CFP-16- Team Confey Can

A remarkably robust can incorporating Solar power, Two way Communication and a myriad of sensors

Liam Murphy, murphy.liam8@gmail.com, Conor Walsh(conorwalsh100@gmail.com), Julia Was(julia.wasz@gmail.com), Alex Bourke(alexwashereawhileago@gmail.com), Gergely Gelert(gergo.gelert@gmail.com), Naoise Barry(naoise.barry@gmail.com) - IRELAND

Abstract:

The Primary objective of the ConfeyCan Cansat was to measure external temperature and air pressure as the can falls from a height of 1000m, to transmit this data in real time to a ground station and to produce meaningful graphs of temperature, pressure and altitude during and after the can's flight. In addition, the Secondary objectives were as follows:

a.) To measure gforces (3D) experienced by the can throughout it's flight and produce meaningful graphs of gForce as a function of both time and altitude

b.) To measure the internal temperature of the can throughout it's flight and compare it to the external temperature as a function of both time and altitude

c.) To provide more accurate pressure (and hence altitude) measurements throughout the flight

d.) To provide GPS data including time, location and altitude throughout the flight

e.) To provide a means of both using and measuring solar energy gathered by the Cansat during it's flight and produce meaningful graphs of solar energy as a function of time.

f.) To measure the Relative Humidity inside the can and produce graphs of Relative Humidity as a function of time and temperature

g.) To demonstrate two way communication between the ground station and the can by (i) sending commands to turn on/off a buzzer located on the can, (ii) requesting the can to broadcast a message that is sent from the groundstation back to the groundstation, (iii) requesting the can to perform a mathematical function and return the answer.

As a result of poor visibility and windy conditions on the day, ConfeyCan was dropped from a plane from a height of approx 410m. Although communication was lost during three periods, our two way communication worked and all sensors gathered data successfully. This data was sent back to the groundstation where it was graphed and analysed. Our results tended to agree with theory and there were no major anomolies. Our temperature data suggests that a temperature inversion occurred on the day of launch. This manuscript will discuss the ConfeyCan Cansat project in terms of materials and structural design, relevance of missions, calibration and testing, programming and error detection . It will provide details and discussion of scientific results, lessons learned and an overall conclusion.

I. INTRODUCTION

Altitude is an essential piece of data for any satellite. The formula used to calculate altitude requires both air temperature and pressure readings and hence the Primary objective of the ConfeyCan Cansat was to measure temperature and air pressure as the can falls from a height of 1000m, to transmit this data in real time to a ground station, remove erroneous data and produce meaningful graphs of temperature, pressure and altitude during and after the can's flight.

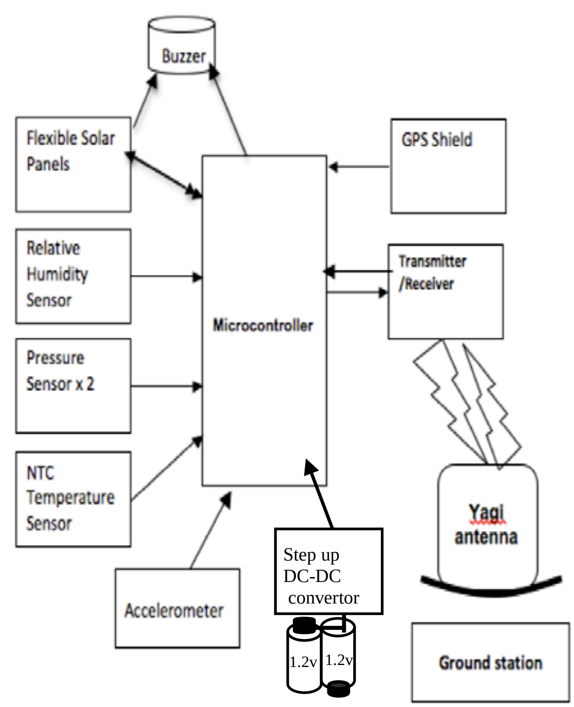
Our Secondary mission additions are explained as follows:

As it is important when designing a satellite to understand the size of the gForces experienced at all stages of flight to ensure future launches are more robustly catered for it was decided to measure gForce. Upon reading about the AIM mission to determine what factors lead to the formation of noctilucent clouds, and also since water is a precursor to human life, it was decided to measure Relative Humidity. We added a more sensitive Pressure sensor and also decided to measure internal temperature. We added GPS location since it is important to have the corresponding location information for each sensor reading and also it allowed for location of the Cansat on landing. To aid location in the event of the Cansat landing in an area where it might not be visible, a solar operated buzzer was also added. This buzzer could be turned on or off from the groundstation at any time before/during/after flight using two way communication. If the Cansat was to be used after landing then it would need a way to recharge its batteries so flexible Solar panels were added to demonstrate both using (to power buzzer) and measuring the solar energy gathered by the Cansat during and after it's flight. It could be important for a Cansat to broadcast messages to other Cansats or to a planet so an additional two way communication function was added to allow the groundstation to transmit a random message to the Cansat and have the Cansat broadcast that message back again. A further two way communication function was also added whereby the Cansat could perform a multiplication function for the groundstation.

