CENTAURUS DESIGN REVIEW

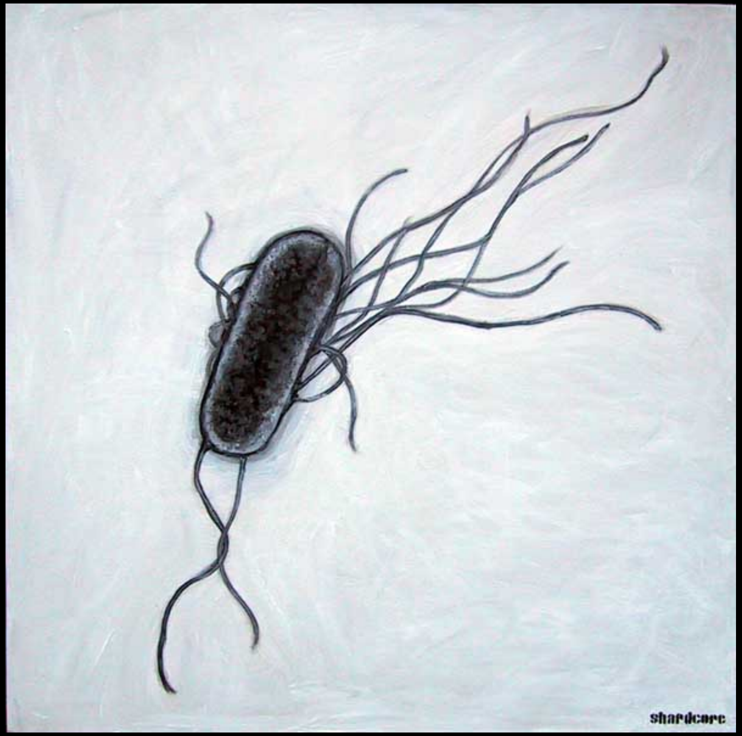
November 13th

Bryan McCarty, Cooper McCormick, Sam Cuthbertson, Abigail Schmid, Jack Carvalho

CONTENTS

DESIGN OVERVIEW

- “The Effects of Simulated Gravity on Bacterial Lag Phase in a micro-gravitational Environment”
  - Research into specific effects of gravity on bacterial growth



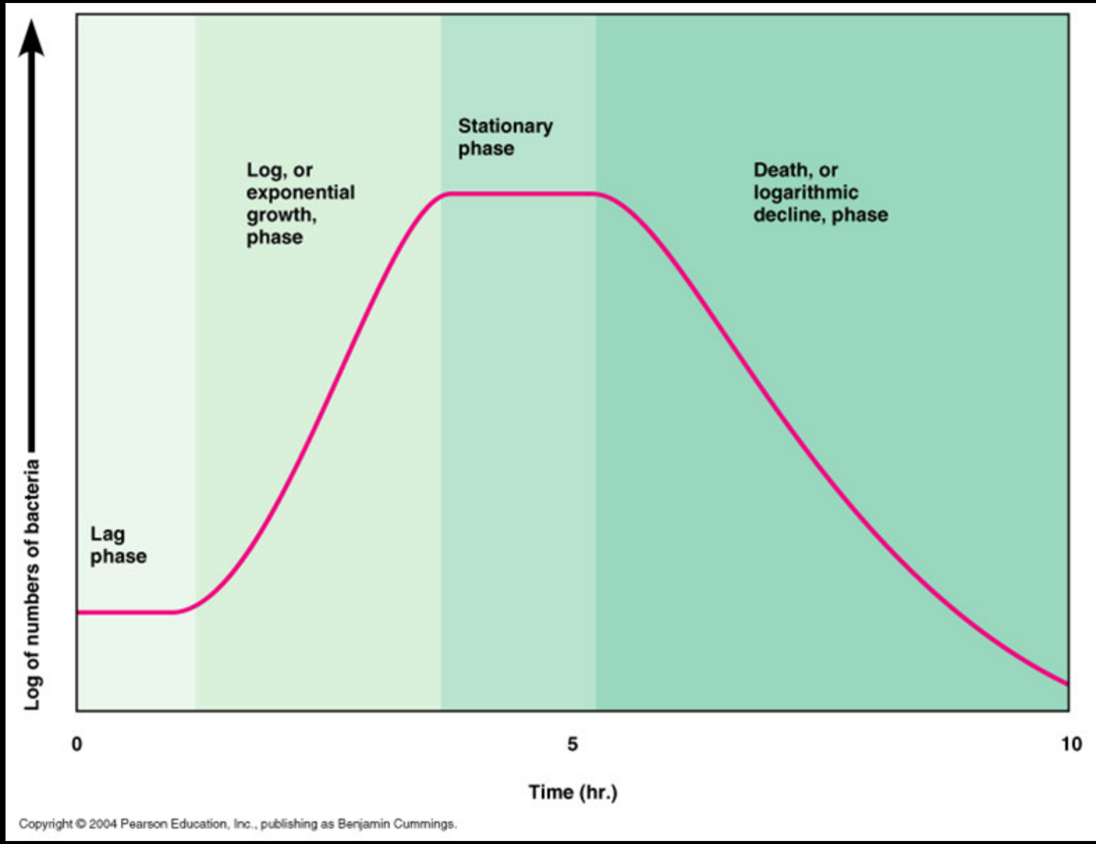
CONCEPTUALIZATION

Similar Research:

- Daniel Madar, Erez Dekel, Anat Bren, Anat Zimmer, Ziv Porat, and Uri Alon— "Promoter activity dynamics in the lag phase of *Escherichia coli"*
- Natalie Leys, Larissa Hendrickx, Patrick De Boever, Sarah Baatout, Max Mergeay— "Space flight effects on bacterial physiology"
- David Klaus, Steven Simske, Paul Todd and Louis Stodieck— "Investigation of space flight effects on Escherichia coli and a proposed model of underlying physical mechanisms"
- M. R. Benoit, W. Li, L. S. Stodieck, K. S. Lam, C. L. Winther, T. M. Roane, D. M. Klaus– "Microbial antibiotic production aboard the International Space Station"
- Rensselaer Polytechnic Institute– "Zero-gravity and low nutrient environment"
- B. Purevdorj-Gage, K. B. Sheehan, and L. E. Hyman– "Effects of Low-Shear Modeled Microgravity on Cell Function, Gene Expression, and Phenotype in *Saccharomyces cerevisiae"*

RESEARCH ANALYSIS

- Lag phase is shorter in space
- Thought to be due to microgravity
- Early method ideas (injection systems)

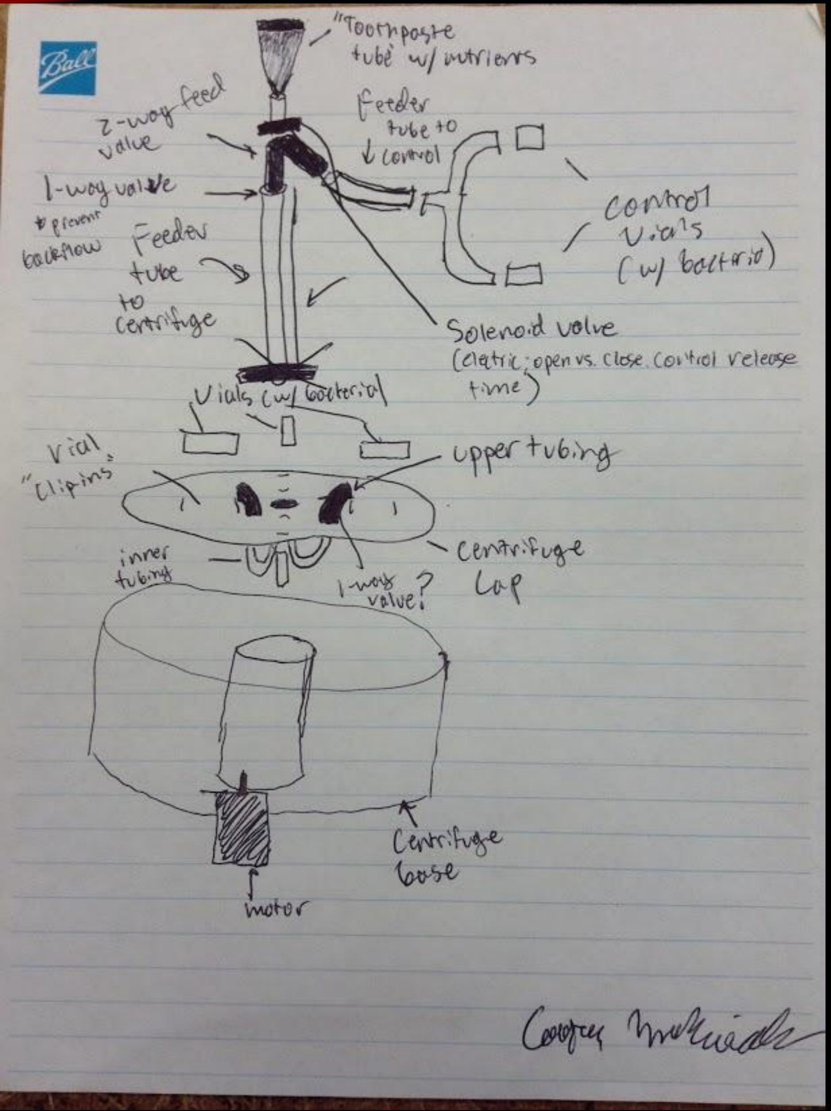


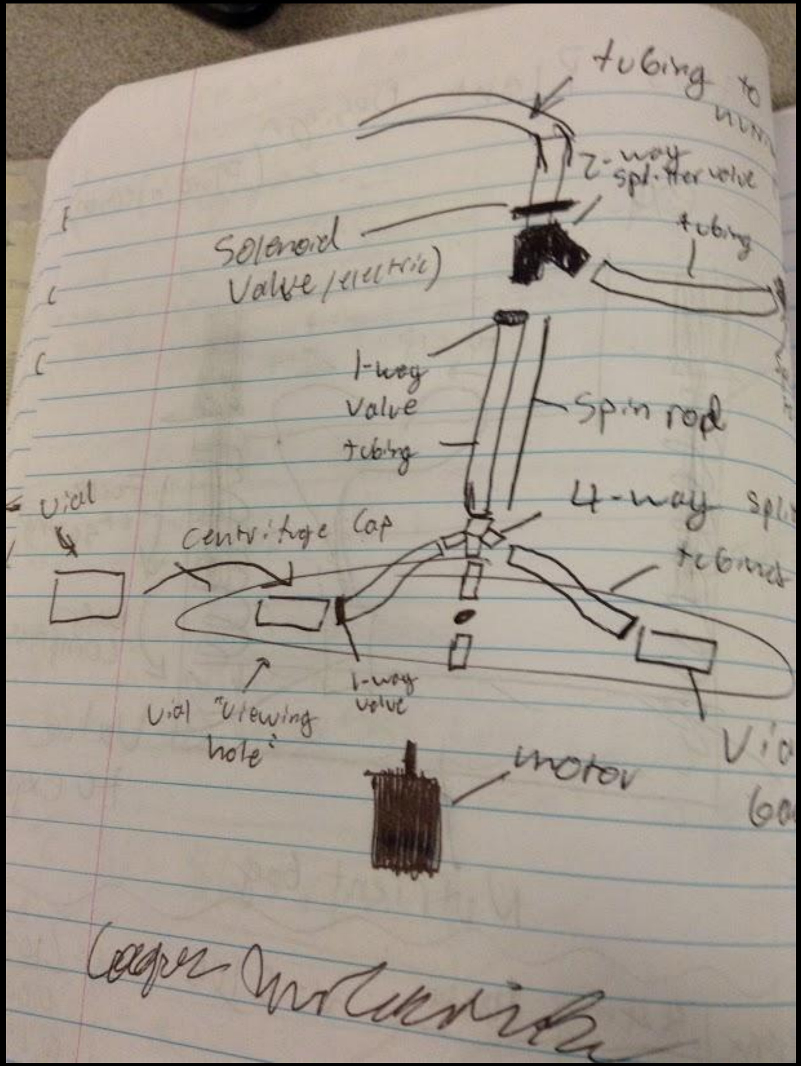
INITIAL DESIGN IDEAS

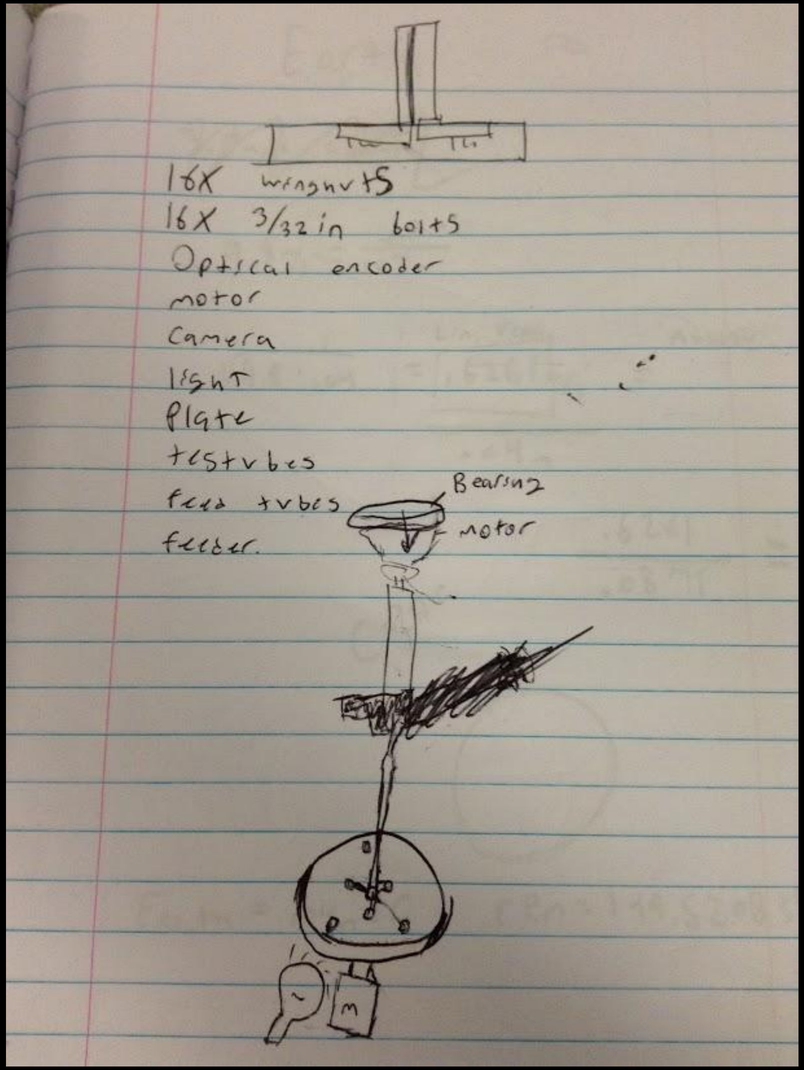
Central nutrient systems vs. individual vials

