Technical Documentation

CC3501 – Group 6

Group members: Keion Larsen, Matthew Duggan, Alex Philp

Contents

IMU Project 3

MK22FN1M0VLH12 3

FXOS8700CQR1 Accelerometer and Magnetometer 3

L3GD20H Gyroscope 4

SWD Connector 5

Coin cell Holder 5

Bluetooth RN4871 Click Module 6

ABM3B-8.000MHZ-B2-T Crystal Oscillator 7

PCB Design 8

Software 10

Appendix 12

IMU Project

The project that has been designed is a wearable inertial measurement unit (IMU). The IMU will be using multiple sensors that will detect users' motion and record acceleration, direction, and orientation.

MK22FN1M0VLH12

The IMU will be using the MK22FN1M0VLH12 (see Fig.6) microprocessor. The K22 is a cost-effective, 32-bit MCU being used to process and communicate the data from the sensors. For the power inputs, all VSS and the VSSA pins will be routed to ground as well as the VREFL pin. VDD and the VDDA pins will be routed to the P3V3 line and the VREFH will also be connected to the P3V3 line with a decoupling capacitor of 0.1uF (C3). The VBAT line pin will also be used by the coin cell holder. The RESET PIN 34 of the microprocessor will also be used by the SWD connector. Ports A0 and A3 will be used for JTAG_TCLK and JTAG_TDI respectively when debugging the microprocessor. Ports A18 and A19 are both GPIO/Analog pins which will be used for the crystal oscillator Port B2 and B3 will be used for SCL and SDA lines of both the accelerometer and magnetometer, and the gyroscope for the I2C components. Ports D0 and D1 are both used for the programmable interrupts for the accelerometer and magnetometer.

FXOS8700CQR1 Accelerometer and Magnetometer

This component is FXOS8700CQR1 (See Fig. 1 U3) which uses the accelerometer to measure the acceleration of a structure. The use of the acceleration is limited as it cannot be used for fine measurements of the rotation. The magnetometer is also used as a compass to describe the direction of the acceleration which will be useful in translating physical movements going in multiple directions. The compass is also required to ensure the direction is always known as the accelerometer and gyroscope can drift over time and lose its orientation. Due to a compass being limited to a 2D plane, the use of a gyroscopic component is required for data in a 3D plane. Component U3 is an I2C component requiring a connection to SCL/SDA lines for data transmission to the microcontroller. The SDA port used through the microcontroller is port B3 while the SCL line is used by port B2. Both the SDA and SCL lines require a pull-up resistor to the 3.3V line of 4.7kΩ (See Fig 2. Resistors R2, R3) to keep the given input high. Both interrupt pins, INT1 and INT2 have been connected to ports D0 and D1 respectively connected via 0Ω resistors (R4, R5) which can be left out if the use of interrupts is deemed unnecessary. For this I2C circuit, slave address 0 has been wired to ground so that slave address 1 is used instead. Slave address 1 is also connected to the 3.3V lines through a 10kΩ pull-up resistor (R6). The reset pin has been wired to ground as well as the two reserved pins and the BYP line with a 0.1uF capacitor for decoupling and limiting. The VDD and VDDIO pins have been connected to the 3.3V lines. The CRST pin is also connected to ground through a 0.1uF (C13) capacitor.