II PROJECT DESCRIPTION

Initially, an NTC thermistor was used to measure external temperature and the MPX4115A sensor was used to measure pressure. Two APC220 transcievers were used to transmit and receive data between Cansat and groundstation. The ADXL345 3D accelerometer was used to measure and transmit the gforce data experienced by the can throughout it's flight. The HIH4030 Relative Humidity sensor was used to measure Relative Humidity.. The BMP180 Barometric sensor was added (accuracy = 0.02mBar) as an improvement to the MPX4115 Pressure sensor whose accuracy was +/- 15mBar. The BMP180 Pressure sensor requires temperature in order to calculate altitude and it has a built in Temperature sensor. This provides the internal temperature readings and the input temperature values for the HIH4030 Relative Humidity sensor which also requires temperature for it's calculation of Relative Humidity. The Thermistor, being located just on the outside of the can gives readings that are slightly lower than the temperature of the internal sensors – as our data below shows so if we used this temperature value it would cause an error in the Relative Humidity readings . The DSS circuits I2C GPS shield was used for location as was a buzzer that could run off low solar power. This buzzer could be turned on or off from the groundstation at any time before/during/after flight using two way communication. This worked by turning on and off a mosfet that was placed in between the solar panel and the buzzer. Powerfilm MPT 4.8-75 flexible solar panels were used to demonstrate both using (to power buzzer) and measuring the solar energy gathered by the Cansat during and after it's flight. The solar power was calculated by measuring the voltage it created across a 1000 ohm resistor and using the formula : $P = V^2/R$..

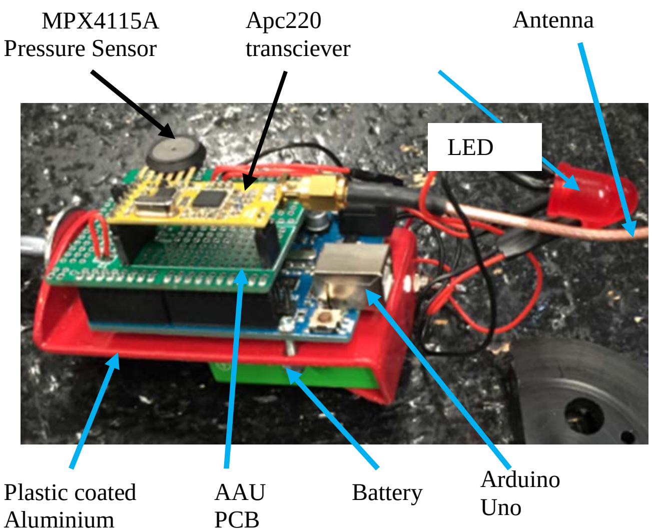
.



II.I Materials and structural design

For the Primary mission we used the standard Irish Cansat kit which included, primarily, the

Arduino Uno, the AAU sensor PCB, an APC220 wireless communication module for radio transmission, a NTC thermistor, MPX4115A pressure sensor and aluminium frame to mount everything onto. The outer casing was made from polycarbonate and the dimensions exactly matched the Cansat requirements of length 115mm and diameter 66mm.. The frame was bent to allow for the 9v battery to fit.

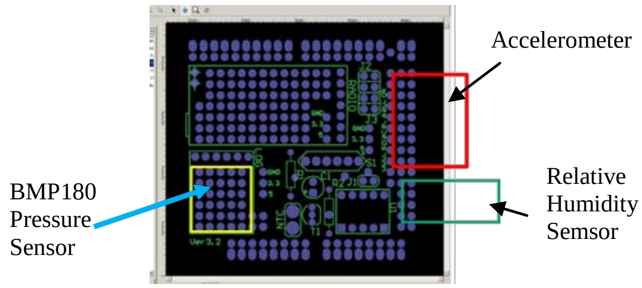


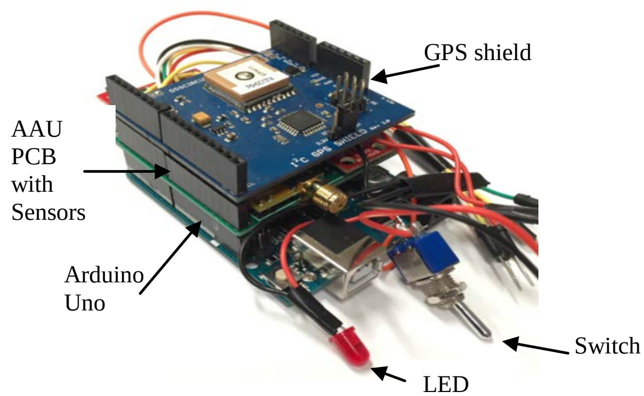
We used coaxial cable for our thread antenna which was connected to the APC220 via a SMA cable. Our cable was cut to reveal a length of coax that was one quarter wavelength. This was calculated using the formula:

