LIDAR CAVE MAPPER

FAMU-FSU College of Engineering
ME & ECE Senior Design

Our Team

Project Manager

Lead ECE

Lead ME

Financial Advisor

Webmaster

Power Engineer

- Alisha Hunt
- James Oliveros
- Spencer Day
- Cesar Rivas
- Hunter Hayden
- Jake Ogburn

Hunter Hayden

The Problem

Cave mapping can be a very expensive undertaking. We need to create a portable cave mapper device for freelance cavers. Current LiDAR devices are expensive and cave mapping is often done using notepads and measuring tape. This project is intended to make LiDAR devices more accessible to cavers by both making it affordable and simple DIY style with open source code.

Design Overview

Hunter Hayden

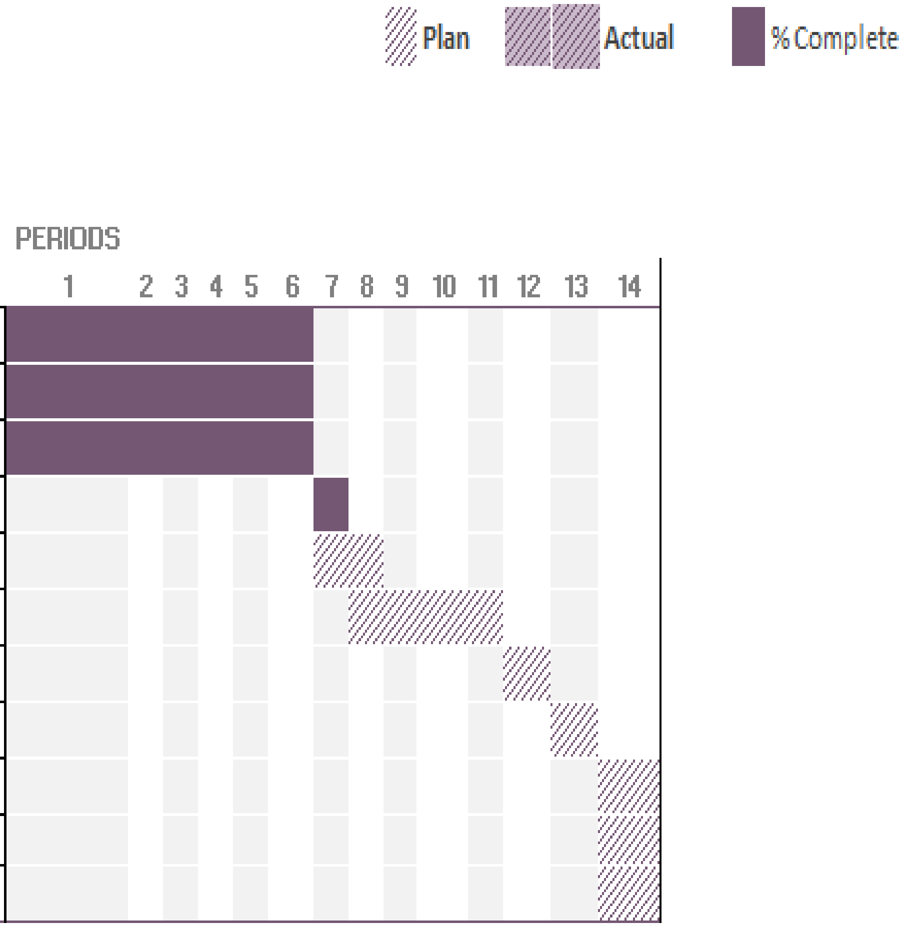
- Must be able to map fully around circle and at least 270 degrees in phi direction.
- Needs to fit in sponsor-provided casing ~12x6x8
- Map at least every .5m from 40m away
- May use alternative mapping styles to save memory
- Convert data into importable 3D image
- Weighs no more than 5lbs
- User friendly (easy to use in dark caves)

Schedule

Hunter Hayden

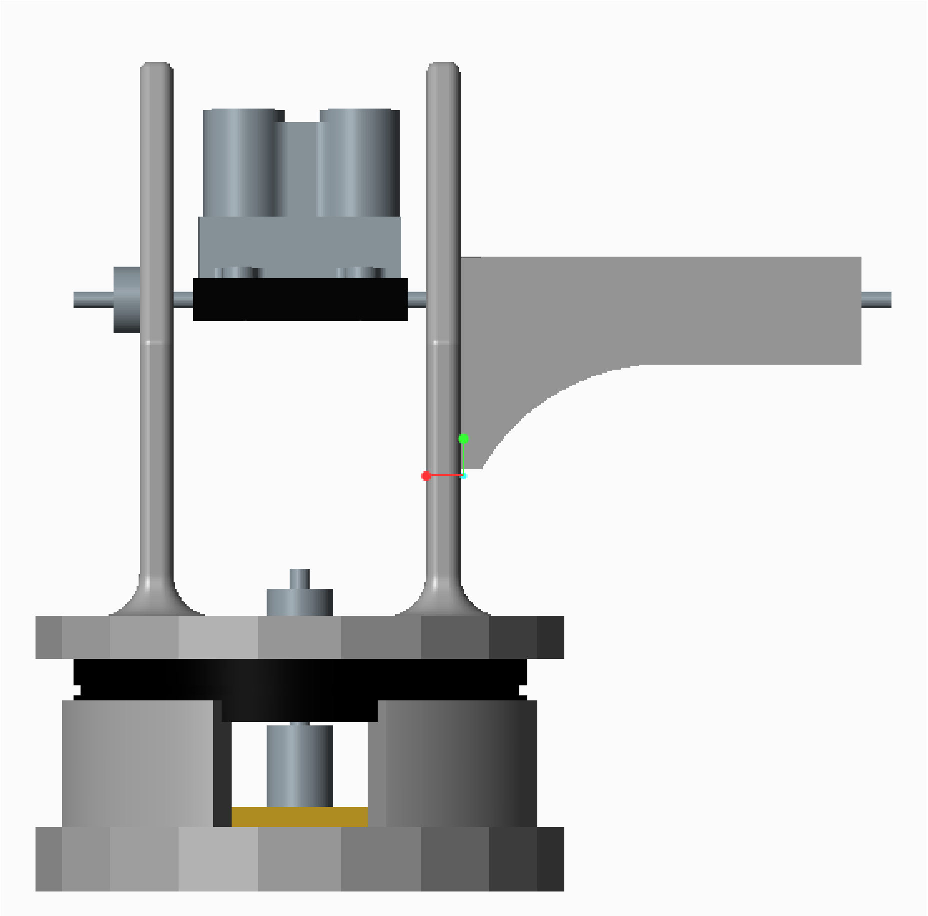
Project Planner

Activity	DUE DATE	PLAN START	PLAN DURATION (Weeks)	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Review Finalized Conceptual Design		1	6	1	6	100%
Begin Testing/Calibration of Electrical Components		1	6	1	6	100%
3D Print parts		1	6	1	6	100%
Midterm Presentation 1		7	1	7	1	100%
Update Website		7	2			
Run Full Scale Tests/Compile Data		8	4			
Midterm 2 Presentation		12	1			
Operational Manual Due	7-Apr	13	1			
Full Scale Test in Marianna Cave System		14	2			
Senior Design Fair	13-Apr	14	1			
Final Project Report and Website Due	21-Apr	14	1			