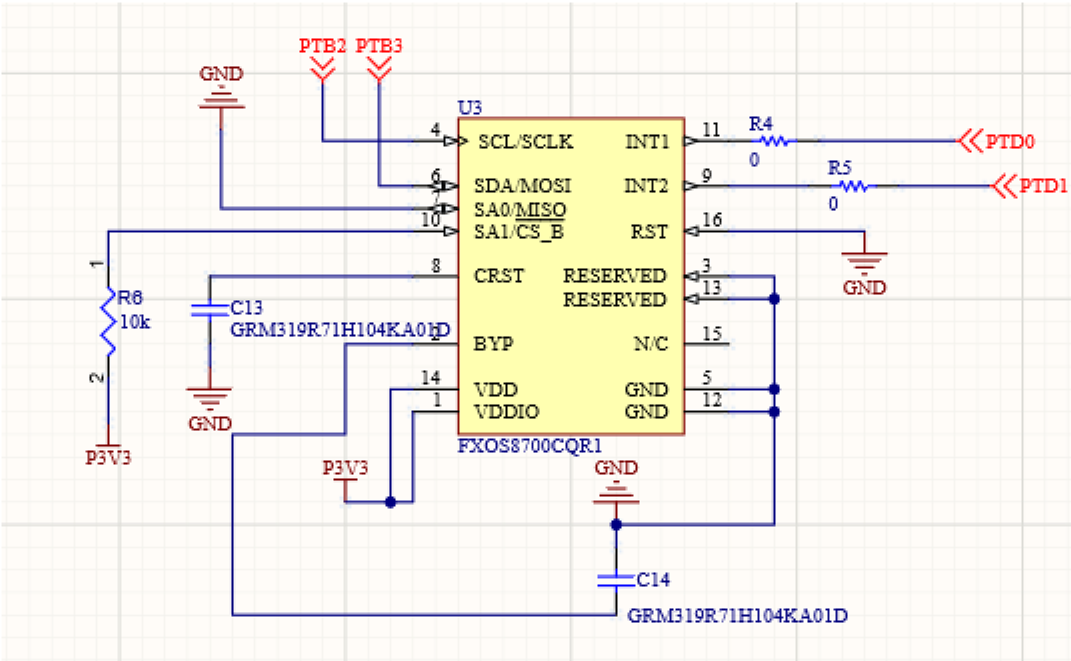


Figure 1: FXOS8700CQR1 Accelerometer and Magnetometer

L3GD20H Gyroscope

Component U2 (See Fig. 2) is the gyroscope which is the L3GD20H Gyroscope. The three-axis digital output gyroscope will be used to detect the changes in orientation of the axis of the IMU. The gyroscope will be used to calculate the orientation to make sure the IMU is changed to the right orientation. The gyroscope is also an I2C component which will require SDA/SCL lines for data transmission. These SCL/SDA lines will be connected to the same lines as U3's SDA/SCL lines these lines will not require additional pull-up resistors due to all I2C components connected in a parallel circuit. Slave Address 0 (SA0) will be used and connected to the 3.3V line. Pins CS and DEN as well as all RES pins and ground pins will be wired to ground. The CAP pin will also go to ground with a 0.01uF capacitor (C12). VDD_IO and VDD will be connected to the 3.3V line as well. Unlike the accelerometer, the gyroscope did not include the INT pins connected to the microcontroller. For the scope of this project, this was done to reduce the number of lines on the PCB design to keep it as small as possible. Although, it is recommended that programmable interrupts are included due to the gyroscope being a critical component for fine calculations in describing the motion of the user.

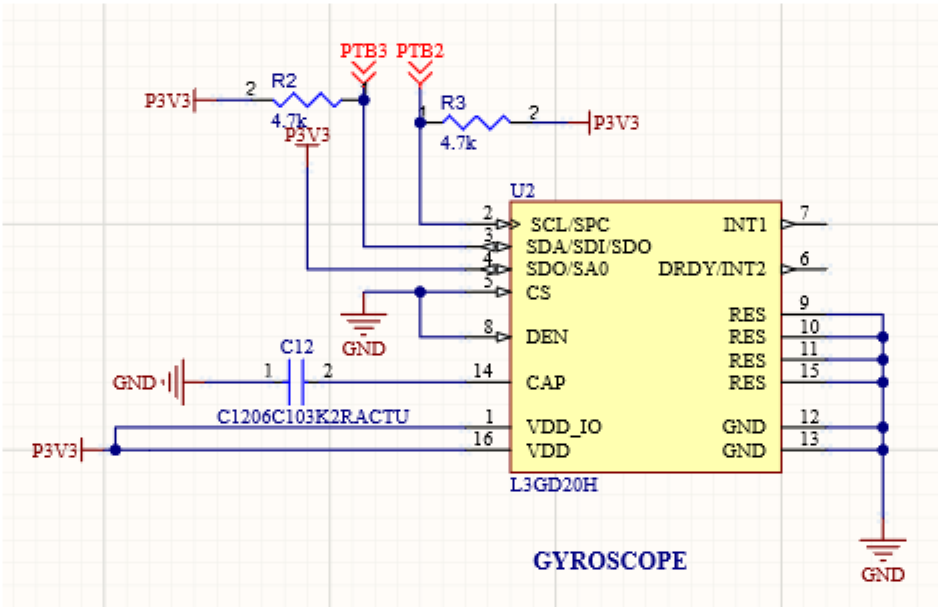


Figure 2: L3GD20H Gyroscope

SWD Connector

Component P3 (See Fig. 3) is a 1.27mm 2x5 through-hole header. This will be used as the SWD connector for debugging and programming the microcontroller directly. Pin 1 of the header will be connected to a 3.3V line while pin 3,5 and 9 are connected to ground. Pins 2 and 4 will be connected to Port A3 and A0 respectively which are connected directly to the microcontroller. A reset pin has also been allocated on pin 10 connected to RESET_B on the microcontroller. This is used to reset the components used by the MCU. For debugging, the AMR J-TAG connector will be used and programmed through Kinetis Design Studio 3.

