CanSat Competition 2017 PLR Sections

Section 1: Charles

Section 2: Kaveh

2.1 Mission Overview

2.2 Mechanical and structural design

2.3 Electrical design

2.4 Software design

2.5 Recovery System

2.6 Ground support equipment

Section 1

Team Organization and Roles

Team Member | Role | Background | Responsibilities | Hours Contributed

Secondary mission:

GWC CanSat’s secondary mission covers two main targets:

Initially, the CanSat is intended to monitor several variables in the general atmosphere, this covers ambient temperature, humidity, and barometric temperature. This aims to support existing hypotheses about trends associated with these variables.

We also aim to create heatmaps of the surface below the CanSat. This requires the capability to accurately and precisely monitor the orientation and position of the CanSat, in addition to using precise thermal imaging.

Launch objectives:

In order for a successful flight of our CanSat, we require all the onboard sensors to perform accurately, precisely and reliably. This involves factors such as adequate vibration control, secure electrical connections to the modules, sufficient cooling throughout the CanSat and internal levels around the electronics.

To produce an accurate and precise heatmap, we also require the rotations of our CanSat to be monitored and limited. We have an onboard and accelerometer to allow us to compensate for positioning changes.

In addition, we require a stable radio connection between our base station and our CanSat. To achieve this, we have purchased a yagi antenna and tested (to the best of our ability) the transmit/receive range between the base station and radio module.

Expected results:

We expect to establish clear trends in the ambient temperature, humidity, and pressure. These trends should hopefully support existing hypotheses about the relationships between altitude and the variables.

We also expect to construct a heatmap of the surface below the CanSat, which will allow us to determine any major thermal activity happening in the target area.

What we aim to measure/investigate

Our CanSat is equipped with a range of sensor modules, capable of recording data on the ambient temperature, humidity, barometric pressure, altitude, GPS longitude, GPS latitude and 3-axis rotation, as well as being equipped with a thermal sensor for surface thermal imaging.

This will allow us to determine several general atmospheric trends which we intend to compare with existing hypotheses. We expect no discrepancies, so these experiments will contribute as supporting evidence.

We will also use the thermal imaging feed, GPS positioning, and rotation data to construct a precise heatmap of the surroundings, this will allow us to investigate any significant and/or unexpected thermal activity in the area around the CanSat launch zone. This can be applied for environmental and emission surveys and allow us to further investigate outdoor heat gains, heat vents, and cool zones.

2.1 Mission Overview

In general terms, our team has divided our CanSat goals into two categories. First of all, we’ve designed our CanSat to monitor general atmospheric trends, via temperature, barometric pressure, and humidity sensors.

Our CanSat is also capable of constructing a heat map of the below surface. In order to accomplish this goal, our CanSat needs to be able to precisely track its orientation and position, as well as measure temperature of the ground surface below the CanSat.

We’ve included a high-precision IMU (Inertial Measurement Unit) incorporating a three-axis gyroscope, three-axis magnetometer, and three-axis compass to provide accurate orientation data via a digital prediction filter, as well as a GPS to compensate for drift in the calculated CanSat position. We’ve also included a low-resolution thermal camera facing from the bottom of our CanSat. Through these sensors, the CanSat is able to map incoming thermal camera data to the region of the ground surface from which it originates, allowing construction of a large-scale temperature map as the CanSat drifts over the ground surface.

A Linux single board computer and Arduino Pro Mini allow for redundant data logging and provide enough speed for all necessary calculations to be done on-CanSat, at 10 Hz for environmental sensors and 100-1000Hz for positional tracking sensors.

Data transmission to the base station is accomplished through an XBEE 868LP and GSM radio. The XBEE 868LP serves as our primary method of communication, dynamically updating the base station data store with an update rate depending on packet loss: typically between 10-50Hz.

If communication is lost with the base station due to hardware, software, or packet loss issues, the CanSat will shift to an emergency location tracking mode, reporting hardware or sensor failures as well as GPS position via texts to a team member’s cell phone.

A rough visualization of our CanSat electronics system is shown in figure (FIGURE NAME):

(NEED BLOCK DIAGRAM)

It is also necessary for our CanSat mechanical, structural, and mechanical design to create an environment in which our sensors can operate at peak accuracy. Our CanSat requirements included a durable and shock-resistant design and an unguided descent at 10.5ms-1. We accomplished this via a steel-nylon internal frame, a shock absorbing fiberglass outer shell, and a ripstop nylon parachute.

