCES IN BIG DATA", Pdfs.semanticscholar.org, 2018. [Online]. Available: https://pdfs.semanticscholar.org/897c/2a86c5a44784e71280e0a039cd4c88ac603c.pdf. [Accessed: 23- Sep- 2018].

[4] Amarinfotech.com. (2018). 10 Major Differences - Bluetooth 5 vs 4.2 - Feature Comparisons. [online] Available at: https://www.amarinfotech.com/differences-comparisons-bluetooth-5-vs-4-2.html [Accessed 24 Sep. 2018].

[9] C. Rider, *Flow Chart for BLE module main, Sacramento, 2019*

[10] D. Rabago, *Flow Chart of project, Sacramento, 2019*

[11] D. Rabago, *Risk Table, Sacramento, 2019*

[12] DTI (team 7), *Feature & assignment table, Sacramento, 2019*

"Gait Analysis." *Wikipedia*, Wikimedia Foundation, 9 Feb. 2019, en.wikipedia.org/wiki/Gait_analysis. Accessed 24 Feb 2019

[13] [14] [15] [16] *Bluetooth Market Update*, Bluetooth Special Interest Group, www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=2ahUKEwiYtf37-tXgAhWoIDQIHSnqCGMQFjABegQICRAC&url=https://www.bluetooth.com/~/media/Files/Marketing/Bluetooth_Market_Update_2018&usg=AOvVaw3NLkDYmWluRiJAoOw8I4Xn. Accessed 24 Feb 2019

[17] Jérémie Bouchaud | March 19, 2014. *RESEARCH NOTE - 86 Percent of Businesses Surveyed to Use Video Conferencing as Part of Their UC Environment by 2018 - IHS Technology*, technology.ihs.com/495121/mems-combo-sensors-continued-potent-growth-last-year-samsung-was-top-buyer. Accessed 24 Feb 2019

[18] “The State of Bluetooth in 2018 and Beyond.” *Bluetooth Technology Website*, blog.bluetooth.com/the-state-of-bluetooth-in-2018-and-beyond Accessed 24 Feb 2019

[19] “About Us.” SymmetryElectronics.com, www.semiconductorstore.com/blog/2014/A-Bluetooth-Low-Energy-Application-Defining-the-BLE-Stack/883/.

ACCESSED 3 MAR 2019

[20] “Introduction to Bluetooth Low Energy.” Memory Architectures | Memories of an Arduino | Adafruit Learning System, learn.adafruit.com/introduction-to-bluetooth-low-energy/gap.

ACCESSED 3 MAR 2019

[21] “Developer Help.” TCP/IP Five-Layer Software Model Overview - Developer Help, microchipdeveloper.com/wireless:ble-link-layer-connections.

ACCESSED 3 MAR 2019

[22] Davidson, Robert, et al. “Getting Started with Bluetooth Low Energy.” O'Reilly | Safari, O'Reilly Media, Inc., www.oreilly.com/library/view/getting-started-with/9781491900550/ch04.html.

ACCESSED 3 MAR 2019

[23] DTI (team 7), *Product Picture, Sacramento, 2019*

[24] “Take a Simpler Route to Low-Power, 9-Axis Motion Tracking .” *ArticleLibrary*, Digi-Key, 21 Feb. 2108, Take a Simpler Route to Low-Power, 9-Axis Motion Tracking . Accessed 28 April 2019

[25] DTI (team 7), *Packet Bulk, Sacramento, 2019*

[26] DTI (team 7), *Clock Capture, Sacramento, 2019*

[27] DTI (team 7), *Sensor Read, Sacramento, 2019*

[28] DTI (team 7), *Discoverable Object, Sacramento, 2019*

[29] DTI (team 7), *Characteristics, Sacramento, 2019*

[30] DTI (team 7), *Individual Subscribe, Sacramento, 2019*

[31] DTI (team 7), *Three Subscriptions, Sacramento, 2019*

[32] DTI (team 7), *Sampling Rate Change, Sacramento, 2019*

# GLOSSARY

Ambulatory- people who are capable of walking; mobile, not bedridden.

