SUN TRACKING SOLAR PANEL ARRAY FOR ARCTIC APPLICATION

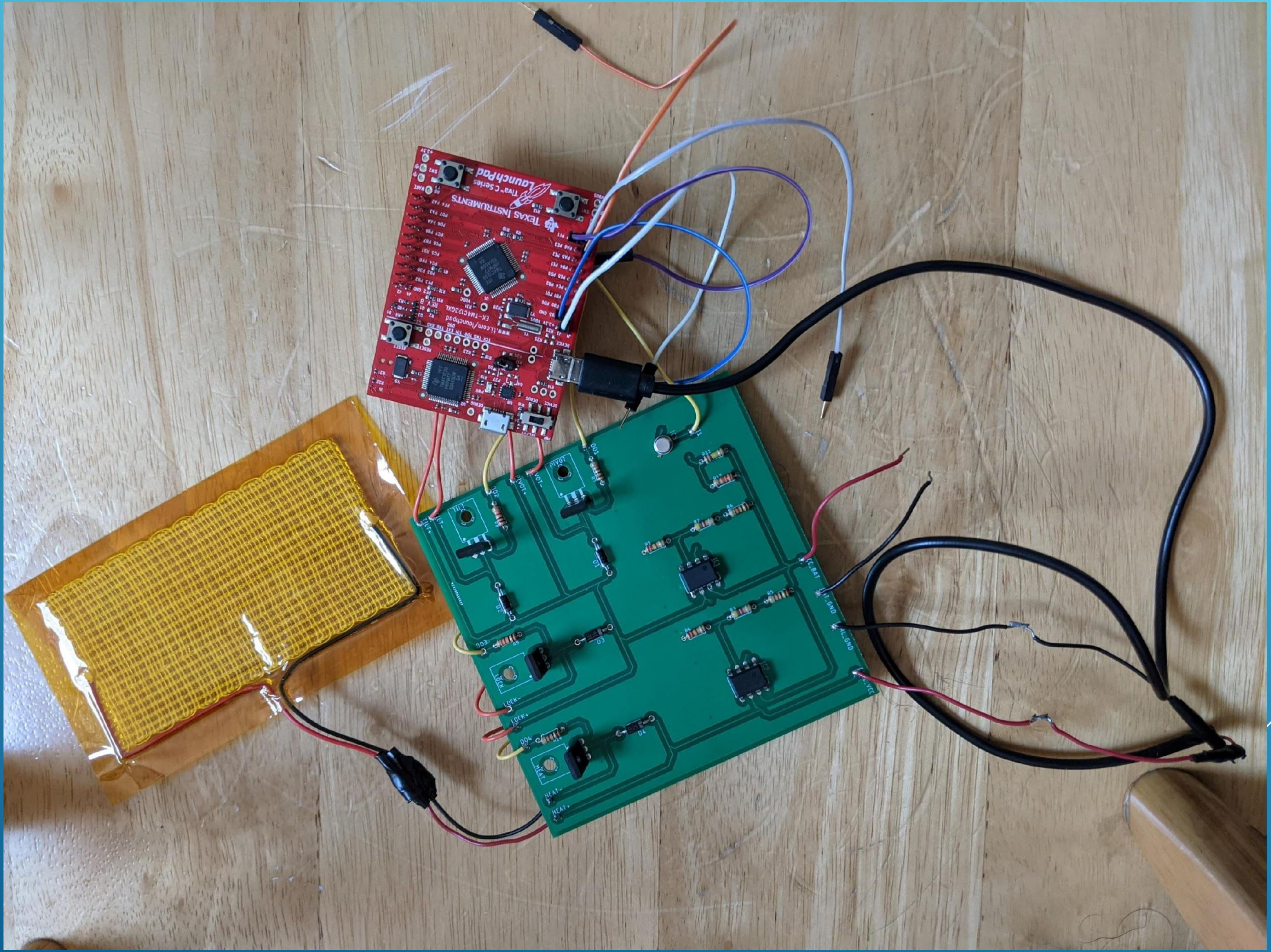
Team 34 – Ashhal Shamsi, Brandon Ewan, Zayd Saeed

Introduction

- Solar Panels exist currently, but they bring a level of inefficiency that we sought to improve
- Specifically we focus on the regions of Alaska as they are colder and harsher
- Our goal was to create a system that allowed for a better solution to clean energy in these environments with the use of more efficient solar panel arrays