2.2 Mechanical and Structural Design

Overview:

Our team has made use of a variety of materials and assembly techniques in our CanSat structure to optimize CanSat durability, modularity, and weight. In accomplishing this goal, we split CanSat mechanical design into three discrete sections: internal frame, outer shell, and electronics. Our current CanSat design is available in browser on SketchFab (FOOTNOTE). We’ve also published the current mechanical design to a Github repository (FOOTNOTE)

The development of our final CanSat design has been a long process. To further clarify our design decisions this report will also describe the prototyping process.

To provide context to each section of our CanSat design, CAD-rendered and real-life images of the assembled final and prototype CanSats are shown in figures (FIGURES) (real life photos do not include the outer shell):

(IMAGE HERE) (IMAGE HERE)

(IMAGE HERE) (IMAGE HERE)

Internal Frame:

In order to maximize durability the team decided to split the electronics housing into a tough internal frame and relatively flexible outer shell, attached via a simple mounting mechanism. The goal of this design was for the outer shell to absorb the large forces due to ground collision, while the internal frame provides a structural rigidity and another layer of protection for the sensitive electronics

Prototype phases:

In the development of our internal frame, the team experimented with a variety of materials. As our school stocked 3mm acrylic sheets, we first tested a bent acrylic frame. However, we found that the bend point was lacking the necessary strength for our CanSat, easily shattering on impact. The team then decided to replicate the original design with a bent aluminum frame.

(CAD render of entire frame here)

The aluminum frame provided the required rigidity, strength, and toughness for our prototype design. While the internal frame was still relatively basic, lacking specific PCB and sensor mount points, it fulfilled our design requirements.

As the team had easy access to a laser cutter via our school, we decided to laser cut top and bottom CanSat plates to provide easy mounting points for antennae, switches, and headers.

(Top and bottom plate images here)

The development of the CanSat internal frame can be seen via the bottom frames of our prototype CanSat revisions.

(IMAGE HERE)

Final design:

In the team’s second design revision, we aimed to use additive manufacturing to allow for a more complex internal frame, maximizing strength, weight, and allowing for sensor, PCB, and battery mounting points directly on the internal frame.

(CAD render of entire frame here)

We again elected to split the CanSat frame into top and bottom components. The top frame was composed of three discrete vertical support bars, providing a slider trenches and screw holes for mounting the outer shell, slots for mounting PCBs, as well as battery-holding prongs and parachute/top plate mounting points. These support bars were attached via a laser-cut top plate, again providing switch and header mounting points, as well as holes to allow for parachute mounts.

(CAD render here)

The bottom frame was more complex. The basic frame was designed as a single piece of material, again with outer shell, PCB, and battery mount and hold points. Additionally, the bottom frame includes a mounting plate for an Omron D6T thermal camera, as well as screw holes to allow for add-on modules to be attached to the CanSat. Currently, these screw holes are occupied by a piece of laser-cut plastic providing a protective bezel for the thermal camera.

(CAD render here)

The internal frame was initially designed to be printed from PLA in a consumer 3D printer. However, we found that this did not provide the desired tolerances:

(Image here)

Croft additive manufacturing offered to sponsor the team with an SLS (selective laser sintering) internal frame built from stainless steel:

(Image here)

This construction did allow for required tolerances but lead to near-maximum mass for the entire CanSat, preventing the team from pursuing any further add-on modules. A conductive steel frame also led to radio range reduction in our testing.

The team then ordered a nylon frame to be constructed via Shapeways again via SLS:

(Image here)

A nylon internal frame, while more flexible than desired, allowed for minimal weight while retaining the toughness and tolerances that our design required while remaining suitably affordable. In order to minimize the flexibility of the internal frame, the team has decided to use a steel top frame paired with a nylon bottom frame.

Outer shell:

Prototype phases:

The team experimented with both 3D printed PLA and fiberglass in the creation of our outer shell. We found PLA to lack strength, easily cracking on impacts, while fiberglass proved to be a difficult material to acquire desired tolerances.

For our prototype outer shell, the team decided to use fiberglass molded around a bottle of the required diameter. The shell was then cut to size and provided with team and sponsor logos, protected with a transparent plastic sheet.