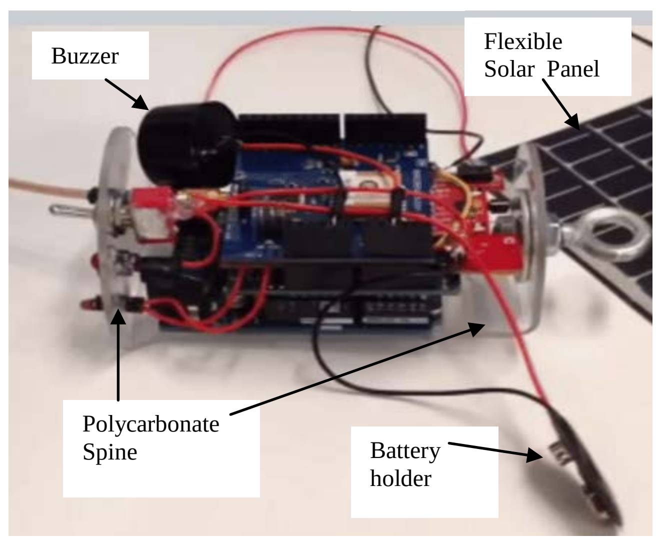
$$L = c/4f = (3x10^8)/4(444 \times 10^6) = 0.169m$$

We modified our initial Cansat as more functionality was added but the basic structure did not change that much. All sensors were soldered to the AAU sensor board. After our regional finals we changed to a thinner polycarbonate spine to reduce signal loss and weight and improve flexibility. The GPS shield was positioned above the AAU board as we found that the signal quality was reduced when it was placed underneath. The locations of the various sensors on the AAU board are indicated in the diagram below. The buzzer circuitry also managed to fit on the AAU board while the buzzer itself was glued to the casing. We had an LED to indicate power on and our flexible solar panel wrapped almost fully around the can on the inside. The Cansat structures were made entirely in our school using school engineering equipment.

AAU PCB New Sensor Locations







II.II Powering the Cansat:

Initially we used a 9V 500mAH battery connected to the Arduino onboard regulator. However, we knew we would get a maximum of 4 hours running time on low current. Since the running can needed 125mA we knew we would get less than that. Experimentation showed we were actually getting approximately 3 hours. Research suggested that the Arduino onboard step down regulator was inefficient and testing showed the efficiency to be 40%.

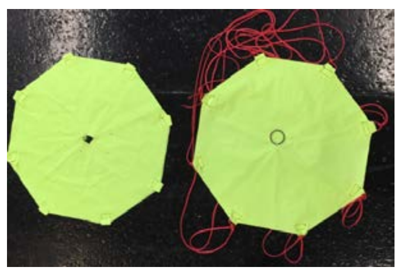
We decided to use two rechargeable 2400maH batteries in conjunction with a micropower step up DC-DC boost convertor/regulator (NCP1402) to power the Cansat. The step up regulator (79% efficient by experiment) allowed us to bypass the less efficient Arduino onboard regulator. We calculated that this new setup would theoretically give us 8 hrs running time compared with the theoretical 4 hrs when using a 9V battery connected to the on board regulator.

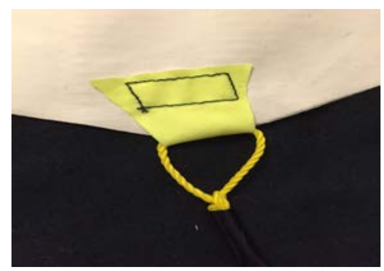
II.II Calibration:

The NTC and MPX4115 were calibrated by an ISO approved local calibration company Butlertech. We calibrated the BMP180 temperature sensor in school using a hairdryer, freezer and an accurate temperature probe. The Relative Humidity sensor was calibrated in a local Institute of Technology (DIT) lab using a calibrated hygrometer in sealed container inside which the relative humidity could be varied. The accelerometer was calibrated using the fact that the vertical gForce should be 1g when the can is stationary. The can was placed in all three positions so that the X,Y, and Z axes were vertical and the gForce for the vertical axis recorded each time. The offset between 1g and the reading obtained in each case was incorporated into the Python program and subtracted/added to the Accelerometer results ensuring the data was accurate.

II.II Parachute:

Terminal velocity depends on Air Density, Parachute Area and Drag Coefficient. Air density was assumed to be constant at 1225Kg/m^3. Drag coefficient value depends on parachute shape and the Area of parachute was calculated by equating drag force and gravitational force, taking into account the mass of the can and the shape of the parachute and using a terminal velocity of 8m/s. Initial testing of both a flat octagonal, a hemispherical and a cross parachute showed our flat octagonal design to provide the velocity closest to 8m/s and it was also the most stable.. We then did drop tests using a drone to determine the optimum material to both give the parachute the required terminal velocity and stability. Tensile tests were carried out to determine the optimum cord material and cord attachment method. We found rip stop nylon to be the best parachute material and 1.5mm diameter twisted synthetic twine to be the best cord material. We also carried out drop tests to address parachute stability testing a variety of central hole sizes. On the day of the final, since it was very windy, we used the largest hole size of 3cm diameter. During the test launch our parachute spun in a circle as it fell so we purchased a swivel and added this to our parachute ring to improve stability. Our parachute was fluorescent yellow as this was easy to spot.

