Testing the Electric Field Circuit of the Earth:

In conjunction with Temperature, Pressure, and Humidity

Team Revolution:

Progress Asoluka, Austin Ginger, Kim Buchanan, Angel Gatchell

Community College of Aurora

Professor/Advisor: Victor Anderson

progressasoluka@yahoo.com, sixminslater@aol.com, dannkimkarissa@aol.com, adgatchell@student.cccs.edu

April 8, 2013
Table of Contents

Abstract ........................................................................................................................................3

1 Introduction .................................................................................................................................3-5

2 Materials ....................................................................................................................................5

2.1 Design & Testing .......................................................................................................................5

• 2.1.1 Arduino Uno
  ▪ Sensors
  ▪ Programming and Testing

• 2.1.2 Electric Field Mill
  ▪ Housing
    ▪ Prototype 1
      ▪ Testing
    ▪ Prototype 2
      ▪ Testing
    ▪ Prototype 3
      ▪ Testing
    ▪ Final Prototype
      ▪ Testing

• Blades

• 2.1.3 Payload Design
  ▪ Flight stimulation Test & Results
    ▪ Drop Test
    ▪ Whip Test
    ▪ Stair Test
    ▪ Cooler Test
    ▪ Stimulated Electric Field Test

2.2 Discussion ..............................................................................................................................14

• Final Electric Field Mill
• Related to NASA
• Launch and Recovery

2.3 Conclusion .............................................................................................................................14

3.0 Acknowledgements

3.1 References ..............................................................................................................................15

3.2 Appendix .................................................................................................................................16

• Arduino Code
Abstract

The global electric circuit is caused by a difference in the electric potential between the atmosphere and the Earth. Climate studies benefit from the analysis of changes in this circuit. High-altitude balloon experiments are valuable resources for the study of the global electric circuit; however, the electric field mills used for detecting changes in the electric field are too expensive and heavy for these experiments. Team Revolution’s goal is to design and engineer a functional prototype for an electric field mill that will be used to map the electric field strength in Earth’s atmosphere. Team Revolution will accomplish this by building an inexpensive and light-weight electric field mill that will withstand the conditions during the high-altitude balloon launch and recovery. Data will be collected using and Arduino Uno, programmed by Team Revolution, and this data will provide an electric field profile from Earth to the stratosphere. Sensors will be integrated into the system to record changes in Earth’s electric field due to altitude, temperature, humidity, and pressure. The results that we expect to gather from the experiment will show that the electric field substantially increases with altitude in the lower atmosphere, but will drop with decreased pressure. The purpose of the experiment is not only to document and study the electric field profile, but also to design, build, and program an efficient electric field mill. If the electric field mill proves to be successful, the design and programming may be used as an example for future student balloon experiments. Once the mission is completed, Team Revolution will present both electric field and functionality data collected during the experiment. Any necessary modifications will be developed and presented at this time.

Introduction

Electric field research has shown that the Earth maintains a negative charge which is countered by a positively charged atmosphere. Changes in the global electric circuit can be greatly affected by the presence of humidity, low pressure, and electrostatic discharge in the form of lightning strikes. The potential that we may expect to observe during the ascent to the final height of 30km will fluctuate due to altitude and humidity. The electric potential has been shown to increase at a rate of approximately 100V/m as altitude increases (being relatively close to the earth), but will begin to drop as the balloon’s altitude continues to increase. It is important to understand the atmospheric effects of the electrical field because climate is directly related to the electric potential between the Earth and the atmosphere [1].
The purpose of the experiment is to create an electric field profile from the Earth to the stratosphere. Team Revolution will accomplish this by building an inexpensive and light-weight electric field mill. Purchasing a professionally manufactured electric field mill will not be possible; the prices begin at $1,000. An electric field mill is a device used for detecting electric fields using a motor to rotate a metal plate over another plate. The plates both have wedges (or slots) cut out of them which creates a shutter-like effect when the top plate is rotating. The sensing plate (the lower plate) remains stationary and is grounded to the mill and this will allow it to detect the electric field while on the high-altitude balloon flight because the charge will be going through the ground wire, which will be linked up to the Arduino for measurement. The shielding plate rotates alternately exposing and hiding the sensor plate from the electric field that is being tested. Grounding the sensor plate allows for a more accurate reading, and keeping the two plates as close to each other as possible allows for less interference and unwanted stray electric charge to build up between the plates, minimizing the distortion of the readings and improving the accuracy of the data that will be collected by the system [2].