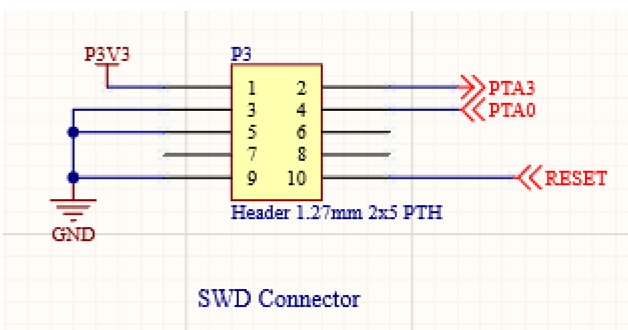


Figure 3: SWD Connector

Coin cell Holder

A coin cell holder (see Fig. 4) has also been implemented into the circuit which allows for a power supply through a coin cell battery. This is to ensure the IMU is portable and does not require a connection to an external power supply. The coin cell is also used to decrease the weight of the IMU and to allow for easier handling. The coin cells do have a drawback which affects how effective the coin cell is at lasting for as long as possible but the IMU is a low-powered device that will not be too demanding. Coin cell batteries are also non-reusable but inexpensive and easy to stack for higher voltages. A lithium battery/s capable of over 4.1V and 169 mA is required for the IMU to work effectively. The coin cell holder (BT1) is a 2 pins surface mount holder capable of holding size CR2032 batteries. The ground pins will go straight to ground with the VCC pin going straight through a 20V, 1A diode (MBR120VLSFT1G) component named D1 for current to flow away from the battery. D1 will cause a voltage step down of at least 0.7V which is why a voltage of at least 4.1V is required in order to compensate. After the diode, a step down 3.3V voltage regulator (VR1) has been implemented into the circuit to allow a higher voltage input to be regulated as a Lithium battery will not be capable of 3.3 volts exactly. The voltage regulator will have an input going through the first line to the voltage regulator which then goes to the output line. The second line parallel to input line will go through a decoupling capacitor of 0.1uF (C11) and then to ground. The decoupling capacitor is used to support the currents on the input path allowing the supply voltage to be suitable for operation. Another line from ground will be connected directly to the voltage regulator and another line will be connected to the output through a 1uF decoupling capacitor (C10). The output of the voltage regulator goes straight to a 3.3V line and the VBAT line to the microcontroller. Another design choice that is implemented into the schematics is the use of a 2x1 header (P4) connected from to ground in pin 2 and through a second diode D2 from pin 1. Pin 1 will be connected to the 3.3V line and VBAT for an optional 3.3V power supply that can be used instead of the coin cell.

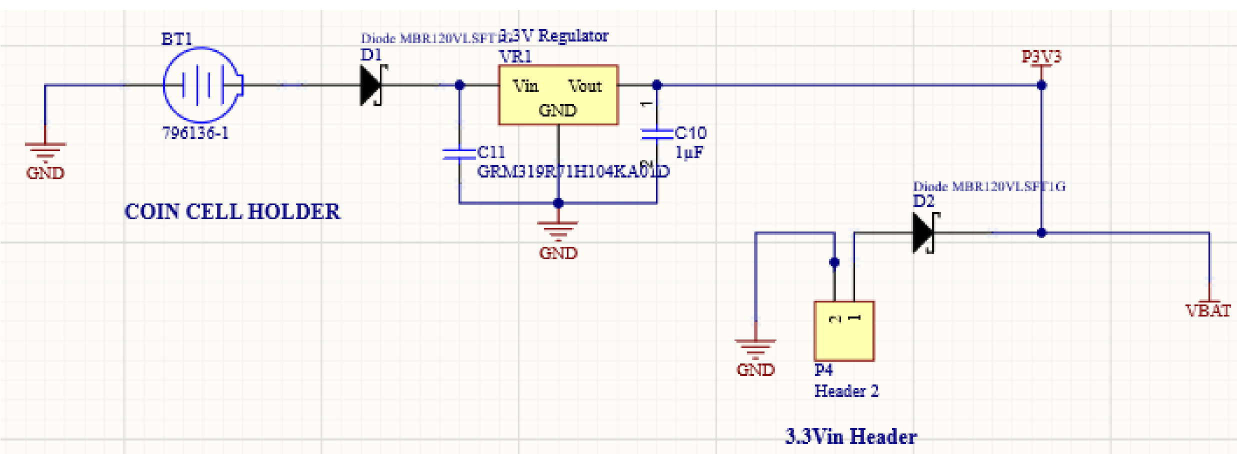


Figure 4: Coin Cell Holder

Bluetooth RN4871 Click Module

The system will also require a Bluetooth module to transmit the data wirelessly. This is a vital component and is required for reliably sending sensors data from the accelerometer, gyroscope, and magnetometer to the user interface. The Bluetooth RN4871 Click module will be used. The PCB will also require a shield which will be two 1x8 Headers (P1, P2) spaced apart for the module to fit. The TX and RX pins (P2, Pin 3 and 4) will be connected to ports D2 and D3 respectively for communication both ways between the module and user interface. The 3.3V and

GND pins (P1, Pins 7 and 8) of the module will also be connected to the 3.3V and ground lines, respectively.

