Colorado Space Grant Consortium

DemoSat
Design Document

Written by:

David Garrett	Budget, Results and Analysis, Compilation/Editing
Amber Bekkali	Design, Management, Test Plan and Results
Donovan Anderson	Ready for Flight, Conclusions and Lessons Learned
Anthony Saccomanno	Mission Overview, Requirements Flow Down
Sergio Chacon	Expected Results, Launch and Recovery

August 22, 2014

Table of Contents

1.0 Mission Overview

The objective of the mission was to measure muon flux with respect to altitude. With the data received, we examined the relationship between muon flux and altitude and thus observe the relativistic effect on ultra-fast particles.

Einstein postulated that objects moving close to the speed of light experienced time differently relative to a non-moving object. This effect is confirmed in our project. A previous UNC project also determined these effects by measuring the differences in muon flux at Greeley, CO and at Berthoud Pass, CO (roughly 2000 meters apart). Their results are as follows:

- Rest lifetime of a muon: 2.2 microseconds
- Flux at Berthoud pass : 21531 muons per minute
- Lorentz factor: 6.76
- Hypothesized muons at Greeley using classical physics: 1005 muons/min
- Hypothesized muons at Greeley using relativistic physics: 13684 muons/min
- Actual muons at Greeley: 13542 muons/min

By using a Geiger counter and Arduino at various altitudes, we hoped to discover results similar to the above experiment. However, our experiment would involve a greater gain in altitude (about 100,000 feet,) revealing even more about the relationship between muon flux and altitude.

Our secondary goal was simply to learn about the process of building the demo-sat and pass down knowledge to future UNC teams.

2.0 Requirements Flow Down

Level 0

- The satellite shall include a way of storing data from the Geiger counter
- The satellite shall include an enclosure capable of sustaining and preserving the Geiger counter, data, and Geiger counter-data interface.

Level 1

- An Arduino was used to convert the voltages used in the Geiger counter into readable values
- The Arduino also read altitude levels from an altimeter
- All the materials would be kept in a box for safe enclosure

- The enclosure of the Geiger counter would be created using PVC pipes
- The temperature would be regulated using heating elements, a Mylar blanket, foam core, and fiberglass insulation

Level 2

- E6000 was added to the electrical components to prevent arcing
- The values (altimeter and Geiger counts) in Arduino were converted into a CSV file to be imported into an SD card (which would be read in excel after the data was collected)
- Choosing where to place the components and putting the flight tube through the center of mass reduced turbulence during flight

3.0 Design

We used basic foam core purchased from JOANN'S Fabric. Using two of the same size foam cores (after being cut, each side and bottom) we layered the foam core and glued the two pieces together with E600 (glue). We glued the module together with JB Weld and let it dry. After it was dry we put a layer of fiberglass insulation on the inside walls of the module. We glued this in with gorilla glue. Finally we were ready to cover the box. On the outside, we used aluminum tape and on the inside we put a space blanket and taped it in. We used the same procedure for the top except we did not use insulation. Once the box was complete, we measured the size of the PVC tube to go through directly in the middle. After it was measured, we cut a hole in the module(centered in the middle of bottom) so the PVC tube would go through. On the bottom, we used a metal washer, and under the washer, we JB Weld-ed it to the box. This is so it is permanent and does not move.

Testing glues

We tested multiple glues with the foam core and let them dry. The next day we tested them in liquid nitrogen to see which one held up the best. Our findings were that the E600 and JB Weld worked the best

Heating System

We went through many different schematics of batteries for the heating system. We found the most efficient is two sets of 4 in series and those put into parallel. This is built with six 1K ohm power resistors in parallel and eight 9V batteries.

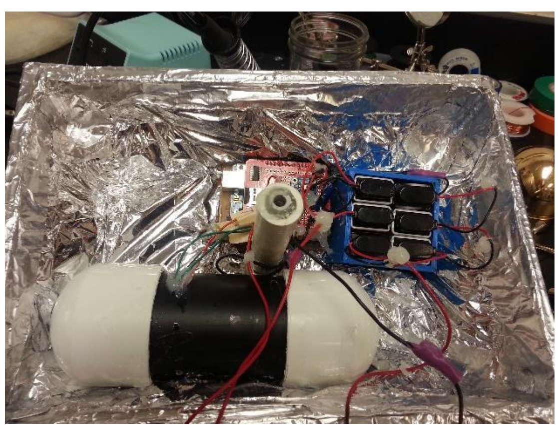
Geiger Counter & Barometer

We came up with a working code for both systems. We did the math and found that we needed four batteries in parallel to be able to sustain three hours of flight.