Team Revolution’s goal is to engineer a functional prototype for an electric field mill that can be used to map the field strength of the Earth’s electric field by sending it on a high-altitude balloon launch. The prototype is being developed on a 3-D Alibre software program (personal edition) and can be printed on a 3-D printer for initial testing. The electric field mill that will be used in the experiment was printed using the 3-D printer and aluminum tape will cover the polymer housing to allow for conductivity. The blades were also designed using the 3-D program and the final prototypes were manufactured from aluminum by Pueblo Community College. The mill will be assembled using the housing, blades, a small hobby motor, and a
motor axle mount. After the assembly, the mill will be tested for functionality during high-altitude flight conditions.

**Materials**

Arduino Uno- the Arduino Uno is the main brain of our experiment and will collect and store our data to an SD card, can be found on their website for ordering.

Protoshoield- The sensors will be soldered to this component.

SD shield- This component contains the mini-SD card which will gather data.

Camera- We are required to fly a camera on our payload, the camera will be programmed to take pictures every 30 seconds and will be independent from the Arduino Uno.

Heater Circuit components- Our payload is expecting to reach regions of the atmosphere as cold as -80 degrees Celsius, a heater is required to keep our Arduino and Electric filed mill motor warm.

Temperature, humidity, and pressure sensors- These sensors will provide vital data to our experiment, which will aid us with the analysis of our data.

Five 9 volt batteries and two AA batteries- The 9 volt batteries are used to power our heater component, Arduino, and EFM motor, while the AA batteries are used to power our camera.

Shield and Sensor plates- These are made of aluminum and will experience the potential differences in the electric field.

11,000 rpm motor- The motor will be rotating the shielding plate, which will allow us to measure the change in the electric filed circuit.

Copper electrical wiring- We will use this for our wiring and circuits.

Electric field mill housing- Our electric field mill housing will house the motor and the two plates.

Foam core- we used this for insulation.

Soldering kit

Hot glue

Aluminum tape

Electric toggle switch

Electric tape

**Design and Testing**

Team Revolution’s experiment will be designed to fit within an 850 gram payload and a $250 budget. This experiment will be an addition to the required components for the payload. The required components on the payload are an Arduino Uno, micro SD card shield, proto shield, external and internal temperature sensors, digital camera, heater, humidity sensor, pressure sensor, and any power sources these components require.

Our main experiment requires an electric field mill in order to gather data about the voltage present throughout the balloon flight. Purchasing a professionally manufactured electric field mill is not possible with the
budget and mass constraints of this project. We will build and design an electric field mill using references from other resources and the team’s own design. The mill was built by designing the electric field mill housing on a 3-D printer using AutoDesk products for 3D design and having the plates machined by our Space Grant colleagues in Pueblo. A small motor, powered by one 9V battery, is required to operate the mill and was purchased locally at a hobby store. The provided Arduino Uno will be used to monitor voltage gathered by the mill.

**Arduino Uno**

The Arduino Uno will be the main brain of the entire payload. It will run the components and gather data on an SD card. Team Revolution will be programming the Arduino Uno to gather data from the sensors and electric field mill for the experiment. Team Revolution will have to test different time increments for data collection and find the timing that reflects the most consistent data recovery. During the programming of the Arduino Uno, each of the components and the systems will be thoroughly tests to ensure that they are working properly. The functionality tests will allow Team Revolution to monitor the efficiency of the individual sensors and the entire system before attaching them to the payload for simulation testing.

