Magnetorquer

By

Jiayin Ling

Reda Lamniji

Riley Stewart

Final Report for ECE 445, Senior Design, Fall 2019

TA: Yangge Li

7 December, 2019

Project No. 34

Abstract

This report lays out the Fall 2019 progress in developing a stronger magnetic based Attitude Determination and Control System (ADCS) for the Laboratory for Advanced Space Systems at Illinois (LASSI). The approach explores an STM32L0 series microcontroller, digital potentiometer, magnetometer, air coil, h-bridge, and a variable voltage chip. We managed to have an extremely compact design that saves space and fulfills out goals.

Contents

1. Introduction **5**

1.1 LASSI Design 5

1.2 Hardware Challenges 5

1.3 Software Challenges 6

2 Design **7**

2.1 Coil Chassis Design 8

2.1.1 PCB Nook 8

2.1.2 Coil Trough and N Estimation 9

2.2 Circuit Design 10

2.2.1 Step-down converter 10

2.2.2 Current Source Module 10

2.2.2.1 Linear Voltage Regulator 11

2.2.2.2 Digital Potentiometer 11

2.2.3 H-Bridge 12

2.2.4 Magnetometer 13

2.2.5 Microcontroller 13

3. Design Verification **15**

3.1 Microcontroller and I2C 15

3.1.1 Read from and Write to MCU from external device 15

3.2 Magnetometer and Accelerometer 15

3.2.1 Setup and read values from magnetometer 15

3.2.1 Sense magnetorquer field 15

3.3 Magnetorquer Coil 15

3.3.1 Double previous designs output magnetic density 15

4. Costs **16**

4.1 Parts 16

4.2 Labor 16

5. Conclusion **16**

5.1 Accomplishments 16

5.2 Future work 16

References **17**

Appendix A Requirement and Verification Table **18**

1. Introduction

Among nanosatellites, magnetorquers are a common form of an Attitude Determination and Control System. In collaboration with the Laboratory for Advanced Space Systems at Illinois (LASSI), this project attempts to improve upon the organizations previous magnetorquer design. The device creates a magnetic moment with a current carrying coil of wire. When placed in an ambient magnetic field, such as the Earth's low orbit magnetic field, the device experiences a torque. This torque, in effect, rotates the magnetic moment into alignment with the Earth's magnetic field. If three magnetorquers are placed such that their magnetic moments are linearly independent, they provide a means for rotating the satellite into any orientation. In the design, LASSI desires a low mass subsystem which can be externally controlled via I^2C by the nanosatellite's main microcontroller. Finally, the coil needs to be capable of creating a magnetic moment of at least $0.54Am^2$. The essential components of this project are: a lightweight coil with high magnetic moment to current ratio, a driver circuit capable of amperage of at least 0.4mA, and an I^2C enabled communication/control module. This paper lays out our efforts and observations contributing to our conclusion: PCB embedded coil's are more compact and efficient than hand wound aircoils, the amperage required can and has been increased significantly, and I^2C is more than capable of controlling a magnetorquer.

1.1 LASSI Design

LASSI had a previous design from which we could reference to create a stronger model. The previous design featured a fully PCB embedded system. Both the coil and driver were designed in a single PCB.

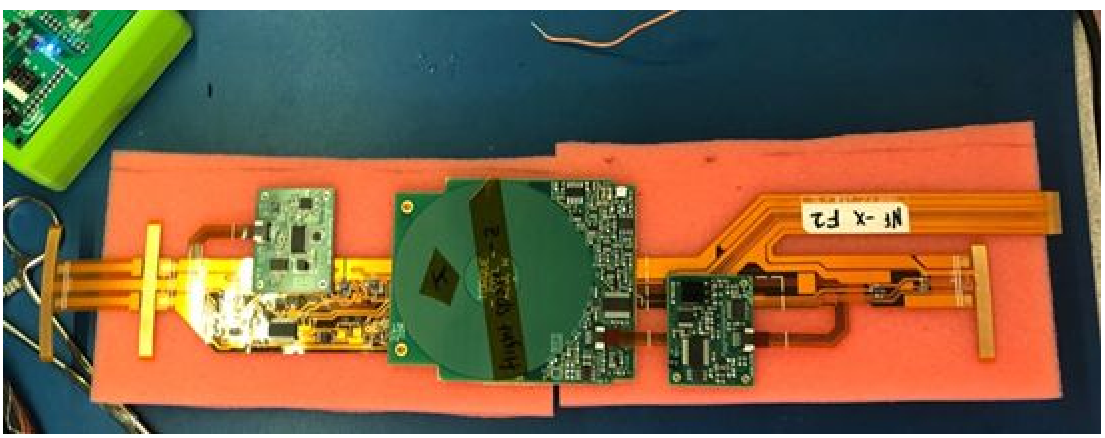


Figure 1.1

1.2 Hardware Challenges

A major part of the success of this project depended on the hardware employed. Considering equation 1.1, there appeared to be two main approaches to increasing the achievable magnetic moment. The first way lies in the product of the number of windings (N) and the area of the coil(A). This product (NA) will be referred to as the geometric coefficient(GC). The most obvious way to maximize the GC is to vastly increase N and A; This however is not desirable considering the restrictions on the size of our device(9cm x 9cm x 1cm). The challenge was to maximize the GC while staying within the specified volume.

$$= NIA$$

Equation 1.1

The second way to increase the magnetic moment is to increase the current provided to the coil. This required a total redesign of the previous circuit, and was also restricted by the provided power lines coming from the Cubesat power rail.

