Aeneas -- Colony I Meets Three-Axis Pointing

Michael R. Aherne, J. Tim Barrett, Lucy Hoag, Eric Teegarden, Rohan Ramadas
Space Engineering Research Center
4676 Admiralty Way, Marina del Rey, CA; 301-448-8509
maherne@isi.edu

ABSTRACT

A dedicated satellite mission is currently under development at the USC Space Engineering Research Center. Named "Aeneas," (after the Trojan warrior who personifies duty and courage) the cubesat will be used to track cargo containers worldwide. To accomplish this feat, the satellite must maintain a 2-degree-accuracy surface track – the first of its kind in cubesat technology.

This paper describes the requirements, design, implementation and tests to date in the areas of: flight dynamics, flight software, deployable spacecraft antenna, store-and-forward software, custom flight processor including MEMS gyroscopes, Doppler-based orbit determination enhancement and mobile ground station dish with helical feedhorn. Details are provided about the attitude control system and communications.

INTRODUCTION AND BACKGROUND

Continual advances in micro-electronics enable more to be done with less. Today's gadgets are smaller in size, lower in weight and consume less in power than their counterparts only a few years ago. This truism enables progress across many industries, and perhaps none so much as in cubesat technology. As the relaxation of constraints is allowing more performance to be packed into each cubic centimeter, nanosatellites are rapidly gaining the capability to address fundamentally important missions.[1]

Recognizing this trend, the USC Astronautics Department and the Information Sciences Institute (ISI) jointly created the Space Engineering Research Center (SERC) - a fast-paced learning environment pairing students with industry experts to push the envelope of nanosatellite technology. SERC's current satellite program is Aeneas, which modifies a 3U (30x10x10cm) National Reconnaissance Office (NRO) specified Colony I Cubesat bus to address a research thrust of the Department of Homeland Security (DHS). The delivery of Aeneas is scheduled for December of 2011 and the flight is manifested for June of 2012. It contains two payloads.

The primary payload speaks to a mission with global reach: tracking cargo containers over the open ocean with a 1-watt WiFi-like transceiver. A current tracking system for cargo containers, designed by our primary payload provider iControl Inc., is capable of identifying the container within a mile from shore, but loses all contact for the majority of the open-water journey. For both government and non-government entities, the ability to track containers in transit is highly valued. This mission uses a custom-built deployable mesh antenna, and stretches the attitude control and power generation capabilities of the Colony I bus to its limits.

The secondary payload is an experimental, next-generation, radiation-hardened flight processor. The result of many government-funded research initiatives, this ITAR-controlled processor is at risk of staying in the "unholy valley" between research and development. On Aeneas, the processor will be space-qualified by performing self-diagnostic checks and reporting the results back to the ground. We hope that by raising the technology readiness level (TRL) we can provide a path to service for this high-performance chip.

In this paper we will discuss the design work and fabrication supporting the primary payload: namely, three-axis pointing and the deployable antenna. We begin by describing the entire cubesat, focusing on those components that will serve a critical role in the success of the mission.

SPACECRAFT OVERVIEW

Aeneas is a modified 3U Colony I Cubesat. The baseline bus, provided by Pumpkin Inc., contains an 8051-based flight processor, attitude determination and control system (ADACS) unit and deployable electrical power system (DEPS) housed in a 3U bus. The ADACS, Maryland Aerospace's MAI-100, contains three miniature reaction wheels and three electromagnetic torque rods in a hermetically sealed enclosure, an external PNI MicroMag3 magnetometer, and coarse sun sensing ability via the body-mounted solar panels. The DEPS, manufactured by ClydeSpace, controls eight SpectroLab Ultra Triple Junction solar panels (three body-mounted, one end-mounted and four deployable) that charge three pairs of Lithium Ion Polymer batteries and provides power distribution.

To achieve the mission goals, significant additional functionality had to be integrated into the bus. Firstly, to provide sufficient power, the deployable panels were modified to deploy to a 90 degree angle with the solar cells facing outward. The Cubesat will be orientated in a sun pointing attitude profile with this “flower petal” panel configuration facing the sun.