Heating Box

We wanted to keep the batteries together in a temperature controlled environment. We made a small box that fits six 9V batteries standing up-right. The box does not have a top, but holds the batteries compacted together. This will help sustain the heat of the batteries with the heating system close by. The box was made by using a 3D printer provided by the physics department. Exact dimensions coming soon.

Box	**We bought 3 large pieces of foam core: $6**
	We used E6000 glue, gorilla glue, and jb welding glue: $15
	We also used fiber glass insulation
	We then made the box by double layering the foam core for insulation as well as adding a layer of fiber glass insulation.
	After the basics of the box was finished we coated the box with aluminum foil tape only on the outside because it tended to be thicker and heavier (we founc this at our school in the stone table lab)
	For the inside we used space blanket and glued it in so that we would be sheltered from the fiberglass insulation while working with the box in the future.
Hardware	We used a Geiger counter from spark fun (SEN-11345 which was $149.95)
	We connect that to a Arduino Uno which would be used later on as the retrieving device to send to storage which was a 32 gb mini SD card.
	We connected a barometer to the Arduino as well which gave us the measurements of the temperature and the altitude via pressure.
	We enclosed the Geiger counter into a PVC tube to avoid and conflict with the high voltage areas when there would be a change in pressure.
	Where ever there were any open soldered areas we used either silicone glue o electrical tape to cover them up to avoid them shorting out.
Heater	We used six 1k ohm 10% power resistors and soldered them in parallel on a bread board.
	We then connected that to four 9V batteries that were in series
	We used an on/off switch to control when the heater was turned on and off.
	All of this fit very nicely into the box!



The Geiger counter and the barometer were both connected directly to the Arduino. This allowed us to store and process minimal amounts for a short period of time so that after 15 min it would be transferred to the SD card which was also connected directly to the Arduino. The heater was mainly used to keep the batteries warm during the flight so that all the electronics could keep running smoothly. The heater also took out the probability that any of the electrical components would react badly with the drop in temperature because it kept the box at a pretty consistent temperature.

Our design just barely made the weight limit. We were a little concerned that we would be overweight, but after we decreased the amount of batteries we needed, we measured in at 1.474 kg before we left. The dimensions of the box came out to be roughly 9 inches by 12 inches by 5.5 inches.

4.0 Management

Hardware	Amber and Donovan
	They were in charge of making sure everything was soldered correctly.
	They were also in charge of making sure the box worked well and the heater was working at the right temperature. This included testing many different ideas and choosing the best.
	This also included making the right design of the box for all of our components and making sure they fit in the box nicely.
Software	Sergio, David, and Anthony
	This included coding the Arduino and learning how the Arduino, barometer, Geiger counter, and SD card would all fit together.
Testing	Everyone

Everybody was working on different things at the same time so we were testing our materials on July 14th, two weeks before the launch. These tests included: the drop test, the swing test, the stairs test, and testing all the software by riding the elevator up and down, going on a hike, and using lead (Pb) to try and block out the muons for a specific time to make sure everything was working correctly.

We made sure everything was working by July 28th, the Monday before the launch.

5.0 Budget

5.1 Money/Parts

Part (Quantity)	Cost	Seller
Sparkfun Geiger Counter (2)	$300 ($150 each)	Sparkfun.com
9V Batteries (10)	$100 ($10 each)	UltralifeCorporation.com
Arduino Uno	$25	Sparkfun.com
MicroSD Shield	$15	Sparkfun.com
Adhesives	$30	Ace Hardware
Adafruit BMP180 (3)	$30	Adafruit.com
Foamcore, Fiberglass Insulation, Space Blanket	$30	Ace Hardware/Joann Fabric
PVC (Flight Tube, Geiger capsule)	$20	Home Depot
Rocker Switches	$20	RadioShack
Power Resistors	$0	UNC Physics Stockroom
TOTAL	$570	--

The most expensive parts were the Geiger counters. Having two of them might seem extraneous, but ordering a backup Geiger counter three weeks prior to launch proved to be a crucial move. Our first Geiger counter stopped working the week of the launch (Apparently, Sparkfun Geiger counters have a poor reliability track record and are prone to failure without apparent reason--future teams should investigate alternative options.) When we called Sparkfun about ordering a new one, we found out that they were out of stock. Having a second Geiger counter handy saved our mission from catastrophic failure.