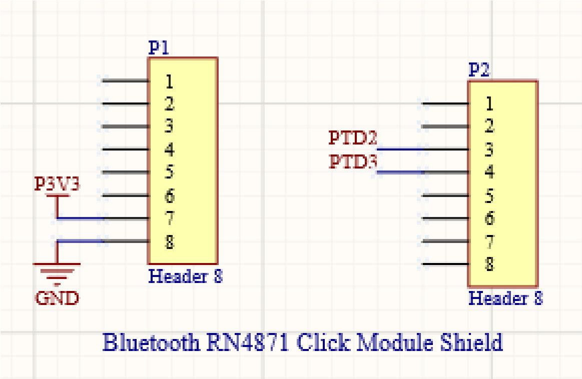


Figure 5: Bluetooth RN4871 Click Module Shield

Bluetooth is a highly critical component that determines a connection for the data acquired by the board which meant using a module was the safest way to achieve this and reduce the risk of an error when soldering the PCB. Using a USB connection is sufficient for data transmission, but it is limiting and requires a long cable which can limit the applications of the IMU and make it uncomfortable for users to wear. Zigbee also could have been used but was decided against due to Zigbee usually having 2.4 GHz or less compared to Bluetooth's 2.4 GHz or more. Zigbee does provide more distance than Bluetooth but Bluetooth provided an easier means of setting a personal network for the IMU's for consumers.

ABM3B-8.000MHZ-B2-T Crystal Oscillator

A crystal oscillator (See Fig. 6) will be used for the IMU as well. This will be used to generate clock pulses for the microcontroller to synchronize different parts of the circuit such as the gyroscope, accelerometer, and magnetometer. These components must be synchronized as the data processed will be time-sensitive in knowing when motions occurred. The oscillator will be using ports A18 and A19 on the microcontroller. The oscillator labelled X1 will be the ABM3B-8.000MHZ-B2-T. One of the GND pins will be connected to ground in parallel with a bypass capacitor of 22pf (C1) to the output of port A19. The second GND pin will be connected in the same manner to the input port A18 with a capacitance of 22pf (C2) as well. The input port will also use a 1MΩ resistor (R1) connected directly to the output port A19 in parallel to the oscillator.