Next, achieving the pointing requirements presented a significant design challenge. To achieve the 2-degree pointing accuracy, a Sinclair Interplanetary SS-411 two-axis digital sun sensor was selected to improve upon the baseline capabilities. To face the sun, it had to supplant the solar cells present on the sun pointing face. We also added three Analog Devices ADIS16260 single axis MEMS gyroscopes to propagate the sun vector during slews and eclipse. To reduce field input from the reaction wheel motors, the magnetometer was relocated from the ADACS interface to a far corner in the payload section. Finally, heritage GNC flight software had to be added to support the ADACS.

For communications, a Microhard MHX425 115kbps transceiver was added for telemetry and command, as well as an AstroDev Neon beacon. Two monopole whip antennas were added to support the two radios. Lastly, two Microchip PIC24-based flight processors were selected to replace the 8051-based flight processor. One flight processor board is a Pumpkin Inc. standard pluggable processor module (PPM) containing the PIC24FJ256GA110 microcontroller. The other has the same PIC24 chip and PPM form factor but was custom-designed by iControl and the Information Sciences Institute to include the three single axis gyros on board.

A summary of all modifications is shown in Figure 1.

Flight Software Details

The flight software for Aeneas is spread across the two flight processor boards. We refer to these pluggable processor modules (PPMs) as “Upper” and “Lower,” in reference to their position on the cubesat stack. Though several core functions are common to both, in general the Upper PPM contains all code relevant to Command and Data Handling (C&DH), while the Lower PPM contains all code relevant to