Overall Design

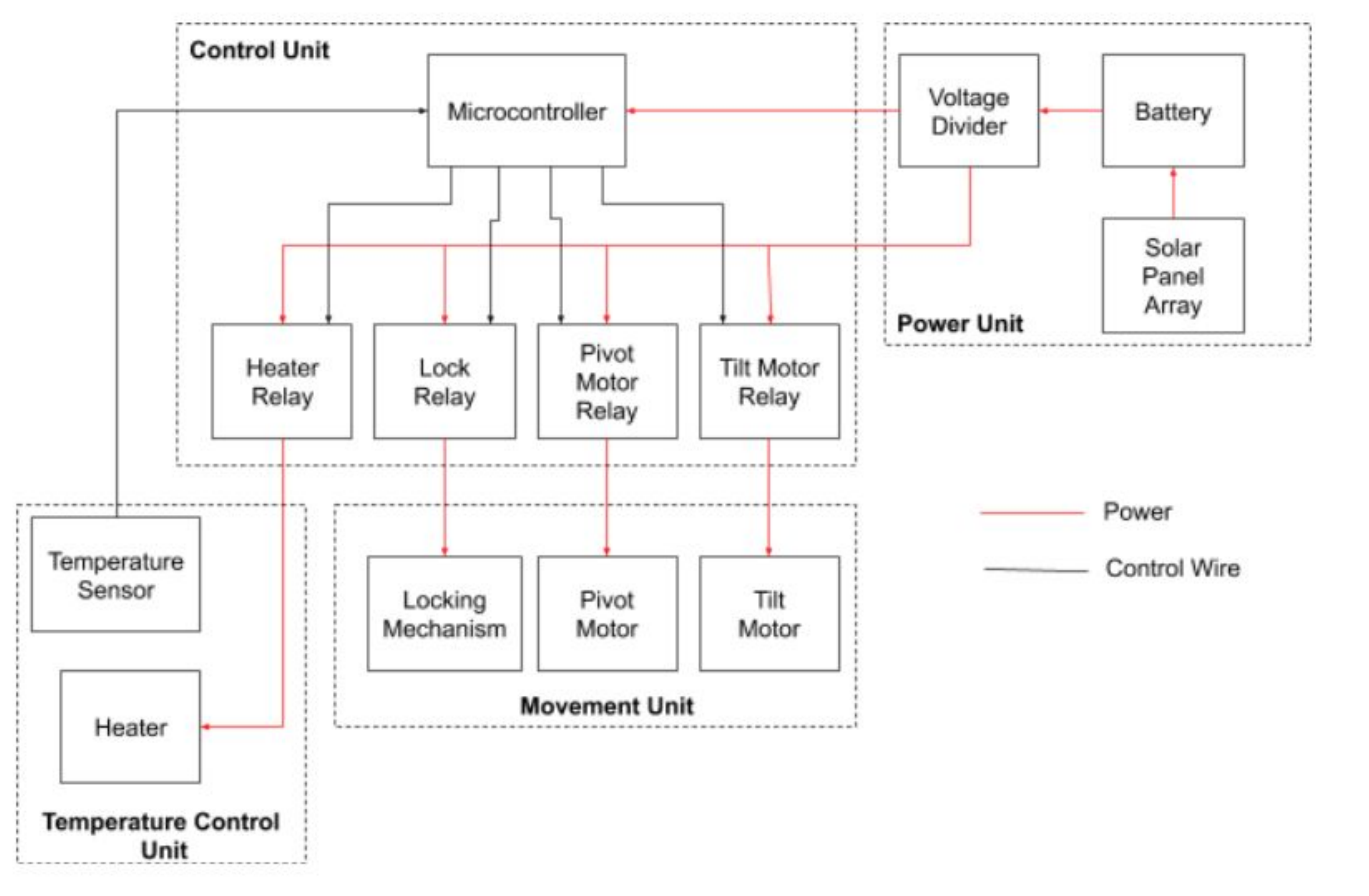
PCB/MICROCONTROLLER LAYOUT



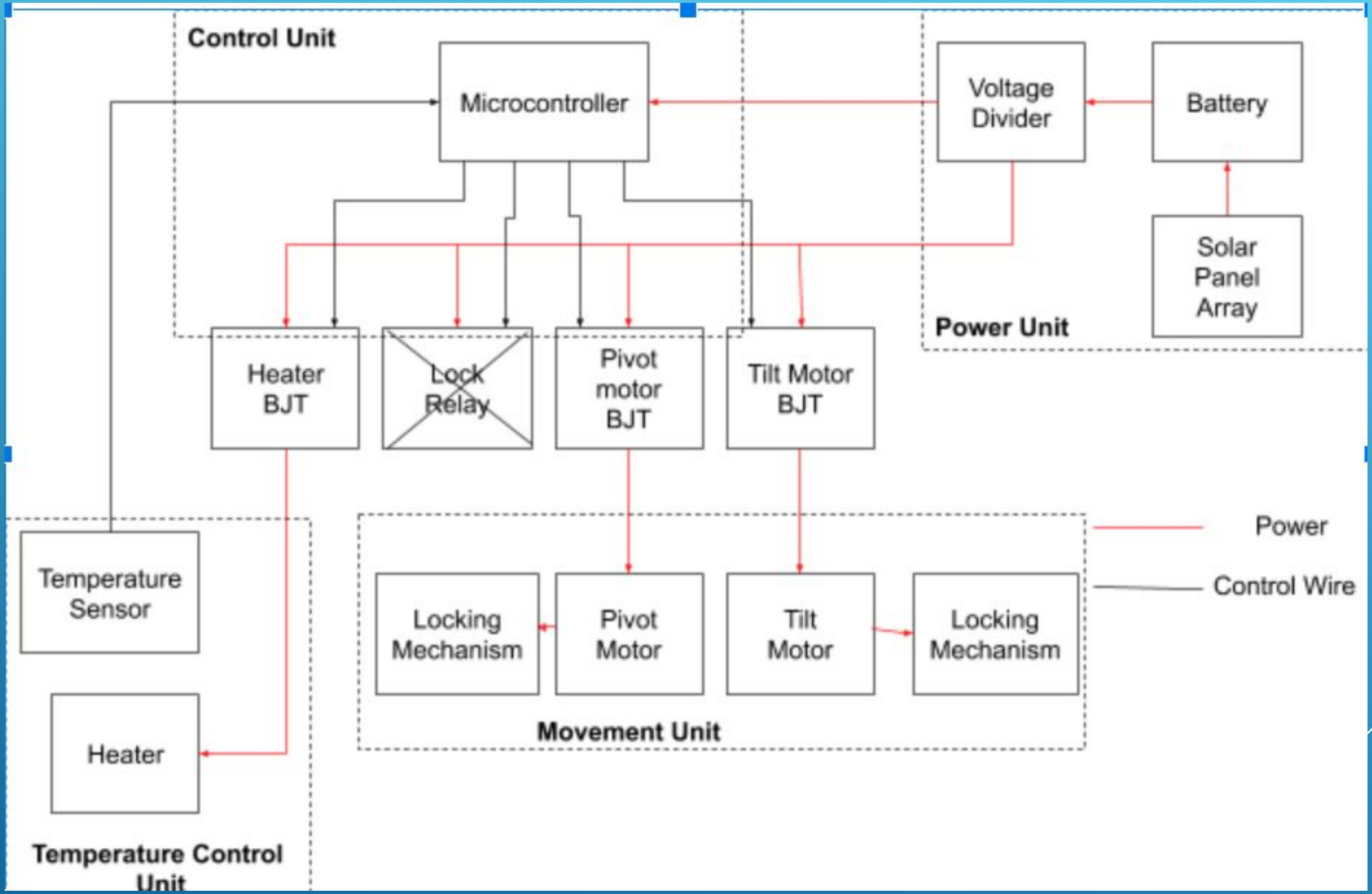
System Overview

- **Hardware**
  - **Microcontroller**
  - **PCB with heater, temperature sensor, and voltage divider**
- **Software**
  - **Sun tracking code**
  - **Heater relay code**

BLOCK DIAGRAM (Original)



BLOCK DIAGRAM (Final)



Hardware Overview

- **PCB**
  - **Voltage Divider**
  - **Temperature Sensor**
  - **Relays**
- **Motors**
  - **Pivot Motor**
  - **Tilt Motor**

Software Overview

- Sun tracking
  - Implemented scientifically proven equations to track sun
  - Tilt motor moves hourly
  - Pivot motor moves daily