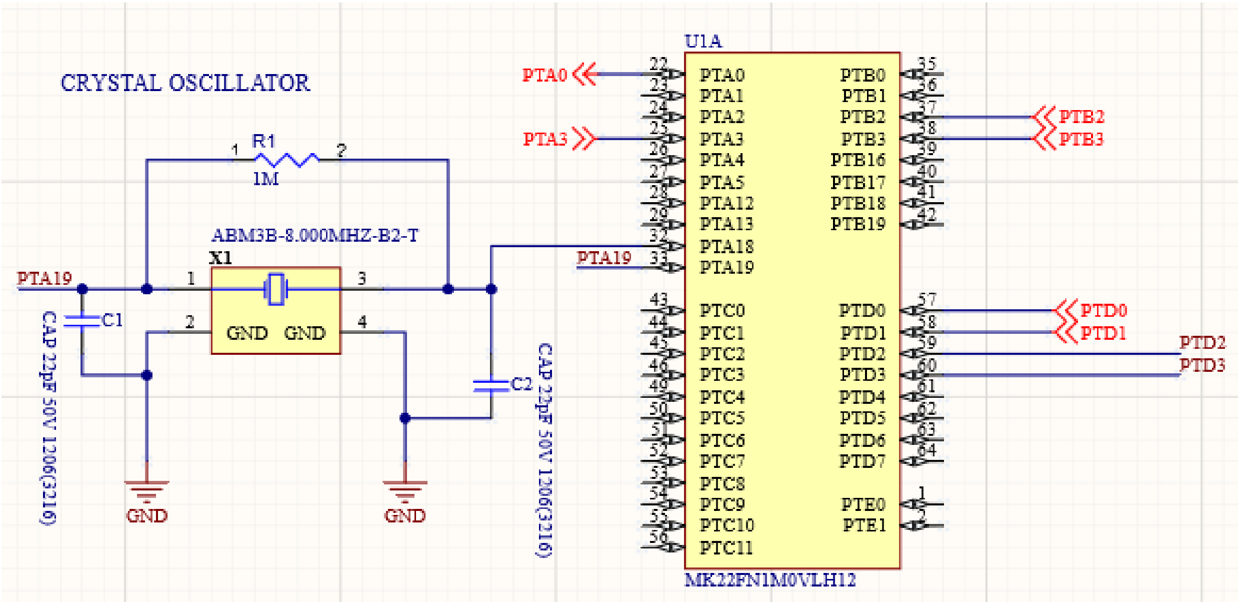


Figure 6: Crystal Oscillator

PCB Design

The design of the physical PCB is a square PCB with the microprocessor kept central to the gyroscope, accelerometer, and magnetometer. The PCB board also includes a row of capacitors connected to the P3V3 line for an additional decoupling capacitor if required. It would be more efficient in terms of reducing space on the PCB to get rid of some of the decoupling capacitors. The decoupling capacitors used should also be placed more effectively. Placement The shield for the Bluetooth module and the power supply are the largest components on the PCB and were also kept on opposite sides of the PCB. A more effective design choice that could be implemented would be to have only the Bluetooth microchip and required components on the PCB instead of a shield. This would reduce the amount of space used for a comfortable wearable electronic. It is possible to also remove the coin cell holder as well and opt for a separate wired 3.3V power supply. This will make sense if the users want to wear multiple IMU's connected to one power supply limiting the need for power management and separate batteries for each IMU. Reducing the coin cell battery would not make a big difference to weight but it would allow for a smaller and more wearable PCB. The shape of the PCB could also be changed to a more circular shape when removing the shield and coin cell holder which allow for more comfortability and flexibility when wearing the IMU.

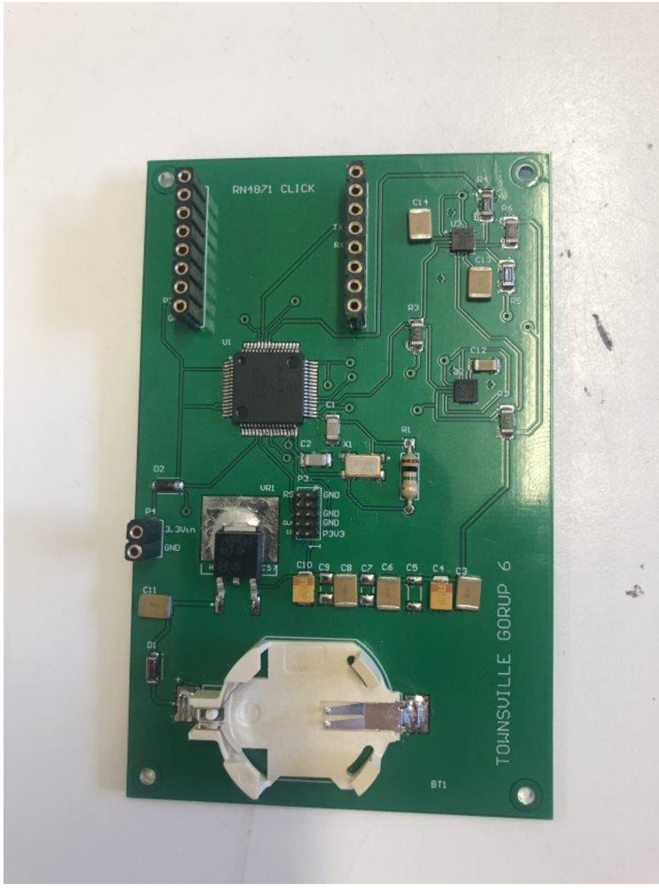


Figure 6: Soldered PCB

Issues did arise once the soldering of the PCB (See Fig. 6) was completed. Soldering the PCB was significantly delayed due to the undelivered MCU. These parts did not arrive until 2 days before the first live demonstration despite the availability of these parts on Element14. Once soldering the PCB was complete, issues arose with the voltage not being high enough for normal operation. When using a multimeter and testing the voltages at the coin cell battery a voltage of 2.7V would arise despite stacked 3V coin cells. A 3.3V supply was also connected through the power headers added which yielded the same results. It was assumed that a short was present in the PCB underneath the gyroscope chip (U2). This is most likely because the pins were not long enough as they were not visible underneath the chip. A recommendation is to ensure that the pins on the PCB come out from underneath the chip so that applying the gyroscope can be done easily despite the size of the chip and shorts can be rectified.

