# CanSat 2011 Critical Design Review

**Team 513**

**Virginia Tech Team Rocket**

# Presentation Outline

- **Systems Overview** – George Bacon
- **Sensor Subsystem Design** – Chris Jennette
- **Descent Control Design** – Younes Taleb
- **Mechanical Subsystem Design** – David Pudleiner
- **Communication & Data Handling Subsystem Design** – Chris Stack
- **Electrical Power Subsystem Design** – Chris Jennette
- **Flight Software Design** – Chris Stack
- **Ground Control System Design** – Chris Jennette
- **Cansat Integration and Test** – Stephanie Butron
- **Mission Operations and Analysis** – Younes Taleb
- **Management** – Stephanie Butron

# Team Overview

| Name | Major | Year |
|---|---|---|
| George Bacon | Aerospace Engineering | Junior |
| Stephanie Butron | Aerospace Engineering | Junior |
| Chris Jennette | Electrical Engineering | Junior |
| Aubrey Knight | Aerospace Engineering | Junior |
| David Pudleiner | Mechanical Engineering | Junior |
| Moises Seraphin | Aerospace Engineering | Junior |
| Chris Stack | Computer Engineering | Junior |
| Younes Taleb | Aerospace Engineering | Junior |
| Greg Woodhouse | Aerospace Engineering | Junior |

# Team Organization

**Team Leaders**
George Bacon
Stephanie Butron

**Faculty Advisor**
Troy Henderson, Ph.D.

**Team Mentor**
Scott Reedy

- Mechanical Systems & Structures Group
  - **Leader** David Pudleiner
  - Moises Seraphin
  - Younes Taleb
  - Greg Woodhouse
- Computer & Electrical Group
  - **Leader** Chris Jennette
  - Chris Stack
  - Younes Taleb
- Mission Operations & Ground Control Group
  - **Leader** Younes Taleb
  - Chris Jennette
  - Chris Stack
- Systems Engineering Group
  - **Leader** George Bacon
  - Stephanie Butron
  - Aubrey Knight

# Acronyms

- **A** - Analysis
- **ADC** – Analog-to-Digital Converter
- **ADR** – Average Descent Rate
- **ALD** – Audible Locating Device
- **API** – Application Programming Interface
- **CDH** – Communication & Data Handling
- **D** - Demonstrate
- **DCD** – Descent Control Design
- **DCS** – Descent Control System
- **DS** – Datasheet
- **EEPROM** – Electrically Erasable Programmable Read-Only Memory
- **EST** – Estimate
- **FIFO** – First in, First out
- **FOS** – Factor of Safety
- **FSW** – Flight Software
- **GCS** – Ground Control System
- **I** - Inspect
- **I2C** – Inter-Integrated Circuit
- **I/O** – Input/Output
- **IDE** – Integrated Development Environment
- **IV** – Initial Velocity
- **M** - Measured
- **MMCX** – Micro-Miniature Coaxial
- **RSSI** – Received Signal Strength Indication
- **S/H** – Shipping and Handling
- **SEC –** Student Engineering Council
- **SPI** – Serial Peripheral Interface
- **SSD** – Sensor Subsystem Design
- **T** - Test
- **VM** – Verification Matrix

# Systems Overview

George Bacon

# Mission Summary

- Launch an autonomous CanSat with a deployable lander containing one large raw hen egg
- CanSat will be deployed from a rocket at an altitude around 1020 meters and start transmitting GPS telemetry
- At 500 meters, the carrier shall release the lander that contains one large raw hen egg
- After separation, the carrier shall have an Average Descent Rate (ADR) of 4 meters per second and the lander shall have an ADR or 5.5 meters per second
- The lander should land without damaging the egg and measure the force of impact with the ground

# Summary of Changes Since PDR

- **Carrier**
  - Electronic components locations decided
  - Materials changed for internal framework
  - Minor electronic changes
  - Length changed
- **Lander**
  - Electronic components locations decided
  - Materials changed for internal framework
  - GPS removed
  - Minor electronic changes
  - Length changed

# System Requirements

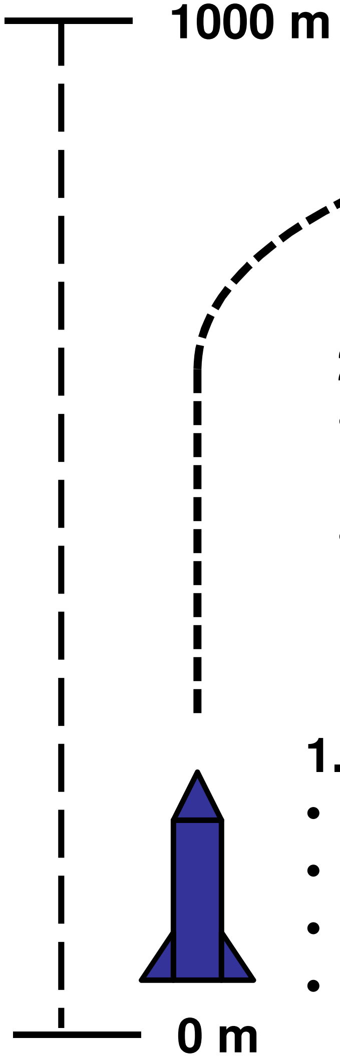
| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| SYS-01 | Total mass shall not exceed 500 g (excluding egg) | Base Mission Requirement | High | None | DCD-01,EPS-03,MS-01 | | X | | |
| SYS-02 | CanSat shall be compatible with the Loc/Precision Forte Rocket as specified in the payload section | Base Mission Requirement | High | None | DCD-02,MS-02 | | X | | |
| SYS-03 | CanSat shall comply with descent and recovery requirements | Base Mission Requirement | Medium | None | DCD-05,DCD-06,DCD-07,CDH-10 | | X | X | X |
| SYS-04 | CanSat shall comply with communication requirements | Base Mission Requirements | Medium | None | CDH-09, GCS-04, GCS-05, GCS-06 | | X | | |
| SYS-05 | CanSat and associated operations shall comply with all field safety regulations as outlined in Field Section Rules section | Base Mission Requirement | Medium | None | None | | X | | |
| SYS-06 | CanSat shall be launched within the assigned launch window | Base Mission Requirement | High | None | None | | | | X |
| SYS-07 | CanSat shall comply with power requirements | Base Mission Requirement | High | None | EPS-07,EPS-08 | | X | | |

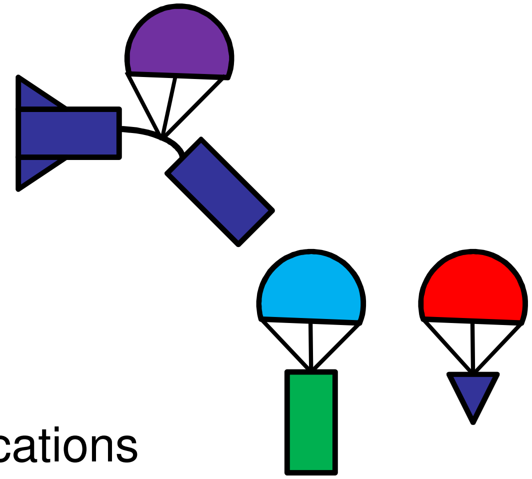
# System Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| SYS-08 | Cost of the CanSat flight hardware shall be under $1000 (Ground support and analysis are excluded) | Base Mission Requirement | Medium | None | None | | X | | |
| SYS-09 | Each team must use their own ground station and comply with telemetry requirements | Base Mission Requirement | High | None | FSW-07,FSW-08,FSW-09 | X | X | | X |
| SYS-10 | Lander CanSat shall measure the force of impact with the ground | Selectable Objective Requirements | High | None | SSD-03,SSD-05 | | X | X | X |
| SYS-11 | CanSat must withstand vibrational forces due to rocket launch | Do not want Cansat components to come apart | High | None | CDH-11, SSD-08, MS-07 | | | X | |
| SYS-12 | CanSat must easily slide out of rocket when deployed | Do not want CanSat to get stuck in rocket | High | None | None | | | X | |

# System Concept of Operations





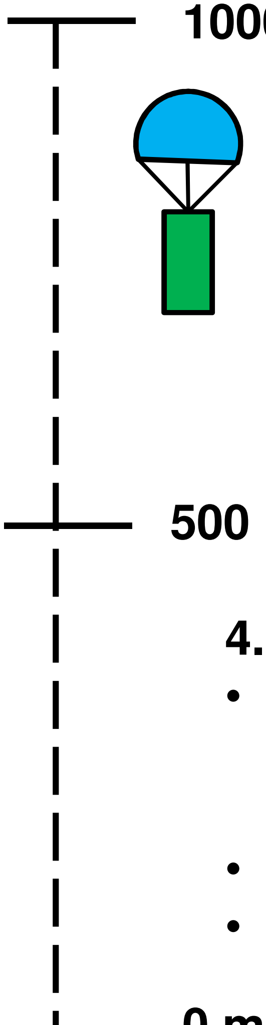
**1. Pre-Rocket Launch (-10:00)**
- Slide components into CanSat
- Insert egg into lander
- Power CanSat on
- Integrate CanSat into rocket payload

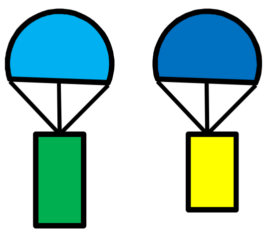
**2. Rocket Launch (00:00)**
- Begin ground communications with CanSat
- Receive telemetry

**3. CanSat Deployed (00:30)**
- CanSat deployed at approximately 1000 m
- Carrier parachute opens
- Altitude and pressure data is received by the Ground Control System (GCS)

# System Concept of Operations

1000 m



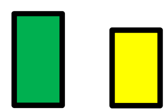


500 m

0 m

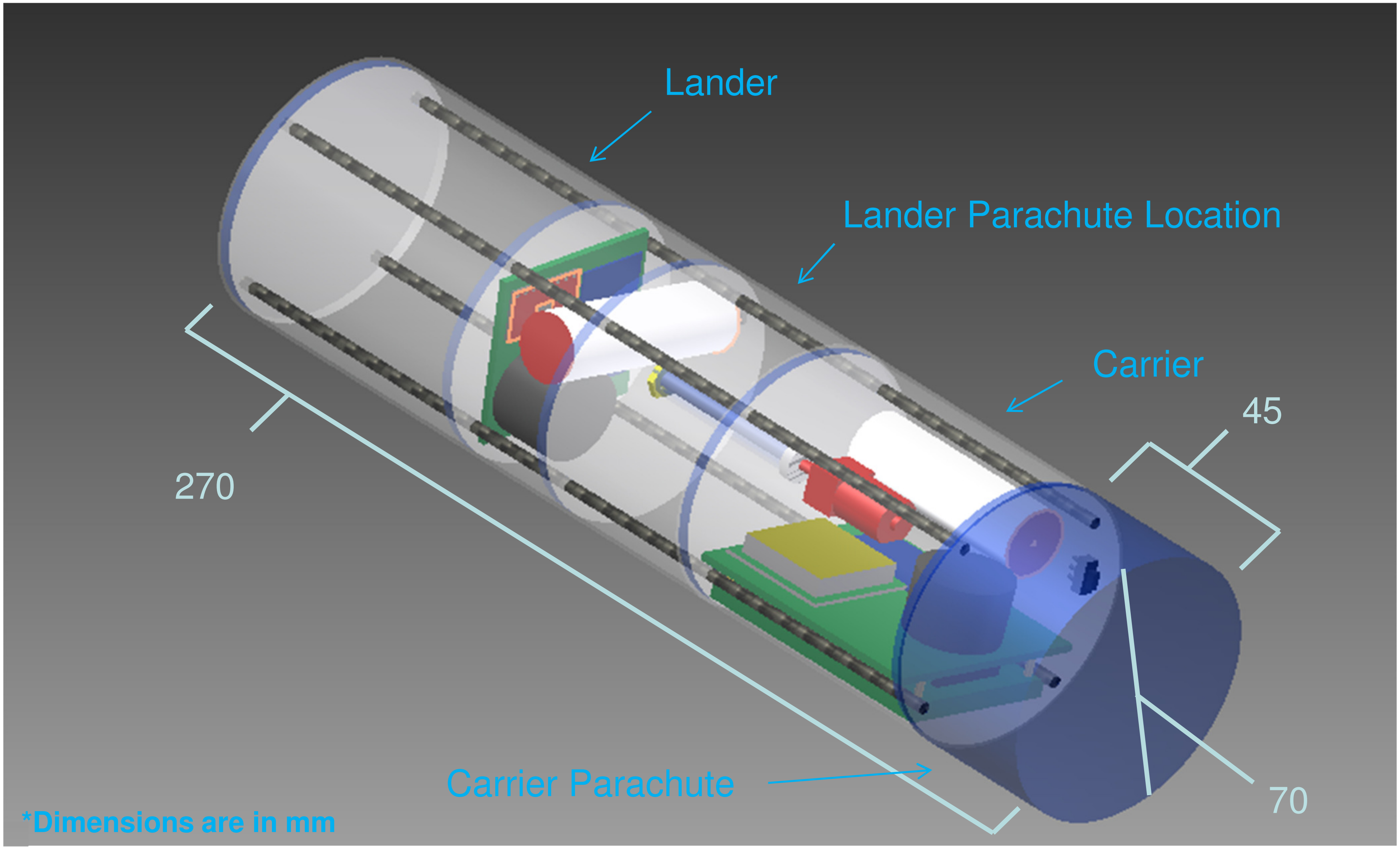
**5. Landing (03:28)**
- Audible beacons activate on both the carrier and lander upon landing
- Impact force recorded by the lander

**4. Separation (01:20)**
- An approximate altitude measurement of 500 meters initiates separation
- Lander is deployed
- Lander parachute opens

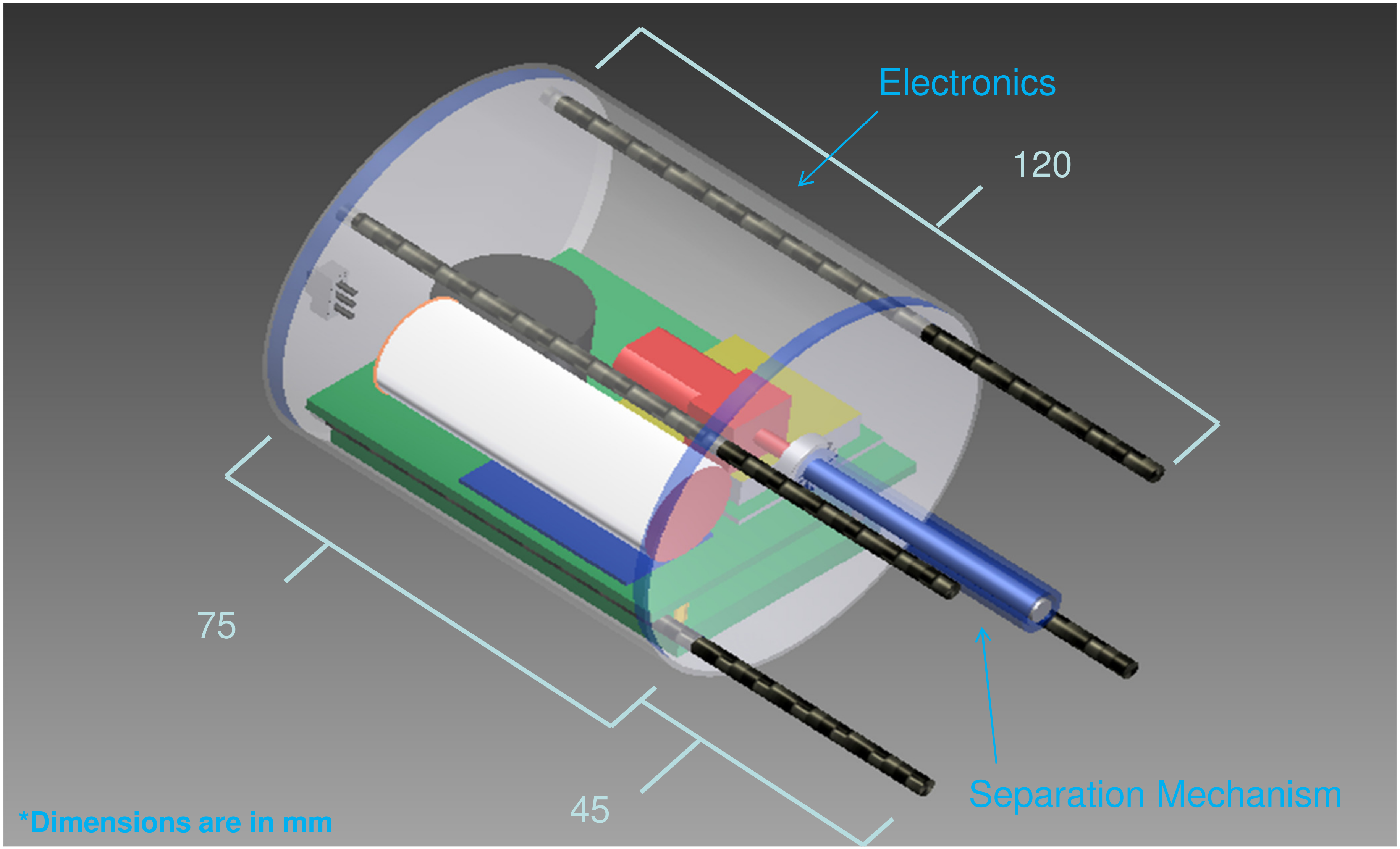


**6. Recovery (13:28)**
- Carrier and lander are retrieved
- Telemetry data is retrieved from the lander via a USB connection