Things like wires and soldering equipment aren't listed since they were already in-house.

Also worth noting, and not listed in the table, is the travel cost associated with the launch itself. Future teams should be mindful of it when planning their budget. Gas and lodging for all team members totals a few hundred dollars--almost as much as the rest of the costs put together.

5.2 Weight

Part/Subsystem	Weight
Geiger Capsule (GC in PVC)	~0.4 kg
9V Batteries (5)	~0.3 kg
Arduino Stack (Uno, SD Shield, BMP180)	~.15 kg
Materials (Foamcore, Adhesive, Insulation, Washers, Flight Tube)	~.6 kg
TOTAL	~1.45 kg

The precise numbers here were not available at the time of writing the report.

6.0 Test Plan and Results

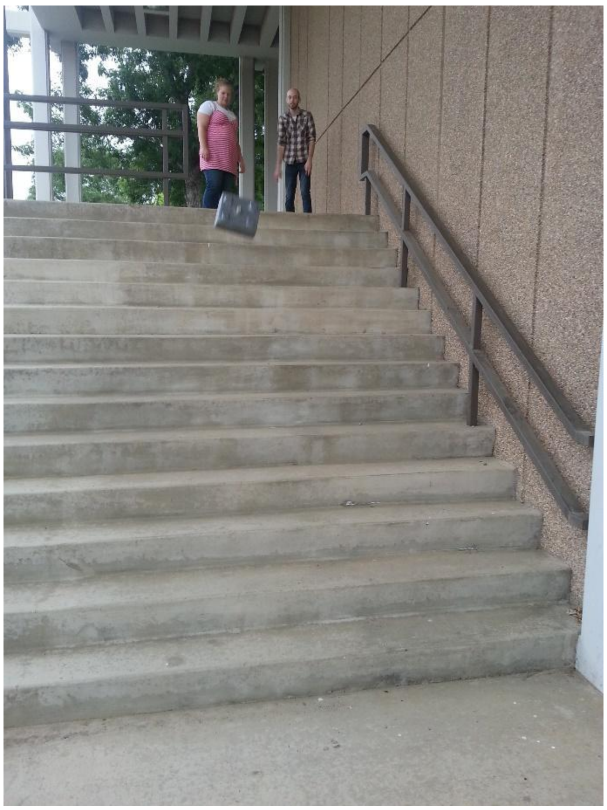
We built a test module exactly like the one above. This time we glued in the smaller pvc tube into the larger one with silicon glue so that the flight string will fit in comfortably. The only thing different is that we used two washers, one on the top and one on the bottom so that it was permanently closed. We added weights to our test module to accurately represent our finished product. The final weight of the test module was 1.727kg. We will now use this to conduct several tests.

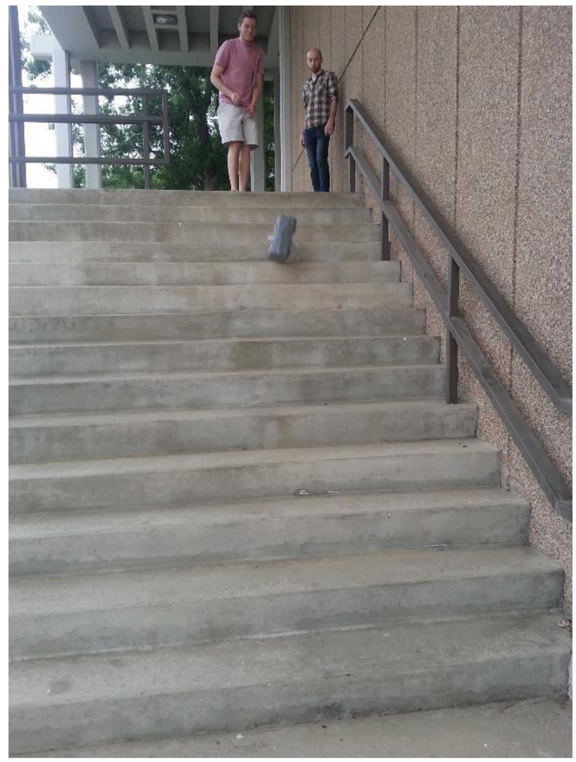
The tests included: the drop test, the swing test, the stairs test, and testing all the software by riding the elevator up and down, going on a hike, and using lead to try and block out the muons for a specific time to make sure everything was working correctly.