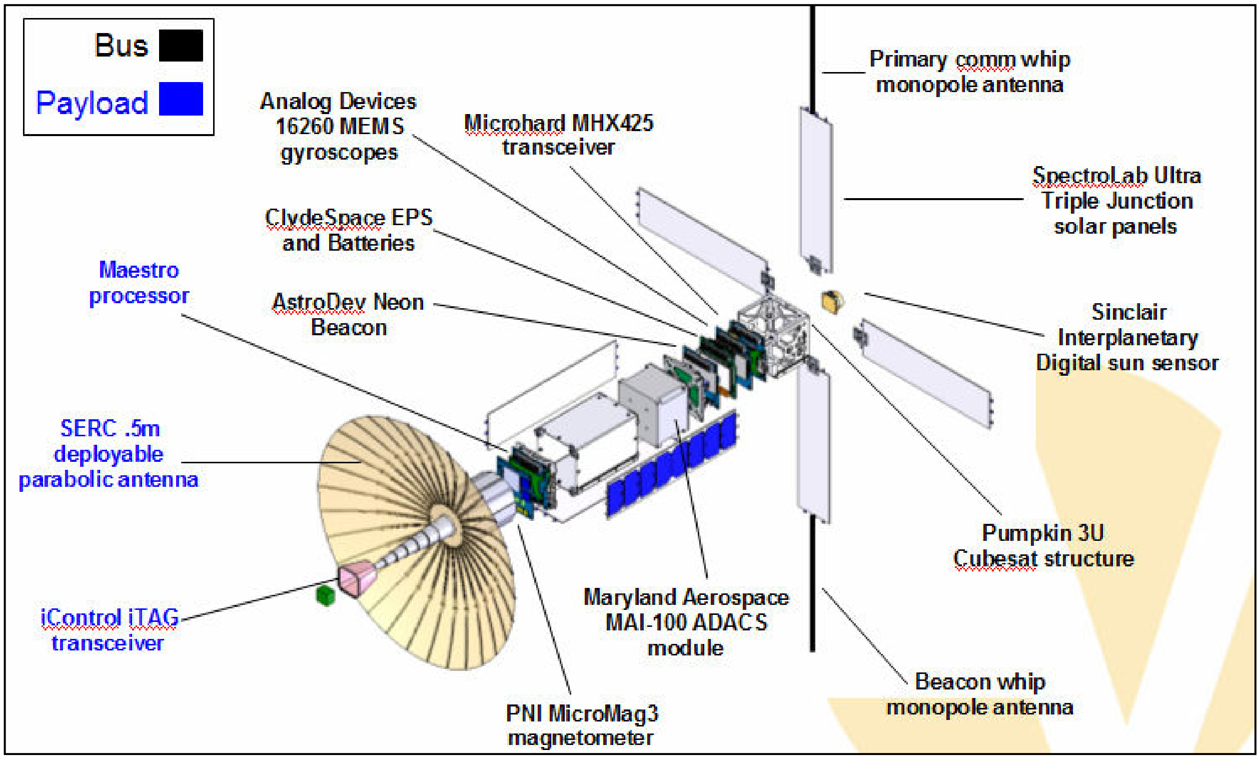

Bus ⬛
Payload 🟦
Analog Devices 16260 MEMS gyroscopes
Microhard MHX425 transceiver
Primary comm whip monopole antenna
SpectroLab Ultra Triple Junction solar panels
ClydeSpace EPS and Batteries
Maestro processor
AstroDev Neon Beacon
Sinclair Interplanetary Digital sun sensor
SERC .5m deployable parabolic antenna
Pumpkin 3U Cubesat structure
iControl iTAG transceiver
Maryland Aerospace MAI-100 ADACS module
Beacon whip monopole antenna
PNI MicroMag3 magnetometer

Figure 1: Aeneas Configuration

Attitude Determination & Control (AD&C). We will occasionally use the term Guidance, Navigation & Control (GNC) interchangeably with AD&C.

The processors are not redundant, but each has the capability to send commands and data to the other, and monitor the other's health and status. Both have been risk-reduced through previous missions – both the C&DH PPM and the custom PPM with gyros flew on USC's CAERUS Cubesat in December 2010. The AD&C software has its heritage in the Miniature Sensor Technology Integration (MSTI) Program of the 1990s.[4]

In addition to the two flight processors, Aeneas has 3 sensors and 2 actuators, as detailed above. The sensors are a set of gyroscopes, a sun sensor and a magnetometer. The actuators are reaction wheels and torque rods. A great deal of care must be taken when polling and commanding this suite, as the magnetic interactions between certain components can corrupt sensor data.

THREE-AXIS POINTING

The flight software on Aeneas has its heritage in the Miniature Sensor Technology Integration (MSTI) program. The purpose of that program was to "perform experiments to characterize a wide variety of Strategic Defence [sic] Initiative Office (SDIO) advanced sensor technologies in the Low Earth Orbit (LEO) space environment."[3] One of the experiments, from which our attitude control software is derived, was to track a missile booster. Intricate details of the control system are presented in McEwen. In this paper, we will summarize the control system and discuss changes we have made to the software.

Overview

To achieve surface track, Aeneas must know both its attitude and its orbital position with respect to the target on Earth. This requires a state vector of 11 elements: 4 for the attitude quaternion, 6 for orbital position and velocity, and 1 element describing the Earth's rotation. During a GNC cycle, the following series of events must occur: the state vector must be updated to reflect the most current satellite position, the attitude must be compared with a desired attitude to generate an error signal, the error signal must be fed through a control law to generate torque commands, and the commands must be acted upon with enough authority to maintain pointing accuracy throughout a pass. These are described below.

Updating the State Vector

The state vector is updated every 250 ms by the flight processor. The update is performed in three steps: processing sensor data, propagating internal models, and using both to compute the current attitude.

Sensor Processing

Aeneas samples from 4 components (gyros, magnetometer, sun sensor and reaction wheels) during every GNC cycle. All of the readings are adjusted to the body frame using appropriate rotation matrices, and the software reuses previous data for any sensor which cannot be read. Two of the sensors - the magnetometer and the sun sensor - are used without filters of any sort. By contrast, the gyros and reaction wheel tachometer readings are filtered. Each of these components is discussed.