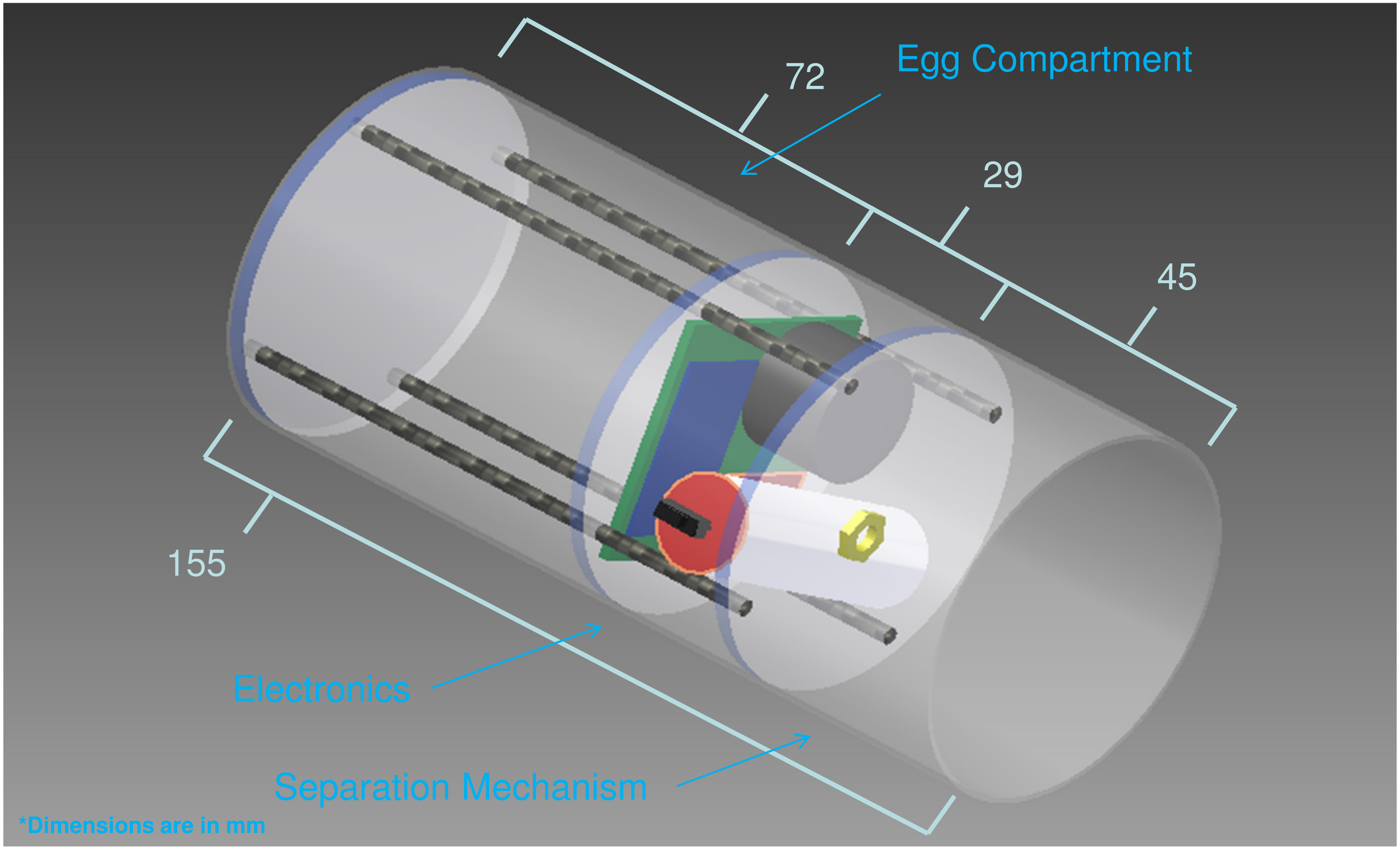
# Physical Layout



# Physical Layout - Carrier

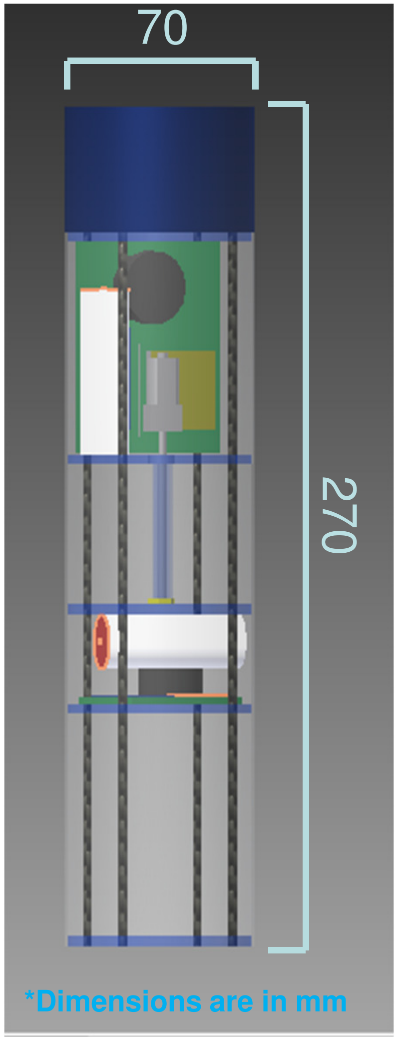


# Physical Layout - Lander



# Launch Vehicle Compatibility

- **CanSat will be placed upside-down in rocket payload**
  - Deployment from rocket will correct orientation
- **Rocket payload dimensions**
  - Height: 279mm
  - Diameter: 72 mm
- **CanSat dimensions**
  - Height: 270 mm
  - Diameter: 70 mm
- **Payload compatibility verification will take place during pre-launch checks with a pre-built rocket payload model**



# Sensor Subsystem Design

**Chris Jennette**

# Sensor Subsystem Overview

| Sensor Type | Model | Purpose | CanSat use |
|---|---|---|---|
| Pressure | BMP085 Breakout | Altitude, velocity, and temperature measurements | Carrier, Lander |
| GPS | 32 Channel LS20031 GPS 5 Hz Receiver | Position, altitude, and velocity measurements | Carrier |
| Accelerometer | ADXL345 | Triple axis measurement of impact force of lander | Lander |

# Sensor Subsystem Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| SSD-01 | All sensors used must be operable at 3.3 V or lower | Arduino microcontroller operates at 3.3 V | Medium | None | None | | X | | X |
| SSD-02 | GPS sensor must sample UTC time, latitude, longitude, mean sea level altitude, and number of satellites tracked | Base mission requirement | High | None | None | X | X | X | |
| SSD-03 | All sensors must be able to sample data at a rate of at least 1 Hz | Base mission requirement | High | SYS-10 | FSW-02 | | X | X | |
| SSD-04 | Pressure sensor shall measure altitude with accuracy of at least 0.5 hPa | Base mission requirement | High | None | None | | X | X | |
| SSD-05 | Accelerometer shall measure impact force at rate of at least 100 Hz | Base mission requirement | High | SYS-10 | None | | X | X | |
| SSD-06 | Sensors shall have serial or $I^2C$ connection interface. | Required to connect to Microcontroller | Medium | None | None | | X | | |

# Sensor Subsystem Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| SSD-07 | Pressure Sensor shall measure temperature with accuracy of at least 1 °C | Base mission requirement | High | None | None | | X | X | |
| SSD-08 | Sensors shall be able to withstand severe vibrations | Rocket Launch | High | SYS-11 | None | | | X | |

# Sensor Changes Since PDR

- **Carrier**
  - None
- **Lander**
  - Using different breakout board for accelerometer.  Now has mounting holes to provide additional stability (Model stays the same)

# Carrier GPS Summary



## GPS module chosen — LS20031

| GPS Unit | Price | Current Draw (mA) | Weight (g) | Accuracy (m) | Start (Cold/Hot) | Dimensions (mm) |
|---|---|---|---|---|---|---|
| 32 Channel LS20031 GPS 5 Hz Receiver | $59.95 | 41 | 14 | 3 | 36/2 | 30 x 30 |

- Accuracy typically $\pm 3m$
- Will be accessing the GPS using a software serial library on the Arduino.
  - Must use PMTK commands to limit messages to only GGA and lower baud rate and refresh time so Arduino is not overrun.
- Will be using NMEA GGA data messages. Ex.
  - $GPGGA,053740.000,2503.6319,N,12136.0099,E,1,08,1.1,63.8,M,15.2,M,,0000*64

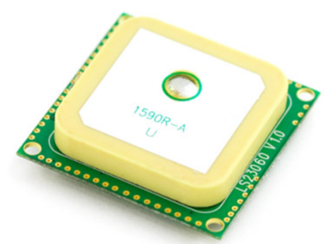


Photo courtesy of http://www.sparkfun.com/products/8975

# Carrier & Lander Non-GPS Altitude & Temp. Sensor Summary

## Pressure sensor chosen – BMP085

| Part | Price | Weight (g) | Resolution (bits) | Connection Type | Size (mm) | Sample Rate (Hz) |
|---|---|---|---|---|---|---|
| BMP085 | $19.95 | 2 | Pressure – 17<br>Temp. – 16 | $I^2C$ | 16.5 x 16.5 | 1 |

- Accurate to $\pm 0.2\ hPa\ \ (1.68\text{m})$ and $\pm 0.5\ °C$.
- Using the $I^2C$ interface, it only requires 4 pins to be connected: *Vcc, Gnd, SDA, and SCL.*
- Requires calibration on startup, the data for which is stored in it's registers.
- Calculates altitude using the equation: $altitude = 4430 * \left[1 - \left(\frac{p}{p_0}\right)^{\frac{1}{5.255}}\right]$
  - Where $p$ is the measured pressure and $p_0$ is the sea level pressure.
  - Range of 0-1000m is equal to $\Delta p$ of about $100\ hPa$.

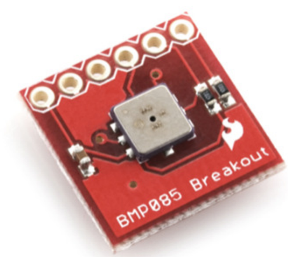



Photo courtesy of http://www.sparkfun.com/products/9694

# Lander Impact Force Sensor Summary



## Accelerometer Chosen – ADXL345

| Part | Price | Weight (g) | Current Consumption (µA) | Connection Type | Size (mm) | Additional Features |
|---|---|---|---|---|---|---|
| ADXL345 | $27.95 | 2 | 145 (0.1standby) | I²C and SPI | 3.05 x 5.08 | 16g res., 3200Hz sample rate |

- At the $\pm 16g$ range, accuracy of $32\frac{LSB}{g}$ or $\pm 0.03125g$, with resolution of $\frac{16}{2^{10}}g = 0.015625g$.
- Will be using $I^2C$ connection, using the pins: *SDA, SCL, CS tied to VCC, and SDO tied to GND.*
- Data will be read in the format: (x,y,z)
- Will begin sampling data when altitude is 30m, and will turn off when altitude is constant for 5 sec.

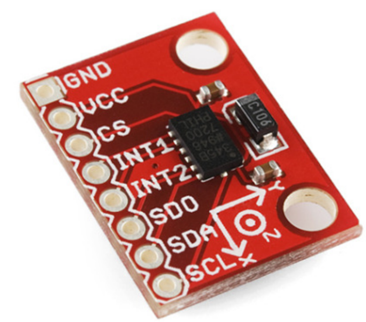



Photo courtesy of http://www.sparkfun.com/products/9156

# Descent Control Design

Younes Taleb

# Descent Control Overview

| Hardware | Model | Purpose | Theoretical Descent Rate |
|---|---|---|---|
| Carrier parachute | SkyAngle Classic II 20 | Maintain the descent rate to within 4.0 m/s +/- 1.0 m/s | 4.46 m/s |
| Lander parachute | SkyAngle Classic II 20 | Maintain the descent rate to within 5.5 m/s +/- 1.0 m/s | 4.83 m/s |

# Descent Control Changes Since PDR

- **Carrier**
  - None, parachute did not change
- **Lander**
  - None, parachute did not change

# Descent Control Requirements



| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| DCD-01 | The parachutes must fit inside the 72 mm diameter cylinder | Base Mission Requirement | High | SYS-02 | None | | X | | |
| DCD-02 | The parachute shall not exceed a packing depth of 70 mm | Allow for sufficient space allocated to the rest of the systems. | High | SYS-02 | None | | X | | |
| DCD-03 | Both parachutes must be reasonably light (under 30 g) | Keep the weight budget from exceeding 500 g | High | SYS-01 | None | | X | | |
| DCD-04 | Parachute shroud lines must be able to support the force of the wind buffeting during deployment | Prevent shroud lines from breaking or tearing the parachutes | High | None | None | | | | X |
| DCD-05 | Parachute must be designed to avoid tangling of shroud lines | Prevent tangling during descent that could lead to a failed recovery | High | None | None | | | | X |
| DCD-06 | Average Descent Rate (ADR) of the CanSat carrier after deployment of the lander shall be 4±1 m/s | Base Mission Requirement | High | SYS-03 | None | | X | X | |

# Descent Control Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| DCD-07 | ADR of the CanSat lander after deployment shall be 5.5±1 m/s | Base mission Requirement | High | SYS-03 | None | | X | X | |
| DCD-08 | Decent Control System (DCS) shall not use flammable or pyrotechnic devices | Base Mission Requirement | High | SYS-03 | None | | X | | |

# Descent Control Hardware Summary



## Descent Control Chosen: SkyAngle Classic II 20 Parachute

| Manufacturer | Price | Diameter (cm) | Shape | Weight (g) | Shroud line length (cm) | Carrier Descent Rate (m/s) | Lander Descent Rate (m/s) |
|---|---|---|---|---|---|---|---|
| SkyAngle | $22.00 | 50.8 | Round | 28.3 | 4.32 | 3.93 | 4.83 |

- Used for lander and carrier
- Appropriate for both the lander and carrier
- Low weight
- Designed for stability
- Includes a pre-installed swivel
- Low porosity
- Strong suspension lines


Photo courtesy of: http://www.apogeerockets.com/parachutes.asp

# Descent Rate Calculation

- **The equation used to size the parachute is:**

$$d = 2\sqrt{\frac{2W}{\rho\pi\, C_d\, V^2}}$$

**W** is the weight of the CanSat
**ρ** is air density at deployment altitude
$\boldsymbol{C_d}$ is the coefficient of drag. (Taken from table provided by SkyAngle)
**V** is the desired descent velocity
**d** is the diameter of the parachute

# Descent Rate Estimates

| Configuration | Weight (g) | Descent rate (m/s) |
|---|---|---|
| Carrier + Lander pre-separation | 500 | 10.2* |
| Carrier post-separation | 181.4 | 3.93* |
| Lander post-separation (egg included) | 273 | 5.65* |

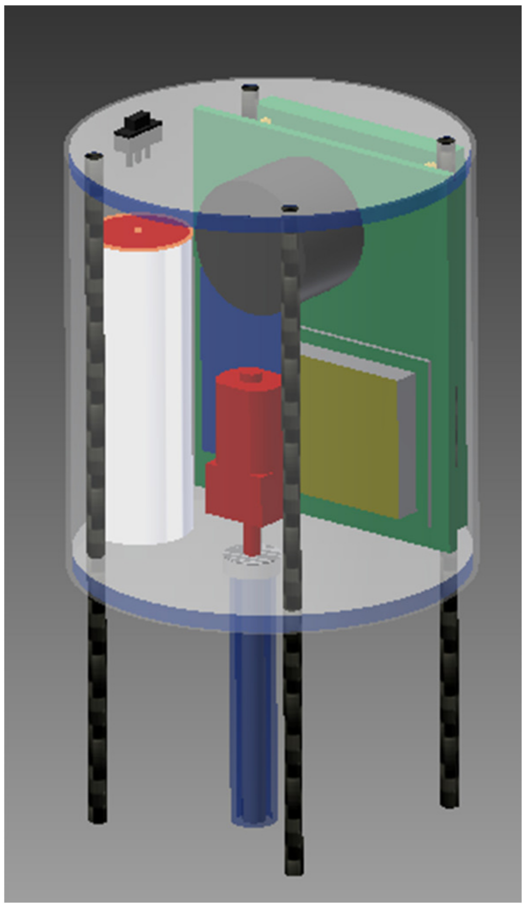
*Using a $C_d$ of 0.8 provided by the manufacturer and http://www.aerospaceweb.org/question/aerodynamics/q0231.shtml

# Mechanical Subsystem Design

**David Pudleiner**

# Mechanical Subsystem Overview

**Carrier**



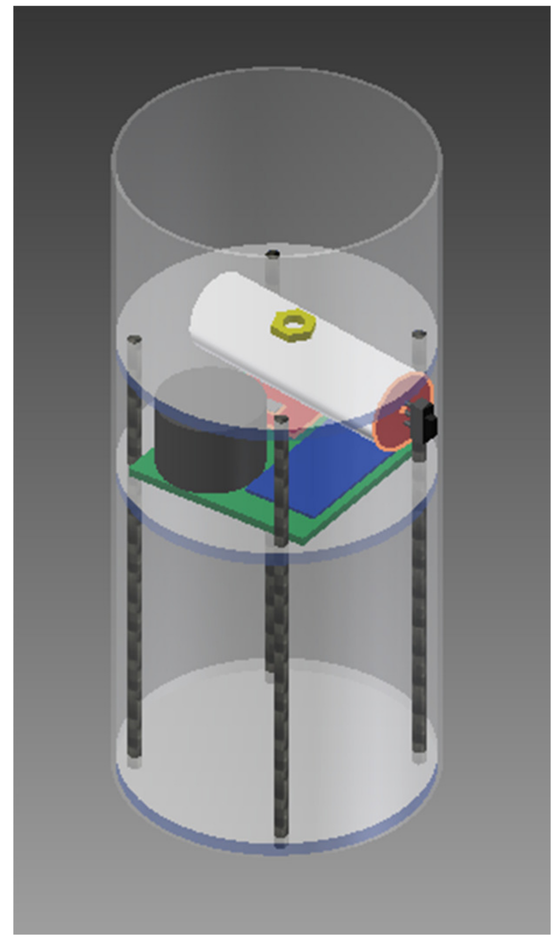
**Electrical Components:**
- Located on top of both carrier and lander to best resist impact