As mentioned before, we tested everything from July 14th to July 28th. We designed how we were going to place the different objects specifically into the box and tested the best way by trying to get the center of mass as close to the center as we possible could. When we had that, we glued in all the components. After we glued in all the components and determined where we would put the switches on the outside.

The test box was built due to suggestions in the Demosat User's Guide. Actually, because our module was structurally identical to a previous UNC team's, we just took that old already-built box and used it for some of the testing. It had already been gutted of most parts, so we attached weights to it to model the weight distribution. We then proceeded with the Drop Test.

The drop test: 4 stories of it. That's us on the roof and that's the test box flying (falling with style) through the air.





The Stair Test. Basically, you throw/roll/boll the test box down some stairs. Not nearly as exciting as the drop test, but it does tell you if anything is going to come off of the module when it spins fast.

The Whip Test -- makes sure your flight string tube design is up to the task. Pretty easy, since fully functional suggestions are provided in the User's Guide.

7.0 Expected Results

The expected results in relation to the payload were that it would not only survive the ascent, descent, and landing, but that it would maintain an internal temperature above -10 degrees Celsius. The Geiger counter is to count cosmic radiation as the payload ascends. This data is to be processed and stored into an SD card by the Arduino. The Arduino is to also record the internal temperature of the payload. To test that the Geiger counter and Arduino would properly detect and record radiation count, we had the system detect the radiation in our computer lab. Our hypothesis is that the cosmic radiation will increase as the payload ascends into the upper atmosphere, then level off as the payload nears its apex. This is based on muon flux data gathered in previous experiments (independent of UNC,) as reported to us by one of our faculty advisors, Dr. Matt Semak.

8.0 Launch and Recovery

8.1 Prelaunch

For the launch, Amber Bekkali will be the one that will be launching the payload, and we will be retrieving the payload as a whole team. Once we arrive at the location where the payload lands, we will not open the payload. Instead, we will bring the payload back to the car or lab before retrieving the data. We prefer to do this because we didn't want to carry the equipment necessary to read the SD card out into the field. Once the box is retrieved and taken back to the lab the data that is stored in the SD card, will be transferred to a laptop. If all went well the Arduino should have formatted the data into an excel sheet so that after the data is retrieved all we will need to do is make a graph of muon counts versus altitude. In testing, recovering the data from the SD card was easy and always worked well.