1.3 Software Challenges

In addition to increasing the magnetic moment, we also faced the challenge of controlling the system via external commands. We needed a microcontroller that could receive and interpret the incoming commands and execute the corresponding functions properly.

As most on-board devices utilizes the I2C interface for communication, we have a total of four devices that have to communicate with each other including the satellite control interface. We needed to familiarize ourselves with initializing and programming a microcontroller which is extremely complex in the C programming language given the limited amount of time we have. Considering that similar ARM microcontroller are used in arduino programming environment, we used a library for a board that uses a similar STM microcontroller to ours and programmed the devices in arduino IDE.

The challenge remains to find the correct library for more complicated devices like the magnetometer/accelerometer. Meanwhile, we have to scan in software for all devices we have on the I2C bus to make sure that electrical connections are valid and stable.

2 Design

Primarily, we looked at redesigning the coil itself as research showed a consensus that PCB embedded coil restricts the amount of magnetic moment achievable. The main goal was maximizing magnetic moment, and so it made more sense to look at more physical designs while bearing in mind that depth should be kept at a minimum. Magnetorquer rods, usually with a core of a permeable material such as Ferrite, produces the largest moment but are too volumetric and heavy for our design. We found that air coils, coils whose core is air or a vacuum in space, were the best for such a small design.

The design of the circuit/PCB for the system had to be very precise with the topology of the board as it was constrained to be 9cm x 9cm. The circuit consists of a microcontroller, a digital potentiometer, an H-bridge, a magnetometer/accelerometer, a linear voltage regulator, and a buck converter. The system uses both on and off board communication via I2C. The simplified block diagram can be seen below in Figure 2.1.