Mechanical Design

Hunter Hayden



CAD Model of final design

- Final drawings have been completed for the LIDAR base and motor mounts
- Rounded corners at the base make the supports sturdier
- Extended motor mount has been resized in order to comfortable seat the second motor

Material and Manufacturing Processes

Spencer Day

- Material: PLA thermoplastic
  - Durable, printable, very inexpensive
  - Derived from composted vegetables rather than petroleum, more environmentally conscious
  - Can be “welded” due to its low melting temperature
- Manufacturing
  - 3D printing
  - Fastening: Welding and non metallic nuts and bolts

Issues Encountered

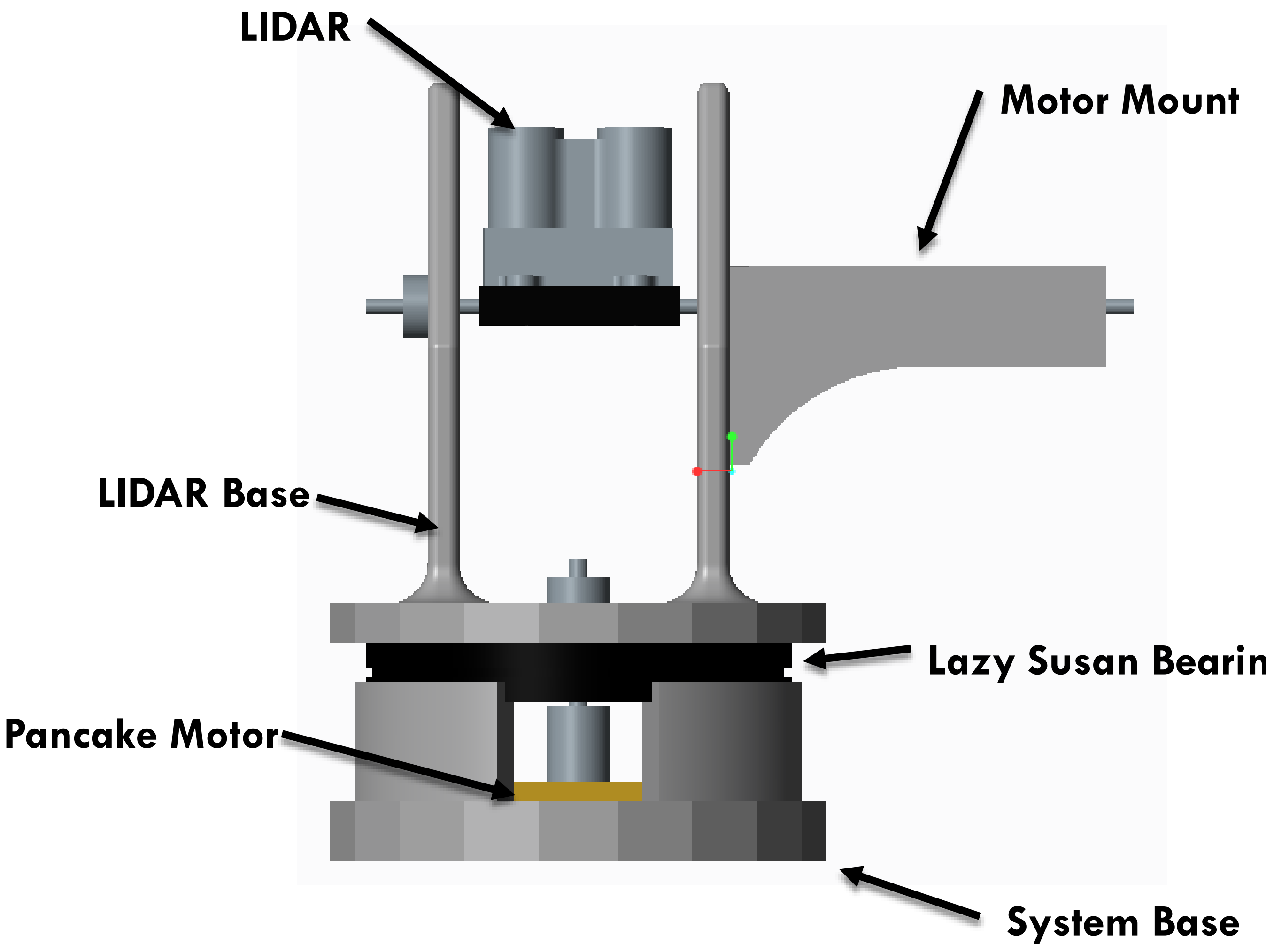
Spencer Day

- LIDAR
  - LIDAR lenses are fragile and have made it difficult to transport
  - A lens cover is being designed to keep the lenses safe
- Pancake Stepper Motor
  - Issues with overheating
  - Motor started to get hot after only 2 hours of operation
  - Possible solutions:
    - Adding spacers to keep the motor air cooled
    - Dropping current and running the test again until proper temperature is achieved

Construction

Spencer Day

- Construction has been optimized for low weight and easy assembly
- Introduction of raised system base and lazy susan bearing ensure no issues with friction