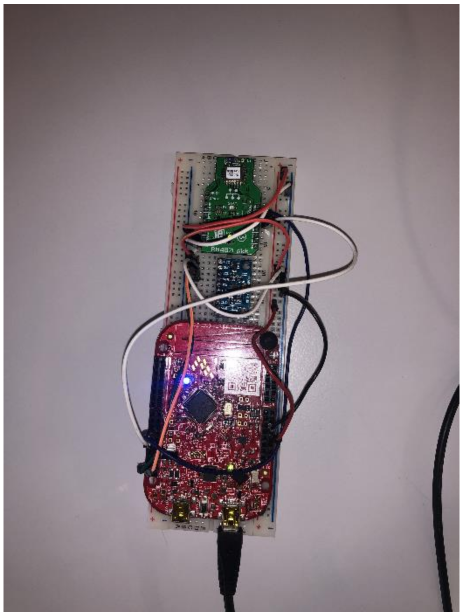


Figure 7: Breadboarded Solution

Software

The software for the IMU was completed in Kinetis Design Studio with a JTAG connector for debugging. Through Kinetis, a new Processor Expert project was created to allow the necessary components and code to be generated for operation.

A breadboarded device (See Fig. 7) was created using a FRDM-K20D50M board, RN4871 Bluetooth module and MPU-6050 breakout board. The breakout board contains an accelerometer and gyroscope and this was used to test the approach for converting the accelerometer and gyroscope values to the Euler angles: pitch, roll and yaw.

In an infinite loop, the program reads data using an InternalI2C Processor Expert component, first sends a 0 to the appropriate gyroscope address to use the gyroscope's internal oscillator, then reads the data from the 6 accelerometer and 6 gyroscope addresses, and converts these to signed 16-bit values. The gyroscope values are divided by 131, which was specified in the MPU-6050's datasheet, to yield a value in degrees/second. The accelerometer readings did not have to be scaled to be in proper units as these are converted to Euler angles using ratios in inverse trigonometric functions.

The accelerometer readings are then converted into pitch and roll using an arctan function. Yaw is unobtainable from the accelerometer values as that angular movement does not change the gravitational vector on the accelerometer. The gyroscope's angular velocity values are also converted to the three Euler angles, using a 32-bit free counter Processor Expert component to keep track of elapsed time between readings and multiply that by the recorded velocity.

The pitch and yaw values are then combined using 70% of the values found from the gyroscope and 30% of the values from the accelerometer. This is to minimize the erroneous readings from the accelerometer caused by vibrations, and to eliminate the gyroscope's drift. The final yaw value is solely from the gyroscope so does have some drift, but the magnetometer's reading would have been used if a functioning PCB was successfully developed.

The data is then sent using the Bluetooth module's UART interface to be received by the Raspberry Pi and over serial. The serial interface would not have been accessible using the PCB implementation but this was used developed to develop a MATLAB script which reads the values and plots them in real time. Further development time would have been used to convert this into something that could run on the Pi to give live plots of the data received from Bluetooth.