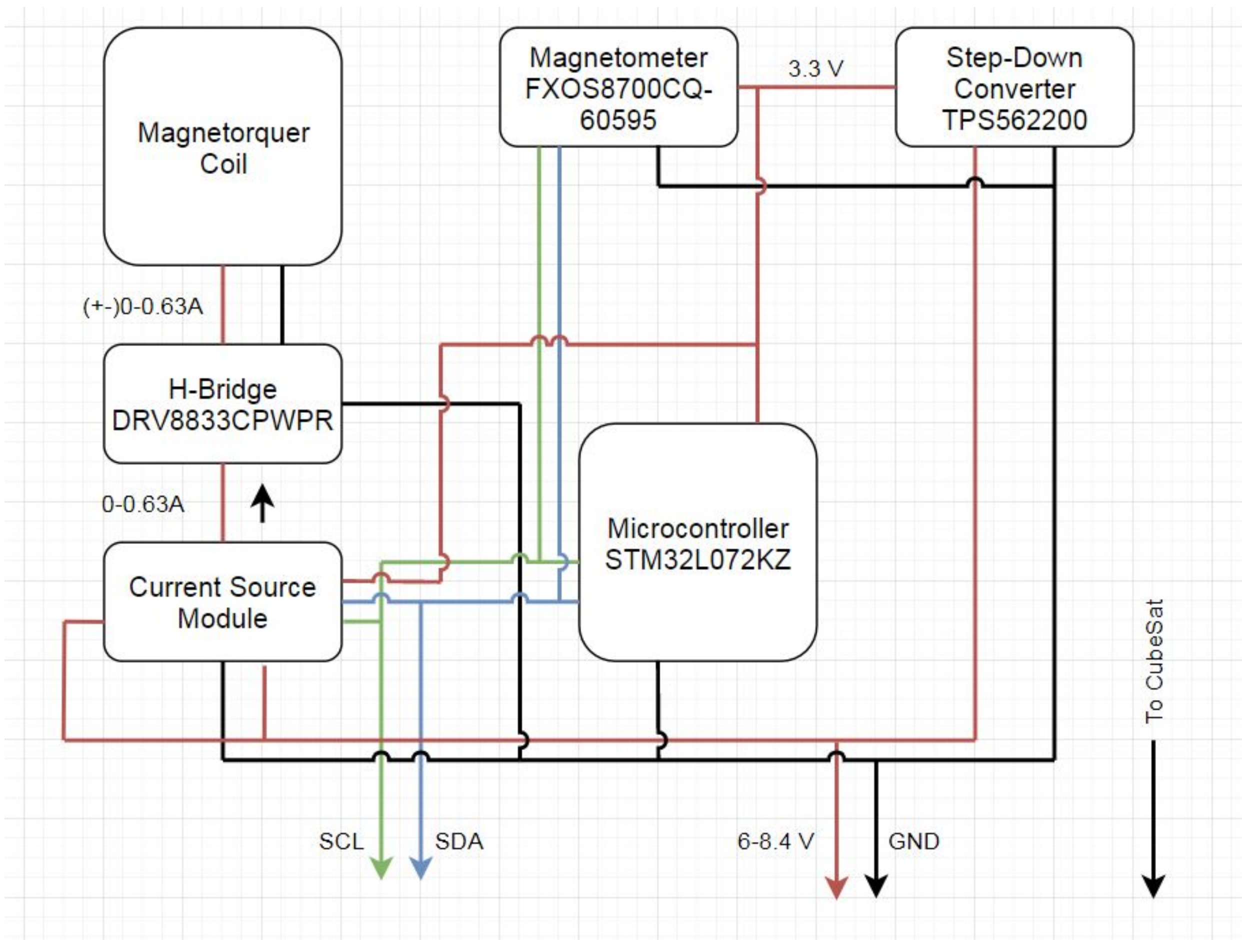


Figure 2.1

2.1 Coil Chassis Design

Bearing in mind that the coil and circuit must share the alloted volume, the coil must be based around a minimal PBC design. The size of the final PCB can be seen in Figure 2.7. Using SolidWorks we created a square spool to contain the wire windings(see Figure 2.2). The dimensions are 9cm x 9cm x 1cm and can be seen in figures 2.3, 2.4 and 2.5. Separating the two main features of the chassis (PCB Nook and Coil Trough)is a wall 2mm in depth with quarter circle corners of 2mm radial curve to reduce pinching of the wire.

2.1.1 PCB Nook

The PCB has a rectangular shape with side dimensions of 35.94mm x 24.51mm. In the design of the Coil Chassis, an extra 0.5 mm leeway was added in each direction to ensure the PCB would fit. This was a success. Additionally the nook features four posts for the pcb to sit on. Ideally these posts would have holes to secure the board via its drill holes, however the 3D printer used had limiting precision disallowing this addition. Finally, the nook contains a throughhole on the side wall for the coil wire to be routed for easy soldering to the board.

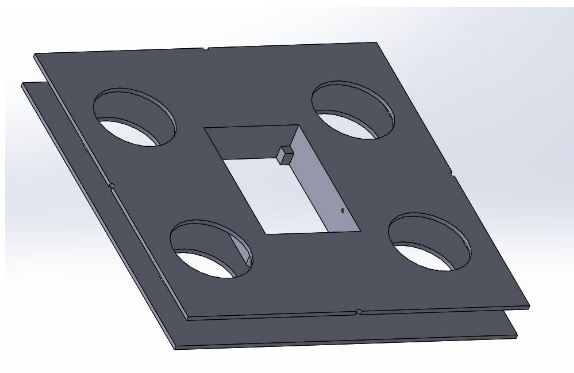


Figure 2.2

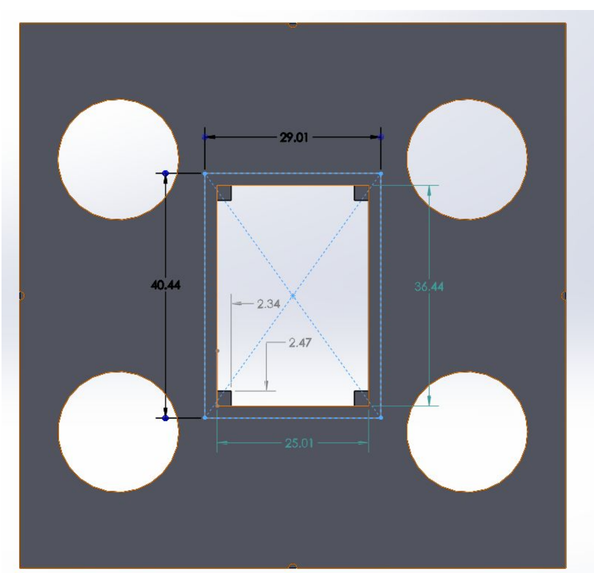


Figure 2.3

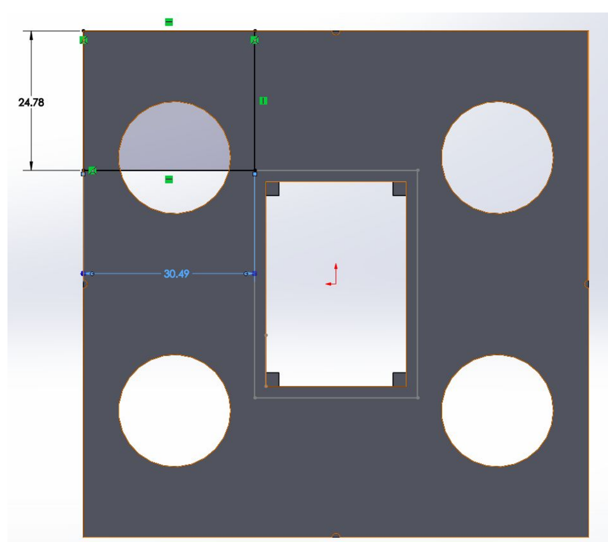
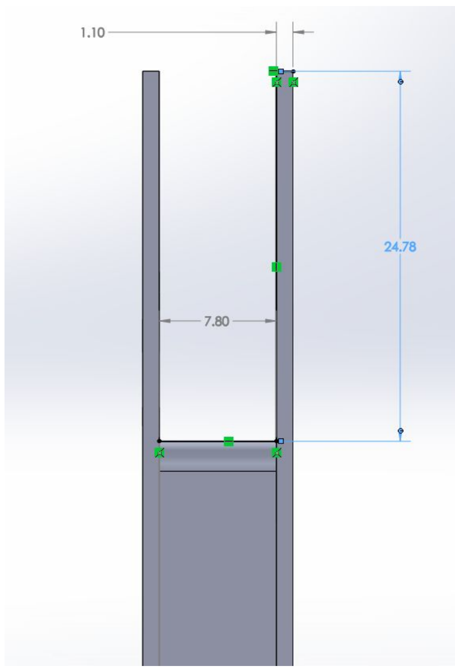


Figure 2.4

2.1.2 Coil Trough and N Estimation

The coil trough is the area of the chassis in which the coil is wound. An excellent way to estimate the amount of windings one can achieve, is to use a calculator which finds the amount of circles which can fit in a given rectangle. The rectangle of our chassis is seen in Figure 2.5. This rectangle combined with the gauge of wire chosen gives us our geometric coefficient. The wire chosen was a 20 gauge copper magnetic wire. Reasons for this are explained in Section 2.2.2.