Storage and Setup

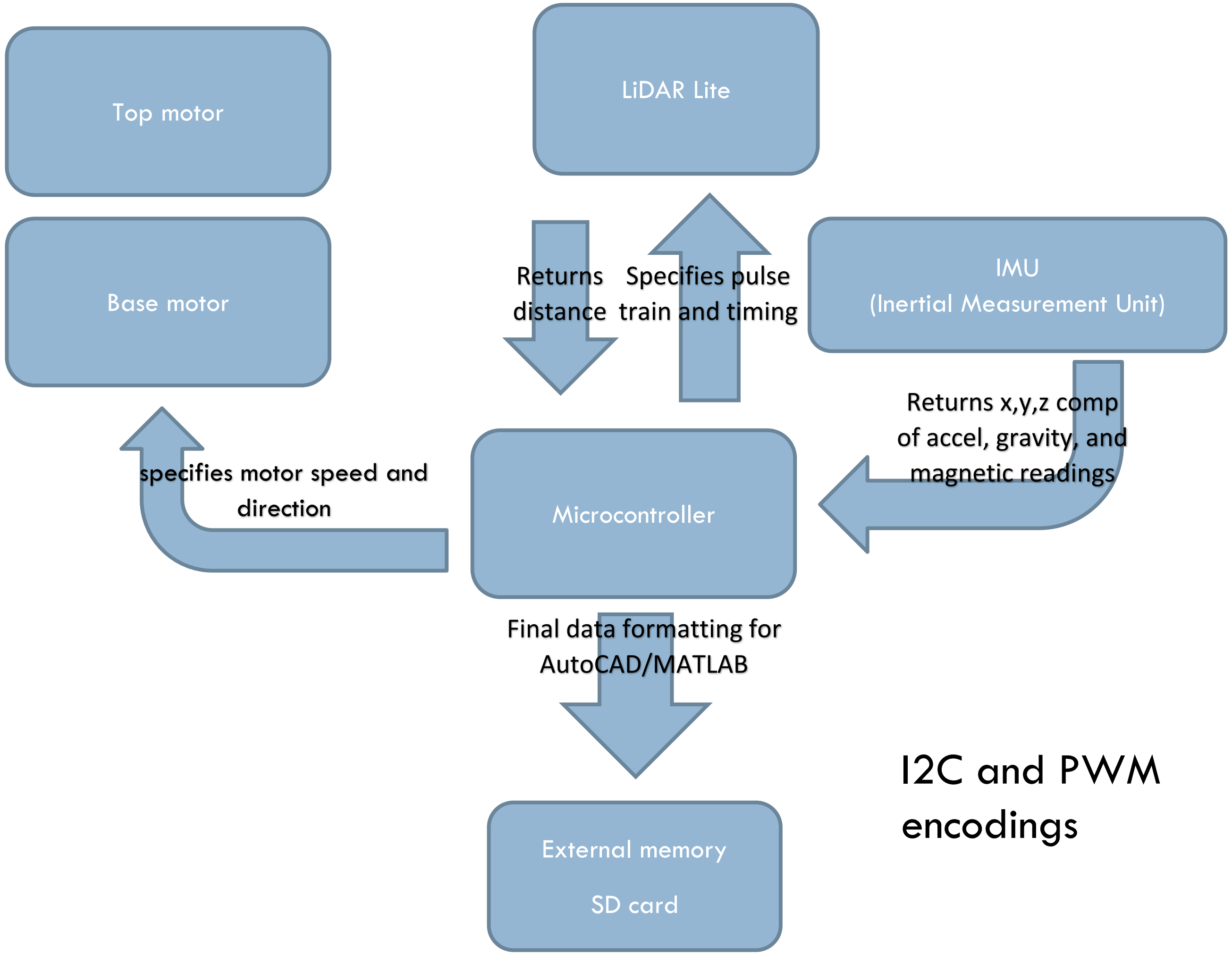
Spencer Day

- Design requirement: All of the mechanical and electrical components of the LiDAR system must fit into a 12”x6”x8” hard plastic case.
- Until this point we see no issue of fitting everything.
- When the final wiring is complete we will mount components as to ensure no complications with setup and breakdown

Data Flow Hierarchy

Alisha Hunt

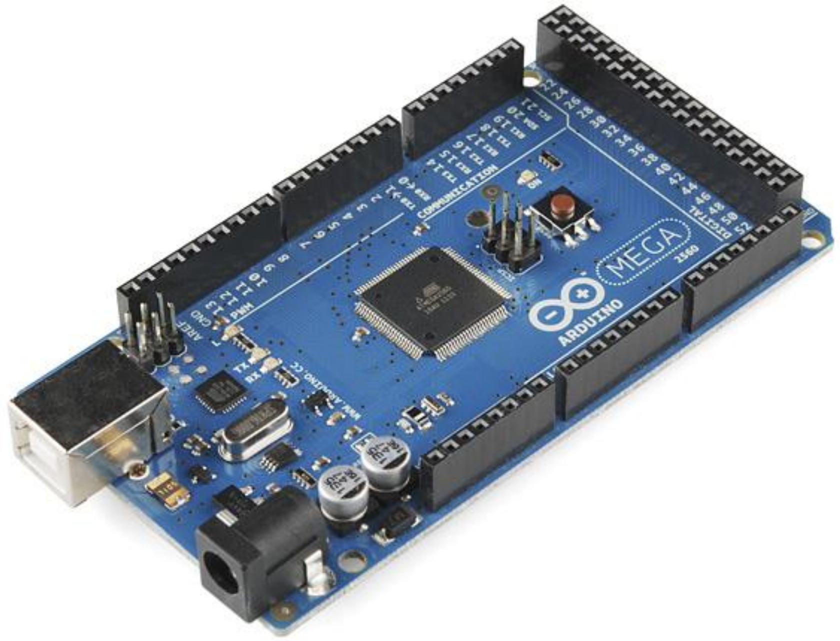
- controller dictates motor speed and direction
- Controller specifies pulse train to LiDAR and LiDAR returns distance
- Inertial Measurement Unit sends position data to Uno
- Processor sends final data to external memory chip



I2C and PWM encodings

Alisha Hunt

Coding Together

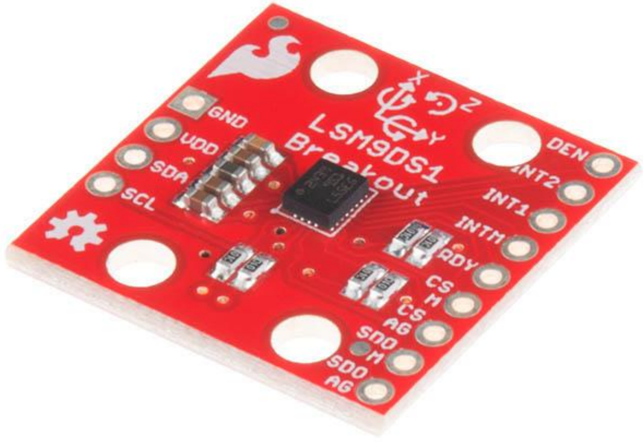


- Individual components coded by separate people and wired on different Arduinos
- Conflict with LiDAR and IMU pin assignments, Arduino Uno has only one set of I2C pins (clock SCL and data line SDA)
- IMU requires SCL and SDA pin for either SPA or I2C encodings
- Ideally each section is a module which is removeable.
- Requires mutual understanding of other's work to combine into running program

Alisha Hunt

Inertial Measurement Unit

- Has 9 degrees of freedom: x, y, & z for angular rotation, acceleration and magnetic force
- Gives geolocation from magnetic fields
- Will track movement of motors
- Will help cancel out potential ringing of motors

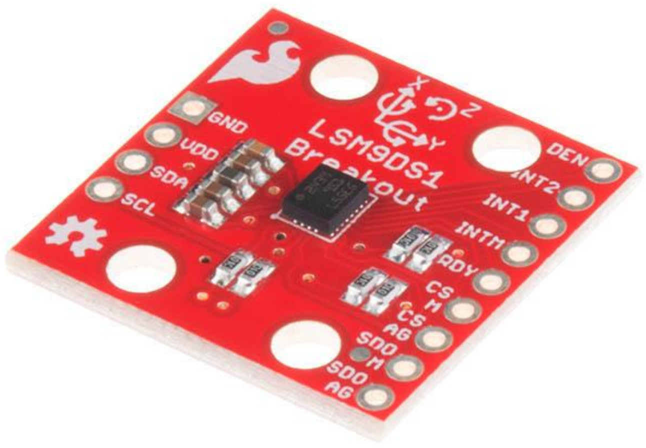


Alisha Hunt

accelerometers

How do accelerometers sense motion?