Magnetometer

The PNI Micromag3 requires no filtering because the errors in the sensor are insignificant compared to the errors from the geomagnetic model to which it is being compared. However, the magnetometer must be polled very carefully, so as not to coincide with the MAI-100's torque rod activation. Within every 250 ms cycle of the MAI-100, the first 156 ms are free from magnetic interference from the torque rods. The obvious solution is to limit polling to these first 156 ms. One difficulty we found with this solution was that the MAI-100 and the PPM are on different internal clocks, which would drift in and out of phase with respect to each other. In other words, 250 ms according to the reaction wheels was different than 250 ms according to the processor.

To solve this problem, we tied an interrupt pin from the processor to the reaction wheels. At the start of every 250 ms cycle, the reaction wheels would pulse this pin, triggering an interrupt in the flight software and allowing the two units to stay in sync.

Sun Sensor

The Sinclair SS-411 sun sensor also requires no filtering, but instead needs some care in interpreting the response. The sun sensor has a host of sophisticated algorithms, but essentially works by taking a snapshot through its lens and processing the image until one pixel crosses a brightness threshold. The advertised bandwidth is 5 vectors per second, or 1 every 200 ms, but this varies greatly with the quality of the image. During testing we have seen solution times range from very fast (<10ms) to very slow (~2 seconds). This poses a challenge for a fixed GNC cycle, particularly because our software is very sensitive to false positives.

Fortunately, the sensor also returns values indicating the "goodness" of the internal solution. We reject any vectors that fail this goodness test, and also reject any vectors that take longer than a GNC cycle to return. We are currently investigating the maximum tumble rate we can sustain with these constraints.

Gyroscopes

The gyroscopes are our most difficult sensor, because the data must be handled very carefully. There are many forms of gyroscopic error, but they essentially come in two categories - *drift* in accuracy and *noise* about the reading. For our purposes we define the drift as the difference between a true zero point and the gyro's zero point. Noise we define as the distribution of samples around a specific point if one were to take multiple readings. Drift changes on the order of hours, noise from sample to sample.

The noise can be considered as a zero-mean Gaussian distribution about the true measurement, and is addressed with a low-pass filter. The readings are put through a discrete single gain/single pole adjustable filter of the form:

$$\frac{Az}{z-B} \tag{1}$$

where A and B have been experimentally chosen as 0.466 and 0.533, respectively.

This filter is essentially a weighted moving average, computing the current output Y[n] by using the following equation:

$$Y[n] = AX[n] + BY[n-1] \tag{2}$$

where Y[n-1] is the previous output, X[n] is the current input, and A and B are weighting factors.

This filter provides some "momentum" to the system, as new inputs are weighted slightly less than old outputs. This has the effect of reducing jitter at the cost of sacrificing some lag to fast-changing inputs (i.e. high accelerations). Since we do not expect to incur any high acceleration on this mission, the filter provides a computationally inexpensive way of smoothing our noise.

Compensating for drift is much more involved, because without a star tracker giving a precise attitude solution, it is difficult to recalibrate the gyros on orbit. We address this problem by relying on heavily processed sun sensor data. Derivatives of unfiltered sun sensor position readings are taken each cycle. These derived angular velocities are too inaccurate for use in short timescales. However, we have discovered that over very long timescales, on the order of 20 minutes (~5000 data points), the errors between these derivatives and the gyro readings can be averaged to produce an accurate estimate of the gyro offset in the pitch and yaw axes. Put more simply, we compare the angular velocity readings from the gyro with the angular velocity derivations from the sun sensor, and over time these comparisons converge to a constant. We use this constant to recalibrate the gyros on-orbit in pitch and yaw. Unfortunately, rolling about the sun vector does not produce a distinct-enough signal to perform this computation. We are currently looking at controlled slew methods to recalibrate the roll axis.

Reaction Wheels

The reaction wheels are not very noisy, but to improve the closed-loop response, the readings are put through a digital lag compensator of the form:

$$\frac{Az+B}{z-C} \tag{3}$$

The output Y[n] is computed by the following equation:

$$Y[n] = AX[n] + BX[n-1] + CY[n-1] \tag{4}$$

where X[n] and X[n-1] are the current and previous samples, and Y[n-1] is the previous output.

A, B and C are weighting factors which sum to 1. We are still adjusting these gains in testing to maximize the performance of the closed-loop system.