- Heater Relay
  - Heater turns on 10 degrees above critical temperature
  - Heater turns off at ambient temperature

Tracking

- For the solar panels to generate the most power the sunlight needs to be perpendicular to the panels
- The sun travels throughout the year changing its angle every hour of every day making the panels lose efficiency throughout the day
- We modeled the sun’s movements using our microcontroller

Tracking Implementation

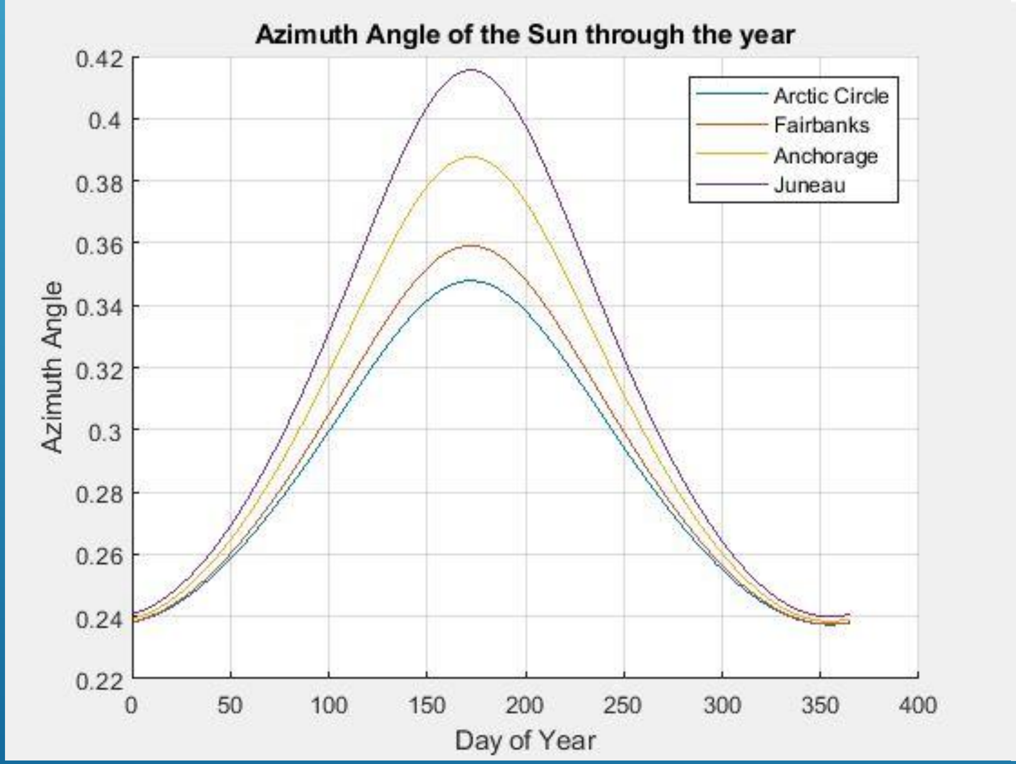
- **Tracking of the Sun is done by using a set of three equations**

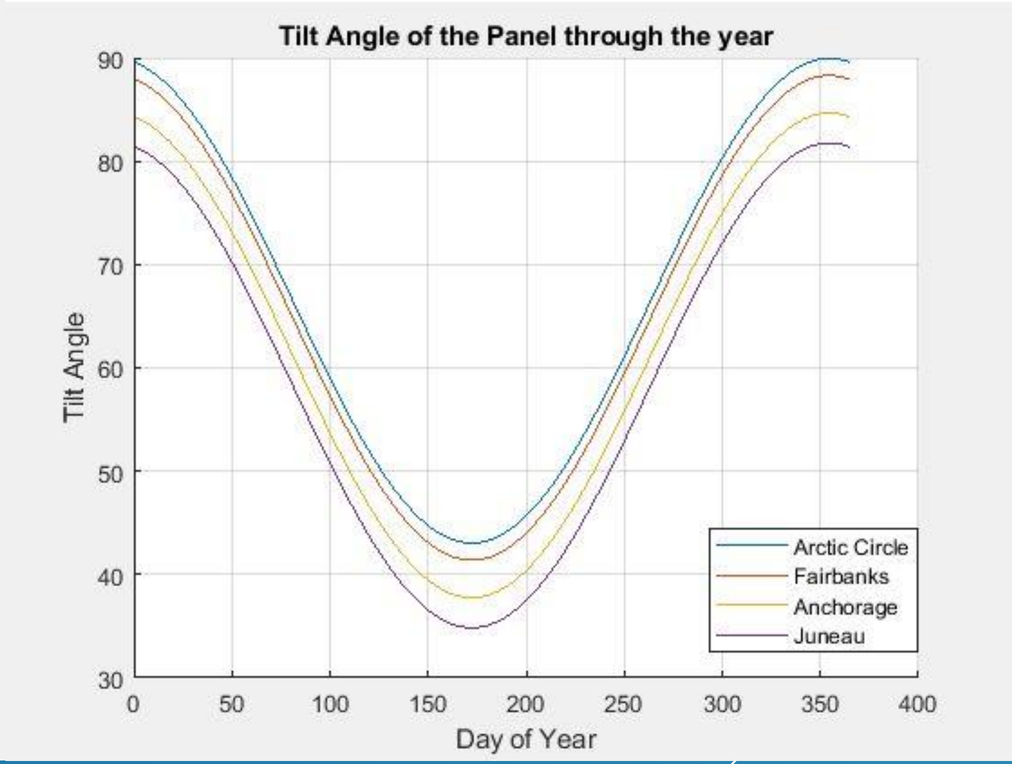
1. δ = 23.45sin(360/365 (n-81)),
   n = day of year