- Piezoelectric effect: microscopic crystals are stressed by movement. The capacitance between crystals changes thus generating voltage.
- Basic tilt is measured from differences in the gravitational field



Alisha Hunt

Using the LSM9DS1 Library

SparkFun LSM9DS1 IMU by **SparkFun Electronics /www.sparkfun.com>** Version **1.1.0** **INSTALLED**
A driver library for the LSM9DS1 IMU. Communicates with the LSM9DS1 over either SPI or I2C, so you can painlessly integrate an accelerometer, magnetometer, and gyroscope into your project.
More info

- Required a manual install. Not compatible with LSM9DSM1 library within Arduino Library manager
- Runs basic functions of IMU from 4 main pins (GND, VDD, SDA, SCL)
- Mathematical assumptions: uses aerospace configuration of matrices (x, y, z) and limits the range of pitch to 90 and roll to 180 degrees
- Uses x, y, and z components of acceleration, gravity, and magnetism to calculate roll, pitch, and heading

Alisha Hunt

IMU Initialization

```
// The SFE_LSM9DS1 library requires both Wire and SPI be
// included BEFORE including the 9DS1 library.
#include <Wire.h>
#include <SPI.h>
#include <SparkFunLSM9DS1.h>

LSM9DS1 imu;
#define LSM9DS1_M 0x1E // Would be 0x1C if SDO_M is LOW
#define LSM9DS1_AG  0x6B // Would be 0x6A if SDO_AG is LOW

#define PRINT_CALCULATED
//#define PRINT_RAW
#define PRINT_SPEED 250 // 250 ms between prints
static unsigned long lastPrint = 0; // Keep track of print time

// Earth's magnetic field varies by location. Add or subtract
// a declination to get a more accurate heading. Calculate
// your's here:
// http://www.ngdc.noaa.gov/geomag-web/#declination
#define DECLINATION -8.58 // Declination (degrees) in Boulder, CO.
```

```
void setup()
{

  Serial.begin(9600);

  // LiDAR pin declarations
  pinMode(2, OUTPUT); // Set pin 2 as trigger pin
  digitalWrite(2, LOW); // Set trigger LOW for continuous read

  pinMode(3, INPUT); // Set pin 3 as monitor pin

  imu.settings.device.commInterface = IMU_MODE_I2C;
  imu.settings.device.mAddress = LSM9DS1_M;
  imu.settings.device.agAddress = LSM9DS1_AG;
  if (!imu.begin())
  {
    Serial.println("Failed to communicate with LSM9DS1.");
    Serial.println("Double-check wiring.");
    Serial.println("Default settings in this sketch will " \
                   "work for an out of the box LSM9DS1 " \
                   "Breakout, but may need to be modified " \
                   "if the board jumpers are.");
    while (1)
```

Alisha Hunt

Data output

- Heading and pitch readings directly correlate to theta and phi coordinates
- Will be using magnetic readings to initially align device
- Alternatively can use heading (calculated using magnetic readings)
- Currently just a text feed, but will be converted to an exportable file

```
void printAttitude(float ax, float ay, float az, float mx, float my, float mz)
{
  float roll = atan2(ay, az);
  float pitch = atan2(-ax, sqrt(ay * ay + az * az));

  float heading;
  if (my == 0)
    heading = (mx < 0) ? PI : 0;
  else
    heading = atan2(mx, my);

  heading -= DECLINATION * PI / 180;

  if (heading > PI) heading -= (2 * PI);
  else if (heading < -PI) heading += (2 * PI);
  else if (heading < 0) heading += 2 * PI;

  // Convert everything from radians to degrees:
  heading *= 180.0 / PI;
  pitch *= 180.0 / PI;
  roll  *= 180.0 / PI;

  Serial.print("Pitch, Roll: ");
  Serial.print(pitch, 2);
  Serial.print(", ");
  Serial.println(roll, 2);
  Serial.print("Heading: "); Serial.println(heading, 2);
}
```

LIDAR v3

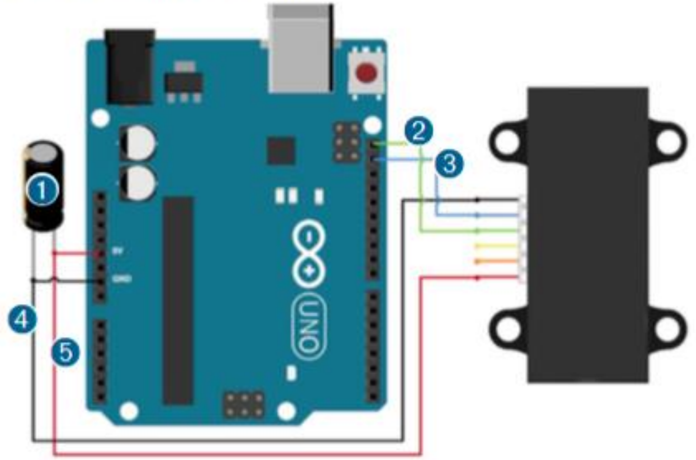
James Oliveros

- 0 – 40m Laser Emitter
- 2.5cm accuracy for distances 1m+
- 4.75-5V DC power consumption
- Max 130 mA current comsumption
- I2C or PWM Interface

LIDAR: I2C vs PWM Wiring

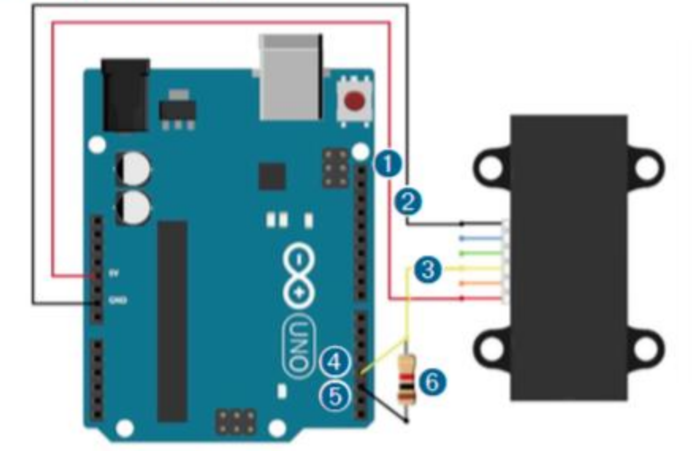
James Oliveros

Standard Arduino I2C Wiring



Item	Description	Notes
1	680µF electrolytic capacitor	You must observe the correct polarity when installing the capacitor.
2	I2C SCA connection	Green wire
3	I2C SDA connection	Blue wire
4	Power ground (-) connection	Black wire
5	5 Vdc power (+) connection	Red wire The sensor operates at 4.75 through 5.5 Vdc, with a max. of 6 Vdc.