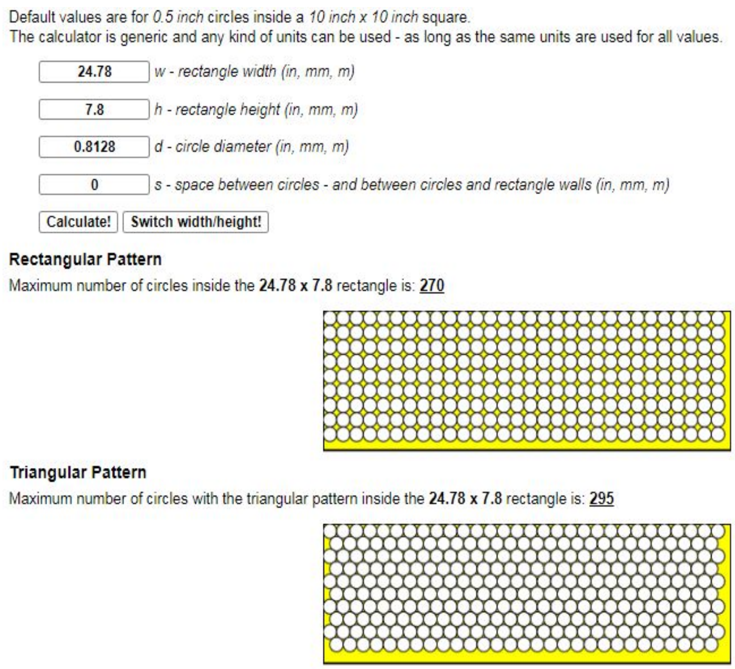


Figure 2.5

Figure 2.6

With dimensions 7.8mmx 24.78mm and a wire diameter of 0.8128mm. The expected number of windings (N) is estimated to be 270 to 295 as seen in Figure 2.6.

The average area was roughly calculated with the integrals shown in Equation 2.1 and 2.2. The answer is displayed in Equation 2.3. Therefore the estimated geometric coefficient is in the range 0.999 to 1.0915.