Mechanized injection

Centrifugal injection

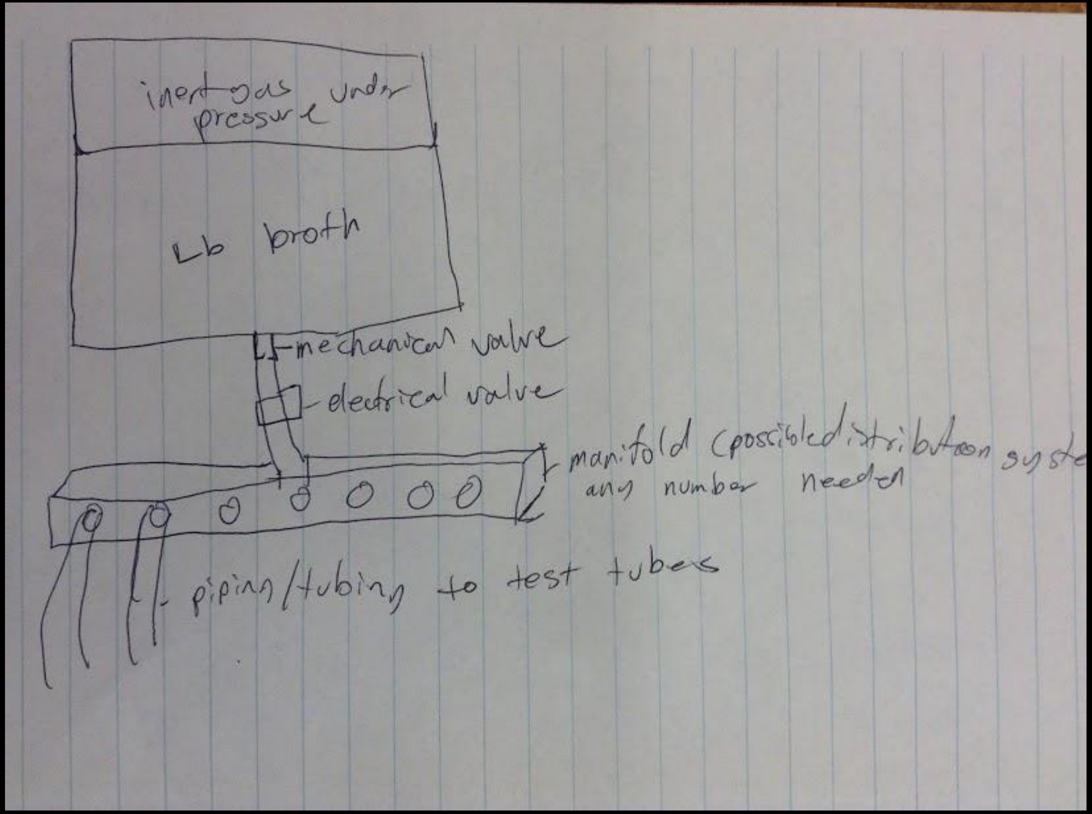


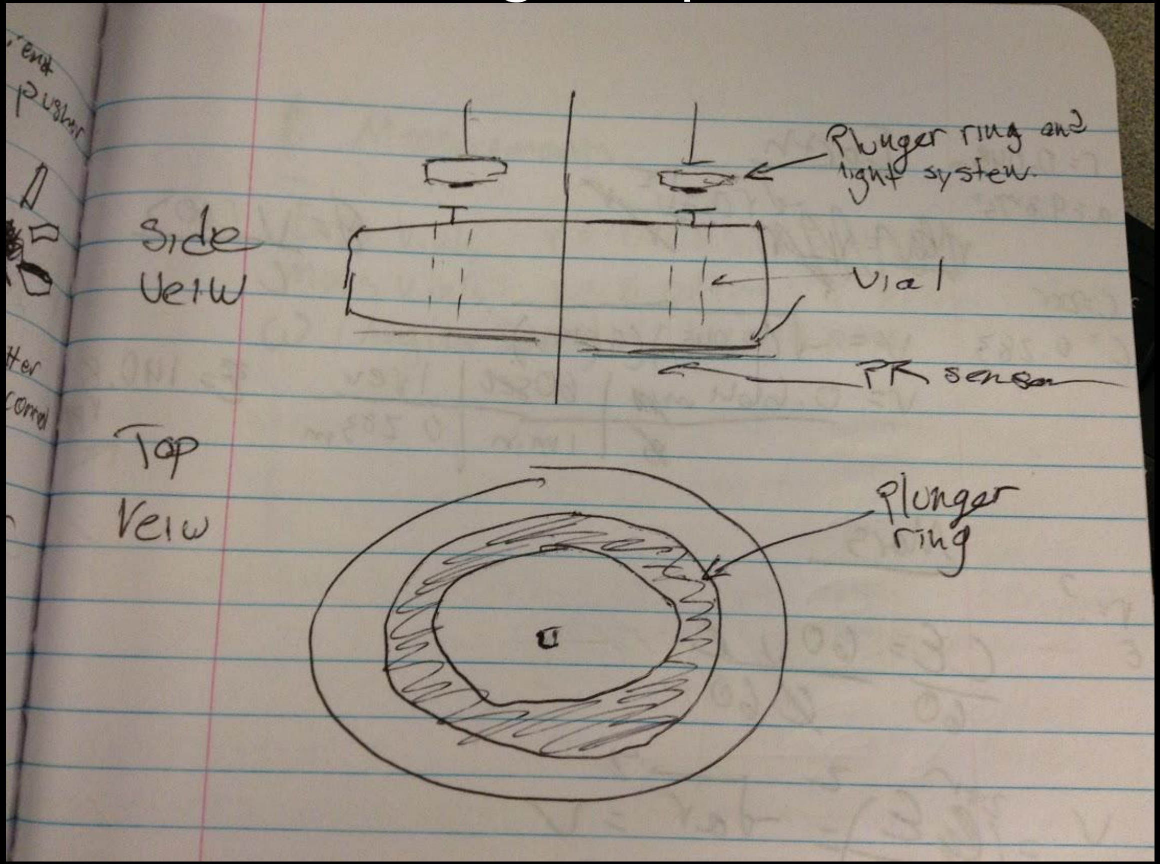




Pressurized injection

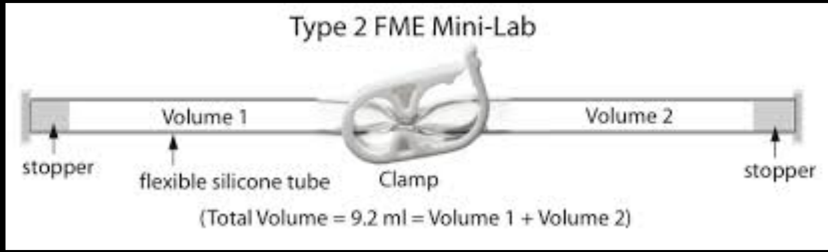
Plunger injection



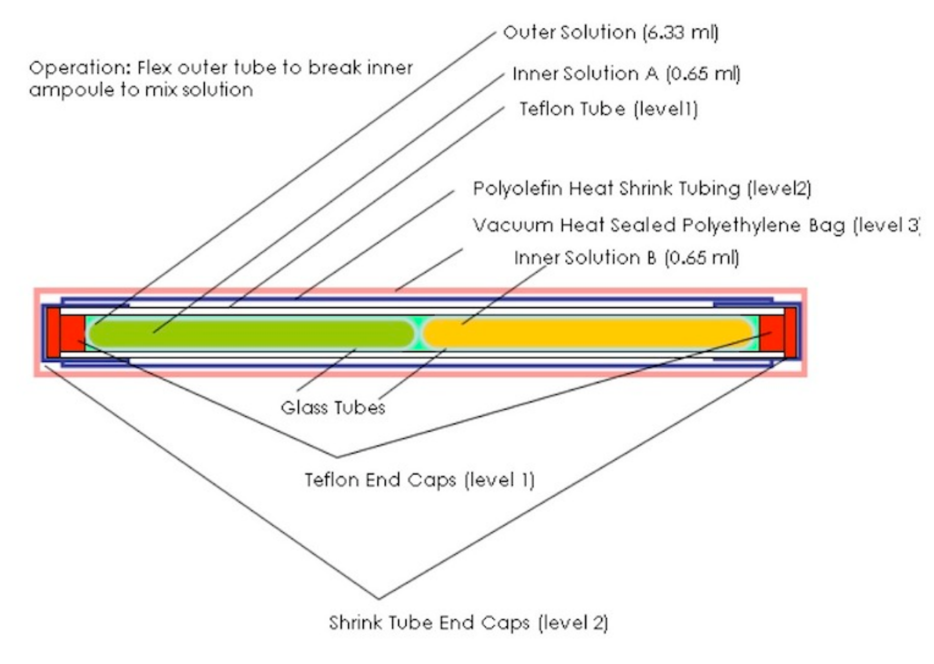


INDIVIDUAL VIALS

Separated in vials



- Bioserve
- “glow stick” concept



Premix and freeze

INJECTION SYSTEM DECISION MATRIX

Decision Matrix

Rate: 1-5	WEIGHT	Gravitational Injection		Mechanized Fluid Injection		Pressurized Fluid Injection		Premixed Vials	
Specifications		**Rating**	**Weighted Score**	**Rating**	**Weighted Score**	**Rating**	**Weighted Score**	**Rating**	**Weighted Score**
Ease of Construction	**2**	3	6	1	2	2	4	5	10
Cost of Materials	**1**	2	2	1	1	2	2	4	4
Availability of Materials	**1**	2	2	1	1	2	2	5	5
Funtionality	**3**	2	6	3	9	3	9	5	15
Probability of Success	**3**	3	9	2	6	1	3	1	3
TOTAL			25		19		20		37

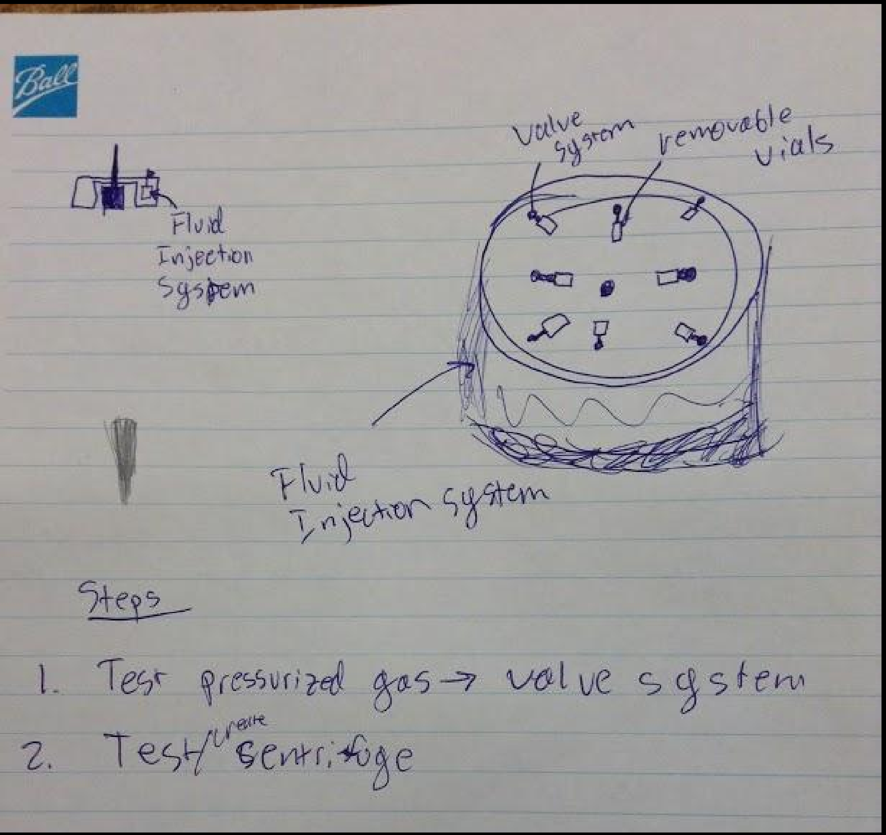
Notes: Originally, we weren't considering the premixed because even though it scored the highest, we thought it had no probability of success. We later learned that premixed would be possible, so we decided on that design.

DEVELOPING TESTABLE SOLUTION

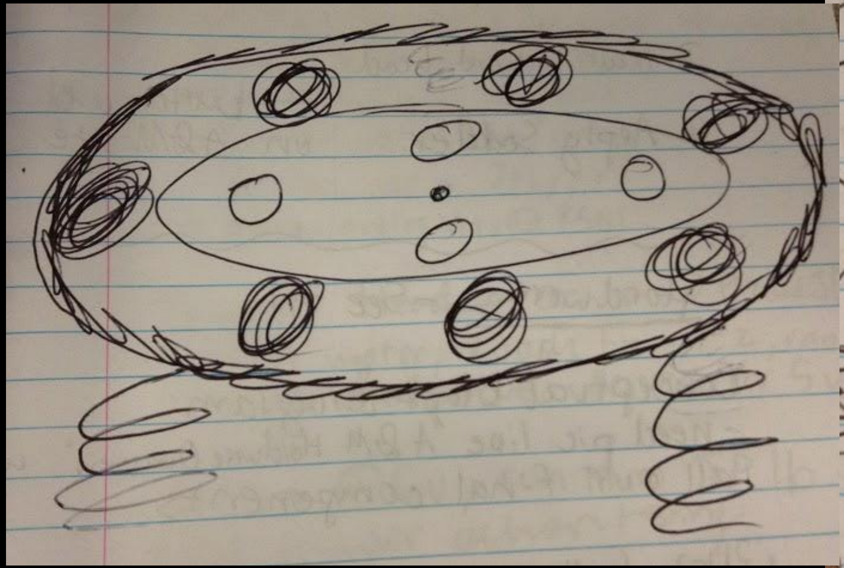
- Components
  - Motor
  - Gear system/centrifuge
  - Casing
  - Sensors
  - Bacteria
- Background info tests
  - Biology
  - Hardware
  - Software
- Data collection methods
  - Use of LED and photoresistor

CENTRIFUGE

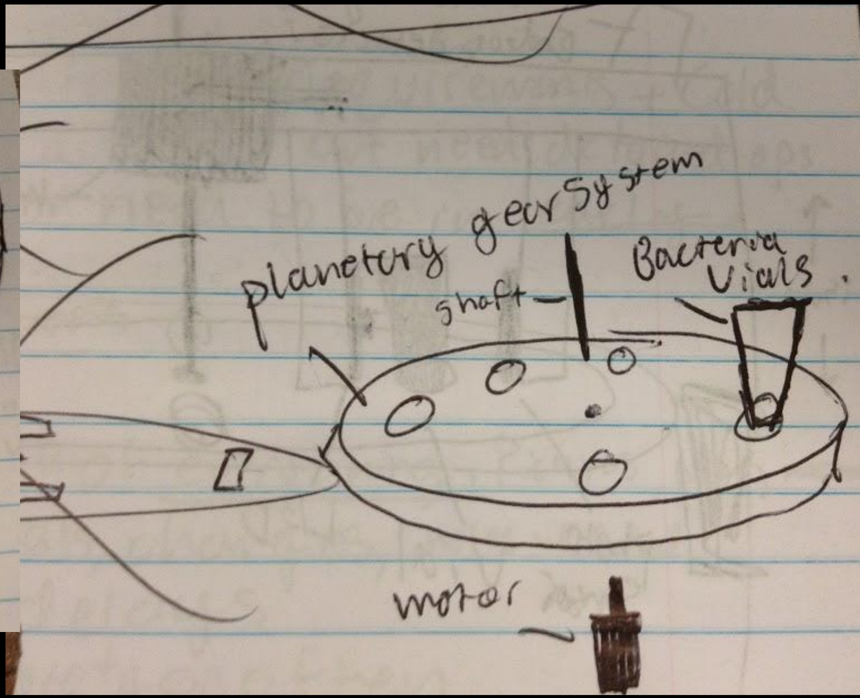
Initial centrifuge



Centrifuge with springs



Planetary gear



$$a = \frac{v^2}{r}$$

$$v = \sqrt{9.81r}$$

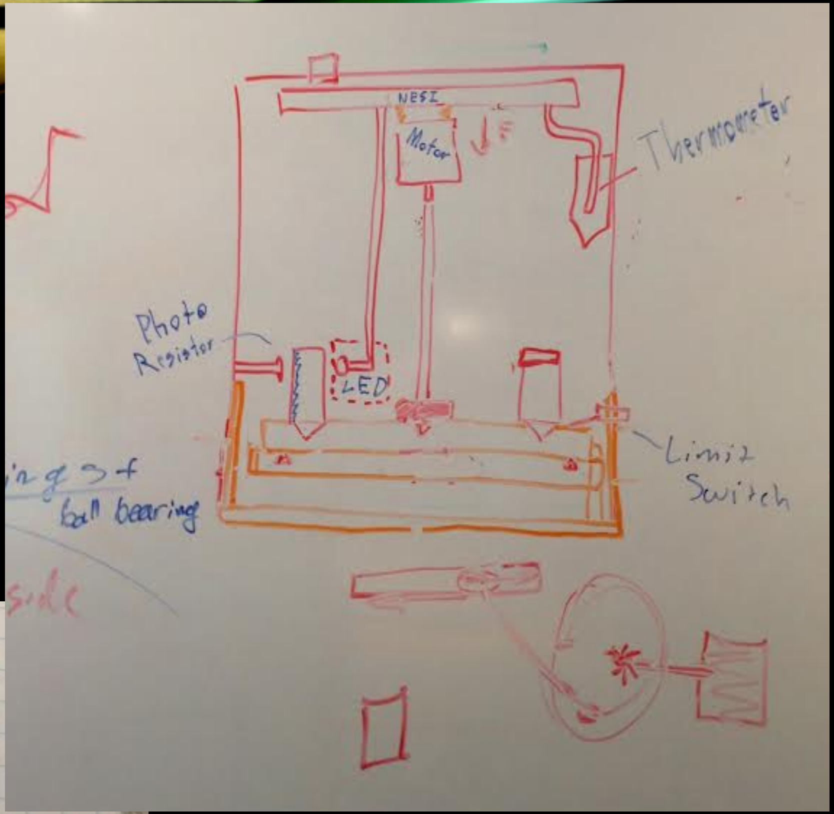
CENTRIFUGE DECISION MATRIX

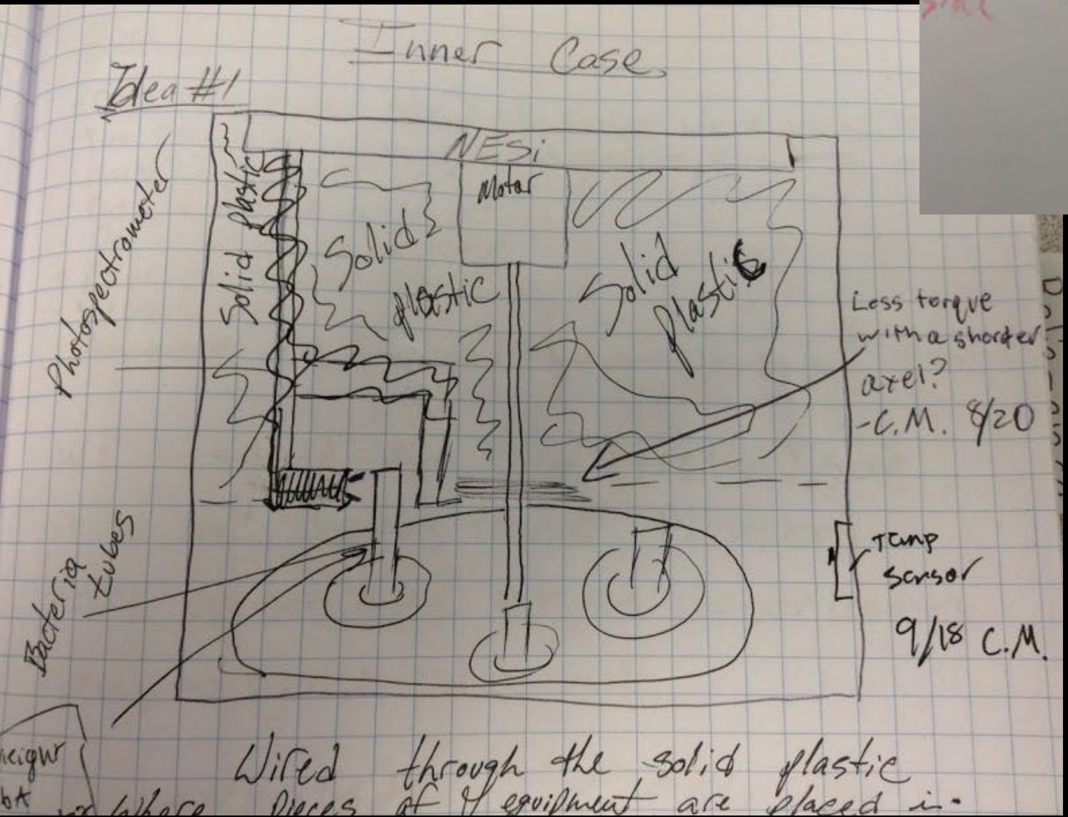
Decision Matrix

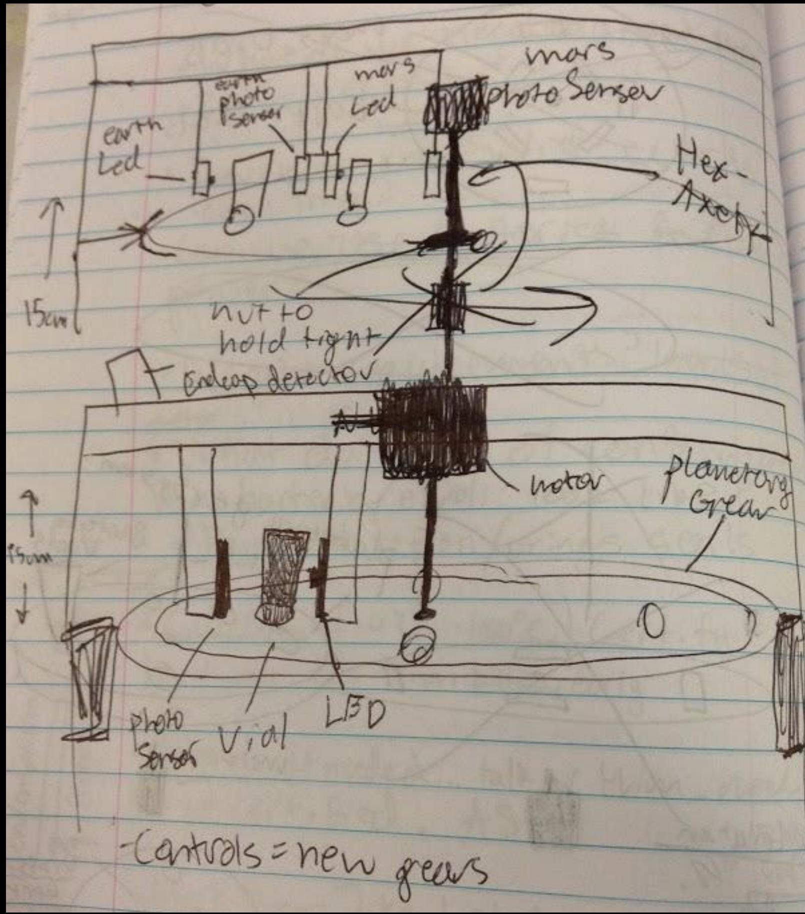
Score: 1-5	WEIGHT	Rotating Centrifuge		Planetary Gear Centrifuge System		Rotating Centrifuge on Springs	
Specifications		**Rating**	**Weighted Score**	**Rating**	**Weighted Score**	**Rating**	**Weighted Score**
Ease of Construction	**2**	4	8	4	8	2	4
Cost of Materials	**1**	4	4	4	4	3	3
Availability of Materials	**1**	4	4	4	4	2	2
Funtionality	**3**	2	6	4	12	3	9
Probability of Success	**3**	1	3	4	12	3	9
TOTAL			25		40		27

Notes: In order to both achieve the correct acceleration of the centrifuge and mix the bacteria for correct growth, centrifuge spinning and bacteria agitation must be considered heavily. We found that coming along with the highest score, the planetary gear system was the best possible option to achieve both goals most efficiently.