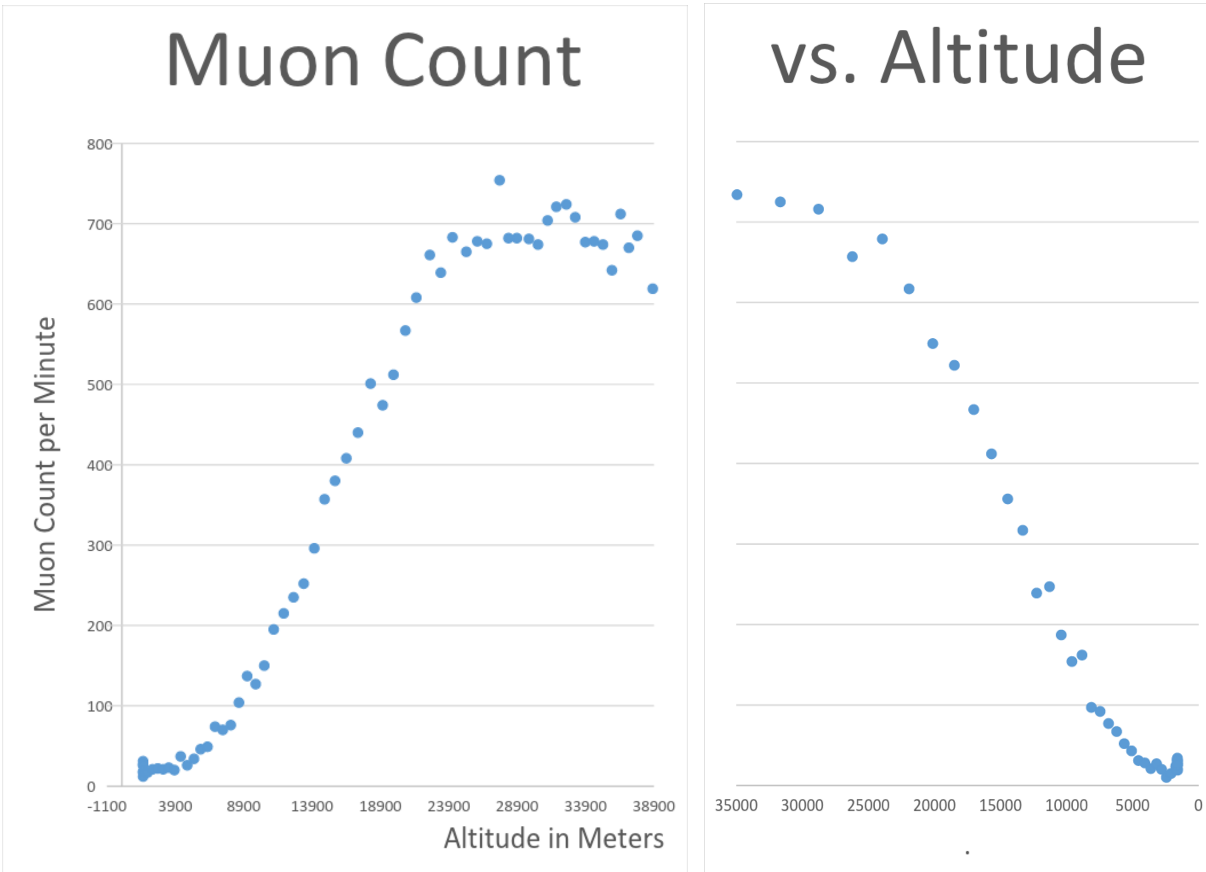
8.2 After launch

After the recovery of our payload, there seemed to be no external or internal damage, and once we removed the SD card from the Arduino, we reviewed the data. It was found that all the data was taken and stored properly. It was convenient that the Arduino formatted the data into a spreadsheet because all we had to do was make a graph of muon count versus altitude. The data taken by the Geiger counter showed us that the counts per minute of cosmic radiation increased as the altitude increased as seen in the graph above. The Arduino also recorded the internal temperature of the box which revealed to us that the internal temperature remained well above our minimum. With all the data collected, and there being no damage done to the payload, we considered this a successful launch.

9.0 Results and Analysis

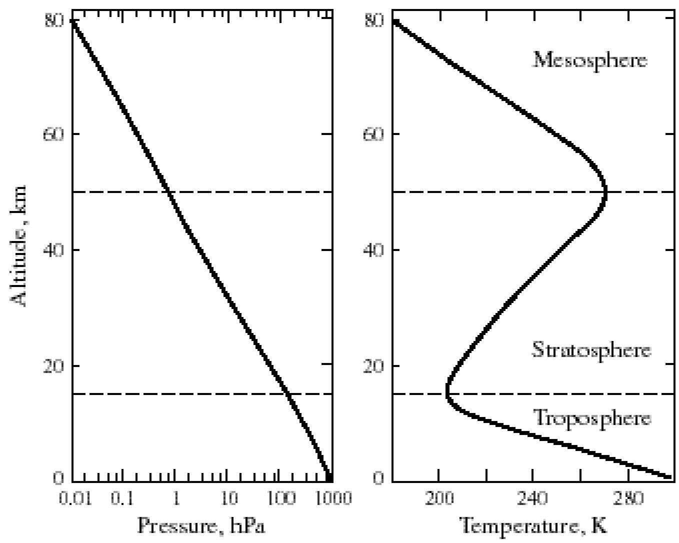
The flight data in full can be viewed here. There are four columns in the spreadsheet: Altitude, CPM, Temp, and Pres. Altitude is measured in meters above sea level. CPM is short for Counts Per Minute, and tells you how many ionization events happened in the Geiger Counter in that minute. This is taken to be a measure of muon flux, since almost all events are caused by muons when a radioactive source is not present. Temp is temperature, measured in Celsius. Pres is Pressure, measured in mmHg. Each row in the spreadsheet represents a one-minute interval.

On the left of the spreadsheet is the ascent data. On the right is descent data. This separation is actually coincidental. It was intended for all data to be collected sequentially and uninterrupted. This was not the case, as there appears to have been a power down/reboot that occurred at some point during the flight. Why this happened, we're not sure. However, it didn't interfere with our data very much, as it was only at the very top of the flight that it happened.



The above graph (two graphs, technically) shows the results of the experiment. The data points agree well with our prediction, which was based on previous data independent of our organization.

Also of interest in our data is the temperature. Temperature appears to have reached an initial low of 16° C at 37,700 meters. This is above our target of 10° C, which was required to keep electronics fully functional. During the reboot incident, temperature seems to have returned to 20° C, which is due to the fact that the atmosphere is actually warmer at an elevation of 35 km than it is at an elevation of 20 km (shown in the graph below, which is taken from this site.) During descent, temperature is seen to drop again to a temporary minimum of 13.3° C, and then rise to a maximum of 42.8° C, or about 109° F, demonstrating that the heater was still operational at that time. (Ambient temperatures on the ground were only about 70° F.)