PWM Arduino Wiring



Item	Description	Notes
1	5 Vdc power (+) connection	Red wire The sensor operates at 4.75 through 5.5 Vdc, with a max. of 6 Vdc.
2	Power ground (-) connection	Black Wire
3	Mode-control connection	Yellow wire
4	Monitor pin on microcontroller	Connect one side of the resistor to the mode-control connection on the device, and to a monitoring pin on your microcontroller.
5	Trigger pin on microcontroller	Connect the other side of the resistor to the trigger pin on your microcontroller.
6	1kΩ resistor	

James Oliveros

LIDAR code in PWM

```
unsigned long pulse_width;

void setup()
{
  Serial.begin(9600); // Start serial communications
  pinMode(2, OUTPUT); // Set pin 2 as trigger pin
  pinMode(3, INPUT); // Set pin 3 as monitor pin
  pinMode(4, OUTPUT); // Set pin 4 to control power enable line
  digitalWrite(4,HIGH); //Turn sensor on
  digitalWrite(2, LOW); // Set trigger LOW for continuous read
}

void loop()
{
  pulse_width = pulseIn(3, HIGH); // Count how long the pulse is high in microseconds
  if(pulse_width != 0){ // If we get a reading that isn't zero, let's print it
        pulse_width = pulse_width/10; // 10usec = 1 cm of distance for LIDAR-Lite
    Serial.println(pulse_width); // Print the distance
  }else{ // We read a zero which means we're locking up.
    digitalWrite(4,LOW); // Turn off the sensor
    delay(1);// Wait 1ms
    digitalWrite(4,HIGH); //Turn on te sensor
    delay(1);//Wait 1ms for it to turn on.
  }
  delay(1); //Delay so we don't overload the serial port
}
```

James Oliveros

Microstepping

1/8th Microstep revolution should be sufficient for our purposes

(200 steps) x (8 microsteps) = 1600 microsteps per revolution

(360 degrees) / 1600 microsteps = .225 degrees per microstep

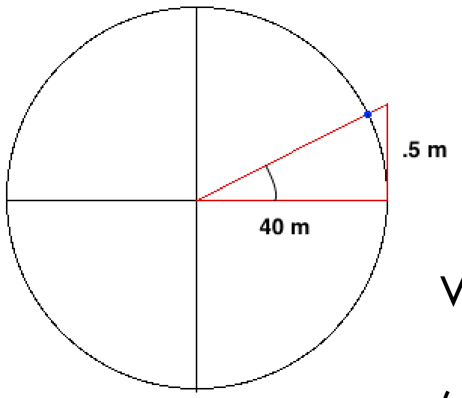
3x.225 = .675 degrees (closest to worst case scenario degree movement)

360 / .675 = 533 measurements per plane

Table 1. Microstep Resolution Truth Table

MS1	MS2	Resolution
L	L	Full step (2 phase)
H	L	Half step
L	H	Quarter step
H	H	Eighth step

EasyDriver A3967 Motor Driver Specifications



Worst-case-scenario: .7162 degrees

(.5 m accuracy @ 40 m distance)

James Oliveros

Arduino Memory Capabilities

(360 / .675) Degrees = 533.3333 Data points
= 533 Data points on one plane

533^2 (for two planes) = 253009 Data points
= 253009 Bytes of Data (1 bytes per data point)
= 253.009 kB

Arduino local flash memory capacity is 256 kB, therefore, some kind of external (SD) memory is required for multiple point of data acquisition

503 x 503 x 20 Milliseconds per data point = 1.405 Hours

LIDAR + IMU code: Setup

James Oliveros

```
#include <Wire.h>
#include <SPI.h>
#include <SparkFunLSM9DS1.h>

LSM9DS1 imu;

//I2C Setup
#define LSM9DS1_M 0x1E // Would be 0x1C if SDO_M is LOW
#define LSM9DS1_AG  0x6B // Would be 0x6A if SDO_AG is LOW

//Output settings
#define PRINT_CALCULATED
//#define PRINT_RAW
#define PRINT_SPEED 250 // 250 ms between prints
static unsigned long lastPrint = 0; // Keep track of print time

//calculate mag field by location here:
// http://www.ngdc.noaa.gov/geomag-web/#declination
#define DECLINATION -8.58 // Declination (degrees) in Boulder, CO.

unsigned long pulseWidth;
```

LIDAR + IMU code: Setup

James Oliveros

```
void setup()
{

  Serial.begin(9600);

  // LiDAR pin declarations
  pinMode(2, OUTPUT); // Set pin 2 as trigger pin
  pinMode(3, INPUT); // Set pin 3 as monitor pin
  pinMode(4, OUTPUT); // Set pin 4 to control power enable line
  digitalWrite(4,HIGH); //Turn sensor on
  digitalWrite(2, LOW); // Set trigger LOW for continuous read

  imu.settings.device.commInterface = IMU_MODE_I2C;
  imu.settings.device.mAddress = LSM9DS1_M;
  imu.settings.device.agAddress = LSM9DS1_AG;
  if (!imu.begin())
  {
    Serial.println("Failed to communicate with LSM9DS1.");
    Serial.println("Double-check wiring.");
    Serial.println("Default settings in this sketch will " \
                  "work for an out of the box LSM9DS1 " \
                  "Breakout, but may need to be modified " \
                  "if the board jumpers are.");
    while (1)
      ;
  }
}
```

LIDAR + IMU code: Implementation

James Oliveros

```
void loop()
{
  pulseWidth = pulseIn(3, HIGH); // Count how long the pulse is high in microseconds

  if(pulseWidth != 0)
  {
    pulseWidth = pulseWidth / 10; // 10usec = 1 cm of distance
    // Serial.println(pulseWidth); // Print the distance
  }
  if ( imu.gyroAvailable() )
  {
    imu.readGyro();
  }
  if ( imu.accelAvailable() )
  {
    imu.readAccel();
  }
  if ( imu.magAvailable() )
  {
    imu.readMag();
  }

  if ((lastPrint + PRINT_SPEED) < millis())
  {
    Serial.print("Distance, Pitch, Heading: ");
    Serial.println(pulseWidth); // Print the distance
    printAttitude(imu.ax, imu.ay, imu.az,
                 -imu.my, -imu.mx, imu.mz);
    Serial.println();

    lastPrint = millis(); // Update lastPrint time
  }
}
```