**Egg Compartment:**
- At the bottom of the lander to get lowest center of gravity for more predictable landing

**Separation Mechanism:**
-Screw and nut based design
- Contains lander parachute

**Internal Frame:**
- Rod and disc layout

**Lander**

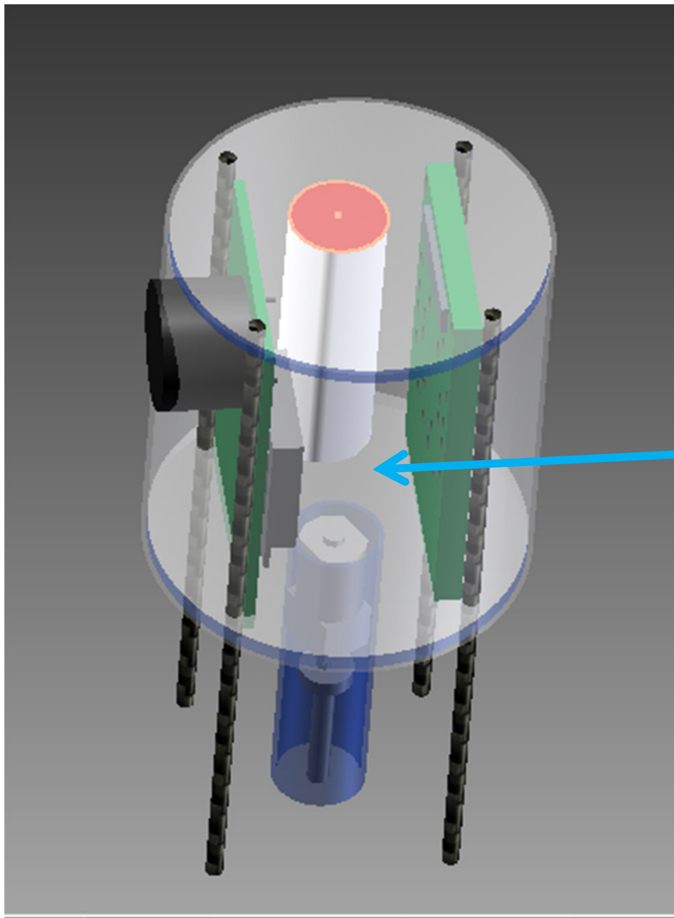


# Mechanical Subsystem Changes Since PDR

- **Carrier**
  - Improved frame material
    - Carbon fiber rods
    - Fiberglass (FR4) discs
  - Relocation/Alteration of Components
    - Motor moved inside the electronics compartment
    - Diameter of the shielding reduced
    - Subsequent reduction in separation mechanism space required due to increased space for parachute
- **Lander**
  - Improved frame material
    - Carbon fiber rods
    - Fiberglass (FR4) discs
  - Relocation/Alteration of Components
    - Changes in component layout in carrier produced a reduction in the lander fuselage length for separation

# Mechanical Subsystem Changes Since PDR

Old Design

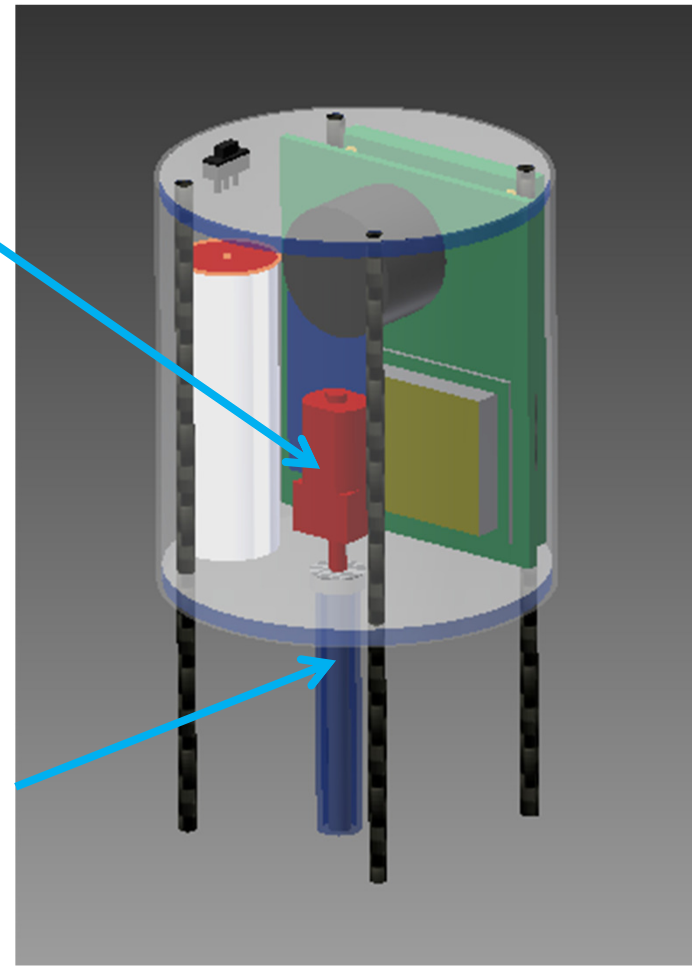


Improved motor placement

Unused Space in electronics compartment

Major reduction in shield tube diameter

New Design



# Mechanical Subsystem Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| MS-01 | Total mass of CanSat (without egg) shall not exceed 500 grams | Base Mission Requirement | High | SYS-01 | None | | X | | |
| MS–02 | CanSat shall fit within 72 mm diameter and 279 mm long payload bay | This is the maximum space that the payload of the rocket will permit | High | SYS-02 | None | | X | | |
| MS-03 | Internal structure shall support electronics during flight and impact | Certain electronics must continue to work after landing | High | None | None | | | X | |
| MS-04 | Use of metal components should be limited in size and number | Metal interferes with radio signal | Medium | None | None | | X | | |
| MS-05 | CanSat shall deploy lander from carrier at 500 m | Lander Requirement from Competition Guide | High | None | None | | | X | |

# Mechanical Subsystem Requirements

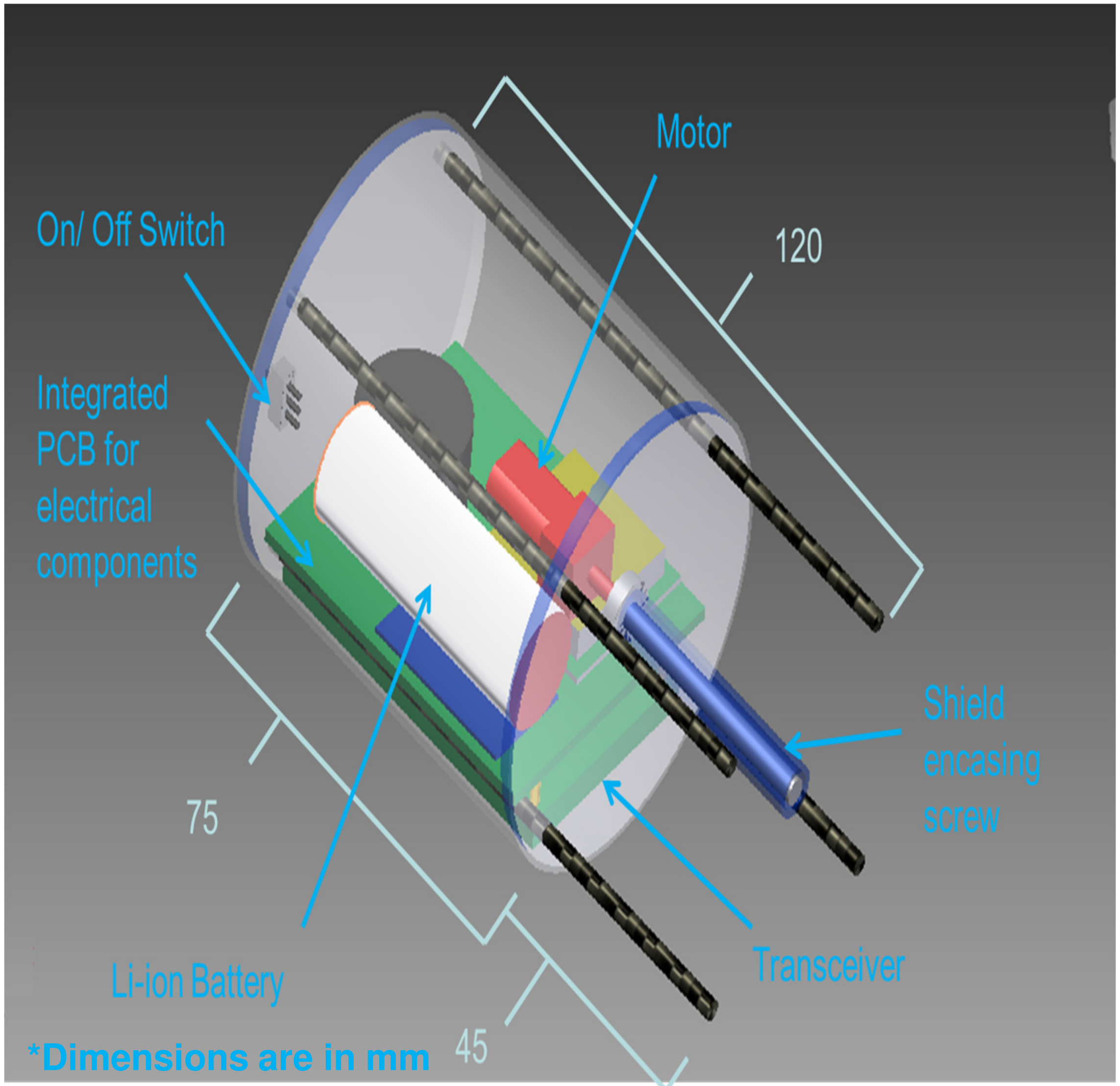
| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| MS-06 | Egg shall not break from impact from landing | Need to preserve precious cargo during impact | High | None | None | | | X | |
| MS-07 | CanSat must withstand vibrational forces due to rocket launch | Do not want Cansat components to come apart | High | SYS -11 | None | | | X | |

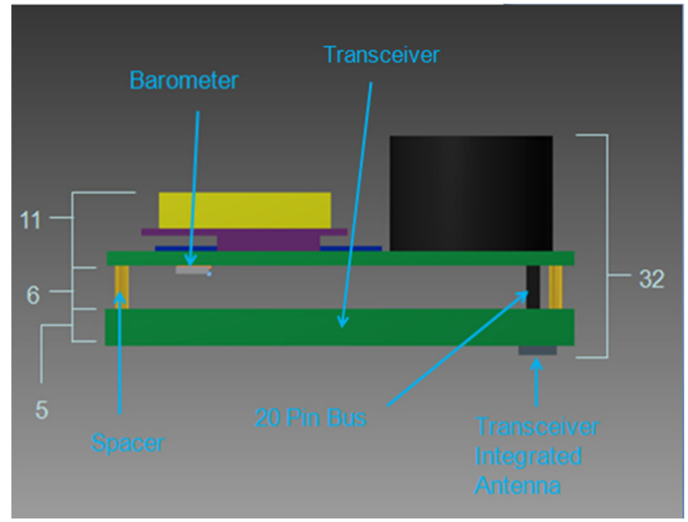
## Egg Protection Overview

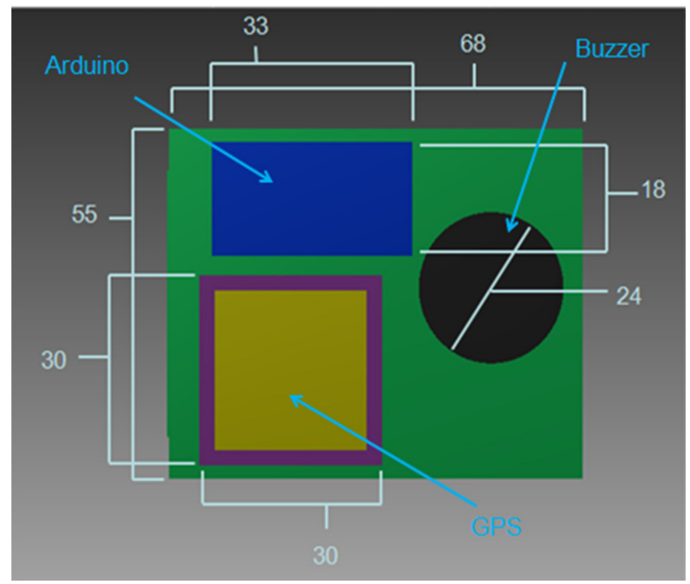
- **Design:**
  - Egg oriented horizontal, wrapped in 2.54 cm of pipe rubber
    - Based on drop testing this design provided the most dependable protection
    - Minimal space occupied
    - Light weight material
    - Easy and reliable to package
    - Egg protected on all sides
  - Bubble wrap
    - Used to take up any extra space in container, prevents any relative velocity between egg and lander

*Top and bottom rubber removed for picture

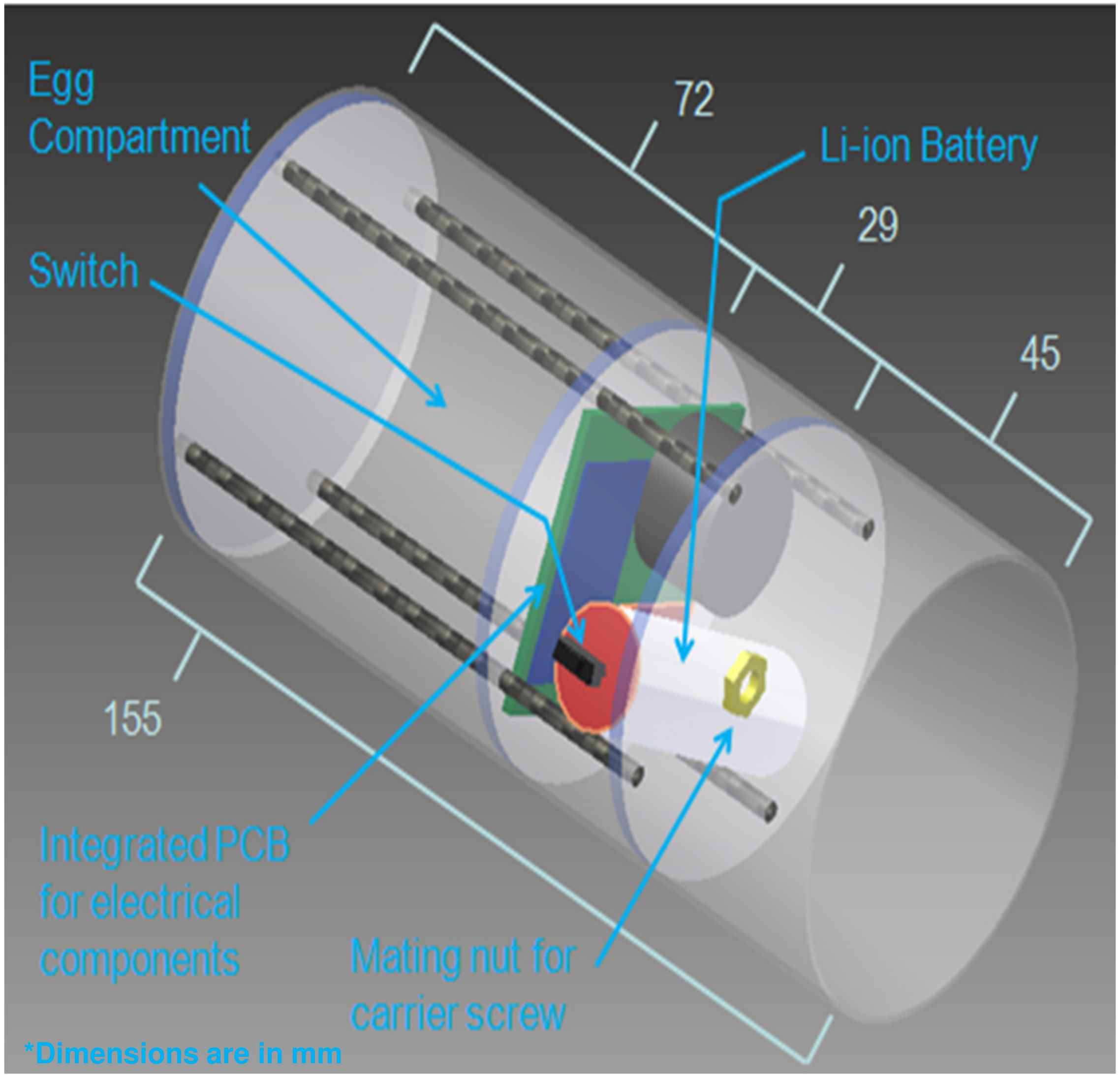
# Mechanical Layout of Components - Carrier