2. Tilt Angle = L - δ,
   L = Latitude

3. φ = arcsin(cos(δ)sin(H)/cos(90-L+δ)),
   H = 15° * (hours before or after noon)

Models of Tracking Algorithm





Angle Calculation Implementation

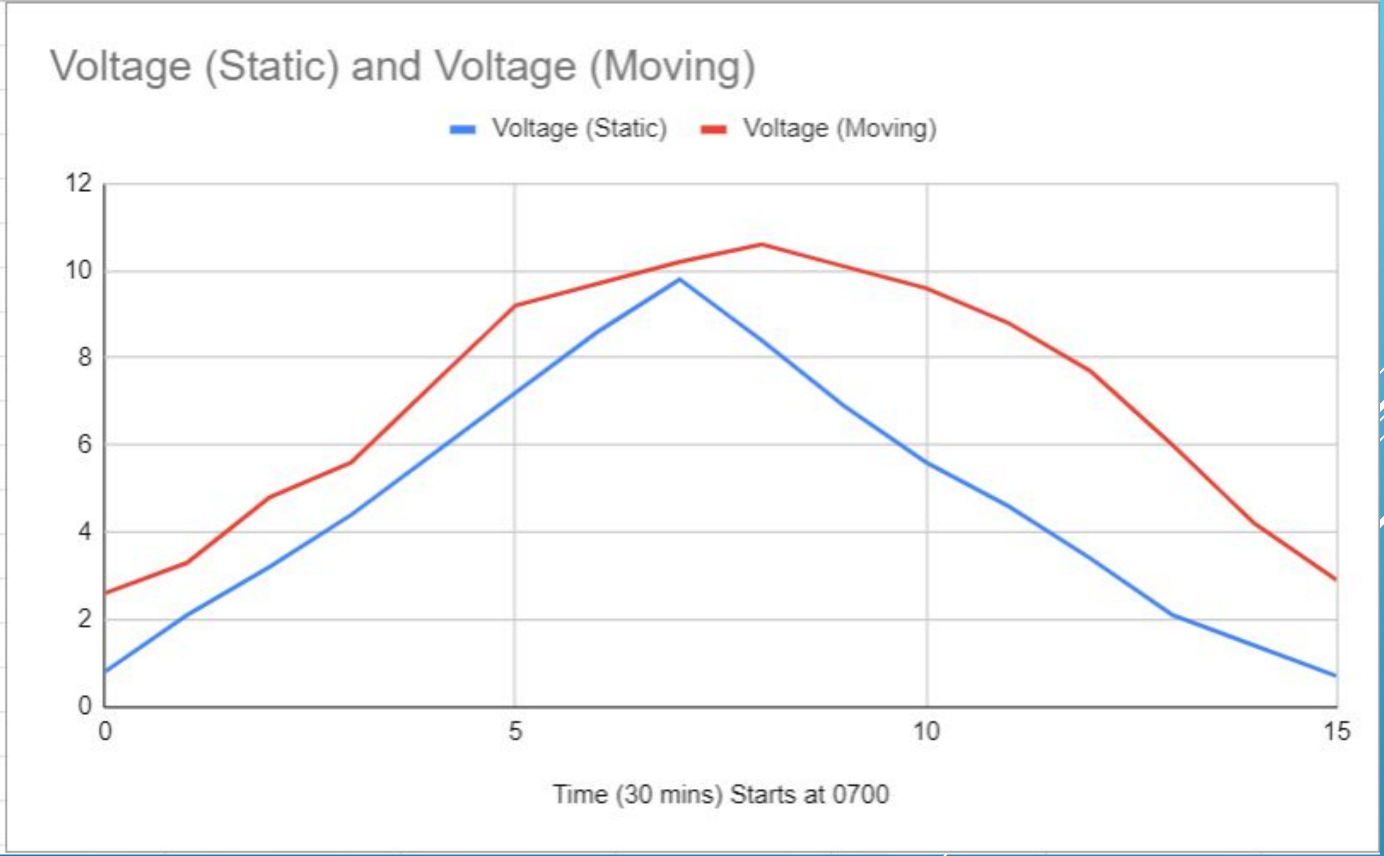
```
double declinAngle(double m){
  if (m == 0)
    m = n;
  double sigma = 23.45 * sin((360/365*(m-81))*torad); //this is the solar declination angle
  return sigma;
}

double altitude(double L1, double m){
  double sigma = declinAngle(m);
  double beta  = 90 - L1*todeg + sigma*todeg; // altitude angle
  return beta;
}

double tilt(double m, double L1){ // this is what will be called for the
  double beta = altitude(L1, m);
  double tilt1 = (90 - beta)*torad;
  return tilt1;
}
```