(Image of outer shell here)

Spring steel mounts with screw plungers on the internal frame meshed with holes in the outer shell. This design, while offering further shock resistance via the spring steel, created difficulties when removing the internal frame from the outer shell.

Final design:

The team again elected to use a fiberglass outer shell. In order to improve tolerances, a mold was 3D printed out of PLA and covered with aluminum foil before the casting process began.

(Render of mold here) (Raw outer shell image here)

In order to minimize the mounting difficulties found in the CanSat prototype, the team decided on a slider system built from 3D printed PLA to allow for outer shell – internal frame mounting, secured via a set of bolts. While this system would lack the shock resistance of the CanSat prototype the team believed it to be a worthwhile tradeoff.

(Render here) (image of new CanSat here with sliders installed)

Electronics

Prototype phases:

In the team brainstorming session, we decided to mount all sensors and electronics to a vertical PCB or protoboard. While a set of vertically stacked, circular electronics boards would theoretically provide more surface area, no single board would provide enough space for the single-board-computer and batteries that we aimed to use. Additionally, a single electronics board would facilitate assembly and reduce the number of points of failure. In order to provide long battery runtimes, the team also planned to include a pair of 2200mAh lithium-ion cells in the 18650 form factor. The relatively large size of these cells led to a press-fit mount as seen in figure (FIGURE HERE).

The thermal camera was placed at the bottom of the CanSat, against the steel frame, and the cellular GSM module was secured via adhesive above the battery. (ENTIRE CANSAT ANNOTATED VIEW HERE)

Final design:

The team maintained a similar sensor payload in the second CanSat revision, although the use of a new sensors allowed us to condense the pressure and humidity sensors into one unit. A more advanced internal frame design also allowed for easier and more secure battery and electronics mounting.

(INSERT RENDER HERE)

Parts List:

2.3 Electrical Design

Sensor and Device Wiring

Prototype Design

Similar to the section describing the mechanical design, the team’s experiences in our prototyping phase provide context to our design decisions in our final CanSat design.

Throughout our CanSat brainstorming, prototyping, and final design, the team used a Linux single-board-computer as the main control board in our CanSat. One of our team’s objectives was for the CanSat to stay relatively independent of the base station- to be able to do all sensor processing on-CanSat, to compensate for any electrical, radio, or hardware failures, and, if necessary, for the base station to be able to alter CanSat code over a radio connection. These objectives required more computing power than was available in the eight-bit Atmel MCUs common in hobby development boards, so the team settled on a Linux SBC: the CHIP, designed by Next Thing Co, for our prototype design. The CHIP includes a 1GHz ARM CPU, 4GB of Flash, and 512MB of RAM. For this design, sensor payload was as follows in figure (FIGURE NAME HERE):

(FIGURE HERE)

The CHIP provided battery regulation and charging, as well as stable and high-current-capable 5V, 3.3V, and 1.8V rails. We experienced no issues powering sensors and radios, though logic level shifting circuitry was necessary for some sensors.

The team did struggle with serial I/O in our prototype design. The CHIP theoretically supports multiple serial ports, but the team was only able to interface with one port echoing the Linux console. Hence we added an Arduino Pro Mini to serve as a serial-I2C interpreter.

Our prototype design saw these components wired to a single protoboard as seen in figure (FIGURE NAME HERE)

Final Design

As mentioned earlier, the electrical design remained relatively constant between revisions. The largest electrical change was in deciding to design custom PCBs to maximize available sensor mounting space.

With custom PCBs, we were able to include surface mount components. This allowed us to mount supporting GPS, XBEE, and level shifting circuitry directly on the PCB, removing the need for expensive breakout boards. We also decided to use the CHIP Pro instead of the CHIP: while it only included 512MB of Flash and 256MB of RAM, it promised lower power consumption and a PCB friendly footprint. We also used a BME280, replacing the DHT22 and BMP180, and offering humidity, pressure, and temperature sensors.

Fearing radio range concerns the team moved from 2.4GHz XBEE-Pro modules to 868MHz XBEE-LP radios, which offer a range of 5km at 10Kbps. The 868MHz XBEE radios are also surface-mount-compatible, reducing space requirements further.