Once the sensors are processed, the next step in updating the state vector is to propagate the internal models.

Internal Models

Aeneas has 3 internal models to assist with AD&C: A geomagnetic model, which returns the predicted local magnetic field based on orbital position, Earth rotation, and the year; a Sun model, which returns a vector from the Earth to the Sun based on the date and time; and an orbit model, which propagates the orbital position and velocity, as well as the Earth's prime meridian rotation, in real-time. All models were developed under the MSTI Program, and the Aeneas team is grateful to iControl Inc. for allowing us to reuse this code.

The Geomagnetic Model

The Geomagnetic Model is an 8th-order International Geomagnetic Reference Field (IGRF). In order to fit the model within the PIC microcontroller's RAM and code space constraints, the Gauss and Schmidt coefficients are stored in non-volatile flash memory. Although this slows down the computation significantly (as the coefficients must be read into RAM during each use), the savings in code space is worth the trade.

It should be noted that during simulation on a desktop, a larger 13th-order model is used for increased accuracy. This model produces readings of the magnetic field that are within tenths of nanoTesla of the official Department of Defense (DOD) World Magnetic Model (WMM).

The Orbit Model

The orbit propagator is a fourth-order Runge-Kutta integrator. There is little documentation accompanying the code, however the propagator accounts for the J2, J3, J4, and J5 orbit perturbation terms. When compared with outputs from Satellite ToolKit, the orbital positions over the course of several days matched to within tens of kilometers.

The Sun Model

The sun propagator also contains little documentation. However, it is a simple algorithm that, based on the current Greenwich Mean Time (GMT), returns a vector to the sun in the Earth-Centered Inertial (ECI) coordinate frame.

Computing Attitude

Aeneas relies heavily on heritage attitude determination algorithms. Here we quote portions of the MSTI attitude controller description from McEwen at length before highlighting the changes we have made:

> **The attitude determination involves four frames of reference [...] The first is the Earth Centered Inertial (ECI) frame, which has its x-y vectors in the equatorial plane, and the x vector points at the Sun on the vernal equinox. The on-board orbit and Sun propagators output vectors coordinatized in these frames.**
>
> **The local vertical (LV) frame has the z axis pointing to the center of the Earth, and the x axis in the direction of flight.**
>
> **The Mission frame represents the desired vehicle orientation, and may move with respect to the local vertical frame. For example, during payload operation, the z axis of the Mission frame will track a point on the Earth's surface. [...]**
>
> **The fourth frame is fixed in the vehicle body. The objective of the attitude control is to minimize the angular error between the Mission frame and the body frame. [...]**
>
> **The attitude determination propagates gyro outputs in order to compute position when the Sun or the Earth or both are out of the field-of-view of their respective sensors.**
>
> **In typical control design procedure, one would linearize the plant, and then implement a linear state estimator for attitude determination. For Msti2, payload operation requires that the vehicle z axis track a point on the Earth's surface, which in turn requires that the vehicle have angular acceleration and deceleration over large angles. This makes linearization of Euler's equation infeasible. For this reason, the algebraic method, as described in Wertz, page 424, was chosen for attitude determination. This method provides an exact large angle quaternion over a wide range of vehicle orientations, and does not require a linearized model of the vehicle dynamics.[5]**

On Aeneas, several changes were made to this baseline software. The local-vertical frame was dropped by incorporating it into a more generic all-purpose Mission Frame. This frame can be used any number of ways: a programmed slew or series of slews, a nadir-pointing or velocity vector-pointing attitude, or a sun-tracking or surface tracking attitude. Any maneuvers that can be characterized by one or more 7-element states (4 quaternion elements and 3 angular velocities) can be used.

On MSTI, gyros were propagated only rarely (when sensors could see neither the Earth nor the Sun). We do not have an Earth lim sensor, so our gyros are propagated for much longer periods of time, whenever the Sun is out of view. This has caused us to revisit the gyro calibration, filtration and propagation algorithms in detail, as previously noted.

The attitude determination algorithm itself has not changed from the baseline.