$$\int_{20.22}^{45} \pi x^2 dx = 86768.78407$$

Equation 2.1

$$\int_{20.22}^{A} \pi x^2 dx = \frac{86768.78407}{2}$$

Equation 2.2

$$A = 0.0037m^2$$

Equation 2.3

2.2 Circuit Design

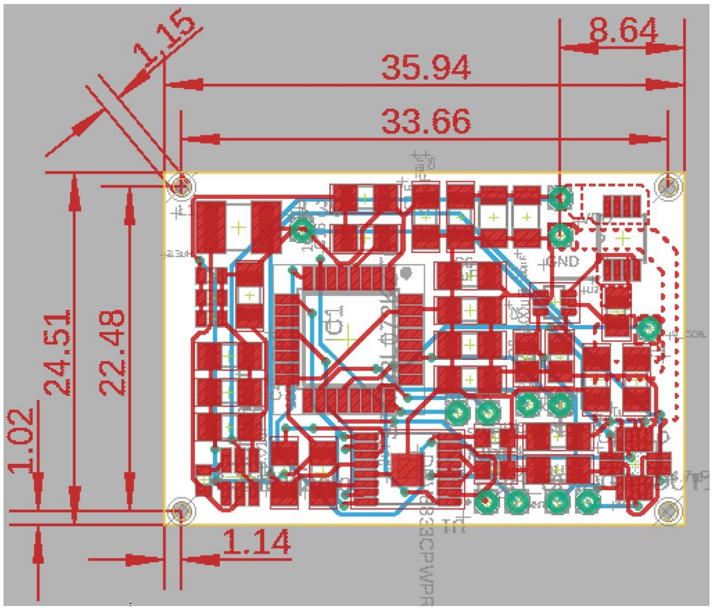


Figure 2.7

2.2.1 Step-down converter

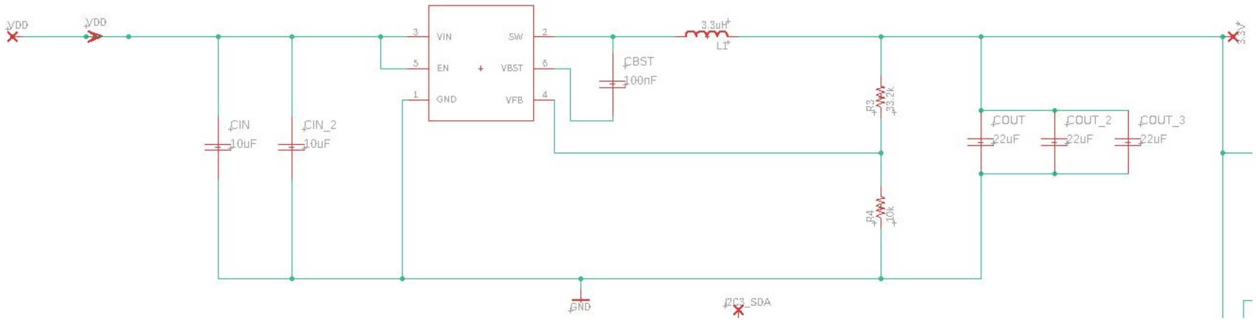


Figure 2.8

The chosen chip was the TPS562200 chip to achieve step-down of unregulated 6.0-8.4V input to a stable 3.3V DC output. We selected this specific chip because it had great, easy to read documentation and had great support from TI in software: TI Power Designer.

2.2.2 Current Source Module

This design employs a topology that is shown in the datasheet of the linear regulator and swaps the fixed resistor connected to the set pin with a digital potentiometer controlled via I2C interface so that we can vary the current as it is a linear function of the potentiometer value. This design is modelled as a current source with capabilities up to a 1.1A output. This output limit helped us narrow down our choice in magnetic wire to a 20 gauge wire with a maximum current of 1.6A.

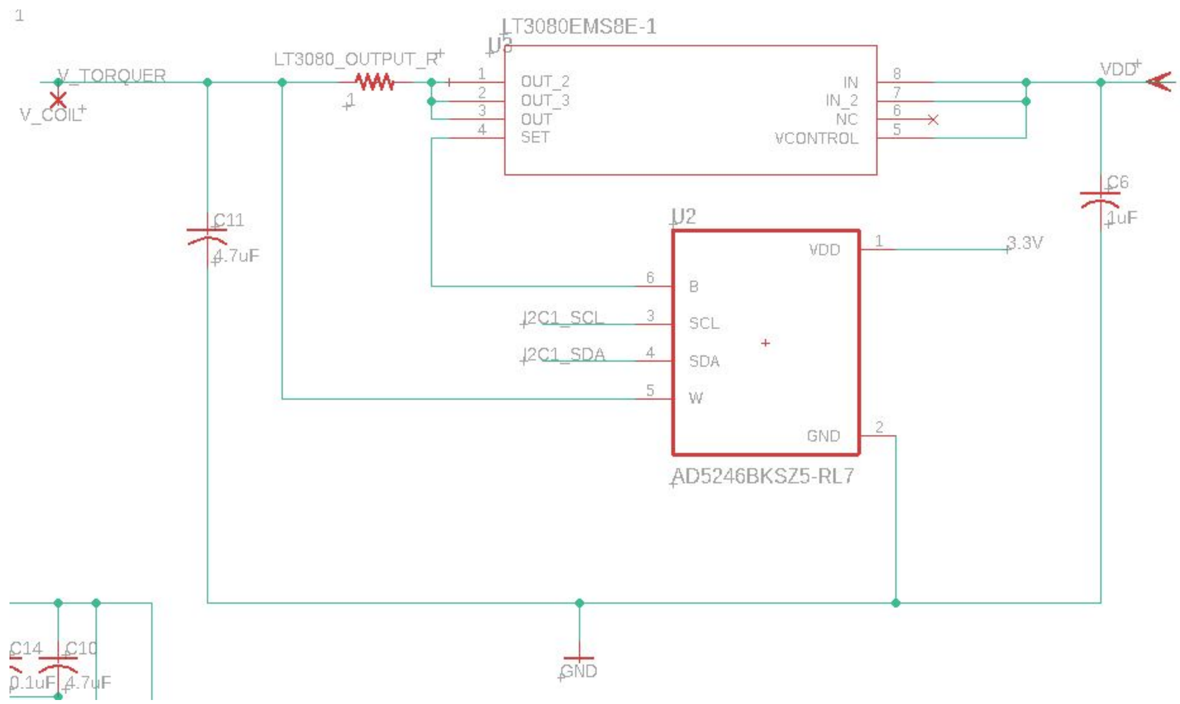


Figure 2.9

2.2.2.1 Linear Voltage Regulator

Provides V_Torquer which supplies the coil with 3.3V steady DC output with current based on the potentiometer. The chip selected for this functionality was the LT3080EMS8E. This chip is current-limited to 1.1A which the circuit is designed to handle. The chip is also thermo-limited to .003%W which was an issue in application. The datasheet provides ample schematics and setups to lay multiple of these chips in parallel as to reduce the power consumption of each chip to achieve 1.1A without any chip hitting its thermo-limit. An example of such a configuration is shown in figure 2.10.