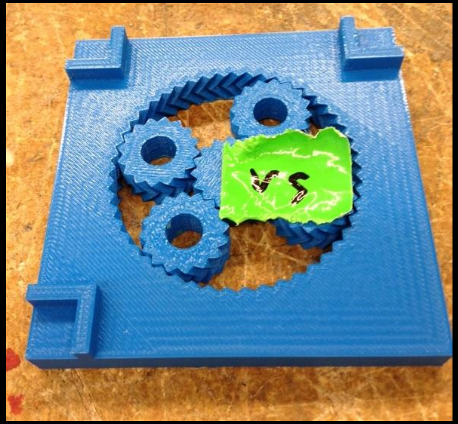
PROTOTYPE SKETCH

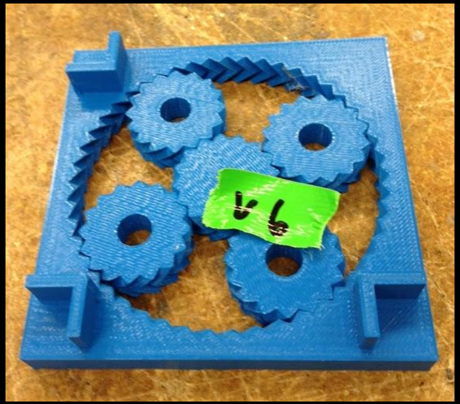


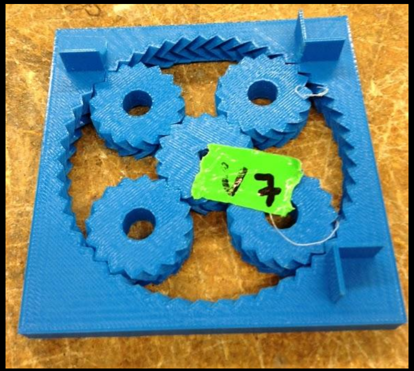


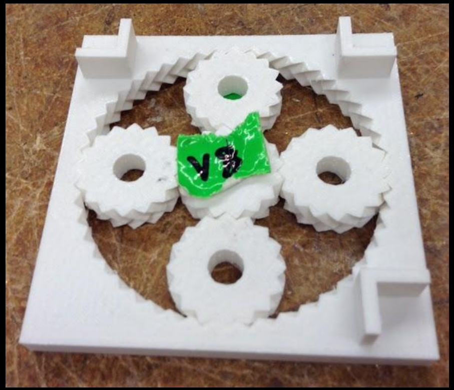


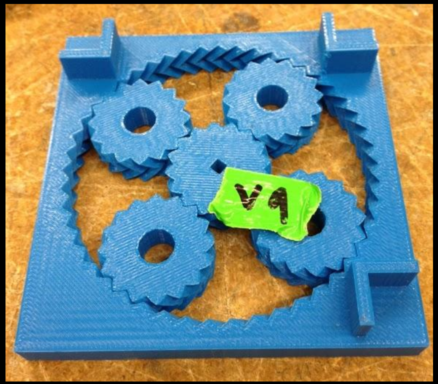
PLANETARY GEAR ITERATIONS

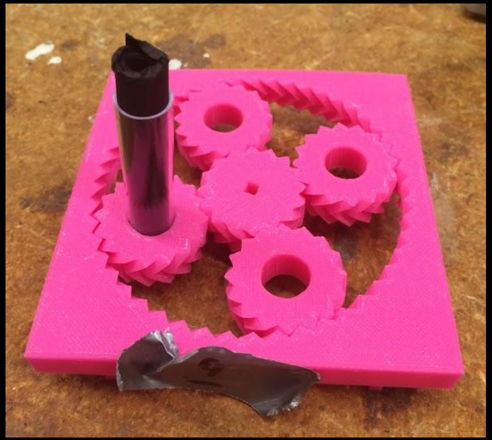




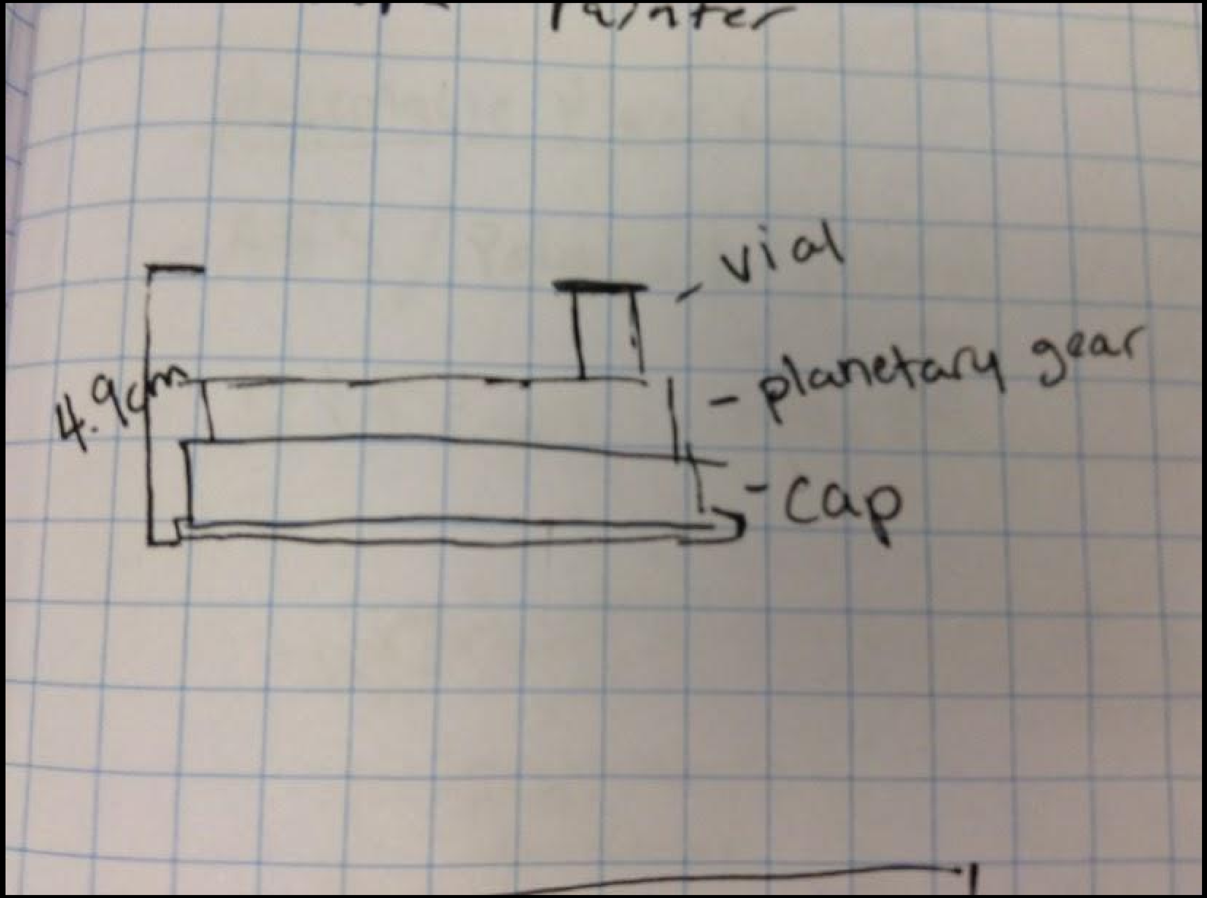


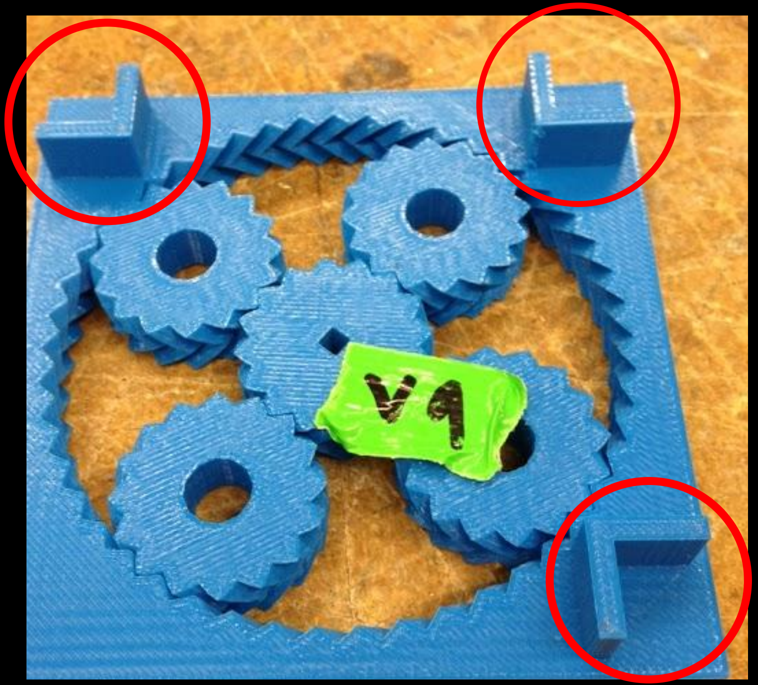




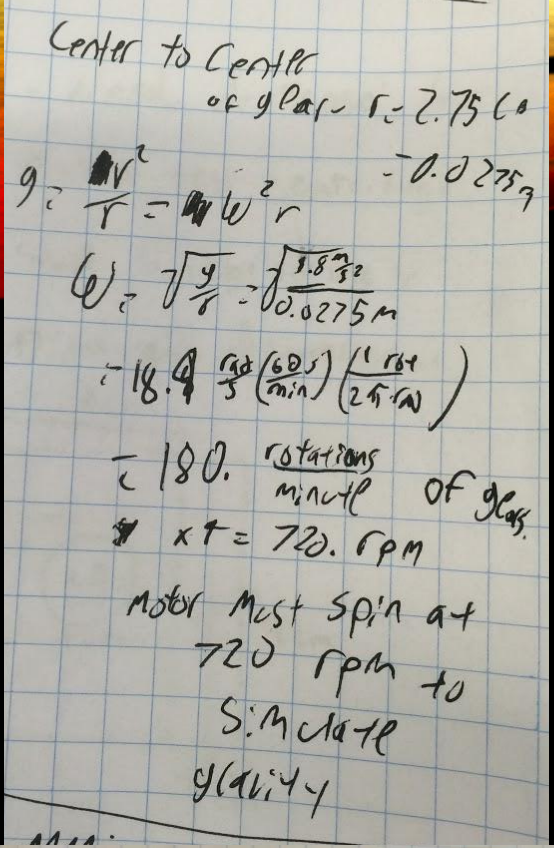


END CAP INSERT





MOTORS



Center to Center of gear – r = 2.75 cm = 0.0275 m

$g = \frac{\cancel{m}v^2}{r} = \cancel{m}\omega^2 r$

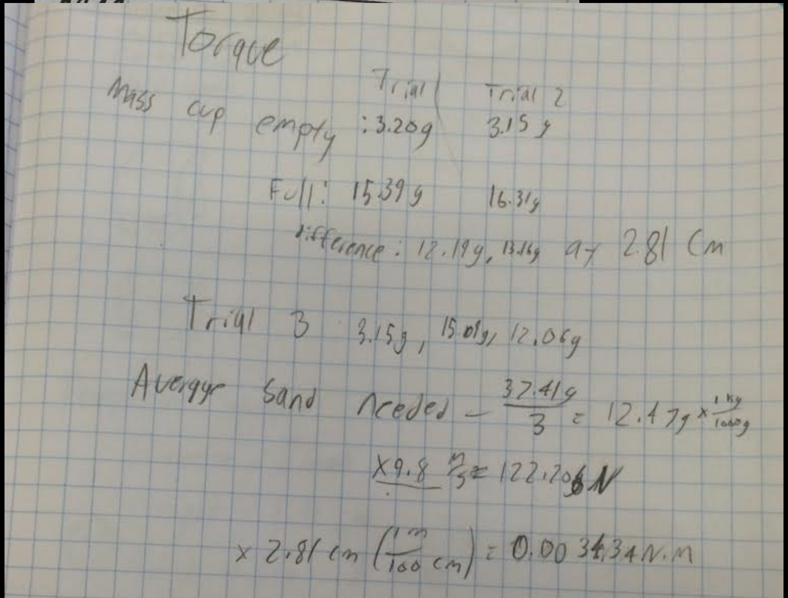
$\omega = \sqrt{\frac{g}{r}} = \sqrt{\frac{9.8\ m/s^2}{0.0275\ m}}$

$= 18.9\ \frac{rad}{s}\left(\frac{60\ s}{min}\right)\left(\frac{1\ rot}{2\pi\ rad}\right)$

= 180. rotations/minute of gear

×4 = 720. rpm

motor must spin at 720 rpm to simulate gravity



Torque

	Trial 1	Trial 2
Mass cup empty	3.20g	3.15g
Full	15.39g	16.31g
Difference	12.19g	13.16g

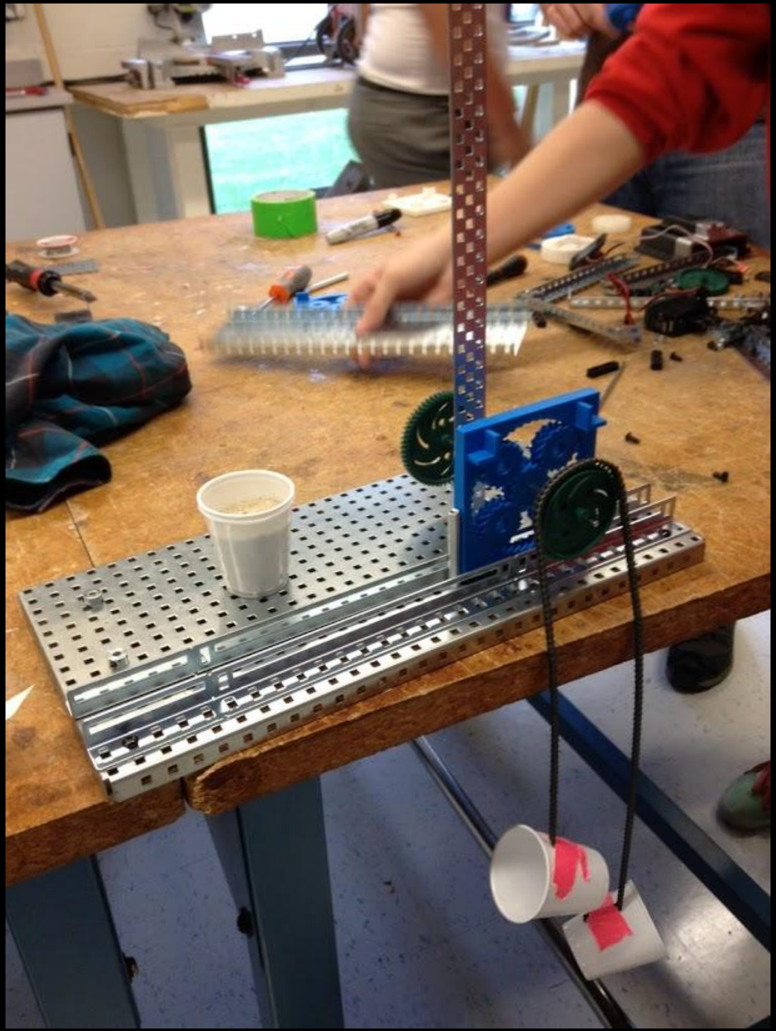
at 2.81 cm

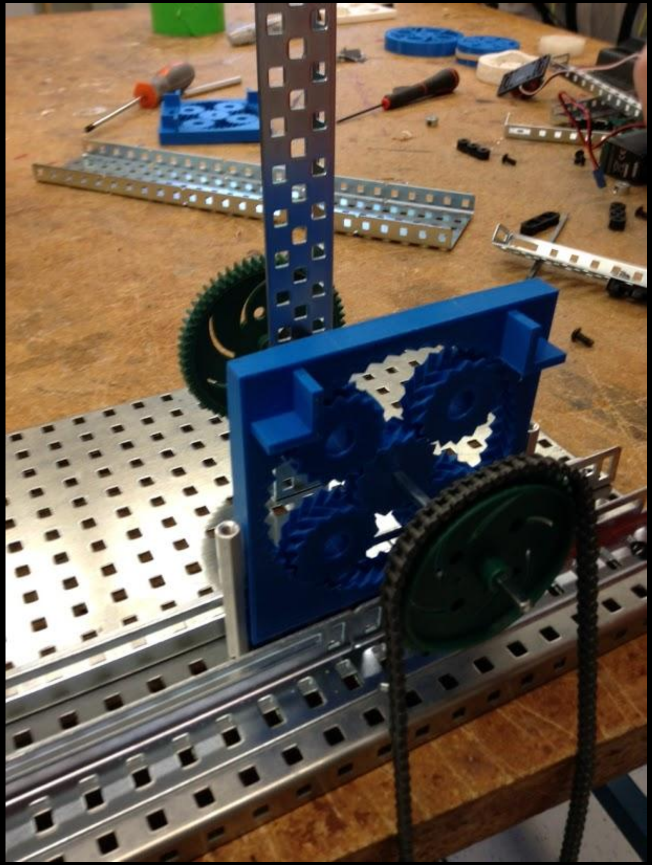
Trial 3 3.15g, 15.21g, 12.06g

Average sand needed = $\frac{37.41g}{3} = 12.47g \times \frac{1kg}{1000g}$

$\times 9.8\ \frac{m}{s^2} = 122.206\ N$

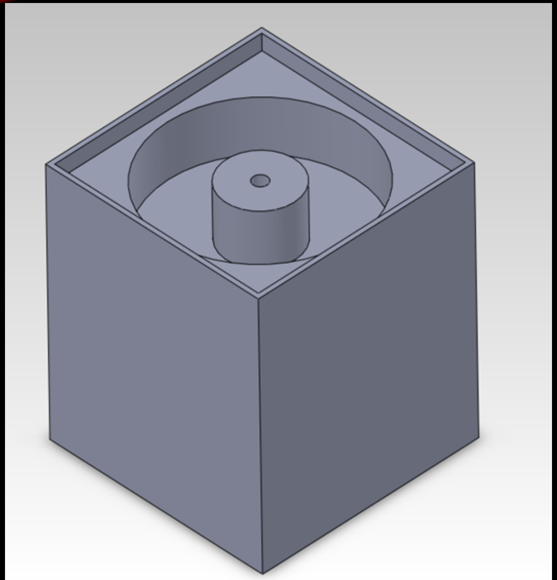
$\times 2.81\ cm\left(\frac{1m}{100\ cm}\right) = 0.003434\ N{\cdot}m$