To do: Programming

James Oliveros

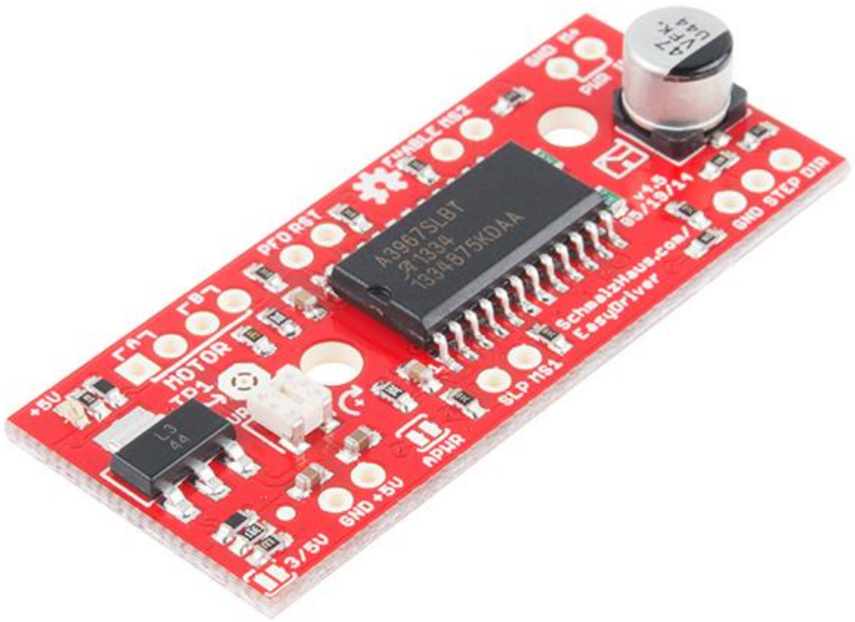
- Add error checking
- Potential average function to remove outliers / smooth out data acquisition
- Add preliminary timer
- Incorporate LIDAR+IMU code and stepper motor code
- Field tests with full scans
- Configure external memory storage
- Data relay to CAD

Easy Driver – Stepper Motor Driver

Cesar Rivas

- Knob on driver controls current supplied to motor.
- “Direction” pin: HIGH = clockwise, LOW = counterclockwise
- “Step” pin: driver makes motor step once every time this pin goes from LOW to HIGH.
- “MS1” and “MS2” pins configure the length of each step.

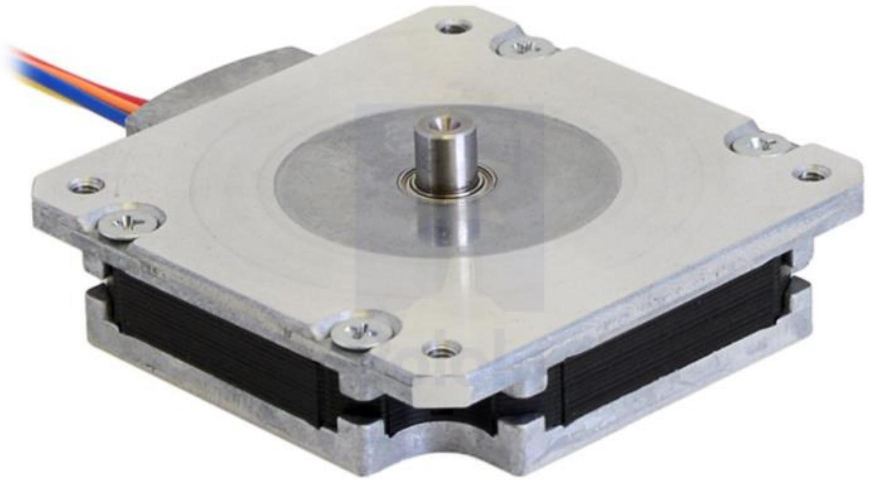
MS1	MS2	Resolution
L	L	Full step (2 phase)
H	L	Half step
L	H	Quarter step
H	H	Eighth step



Sanyo Pancake Stepper Motor

Cesar Rivas

- Provides horizontal stepping functionality.
- 1.8° step angle: 200 steps/revolution
- Holding torque: 14 (oz-in)
- Flat profile, 16 mm including shaft
- Current rating: 1 (A) per coil
- Voltage rating: 4.5 (V) per coil
- Resistance: 4.5 (Ω)
- Inductance: 2 (mH)



Nema 8 Stepper Motor

Cesar Rivas

- Provides vertical stepping functionality.
- 1.8° step angle: 200 steps/revolution
- Holding torque: 3.0 (oz-in)
- Current rating: 0.6 (A) per coil
- Voltage rating: 4.5 (V) per coil
- Resistance: 6.5 (Ω)
- Inductance: 1.7 (mH)

Stepper Motor Module

Cesar Rivas

```
#define hMS1 11  //Defines pins on microcontroller
#define hMS2 10
#define hstep_pin 9
#define hdir_pin 8

#define vMS1 7
#define vMS2 6
#define vstep_pin 5
#define vdir_pin 4
```

Stepper Motor Module

Cesar Rivas

```
void setup()
{
  Serial.begin(9600);

  pinMode(hstep_pin, OUTPUT);  //Setup for horizontal easy driver
  pinMode(hdir_pin, OUTPUT);
  pinMode(hMS1, OUTPUT);
  pinMode(hMS2, OUTPUT);
  digitalWrite(hstep_pin, LOW);
  digitalWrite(hdir_pin, LOW); //Sets stepping direction to counterclockwise
  digitalWrite(hMS1, LOW);  //Sets pancake motor to 1/8th microstep mode
  digitalWrite(hMS2, LOW);

  pinMode(vstep_pin, OUTPUT); //Setup for vertical easy driver
  pinMode(vdir_pin, OUTPUT);
  pinMode(vMS1, OUTPUT);
  pinMode(vMS2, OUTPUT);
  digitalWrite(vstep_pin, LOW);
  digitalWrite(vdir_pin, LOW); //Sets stepping direction to counterclockwise
  digitalWrite(vMS1, LOW); //Sets Nema 8 motor to 1/8th microstep mode
  digitalWrite(vMS2, LOW);
}
```

Stepper Motor Module

Cesar Rivas

```
void scan() //Function for vertical scanning of 10 points
{
  for(int i=0;i<10;i++)
  {
  digitalWrite(vstep_pin, HIGH);
  digitalWrite(vstep_pin, LOW);
  delay(500);
  }
}
```