Attitude Control

Again, we borrow heavily from MSTI:

> **The attitude control algorithm ... computes a requested torque T_r, based on position and rate errors. The basic scheme is proportional-derivative, with slightly different variations for despin, large angular error, and small angular error. During despin, requested torque is based on angular velocity:**

$$\vec{T}_r = -I^B 2K_r{}^{ECI}\vec{\omega}^B \qquad \textbf{(5)}$$

> **where I^B is the vehicle inertia matrix, K_r is the rate gain, and ${}^{ECI}\vec{\omega}^B$ is the angular velocity of the body (vehicle) with respect to the ECI frame. […]**
>
> **When the control is passed a large angle $q_{B/M}$, the requested torque is**
>
> $$\vec{T}_r = I^B(\omega_s \frac{\vec{q}}{\|\vec{q}\|} - {}^{ECI}\vec{\omega}^B)K_r + {}^{ECI}\vec{\omega}^B \times I^W\, {}^B\vec{\omega}^W \qquad \textbf{(6)}$$
>
> **where $\vec{q}$ is the vector part of $q_{B/M}$, I^W is the inertia matrix of the three reaction wheels, and ${}^B\vec{\omega}^W$ is the angular velocity of the wheels with respect to the body frame, ω_s is a (scalar) commanded slew rate. This causes a rate controlled slew about the Euler axis. The second term above accounts for stored wheel momentum.**
>
> **When $q_{B/M}$ represents a small angular error, the requested torque is**
>
> $$\vec{T}_r = -I^B(2K_p\vec{q}_{B/M} + K_r\, {}^M\vec{\omega}^B) + {}^{ECI}\vec{\omega}^B \times I^W\, {}^B\vec{\omega}^W \ \textbf{(6)}$$
>
> **This essentially is PD control. K_p and K_r are position and rate gains, and ${}^M\vec{\omega}^B$ is the angular velocity of the body (vehicle) with respect to the mission frame.[5]**

On Aeneas we have compacted the design, combining the despin and large-angle control modes into one. Both are now considered rate-controlled slews (i.e. the proportional error in the PD controller is overridden to be zero). The target rate of the "despin slew" is zero, whereas the rate of a normal slew is settable via commanded upload.

Other than gains, no changes were made to the small-angle controller.

Expected performance

To test the performance of the system, the flight software is run in a closed loop simulator. There are essentially two ways the simulation can be conducted, allowing us to emphasize rapid development or realism as necessary. Through our simulations, we've identified a couple areas of concern from a risk standpoint. We will describe how the simulations are conducted, the areas of concern we have, and the latest results.

In the first setup, the flight software is compiled into a Dynamic Link Library (DLL) and run with an Environment DLL in LabView.[4] This setup occurs entirely in a single laptop computer and involves no actual flight hardware. The setup leverages the resources of the simulation computer and runs much faster than real-time. We can simulate 12 hours of orbit time in just a few minutes. The outputs of the simulation are saved to Excel files, and custom-built macros can further format the output for use in STK. End-to-end, a change in the code can be viewed in a 3d STK model in about 20 minutes. This setup allows for rapid software development, and the ability to see a model react to changes gives our software developer an intuitive sense of the look and feel of nominal and off-nominal situations.

However, that setup makes use of resources that are not available on orbit. In order to test the real system, a second setup is used with the flight processor in the loop. Flight software is compiled to the PIC microprocessor, and runs in a loop with a laptop running LabView and the Environment DLL. Outputs from the LabView environment model are fed to the PIC, which makes attitude control decisions as if it were in orbit. This setup runs in real-time, due to the large amounts of data passed back and forth on the serial connection between the laptop and the PIC. Although this method is much slower, it gives us confidence that the memory and speed limitations of the microcontroller do not affect the outputs of attitude determination and control.

Using both of these setups throughout development, we have run several hundred simulations of Aeneas on orbit. As the simulations became more refined, they incorporated better error models and more accurate mass properties. From these improvements we've discovered some driving concerns for our mission. We discuss below the biggest of these - aerodynamic torque.