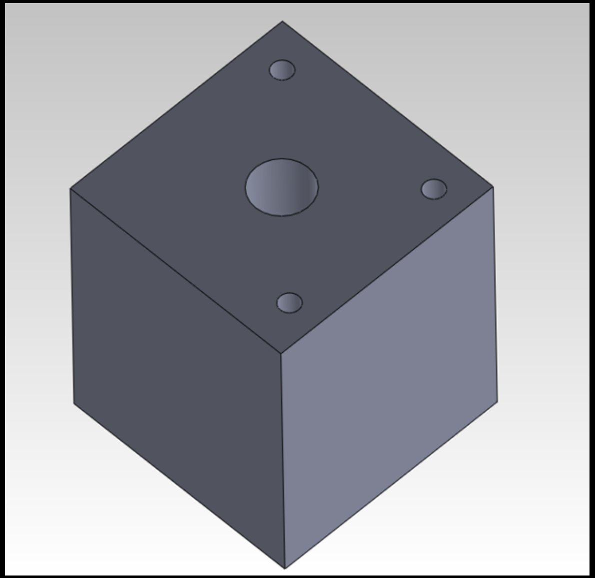




CASING 1

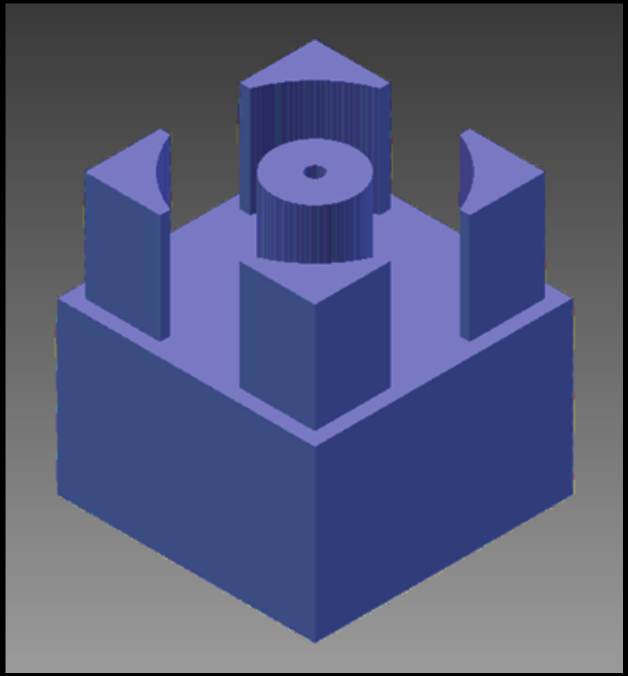
Initial design

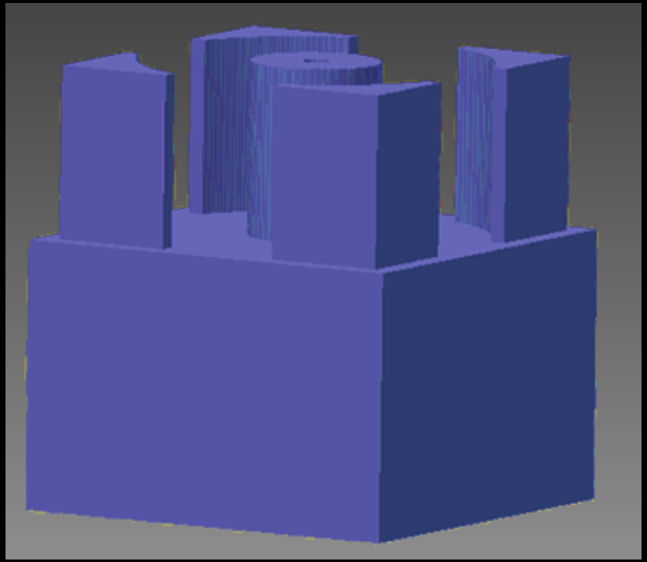




CASING 2

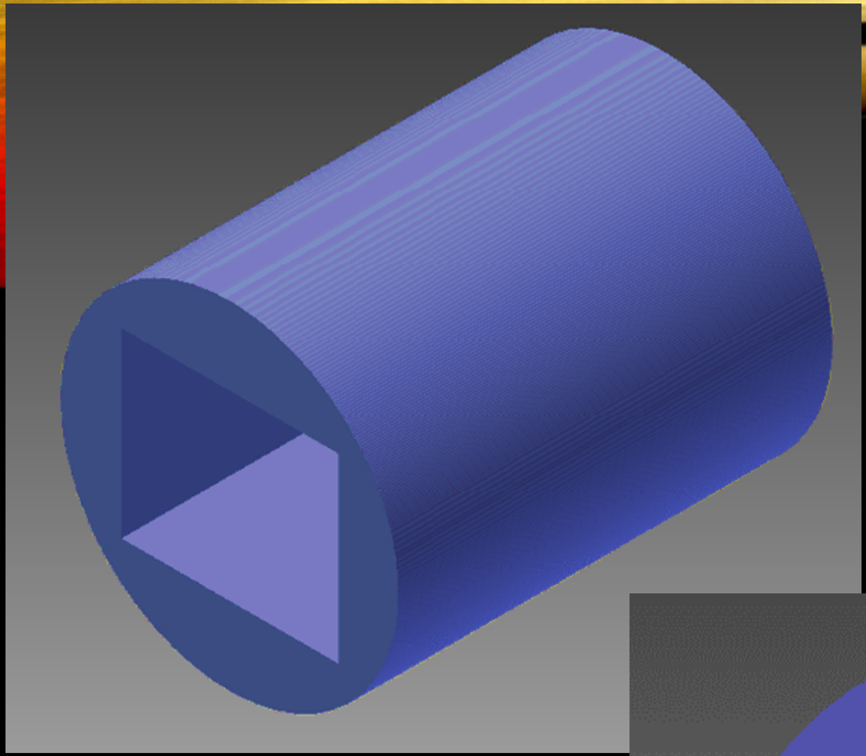
Final edit before the current version



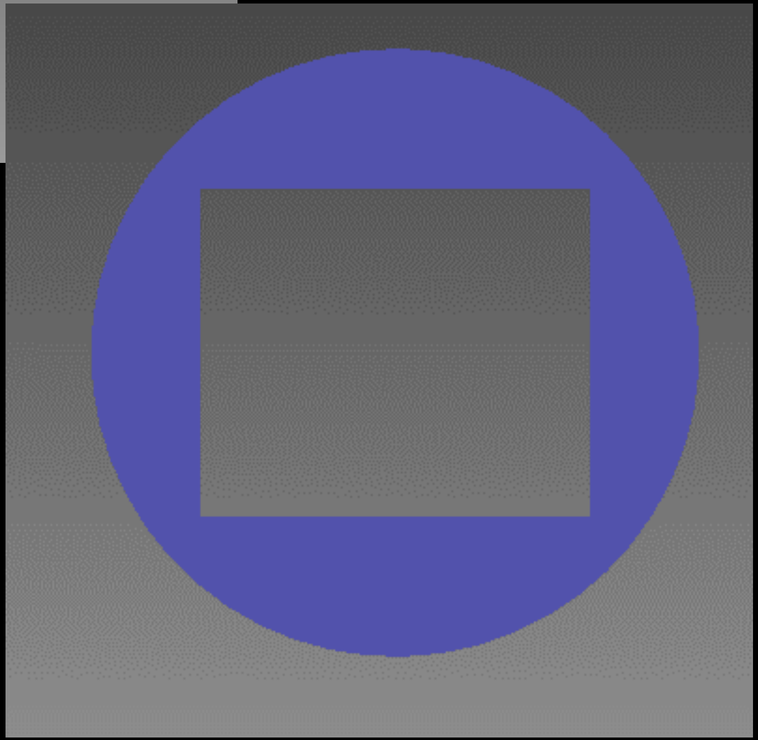


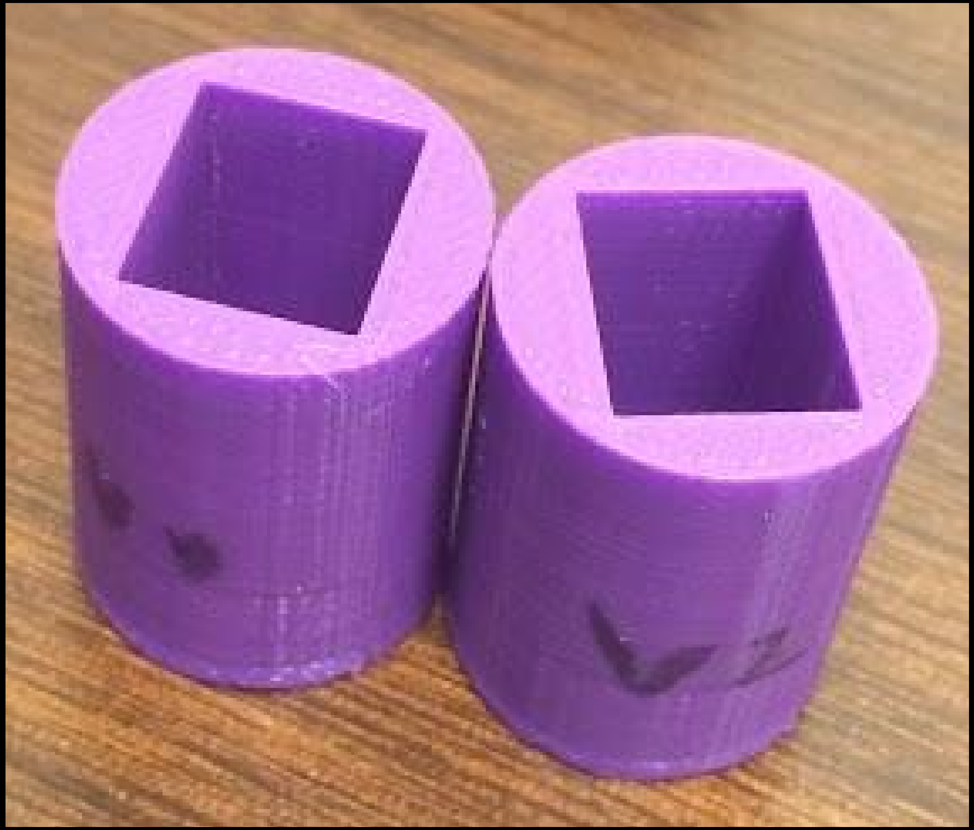
Note: there were smaller intermittent saves between these versions

PRELIMINARY SENSOR HOUSINGS



- Motor sleeve
  - 5 edits
  - Each edit--small dimension changes





PROTOTYPE

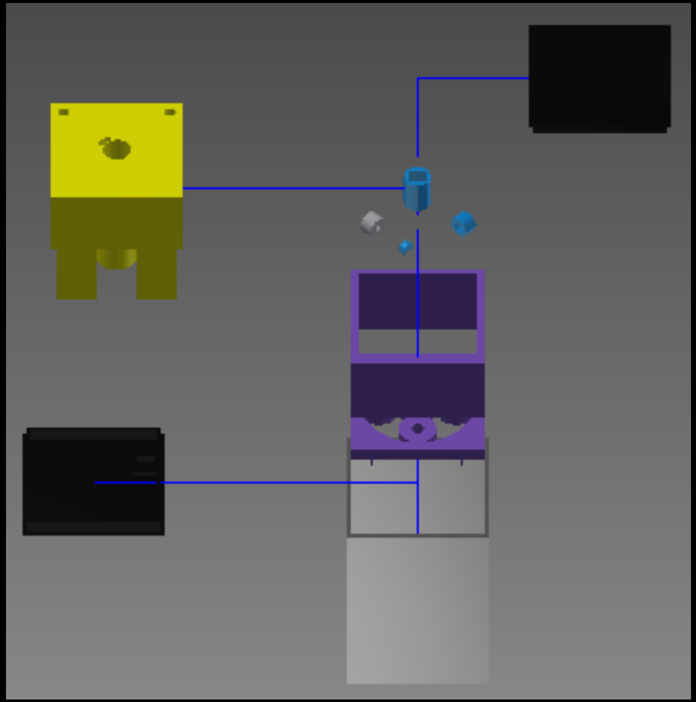
Prototype
- Overview from:
  - Hardware
  - Software
  - Biology