Tracking Verification

Time (30 mins) Starts at morning	Voltage (Static)	Voltage (Moving)
0	0.8	2.6
1	2.1	3.3
2	3.2	4.8
3	4.4	5.6
4	5.8	7.4
5	7.2	9.2
6	8.6	9.7
7	9.8	10.2
8	8.4	10.6
9	6.9	10.1
10	5.6	9.6
11	4.6	8.8
12	3.4	7.7
13	2.1	6
14	1.4	4.2
15	0.7	2.9



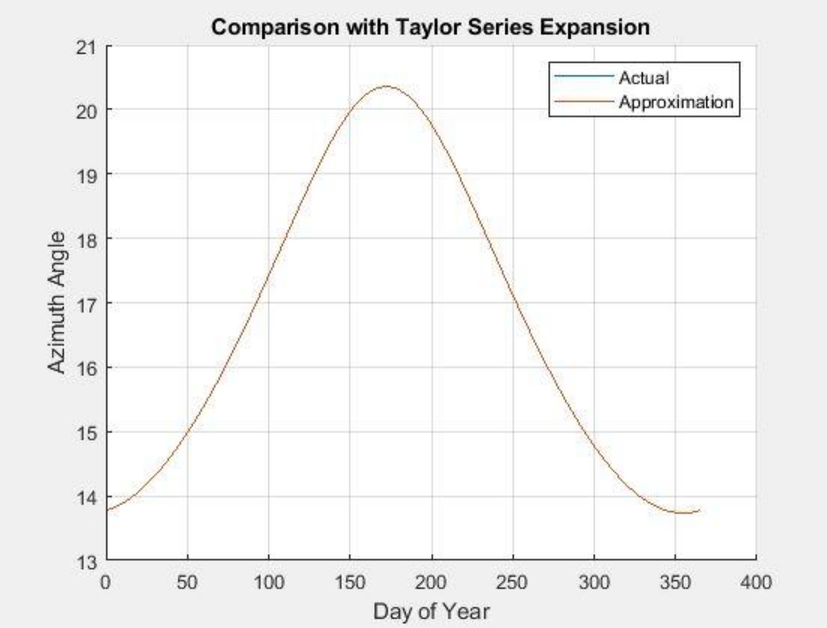
Pivot calculations

```
double pivot(double m, double H1, double L1){ // this is what is called to get the pivot angle
  double sigma = declinAngle(m);
  double beta = altitude(m, L1);
  double lambda = arcsin(cos(sigma)*sin(H1) / cos(beta));
  double cosH = cos(H);
  double tanS = tan(sigma);
  double tanL = tan(L);
  if (cosH >= (tanS/tanL))
    return lambda;
  return pi + lambda;
}
```

Comparison with Taylor Expansion

```
double arcsin(double len){
  double temp = len +
              pow(len, 3)/6.0 +
              3*pow(len,5)/40.0 +
              5*pow(len,7)/112.0 +
              35*pow(len,9)/1152.0; // had to create arcsin because it didnt exist :p
  return temp; // need to turn this into a fourier series to return the arcsin without any issues
}
```

- **Percent error of the expansion 0.0000233%**



Temperature

- **Part of our consideration was dependent on the temperature for the device since the environment in Alaska is not optimal for electrical components such as ours**

- **Devices that do not need heating:**
    - **Solar Panels:**

- **Devices that do need heating**
    - **Batteries**
    - **Motors**
    - **PCB components**

Insulation/Heating

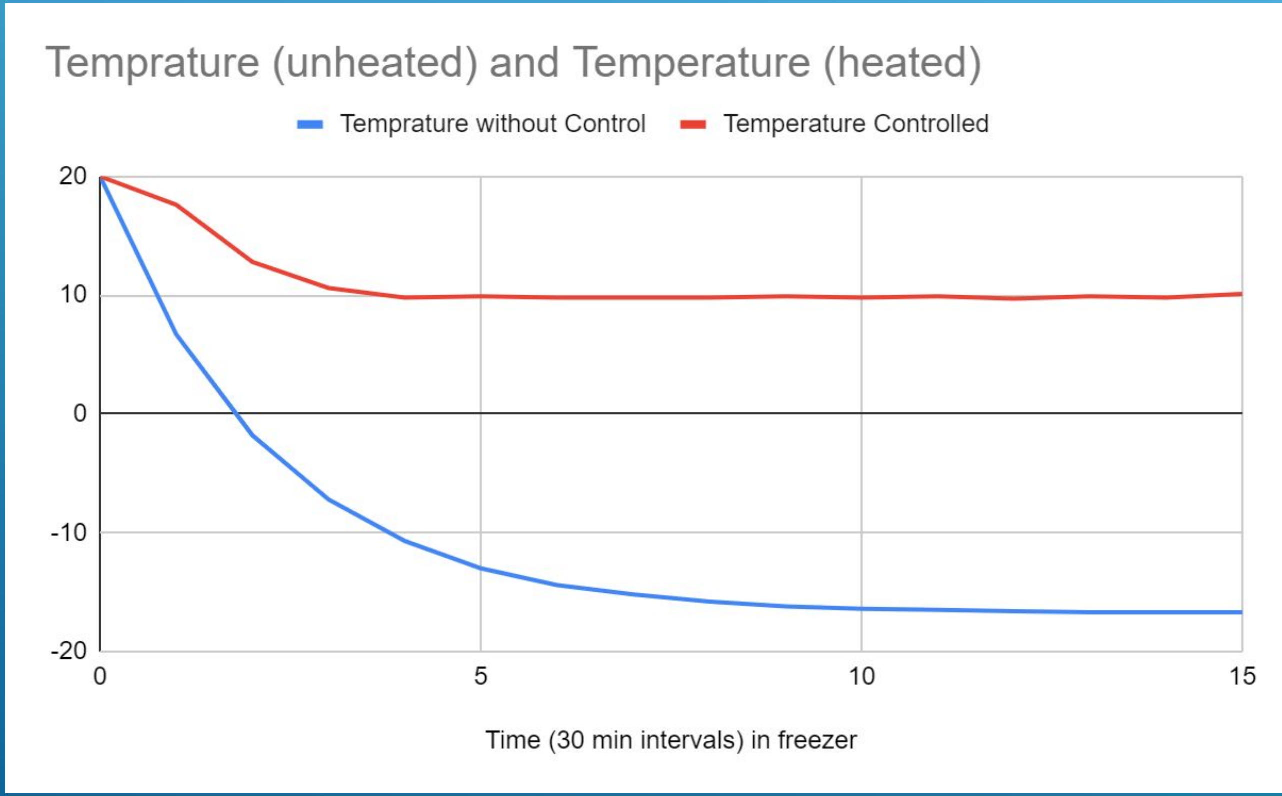
- To counter the colder elements, the solar panels are exposed to the environments while the other devices are not.