(PCB PICTURES)

With our new internal frame design the team was able to expand to two PCBs: a primary PCB with control, radio, and some sensor circuitry, and a secondary PCB with a larger array of sensors. We also included redundant level shifting and step-down voltage converters to ensure a reliable system.

As with our mechanical design, the team’s electrical schematics and layouts are available at (GITHUB), though a condensed schematic is shown in figure (FIGURE).

Radio specifications:

Prototype Design:

Our prototype design included two radios: an XBEE Pro 2.4GHz and a SIM800L GSM module. This dual radio design, as seen in the schematic in figure (FIGURE HERE), provides redundancy and assists emergency recovery of the CanSat. If the primary radio link is lost, or CHIP failure is experienced, the CanSat GPS coordinates will be sent via SMS to a team members’ cell phone.

The XBEE Pro is a simple all-in-one radio and modem module allowing for long range encrypted communication over the 802.15.4 Zigbee protocol. At peak bandwidth, the modules are capable of a 250kbps transmit rate in either point to point or multi-point networks. The XBEE Pro consumes 215mA during data transmit and has a maximum range of 1.5km

The team decided to use the XBEE module due to these features. The Zigbee protocol is widely used, making our CanSat compatible with thousands of data logging devices already in use, and the module’s integrated spread-spectrum technology allows for reliable long-range communication in heavy interference. Finally, the modules can allow a transparent serial link between the CanSat and base station: the base station can access the CHIP Linux terminal with packetization handled onboard the XBEE.

The SIM800L was chosen due to its cost, ubiquity, and large feature set. At $6.99 per unit including a breakout board, the module was easy to fit in our CanSat budget. Additionally, the SIM800L provides easy configuration via the well-documented AT command set. The SIM800L consumes approximately 40mA in receive mode, with millisecond spikes of up to 2A while transmitting.

Final Design:

The team decided to continue using the SIM800L in the final CanSat design due to its ease of use. However, the move to custom PCBs allowed the team to utilize the XBEE 868LP: the only other long range XBEE module permitted for use in the European Union, which operates as a spread-spectrum device at 863-870MHz. The XBEE 868LP has the added benefits of lower power consumption at 48mA and a longer maximum range of 8.4km. These characteristics lead to a smaller transmit rate of 10kbps. However, the team accepted this loss due to the smaller packets resulting from rewritten CanSat code.

Battery Specifications:

Our battery choice has remained constant through our prototype and final designs, with a pair of 2200mAh lithium ion cells connected via a load balancer.

There are many downsides associated with the use of lithium-ion cells. Their high energy density can lead to thermal runaway and combustion, damaging the CanSat, and battery performance suffers in cold temperatures. However, it is important to note that the CanSat will be dropped from a height of 1km: too low to see any major temperature delta. Additionally, our prototype CanSat has shown that the outer shell – internal frame provides enough support to protect the batteries. The team has also decided to include foam padding at the bottom of the battery compartment to further prevent damage.

CanSat power usage and estimated runtime:

GPS 20mA

Sensors 10mA total

CHIP Pro 270mA mean

Arduino 15mA

SIM800L 40mA

XBEE 48mA

Inefficiencies in power supplies 50mA

Total 413mA

Calculated 10.65 hrs

However, our CanSat is not always transmitting data, and hence in real-world tests, the CanSat has run for between 12 and 13 hours.

Software Design:

As both design revisions of the CanSat incorporated similar software only the final version will be described here.

On-CanSat

The CanSat software is divided between the main control, data logging, and processing loops running on the Linux SBC, and the control and interpretation loop running on the Arduino Pro Mini. All of our CanSat code is available in a Github repository.

Linux SBC:

The overall program flow is illustrated in figure (FIGURE HERE), though it is detailed further below.

All code used in the Linux system of the CanSat was written in Python, with a heavily customized version of Vim as a development environment. The program runs several tasks in parallel: packetization for radio transmission, data logging and backup, sensor and device control, and large-scale data processing for motion tracking and heat map creation.