BLE- Bluetooth low energy is the variant aimed to be used in small devices allowing internet of things concept to become more of a reality. The low energy being the key part in small device development.

CSV- comma-separated values is a type of file format. The values within the data are separated with a comma as the delimiter.

IMU- inertial measurement unit is an electronic sensor that can measure multiple areas including force, angular rate, magnetic field, using a combination of accelerometers, gyroscopes, and magnetometers.

IOT- internet of things is a concept where many physical devices are interconnected on the internet allowing data to be exchanged and processed.

LMWH- Low-molecular-weight heparin is used in prevention of blood clots and treatment of venous thromboembolism and myocardial infarction.

Metric- a system or standard of measurement.

Orthopedic Surgeon- A surgeon who specializes in diagnosis and preoperative, operative, and postoperative treatment of diseases and injuries of the musculoskeletal system.

Orthopedic Surgery- Surgery that focuses on areas such as musculoskeletal trauma, spine diseases, sports injuries, degenerative diseases, infections, tumors, and congenital disorders.

Practitioner- a person who engages in discipline, particularly in medicine.

ROM- range of motion or range of travel refers to the linear or angular distance a moving object travels while attached to another.

TKA- total knee arthroplasty or knee replacement is when the knee is replaced to relieve pain and disability.

# Appendix A
## User Manual

1. Patient Use
    a. Download the DTI app from the apple store on your mobile device.
    b. Unplug device from power outlet if charging
    c. Start up the DTI application on mobile device
        i. Application will automatically detect device and connect
        ii. If device can't be connected, verify power is on. Reset button can be pressed as well.
    d. Strap device onto desired limb.
    e. Device will now collect data passively.
    f. Data can be accessed in documents folder of mobile device
2. Technical use
    a. Navigate to Nordic infocenter: Getting started
        i. Setup environment for Keil
        ii. Connect SWD debugging cable to BLE module and PC
        iii. Upload code
        iv. Connect power source

# Appendix B
# Hardware

Block diagram & documentation at block level. Schematics and documentation to component level.