**Sensors.** We are attaching four different sensors to the Arduino, including analog and digital temperature, pressure, and humidity sensors. These sensors will provide important data to our experiment and will be used in the analysis of our data. These sensors will be calibrated to optimize its accuracy, and the Arduino will be programmed to recognized these sensors and collect data from them.
**Programming and testing.** We programmed the Arduino Uno using the software provided by the Arduino Company. We compiled our entire program in one sketch folder, including the programming for the humidity, temperature, and pressure sensors; we also included our EFM code in this folder. Upon collecting data from these sensors, our Arduino writes them to the SD card every set amount of time; the time it takes to write data to the SD card is dependent on the programmer and also how many sensors or electronics are wired to the Arduino, the higher the amount of sensors the more time it takes to collect and write to SD card.

We performed various tests on the Arduino and to the sensors to ensure that they are working and flight ready. We tested in different environments and conditions, after the mission simulation test we are confident that the sensors and Arduino are working and recording accurate data. Although we are not able to test our EFM because it has not been completed yet, we are confident that the Arduino will be able to recognize, collect, and record data from it.

**Electric Field Mill**

The first possibility of the electric field mill construction is to use outside resources combined with the team’s own design. An online paper details the build of an inexpensive electric field mill used in a high altitude balloon experiment. This document would be the most widely referenced in the build of Team Revolution’s mill [3]. This design utilizes sheet metal, a 10 cm diameter soup can, and an 11,000 rpm motor for the electric field mill. The soup-can will be cut down to a 3 cm depth and a hole will be drilled in the bottom to mount the motor to the bottom of the can, which will have the same diameter as the motor being installed, with a 1/100cm tolerance to allow for thermal compression due to the temperature change. To make the sensor and shield plates, sheet metal will be cut into two 9.9cm diameter circles and several wedges are cut out. The wedges are cut out in order to alternately expose the sensing plate to the electric field every 15 degrees. Holes will be drilled into the center of these plates and a motor extension axle will be used to mount them to the motor. Once the motor is mounted to the housing, the extension will be attached to the motor with the rotating plate attached; the sensor plate will face out of the mill, with the axle protruding, but not being attached to sensor plate to allow the cover plate to rotate at operation speeds. The sensor plate must be grounded; this can be done by connecting it to a grounded. The grounded wire will also be used to connect to the Arduino Uno in order to collect the electrical field information. A 9V battery will be used to power the motor and it will be secured by soldering the cap to the motor. Once the mill is assembled the motor and plates can be tested for compatibility.

Alternately, Team Revolution’s designer is constructing a plan in a 3-D program which can be printed on the 3-D printer and machined by the Space Grant students in Pueblo. Using this design, the team will need to assemble the mill by soldering certain hardware, attaching the motor and motor axle mount, attaching the battery, and grounding the sensor plate with the wire which will also be used to connect to the Arduino Uno.

**Housing.** The housing houses the motor and the blades and is one of the 3 main components of our electric filled mill. Originally we planned to send our 3-D design to our colleagues in pueblo to be machined out of aluminum, but this proved to be problematic because it was going to be too expensive, and cutting a block of aluminum to meet our requirements was going to be too difficult.

We decided to use the Community College of Aurora’s 3-D printer to print the housing of our electric filled mill. Although we were successful in doing this, there was one problem, our printer uses plastic material as its it printing material and we need our housing to be made out of some sort of metal for it to be conductive and effective at measuring the change in electric field potential. To solve this we wrapped our EFM housing with aluminum tape.

**Prototype one.** The base of our first prototype is a cylinder of 10 cm diameter with one closed end and one open end. The cylinder will house the sensor and shield plates, the axle mount to the motor, and the ground wire attached to the sensor plate. On the closed end of the cylinder is a smaller cylinder, intended to be housing for the motor. Building housing for the motor will increase stability during the balloon flight. There is also a conical shape connected to the larger cylinder and the motor housing with supports inside that will either be machined together or soldered by the team. This is also to provide more stability for the electric field mill during the flight.
Testing. Our first design containing support beams on the inside was impractical, because the beams were thin and it would prove difficult to weld aluminum to fit our design. Our first prototype also took up a lot of space, and was structurally unstable.