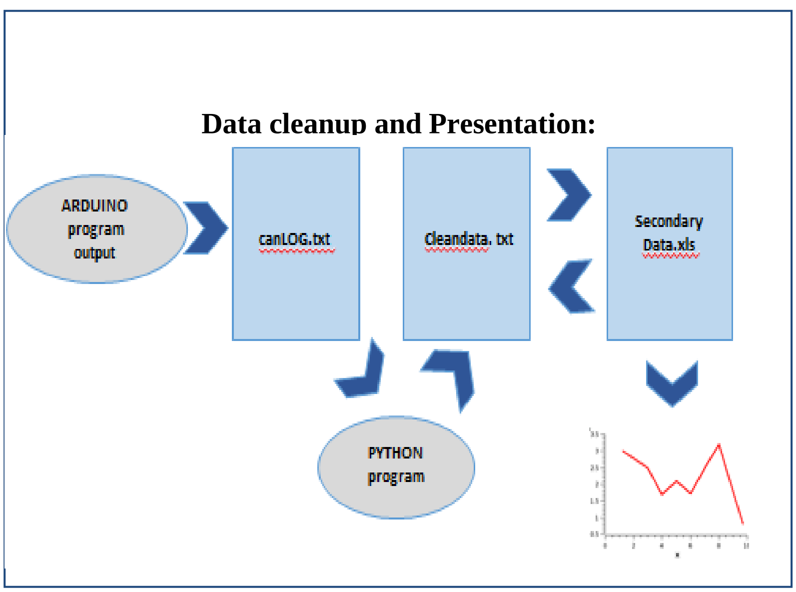




Cord attachment technique

II.II Collecting the Data

Due to the fact that the Arduino Uno only has 5 Analog Inputs, three sensors with I^{2}C capability were chosen. Using I^{2}C protocol however does increase power consumption and reduce clock speed. To minimise Arduino power usage and improve speed, calculations were kept to a minimum within the Arduino program and data was transmitted to the ground station in as raw a form as possible. For ease of editing and debugging, the Arduino program was organised mostly into functions which were called from within the main loop. As mentioned earlier, data was transmitted via the APC220 transceiver using a quarter wavelength thread antenna. To investigate the best method of receiving data a Yagi antenna and a Parabolic Dish were tested over distances greater than 1000m and the Yagi was shown to be far superior for distances > 1000m. The program RF Magic was used to program the APC220 transceivers to a frequency of 444 MHz and a bit rate of 9600bps. The ground station used the terminal emulation program 'Terminal v1.9b' to communicate with the Arduino. Through Terminal, the data received from the Arduino could be stored in a raw data file. A Python program was then written to tidy up the raw data, remove any corrupt data, perform necessary calculations on the data and save the new clean data to a 'cleandata' file. The data coming from the Arduino included a checksum number (simply a total for all the sensor values being sent) which allowed the Python program to filter out corrupt lines of data. Calculations performed on the data included: altitude above take off ground level based on MPX4115A and BMP180 pressure sensors, calibration adjustments, resultant gforce, vertical velocity, solar power, time in hrs, mins, secs, msecs. The raw data file remained untouched in case of an error occurring during the running of the Python program. An Excel file 'SecondaryData' was set up to point at the data in the 'cleandata' file and all x/y scatter graphs of interest were set up so that, on refresh, the graphs for a new set of data were instantly visible and could be analysed and incorporated into other presentations easily. On the day of the final we were getting a line of data approximately every 0.6 secs.