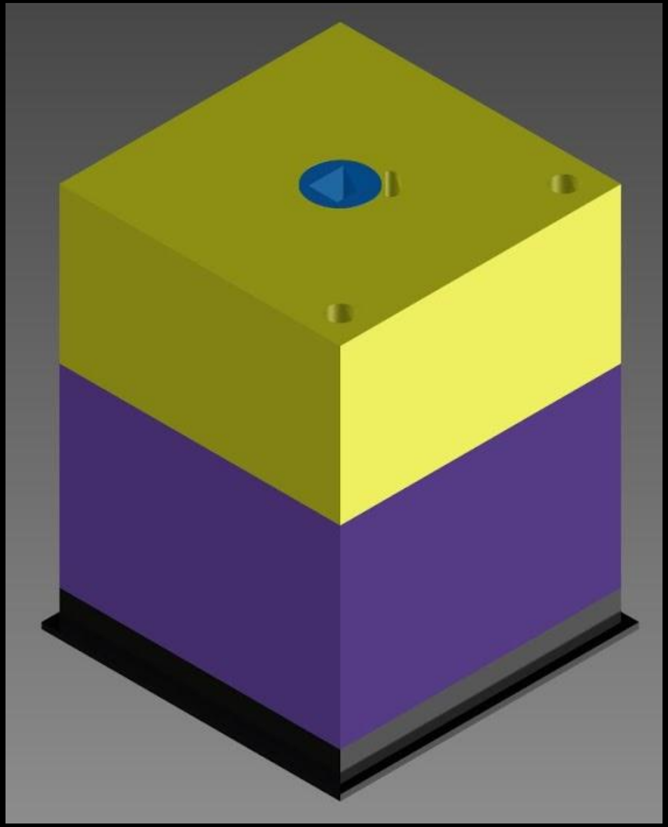
Materials list

CASING ASSEMBLY

Outside of Nanorack box

Parts assembly

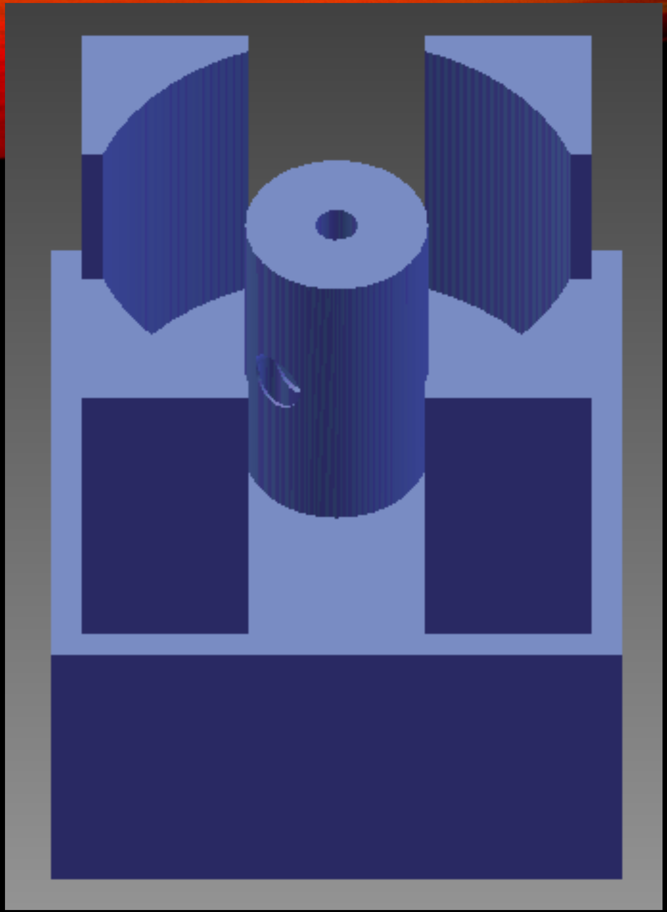




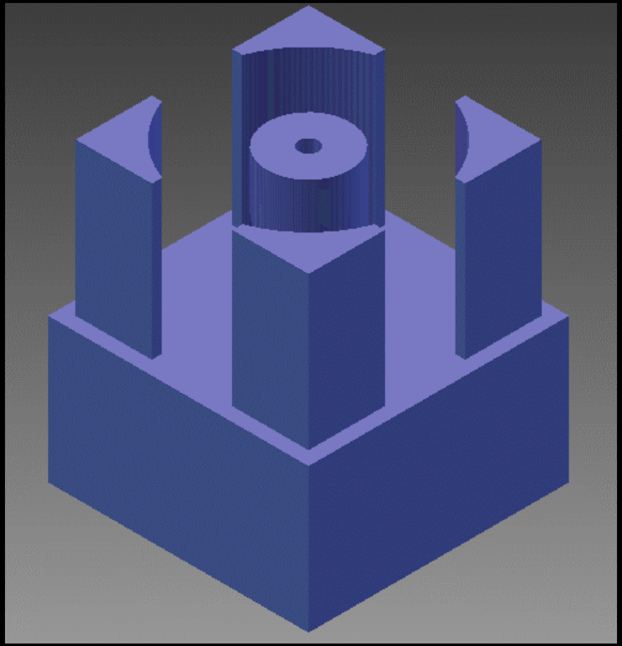
Interior components

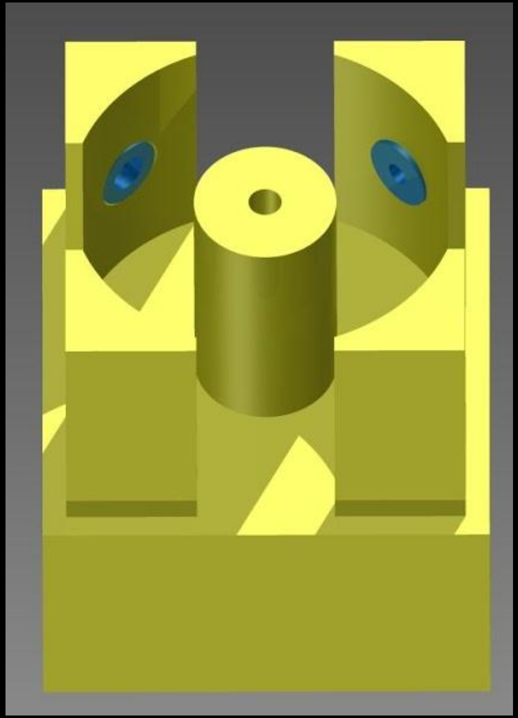
Inner casing detached from box

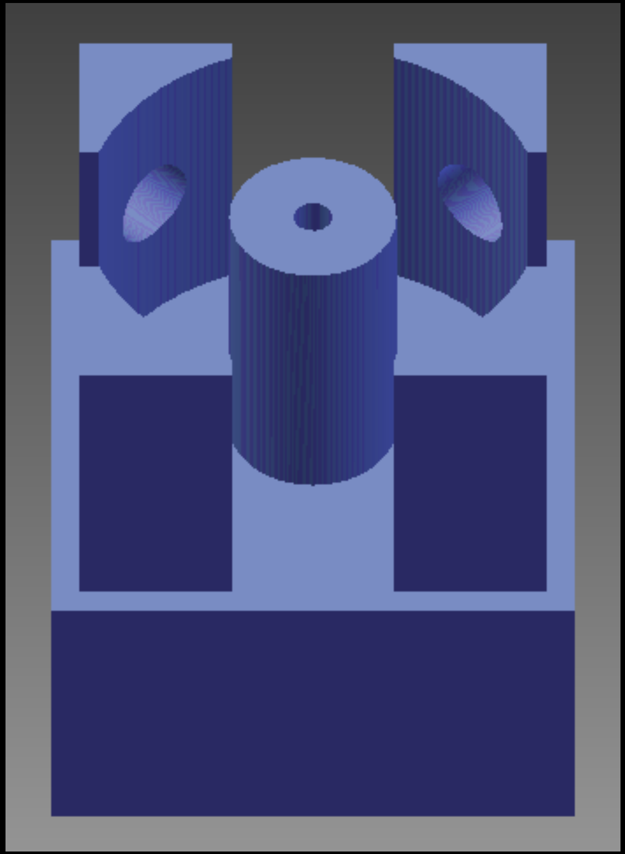
CURRENT INNER CASING



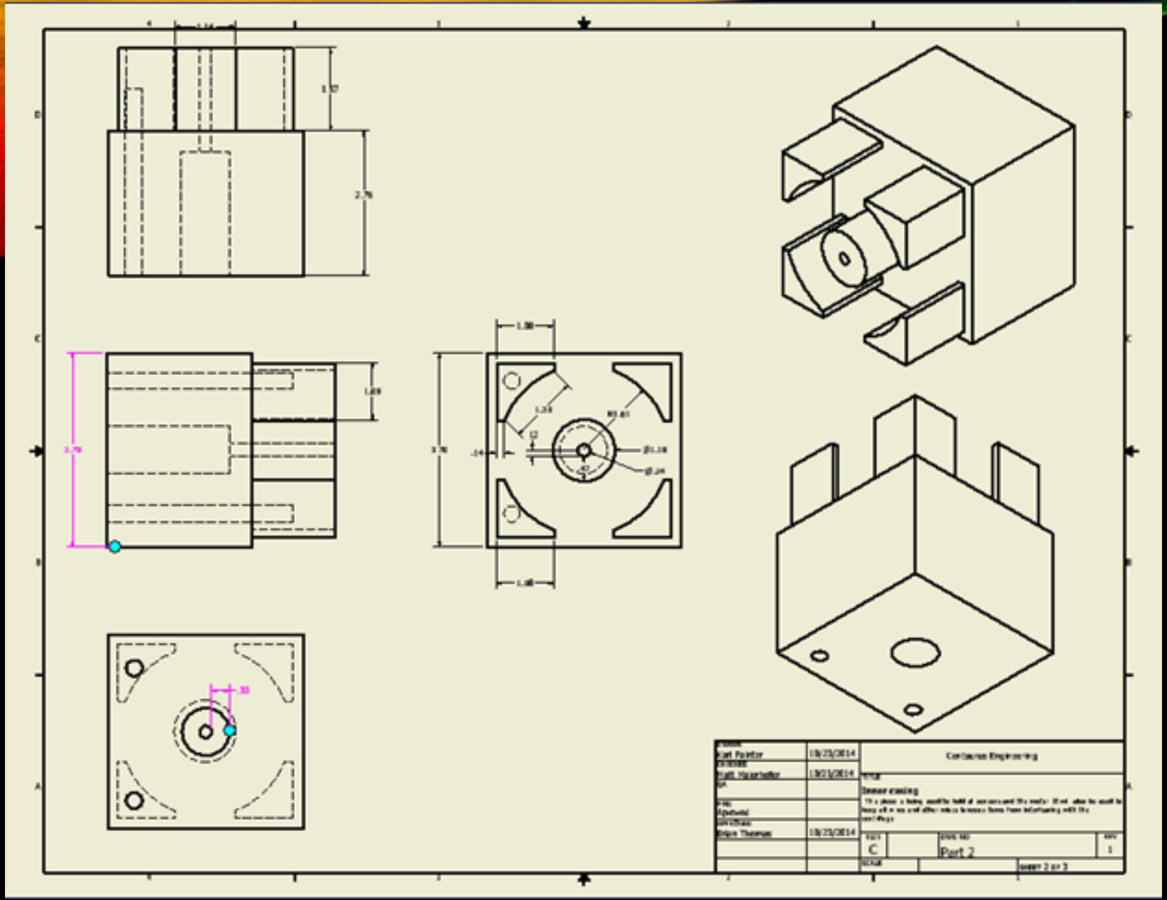
- Taller to fit vials
- Sensor housings

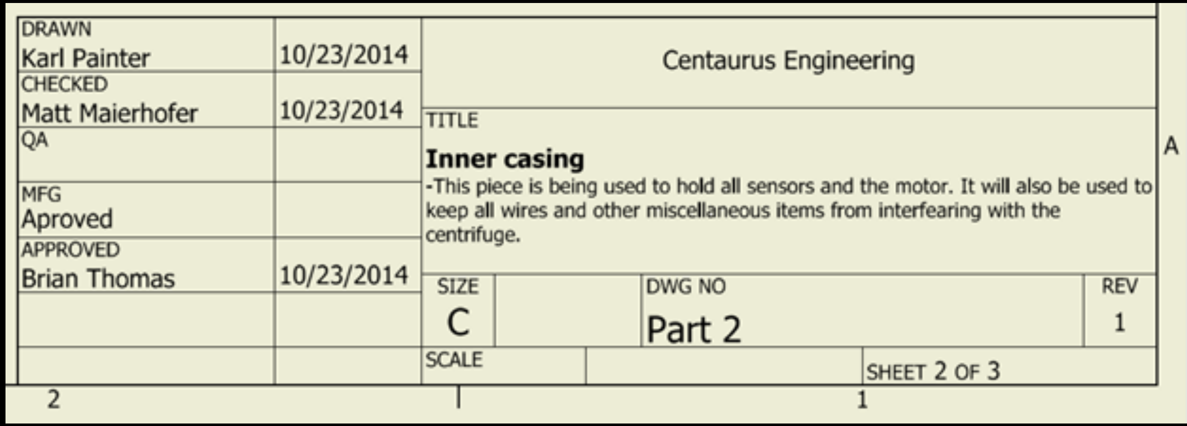




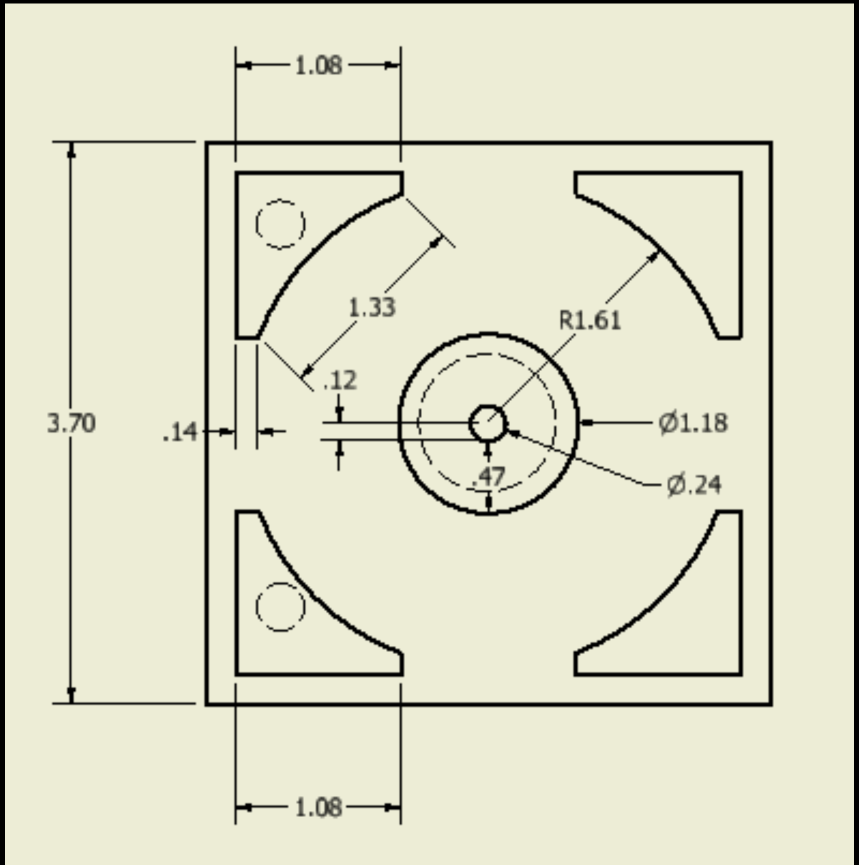


CURRENT INNER CASING



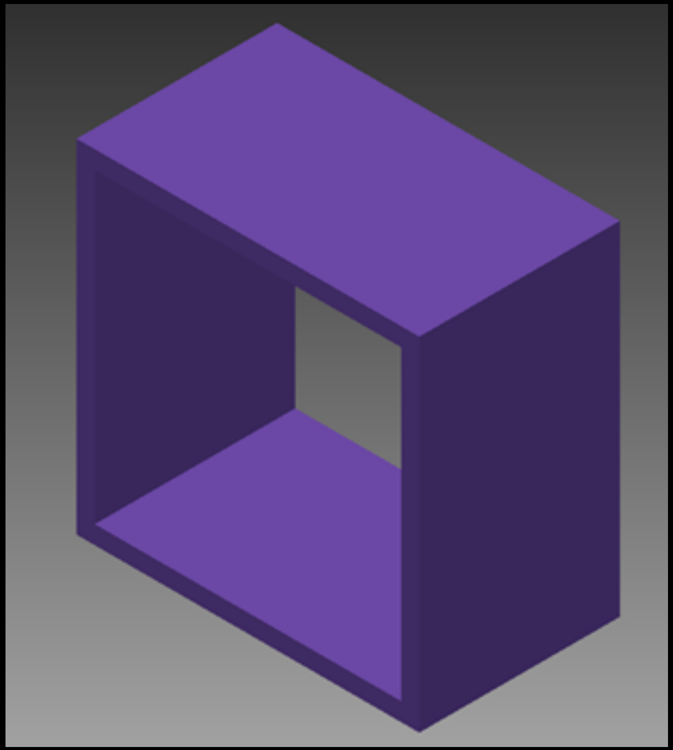


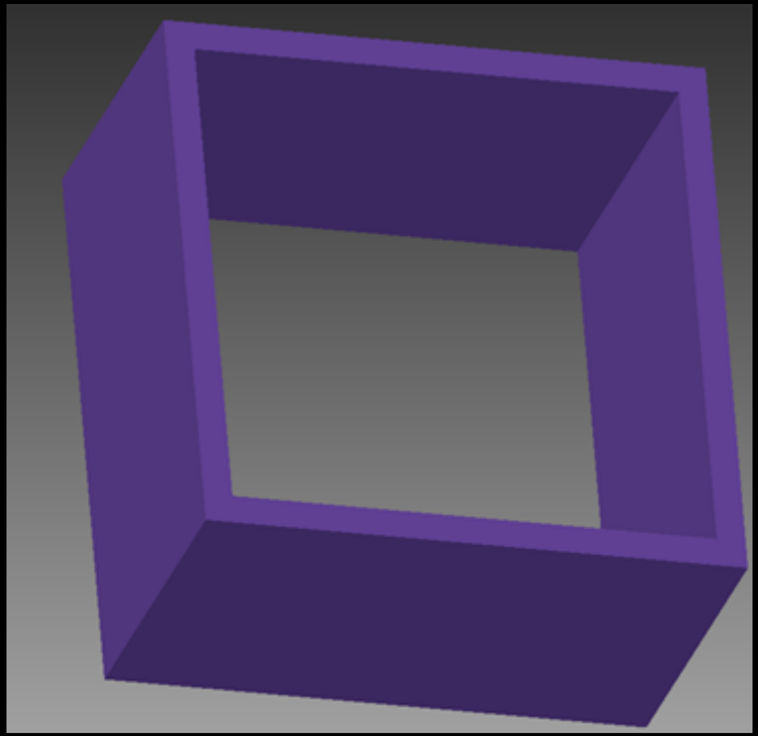
DRAWN Karl Painter	10/23/2014	Centaurus Engineering		
CHECKED Matt Maierhofer	10/23/2014	TITLE **Inner casing** -This piece is being used to hold all sensors and the motor. It will also be used to keep all wires and other miscellaneous items from interfearing with the centrifuge.		
QA				
MFG Aproved				
APPROVED Brian Thomas	10/23/2014			
		SIZE C	DWG NO Part 2	REV 1
		SCALE		SHEET 2 OF 3