Prototype two. Our second prototype used the same basic design as the first prototype, but the support beams were removed and replaced by a cylindrical bottom. Unlike our first prototype which had a cylindrical top and a conical base, our second prototype has a cylindrical top, a conical middle and a cylindrical bottom.

Testing. Team revolution second prototype was very robust, but weighed too much, and one of the main concerns of our experiment is the mass limit.

Prototype three. Our third prototype was just a cylindrical structure, with holes at the bottom. The function of these holes is a place where our wire from the Arduino is threaded through to the sensor plate of the electric field mill.
Testing. Our third prototype carried over the robustness of prototype two, but was still too heavy, and the way it was printed from the 3-D printer made it unstable. This prototype also exposed the motor to cold air which would most likely cause our motor to stop working.

Since our housing was not going to be made out of aluminum any more, we were no longer going to use the logic that the metal will give it stability and structure; therefore our final electric field mill design must be structurally stable and resolve the problem that our previous prototype had.

Final prototype. For our final prototype, we decided to use the same design as prototype 3, but to move up the bottom cylindrical base, so the motor mount is underneath our base. The placement of the base will allow the payload to be shielded better from the extreme conditions in the atmosphere.

Testing. We tested our final EFM prototype; the new design solved all the problems we had from the previous prototype. It was structurally sound, stable, and weighed half of prototype three. This prototype protects the motor from the cold air, the upper part of the EFM will be exposed to the atmosphere by placing it in a hole cut out of the payload and the new structure limits the motor’s exposure to cold air by moving the base up. The motor and all other components will also be heated preventing them from getting too cold.
**Blades.** The sensor and shield plates have also been designed using the 3-D program and are designed to have 12 wedges cut out at 15° each and a 1.5 cm hole through the center for the motor mount. These plates have been designed with a rim on the outer edge to increase stability and reduce the chances of warping during operation and flight.

Luckily, our Space grant colleges in Pueblo agreed to construct our blade designs for us. Initially we designed 3 blades, one with just two sectors cut at 90° degree angles, the second has three sectors each cut at 60° degree angles, and the third had twelve sectors at 15° degree angles each.

The only difference we expect from these blades is that more sectors will expose and cover the sensor plate faster, thus a smaller the sine wave graph will be created as compared to a blade with fewer sectors at the same amount of time.

Our space grant colleagues constructed two of the blades, the 2 and 3 sector blades; we presumed that the 12 sector blade was not constructed because it can prove very difficult to cut 12 wedges out of a circular aluminum plate leaving little to no room for error.

---

**Payload Design**

The payload will be built using foam core, hot glue, and aluminum tape. The sensors, camera, and electric field mill will be secured using zip ties attaching them to the heated isolation chamber. Holes will be cut into the foam core to expose the camera lens, electric field mill, and switches to the outer surface.

Our payload is to be designed in such a way that it will withstand the extreme temperature and low pressure at an altitude of 30 km. It must be properly insulated and should maintain a temperature above -10°C; this will be achieved by installing a heater component that will be in operation during the flight ensuring that all electronic components and sensors remain above -10°C. The payload will also be built in such a way that it is unaffected by the devastating conditions caused by the balloon’s burst and landing. All of the components will be tightly secured and the insulation will be strategically placed in order to protect these components. Specifically, the motor, the Arduino, and any other components that have a specific operating temperature will all be isolated in the construction.
design to share a compartment that is a heated chamber, allowing for as close to optimal operation temperatures as possible. All of our experiments will also be arranged into our payload in a way that the mass is equally distributed and the center of mass can easily be identified.

**Functional Block Diagram of our System**

![Functional Block Diagram](image)

Top and side view of the inside of our payload designed using the Alibre 3-D computer software
Flight Stimulation Testing

The functionality tests will allow Team Revolution to monitor the efficiency of the individual sensors and the entire system before attaching them to the payload for simulation testing. Once the components are all in place, Team Revolution will make sure that all of the components work together properly. When the system is proven to work efficiently, the simulation tests will ensure that the payload and experiment will withstand the balloon flight. The drop and stair tests will simulate the landing of the payload; these consist of dropping the payload from two-stories high and down flights of stairs. To simulate the extreme temperatures in the stratosphere the payload, with all systems operating, will be placed in a cooler with dry ice for several hours. To simulate the conditions of extreme force after burst, the payload will be expected to operate while rotating on a rope at high speed with a directional change. These simulation tests will determine whether all components of the payload will be successful during the mission. Any faults in operation will be identified and repaired prior to launch.