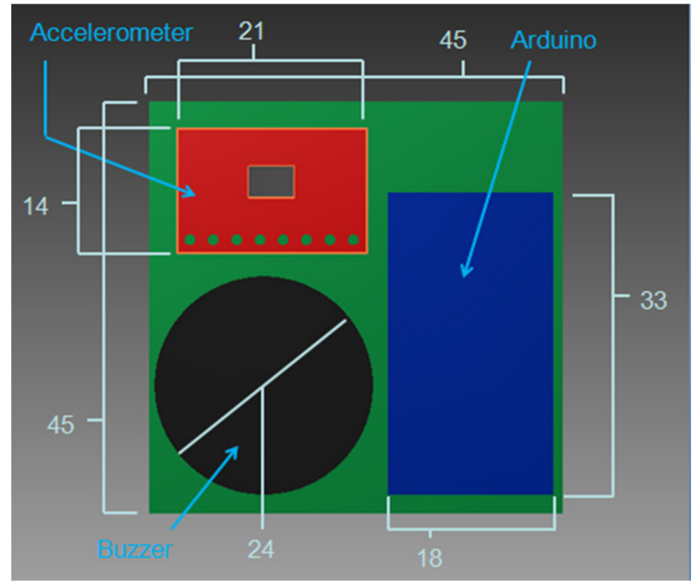


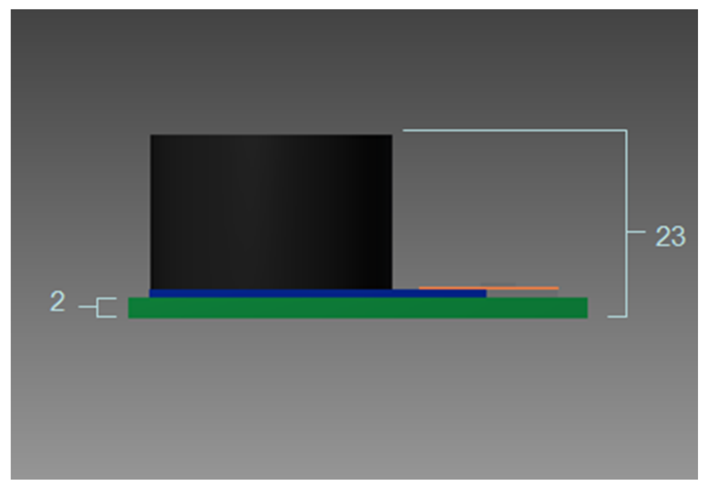



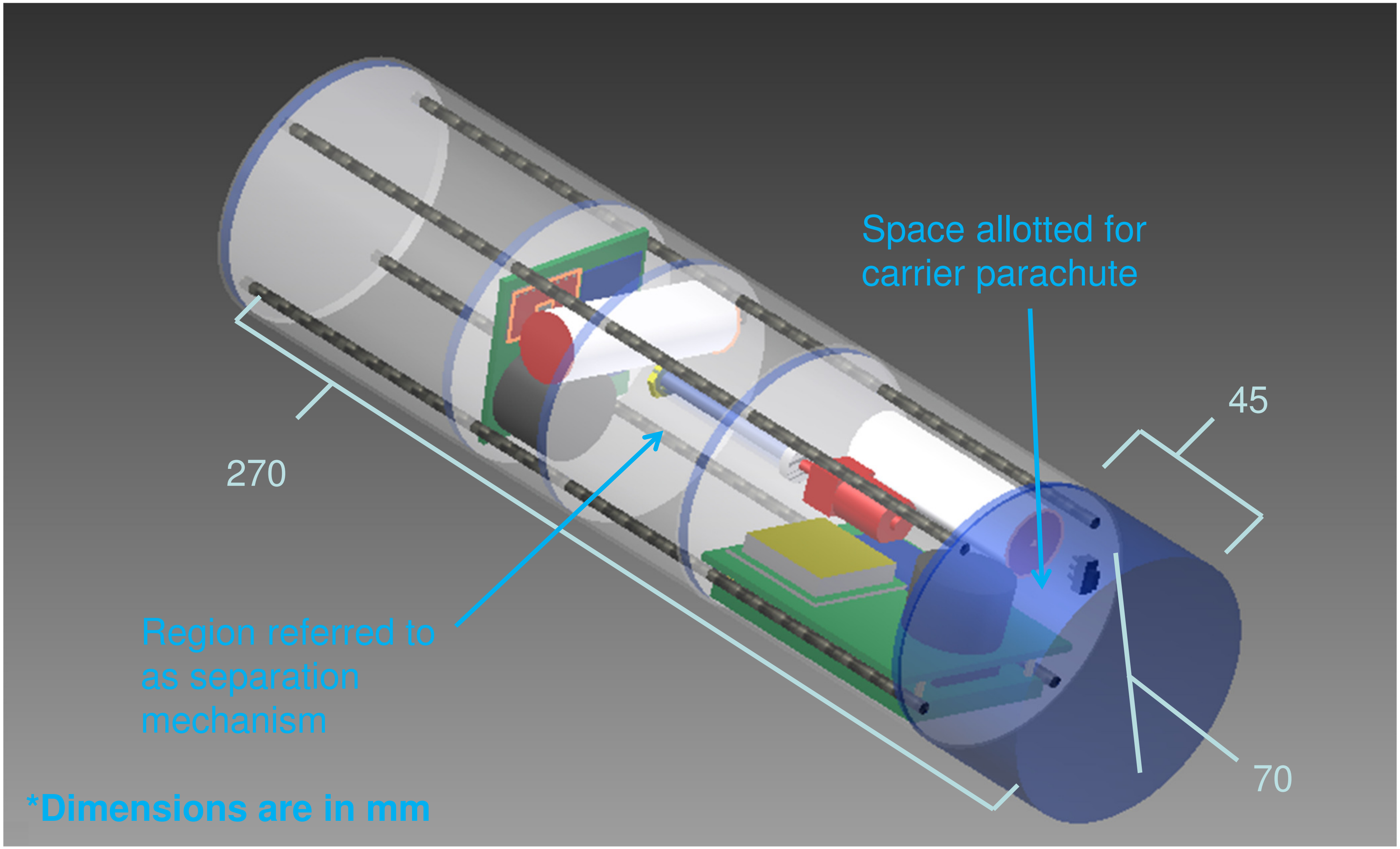
# Mechanical Layout of Components - Lander



# Mechanical Layout of Components – Entire CanSat



# Material Selection

## Outer Shell Material – Plastic (Polyethylene)

| Part | Price | Tensile Strength (MPa) | Density ($g/cm^3$) | Availability | Size (mm) |
|---|---|---|---|---|---|
| Tennis Ball Can | $3.95 | 55-75 | 1.40 | High | 72 (diameter) |

- Highly available and very low cost
- Easy to modify and fabricate to desired specifications
- Best combination of low density and relatively high strength
- Used for both lander and carrier


Photo Courtesy of: http://www.recommsports.com/store/kenko-soft-tennis-ball-starter-set.html

## Frame Rod Material – Carbon Fiber

| Part | Price | Tensile Strength (MPa) | Density (g/cm$^3$) | Availability | Size (mm) |
|---|---|---|---|---|---|
| Carbon Fiber Rod | $7.95 | >1000 | 1.49 | Moderate | 3.175 (dia.) x 100 |

- Available at several online hobby shops
- Rigidity allows for a much smaller diameter, reducing the volume that the frame occupies
- Best combination strength and density
- Used for both lander and carrier
- Reasonably Priced
- Non-conducting so as not to interfere with electronics

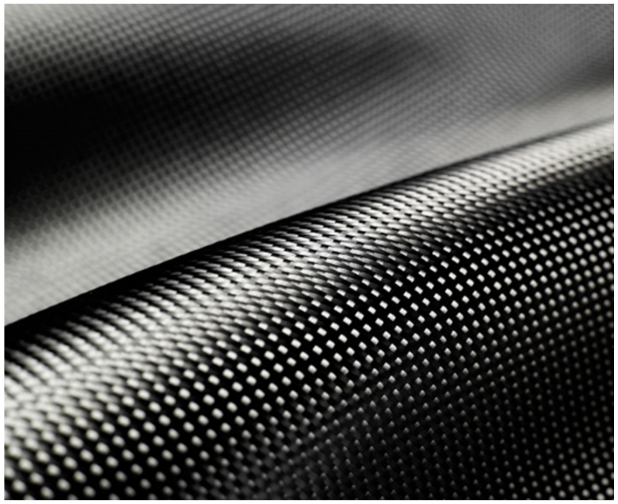


Photo Courtesy of: http://www.idea.ideabing.com/2010/08/02/ideabing-daily-roundup-carbon-fiber-edition/

## Frame Disc Material – Fiberglass (FR4)

| Part | Price | Tensile Strength (MPa) | Density ($g/cm^3$) | Availability | Size (mm) |
|---|---|---|---|---|---|
| FR4 Sheet | $11.70 | 310 | 1.85 | Moderate | 305 x 400 x 1 |

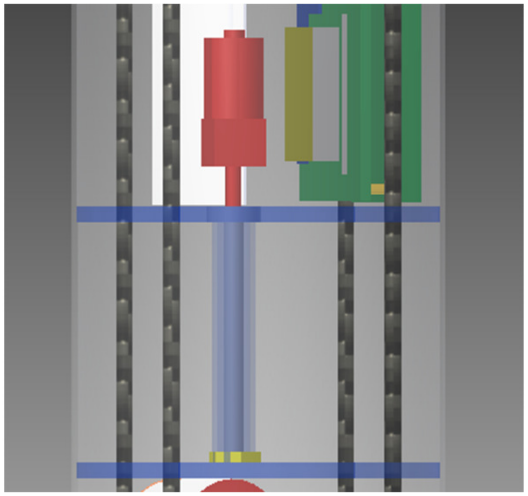
- Readily available at a low price
- The rigidity of the material allows for a thinner disc to be used, increasing the volume inside the CanSat
- Have worked with this material previously
- Used for both lander and carrier
- Non-conducting so as not to interfere with electronics

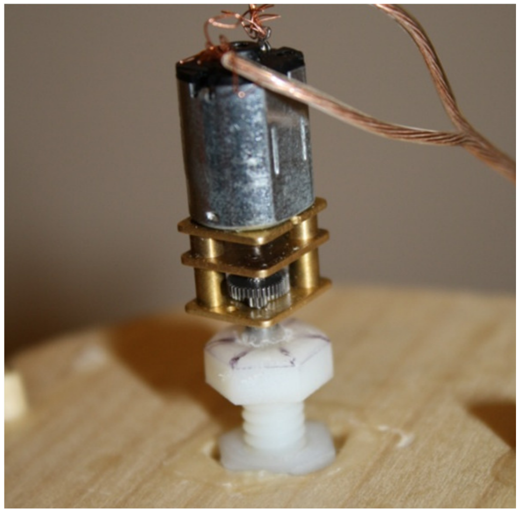
Photo Courtesy of: http://www.bridgat.com/g10_fr4_epoxy_fiberglass_cloth_laminated_sheet-o218605.html

# Carrier-Lander Interface



- **The mechanism connecting the carrier and lander will be a screw and nut**
- **A motor will actuate the screw to spin and release from the nut**
  - Parachute will be wrapped around the shield
  - Possible rip chord to ensure proper deployment
  - Prevention of accidental separation because of motor gear reduction





# Mass Budget

## Lander

| Component | Mass (g) | Uncertainty (%) |
|---|---|---|
| Frame | 70 | +/- 20 E |
| Parachute | 80 | +/- 5 M |
| Exterior Shell | 20 | +/- 20 E |
| BMP085 Pressure Sensor | 2 | +/- 5 D |
| Arduino Pro Mini 328 - 3.3V/8MHz | 2 | +/- 5 D |
| Li-Ion 14500 Battery | 20 | +/- 5 D |
| Buzzer 668-1028-ND | 6 | +/- 5 D |
| EEPROM | 1 | +/- 20 E |
| Switch | 2 | +/- 5 D |
| PCB | 10 | +/- 20 E |
| Egg Protection Material | 10 | +/- 5 M |
| Accelerometer ADXL345 | 2 | +/- 5 D |
| | | |
| **Lander Total** | **225** | |

## Carrier

| Component | Mass (g) | Uncertainty (%) |
|---|---|---|
| Frame | 60 | +/- 20 E |
| Parachute | 80 | +/- 5 M |
| Exterior Shell | 20 | +/- 20 E |
| BMP085 Pressure Sensor | 2 | +/- 5 D |
| Arduino Pro Mini 328 - 3.3V/8MHz | 2 | +/- 5 D |
| Li-Ion 14500 Battery | 20 | +/- 5 D |
| Buzzer 668-1028-ND | 6 | +/- 5 D |
| GPS sensor | 14 | +/- 5 D |
| RF Module Laird AC4790-200A | 21 | +/- 5 D |
| Motor | 8 | +/- 5 D |
| Switch | 2 | +/- 5 D |
| PCB (including bus and spacers) | 15 | +/- 20 E |
| | | |
| **Carrier Total** | **248** | |
| **CanSat Total** | **473** | |

# Communication & Data Handling Subsystem Design

Chris Stack

# CDH Overview

- **Lander**
  - Data from the voltage divider, accelerometer, and barometer is received by the Arduino Pro Mini and stored on external EEPROM
  - Retrieved later by reading from the Arduino using an FTDI Cable
- **Carrier**
  - Data from the GPS, barometer, and voltage divider is received by the Arduino Pro Mini and transmitted by the AC 4790 transceiver to the Ground Control System (GCS) for redundancy data will also be stored on an external EEPROM

# CDH Changes Since PDR

- **Carrier**
  - Added a Microchip Microchip 24LC256 EEPROM on carrier for data backup
- **Lander**
  - Changed from an Microchip Microchip 24LC256 EEPROM to a Microchip 24AA1025 EEPROM on lander

# CDH Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| CDH-01 | Shall transmit or store all subsystem data | Base Mission Requirement | High | None | None | X | | X | X |
| CDH-02 | Microcontroller will allow $I^2C$ and serial communication. (2 pins needed for I2C and 4 pins for serial) | All subsystems use either of these two protocol. | High | None | None | | X | | |
| CDH-03 | Microcontroller shall operate at high enough frequency to manipulate data and still output at 0.5 HZ (at least 1 MHZ) | Data from all sensors must be integrated into one packet | Medium | None | None | | X | X | |
| CDH-04 | External memory for lander shall store at least 24 Kb | Data from descent and impact will be at least this large | Medium | None | None | | X | | X |
| CDH-05 | Shall store and transmit all data using Hexadecimal encoding | Using Hexadecimal encoding will lower the number of bytes needed to be stored and transmitted | Low | None | None | | X | | |
| CDH-06 | Communications radio shall be Laird AC4790-200 | Base Mission Requirement | High | SYS-04 | None | | X | | |
| CDH-07 | Communications shall use the Laird AC4790 packet format | Base Mission Requirement | High | SYS-04 | None | | X | | |
| CDH-08 | Radio must not use the broadcast mode | Base Mission Requirement | High | SYS-04 | None | | X | | |

# CDH Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| CDH-09 | CanSat shall autonomously terminate telemetry transmissions within 5 minutes of landing, verified by Ground Control System (GCS) | Requirement set by CG | High | SYS-04 | None | | X | | |
| CDH-10 | Audible locating device s(ALD) hall activate upon | Base Mission Requirement | Medium | SYS-03 | CDH-10a,CDH-10b,CDH-10c | | X | | X |
| CDH-10a | ALD shall not be activated during launch and launch activities | Base Mission Requirement | Medium | SYS-03,CDH-10 | None | | | | X |
| CDH-10b | ALD shall operate for at least 1 hour following activation | Base Mission Requirement | Low | SYS-03,CDH-10 | None | | | | X |
| CDH-10c | ALD shall be at least 95 dB | Needs to be loud enough to hear from a distance | Low | SYS-03,CDH-10 | None | | X | | |
| CDH-11 | Arduino will operate under severe vibrations | Do not want connections to go bad during launch or landing | Medium | SYS-11 | None | | | X | |

# Processor Selection

## Processor Chosen: Arduino Pro Mini 3.3V

| Microcontroller | Input Voltage (V) | Current (mA per pin) | Clock Frequency (MHz) | Digital Pins | Analog Pins | Flash Memory (kb) | EEPROM (bytes) | Weight (g) | Dimensions (mm) | Price |
|---|---|---|---|---|---|---|---|---|---|---|
| Arduino Pro Mini 328 | 9 | 40 | 8/16 | 14 | 6 | 16 | 512 | 1 | 17.8 x 33.02 | $18.95 |

- Used for lander and carrier
- Chosen for large user base, abundance of interfaces, small size, and 3.3 V operating voltage
- Downsides are an 8 MHz clock, no onboard USB, and less memory then other Arduinos

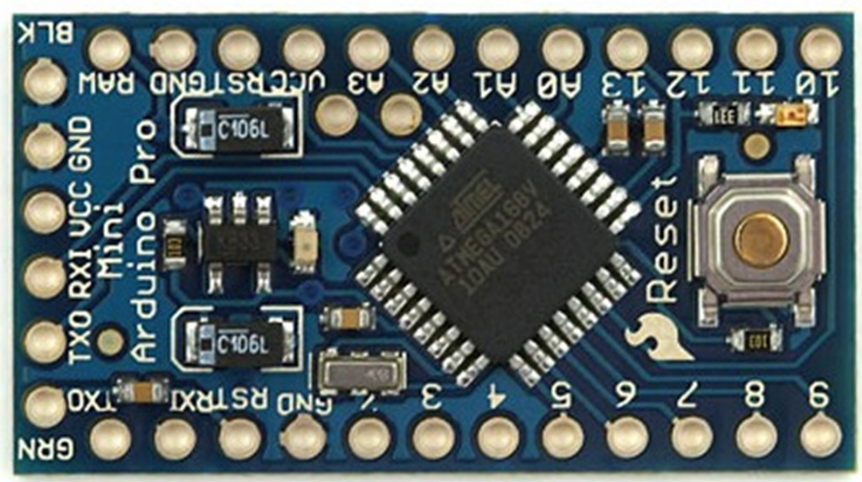



Photo courtesy of: http://arduino.cc/en/Main/ArduinoBoardProMini

# Memory Selection



## Memory Chosen: Microchip 24LC256 and 24AA1025

| Memory Unit | Input Voltage (V) | I$^2$C Clock Frequency (kHz) | Current Draw (mA) | Protocol | Memory (kb) | Price |
|---|---|---|---|---|---|---|
| Microchip 24LC256 | 2.5-5.5 | 400 | 3 | I$^2$C | 32 | $1.95 |
| Microchip 24AA1025 | 2.5-5.5 | 400 | 3 | I$^2$C | 128 | $4.28 |