The base level of the program occurs inside the i2c-sensors and arduino-interface modules. These modules abstract the raw bus interfaces for the sensors and devices present on the CanSat, enabling simple read functions and easy set functions for sensor power, update rate, and full-scale deflection. The datastore module is built on i2c-sensors and further abstracts the CanSat sensor payload into a variety of different ‘modes’. So far, the team has implemented:

1. min\_power: all sensors and devices aside from the Arduino, CHIP, and radio are disabled

2. all\_active: all sensors and devices are active at full power, update, and deflection

3. envir\_log: only sensors related to environmental data logging (humidity, temperature, and pressure) are active

4. track\_pos: only sensors required for positional tracking (accelerometer, magnetometer, gyroscope, barometer, and GPS) are active

5. heat\_map: only sensors required for heat map creation (accelerometer, magnetometer, gyroscope, barometer, GPS, and thermal camera) are active

These sensor modes provide easy CanSat control, maximizing performance and battery life. The datastore module contains a variety of other features, including the ability to check sensor failure or i2c communication failure via the use of error catching code, and an implementation of functions abstracting calibration routines present in i2c-sensors.

The data collected from datastore and from arduino-interface is also used to calculate CanSat position. This is done via a Kalman filter, which calculates accurate pitch and roll from the accelerometer and gyroscope. The accuracy of this data is further improved via a tilt-compensated magnetometer. Via these filters, the CanSat is able to accurately track its 3D orientation.

However, translational positional tracking via accelerometer integration is prone to drift, and hence bounding sensors are necessary. This is accomplished via the barometer, from which altitude can be calculated, and the GPS, from which x-y position can be calculated to a precision of +-5m.

Once position and heading can be accurately calculated, the CanSat then is able to map data collected from each pixel of the thermal camera to various regions on the ground. This can be done via simple trigonometry and rotation matrices. However, in order to ensure fast update rates, pixels are assumed to represent square regions of a flat ground surface, scaled by the CanSat displacement above the ground. These algorithms are accessible in the CanSat codebase and run in a distinct thread from data logging and radio communication.

With data collected and processed from CanSat sensors, it is prepared for transmission via the packet module. The packet module, after receiving a data request from the control module, collects all required sensor data and assembles them into a packet. The contents of a packet can be arbitrarily requested and can include current sensor data points, stored sensor data points, sensor error data, positional data, and temperature maps. It is also possible to request pre-made packets with data from the modes described in datastore. Packets are also appended with a checksum, which allows the base station to confirm packet validity. Packet structure is detailed in the packet module in the CanSat codebase.

Overall CanSat control is done via the control module, which serves to interpret base station packets, change all necessary sensor settings, and instantiate a module to log data to a text file. The control module also allows for text messaging if many malformed packets are received.

The Linux subsystem of the CanSat receives approximately 26kbps of data. However, much of this data is due to the high update rate for inertial sensors required for accurate positional tracking and is therefore not sent to the base station. Packets are also not sent at regular intervals: the dynamic nature of the CanSat packetization and control algorithms has led to a tested update rate of 2-5Hz over a radio link.

Arduino:

Written in C++ in the Arduino IDE, the Arduino codebase is relatively simple. Via Adafruit libraries GPS position, speed, altitude, and validity is read from the GPS module and split into character lists, which are then transmitted to the Linux subsystem. The Arduino codebase also allows for SIM800L text messaging integration.

Base Station:

The base station codebase allows easy visualization of CanSat data and overall control of the CanSat. It uses two discrete loops, one requesting data from the CanSat and the other graphing said data in a graphical user interface and providing CanSat status information. Via a set of buttons, the base station allows for sensor, power mode, and deflection settings to be constructed in packets and sent to the CanSat. The base station also performs independent data logging of received data. The base station in test mode is visible in figure (FIGURE HERE), and program flow can be seen in figure (FIGURE HERE).

The base station code is written in Java / JavaFX within the Eclipse IDE.

Recovery System:

The team is currently using a simple hexagonal parachute made from ripstop nylon, which has led to a tested descent rate of approximately 10.5ms-1. The parachute is attached via a set of strings that loop within the upper internal frame, as can be seen in (FIGURE HERE) (FIGURE (CAD) HERE).

With an estimated 1000m drop, this design will lead to a flight time of approximately 95 seconds. However, the team is working on implementing a more advanced, custom parachute with a cruciform design to improve CanSat stability during flight.

Ground Support Equipment:

Our base station design is relatively simple, containing only an XBEE 868LP module, an acrylic box with a development board serving as an XBEE mount and USB-Serial converter, as well as an 11dBm Yagi-Uda antenna. The USB-Serial converter is attached to a laptop running the base station software. The assembled base station can be seen in figure (FIGURE HERE).