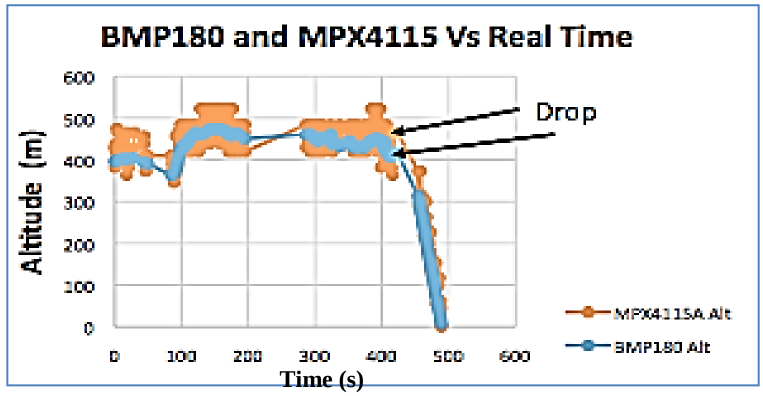


Fig 1: Altitude Vs Time for BMP180 and MPX4115

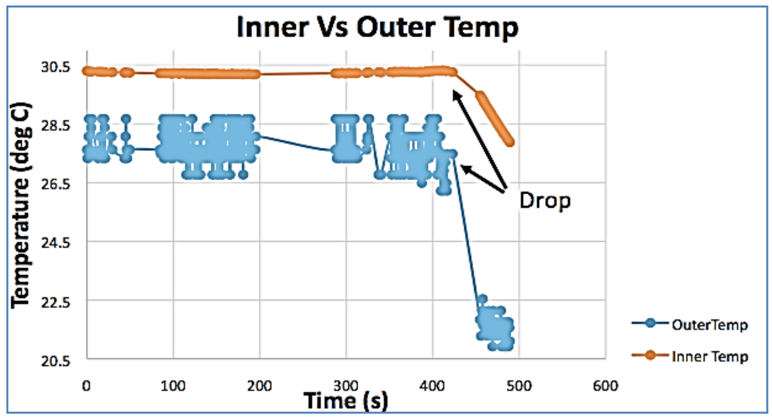


Fig 2: Internal Vs External Temperature

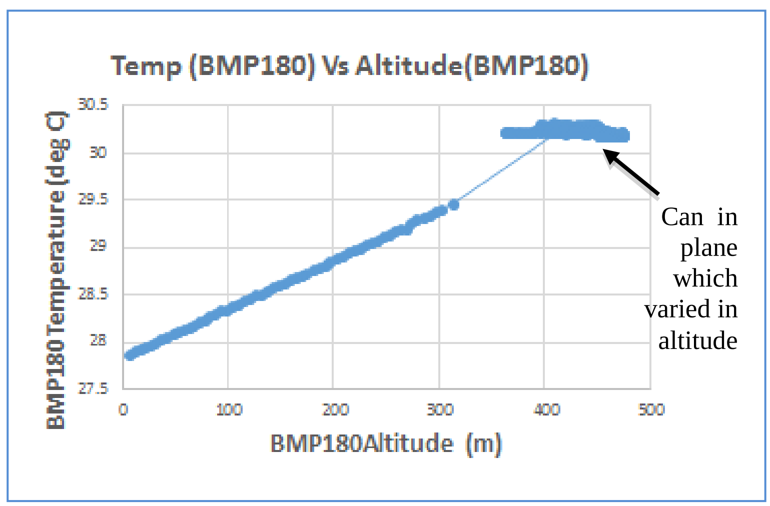


Fig 3: BMP180 Temperature Vs Altitude

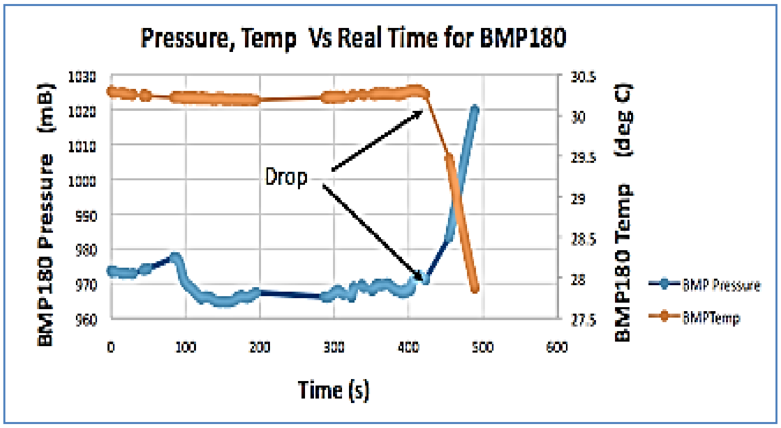


Fig 4: BMP180 Pressure and Temp Vs Time

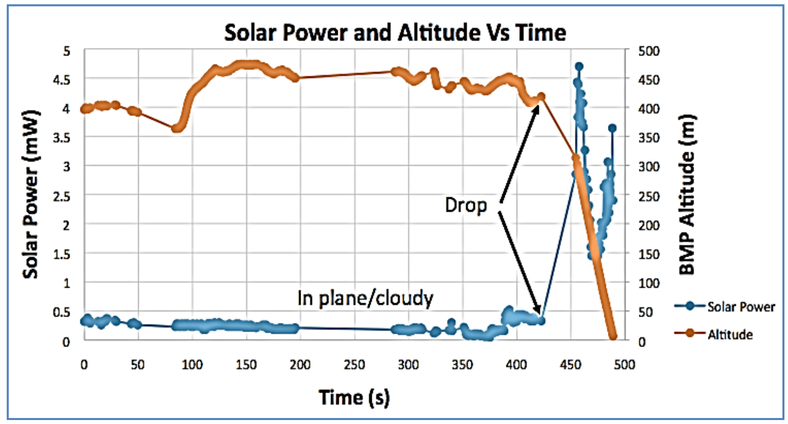


Fig 7: Solar Power and Altitude Vs Time

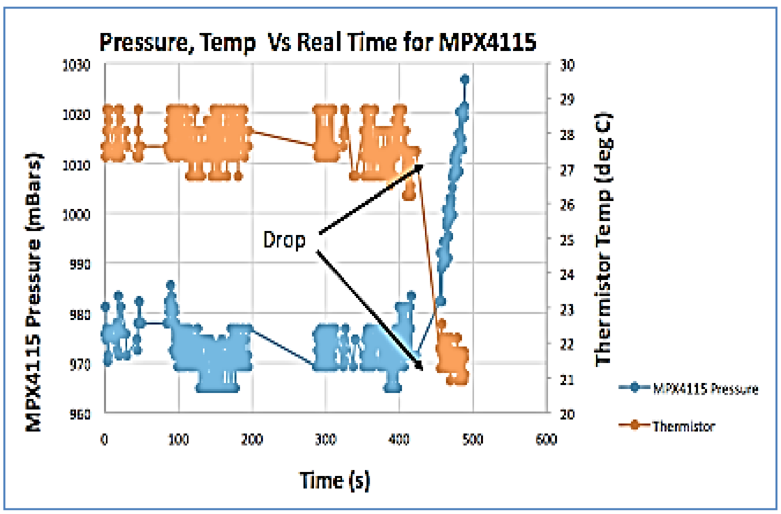


Fig 5: Pressure and Temp Vs Time MPX4115

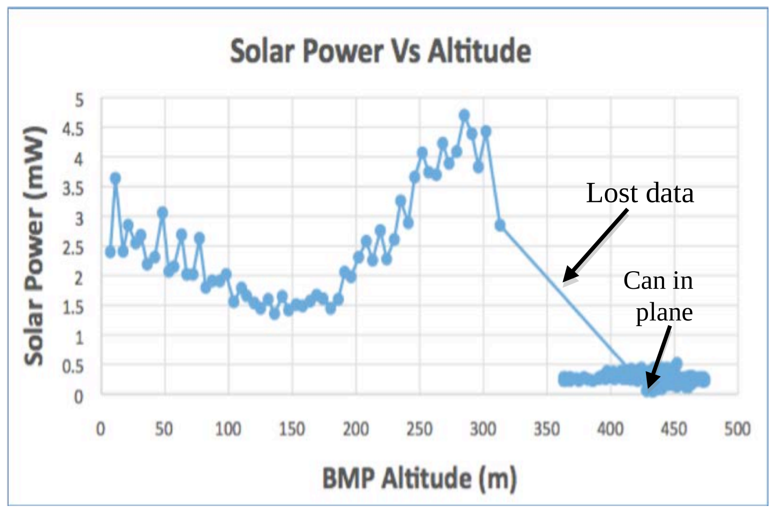


Fig 8: BMP180 Pressure and Temp Vs Time

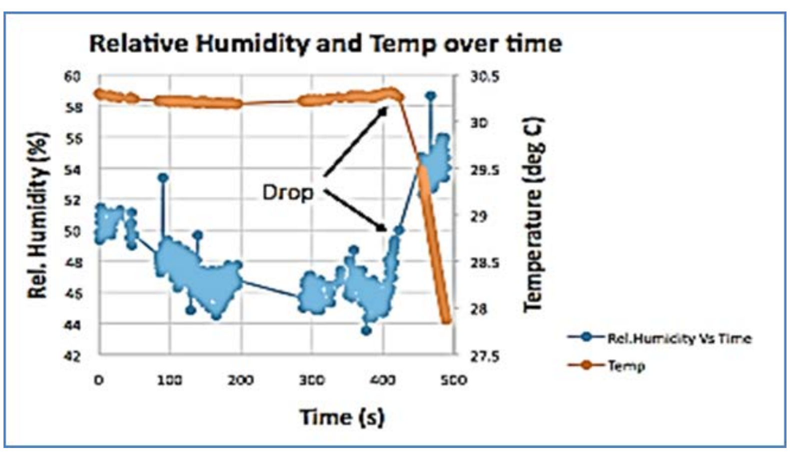


Fig 6: BMP180 Pressure and Temp Vs Time

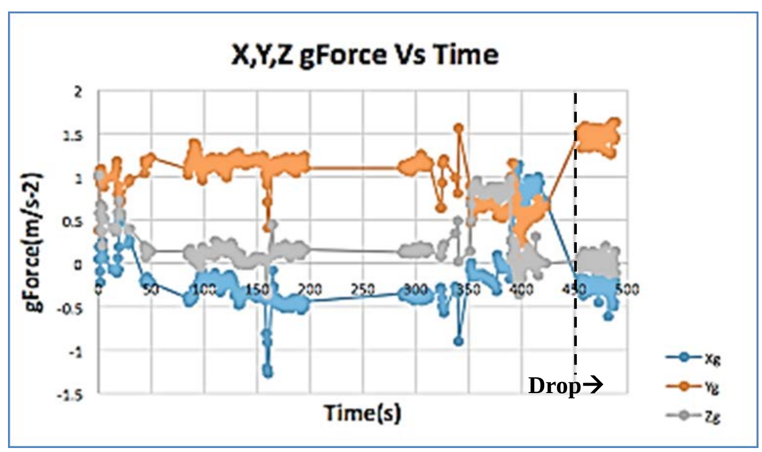


Fig 9: X,Y,Z gForces Vs Time

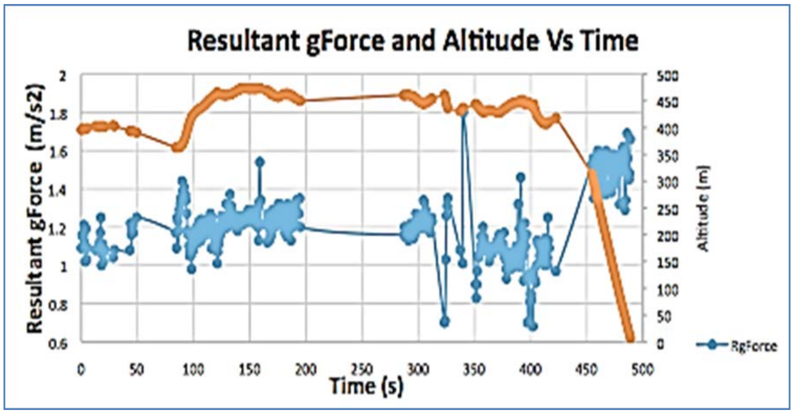


Fig 10: Resultant gForce, Altitude Vs Time

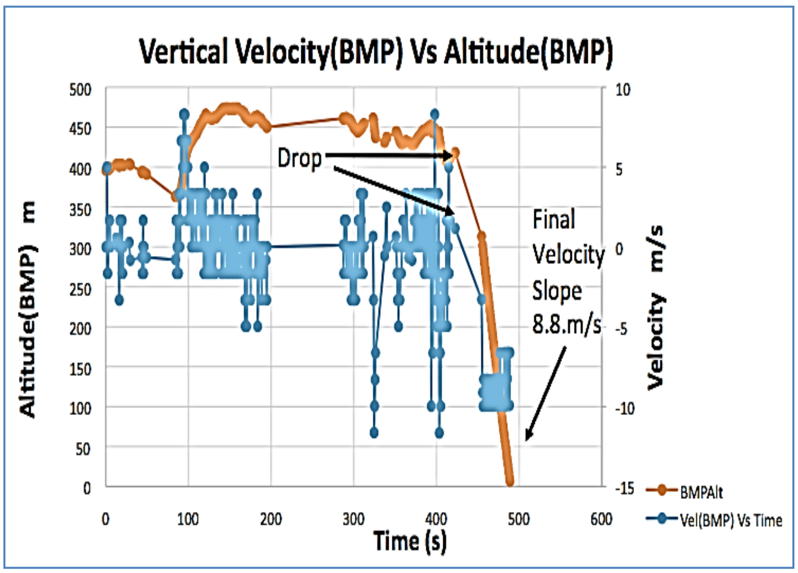


Fig 11: Vertical Velocity Vs Altitude (BMP180)

IV DISCUSSION

As mentioned, due to unforeseen very windy and cloudy weather conditions, ConfeyCan was dropped from a height of approx 410 m from a plane instead of 1000m from a rocket. The drop was delayed from 12:30 to the much later time of 8:10p.m. From approx.. 5:00p.m. onwards we were at the drop site awaiting the drop. We therefore did not get nearly as much time as we would have liked to analyse our data, compare with the results of other groups or to investigate local parameters which may have contributed to our results. Due to the fact that the plane took off from the aerodrome which was several km away, we did not start collecting data until the plane came within range. This meant we had to adjust our python program as it had been programmed to measure height from take off by subtracting the initial height reading. However, we managed to produce plenty of interesting graphs.