- Chose to use Microchip 24AA1025 on lander
  - Needs a large amount of memory to store accelerometer data
  - Well documented examples with Arduino
- Chose to use Microchip 24LC256 on carrier
  - Needed to backup data in case of problems with transmission
  - Same protocol as lander module so code is portable

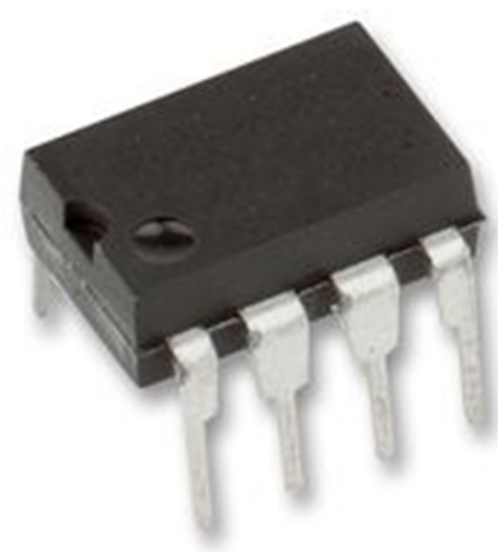


Photo courtesy of: http://cdn.sigma.octopart.com/308656/image/Microchip-24LC512-I/P.jpg

# Carrier Antenna Selection



## Antenna Chosen: gigaAnt Integrated Antenna

| Antenna | Price | Type | Gain (dB) | Length (cm) | Interface |
|---|---|---|---|---|---|
| gigaAnt Integrated Antenna | $1.35 | Microstrip | -0.5 | 3 | N/A |

- Comes pre-mounted on transceiver PCB making mounting easy, as well as saving space
- Approved for use as noted in data sheet

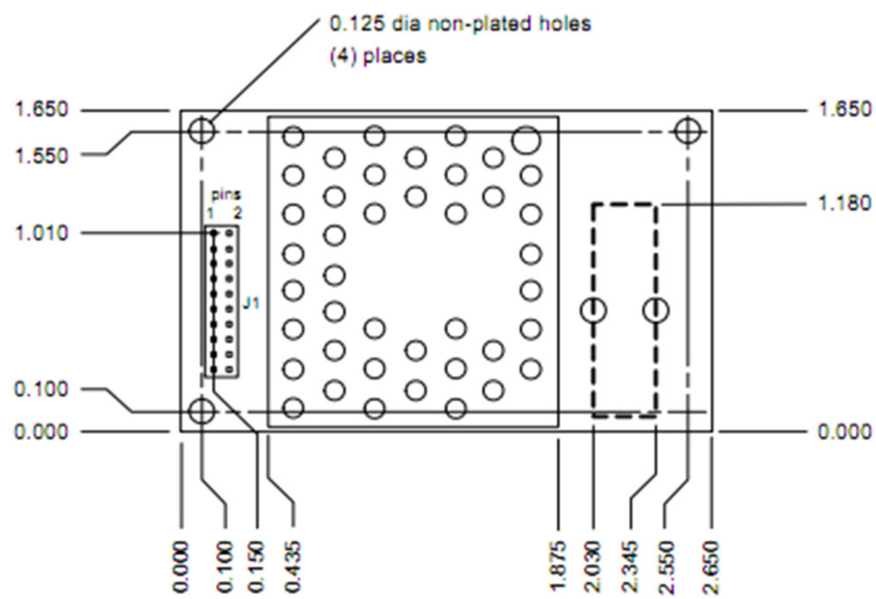



Photo courtesy of Laird 4790 Radio Users Guide

# Data Package Definitions

- **Radio – AC4790**
  - Operates over serial TTL connection, requires a series of hex commands outlined on next slide before data can be sent
- **GPS – LS20031**
  - Operates over serial TTL connection, outputs different data sets at different speeds depending on commands given to it
- **Barometer – BMP085**
  - Operates over $I^2C$, which means we have to start a transmission to its address, send a request for data, and then tell it what data we want and wait for it to be available
- **Accelerometer – ADXL345**
  - Operates over $I^2C$, which means we have to start a transmission to its address, send a request for data, and then tell it what data we want and wait for it to be available
- **Storage device protocols - Microchip 24LC256 and 24AA1025**
  - Operates over $I^2C$, which means we have to start a transmission to its address, send a request for data, and then tell it what data we want and wait for it to be available

# Communication Configuration

- **Mission guidelines require that we must use the AC4790 transceivers in API mode to reduce interference**

**Commands to send data using an AC4790 Transceiver in API Format**

| Command Number | Hex Value | Function |
|---|---|---|
| 1 | 0x81 | Tell the AC4790 to transmit using API format |
| 2 | Bytes in payload | Tell how many bytes are being sent |
| 3 | Session Count | Refresh the session count |
| 4 | Number of transmit retries | The number of times to retry |
| 5-7 | Address Bytes | MAC address of destination receiver |
| 8 - ? | Hex values of the payload | Payload to be sent |

**Format of data received by AC4790 at GCS**

| Byte Number | Hex Value | Function |
|---|---|---|
| 1 | 0x81 | Tell the AC4790 to receive in API format |
| 2 | Bytes in payload | Tell how many bytes are being received |
| 3 | RSSI | Quality of connection |
| 4 | RSSI | Quality of connection |
| 5-7 | Address Bytes | MAC address of source transceiver |
| 8-? | Hex values of the payload | Received Payload |

# Carrier Telemetry Format

- **Data included in Carrier Transmissions:**
  - From GPS: UTC time, latitude, longitude, altitude, and satellites tracked
  - From barometer: pressure and temperature
  - From voltage divider: battery voltage
- **76.8 kbps data rate through air, 57600 bps on CanSat and GCS**
- **Tentative Format:**
  - All values in hex
  - 38 characters including parity

# Autonomous Termination of Transmissions

- **Change in pressure constantly monitored by Arduino, when pressure is constant for 5 minutes transmission will be terminated**
  - Will have a pre-launch check that does not turn off until pressure has changed by a significant amount so Arduino does not end transmission before takeoff.
  - Ground station will verify end of transmission by displaying the message "Lost Contact"

# Carrier Locator Device Summary

## Locator Chosen: AI-2429-TWT-R

| Part | Price | Loudness (dB) | Size (mm) | Weight (g) | Power Req. |
|---|---|---|---|---|---|
| AI-2429-TWT-R | $4.51 | 100 | 23.8 (Dia.) x 16 | 6 | 8 mA<br>3-20 VDC |

**Device chosen – AI-2429-TWT-R**

- Smaller size and weight to better suit space requirements

**Method used for enabling/disabling**

- Will activate when pressure detected is constant for 30 sec.
- Will have a pre-launch check that does not turn off until pressure has changed by a significant amount so Arduino will not activate prior to launch.
- Can be turned off by power switch on CanSat


Photo courtesy of
http://search.digikey.com/scripts/DkSearch/dksus.dll?Detail&name=668-1028-ND

# Electrical Power Subsystem Design

**Chris Jennette**

# EPS Overview

| Part | Purpose |
|---|---|
| Li-Ion 14500 Battery | Powers all components within carrier and lander |
| Slider Switch | Controls power from battery to main circuit |
| Relay (Transistor) | Allows power to flow from battery directly to actuator when signal from Arduino is detected |
| Voltage Divider (resistor circuit) | Scale down total power to measure voltage using Arduino ADC |

# EPS Changes Since PDR

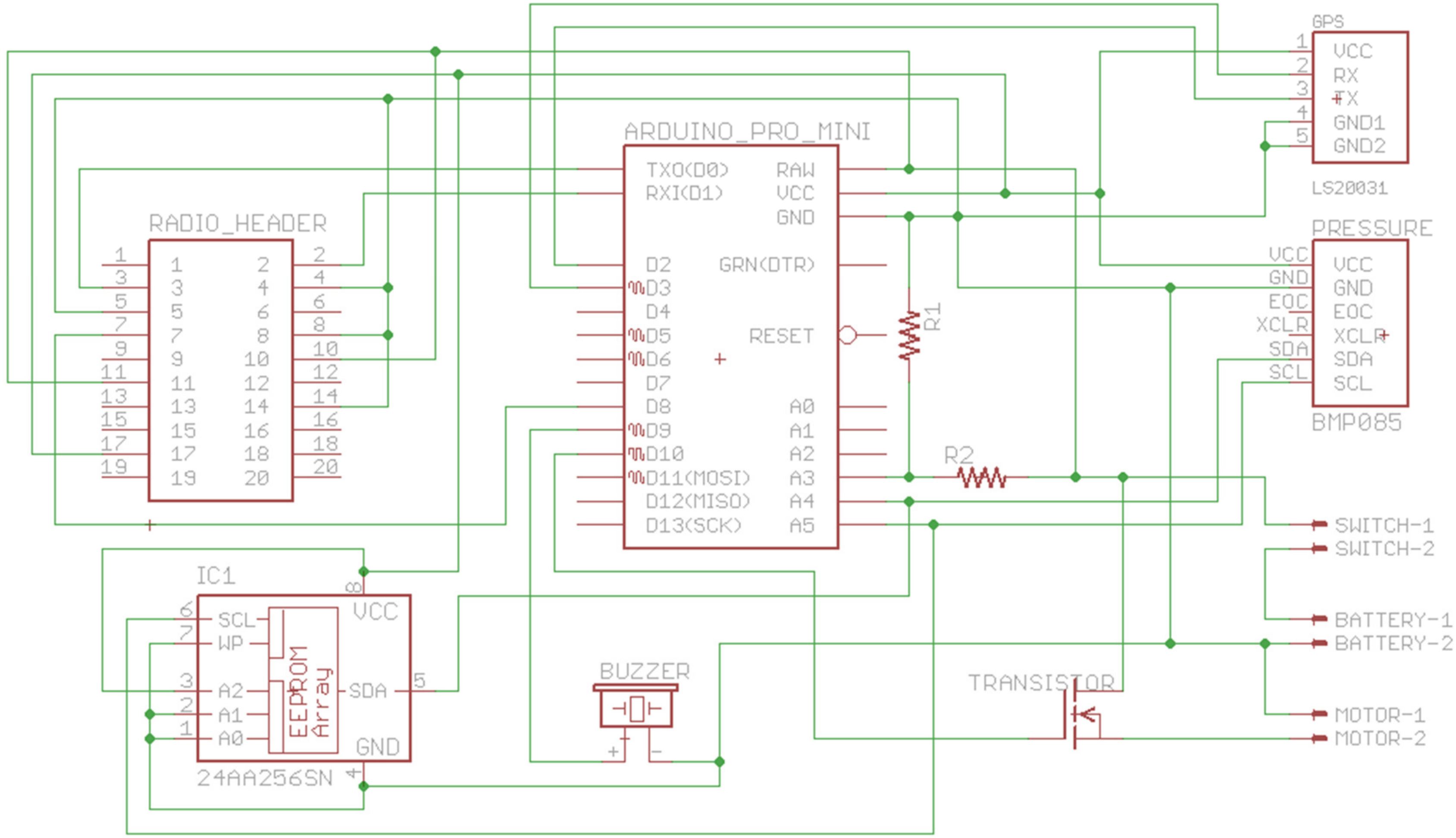
- **Carrier**
  - Removed external power indicator LED
  - Added onboard storage of data (EEPROM)
  - Connected Transceiver Vcc pin directly to battery
- **Lander**
  - Removed GPS
  - Removed external power indicator LED
  - Connected Transceiver Vcc pin directly to battery

# EPS Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| EPS-01 | Battery shall output at least 3.3 V for duration of flight | Arduino requires a 3.3 V input | High | None | None | | X | X | |
| EPS-02 | Battery shall be able to discharge a peak of 900 mA | When actuator is in use, current drawn will spike | Medium | None | None | | X | X | |
| EPS-03 | Battery shall be adequately light (less than 50 g at least) | Weight is a limiting factor in our design choice | Medium | SYS-01 | None | | X | | |
| EPS-04 | Battery shall have a high enough capacity to last the duration of the mission (350 mAh) | CanSat needs power for duration of mission | High | None | None | | X | | X |
| EPS-05 | Will use Arduino's ADC to measure battery voltage with resolution of 10 bits (3.2 mV) | Base Mission Requirement | Medium | None | None | | X | | X |
| EPS-07 | Carrier and lander shall have an external power control switch that turns on and off | Base Mission Requirement | High | SYS-07 | EPS-07a | | X | | |
| EPS-07a | On-off switch shall be connected directly to battery | To cut off carrier power | Medium | SYS-07, EPS-07a | None | | X | | |
| EPS-08 | Cannot use lithium polymer (LiPo) batteries | Base Mission Requirement | High | SYS-07 | None | | X | | |

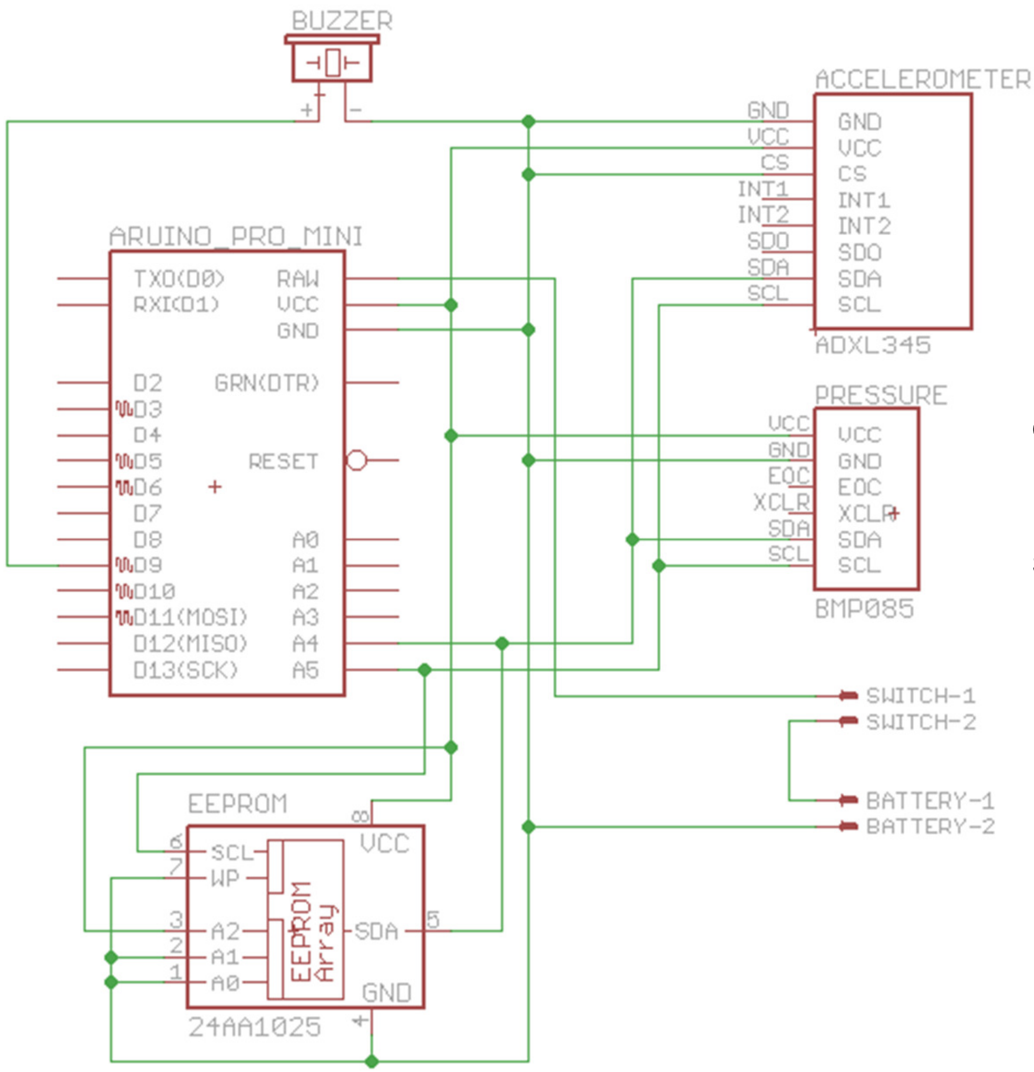
# Carrier Electrical Block Diagram

2011 Texas
Annual CanSat Competition




-***Raw*** pin corresponds to direct battery connection (3.8V).
-***Vcc*** pin corresponds to regulated 3.3V source.
-All of the other pins on the Arduino have a max of 3.3V

# Lander Electrical Block Diagram




-***Raw*** pin corresponds to direct battery connection (3.8V).

-***Vcc*** pin corresponds to regulated 3.3V source.