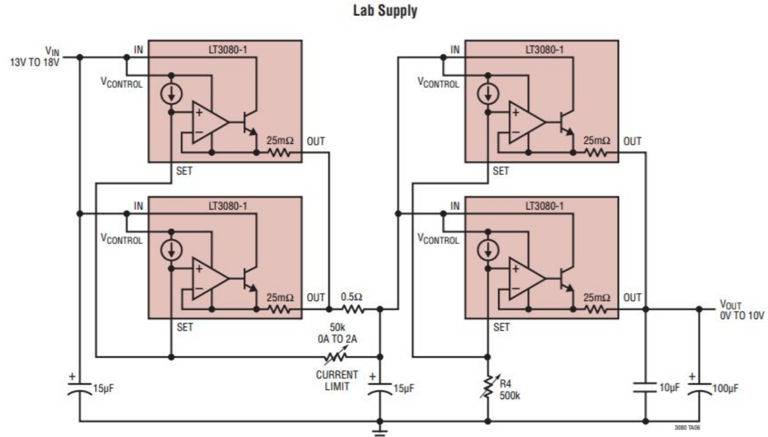


Figure 2.10

2.2.2.2 Digital Potentiometer

The choice of digital potentiometer was made based on its maximum setting of resistance. A 100kΩ digital potentiometer was chosen, the AD5246BKSZ5-RL7. It provides up to 100kΩ via I2C with 128 possible settings. This chip is used to limit current to the coil to control field strength i.e: magnetic moment **μ** = N I A. This chip is sufficiently powered by 3.3V DC.

2.2.3 H-Bridge

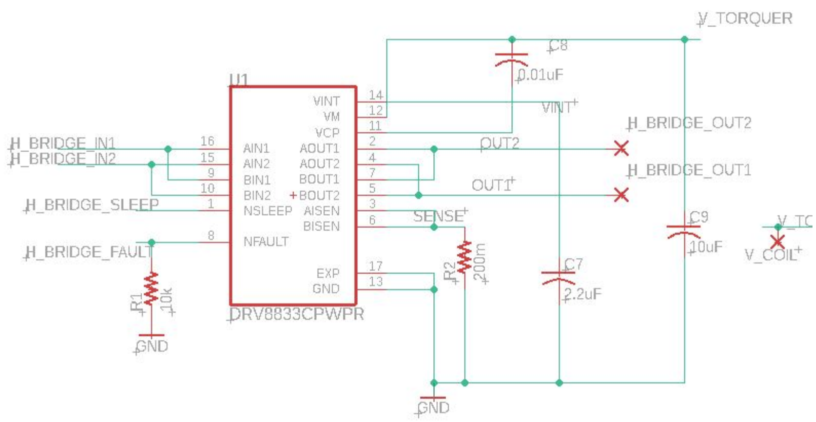


Figure 2.11

The H-Bridge of the system is the DRV8833CPWPR which outputs 1.5A RMS and 2A peak-to-peak. It can be run in forward or reverse as set by the microcontroller driving so that the system can ideally provide -1.1A to 1.1A to the coil. The input pin configuration for the H-Bridge is shown below in table 2.1.

xIN1	xIN2	xOUT1	xOUT2	FUNCTION
0	0	Z	Z	Coast/fast decay
0	1	L	H	Reverse
1	0	H	L	Forward
1	1	L	L	Brake/slow decay

Table 2.1

2.2.4 Magnetometer

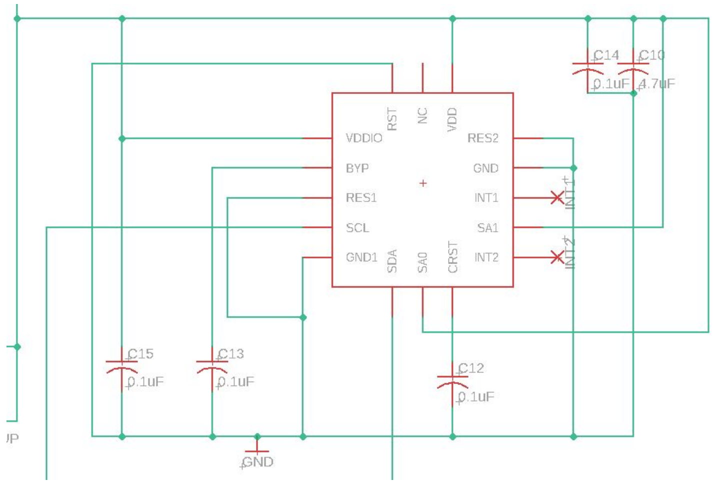


Figure 2.12

The magnetometer for this system is the FXOS8700CQ which is an Adafruit chip that easily configures and works with arduino which is helpful in debugging the system. It reports magnetic and acceleration to the microcontroller as well as informing the rest of the satellite of the system's status through the microcontroller. The chip must be calibrated which can most efficiently be done by using a helmholtz cage and calibrated magnetometer to achieve linearity in its measurement.