Drop test, whip test, stair test. These tests all proved that our payload will perform successfully at protecting our experiment. After we performed the drop test, whip test, and stair test, our payload remained intact.
Cooler test & Results. We placed our payload into a cooler that contains dry ice, and then we closed it and let it run for two hours. After two hours, we took out our payload and noticed a couple of problems. First our heater circuit was disconnected, therefore was not providing heat to the payload, this could be fixed easily. Second when we viewed our data log from the Arduino, we found that that the internal temperature got as cold as -5°C, this confirmed that our Arduino and EFM can work at this temperature.

We also noticed that the 9 volt battery that powered our motor died, at about 56 minutes into the cooler test and failed 64 minutes short of what we required. We wanted to make sure that our battery didn’t stop because of the -5°C temperature inside our payload, so we used a Multimeter to test the voltage output of the battery and found it to be 1 volt. From this we decided to use one additional battery to power our payload, but even then we were 8 minutes short, and adding a third battery to power our motor was not ideal because of our mass constraints. To solve this problem we are going to add a resistor to our EFM circuit, specifically we are going to add a potentiometer.

The above image explains how the potentiometer will help ensure that our electric filed mill runs for the entire flight.

We are planning on using one battery and a potentiometer, which we calibrated to be the same resistance as the motor, and we expect this to double battery life. If we increase the resistance on the potentiometer a little higher than the motor this will increase battery life; because it will cause the motor spin slower, we will gather even more accurate readings.

Since we mainly care about the data we collect while our balloon is ascending, i.e. before burst, which we estimate to be about 110 minutes, if our motor can last for that long, then we do not need to add a second battery.

Stimulated electric field test & results. Before we fly our electric field mill we are going to simulate an electric field, and will be using this as our control. The electric field will be simulated by covering cardboard sheets with aluminum foil and placing our EFM in the middle of the cardboard sheet using a hole that is cut. The two sheets represent a parallel plate capacitor, being linked up to a voltage source and modify its capacitance using these power generators and changing the distance between the plates to get the strength of field we want. We are then going to run our Arduino program and plot the data we collected on a graph. We will perform this test many times to determine what sample rate is idea for a smooth sine wave.

We expect our data to look like a sine wave because our EFM, specifically the two blades, the sensor plate and shielding plates will act like a shutter. The shielding blade alternately exposes and covers the sensor plate as it
Testing the Electric Field Circuit of the Earth

rotates, thus creating a sine wave effect; when it is aligned with the sensor plate, the voltage potential is the lowest, and when the sensor plates are exposed, i.e. not aligned with the rotating plate, the voltage potential is at its highest.

Discussion

Our plan for this experiment is to build and launch an electric field mill which will be used to collect data about the electric field in the atmosphere during a high-altitude balloon launch and recovery. The purpose of this experiment is to determine the changes in Earth’s electric field due to altitude, temperature, humidity, and pressure. Data collected in this experiment can be used to chart the global electric circuit and may benefit studies on global climate change.

Launch and Recovery

The Colorado Space Grant students from Community College of Aurora will be launching their payloads on Saturday April 13th, 2013 on Eaton, Colorado. Our payloads will be attached to a helium balloon, along with a GPS system that we are going to use to track and retrieve our payloads. After we have retrieved our payloads, we are going to analyze the data collected by the sensors and EFM and stored to the SD card.

Related to NASA

Since we can predict weather based on temperature humidity, and pressure, which also affect the electric field; therefore we can use the electric field to predict weather as well. One of our main goals with this experiment is to make an electric field that is simplified, light weight and inexpensive. This is of importance to NASA and researchers interested in the Electric field of the Earth because light weight experiments reduce cost, for most high-altitude and space launches cost is directly related to mass. Another reason is because our EFM is inexpensive and versatile and can be built and used by high school students or anyone who desires to do a similar experiment.