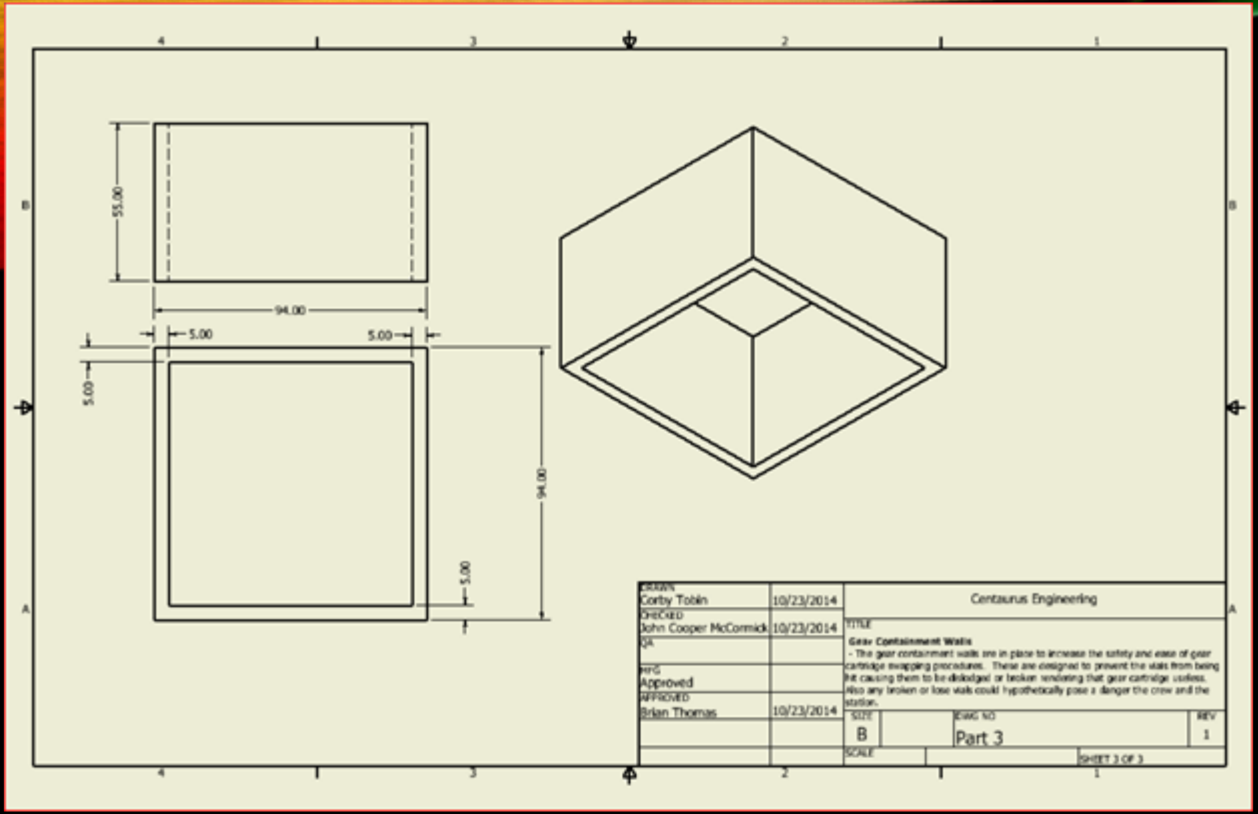
Measurement units: in

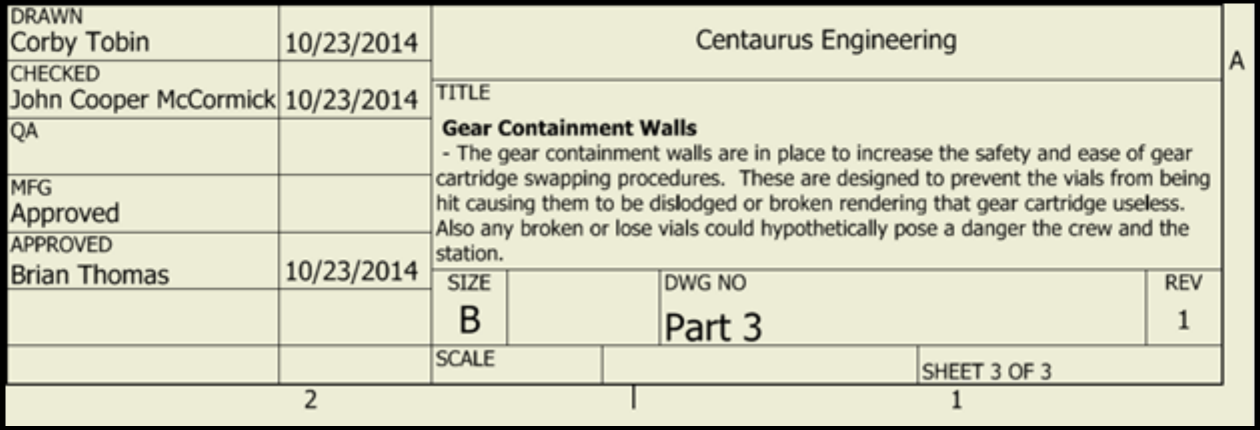
CURRENT CONTAINMENT WALL

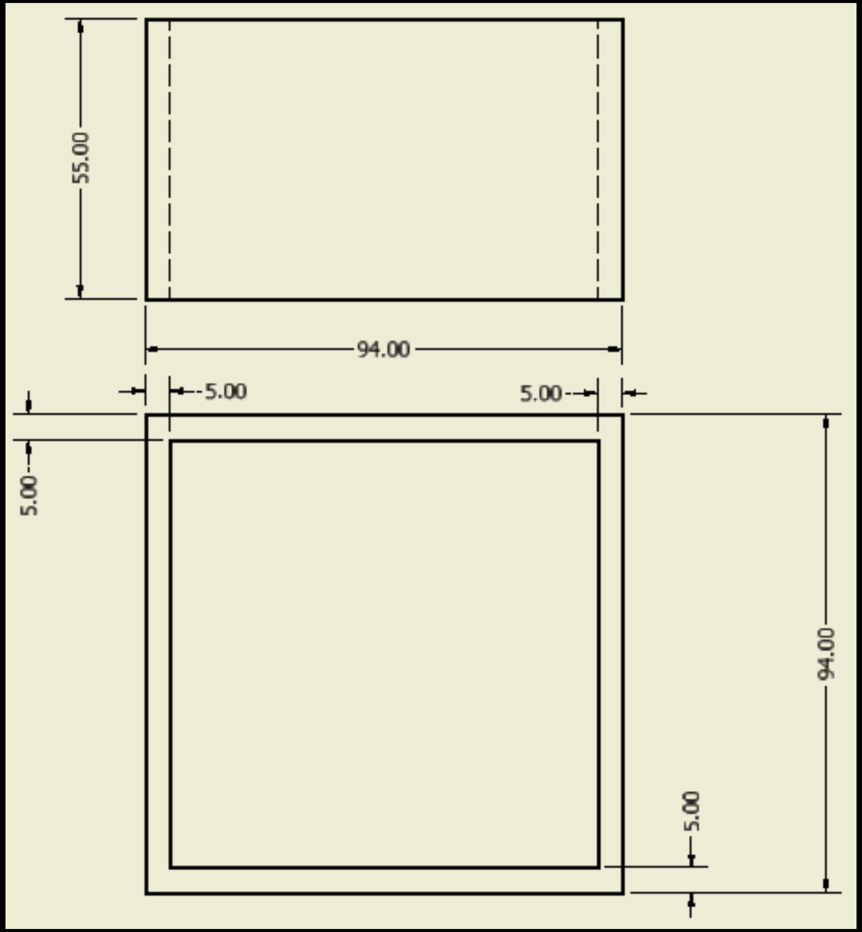




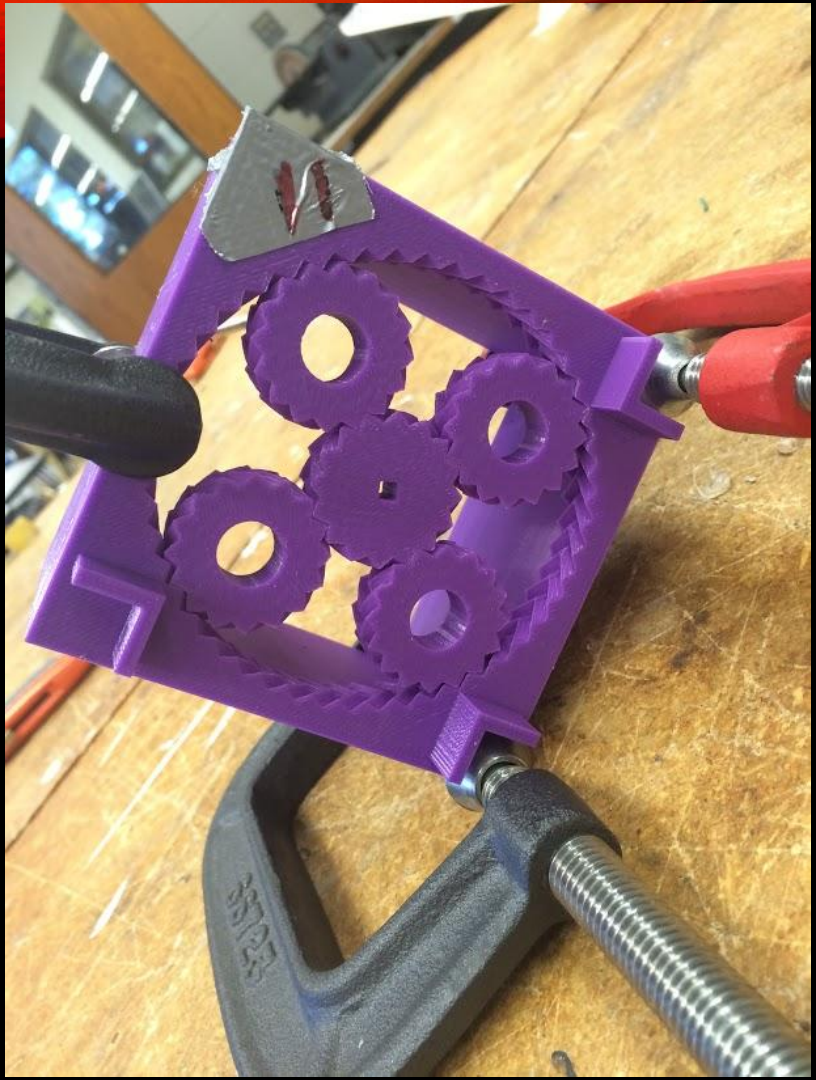
CURRENT CONTAINMENT WALL



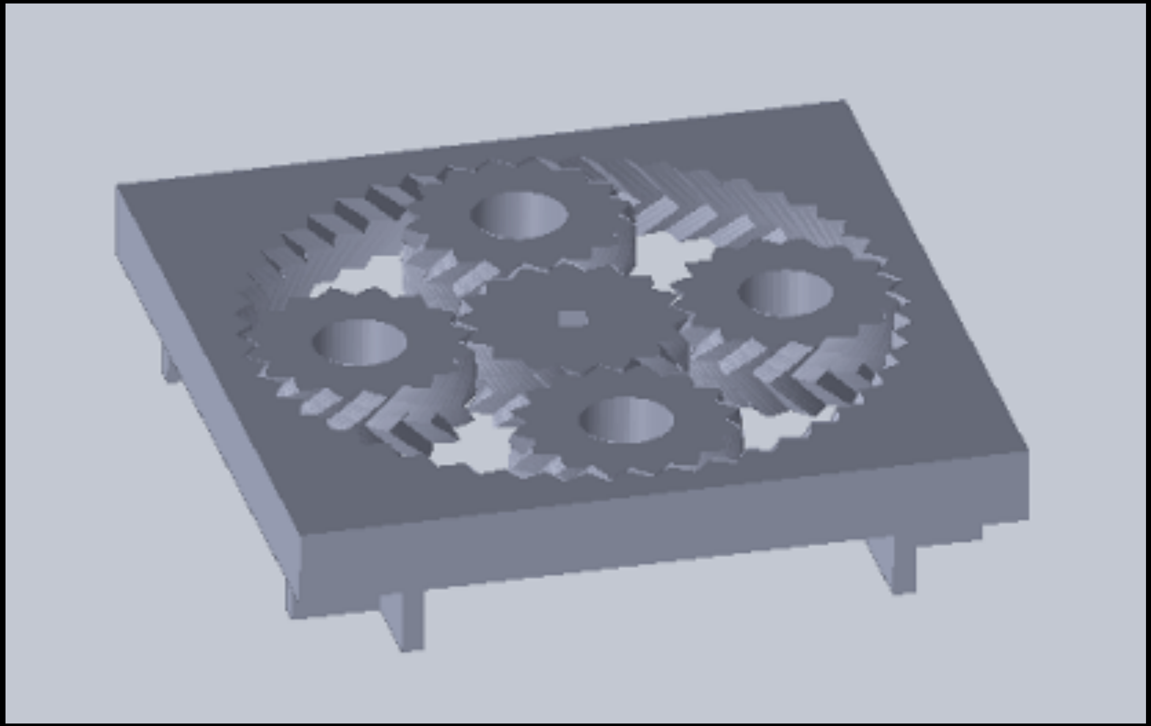




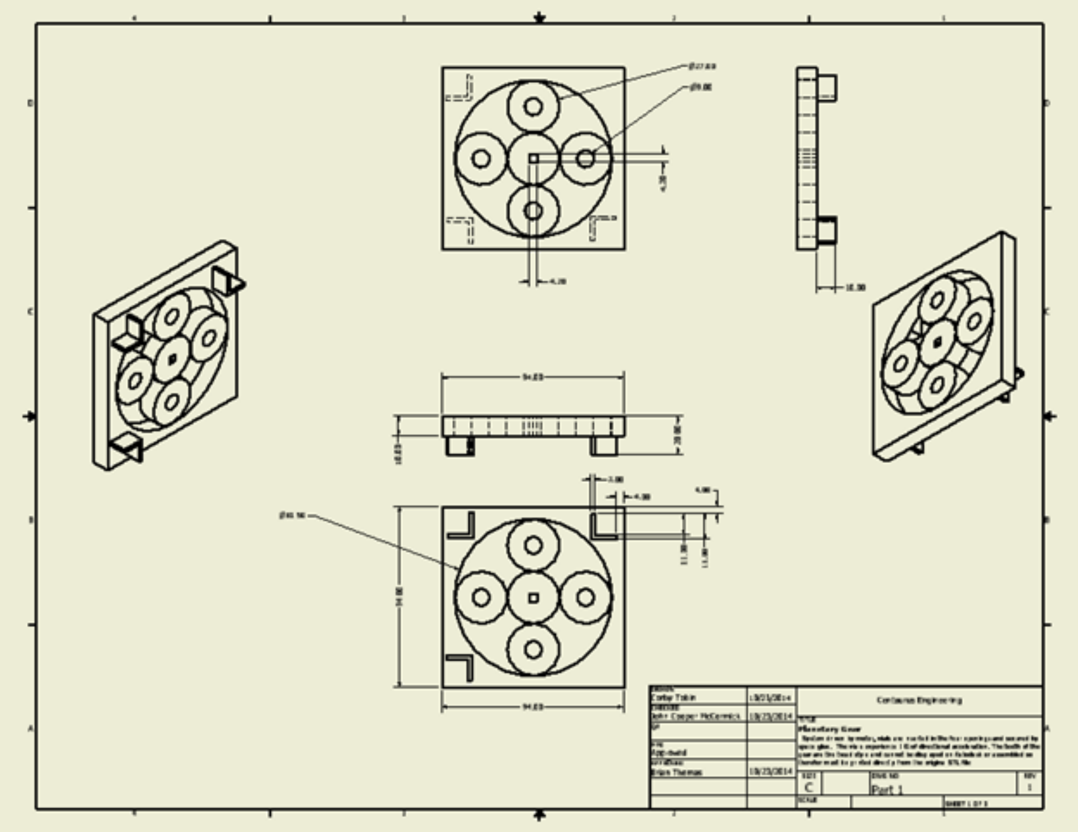
Measurement units: mm

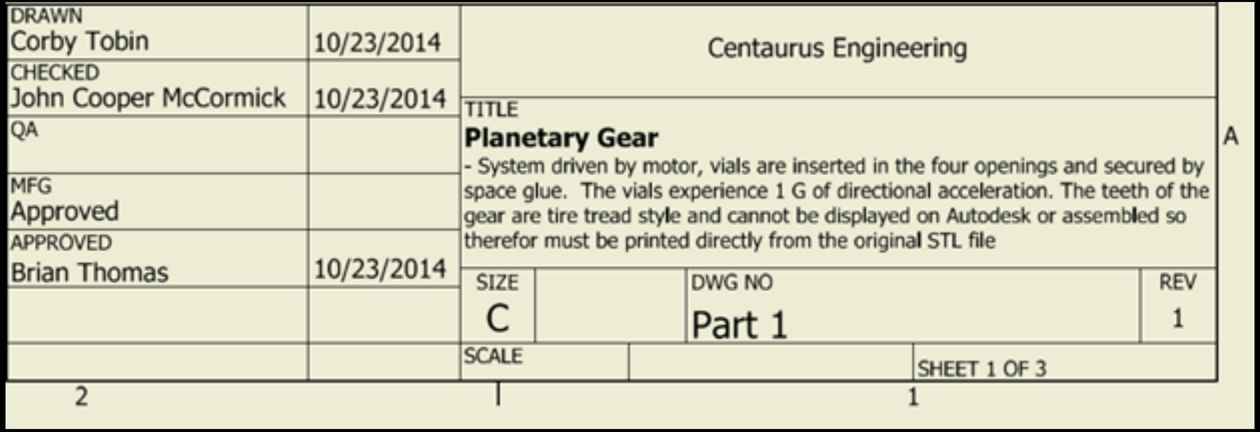


CURRENT PLANETARY GEAR SYSTEM



CURRENT PLANETARY GEAR SYSTEM

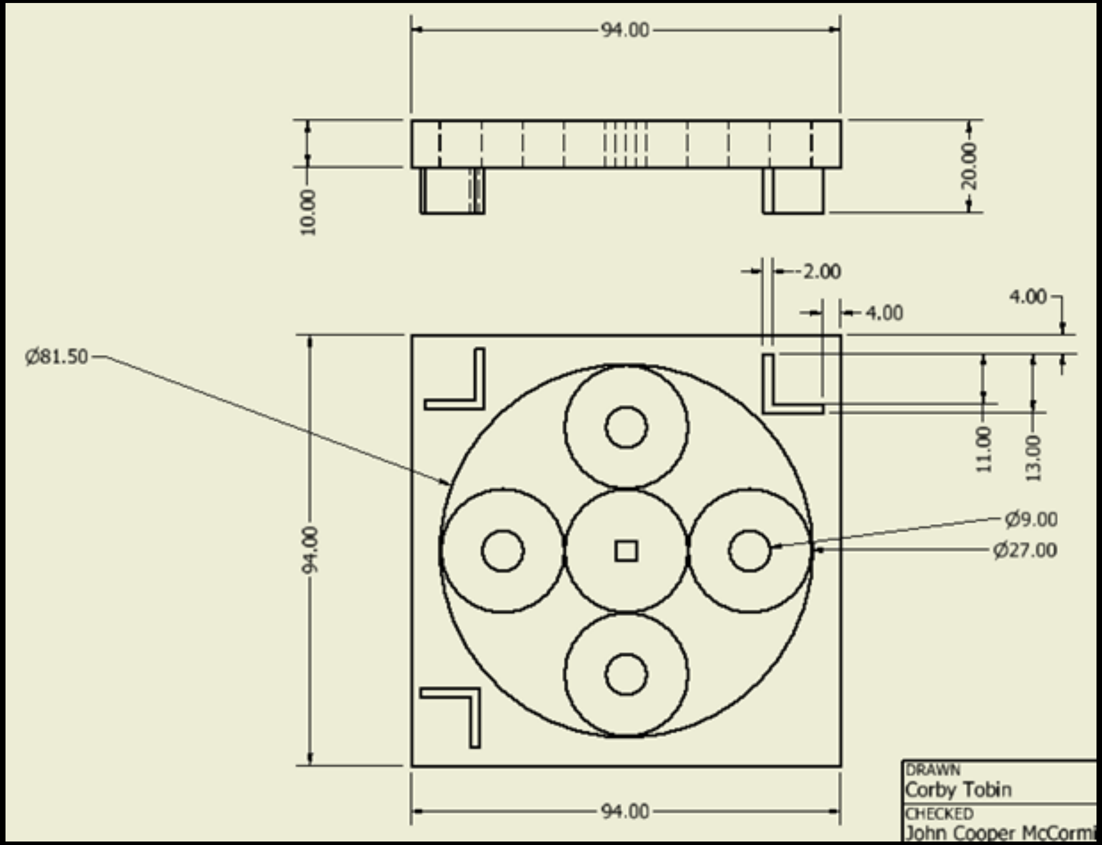




DRAWN Corby Tobin	10/23/2014	Centaurus Engineering			
CHECKED John Cooper McCormick	10/23/2014				
QA		TITLE **Planetary Gear** - System driven by motor, vials are inserted in the four openings and secured by space glue. The vials experience 1 G of directional acceleration. The teeth of the gear are tire tread style and cannot be displayed on Autodesk or assembled so therefor must be printed directly from the original STL file			
MFG Approved					
APPROVED Brian Thomas	10/23/2014				
		SIZE C	DWG NO Part 1		REV 1
		SCALE		SHEET 1 OF 3	