2.2.5 Microcontroller

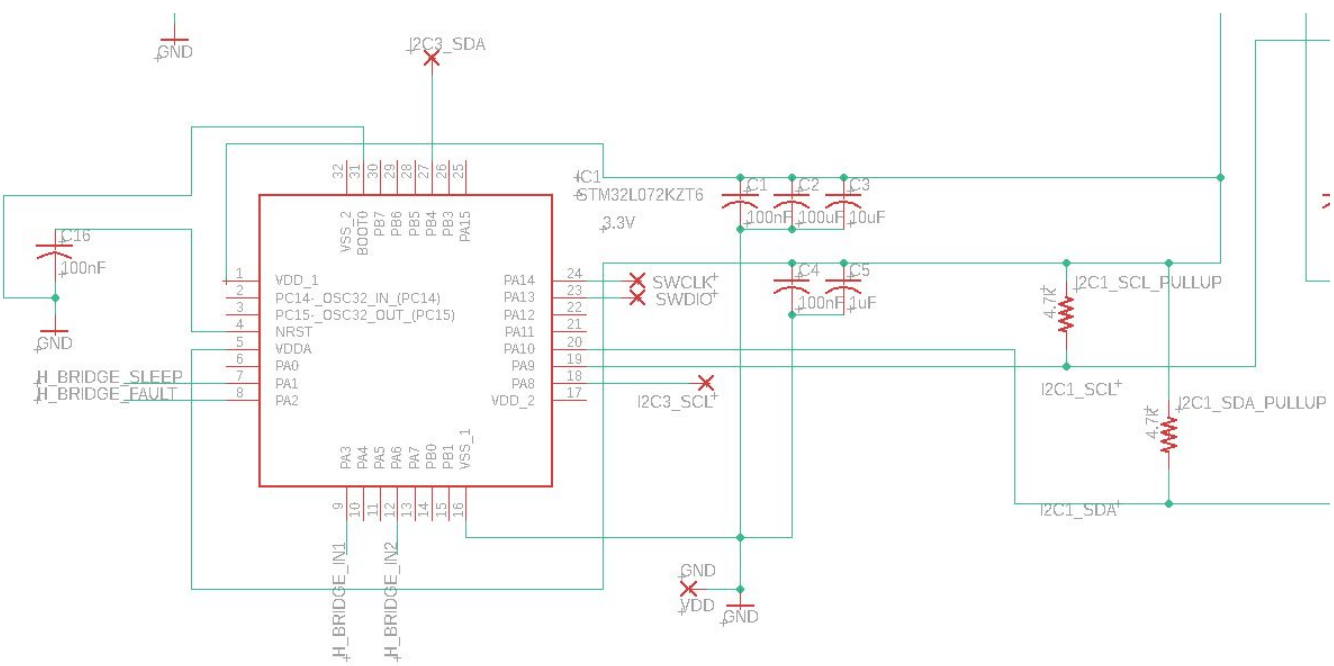


Figure 2.13

This chip, the STM32L072KZT6, is the brain of the system and uses I2C to communicate with the necessary devices to control the current outputted to the coil as well as read the status and magnetic field strength.

Arduino was used for programming the microcontroller for ease of debugging via serial output terminal through the arduino COM port. We used external libraries: stm32duino, stm32-ArduinoTools, and fxos8700cq-arduino for easy programming of our microcontroller and magnetometer. In order to circumvent having an arduino supported breakout board for the chosen microcontroller, we were able to select a breakout board similar to our PCB layout and a chip series similar to the system's (also low-power). Snippets of the code are shown below.

```
void setup()
{
    pinMode(PA1, OUTPUT);
    pinMode(PA3, OUTPUT);
    pinMode(PA6, OUTPUT);
    digitalWrite(PA1, HIGH);
    digitalWrite(PA6, LOW);
    digitalWrite(PA3, HIGH);
    Wire.setSDA(PA10);
    Wire.setSCL(PA9);
    Wire.begin();
    Wire.beginTransmission(4); // transmit to device #4
    Wire.write("Started");        // sends five bytes
    Wire.write(1);              // sends one byte
    Wire.endTransmission();    // stop transmitting
  /* Initialise the sensor */

  sensor.init();
  Wire.beginTransmission(46); // transmit to device #46 potentiometer
                              // device address is specified in datasheet
  //Wire.write(byte(0x1A));            // sends instruction byte
  Wire.write(30);             // sends potentiometer value byte

  Wire.endTransmission();     // stop transmitting
}
   byte val = 0;
   byte x = 2;

void loop()
{


  Wire.endTransmission();     // stop transmitting
 */
  sensor.readAccelData();
  sensor.readMagData();
  Wire.beginTransmission(4); // transmit to device #4

  Wire.write(sensor.magData.x);        // sends five bytes

  Wire.endTransmission();    // stop transmitting
  x+=1;
  delay(1000);

}
```

Figure 2.14

```
#include <Wire.h>

void setup()
{

  Wire.begin(4);                // join i2c bus with address #4
  Wire.onReceive(receiveEvent); // register event
  Serial.begin(9600);           // start serial for output

}

void loop()
{
  delay(100);
//  Serial.println("sb_loop");
}

// function that executes whenever data is received from master
// this function is registered as an event, see setup()
void receiveEvent(int howMany)
{
  while(1 < Wire.available()) // loop through all but the last
  {
    char c = Wire.read(); // receive byte as a character
    Serial.print(c);         // print the character
  }
  int x = Wire.read();    // receive byte as an integer
  Serial.println(x);         // print the integer
}
```

Figure 2.15

A few limitations of the selected chip should be noted. The I2C3 channel was not functional on any of 5 tested chips. The SDA and SLA lines were probed with an oscilloscope and noise was seen to be the output. Upon online investigation through the manufacturer's forum, it should be noted that this was an unresolved problem faced by a few other users as well. The chip is also not well supported and so in future work it should be in the system's best interest to use a different STM32L0 (low-power) series chip.

3. Design Verification

This section outlines the various verifications needed to confirm our success.