Final Electric Filed Mill

Our final electric mill design used the 2 sector blades as the rotating and sensor plate, this is because it will be much easy to optimize our Arduino code to get a much smoother sine wave with a less complicated blade. For our housing we are using our final prototype design, which will be wrapped with aluminum tape to make it conductive. For our 11,000 rpm motor, we have attached a potentiometer to it, for the purpose of increasing the battery life that powers it.

Although we have not performed an electric field stimulation test, we believe that we will need a voltage amplifier. This is because on sheer speculation there is a chance that the electric field potential might not be strong enough to be detected by our electric field mill relative to its connection to the Arduino Uno.

Our final Electric Field Mill weighs 152.5 grams; when broken down into individual pieces, the blades(42g), housing(37g), motor(27.5g), one 9 Volt battery (46g) for a total of 152.5 grams.

Conclusion

Team Revolution has constructed a payload for high balloon launch experiment, and after performing many different testing, such as the drop test, and cooler test, our payload design is physically structural enough to survive the physical aspect of the flight, such as lift off and burst. The test also showed us that our payload is mechanically sound enough to survive the extreme conditions of our atmosphere, such as regions that can get as cold as -80⁰ Celsius.

Even though our EFM has not been flown yet, nor has I be fully tested for functionality in terms of what we want it to achieve, we are confident that our final product will be able to measure the electric field potential of the earth from the Troposphere all the way up to the Stratosphere. We expect the data collected by the Arduino Uno to look like a sine wave, and we will confirm this when we perform our stimulated electric field test. Since we have tested our Arduino Uno along with the sensors many different times, we expect them to be fully operational.
One can argue that our EFM might not be ideal in terms cost, because we had a 3-D printer, and nice friends that were willing to construct aspect of our EFM for us, although this is true, our EFM is still a much better alternative as to purchasing a professional EFM, in terms of cost and mass. Of course a professional EFM will most likely have a more accurate reading than our EFM, or a better tolerance level, although depending on the experiment, the mass of our EFM alone makes up for it.

Acknowledgements

We like to thank Dr. Victor Anderson for accepting to teach this class and give CCA students an opportunity to be part of the Space Grant program. For also overseeing all of our experiments and shedding light were needed. We also extend our appreciation to Tara Croom, for helping with a really important role for this class, which is ordering our required parts for our experiments. We also like to thank Brian Sanders for helping us with the Arduino coding, and teaching us the electrical components of our system as a whole. Finally we like to thank our space grant colleges in Pueblo for manufacturing our EFM blades for us.

References


Appendix

Arduino code

/*------------------------------------------------------------------------*/

Name: BalloonSatDataLogger.ino

Date: 5/24/2012 Initial

Description: Analog reader to capture 3 analog and 3 acceleration readings.
Additionally an OneWire Thermal Sensor is read from digital pin 3. If OneWire temp
sensor is enabled, the max data sample period is ~100ms.
Reading are taken and logged to SD card at rates defined below.

Revision:
1.0 05/24/2012 Initial
1.1 06/06/2012 Added OneWire Sensor support.
1.2 02/26/2013 Removed VRef =3.3 and returned it to normal 5V reference
1.3 03/20/2013 Editorial clarifications
1.4 03/26/2013 Revised for Balloon flight Progress Asoluka

License: Share and Share a like, Creative Commons 3.0
http://creativecommons.org/licenses/by-sa/3.0/

Author: Colorado Space Grant Consortium 2012

Credits:

--------------------------------------------------------------------------*/

//--------------------------------------------------------------------------//

// Configuration and toolbox setup
// External Libraries for writing to the SD Card
#include <SdFat.h>
#include <SdFatUtil.h>

// Include files for One Wire Temperature Sensor
#include <OneWire.h>
#include <DallasTemperature.h>

// Controls for the data logging system
#define LOG_INTERVAL     200   // milli seconds between entries
#define ECHO_TO_SERIAL    1   // echo data to serial port, 0->, for flight, 1-> for development and ground
#define SYNC_INTERVAL  1000   // mills between writes to the the SD card.