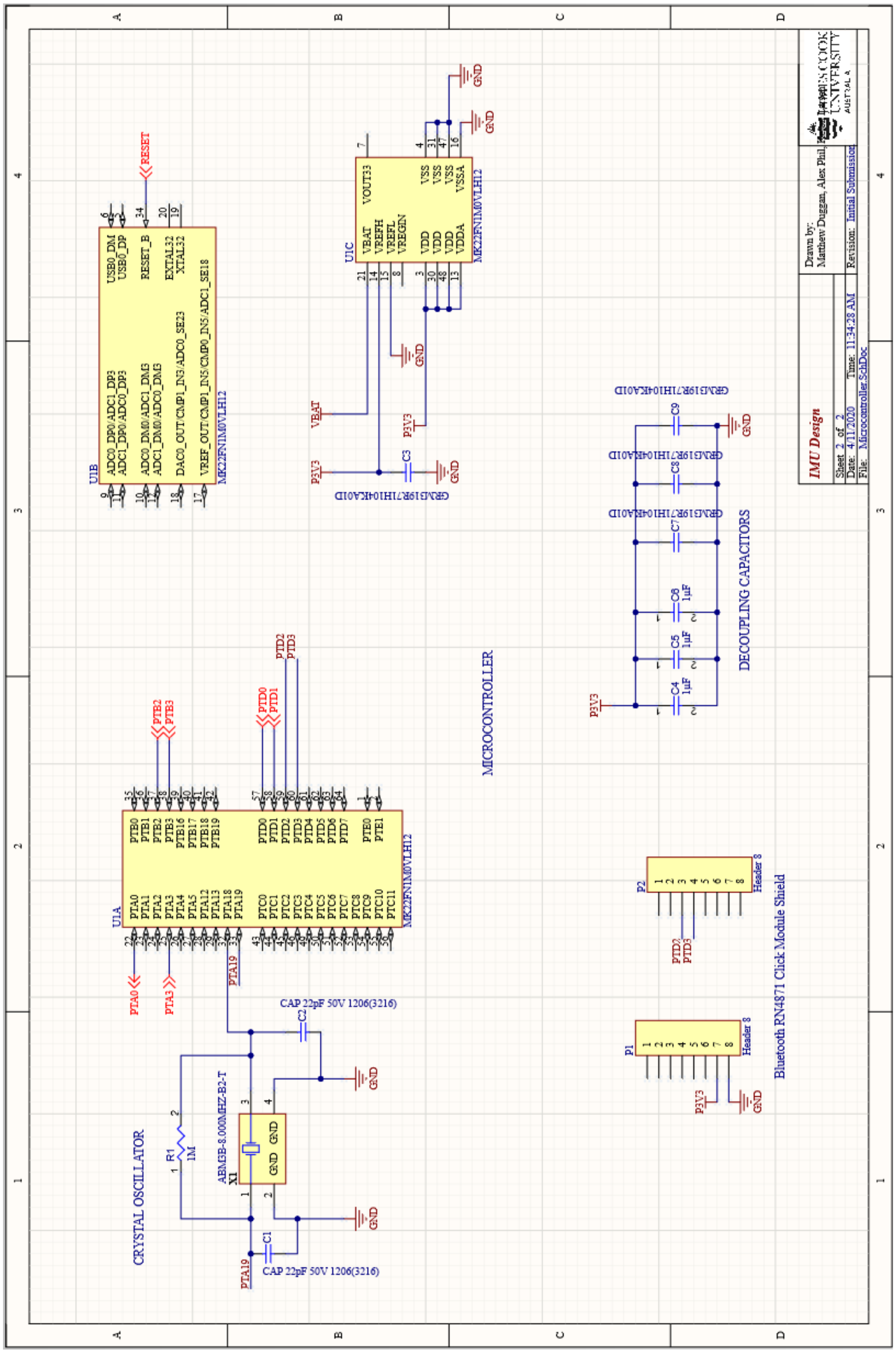
This code for the breadboarded implementation also contains some basic error checking. If the first message to the I2C device is not correctly received by the device, an error message is sent both to serial and over Bluetooth, and the program will stop looping. Without this, if a wire connecting the microcontroller to the MPU-6050 was disconnected, the program would continue running with erroneous values, and this could have been extended to the PCB implementation to give warnings if the microcontroller was not communicating correctly with either I2C device, and detail which it was.

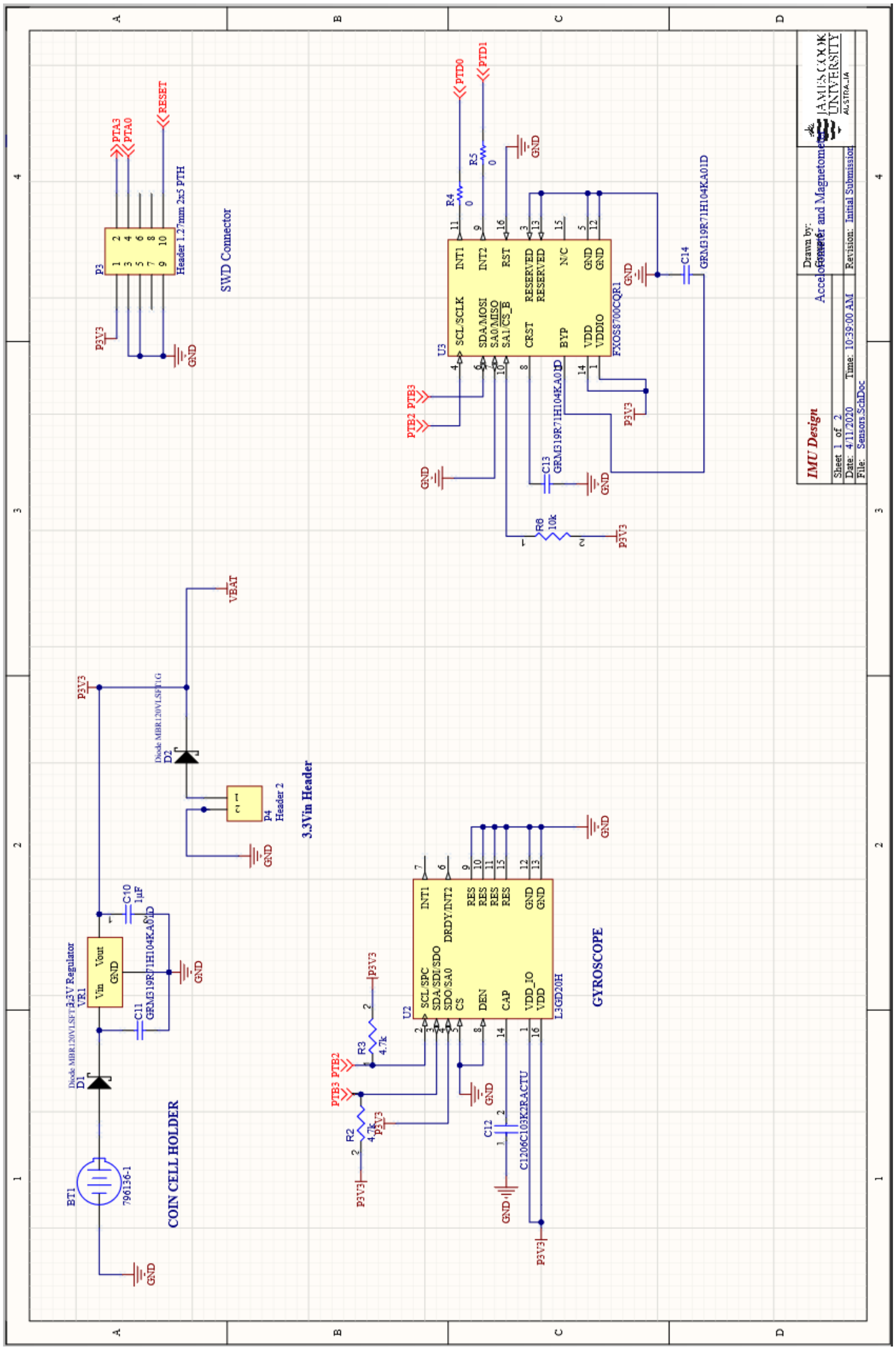
Since a different component was used for the breadboard implementation due to the unavailability of breakout boards of the accelerometer/magnetometer and gyroscope components used on the PCB, several changes to the code would be made if a functioning PCB was developed. The slave address of the gyroscope would be changed to 6A in hexadecimal, with the values stored in the six registers starting from 28 hexadecimal. The least significant byte and most significant byte are also switched compared to the MPU-6050. A second InternalI2C component would be added for the combined magnetometer and accelerometer, with its slave address being 0x1E. The accelerometer values are stored in the six registers starting from 0x01, while the magnetometer readings start from 0x33.