3.1 Microcontroller and I2C

3.1.1 Read from and Write to MCU from external device

After coding the devices correctly we confirmed that the MCU can act as a master on the I2C bus to print anything we want onto the console which is connected to the arduino board we used to simulate a satellite. Similarly, it can request data from the arduino. Simply switching the code of the two confirms the case where the satellite acts as a master works.

3.2 Magnetometer and Accelerometer

3.2.1 Setup and read values from magnetometer

When the arduino IDE setup takes one line of code in the arduino IDE and reading values takes another line. We the data is passed to the arduino we use to simulate the satellite to be printed to console.

3.2.1 Sense magnetorquer field

Using the arduino IDE library for our chip we can simply print the raw reading from the IC onto console via COM port. When the reading changes we know a field is generated by the coil. To know the precise reading, we need to calibrate the IC in a helmholtz cage.

3.3 Magnetorquer Coil

A major requirement was to increase the magnetic field density outputted by the coil two-fold.

3.3.1 Double previous designs output magnetic density

The previous design gave an output of 0.7 to 0.8 Gauss. The primary way to test this is to use an industrial grade magnetometer. The LASSI team has a high grade magnetometer that was used to give a reading for our project. The magnetometer maxed out its reading at 1.4 Gauss.

4. Costs

4.1 Parts

Table 4.1 Parts Costs

Part	Manufacturer	Retail Cost ($)	Bulk Purchase Cost ($)	Actual Cost ($)
TPS562200DDCR	Texas Instruments	1.05	0.44	1.05
LT3080EMS8E-1	Analog Devices	4.85	2.66	4.85
AD5246BKSZ100-R L7	Analog Devices	1.56	0.64	1.56
DRV8833CPWPR	Texas Instruments	1.79	0.77	1.79
FXOS8700CQ	NXP Semiconductors	4.72	1.83	4.72
STM32L072KZT6	STMicroelectronics	4.04	2.13	4.04
Fixed Electrical Components	Various	4.8	N/A	4.8
Total		**22.8**	**N/A**	**22.8**

Table 4.1

4.2 Labor

$$3 * \$40/hr * 10hr/week * 12\ weeks * 2.5 = \$36{,}000$$

Equation 4.1

5. Conclusion

5.1 Accomplishments

The system's success was outlined by the CubeSat team's own test where they measured using their magnetometer to find that our field strength was 1.4 Gauss, the limit of the device and much higher than the previous design (.7-.8 Gauss).

In the end our coil contained roughly 210 windings with an average area of roughly $0.0037m^2$. With an output current ranging from 0.43A to 0.63A this estimates our magnetic moment to be $0.334Am^2$ to $0.490Am^2$. Although this does not meet the requirements, the coil was hand wound, which introduces a lot of inefficiencies. Assuming the coil was machine wound and met the minimum estimated coil windings of 270, the magnetic moment would be in the range of $0.42957Am^2$ to $0.629Am^2$.

5.2 Future work

Future additions and improvements to the magnetorquer range widely. Most notably, a future project could be redesigning the coil using our current circuit design to minimize weight and trade between maximum achievable moment and weight to produce the most efficient design for the CubeSat ADCS. A suggestion on this is to use machine wound coils and maybe even to consider returning to a PCB-embedded design with much thicker traces and more layers perhaps with dedicated current to different layers. As aforementioned, it would be wise to switch microcontroller to a more supported chip with multiple I2C channels.

To successfully achieve the current that the system is capable of producing, multiple linear voltage regulators should be placed in parallel as discussed in section 2.2.2.1. This reduction of heat on a single chip will increase the maximum current outputted substantially. If this caused a current output of 0.8A (only 72 percent of what the chip is rated at) the magnetic moment of our system could potentially be. $0.799Am^2$.

Finally, the PCB can be reshaped into a circle for a more efficient coil chassis shape. This is of course irrelevant if a PCB based wire coil is chosen as an alternative to our air coil design.

References

[1] “Wire Gauge and Current Limits Including Skin Depth and Strength.” *American Wire Gauge Chart and AWG Electrical Current Load Limits Table with Ampacities, Wire Sizes, Skin Depth Frequencies and Wire Breaking Strength*, PowerStream Technology, www.powerstream.com/Wire_Size.htm.

[2] Engineering ToolBox, (2014). *Circles within a Rectangle*. [online] Available at: https://www.engineeringtoolbox.com/circles-within-rectangle-d_1905.html [Accessed Day Mo. Year].

[3] “WebBench Power Designer.” *Power Designer*, Texas Instruments, webench.ti.com/power-designer/switching-regulator.

[4] Miller, Duncan. “Design Optimization of the CADRE Magnetorquers.” *Aerospades*, 2 May 2013, www.aerospades.com/uploads/3/7/3/2/37325123/cadre_torquers.pdf.

Appendix A Requirement and Verification Table

Table A.1 System Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Microcontroller and I2C a. Read from and Write to MCU from external device	1. Verification a. Using arduino, send and receive information	Y
2. Magnetometer and Accelerometer a. Setup and read values from magnetometer b. Sense magnetorquer field	2. Verification a. Read and print magnetometer values to terminal b. Calibrate magnetometer in helmholtz cage, then use calibration model to determine influences by magnetometer	Y N
3. Magnetorquer Coil a. Double previous designs output magnetic density of 0.7-0.8 Gauss	3. Verification a. Measure the magnetic field density output of the magnetorquer with the magnetometer provided by LASSI	Y