// Time keeper
uint32_t syncTime = 0; // time of last sync()

// Defining the usage of the SD card
Sd2Card card;
SdVolume volume;
SdFile root;
SdFile file;

// Defining number of items to log and where the sensors are plugged into the Arduino
// 1 Temp, 1 Pressure, 3 Accel and 1 OneWire
const unsigned int numberOfSensors = 5;

// Pin Name and Pin number on the Arduino Analog Port
const unsigned int thermo0 = 0;
const unsigned int pressure = 1;
const unsigned int humidity = 2;
const unsigned int mill = 3;

// Names for each thermal sensor
DeviceAddress oneWireThermo0;

// Data wire is plugged into pin 3 on the Arduino
#define ONE_WIRE_BUS 3
#define TEMPERATURE_PRECISION 12

// Setup a oneWire instance to communicate with any OneWire devices (not just Maxim/Dallas temperature ICs)
OneWire oneWire(ONE_WIRE_BUS);

// Pass our oneWire reference to Dallas Temperature.
DallasTemperature sensors(&oneWire);

// Titles for column headings. Should map to: sensor0, sensor1, ...
String analogSensorNames[numberOfSensors] = {
"Thermo0 (Deg C)", "Pressure (PSI)", "Humidity (HumiditySensor RH)", "Mill (V)", "ThermoExternal (Deg C)"};

// Temporary data variable used while reading the analog voltages. 0-1023 maps to 0-5V
float data = 0.0;

// Define an LED pin used to show active logging of data
const int LEDwritingStatusPin = 7; // Flashing LED on this pin indicates a good save to the SD card
int LEDwritingStatus = 0; // Flips on/off/on when a valid vectors are received
void setup(void)
{
    Serial.begin(9600);  // Define the baud rate when talking to the computer
    Serial.println();    // Start new text on a new line
    pinMode(10, OUTPUT); // To make the SPI (microSD card) interface work Set pin 10 to an Output.
    pinMode(8, OUTPUT);  // Define the Chip Select pin for the SparkFun SD card
    initializeSDCard();  // Initialize the SD Card and insure it is readable.
    pinMode(LEDwrittingStatusPin, OUTPUT); // Set activity LED pin as an output

    // Setup the Sensor Bus. If more than one sensor is required, follow this pattern
    if (!sensors.getAddress(oneWireThermo0, 0)) Serial.println("Unable to find address for Device 0");
    sensors.setResolution(oneWireThermo0, TEMPERATURE_PRECISION);
}

// Run Many times - the main code

void loop(void)
{
    //Get OneWire sensors while waiting for the log interval
    sensors.requestTemperatures();

    waitForLogInterval();  // Wait for a log interval to take a new batch of data samples

    //Read 1st sensors
    data = analogRead(thermo0);
    data = (data*5.0/1024-0.500)/0.010;  //Convert from TMP36 to Engineering Units (Deg C)
    file.print(','};
Testing the Electric Field Circuit of the Earth

file.print(data);
Serial.print(’,’);
Serial.print(data,2);

//Read 2nd Sensor
data = analogRead(pressure);
data = (data*5.0/1024);  //Convert to Voltage
data = (data - (0.1*5.0))/(4.0/15.0);  //Convert from Counts to Engineering Units (Psi)
file.print(’,’);
file.print(data);
Serial.print(’,’);
Serial.print(data,3);

//Read 3rd Sensor
data = analogRead(humidity);
data = data*(5.0/1024);
data = (data - (5.0*0.16))/(5.0*0.0062);
file.print(’,’);
file.print(data);
Serial.print(’,’);
Serial.print(data, 3);

//Untested EFM code
data = analogRead(mill);
data = data*(5.0/1024);
file.print(’,’);
file.print(data);
Serial.print(’,’);
Serial.print(data, 5);