-All of the other pins on the Arduino have a max of 3.3V

-Pin D13 on the Arduino is an LED

# Carrier Power Budget

| Component | Current (mA) | Voltage (V) | Power (mW) | Expected Duty Cycle (Time on in min.) | Uncertainty (±%) | Total Energy Consumed* (mWh) | Source |
|---|---|---|---|---|---|---|---|
| Arduino | 25 | 3.3 | 82.50 | 60.00 | 20 | 82.50 | EST |
| Arduino I/O pins | 120 | 3.3 | 396.00 | 5.00 | 30 | 42.90 | DS |
| GPS | 41 | 3.3 | 135.30 | 5.00 | 10 | 11.28 | DS |
| Barometer | 0.007 | 3.3 | 0.02 | 5.00 | 10 | 0.00 | DS |
| EEPROM | 0.4 | 3.3 | 1.32 | 5.00 | 10 | 0.11 | DS |
| Transceiver | 108 | 3.3 | 356.40 | 5.00 | 10 | 29.70 | DS |
| Actuator | 550 | 3.7 | 2035.00 | 0.20 | 30 | 6.78 | DS |
| Buzzer | 8 | 3.3 | 26.40 | 60.00 | 10 | 26.40 | DS |
| **Total** | | | | | | **199.67** | |
| Available (mWh) | | | | | | 2775.00 | |
| Margin (mWh) | | | | | | 2572.33 | |

*Total Energy Consumed has been calculated as the $Total\ Energy * (1 + Uncertainty)$

# Lander Power Budget

| Component | Current (mA) | Voltage (V) | Power (mW) | Expected Duty Cycle (Time on in min.) | Uncertainty (±%) | Total Energy Consumed* (mWh) | Source |
|---|---|---|---|---|---|---|---|
| Arduino | 25 | 3.3 | 82.50 | 60.00 | 20 | 82.50 | EST |
| Arduino I/O pins | 120 | 3.3 | 396.00 | 5.00 | 30 | 42.90 | DS |
| Barometer | 0.007 | 3.3 | 0.02 | 5.00 | 10 | 0.00 | DS |
| EEPROM | 0.4 | 3.3 | 1.32 | 5.00 | 10 | 0.11 | DS |
| Accelerometer | 0.145 | 3.3 | 0.48 | 5.00 | 10 | 0.04 | DS |
| Buzzer | 8 | 3.3 | 26.40 | 60.00 | 20 | 26.40 | DS |
| **Total** | | | | | | **151.95** | |
| Available (mWh) | | | | | | 2775.00 | |
| Margin (mWh) | | | | | | 2623.05 | |

*Total Energy Consumed has been calculated as the $Total\ Energy * (1 + Uncertainty)$*

## Battery Chosen – Li-ion 14500

| Part | Price | Type | Capacity (mAH) | Weight (g) | Voltage (V) | Size (mm) |
|---|---|---|---|---|---|---|
| Li-ion 14500 pre-wired with PCB | $9.95 | Li-ion | 750 | 20 | 3.7 | 18 (diameter) x 54 |

- If the current drain approaches 2A, the output will go to zero.
- Will connect to circuit board with a polarized connector, to prevent negative voltage input to the system.
- Used for Lander and Carrier.
- Smallest package and weight available (within reason)

Photo courtesy of http://www.batteryspace.com/li-ion14500battery37v750mah277whbatteryprewiredwithpcb3alimit.aspx

# Battery Voltage Measurement

- We will be using the Arduino's ADC to measure the voltage, which has a resolution of 10 bits. $\left(\frac{3.3\ \text{V}}{2^{10}} = 3.22\ mV\right)$
- Will use high valued resistors (for low current drain) to make a simple voltage divider and bring the voltage down from 4.2 V max to 3 V max
- Resistor values will be 100 kΩ and 270 kΩ
- Will calculate battery voltage using the ratio:

$$\frac{V_{meas.}}{V_{battery}} = \frac{270\ k\Omega}{100\ k\Omega + 270\ k\Omega} = \frac{27}{37}$$

# Flight Software Design

Chris Stack

# FSW Overview

- **Common between Lander and Carrier**
  - Programming Languages
    - C/C++
  - Development Environment
    - Arduino IDE
- **FSW Architecture for Carrier**
  - Runtime loop: monitors time and requests updates at a rate of 0.5 Hz or greater, builds data string, which is stored onboard and also sent to ground station using AC4790, monitors altitude and initiates separation as well as disabling of radio and turning on buzzer
- **FSW Architecture for Lander**
  - Runtime loop: monitors time and requests updates at a rate of 0.5 Hz or greater. Builds data string from data and stores onboard for later retrieval. Monitors altitude to determine when to turn on and turn off accelerometer as well as buzzer.

# FSW Changes Since PDR

- No changes, progress is being made on integrating various sensors code

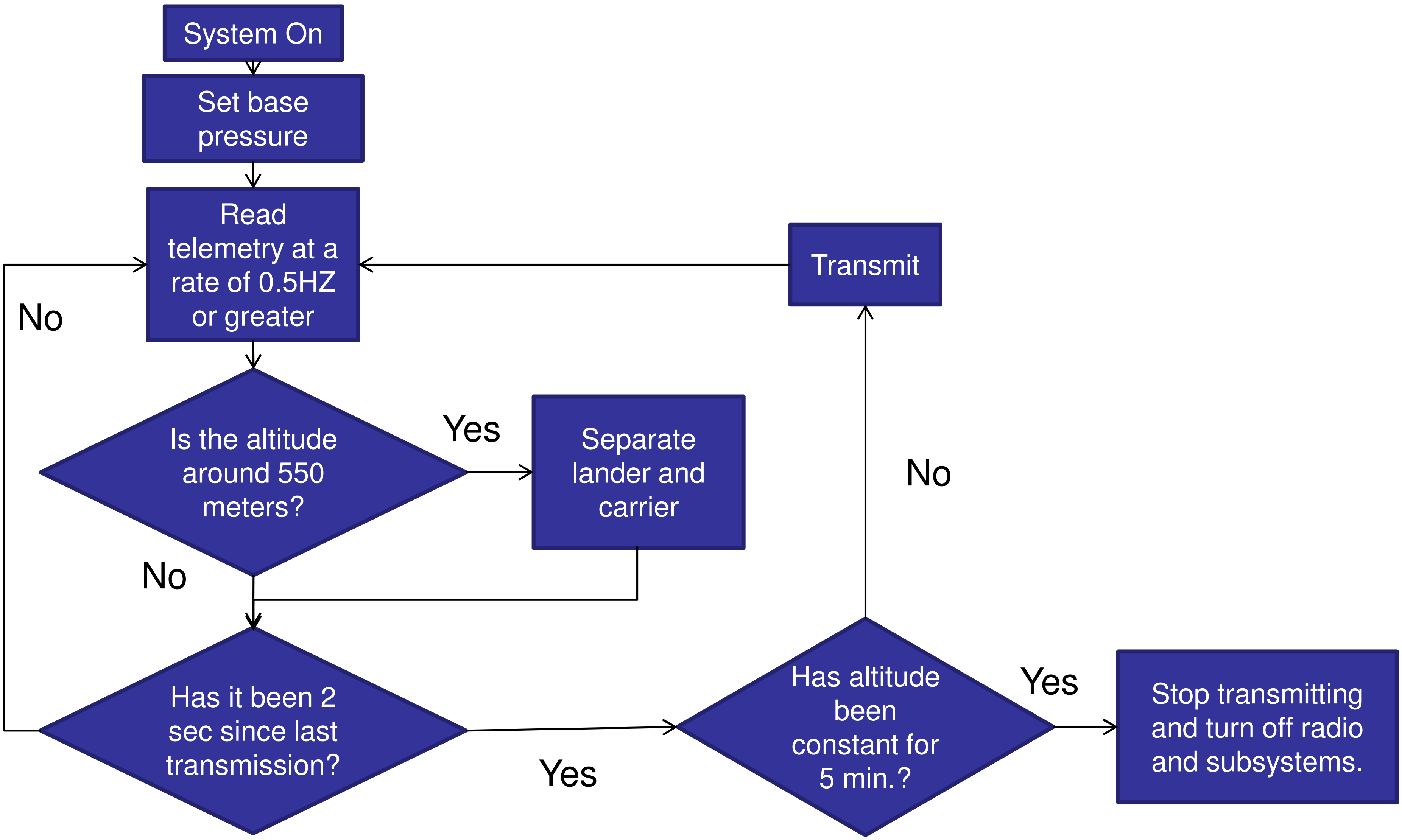
# FSW Requirements

| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| FSW-01 | For testing sample programs will be made to simulate significant events such as the need for separation and approaching ground to ensure the appropriate action is taken | Our separation mechanism and other things like it need to be tested before actual drop testing. | High | None | None | | | X | X |
| FSW-02 | Data will be sampled for all subsystems except for accelerometer at rate of at least 0.5 Hz | Transmission must occur every 0.5 Hz so this will ensure new data for all subsystems | Medium | SSD-03 | None | | X | | X |
| FSW-03 | Baud rate of 57600 bps | This is the optimal baud rate for both the GPS and transceiver. | Low | None | None | | X | | |
| FSW-04 | Data will be stored using an external memory module through $I^2C$ protocol | The Arduino microcontroller does not have enough on-board memory so we have selected an external module. | Low | None | None | | X | X | |
| FSW-05 | The carrier will interface with the ground station through transceiver uplink. | Base Mission Requirement. | Low | None | None | | X | | X |

# Carrier Cansat FSW Overview

- **GPS**
  - Update rate 1 Hz
  - Serial hardware interface
  - NewSoftSerial Library used for connection
- **Barometer**
  - Update rate 1 Hz
  - $I^2C$ hardware interface
- **Voltage Divider**
  - Update rate of 1 Hz
  - Connected to an analog pin
- **Transceiver**
  - Transmit at a rate of 0.5 HZ
  - Serial interface
  - Throughput: 38 bytes per second
- **EEPROM**
  - Data stored at a rate of 0.5 HZ
  - Throughput 38 bytes per second

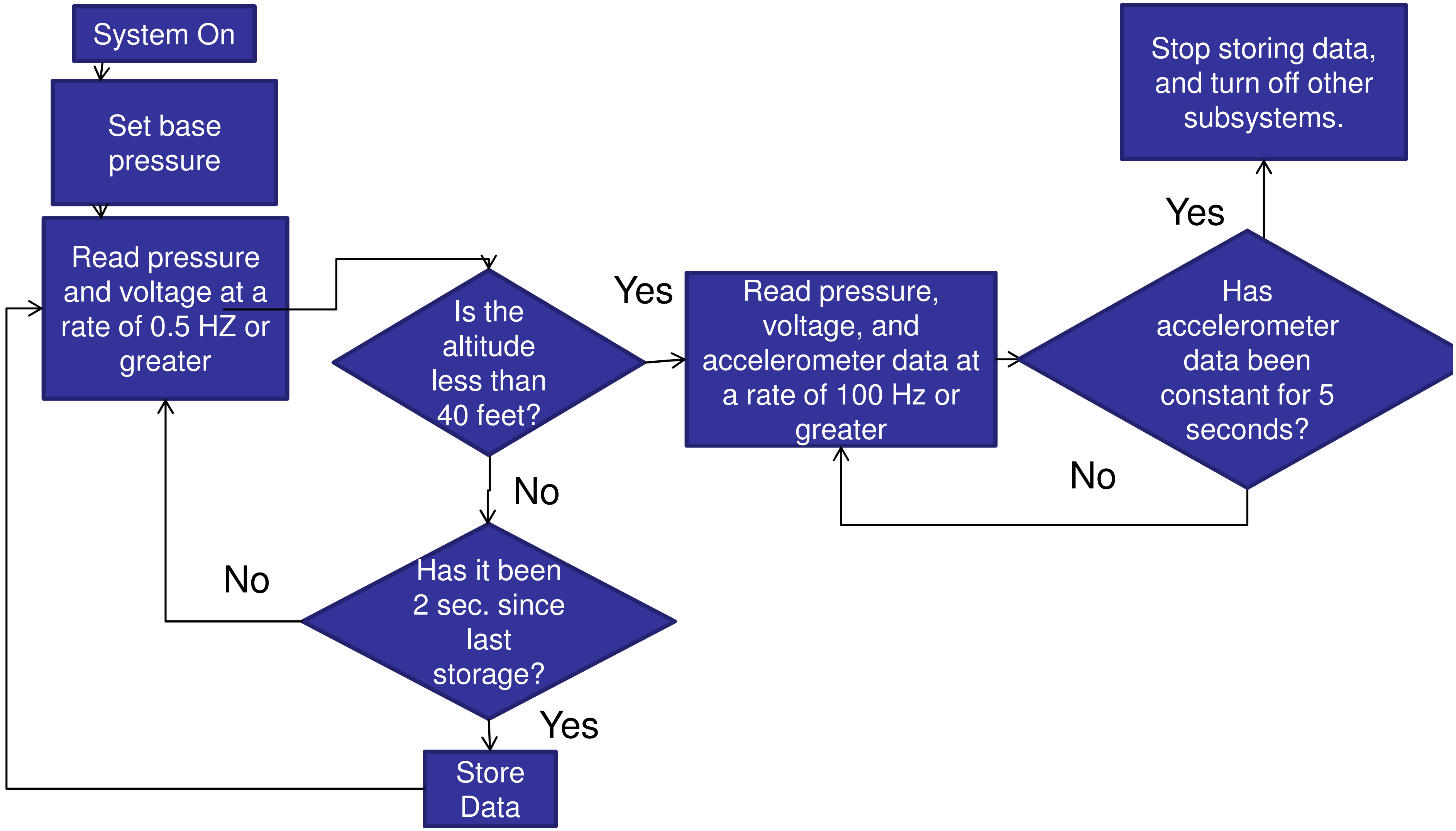
# Carrier Software Flow Diagram or Pseudocode



# Lander Cansat FSW Overview

- **Barometer**
  - Update rate  1 Hz
  - $I^2C$ hardware interface
- **Voltage Divider**
  - Update rate of 1 Hz
  - Connected to an analog pin
- **Accelerometer**
  - Update rate of 100 Hz
  - $I^2C$ hardware interface
- **EEPROM**
  - Data stored at a rate of 0.5 HZ
  - Throughput – 600 bytes per second minimum

# Lander Flow Diagram or Psuedocode



# Ground Control System Design

**Chris Jennette**

# GCS Overview

- Antenna receives signal from carrier
- Received by transceiver
- Data transferred to computer from transceiver over an FTDI cable
- Data processed, displayed, and stored by GCS

# GCS Requirements



| ID | Requirement | Rationale | Priority | Parent(s) | Child(ren) | VM | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | A | I | T | D |
| GCS-01 | GCS must be able to parse and graph data from carrier and lander. | Mission requirement to display real-time data. | Medium | None | None | | | | X |
| GCS-02 | Antenna must be free of interference. | Interference would cause signal loss and could cause loss of communication | Low | None | None | | | | X |
| GCS-03 | Antenna will be placed to aim in the direction the wind is blowing to ensure coverage during entire flight. | Current selection of a directional antenna means more care needs to be taken in antenna placement. | Low | None | None | X | X | | X |
| GCS-04 | GCS antenna shall be elevated a minimum of 3.5 m from the ground level | Base Mission Requirement | High | SYS-04 | None | | X | | |
| GCS-05 | GCS will verify that transmissions have stopped from the carrier 5 minutes after landing | Base Mission Requirement | High | SYS-04 | None | | X | | |
| GCS-06 | Must have own ground station | Base Mission Requirement | High | SYS-04 | None | X | | | X |

## Chose Omni-Directional SG101N-915 Antenna

- Typical Gain of 5dB
- Antenna will be angled away from launch site to account for drop in coverage at 90°
  - Optimum angle for antenna can be found with the following equation: $\theta = \cos^{-1}\left(\frac{x}{400}\right)$, where $x$ is the distance $(\text{in } m)$ from the launch site

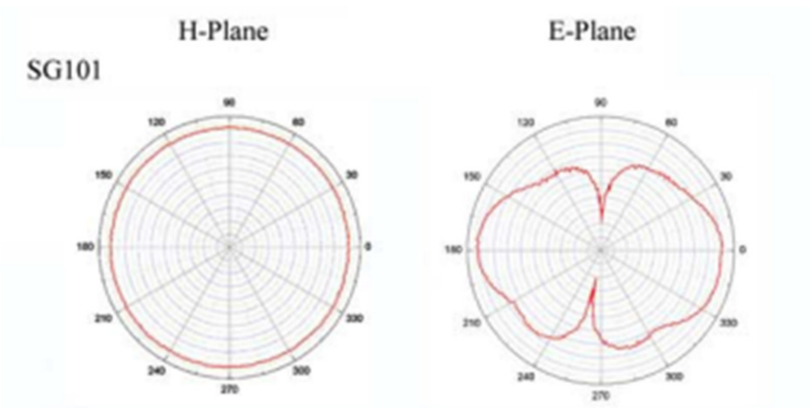
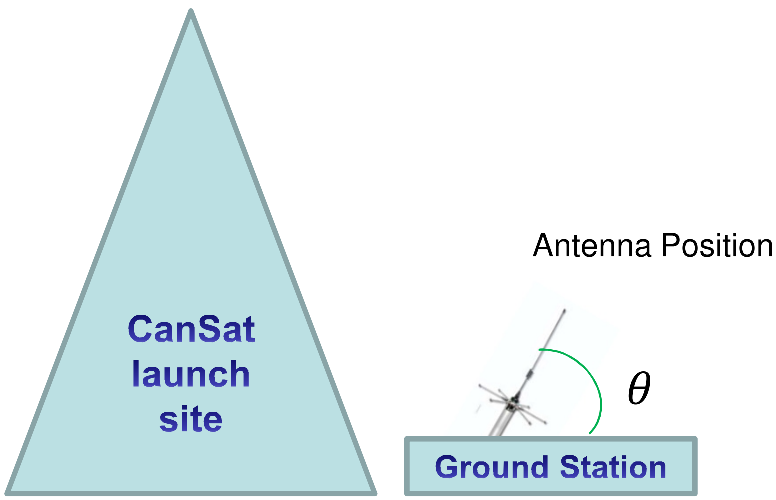



Photo Courtesy of
http://media.digikey.com/pdf/Data%20Sheets/Nearson%20PDFs/SG%20Series.pdf