Aerodynamic Torque

The Colony I cubesats are designed to fly in what we refer to as "dart mode" - the long axis of the cubesat is aligned with the orbital velocity vector, as if the satellite were a dart. However, our original concept of operations had the satellite flying perpendicular to this, so that one end of the "dart" was nadir-pointing. This is where we would place our antenna, and we would sweep across the surface of the Earth looking for a connection.

This concept of operations fails miserably for a nanosatellite whose only method of momentum dumping is (relatively weak) torque rods. In addition to power problems, aerodynamic pressure on the antenna causes external torques throughout the orbit that rapidly overwhelm the control system.

Seeing this, we changed our concept of operations to be sun-pointing with the solar cells rather than nadir-pointing with the antenna. Aerodynamic torques still pose a difficulty, but our new conops distribute these torques evenly about the satellite, canceling each other out. We are currently re-running simulations with more accurate models, and initial results indicate we are within our control authority.

CLOSING THE LINK

Antenna Design

The primary mission of Aeneas is a proof-of-concept communications link requiring 3-axis pointing and surface tracking with a nanosatellite. The satellite is intended to receive communications from a 1W transmitter on the ground and then downlink with another station on the ground. To achieve these goals, the satellite will have a 0.5 m deployable antenna. The antenna will consist of a mesh constrained by 30 ribs with 2 joints each, as well as a deployable central mast housing the transceiver. The rib and mesh design will allow the packaged antenna to fit within the small form factor required to fit inside the payload section of the Colony I 3U cubesat, about 10x10x16 cm. When stowed, a central hub connected to the ribs and the mast will be lowered inside a canister within the payload section; upon deployment, a spring will force this hub out of the canister which will release the ribs.

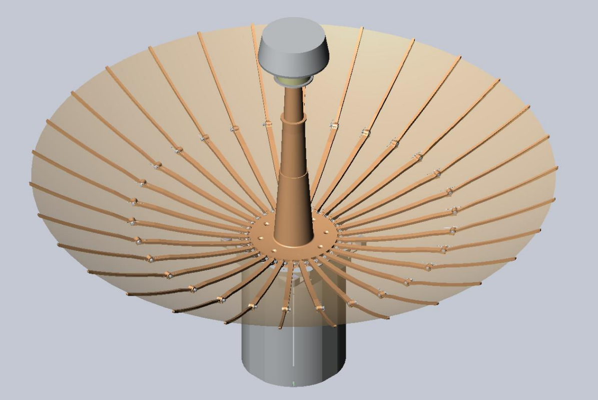


Figure 2: Dish Deployed

Antenna Construction

The construction of the antenna consists of attaching the ribs and mast to the central hub, and attaching the mesh to the hub/rib assembly. The ribs are attached via rivets and are held in the open position with torsional springs. The central mast with the sub-reflector and transceiver is constrained by the tapered shape of the mast and is held in place with a linear spring.

Once the construction of the structure of the dish is complete, the mesh surface is laid down on a mandrel with the appropriate curvature. The mesh is then cut to slightly larger than the final size and weights are attached at equal distances around the perimeter of the mesh. The mandrel and mesh are then subject to vibration. This vibration is intended to relax the mesh to help it to maintain a parabolic shape when attached to the ribs.

After the vibration cycle, the dish structure is placed on top of the mesh on the mandrel, which has a hole in the center to allow for the antenna mast to protrude. The mandrel has slots at 12° increments, which help to ensure equal rib spacing and provide access to the front of the dish at the points where the mesh is to be connected to the ribs. This connection is achieved by running wire through the mesh and through holes in the ribs. The ends of the wire are then twisted together, lowered into slots in the ribs, and sealed with epoxy. This will prevent them from becoming untwisted and will also reduce the possibility of snagging.

After the mesh has been affixed to the ribs, the outer perimeter of the dish is sealed with strips of Kapton-backed film adhesive. This will close the end of the mesh, which will help prevent snagging. It will also spread the load between the ribs due to mesh tension, which will reduce distortion of the dish shape.