A

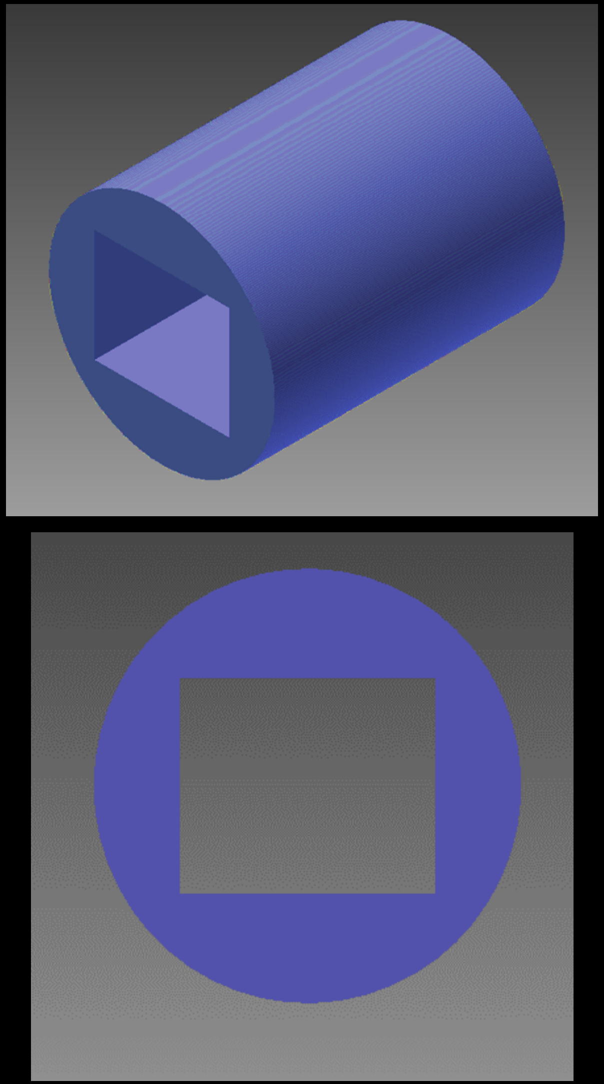
2 1



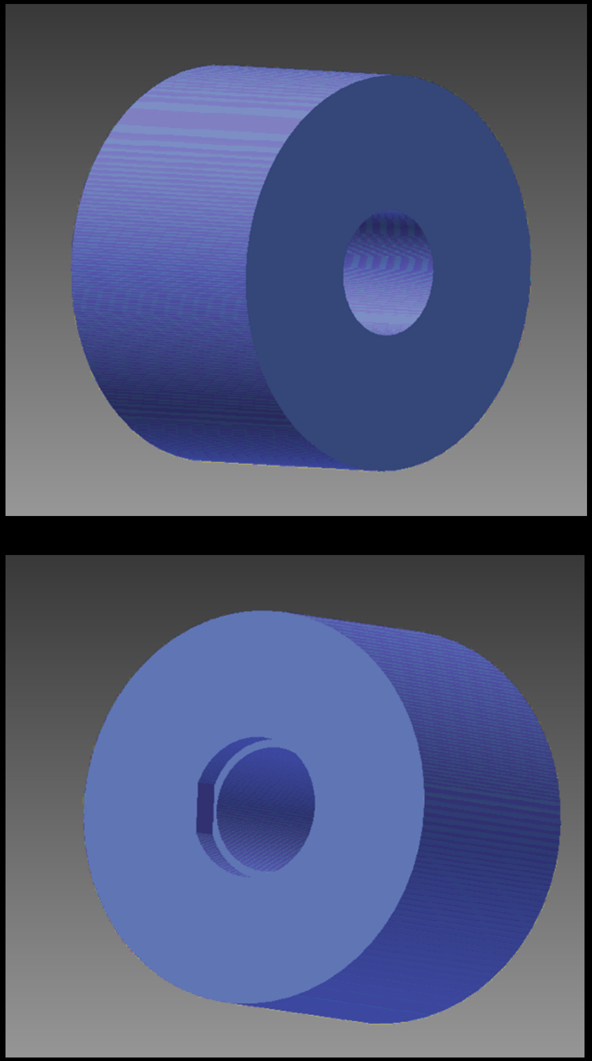
Measurement units: mm

SENSOR HOUSINGS

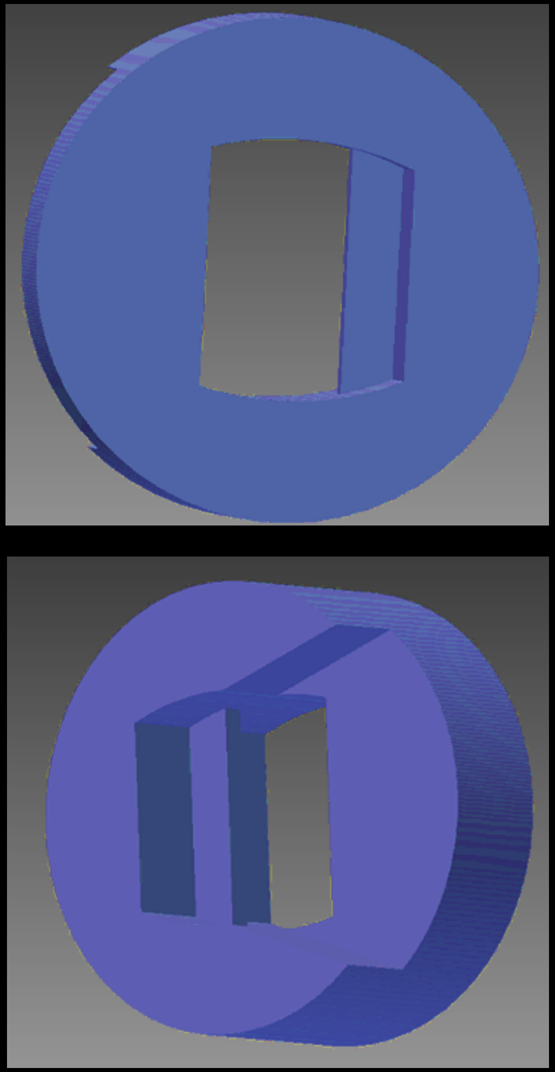
Motor sleeve



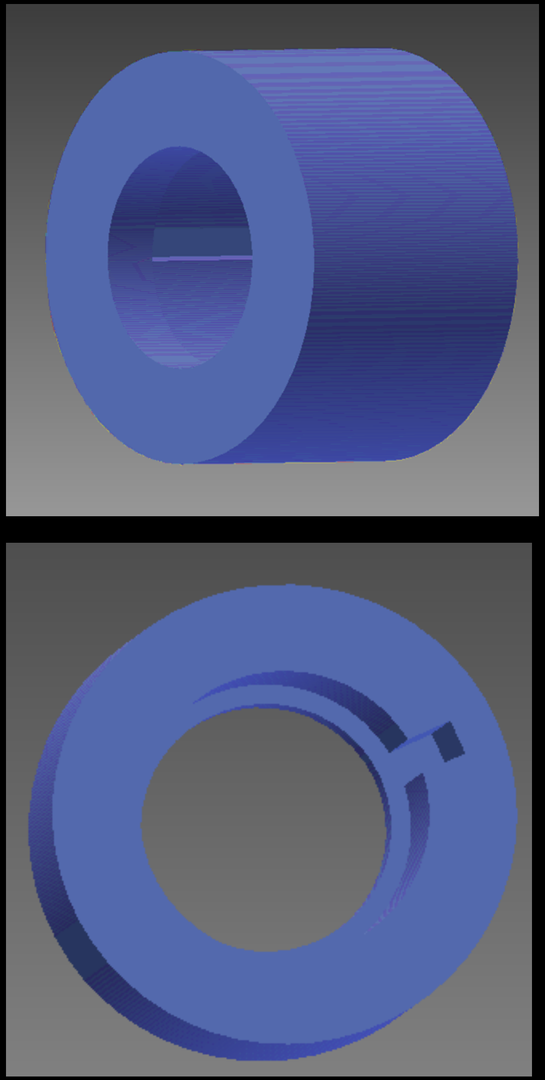
LED



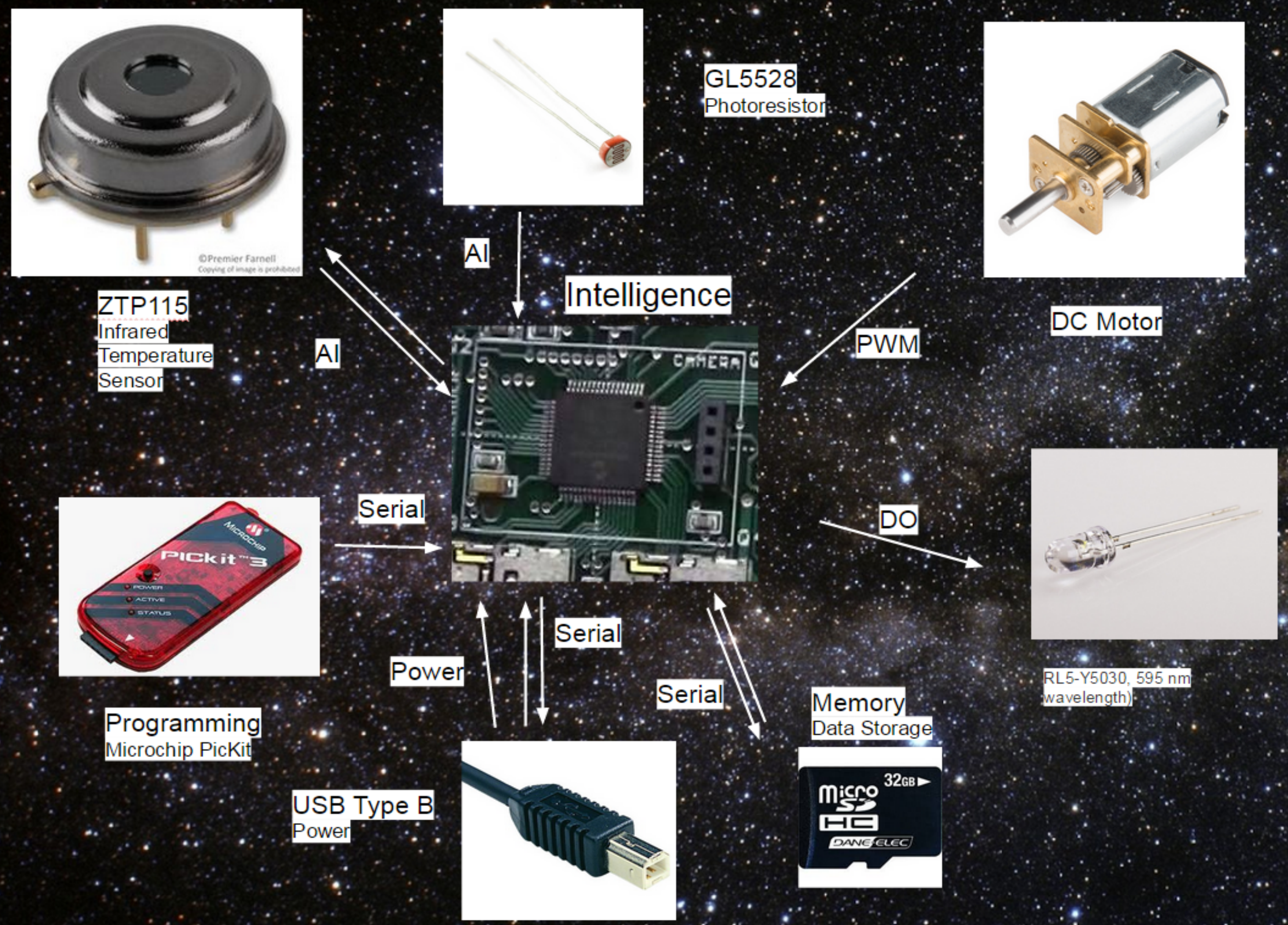
Photoresistor



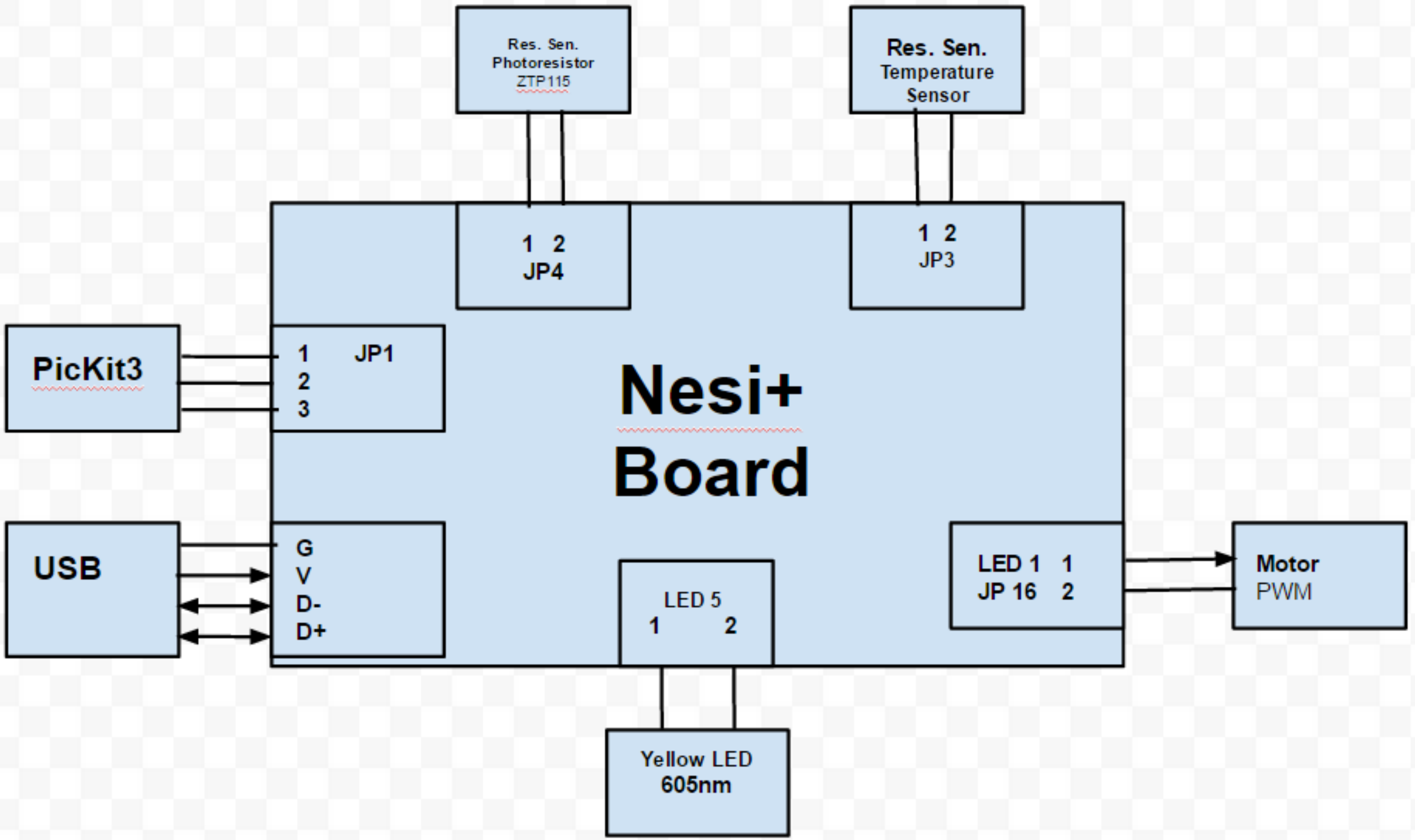
Temperature sensor



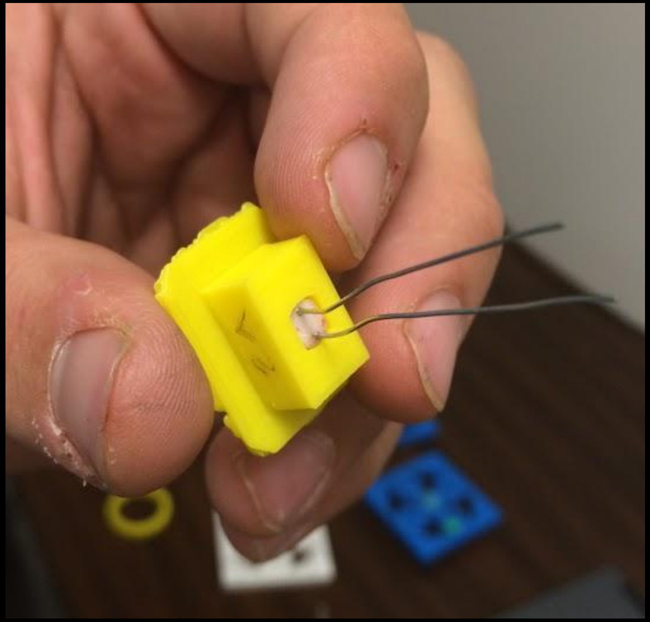
CONCEPTUAL BLOCK DIAGRAM



FUNCTIONAL BLOCK DIAGRAM

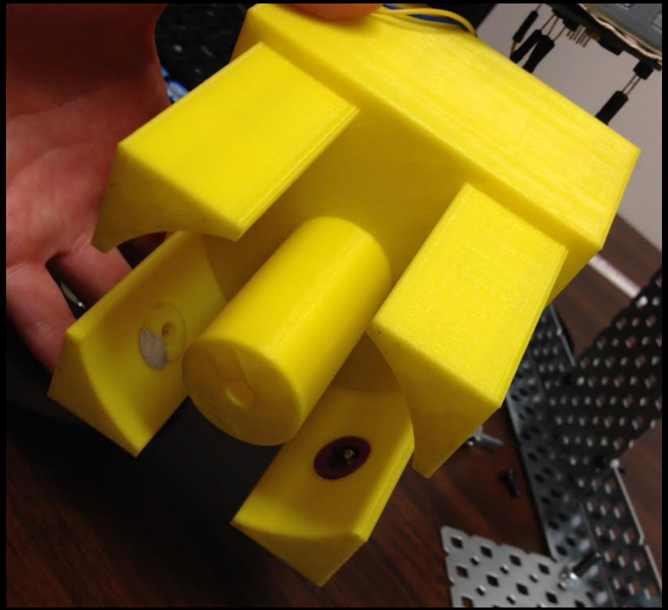


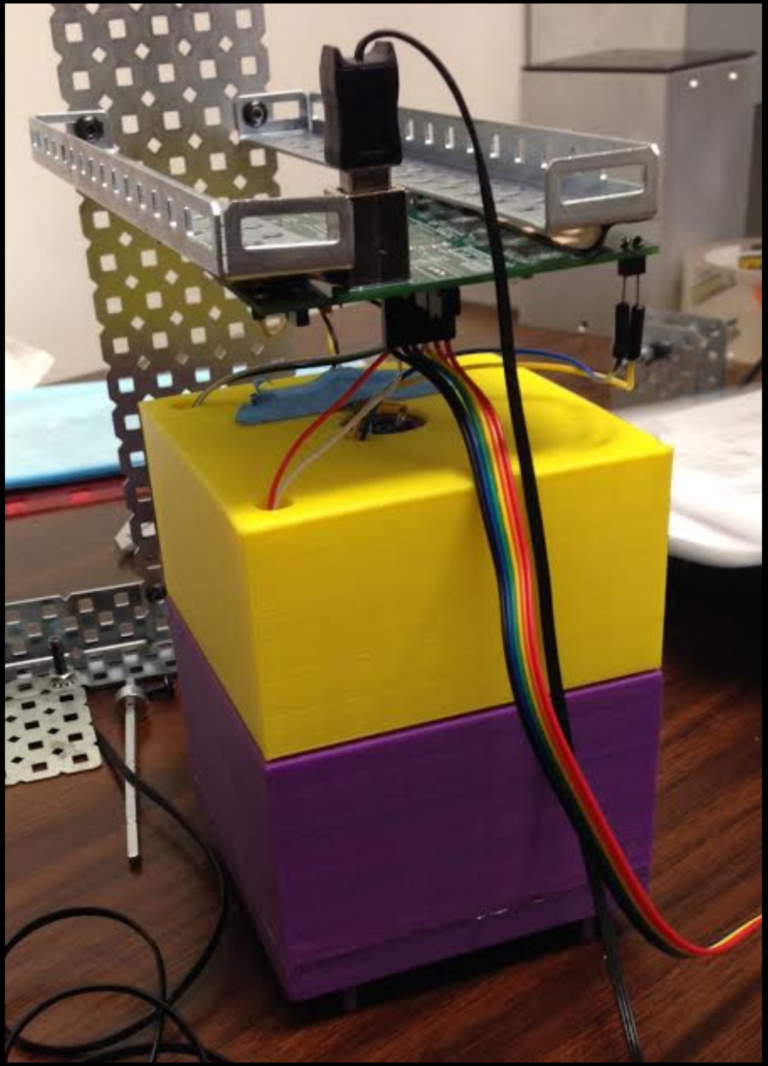
SENSORS AND MOTOR



- LED
- Photoresistor
- IR temp
- Motor

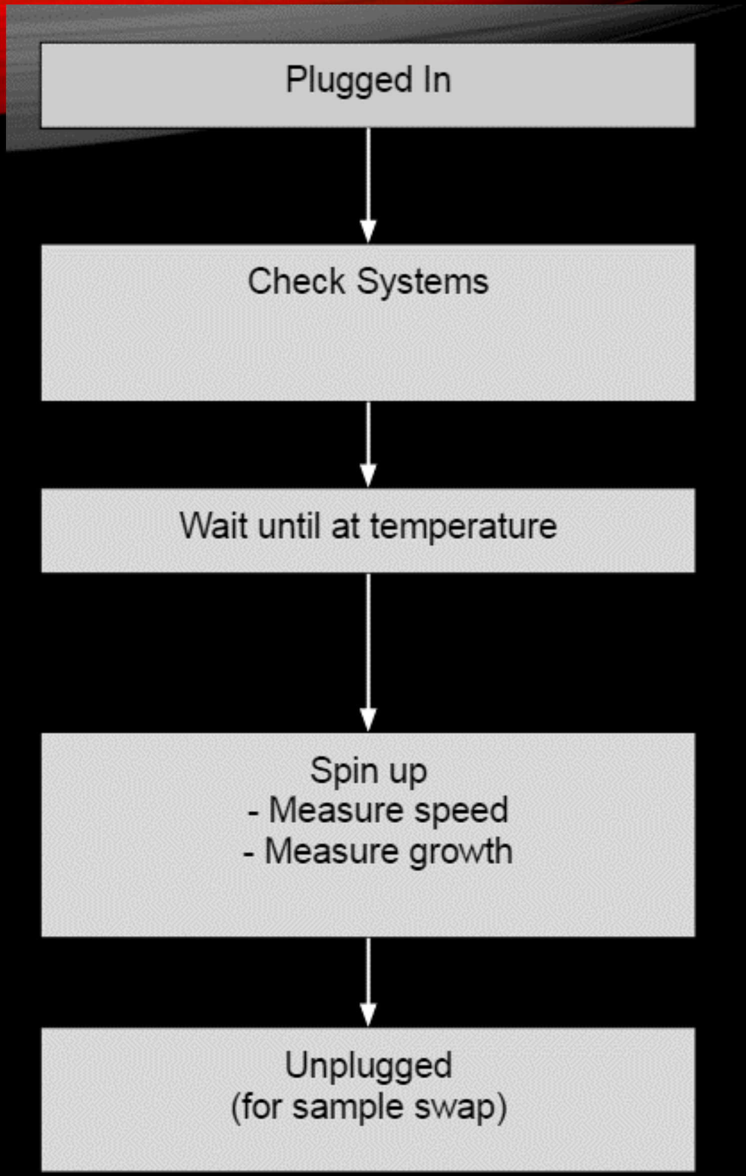
HOW IT FITS TOGETHER





PROTOTYPE CODE

Master flowchart



<u>List of Constants:</u>

1. SUPERLIGHT_VALUE = Value for refracted light on the photoresistor
2. LIGHT_VALUE = Value for an unobstructed photoresistor
3. DARK_VALUE = Value for an obstructed photoresistor
4. GROWTH_TEMP = Value of the IR Thermal Resistor at 4C
5. IDEAL_MSPR = Ideal Milliseconds per Vial
6. SLUSH_MSPR = Slush Zone for above value
7. debug = 1 or 0, set to 1 by pressing button on boot

<u>List of Abbreviations:</u>

1. DC = Dutycycle of the motor (ledB)
2. PR = Value of the photoresistor

<u>Utility functions:</u>

```
void logToScreen(String str, int i) {

  if (debug) {

     usb.print(str);
     usb.print("\r\n");

  } else if (i > 1) {

     dataLog.add(str, i);

  }

}
```

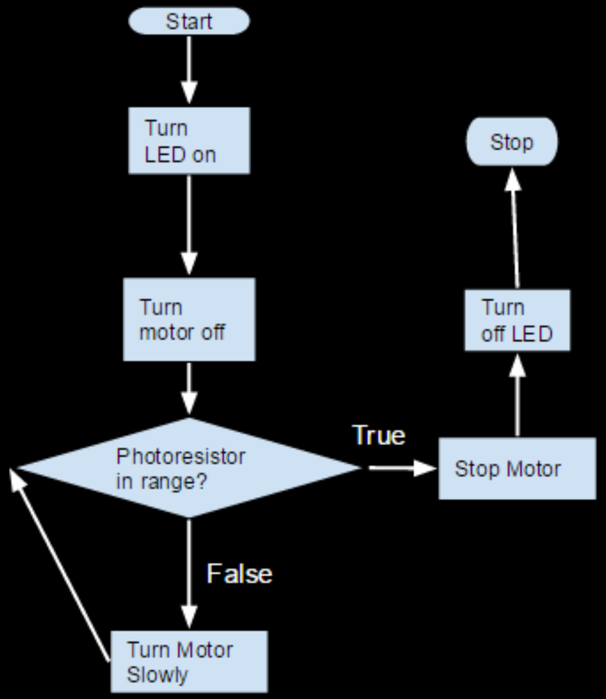
All code viewable at: https://github.com/ISSCentaurus/MGE

CHECK SYSTEMS

Pseudo code

1. Start
2. Code turns LED on
3. Motor turns off
4. Checks for photoresistor output
    a. If any (true), stop code
5. If none (false), turn motor slowly
    a. If photoresistor NOT receiving light, loop back to #5
6. If light received, stop motor
7. Turn off LED
8. End code sequence

Flowchart



Code

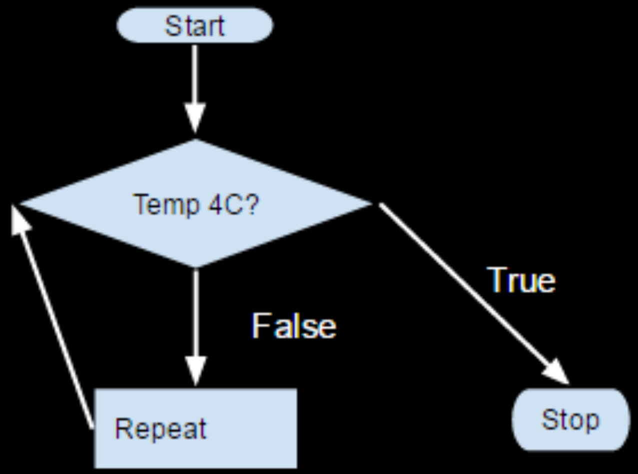
```
int checkSystems() {
   ledB.dutycycle(0);
   ledR.dutycycle(100);

   //check photoresistor
   if (resistiveSensors.readQ2() >= LIGHT_VALUE && resistiveSensors.readQ2() <= SUPERLIGHT_VALUE) {
       //Turn motor slowly to obstruct the photoresistor
       while (1) {
           ledB.dutycycle(100);
           wait(.1);
           ledB.dutycycle(0);
           if (resistiveSensors.readQ2() > DARK_VALUE && resistiveSensors.readQ2() < SUPERLIGHT_VALUE) {

               logToScreen("Had to spin motor to find vial", 0);
               return 1;

           }
       }
   }


   return 1;
}
```

WAIT UNTIL AT TEMPERATURE

Pseudo code

1. Start
2. Read Temperature - Is it to 4C?
   a. If not, (false) repeat 2.
3. Return

Flowchart



Code

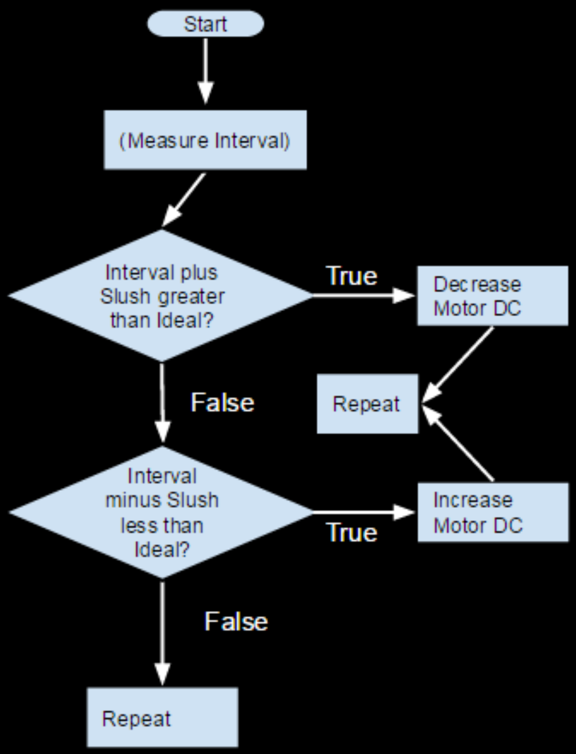
```
int waitUntilTemp() {
   while (1) {

       if (resistiveSensors.getQ1(10, 5) >= GROWTH_TEMP) {
           logToScreen(sprintf("Temperature checked, was %d", resistiveSensors.readQ1()), 0);
           return 1;
       }

       logToScreen("Waiting to reach temp", 0);

   }

}
```

SPIN UP/MAINTAIN SPEED

Pseudo code

1. Start
2. Start "Measure Interval" function
3. Is interval greater than ideal?
   (+ a 'slush zone')
   a. If true, increase motor DC
   b. Return
4. Is interval less than ideal - slush?
   a. If true, decrease motor DC
   b. Return
5. Loop back to step two.

Flowchart



Code

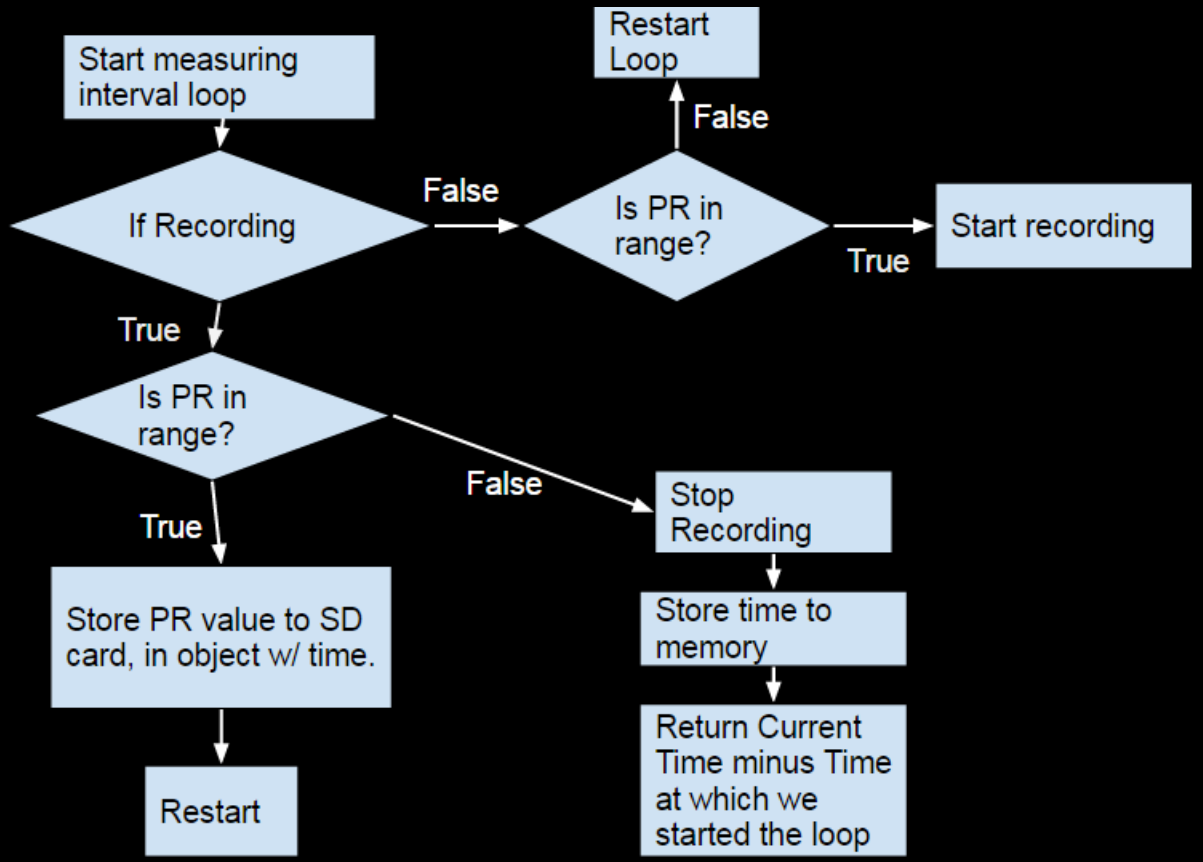
```
int maintainSpeed() {

   //Speed up - Motor too slow.
   if (interval() < (IDEAL_MSPR - SLUSH_MSPR)) {
       dutycycle = dutycycle + 1;
       ledB.dutycycle(dutycycle);
       logToScreen("Maintaining speed - was too slow", 0);
       return 0;
   }
   //Slow down - Motor too fast.
   if (interval() > (IDEAL_MSPR + SLUSH_MSPR)) {
       dutycycle = dutycycle - 1;
       ledB.dutycycle(dutycycle);
       logToScreen("Maintaining speed - was too fast", 0);
       return 0;
   } else {
       logToScreen("Maintaining speed.", 0);
       return 1;
   }
}
```

MEASURE INTERVAL-PART 1

Pseudo code

1. Start
2. Are we recording?
   a) Not Recording:
      I. Is PR > Refracted or < Dark?
         i. True: Start recording! Restart loop.
         ii. False: Restart loop.
   b) Recording:
      I. Is PR > Refracted or < Dark?
         i. True: Keep recording, store data. Restart loop.
         ii. False: Stop recording, return interval.