The first thing to note is that we have three timeframes where we lost data. On two occasions this was because the plane went well beyond the 1000m as it flew back and forth trying to find a suitable opportunity to drop the can. The third loss of data occurred either at or close to ejection from the plane. We lost data for 32 secs at this time. It may have been related to how the can was put into the drop pipe on the plane but we never managed to get exact details from the flight crew.

Looking at Fig 1 we can see that the BMP180 sensor is indeed far more precise in it’s calculation of Altitude than the MPX4115A. However both sensors agree generally about the variations in altitude and drop time. According to the BMP180, the drop occurred from a height of 411m.

Similarly in Fig 2 we can see that the BMP180 temperature sensor seems to be more precise than the thermistor although this could also be due to the fact that the thermistor is located at a hole on the side of the can and could be more susceptible to wind effects or even more responsive to temperature changes. We also can see from this graph that there appears to be a difference between internal and external temperature of approximately 3 deg C with the inner temperature being hotter which is as expected due to the emission of heat from electrical components. This difference in temperature is constant until the Cansat drops and then we see that the thermistor appears to be affected more dramatically by the change in altitude and the difference between the inside and outside is closer to 6 deg C. We feel that the reason for this is that the can had poor air flow through it due to adding extra glue for security and having to add extra ballast inside it before launch to meet the mass requirements. Restricted airflow would result in there being not enough time for the change in outside temperature to be transmitted through to the sensors inside the can.

Fig 3 shows a linear increase in temperature with increasing altitude. The flat line of data as marked in the diagram can be ignored as this is when the can was inside the plane and the temperature will have remained constant even though the plane varied it's altitude considerably. This was a surprising relationship because we would expect temperature to fall as altitude increases due (i) to being further away from the hotter earth and (ii) as air expands due to lower pressure it's temperature should fall.

Figs 4 and 5 concur with this unexpected result showing that, for both pressure and temperature sensors, as pressure increases (as Cansat falls) temperature decreases when we would expect the air to contract and the can to get hotter. This is what has always been noted on previous Cansat flights. The best explanation for this is that a ground temperature inversion occurred. This would have been due to the fact that the can was dropped into a hollow area where cold air can spill in from higher areas causing a temperature inversion on the lower ground. By the time we analysed our results we could not find local weather data to confirm this – we should have asked the other teams whether they saw the same effects but did not get a chance..