//One Wire Temp sensor
float tempC = sensors.getTempC(oneWireThermo0);
file.print(',
file.print(tempC);
Serial.print(',,
Serial.print(tempC, 2);

//End this current record by starting a new line
file.println();
serialEndRecord();

//Write to the SD card and if a write to SD is necessary then change the LEDwritingStatus
if (writeDataToSD()) {
    //Change the State of the writingStatus LED from 0-->1, or 1-->0
    ++LEDwritingStatus;
    if (LEDwritingStatus > 1) {
        LEDwritingStatus = 0;
    }
}

//Update the activity LED to show a successful write cycle
digitalWrite(LEDwritingStatusPin, LEDwritingStatus); // Change the LED state

Name. BalloonsatDataLoggerSupport
// store error strings in flash to save RAM
#define error(s) error_P(PSTR(s))

void error_P(const char *str)
{
    PgmPrint("error: ");
    SerialPrintln_P(str);
    if (card.errorCode()) {

PgmPrint("SD error: ");
Serial.print(card.errorCode(), HEX);
Serial.print(',');
Serial.println(card.errorData(), HEX);
}
//Nominally the program would fail here. Attempt to recover
delay(100);
Serial.print("re-init SD card");
initializeSDCard();
delay(100);

}

void initializeSDCard()
{
    // initialize the SD card
    if (!card.init()) error("card.init");

    // initialize a FAT volume
    if (!volume.init(card)) error("volume.init");

    // open root directory
    if (!root.openRoot(volume)) error("openRoot");

    // create a new file
    char name[] = "LOGGER00.CSV";
    for (uint8_t i = 0; i < 100; i++) {
        name[6] = i/10 + '0';
        name[7] = i%10 + '0';
        if (file.open(root, name, O_CREAT | O_EXCL | O_WRITE)) break;
    }
if (!file.isOpen()) error("file.open");
Serial.print("Logging to: ");
Serial.println(name);

// write header
file.writeError = 0;
file.print("millis (ms)\n");
#if ECHO_TO_SERIAL
Serial.print("millis (mS)\n");
#endif //ECHO_TO_SERIAL

// Need to redo this for the proper column headings

for(int i = 0; i < numberOfSensors; i++)
{
labelHeading(analogSensorNames[i]);
}
file.println();
#if ECHO_TO_SERIAL
Serial.println();
#endif  //ECHO_TO_SERIAL

if (file.writeError || !file.sync()) {
error("write header");
}

void waitForLogInterval()
{
    // clear print error
    file.writeError = 0;
}
delay((LOG_INTERVAL -1) - (millis() % LOG_INTERVAL));

// log time
uint32_t m = millis();
file.print(m);
#ifdef ECHO_TO_SERIAL
  Serial.print(m);
#endif //ECHO_TO_SERIAL
}

void serialWrite( unsigned int data)
{
#ifdef ECHO_TO_SERIAL
  Serial.print(',');
  Serial.print(data);
#endif //ECHO_TO_SERIAL
}

boolean writeDataToSD()
{
  //Write the Data to SD card
  if (file.writeError) error("write data");
  //don't sync too often - requires 2048 bytes of I/O to SD card
  if ((millis() - syncTime) < SYNC_INTERVAL) return 0;
  syncTime = millis();
  if (!file.sync()) error("sync");
  return 1;
}

void labelHeading(String headingName)
```cpp
{
    file.print(" ");
    file.print(headingName);
    #if ECHO_TO_SERIAL
    Serial.print(" ");
    Serial.print(headingName);
    #endif //ECHO_TO_SERIAL
}

void serialEndRecord()
{
    #if ECHO_TO_SERIAL
    Serial.println();
    #endif //ECHO_TO_SERIAL
}

/// Support functions for Thermal Sensors

// function to print a device address
void printAddress(DeviceAddress deviceAddress)
{
    for (uint8_t i = 0; i < 8; i++)
    {
    // zero pad the address if necessary
    if (deviceAddress[i] < 16) Serial.print("0");
    Serial.print(deviceAddress[i], HEX);
    file.print(deviceAddress[i], HEX);
    }
}
```