Section 3:

Resource Estimation

Budget

(FIGURE HERE)

External Support

The team has received sponsorships from:

(FIGURE HERE)

The team has also received support from George Watson’s College via access to tools in their Technology department, Heriot Watt University for milling and bending aluminum, and from Tecbridge Circuits for PCB design assistance.

Test Plan:

Radio Testing:

We have tested our Yagi-Uda antenna and duck antenna with XBEE 868LP modules, simulating the base station and CanSat, in a nearby large field. We obtained a range of 2.3km, although we had lost line of sight due to tree cover.

We have also tested the XBEE 868LP modules with 1 (LAMBDA) antennae, leading to a range of 2.5km under similar conditions.

Outer Shell Testing:

Our outer shells have been tested and proven viable in the UK CanSat competition. Before the UK Competition, we had drop-tested weighted outer shells without parachutes from a height of 10m and filmed their impact in slow motion. This video confirmed the spring-like properties of the fiberglass that would provide protection to the CanSat.

Parachute Testing:

Our parachutes have also been proven viable in the UK CanSat competition with a 10.5ms-1 descent rate. Before the competition, we tested descent rate with 300g masses and a parachute attachment mechanism used in the actual CanSat. The parachute deployed successfully in 28 / 30 tests: the two failures had the parachute mostly unfurl before the CanSat

Primary Mission Testing:

The primary mission has been tested and confirmed working via long-term data logging over a period of three hours. The CanSat successfully logged and transmitted data to the base station.

Secondary Mission Testing:

The translational position and orientation tracking have been confirmed working via in-house tests at low acceleration. We plan to test positional tracking in a drop-test scenario before the competition.

We have also confirmed that the thermal camera mapping algorithm functions provided accurate data from the positional tracking algorithms.

Outreach

Our team has taken a multifaceted approach to CanSat outreach, presenting at school events, events open to the public, and via websites and social media.

Online Presence

Our team has social media available on many platforms, including (FOOTNOTES) Facebook (GWC CanSat), Instagram (@CanSatgwc), Twitter (@CanSatgwc2016). Frequent updates and usage of social media tags allow us to draw people interested in our team and in the competition to our pages. We have and have planned to continue frequently updating our social media pages, through the entirety of the team’s journey through the UK and EU competitions. (FIGURE HERE)

The team has also designed a website to provide further online presence to our CanSat, to the CanSat competition, and to our sponsors. Our website is also frequently updated: a current snapshot can be seen in figure (FIGURE HERE).

The team has also created a sponsorship brochure for the upcoming season, which can be seen in figure (FIGURE HERE).

Our CanSat and team have also received substantial press, including articles on and in 3dprint.com (FOOTNOTES), tctmagazine.com, stem.org.uk, Edinburgh Evening News, STEMLearningUK Twitter, and GWC\_News Twitter.

In-Person Outreach

The team has also performed several in-person outreach events. We have run presentations in technology and physics for school open days, and have run full-school presentations to expand on our team and on the competition. In one full school presentation, we presented our success at the UK competition in the context of the successes and failures we experienced as a team new to the CanSat competition, explaining the difficulties we’ve faced and the solutions we’ve implemented to solve these issues.

We have also initiated and ran educational outreach events for 9-10-year-old students. In the first session of the two-part workshop, students designed parachutes to meet certain descent velocity criteria: a reflection of the challenge we experienced in the UK CanSat competition. In the second section of the workshop, groups of students designed enclosure and padding to protect an egg dropped with their previously built parachute. To conclude the workshop, students tested their enclosure and parachute in a real-world egg drop. Some pictures of the event can be seen in figure (FIGURE HERE).

Finally, the team presented in the Edinburgh Mini Maker Fair during the Edinburgh Science Festival. This gave the team the opportunity to present our progress within the competition as well as speak to other people interested in what we have been doing. This included a slideshow of pictures of the CanSat and the team’s trip to York for the UK National Competition as well having a demonstration of the working CanSat, plotting data on graphs on a display. The team was also open to questions to the general public during the entirety of the Maker Faire.

CanSat Requirement Specifications:

(INSERT FIGURE HERE)