Looking at Fig 6, we see that Relative Humidity rises as temperature drops. This is what we expected since:

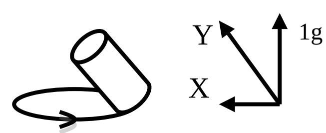
$$\textbf{\textit{Relative humidity}} = \frac{\textbf{\textit{amount of water vapour present in air x 100}}}{\textbf{\textit{Amount needed for Saturation at that temperature}}}$$

As temperature drops, the amount of water vapour needed to reach saturation point drops so the denominator gets smaller and hence the RH should increase as temperature decreases.

Fig 7 and Fig 8 shows how solar power varied with altitude. We can see from Fig 7 that when the can was in the plane there was little or no solar power received despite the plane dipping and rising– this explains the flat line with altitude in Fig 8. However, we can see from Fig 8 that once the can leaves the plane there is a steady decrease in solar power with altitude until the can reaches 150m where the solar power then seems to increase steadily until it reaches the ground. It was expected that the solar power would decrease with altitude . The effect seen here may be a result of the windy weather and the thick layers of cloud that were spread out. It may have been the case that the can was blown sideways out from under the influence of the heavier clouds so that although altitude was decreasing visibility started to improve. We had intended to video the flight and brought the equipment but unfortunately we did not have the visibility from where we were located. It may also be due to the fact that the solar panel did not extend exactly all around the can and there was a small gap where there was no solar panel. Hence, since the can was rotating there might not always have been a constant area exposed to the same amount of light.

Fig 9 shows the X,Y, and Z gForces on the can. The Y axis is in the vertical direction when the can falls so, not surprisingly, we see the Y gForce rise as the can falls. However it seems surprising that the actual gForce in the Y direction is 1.5g as the can falls. This can be explained by the fact that a swivel was added to the can making it spin. As a result the can may have been tilted and rotating:

This would mean at any time the force in the Y direction would be the resultant of the gravitational 1g and the force in the x direction (towards the centre of the circle being marked out):



Calculations show that if the y reading is 1.5g then the X reading should be :

$$\mathbf{X = \sqrt{(1.5g^2 - 1g^2)} \ \ = \ 0.25g}$$

which is close to the value seen in Fig 8.

Fig 10 tells us that the maximum resultant gForce felt by the can was 1.66g as it was falling. We feel this would have been a much more interesting graph if the can had travelled in the rocket as we have graphs from our National final showing how the resultant gForce closely agrees with the various stages of the rocket launch/ejection.

Finally, Fig 11 shows a graph of the vertical velocity and altitude over time. The straight line of the altitude graph shows that the can fell with a terminal velocity of 8.8m/s which was well within our 8 – 11 m/s range.

V CONCLUSIONS

Despite the fact that the Cansat did not get to go up 1000m in a rocket, it is still felt that this project was a huge success. It achieved all of the goals that were set out although we did identify several areas we could have made improvements.We successfully gathered a huge variety of data and have been able to present and explain our results. We have learned that we need to put more importance on gathering local weather/ meteorological data prior to launch and talking to other teams to see if our findings agree. We have identified a variety of ways to improve on our project and have been inspired with many new ideas for future projects. We need to ensure that future missions have enough room within to allow internal sensors to match with external conditions. Initially, we felt that keeping weight of can to a minimum would be advantageous in a Satellite as weight usually costs money in terms of fuel/space but, due to the minimum weight requirement, we had to add n=ballast which took up much needed air space. Hence, we will need to consider using heavier materials in future missions. We would like to look into other methods of receiving data and also making our can more robust as we are still unsure as to why we lost data during the final. Having demonstrated that we can successfully gather and measure solar power we would now like to further investigate the use of solar power to run a Cansat on the ground. We would like to look into controlled landing and a better way to attach and extend solar panels and orienting those panels to maximize solar power received.

We also measure our success by the huge amount of learning has taken place over the entire timeframe of the project. In brief, the major areas of learning would be : learning to work as a team, learning more about each team member, their strengths and about our own strengths and our own weaknesse, learning when to and when not to give up on something, learning that where there's a will there's always a way, to be prepared for all eventualities, to expect things to go wrong and try to anticipate what they might be, learning how an Arduino works and how to program one, learning to program in Python, learning a lot more about Excel, how to Solder, how to plan a project, the relevance of Calibration and how to Calibrate, learning about Relative Humidity, gForces, Pressure, Temperature, Thermistors, Solar Panels, Power consumption and how all of our sensors work, learning to prepare presentations, videos and reports, to stand up in front of peers, teachers and sponsors and talk about the project on a range of levels from the very technical (Xilinx) to the less technical (Primary schools) and everything in between. Throughout the project thanks to optimism and teamwork we always managed to overcome the many obstacles thrown into our path and this has made us better individuals recognizing the importance of being part of a team and the importance of what we are studying in school not only in Mathematics, Computing and the Sciences but also in terms of Languages and Communication Skills. Finally, we have learned how robust our system and it's components actually are in that it has survived a myriad of bumps, resoldering, dismantling, power connection errors – not to mention the fact that most of the components in our European Cansat spent 15 mins under water during an eventful drop test that saw our can drift and disappear into a river only three weeks before the European final

And it still works!!