The IMU will use Bluetooth connectivity for data transmission. This wireless approach combined with its small size allows more practical applications of the IMU, such as for measuring the movements of athletes for analysis. A Raspberry Pi is used to pair with the IMU which will then receive incoming data as it comes from the PCB (every tenth of a second). Once new data has been received, the Raspberry Pi will attach it to a file called "IMUData.csv" and print it over the serial, allowing the user to view changes in real time and access them later. Initially the program was going to update a website with the gathered data, however the rate limit of one update per 15 seconds created an issue. Using the site would mean either only receiving data once every 15 seconds, which would limit its usefulness as an IMU, or collecting data and updating the site afterwards, which meant that the Pi would have to run for up to an hour to record even 15 seconds of data. Eventually it was decided that this simply would not work for our purposes, and thus the simpler file storage was chosen instead. With the freedom granted by this it was possible to allow the program to run for a variable amount of time, reducing excess data and wasted time with less user interaction.

Appendix

Circuit Schematics





SWD Connector
P3
Header 1.27mm 2x5 PTH
PTA3
PTA0
RESET
P3V3
GND
COIN CELL HOLDER
BT1
796136-1
D1
Diode MBR120VLSFT1G
3.3V Regulator
VR1
Vin
Vout
GND
C10
1µF
C11
GRM319R71H104KA01D
D2
P4
Header 2
3.3Vin Header
VBAT
GYROSCOPE
U2
L3GD20H
SCL/SPC
SDA/SDI/SDO
SDO/SA0
CS
DEN
CAP
VDD_IO
VDD
INT1
DRDY/INT2
RES
R3
4.7k
R2
4.7k
PTB2
PTB3
C12
C1206C103K2RACTU
U3
FXOS8700CQR1
SCL/SCLK
SDA/MOSI
SA0/MISO
SA1/CS_B
CRST
BYP
VDDIO
INT1
INT2
RST
RESERVED
N/C
R4
0
R5
0
PTD0
PTD1
C13
GRM319R71H104KA01D
C14
GRM319R71H104KA01D
R6
10k
IMU Design
Accelerometer and Magnetometer
Drawn by:
Sheet 1 of 2
Date: 4/11/2020
Time: 10:39:00 AM
Revision: Initial Submission
File: Sensors.SchDoc
JAMES COOK UNIVERSITY AUSTRALIA

Data Format of the Sensors taken is just Pitch, Roll and Yaw separated by backslashes in a CSV file once data transmission from Bluetooth to the Pi has occurred.