# GCS Antenna Selection

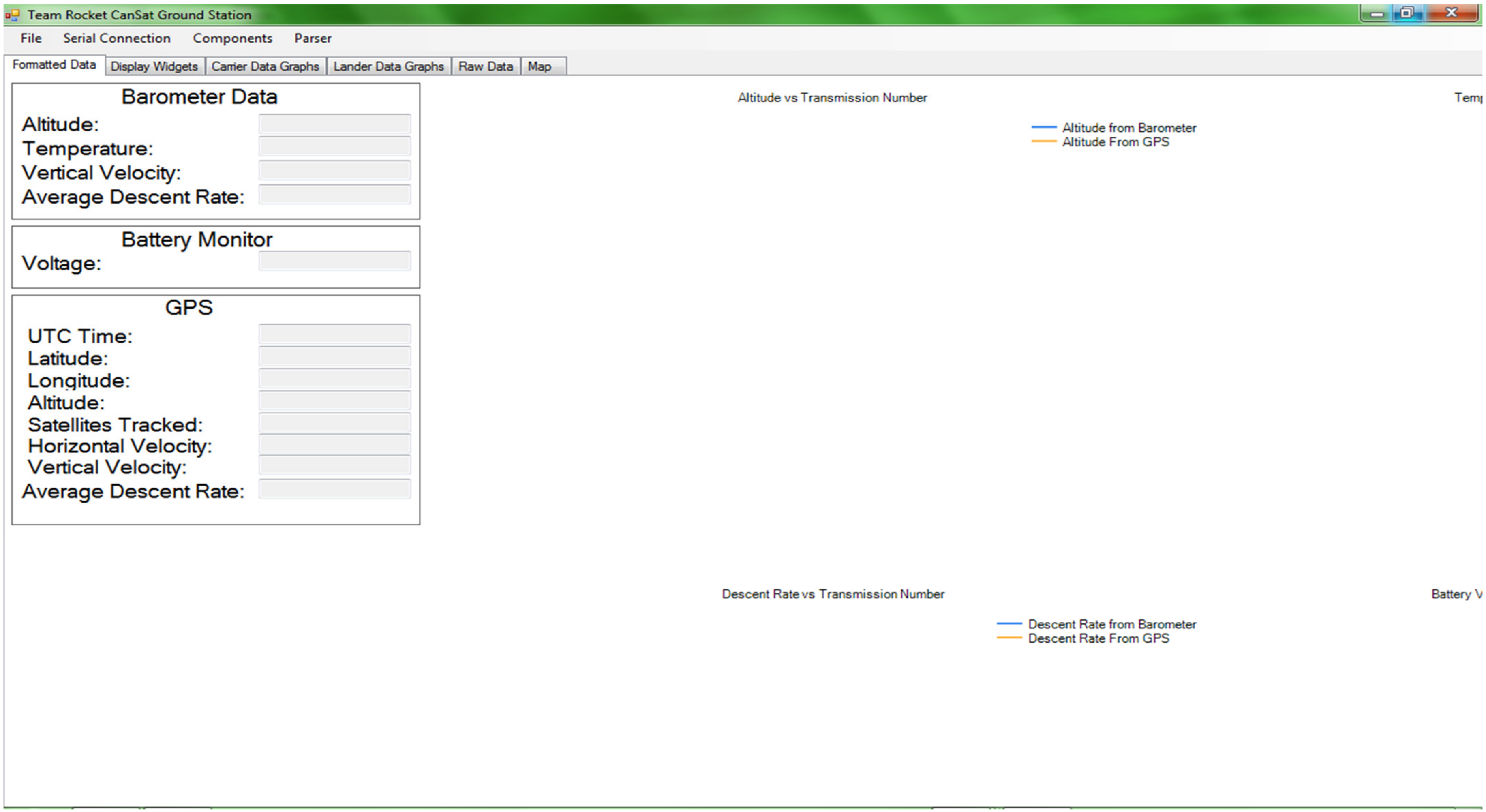
- **Link Budget**
  - Calculated as: $P_{RX} = P_{TX} + G_{TX} + G_{RX} - L_{RX} - 20\log\left(\frac{4\pi * d}{\lambda}\right)$

$$\boldsymbol{P_{RX}} = 10\log(100mW) - 0.5dB + 5dB - 1dB - 90.61dB = \boldsymbol{-67.11dBm}$$

  - The sensitivity of the Laird AC-4790 module is *-100dBm*, which leaves plenty of room for unplanned propagation losses.

# GCS Antenna Selection

- **Ground Control Software**



# CanSat Integration and Test

**Stephanie Butron**

# CanSat Integration and Test Overview

**1. Communication**
- First system because vital to other subsystems

**2. Sensors**
- First subsystem to be integrated because relied upon by FSW
- Date: 4/2/2011

**3. FSW**
- Second system to be integrated because needed to monitor subsystems
- Date: 4/9/2011

**4. Electrical**
- Third system to be integrated so structural assembly can begin
- Date: 4/16/2011

**5. Mechanical**
- Fourth system to be integrated to allow for descent control
- Date: 4/23/2011

**6. Descent Control**
- Fifth system to be integrated for full-scale testing
- Date: 5/2/2011

**7. Ground Control**
- Last system to be integrated because not located within CanSat
- Date: 5/14/2011

# CanSat Integration and Test Overview

**Structural Assembly of Lander**

1) All interior electronics are assembled onto the PCB board, which will be horizontally mounted on a carbon fiber disk and held in place by a mounting plate
2) Four carbon fiber rods will be slid into four pre-cut holes on the mounted disk and fixed in place by epoxy
3) Two carbon-fiber disks will then be attached to the free-ends of the four rods and sealed with epoxy
4) Attach the on/off switch to the top disk (outside disk closest to electronics)
5) Attach the nut to the external face of the top disk
6) The Lander module is slid into the outer plastic shell (tennis-ball container)

**Structural Assembly of Carrier**

1) The interior electronic components are assembled onto another PCB board. The board will be vertically mounted onto both of the disks with L-brackets on two of the four carbon fiber rods
2) Mount the motor to the bottom disk with epoxy and attach the on/off switch to the top disk with epoxy
3) Attach the screw to the motor and attach the shielding around the screw to the external face of that disk
4) Slide the Carrier module into it's corresponding outer shell

**Structural Assembly of CanSat**

1) Carrier Module fitted into pre-indented holes on the Lander module's top disk
2) The screw is threaded into the nut on that same disk of the Lander module

# CanSat Integration and Test Overview

| What to Accomplish | Constraints | Equipment Used | Procedure | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| • Vibration testing | •N/A | • Integrated CanSat<br>•Shaker | • Test at various frequencies to simulate rocket vibrations | • CanSat cannot separate<br>•Must maintain electrical connections<br>•Structure shall not come apart | • Test to be completed at end of April |
| •CanSat can deploy from rocket | •CanSat will not be under vibrational effects | •Payload cylinder<br>•Integrated CanSat | •Test at various speeds to see if CanSat can deploy from payload | •CanSat deploys out of payload cylinder | •Test to be completed at end of April |
| •Full scale CanSat drop test | •Not at 1000 m | •CanSat<br>•Ground station equipment | •Drop CanSat from high enough distance to test separation and descent rates | •CanSat separates and descent rates are within correct bounds | •Test to be completed in early May |

# Sensor Subsystem Testing Overview

| What to Accomplish | Constraints | Equipment Used | Procedure | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| • Verify whether GPS sensor communicates with microprocessor | • Only tested with high baud rates | • Arduino Pro-Mini microprocessor, GPS, Laptop Computer, & LED | • Tested sending basic messages between Arduino and GPS | • GPS data transmitted from Arduino to computer confirms Arduino's ability to deliver/receive data | • The Arduino's maximum baud rate from GPS is 14400<br>• GPS successfully delivered/received data from Arduino to computer with update rate (1 Hz) and software serial |

# Lander Impact Force Sensor Testing

| What to Accomplish | Equipment Used | Procedure | Results |
|---|---|---|---|
| •Test working code for accelerometer | •Arduino | •Develop code for accelerometer<br>•Upload code and monitor output of accelerometer | Code for accelerometer is working properly, and properly outputs in the (x,y,z) |
| •Once sensors have been mounted, test impact of Lander by estimating descent rate | •Complete Lander System | •Drop Lander at pre-determined height to test data gathered during impact | To be conducted 4/3/11 |

# DCS Subsystem Testing Overview

| What to Accomplish | Equipment Used | Procedure | Constraints | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| • Determine the descent rate of the lander using the SkyAngle parachute | •SkyAngle Parachute<br>• Fishing weights<br>•Tennis ball can<br>•Duct tape<br>•Rope<br>•Measuring tape<br>•Stopwatch | •Use the rope to measure height<br>•Place fishing weights in tennis ball can for approximation of lander weight<br>•Use duct tape to attach parachute to top of the can while swivel is free to spin<br>•Drop prototype and measure time it takes to land | •Controlled indoor environment does not account for wind effect in outdoor environment | •Ability to land at a descent rate of 5.5m/s +/- 1 m/s | •Landing speed about 7 m/s<br>•Height of test is insufficient to definitively fail the descent control system |
| •Determine the drift distance of the lander in the presence of wind | •Rope<br>•Measuring tape | •Measure distance from the point the rope perpendicularly touches the ground to where load lands | •Controlled indoor environment does not account for wind effect in outdoor environment | N/A | •Drift displacement is 6 ft<br>•Unreliable result since parking garage blocks wind |
| •Drop testing from a bridge<br>•Simulate the weight configurations of the lander, carrier, or both<br>• Measure the descent rate, drift distance and probability of parachute deployment due to different packing configurations<br>•Examine safety of egg | •Rope<br>•Grade A Egg<br>•Fishing weights<br>•Racket-Ball can<br>•Duct tape<br>•SkyAngle Parachute<br>•Stopwatch<br>•Fishing Pole | •Use rope to measure the height of the test<br>•Simulate the weight of each component using fishing weights<br>•Use duct tape to attach the parachute to the ball can<br>•Mark point right underneath the release point<br>•Record packing volume folding technique | •Controlled indoor environment does not account for wind effect in outdoor environment | •Lander descent rate of 5.5m/s +/- 1m/s<br>•Carrier descent rate of 4m/s +/- 1 m/s<br>•No cracks or breakage of egg | •To be completed 4/1/11 |

# Mechanical Subsystem Testing Overview

| What to Accomplish | Equipment Used | Procedure | Constraints | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| •Determine weight range motor shaft will be able to support under free fall conditions | •Actuator<br>•Wooden Platform<br>•Fastening Screw<br>•Fishing Weights<br>•Super-Glue<br>•Racquet-Ball Container | •Use Super-Glue adhesive to stick motor to wooden platform, which creates motor shaft<br>•Placed 750 g fishing weight in 50 g intervals in racquetball container held by the motor shaft Mechanism | •Controlled indoor environment does not account for wind effect in outdoor environment | •Capability to hold load without fracture of adhesive and plastic screw at maximum load (750 g) | •The motor shaft holds container load up to the maximum weight (750 g) |
| •To validate the motor performance of the motor-shaft | •Stopwatch | • Test apparatus with maximum weight reached in previous objective during specified amount of time (10 minutes) | •Controlled indoor environment does not account for wind effect in outdoor Environment | •Motor shaft capability to hold the load in a span of 10 minutes | •The motor shaft held the load without adhesive breakage past 10 minutes |
| •Determine how much cushion should be available in egg compartment too withstand impact force | •Rope<br>•Grade A Egg<br>•Racquetball container<br>•Fishing weights<br>•Camera | •Tied the prototype to rope<br>•Cushioned egg with different materials for each drop<br>•Dropped load 41 feet | •No control over orientation of the way the carrier will land (corner, base, etc.) | •Success if egg shell does not break or crack | •Test scheduled to be conducted 4/1/2011 |

# Mechanical Subsystem Testing Overview: Motor Shaft Testing

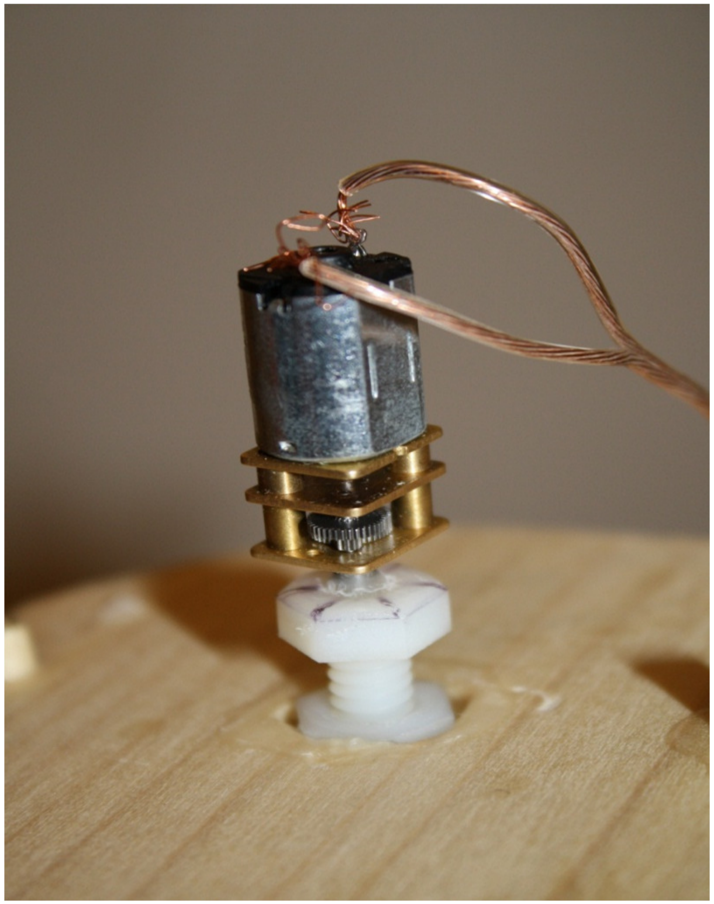


**Figure 1. The motor is attached to the plastic screw through super glue adhesive while screwed in to the wooden platform**

**Figure 2. The Motor shaft is connected to the racquet-ball container shell through 4 dowel rods**

# CDH Subsystem Testing Overview

| What to Accomplish | Constraints | Equipment Used | Procedure | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| •Verify communication between the Arduino and ground station using API format through the transceiver | N/A | •Laptop Computer<br>•Transceiver<br>•Arduino Pro-Mini Microprocessor<br>•Ground-station Software | •Tested sending basic messages between Arduino and ground station | •Basic messages delivered and received between Arduino and ground station | •Tests were successful<br>•Still need to range test and parse data out of ground station version |
| •Verify functionality of the Arduino micro-processor | N/A | •Arduino Pro-mini microprocessor<br>• FTDI Cable<br>•Breadboard<br>•Wires<br>•LED<br>•Pins<br>•Laptop Computer | •Soldered pins to Arduino and tested using FTDI cable | •Verify ability for the Arduino to deliver/receive data by receiving messages on a computer | •Successfully received data from Arduino on computer |

# EPS Testing Overview

| What to Accomplish | Constraints | Equipment Used | Procedure | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| •Determine the minimum starting and operating current for the motor | N/A | •Potentiometer<br>•Lithium-Ion battery<br>•Multi-meter | • Used trim potentiometer in series with the Lithium-Ion battery (@3.8 V) to test minimum starting current and minimum operating current | •Display of current output ensures system is functional | • Minimum starting current is 156 mA (trim pot @ 24.5 Ω)<br>•Minimum operating current is 26.3 mA (trim pot @ 145 Ω) |
| •Determine battery life at a specific drain rate | •Power rating of resistors used is low (didn't handle full amount of current) | •Lithium-Ion battery<br>•Resistors<br>•Multi-meter | •Drained battery using 4-100 ohm resistors in parallel<br>•Used multiple resistors to minimize current through each one | •Carrier requires 60 mAH, so battery needs to supply at least 80 mAH | •Equivalent resistance was 25 Ω, which gave a current drain of ≈148 mA<br>•Battery lasted about 4 hours at this drain rate, giving a total of 592 mAH |
| •To measure the battery voltage via Arduino ADC | N/A | •Arduino<br>•Battery<br>•Resistors<br>•Multimeter | •Connected output of voltage divider to Arduino's ADC and converted the measured value to a voltage level<br>•Took several averages<br>•Used multi-meter to confirm accuracy of Arduino's ADC | •Value correctly read by ADC | •Voltage read correctly<br>•Will perform conversions necessary on ground station in order to conserve computing power in carrier<br>•Use floating point data type |

# FSW Testing Overview

| What to Accomplish | Constraints | Equipment Used | Procedure | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| •Verify communication between the Arduino and ground station using API format through the transceiver | N/A | •Laptop Computer<br>•Transceiver<br>•Arduino Pro-Mini microprocessor<br>•Ground station Software | •Tested sending basic messages between Arduino and ground station | •Basic messages must be delivered and received between Arduino and ground station | •Tests successful<br>•Still need to range test and parse data out of ground station version |

# GCS Testing Overview

| What to Accomplish | Constraints | Equipment Used | Procedure | Pass/Fail Criteria | Results |
|---|---|---|---|---|---|
| •Verify the GCS can receive and parse serial data<br>•Display and graph the data | •Until FSW is fully working, only able to test with simulated data | •Laptop and Arduino now. Eventually laptop, transceiver and antenna. | •Tested sending hex data to the computer from the Arduino, and parsing the received data into readable format | •Verify that the GCS correctly parses the serial data into readable results | •The GCS was able to successfully display the serial data |

# Mission Operations & Analysis

**Younes Taleb**

# Overview of Mission Sequence of Events

- **Pre-launch Sequence of Events**

| | |
|---|---|
| Arrival at Launch Site | • Set up ground station<br>• Connect communication devices<br>• Make any last minute assemblies to the CanSat<br>• Measure wind speeds for landing estimates |
| Pre-Launch Checks | • Go through pre-flight checklist<br>  • Check subsystems<br>  • Ground station operational<br>  • Antenna positioned correctly<br>• Estimate of landing zone made |
| Integration into Rocket | • Slide modules into CanSat<br>• Fit egg into lander<br>• Power CanSat online<br>• Integrate CanSat into rocket payload |

# Overview of Mission Sequence of Events

- **Launch Sequence of Events**

| | |
|---|---|
| Rocket Launch | • Begin communications with CanSat<br>• Receive telemetry |
| Deployment | • Carrier parachute deployed<br>• Pressure and temperature data is acquired and plotted |
| Separation | • An approximation of 500 m initiates separation<br>• Lander parachute deployed |
| Landing | • Audible beacons start on both carrier and lander upon landing<br>• Impact force recorded by the lander |

# Overview of Mission Sequence of Events

- **After-Launch Sequence of Events**

**Recovery**
- Carrier and lander are retrieved
- Egg is checked for damage

**Analysis**
- Data is retrieved from the lander via USB cable and analyzed
- Telemetered data from the carrier is analyzed

**Post Flight Analysis**
- Presentation is created from results of the flight
- Presentation is given the next day

# Lander Landing Coordinate Prediction

- **Wind speed/direction algorithm will be used to calculate landing coordinates.**
- **Code will take into account values of:**
  - A preliminary prediction can be made using Wind forecast data from NOAA to estimate a trajectory. This code is scripted to work with a Google maps web interface.
  - The primary prediction tool will utilize telemetered GPS data from the carrier, wind speed data and descent rate to solve for the drifting of the lander.