- The rest of the components are covered with fiberglass insulation: comparable to cellulose as part of our original requirements
    - (Fiberglass was more easily accessible)

- Heating was used to supplement the insulation
    - The heating pad depends on a temperature sensor to make sure it does not dip too low

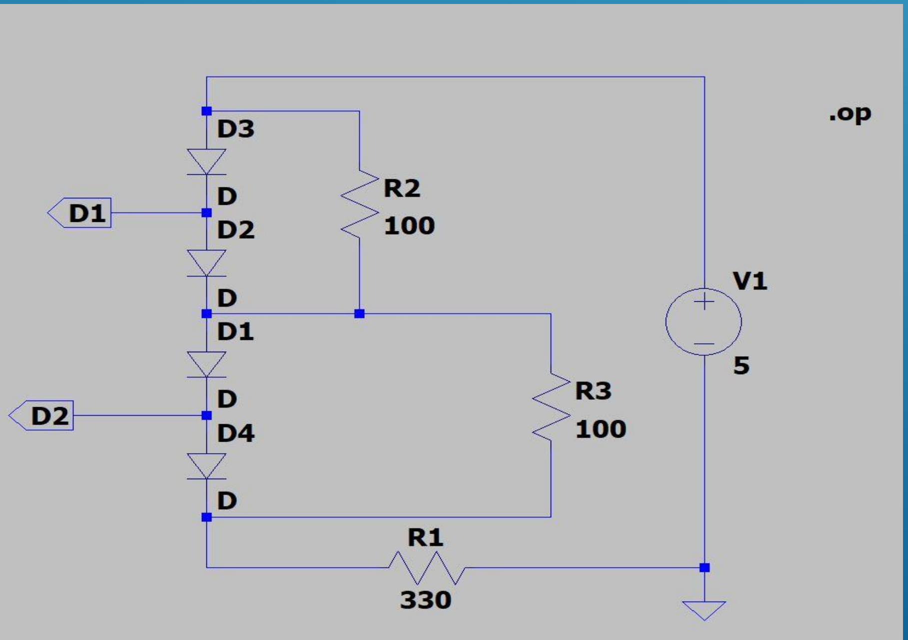
Insulation/Heating Verification

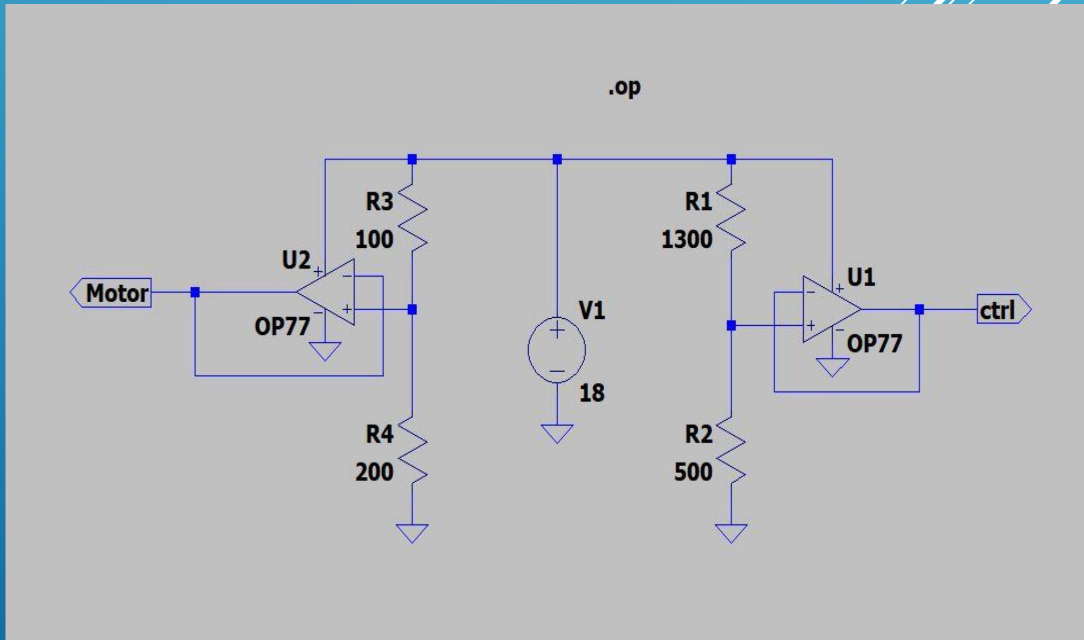
- **A modern freezer decreases to roughly -18°C**