We also aided some high school students involved in UNC's FSI (Frontiers of Science Institute) program in sending up a module with a video camera. The videos from that folder can be found in this folder. Said folder also contains pictures taken on the ground during testing and after the flight.

10.0 Ready for Flight

Our first flight was very successful. We were able to collect the data and video from both the FSI payload and our main payload. The structure endured the flight and was safely retrieved after the duration of the flight. The box also maintained a sustained a temperature that allowed all of the units in the internal structure of the payload to function properly. So, for the future flights we have kept the heating system the same. We did all of the tests for the payload that were required before we could do the first flight. These tests were successful and proved to be useful as our design allowed for a successful data gathering. We would like to have a better strategy for our camera, so it does not need to be on before being in the air. (A method to turn on the camera before launch) As we wasted the camera battery life before the BalloonSat was in the air. Before our second flight we would also like to make our payload more unique on the outside by maybe coming up with a better design, painting, or customizing the outside. However, this is not a high priority because this will not affect the actual payload. We will not be coating the Geiger counter for the next flight as it ruined the one before two days before launch. (We assume it was the insulator-black coating that cause it to malfunction) One of our team members were able to program the other one and get our payload ready for the flight. We will store our payload in the physics department storage lab, while we wait for the second flight. We will allow future teams to see what we did and what we accomplished. To activate the payload, all you need to do is turn on both of the switches. This activates all the parts on the inside of the payload. If our payload is used after six months for another launch, be sure to use new batteries, check the programming for the Geiger counter, and make sure all wires are still attached.

11.0 Conclusions and Lessons Learned

Our team has learned quite a bit from this year's Demosat program. We learned how to functionalize and improve a heating system by wiring parallel and series resistors. We were able to test different designs and find which lasted longer and sustained the heat within the payload. We also learned to be patient when it came to testing the test box payload. We had to take time as we ran through each of the required structural and environmental tests. Most importantly, we learned how to communicate and work together as a team. As future leaders, we were able to come together often to work on this project by effectively working together and using each other's strengths for the project. By doing this we crafted a successful payload that was successful for the launch. For next time, I would like to put a camera in our payload and I would like to put in a device to measure the Carbon dioxide in the atmosphere. CSU had a payload that successfully gathered this data, and I feel it would be very doable to do this and still remain in our budget. I feel we can add both the camera and a device to measure carbon dioxide and be in our weight limit. I would also look at ways to improve the overall structural design, as ours was successful but simplistic.

12.0 Message to Next Year

Amber

Do not procrastinate! Things do go wrong. Designs sometimes do not work! We were working our modified design continuously for close to two weeks. We also built our box before we really had any components. I think if I were to redo this project it would have been better to design the box around the material instead of the material around the box.

David

Work on the project at least two times a week. Otherwise, you spend too much time just remembering where you left off.

At the beginning of the project, look over the documents for each design review (Preliminary Design Review, Critical Design Review, Launch Readiness Review) and for this final report. All of these will help you plan your progress in advance.

If you just want to participate and send some kind of thing up, work on the module for a month and a half or two months. But don't be surprised when it's launch day, you're looking at the other schools' payloads, and you're thinking, "I should have put more time into this." If you want to send a *really good* module up, give yourself another month or two. Be ready and willing to build rough drafts of your box and to modify your design. Get enough materials to build rough drafts, and don't hesitate to buy more materials until you get it right.

Donovan

Our overall payload was a success and we were able to gather all of our information after the launch.There were a few things we wanted to do, but time management was an issue. I would highly suggest for next year's DemoSat, to find a set schedule that works for majority of the team members and have a task list for each member. Be sure to set tasks to individuals who are willing to put in the time and if they are comfortable with the area assigned. This will allow for a successful team and the project will be done before the deadline. It would also be a plus to make sure everyone is communicating with each other. For the payload, I would look into adding a camera as we found that the camera worked very well inside the FSI payload design. I would also look into adding a device to measure something extra like carbon dioxide levels. Our heating system was incredibly successful as it sustained the internal temperature of the payload. We also found that using four batteries for the heating system and two batteries for the Arduino worked best. (Total of six batteries) Be sure to redo all the tests and make sure the payload is ready for launch.