Flowchart



MEASURE INTERVAL-PART 2

Code

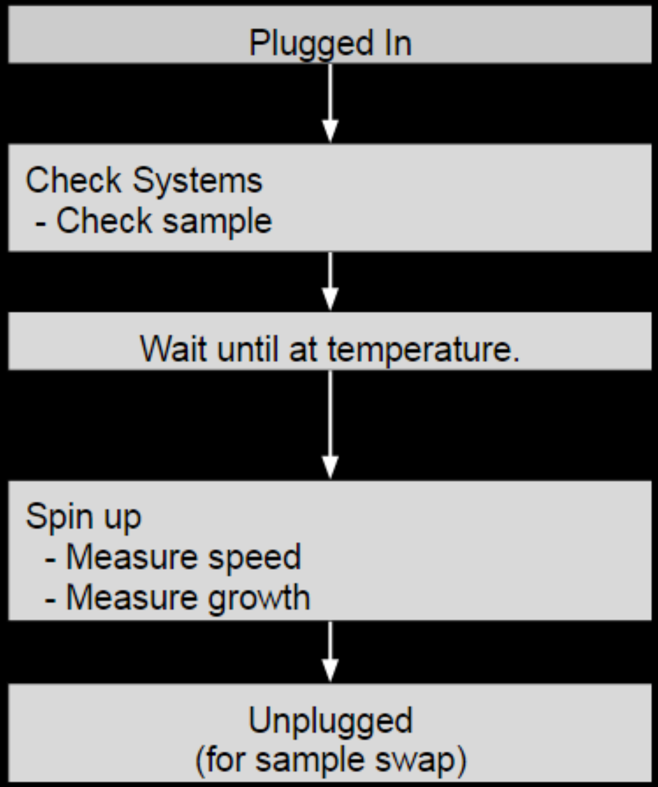
```
int interval(void) {
   DateAndTime past = dateTime.get();
   int recording = 0;

   while (1) {
       if (recording) {

           if (resistiveSensors.readQ2() > DARK_VALUE || resistiveSensors.readQ2() > SUPERLIGHT_VALUE) //Is there anything in the way?
           { //Yes - something in the way.

               logToScreen(sprintf("Light: %d at %s", resistiveSensors.readQ2(), dateTime.getStamp()), 3); //Log Light Level

           } else { //No - nothing in the way

               return (dateTime.sub(dateTime.get(), past).second)/1000; //Return the interval

           }

       } else { //Not recording

           if (resistiveSensors.readQ2() > DARK_VALUE || resistiveSensors.readQ2() > SUPERLIGHT_VALUE) { //Is there anything in the way?

               recording = 1;
               past = dateTime.get();

           }

       }

   }

}
```

MAIN CODE

Pseudo code

1. Check Systems
2. Wait for Temperature
3. Spin - Loop around spin until unplugged.

Flowchart



Code

```
checkSystems();

waitUntilTemp();

while(1){
maintainSpeed();
}
```

STERILIZATION

- Vials
  - UV
- Broth
  - Autoclave
  - Pressure cooker
- Lab space

SCRIPTS + TEST CODE

SerialLogger.py

14 lines (13 sloc) | 0.378 kb

```python
import serial
import csv

ser = serial.Serial()
port = input('COM Port [10]: ')  or 10
destination = input('Output File [./data.csv]: ') or './data.csv'
ser.port = int(port)
ser.open()
with open(destination, 'w', encoding='utf8') as f:
    writer = csv.writer(f)
    while True:
        str = ser.readline().decode("utf-8")
        print(str)
        writer.writerow([str,])
```

Collect and record data

SerialLogger.ino

33 lines (26 sloc) | 0.958 kb

```c
/*
  SerialLogger
  Turn an LED on, and log the value of a photoresistor every 2 milliseconds.

  This example code is in the public domain.
*/

const int analogInPin = A0;  // Analog input pin that the photoresistor is connected to
const int led = 13;    // Blue wire is positive

int sensorValue = 0;     // Value read from the photoresistor

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
  Serial.begin(9600);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the voltage level)
  // read the analog in value:
  sensorValue = analogRead(analogInPin);

  // print the results to the serial monitor:
  Serial.println(sensorValue);

  // wait 2 milliseconds before the next loop
  // for the analog-to-digital converter to settle
  // after the last reading:
  delay(2);
}
```

Collect and record data

Arduino Code

usbDebug.c

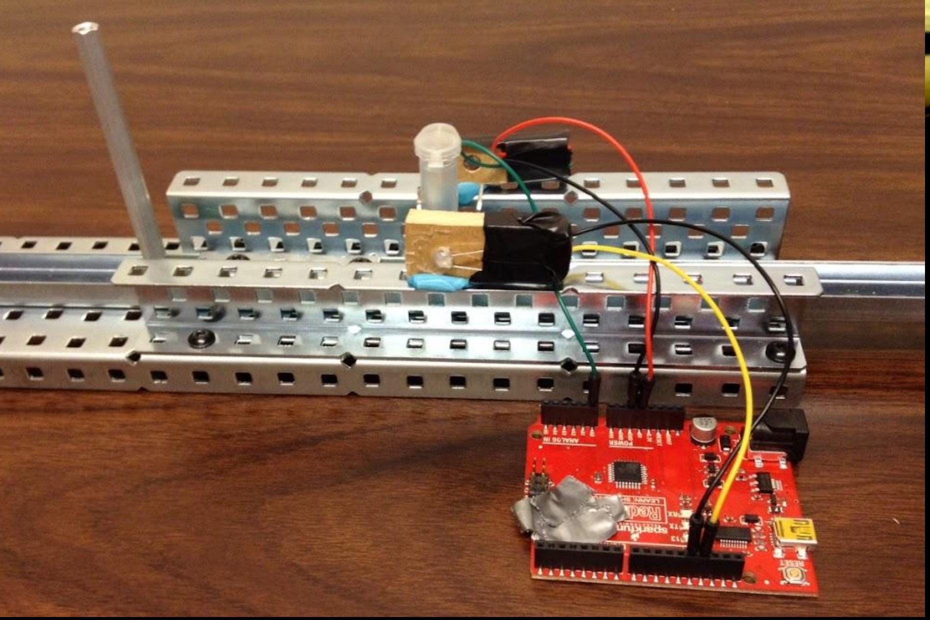
42 lines (28 sloc) | 0.919 kb   Raw Blame History

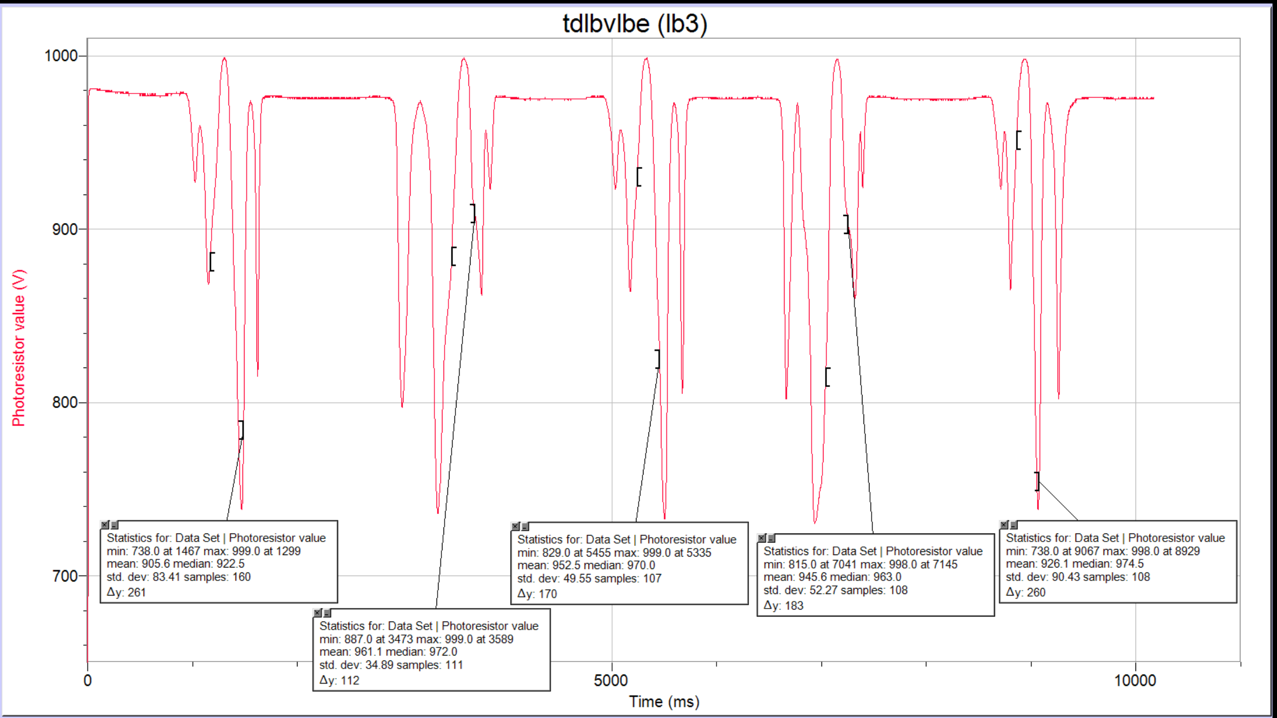
```c
/*
 * File:   usbDebug.c
 * Author: Sam Cuthbertson
 *
 * Created on November 6, 2014, 10:55 AM
 */

#include <nesi.h>
#include <math.h>

float getTemp(int);

int main(void)
{
    // Initialize all modules
    nesi.init();

    // Connect the USB COM interface
    usb.connect();

    while(1)
    {

        int val = resistiveSensors.getQ1(10,50);    // Read the tempurature sensor value
        int val2 = resistiveSensors.readQ2();   // Read the photoresitor value

        ledB.dutycycle(100);    // Power the motor at full
        ledR.dutycycle(100);    // Power the LED at full

        usb.printf("TempValue: %d Temp: %f Photo: %d \r\n", val, getTemp(val), val2);   // Log all the data over USB

    }

    return 0;

}

float getTemp(int i) {  // Forumla found by Brooks McDonald and Tray Guess

    return 70.815-20.33*log(.33796*pow(1.005989,i));    // Convert resistance value into temperature

}
```

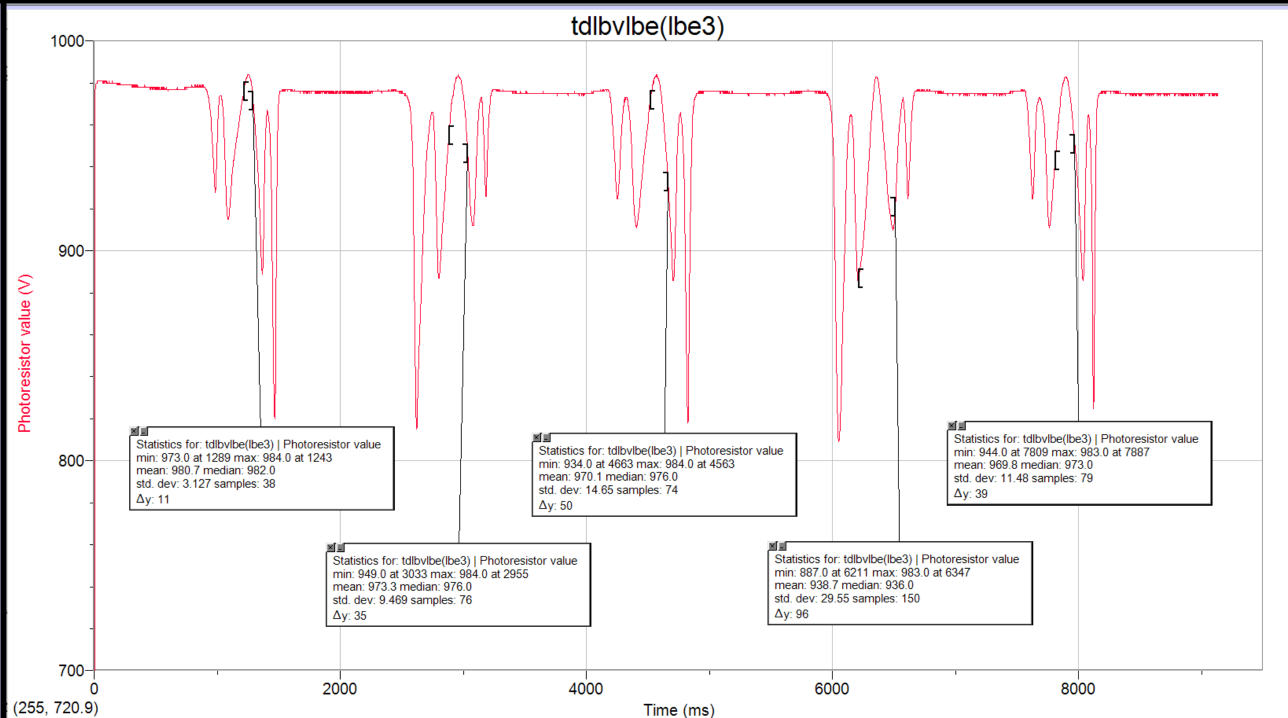
Turns things on – tests wiring

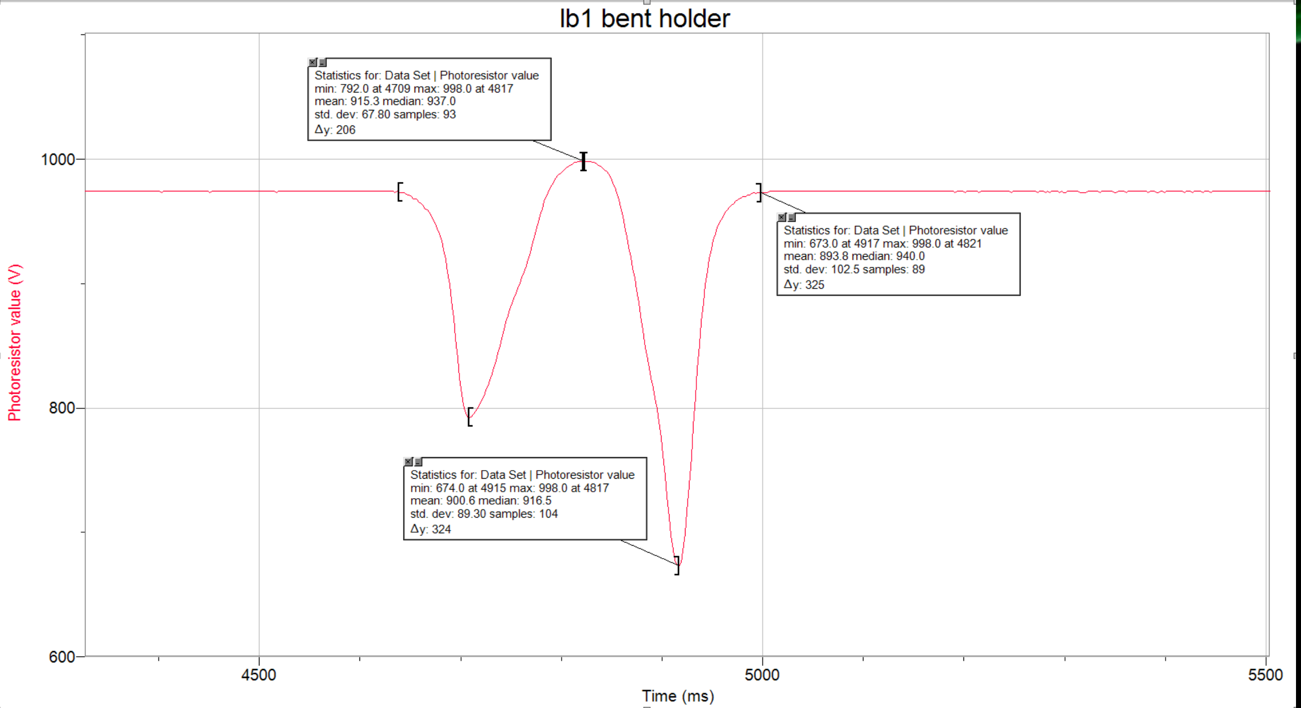
TESTING LIGHT SENSOR WITH BACTERIA

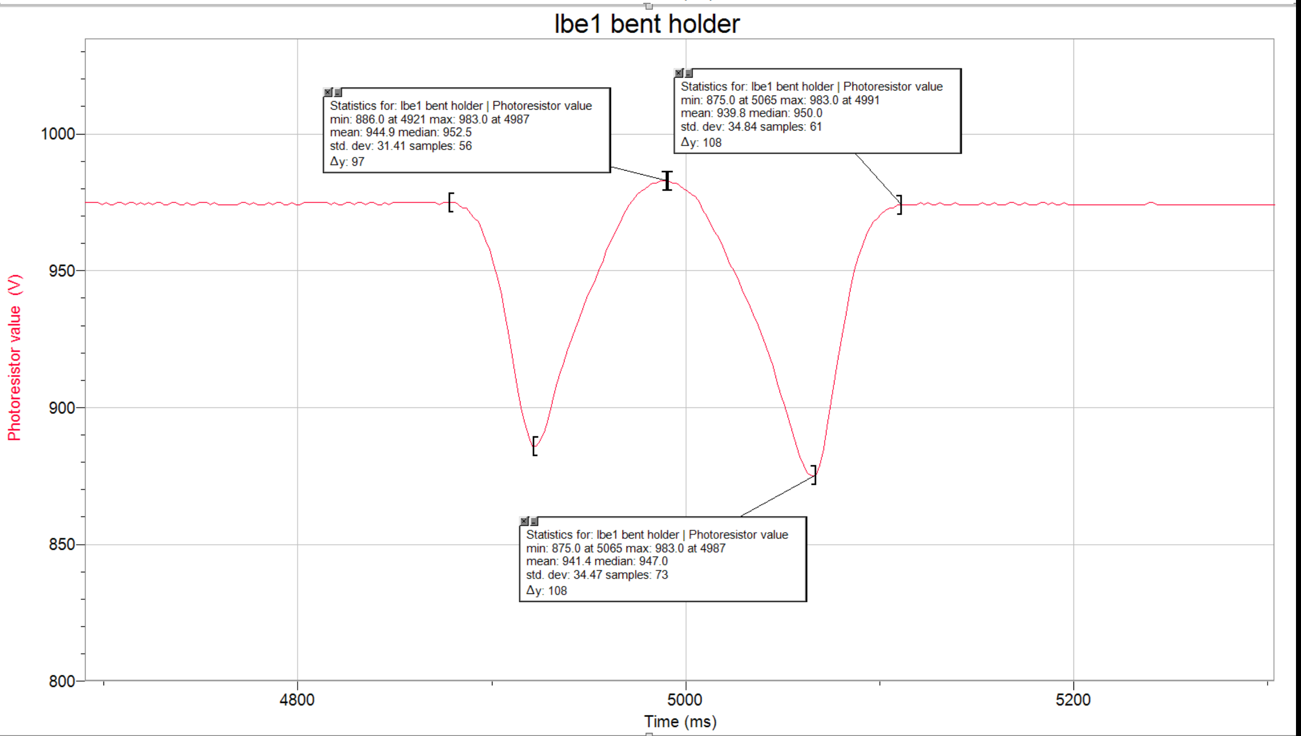
- Light sensor in dark room
- LB vs LB with e. coli



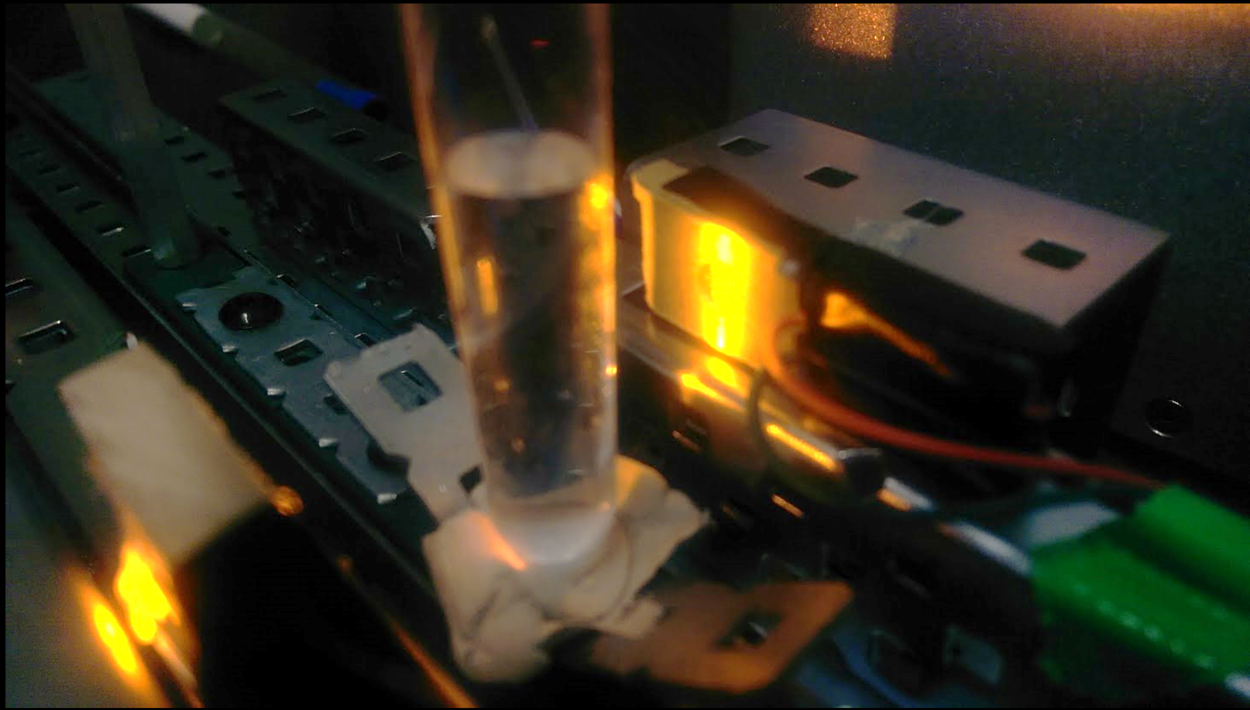








DESIGN ANALYSIS



How will we analyze data?

- Ground tests
  - Earth's gravity with centrifuge turning back and forth to agitate
  - Same code, same procedure, experiencing 1.41 G
- Data back from the ISS

MATERIALS

- Prototype List
- Alterations before launch
  - Aluminum tape wrap
  - PLA replaced by ULTEM 9085

IMPLEMENTATION AND OPTIMIZATION

- How to refine/revise
  - More gear iterations with self aligning drive shaft
  - Wrap inner casing with aluminum tape—fireproof
  - Create/Test parts with final materials
  - Design/Test tops for separate gear systems
  - Countersunk screw holes in each side—hold inner casing in place
    - Epoxy screws for launch

PROJECT EVALUATION

- How will we know if it worked?
  - If lag phase recorded aboard ISS matches the lag phase of ground based testing, gravity will be isolated as the causal variable in change of lag phase. If results do not match further testing is required to determine other possible variables such as magnetism or radiation.
- Follow up questions for future testing
  - If not the lack of gravity, then what changes lag phase? (radiation, magnetism, etc.)
  - If gravity, how do we know that’s the only factor?
  - How does this change how we travel in space?

A SPECIAL THANKS TO...

Mentors

- Alli Westover
- April Spinale
- Paul Ellis, Jo Charlton, Fan Yeung
- Tracy Copp, Denise Henry
- Hector Paz
- Brian Thomas

Team members

- Mikayla Whiteaker
- Matt Maierhofer
- Corby Tobin
- Jordan Hammer
- Tristan O'Connell
- Brooks Macdonald
- Kate Dickinson
- Justin LaVelle
- Mary Hanson
- Joe Sagrillo
- Karl Painter
- Nate Weston
- Melanie Smith
- Meghan Donohoe
- Chris West
- Shayne Albertson
- Alexis Engel
- Dinah Bowman
- Nikhil Krishnan

QUESTIONS