IMU Sensor on DK-20948 Evaluation Board

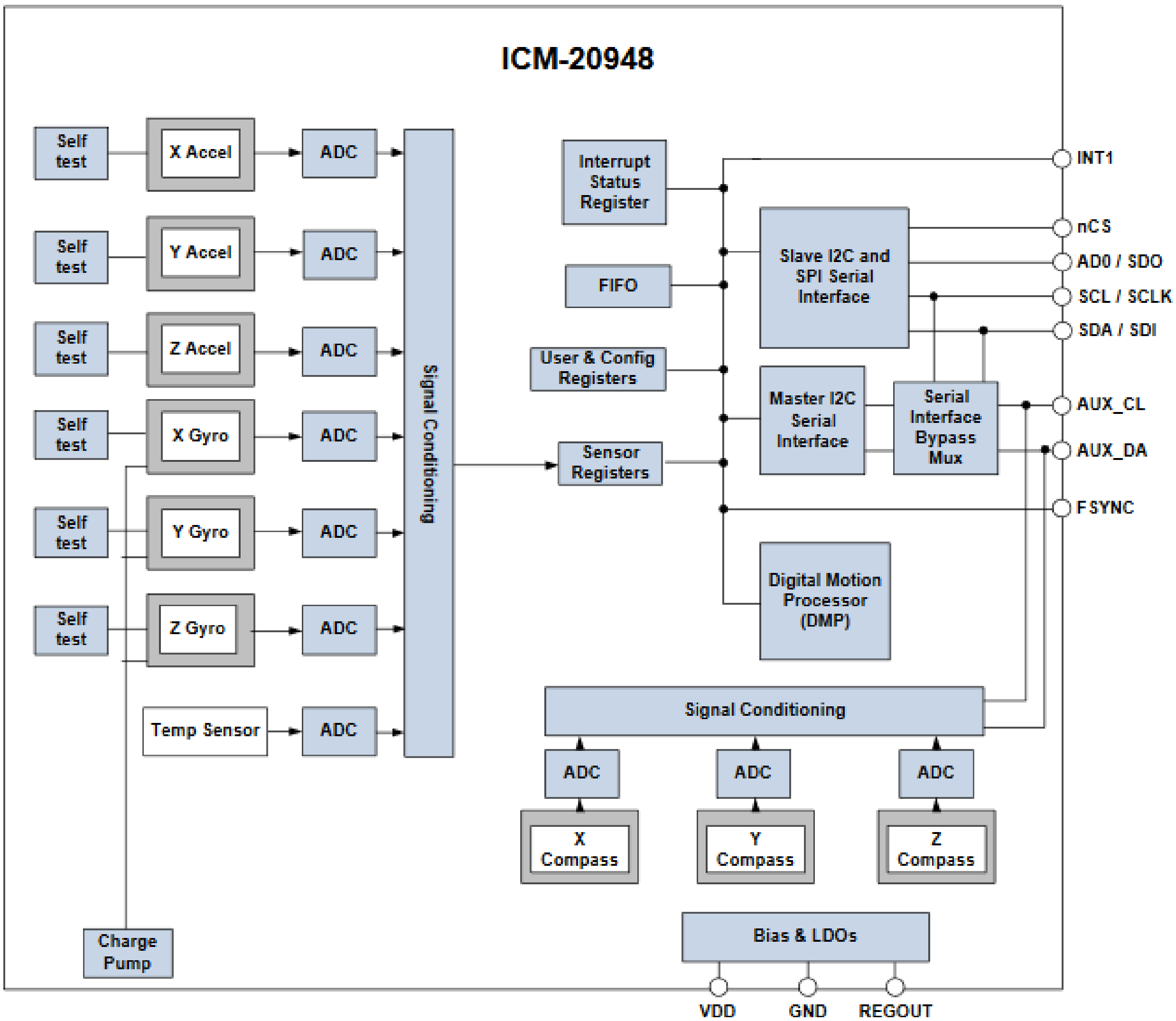


*Figure 20 [24]*

BLE Module on BLE Evaluation Board

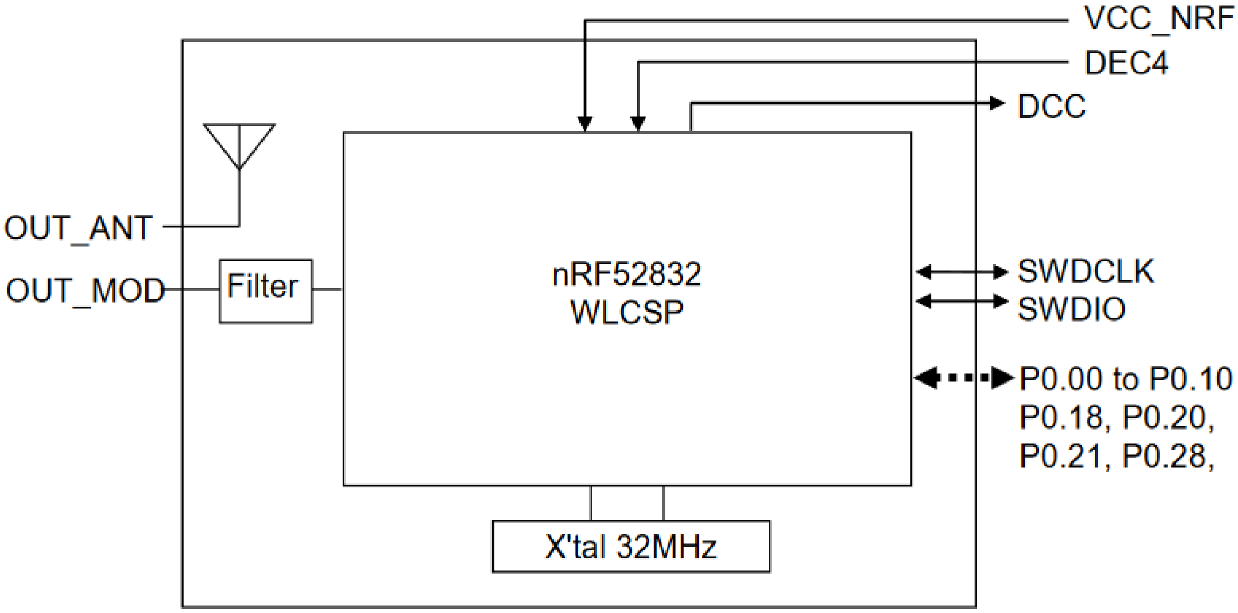


*Figure 21 [24]*

# Appendix C
# Software

Block diagram & documentation at block level. Flowcharts, pseudo-code, and documentation to subroutine level.