Voltage Divider Design

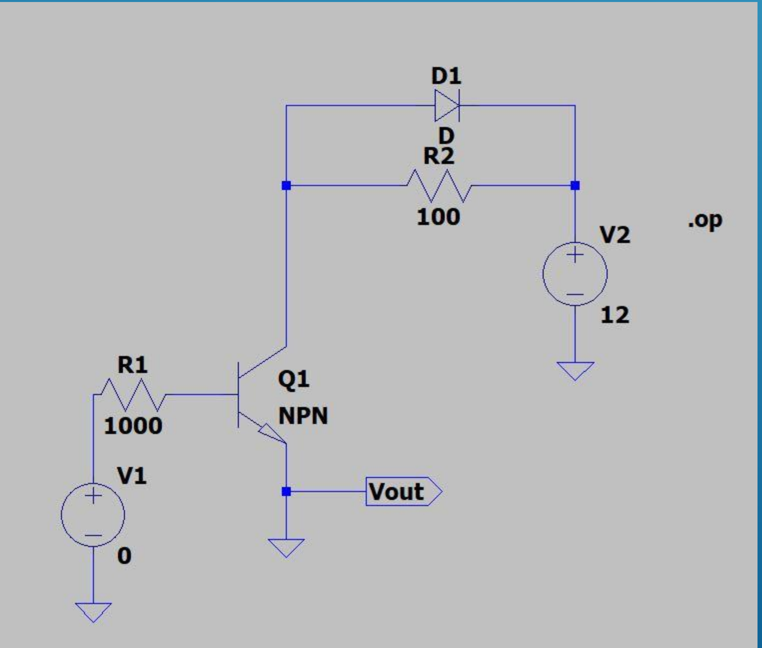
- **Voltage divider needed to divide 18V into 5V for all the devices**
- **The initial idea was to use diodes and create a constant voltage across them**
- **Next idea was to use voltage following op-amps**