Sources:
http://habhub.org/predict/
http://www.srcf.ucam.org/cuspaceflight/wiki/doku.php?id=prediction#landing_prediction

# CanSat Location and Recover

- **The carrier and lander will be equipped with a 100 dB buzzer**
  – Will operate for 60 minutes
  – Allow for recovery even if the CanSat lands in a remote location
- **Parachute drift algorithm will be used to estimate landing location**
- **After recovery, the telemetry data will be retrieved from the lander via a USB cable**

# Mission Rehearsal Activities

- **Ground system radio**
  - When: 2/26
  - Results: Can communicate status message between a simulated carrier and the ground station.
- **Loading the egg payload**
  - Rehearsed during egg drop tests (2/12)
  - Results: Placing the egg horizontally is the best configuration to avoid breaking.
- **Powering the CanSat on/off**
  - When: 3/14
  - Results: Powered the Arduino using a battery and switch
- **Launch configuration preparations**
  - Rehearsed during the drop test (3/30)
  - Will determine the ideal packing of the parachute and placement of electronics to maximize volume.

# Mission Rehearsal Activities

- **Loading the CanSat into the launch vehicle**
  - When: 4/25
  - Will be rehearsed during the test rocket launch conducted at Virginia Tech.
- **Telemetry processing, archiving, and analysis**
  - When: 3/28
  - Results: Processing GPS barometer data and voltage divider
- **Recovery**
  - When: 3/30
  - Will test the audible range of the buzzer

# Management

**Stephanie Butron**

# Status of Procurements

| Part | Quantity | Order Date | Status |
|---|---|---|---|
| Microprocessor | 2 | 1/21/2011 | Received |
| Carrier Parachute | 1 | 1/21/2011 | Received |
| RF Module | 1 | 1/23/2011 | Received |
| Buzzer | 2 | 1/23/2011 | Received |
| GPS | 1 | 1/23/2011 | Received |
| Antenna | 1 | 1/23/2011 | Received |
| Motor | 1 | 2/06/2011 | Received |
| Barometric Pressure Sensor | 1 | 2/06/2011 | Received |
| Accelerometer | 1 | 2/06/2011 | Received |
| Battery | 2 | 2/06/2011 | Received |
| Battery Charger | 2 | 2/06/2011 | Received |
| EEPROM | 1 | 2/28/2011 | Received |
| Transistor | 2 | 2/28/2011 | Received |
| Wire Housing | 2 | 3/14/2011 | Received |
| Break-Away Headers | 6 | 3/14/2011 | Received |
| Lander Parachute | 1 | TBD | Not Ordered |
| Cables for GS Antenna | 2 | TBD | Not Ordered |

# Budget - Hardware

| Component | Model Name | Quantity | Unit Cost | S/H | Price Definition |
|---|---|---|---|---|---|
| Battery | Li-Ion 14500 Battery | 2 | $19.90 | $10.35 | Actual |
| Battery Charger | Smart Charger for 3.7 V Li-ion Battery | 1 | $4.28 | | Actual |
| Transceiver | 20CIR VERT RECPT | 2 | $3.76 | $7.52 | Actual |
| Break Away Headers | Break Away Headers-Right Angle | 2 | $1.95 | $7.51 | Actual |
| Break Away Headers | Break Away Headers-Straight | 2 | $2.50 | | Actual |
| Break Away Headers | Break Away Headers-Female | 2 | $1.50 | | Actual |
| | Microchip 24AA1025 | 1 | $12.95 | $15.07 | Actual |
| Buzzer | 668-1028-ND | 2 | $4.51 | | Actual |
| Accelerometer | Triple Axis Accelerometer Breakout - ADXL345 | 1 | $27.95 | $9.28 | Actual |

# Budget - Hardware

| Component | Model Name | Quantity | Unit Cost | S/H | Price Definition |
|---|---|---|---|---|---|
| Microcontroller | Arduino Pro Mini 328 - 3.3V/8MHz | 2 | $18.95 | $28.03 | Actual |
| Barometric Pressure Sensor | BMP085 Breakout | 2 | $19.95 | | Actual |
| Decent Control | 56:1 Micro Geared Motor | 1 | $18.99 | $5.60 | Actual |
| Parachute | 20" Sky Angle Parachute | 1 | $31.95 | $11.00 | Actual |
| Antenna | Antenna OUTDR OMNIDIR 915MHZ STR | 1 | $52.50 | $8.86 | Actual |
| GPS | 32 Channel LS20031 GPS 5Hz Receiver | 1 | $59.95 | $8.44 | Actual |
| Hardware | Miscellaneous | 1 | $58.41 | | Budgeted |
| | | | | | |
| Cost of Components | | | $413.02 | | |
| S/H | | | $111.66 | | |
| **Total Cost of Components** | | | $524.68 | | |

# Budget – Other Costs

## Transportation/Hotel/Food Costs

| Description | Cost | Price Definition |
|---|---|---|
| Van Rental: Virginia Tech Fleet Services (12 Passenger Van) | $520 | Budgeted |
| Hotel | $400 | Estimated |
| Food | $750 | Estimated |
| **SUBTOTAL** | **$1,670** | |

*Note*: Transportation cost covers all three Virginia Tech CanSat Teams

## Ground Control Station Costs

| Description | Cost | Price Definition |
|---|---|---|
| Laptop Computer | $0 | Estimated |
| Arduino Software | $0 | Estimated |
| GCS Software | $0 | Estimated |
| Transceiver Software | $0 | Estimated |
| Miscellaneous | $135 | Estimated |
| **SUBTOTAL** | **$135** | |

# Budget – Total Cost & Income

## Total Cost

| Description | Price |
|---|---|
| Hardware | $524.68 |
| Transportation/Hotel/Food Costs | $1,670 |
| Ground Control Station Costs | $135 |
| **Total Cost Overall** | **$2,329.68** |

## Income

| Income Source | Date | Amount |
|---|---|---|
| SEC | Fall 2010 | $400 |
| SEC | Spring 2011 | $350 |
| Slush Funds | Spring 2011 | $933 |
| SEC | Spring 2011 | TBA |
| **CURRENT TOTAL INCOME** | **$1,683.33** | |

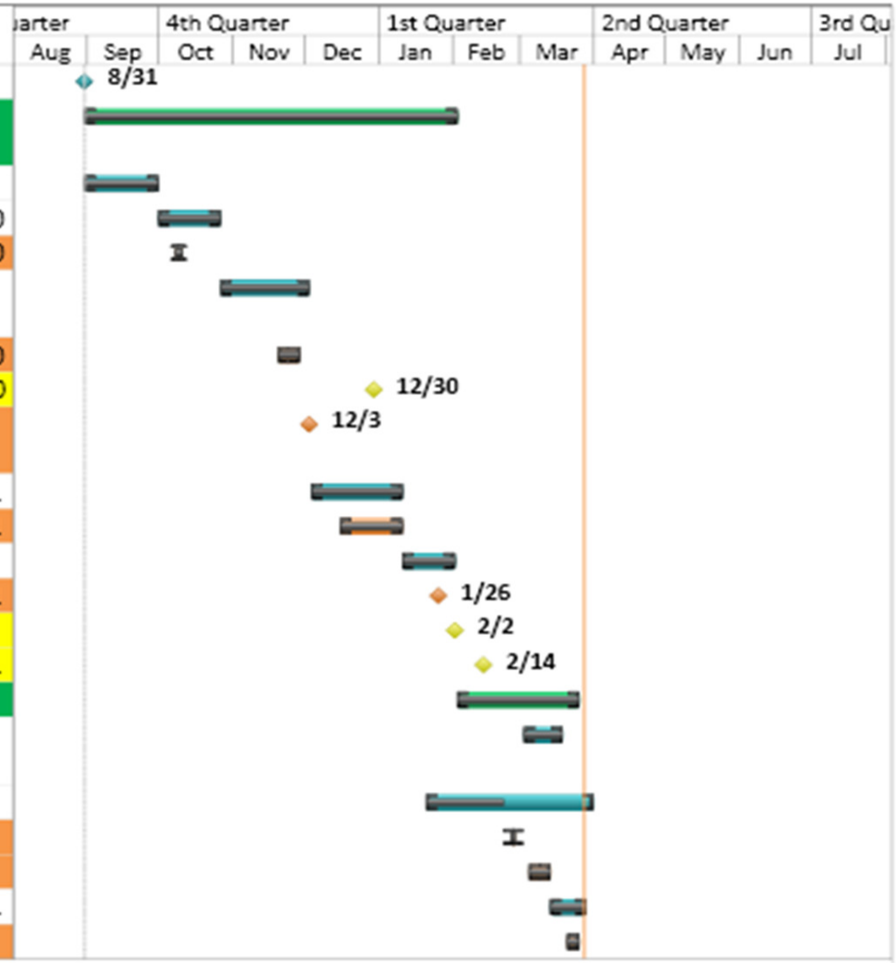
*Note: Slush Funds originate from Virginia Tech's Division of Student Affairs and SEC funds originate from Student Engineering Council*

CANS@VT TEAM ROCKET

# Schedule

2011 TEXAS ANNUAL CANSAT COMPETITION

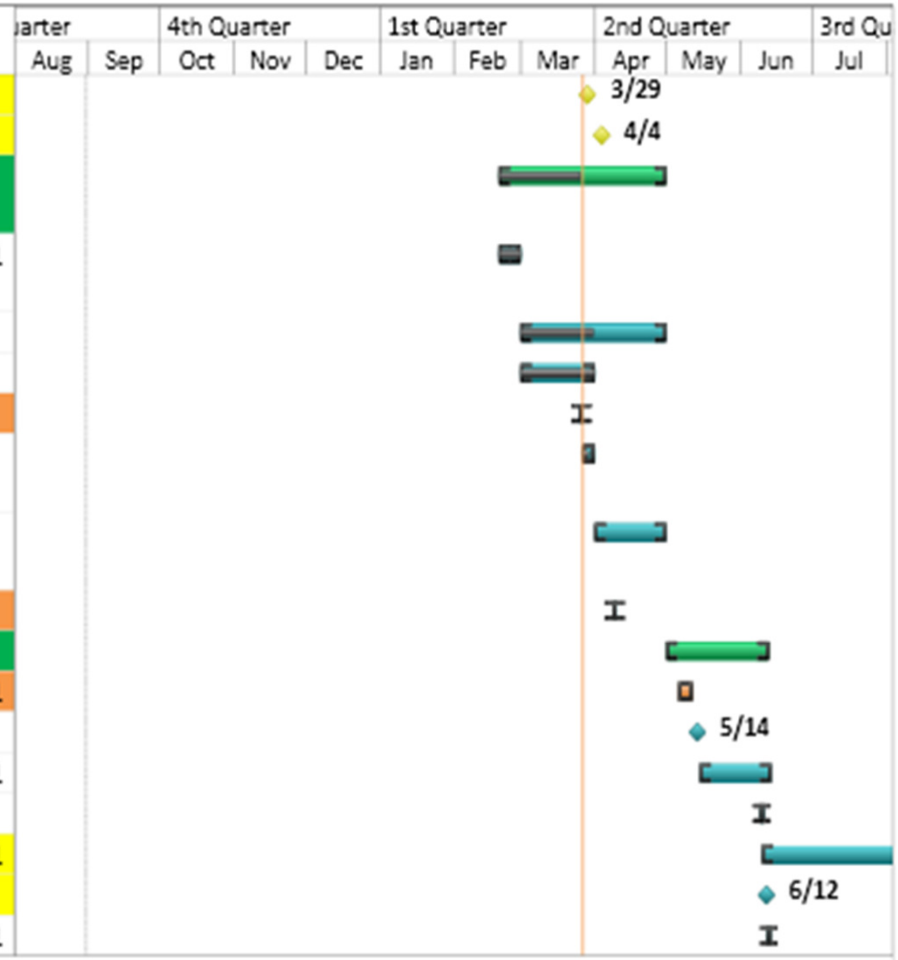
| ID | | Task Name | Duration | Start | Finish |
|---|---|---|---|---|---|
| 1 | ✓ | Start Project | 0 days | Tue 8/31/10 | Tue 8/31/10 |
| 2 | ✓ | Preliminary Design Phase | 112 days | Tue 8/31/10 | Wed 2/2/11 |
| 3 | ✓ | Project Definition | 23 days | Tue 8/31/10 | Thu 9/30/10 |
| 4 | ✓ | Conceptual Study | 18 days | Fri 10/1/10 | Tue 10/26/10 |
| 5 | ✓ | VT Fall Break | 2 days | Fri 10/8/10 | Sun 10/10/10 |
| 6 | ✓ | Preliminary Design Analysis | 27 days | Wed 10/27/10 | Thu 12/2/10 |
| 7 | ✓ | VT Thanksgiving Break | 7 days | Sat 11/20/10 | Sun 11/28/10 |
| 8 | ✓ | Applications due | 0 days | Thu 12/30/10 | Thu 12/30/10 |
| 9 | ✓ | VT AOE Intial Design Review | 0 days | Fri 12/3/10 | Fri 12/3/10 |
| 10 | ✓ | Design Improvement | 27 days | Sat 12/4/10 | Mon 1/10/11 |
| 11 | ✓ | VT Winter Break | 18 days | Thu 12/16/10 | Mon 1/10/11 |
| 12 | ✓ | Assemble PDR | 16 days | Tue 1/11/11 | Tue 2/1/11 |
| 13 | ✓ | VT AOE PDR | 0 days | Wed 1/26/11 | Wed 1/26/11 |
| 14 | ✓ | PDR Slides due | 0 days | Wed 2/2/11 | Wed 2/2/11 |
| 15 | ✓ | PDR Presentation | 0 days | Mon 2/14/11 | Mon 2/14/11 |
| 16 | ✓ | Critical Design Phase | 37 days | Thu 2/3/11 | Fri 3/25/11 |
| 17 | ✓ | Finial Design Improvement | 12 days | Thu 3/3/11 | Fri 3/18/11 |
| 18 | | Procurement | 50 days | Fri 1/21/11 | Thu 3/31/11 |
| 19 | ✓ | VT Academic Outreach | 1 day | Sat 2/26/11 | Sat 2/26/11 |
| 20 | ✓ | VT Spring Break | 7 days | Sat 3/5/11 | Sun 3/13/11 |
| 21 | ✓ | Assemble CDR | 11 days | Mon 3/14/11 | Mon 3/28/11 |
| 22 | ✓ | VT E-week | 5 days | Mon 3/21/11 | Fri 3/25/11 |




**Design Phase** | **Competition Task** | **Virginia Tech Task**

# Schedule

| ID | | Task Name | Duration | Start | Finish |
|---|---|---|---|---|---|
| 23 | ✓ | CDR Slides due | 0 days | Tue 3/29/11 | Tue 3/29/11 |
| 24 | | CDR Presentation | 0 days | Mon 4/4/11 | Mon 4/4/11 |
| 25 | | Testing and Integration Phase | 52 days | Sun 2/20/11 | Sat 4/30/11 |
| 26 | ✓ | Finialize Component Testing Schedules | 7 days | Sun 2/20/11 | Mon 2/28/11 |
| 27 | | Build | 45 days | Tue 3/1/11 | Sat 4/30/11 |
| 28 | ✓ | Component Testing | 23 days | Tue 3/1/11 | Thu 3/31/11 |
| 29 | | VT Academic Outreach | 1 day | Sat 3/26/11 | Sat 3/26/11 |
| 30 | | Finialize Systems Testing Schedule | 5 days | Sun 3/27/11 | Thu 3/31/11 |
| 31 | | Systems Testing and Integration | 22 days | Fri 4/1/11 | Sat 4/30/11 |
| 32 | | VT Academic Outreach | 1 day | Sat 4/9/11 | Sat 4/9/11 |
| 33 | | Operations Phase | 32 days | Sun 5/1/11 | Sun 6/12/11 |
| 34 | | VT Final Exams | 4 days | Fri 5/6/11 | Wed 5/11/11 |
| 35 | | OAR | 0 days | Sat 5/14/11 | Sat 5/14/11 |
| 36 | | Mission Planning | 22 days | Sun 5/15/11 | Mon 6/13/11 |
| 37 | | Travel to Texas | 2 days | Thu 6/9/11 | Fri 6/10/11 |
| 38 | | Competition | 67 days | Fri 6/10/11 | Mon 9/12/11 |
| 39 | | PFR | 0 days | Sun 6/12/11 | Sun 6/12/11 |
| 40 | | Travel to VT | 2 days | Sun 6/12/11 | Mon 6/13/11 |




Design Phase | Competition Task | Virginia Tech Task

# Conclusion



- **Major Accomplishments**
  - Experimental testing of separation of Lander and Carrier completed
  - Communication testing competed
  - Ordered and received hardware needed
  - Successfully tested flight software program for microprocessor
  - Validated the strength of the motor shaft to hold the carrier weight during flight
- **Current Development**
  - Focus on Mechanical Design efficiency of Descent Control
  - Currently testing hardware component systems
- **Unfinished Work**
  - Need to develop programming code for Accelerometer