BLE module main operation

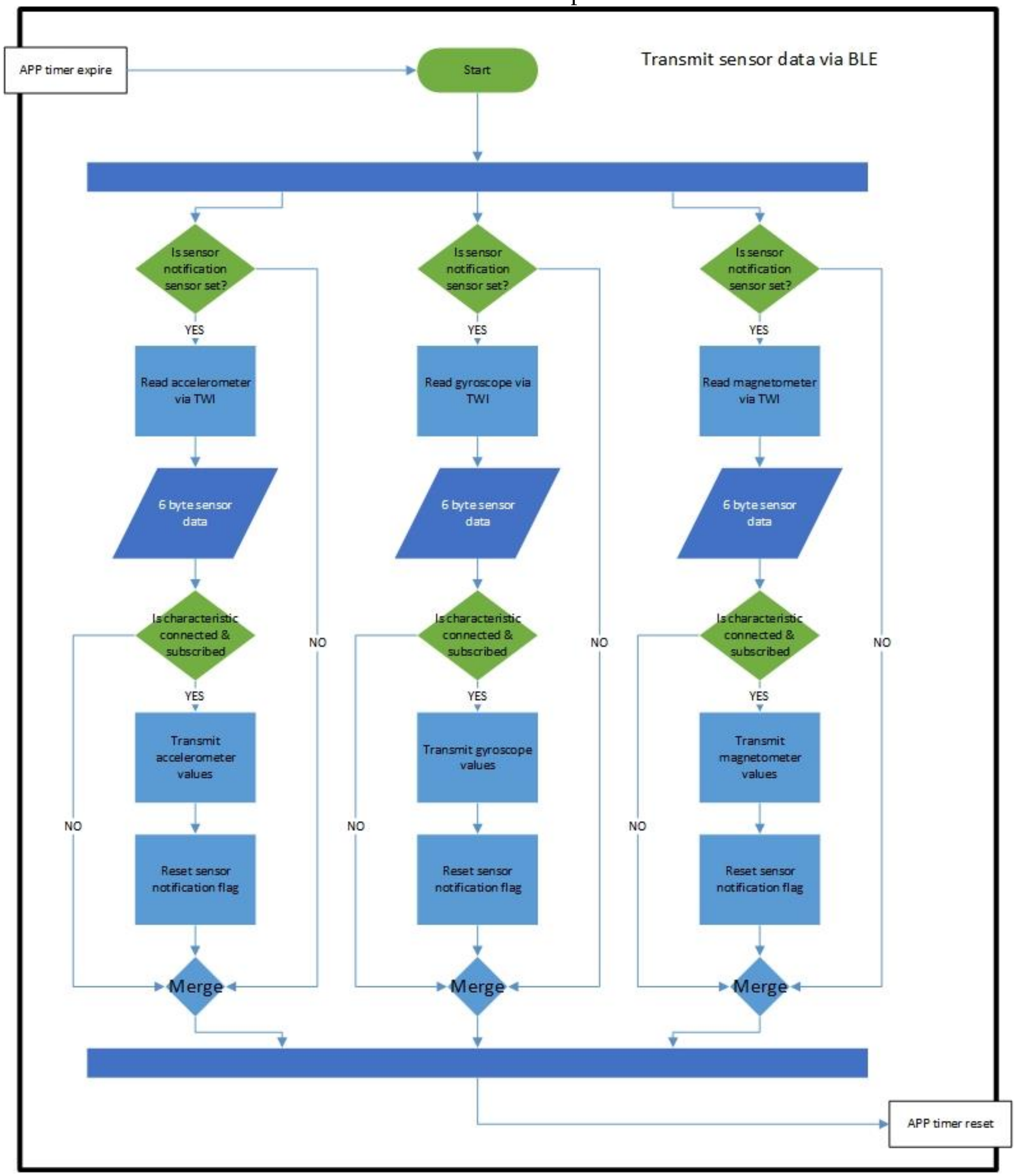


*Figure 22 [9]*

# Appendix D
# Mechanical

There were no mechanical portions of our project.

# Appendix E
# Vendor Contacts

**Symmetry Electronics**

Office: 310.727.3022
Mobile: 310.363.5035
SymmetryElectronics.com

**Consensus Orthopedics**

1115 Windfield Way, STE, 100

El Dorado Hills, CA 95762

E: ahabib@consensusortho.com

P: 916.355.7100

**Device Lab Inc.**

www.devicelab.com

3002 Dow Ave. Suite 124 | Tustin CA 92780

Tel: 714-263-0878 | Mobile: 949-338-4308

# Appendix F
# Resumes

## Cody J. Rider

**OBJECTIVE**

To obtain a position involving design analysis, tools flow, verification, script &/or tool development, and other engineering duties.

**EDUCATION**

**Bachelor of Science, Computer Engineering** Expected: May 2019
California State University, Sacramento, CA Current GPA 3.87, Senior Standing

**SKILLS**

| | |
|---|---|
| **Hardware Description Languages:** | VDHL, Verilog |
| **Scripting/Programming Languages:** | C, C++, Python, Java, Assembly (x86, ARM), MATLAB |
| **Engineering Tools:** | Xilinx ISE |
| **Tools/Packages:** | PSPICE, MS Office, MS Visio, Multisim |
| **Platforms/systems:** | Windows, Unix, Bash, Debian |
| **Electronic Test Equipment:** | Multimeter, Oscilloscope, Frequency Counter, Signal Generator, Distortion Analyzer, Modulation Analyzer, Power Supply |
| **Organization and Communication:** | Strong team communication, self-motivated, high attention to detail, analytical, excellent oral & writing capabilities |

**PROJECTS**

**Hornet Hyperloop Pod Prototype**

- Part of a 3-person team building a scaled-down prototype focused on automation of travel. Led design and implementation of motor control through H-bridges, and CAN bus communications for sensor inputs. Sensor and motor input/output interface in Python.

**WORK EXPERIENCE**

**Missile & Space Systems Electronic Maintenance Tech/Team Lead/Trainer** USAF, Great Falls, MT 2008-2014

- Maintained nuclear weapon capabilities by troubleshooting and repairing weapon systems such as security, cooling, communications, power distribution, and cables.
- Extensive use of Electronic equipment and schematics for testing electrical components, equipment, and systems.

**AWARDS**

**USAF** - 2009 Spring Airman of the quarter.
**USAF** - Recipient of early promotion award "Below the Zone" for distinguished work ethic and leadership.