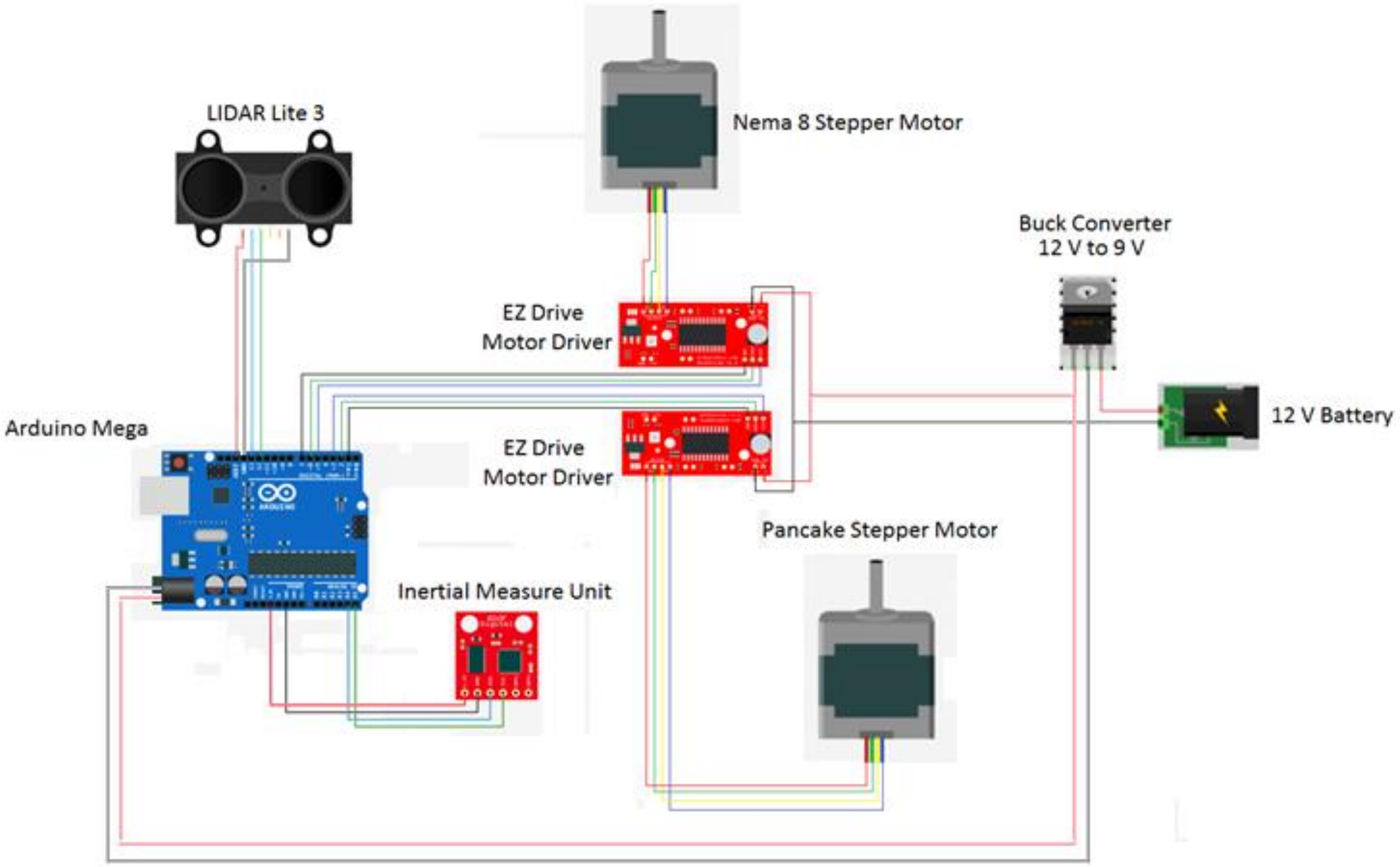
Stepper Motor Module

Cesar Rivas

```
void loop() //Loop for 200-point scan
{
  int j;
  for(j=0;j<10;j++)
  {
    digitalWrite(vdir_pin, HIGH); //sets vertical stepping direction to clockwise

    scan(); //Upward scan of 10 points

    digitalWrite(hstep_pin, HIGH); //steps horizontal motor once
    digitalWrite(hstep_pin, LOW);
    delay(500);

    digitalWrite(vdir_pin, LOW); ////sets vertical stepping direction to counter clockwise

    scan(); //Downward scan of 10 points

    digitalWrite(hstep_pin, HIGH); //steps horizontal motor once
    digitalWrite(hstep_pin, LOW);
    delay(500);
  }
  delay(10000);
  return 0;
}
```

Jake Ogburn

Wire Diagram

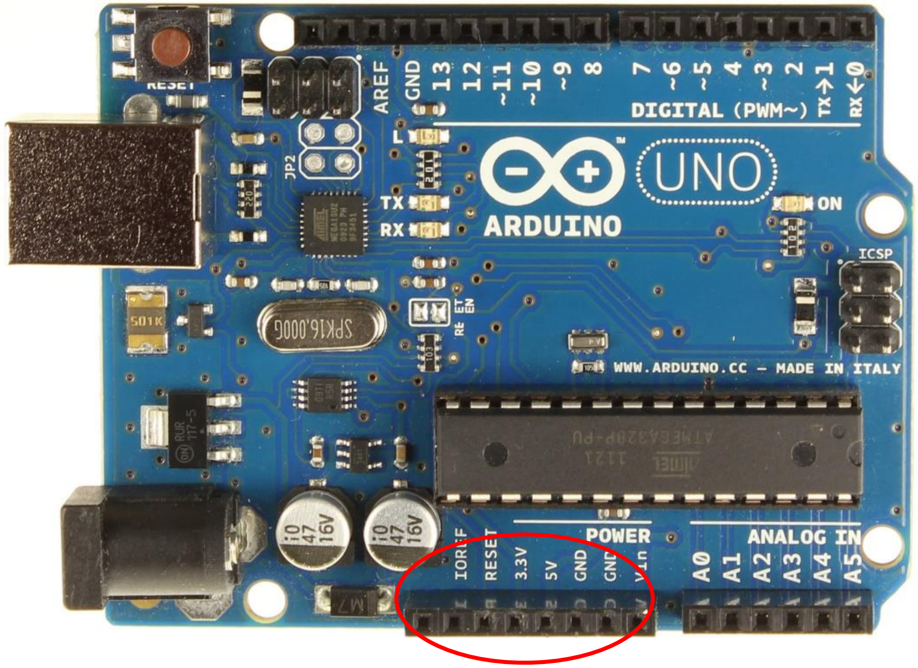


Jake Ogburn

Power Supply: Arduino Uno

	Voltage (V)	Current (mA)
LIDAR Lite v3	5	135
Inertial Measurement Unit	3.3	0.6

Arduino Uno Onboard Voltage Supply	
Output Voltage (V)	Max Current (mA)
5	500
3.3	50



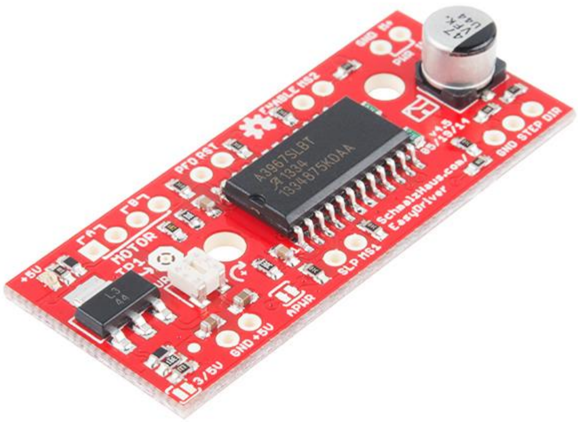
Arduino Uno voltage supply pins

Jake Ogburn

Power Supply: EZ Drive Motor Driver

	Voltage (V)	Resistance/Coil (Ω)	Current (A)
NEMA 8 Stepper Motor	4.0	6.5	0.6
Sanyo Pancake Stepper Motor	4.5	4.5	1.0