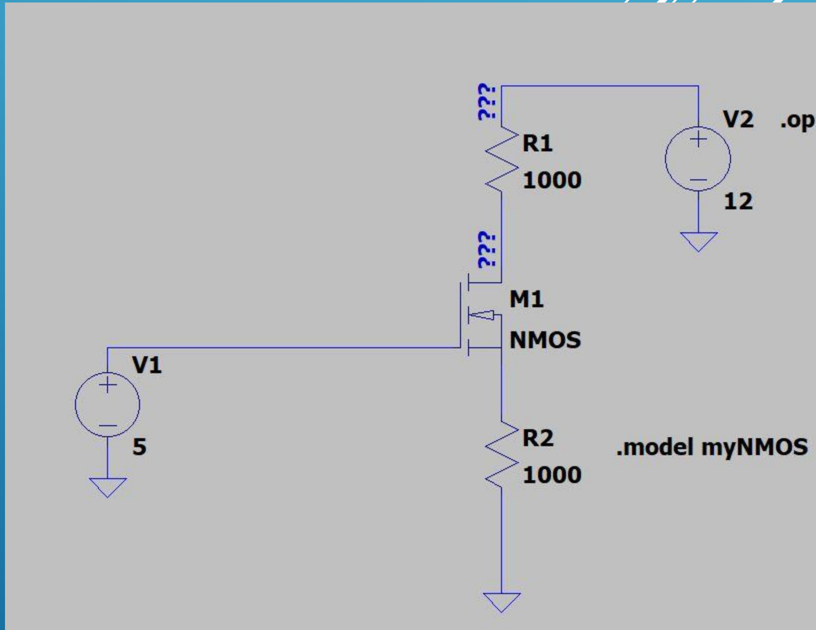




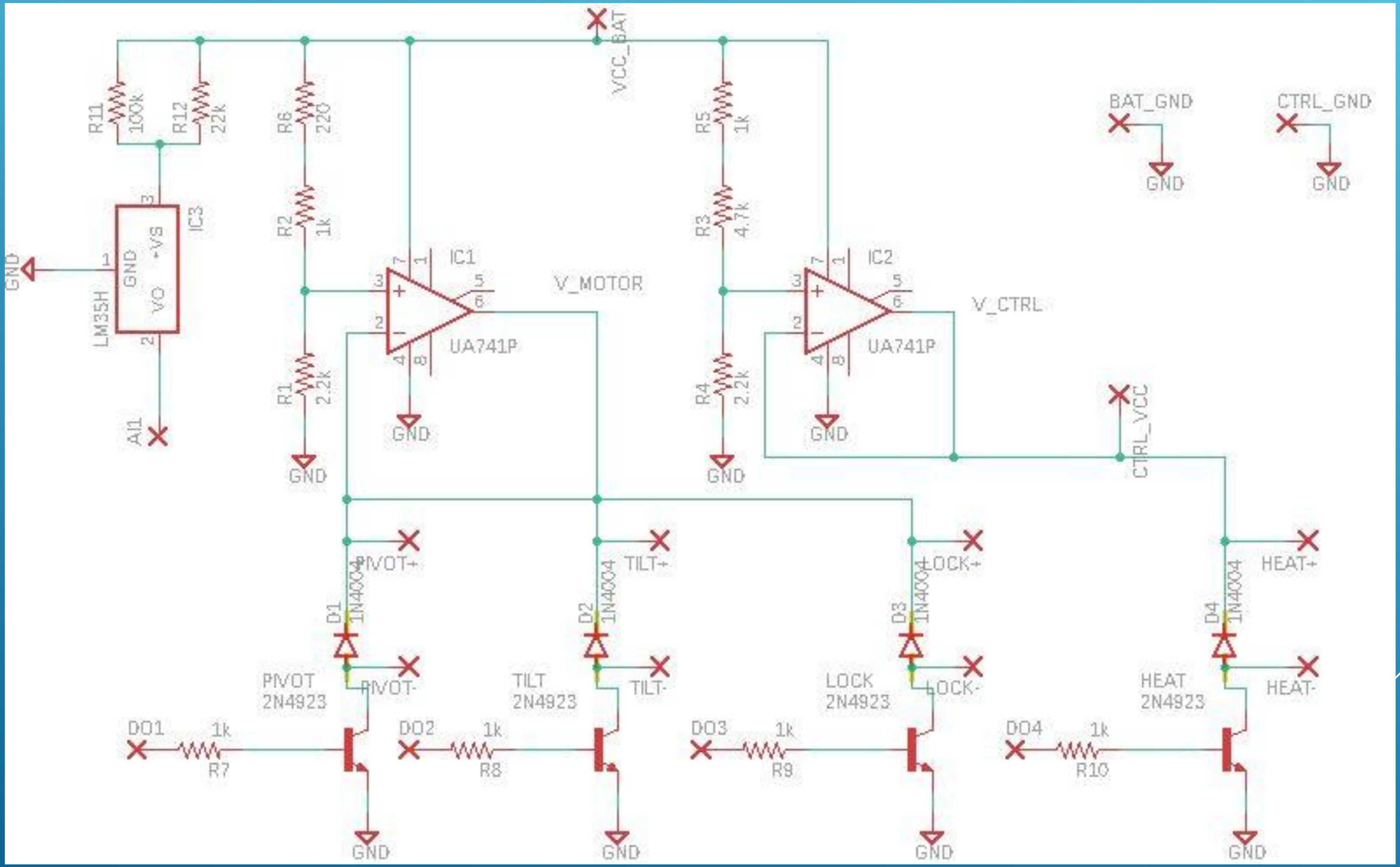
Relay Design

- Needed the relay to act as a controllable switch
- Decided to use transistors like a controllable switch
- BJT design was the best for our needs





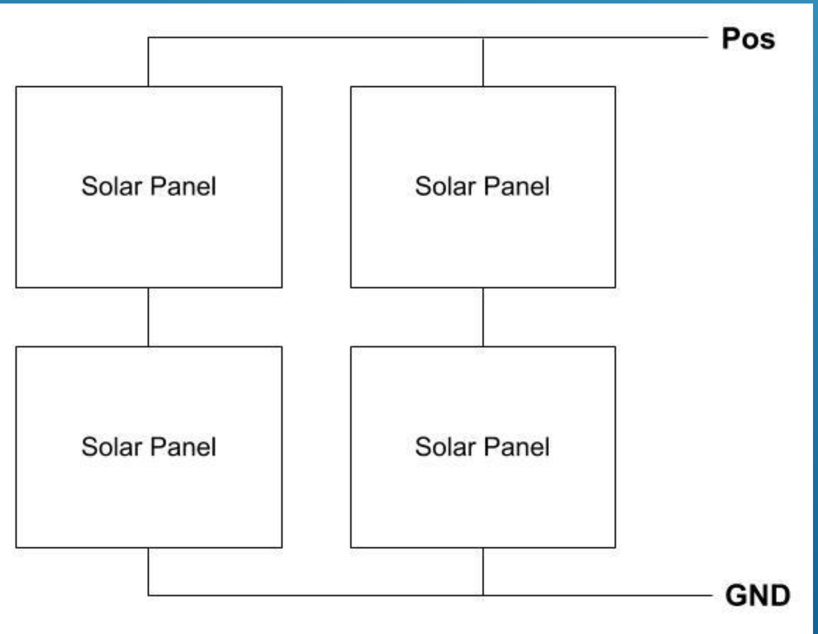
PCB Schematic

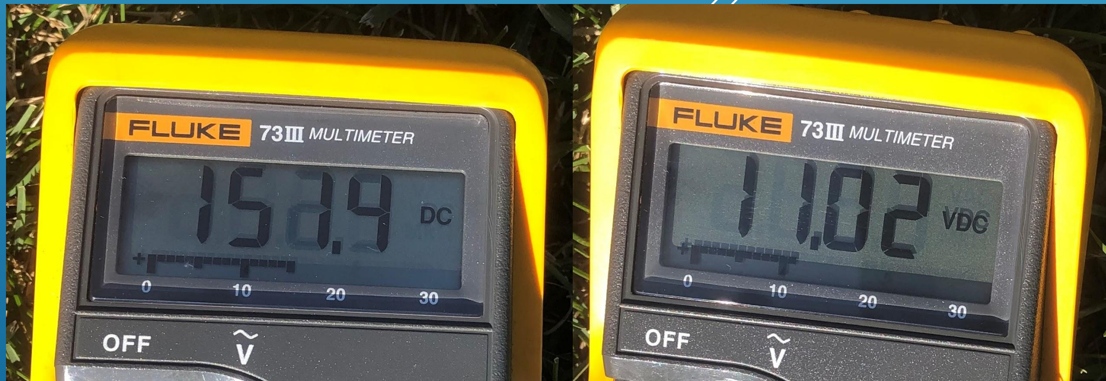


Solar Panel Array Wiring

- **Wanted to have the solar panels supply enough power to power the internal components**

- **To do this two panels in series were wired in parallel with another two panels in series**





Design Challenges

- Originally power was going to be driven through Op Amps with our circuit this was switched to having our circuit use BJTs to connect power from our microcontroller since the microcontroller could not supply enough current

- Cellulose was swapped with fiberglass for this model

- We used two nine volt batteries as a proof of concept because a rechargeable battery for the arctic environment was too expensive

- The Microcontroller only worked with windows 7

Error Analysis:

Angles were start about 5 degrees off from our analysis. This error would get slightly worse throughout the day. Implementing PID control would have helped maintain a higher level of accuracy throughout the day.

Features and Conclusion

- **Scientifically proven algorithm implemented by microcontroller to ensure accurate sun tracking based on location and time of year**

- **Temperature sensor and inductive heater to ensure thermal regulations**

- **Scalable design**

Questions?