**AFFILIATIONS**

**Power & Controls Lead** Hornet Hyperloop, CSUS, CA Oct 2017-May 2018

- Leading the controls & power team of the Hornet Hyperloop to design the controls subsystem for the pod including navigation system, telemetry readings, and speed control.

**INTERNSHIP**

**Hardware Design Intern** Aruba Hewlett Packard Enterprise, Roseville, CA May 2018-Aug 2018

- Developed margin testing infrastructure for ASIC products. IOT focused interconnection of devices with telemetry collection and temperature control from a UI.

# David Rabago

## EDUCATION :

Bachelor of Science in Computer Engineering — December 2019
California State University Sacramento — GPA: 3.66

## LANGUAGE COMPETENCIES :

Java, C, Verilog, HTML, CSS, UNIX/Linux, Assembly(x86), JavaScript(limited), PHP, Python(limited).

## CONCEPT COMPETENCIES :

Embedded Systems, Microcontrollers, Circuit Analysis, Version Control, Object Oriented Programming,

## WORK EXPERIENCE :

**Software Intern at BlackRock** — May 2018-August 2018

Worked to expose connectivity and provide the company a way to access data more easily. Collaborated with multiple teams across company, involved early in design process, and took lead role to integrate with company data layers. Provided nonexistent functionality to several teams and business leaders.

+ Built an API to run db queries, populate fields of POJO, enter this POJO as a record in a Cassandra repo. Restructured the current data structure to maximize effectiveness, then automated this process.

+ Created API to search against the company's data model by leveraging integration with existing data layers.

+ Constructed http service method that returned objects related to a search against a Cassandra/Solr database. Provided nonexistent search functionality against the company's data model to front-end developers.

**Software Intern at VSP Global** — May 2016-August 2016

+ Wrote unit tests for new functionality, edited existing test for modified functionality.

+ Built a Splunk dashboard to visually represent the performance of an API.

+ Used JBehaves to test scenarios of code to ensure proper functionality.

+ Edited JSON files to accommodate changes made to JBEhave scenarios.

## PROJECTS / EXPERIENCE :

**Micromouse Competition** (1st Place winner) — September 2016 - November 2016

+ Implemented encoders, infrared sensors, and ultrasonic sensors to navigate an arduino mouse.

**ACM Game Developer's Competition** (1st Place winner) — March 2016

+ Built an interactive, two-player, 2D game with a team of three using LUA and LOVE frameworks.

**Circuit Solver** (Personal Project) — February 2016

+ Created a java program to solve a circuit by reducing it to its smallest equivalent circuit.

## AFFILIATIONS :

**Tau Beta Pi** (Honors Society Member) — November 2016-Present

**National Action Council for Minority Engineers** (NACME Scholar) — August 2016-Present

**Society of Hispanic Professional Engineers** (Regional Representative+Webmaster) — August 2016-July 2018

+ Counseled chapter leaders across entire west-coast region and facilitated regional conference, tournament, etc.

+ Lead a committee to design the current website from scratch using pure html and css.

**Engineering and Computer Science Joint Council** (Vice Expo Chair) — Spring 2016

**California Boys State Representative** — Summer 2014

Bryan Strong

## GOAL

Obtain a position in control systems engineering or applications engineering

## PROJECTS

### Micro Mouse

- Develop Micro Mouse platform
- Decide on Micro-Controller
- Program in C for Micro Mouse controls

### Machine Vision Water Tank

- Use Machine Vision to monitor water level
- Use PID controls to control water level using water pumps
- Program Arduino controller for control system

## EXPERIENCE

**Engineering Prototype Assembler** Current

*Weslan Systems*

- Read blue prints
- Fabricate hardware
- Assemble, wire and test clean room fan assemblies

**Math Tutor** 1/1/2016-1/1/2018

*Cosumnes River College*

- Tutor students in math
- Attend mandatory tutor meetings

## EDUCATION

**Sacramento State College** Current

*Sacramento, California*

- Electrical Engineering (Control Systems)