	Supplied Voltage (V)	Current (A)
EZ Drive Motor Driver	1.95	0.30
EZ Drive Motor Driver	2.25	0.50



EZ Drive Motor Driver

Jake Ogburn

Power Supply: 12V Battery

	Voltage (V)	Current (A)
Arduino Mega	9.0	0.2
EZ Drive Motor Driver	9.0	0.2
EZ Drive Motor Driver	9.0	0.3

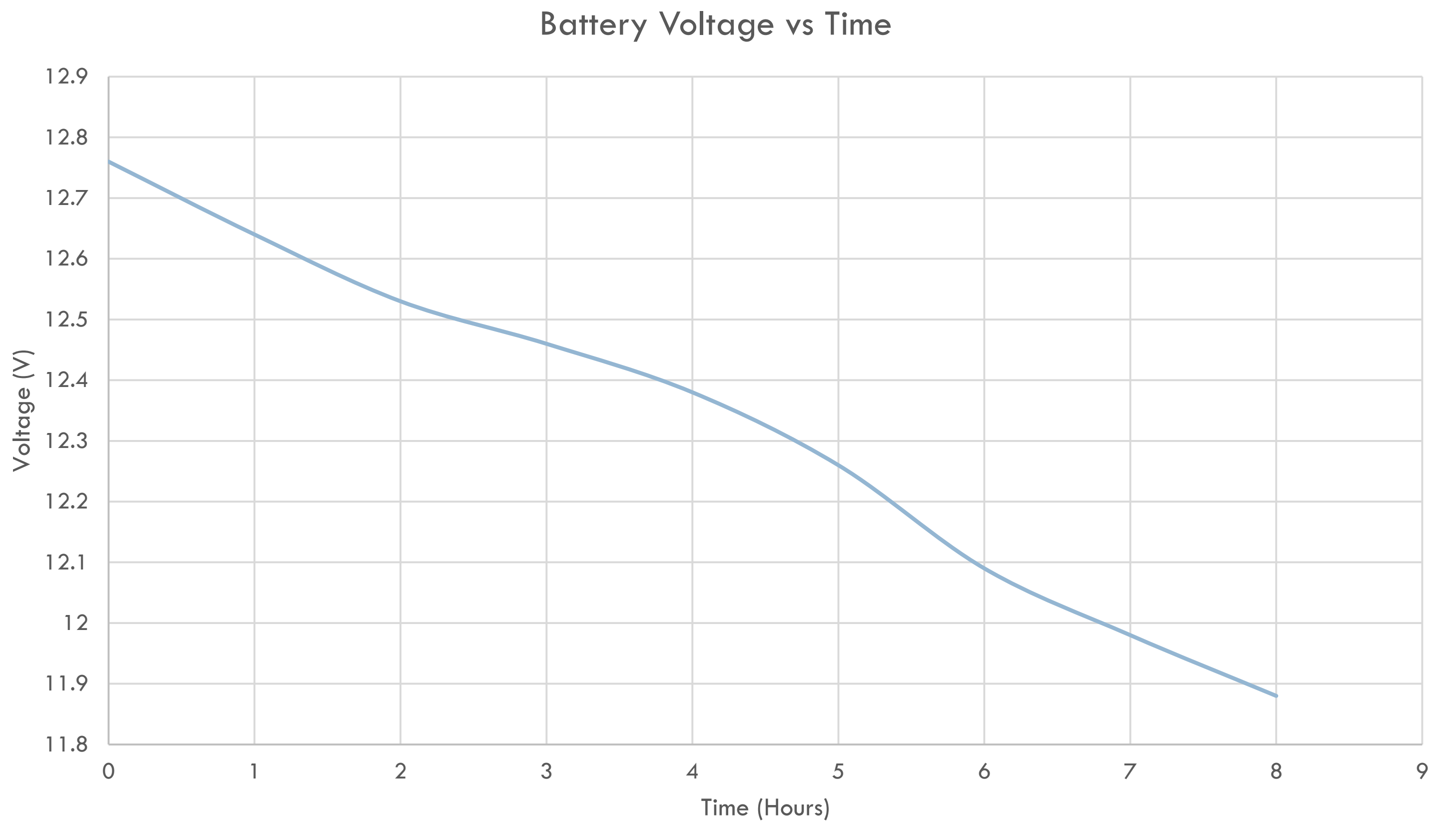
Current Consumption:	0.7 A
Current Consumption for One Target Area:	2.8 A

Battery Capacity:	7.0 Ah
Target Areas Mapped:	2.50

Power Sonic PS-1270 F1 Battery

Jake Ogburn

Battery Performance



Jake Ogburn

System Values

Time (Hr)	Battery Voltage (V)	Buck Converter (V)	5 V Output (V)	3.3 V Output (V)
0	12.76	9.03	5.1	3.3
1	12.64	9.03	5.1	3.3
2	12.53	9.03	5.1	3.3
3	12.46	9.03	5.1	3.3
4	12.38	9.03	5.1	3.3
5	12.26	9.03	5.1	3.3
6	12.09	9.03	5.1	3.3
7	11.98	9.03	5.1	3.3
8	11.88	9.03	5.1	3.3

References

- "About - ILMF." ILMF. N.p., n.d. Web. 11 Oct. 2016.
- "Caving Projects." Southern Arizona Grotto. N.p., n.d. Web. 10 Oct. 2016.
- "LIDAR-Lite V3." - SEN-14032. N.p., n.d. Web. 4 Oct. 2016.
- "Slamtec's RPLIDAR A2 Has a Range of 6 Meters And Can Take Up To 4000 Samples of Laser Ranging per Second ($480)." Into Robotics. N.p., n.d. Web. 4 Oct. 2016.
- "Underwater Cave (low Poly)." 3D Model :. N.p., n.d. Web. 10 Oct. 2016.
- "Vermont Center ForGeographic Information." VCGI LiDAR Program. N.p., n.d. Web. 11 Oct. 2016.
- "What Is LIDAR." US Department of Commerce, National Oceanic and Atmospheric Administration. N.p., n.d. Web. 6 Oct. 2016.
- England, Emily. "The Effigy Mounds and Mallam | Anthropology | Luther College." The Effigy Mounds and Mallam | Anthropology | Luther College. N.p., n.d. Web. 13 Oct. 2016.
- Nguyen, Quyen. "Running a Raspberry Pi from 6 AA Batteries." Tutorial Raspberry Pi :. N.p., 1970. Web. 12 Oct. 2016.
- S. Kish and B. Broedel, "Introduction to Cave Mapping Design Project," 2016.
- S. Kish, "Mapping of the Wakulla Springs Conduit-Cave System Using LIDAR and Inertial Navigation Sensor System– Prototype Modeling,".
- "Arduino vs. Raspberry Pi: Mortal Enemies, or Best Friends?" *Digital Trends*. Digital Trends, 2015. Web. 17 Nov. 2016.

- Questions?