Antenna Stowage

Due to the relatively complex nature and small size of the dish, stowage presents an interesting problem. Because the ribs are bound together by a relatively tightly stretched mesh when deployed, the ribs must essentially be folded inward simultaneously. Failure to do so may result in cross-axis torques on the joints of the ribs. Because of the small size of the antenna hardware, these torques may be sufficient to cause damage to the rib joints, which could result in breakage or misalignment. Additionally, the stowage scheme needs to happen in a slow and controlled manner, as the mesh must be carefully managed to prevent any sort of entanglement upon deployment. This problem is further compounded by the design of the dish. Each rib consists of an inner and outer portion, connected by a spring-loaded hinge joint. This rib assembly is then attached to the inner hub by a similar spring-loaded hinge joint. Due to the positioning of the rib sections in the stowed configuration (Figure 3), the outer portion of the rib must be folded before the inner portion of the rib. However, as both sections are jointed, an upward force on the outer portion of the rib will result in actuation of the rib joint connecting the inner rib to the hub rather than the joint connecting the inner and outer ribs due to a larger lever arm. Furthermore, the dish cannot be readily accessed from the front due to the mesh.

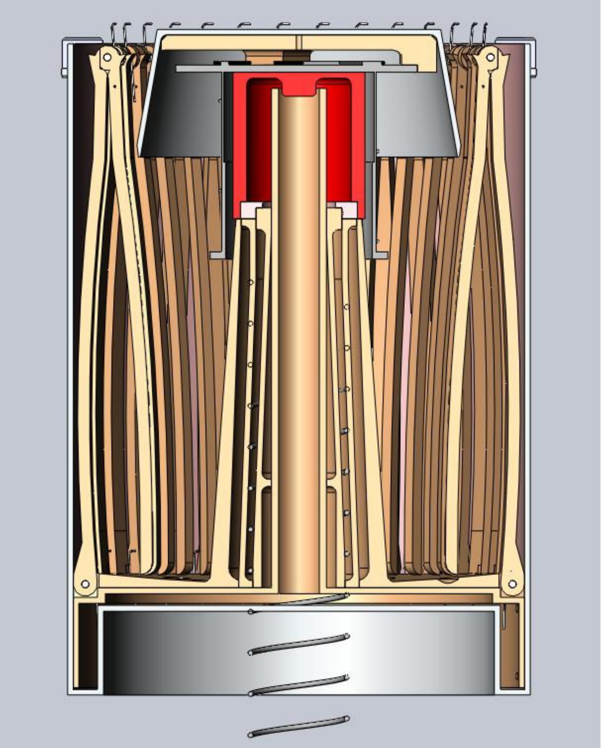


Figure 3: Dish Stowed

To solve these problems, a solution using rapid-prototyped extruded ABS components is implemented. The rib joints have a small amount of separation from the mesh to prevent pinching and to allow the ribs to close further in the stowed position. The stowage scheme uses a 3-D printed ratchet device that can attach to the rib joint using that gap, shown in Figure 4.

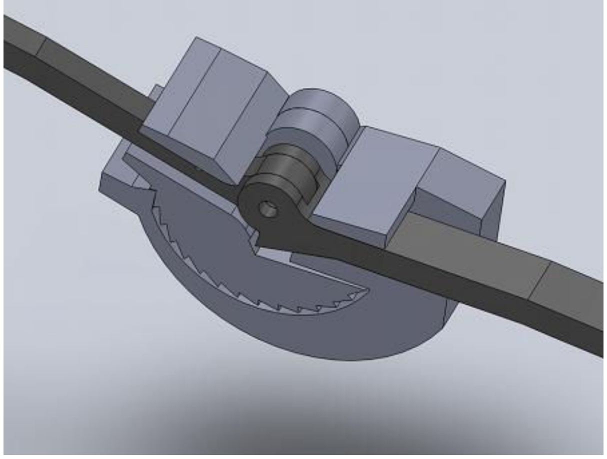


Figure 4: Ratchet

The rapid-prototyped nature of these parts allows for them to be form-fitting to the ribs, and cheaply manufactured in the relatively small numbers needed for this operation. The form-fitting nature will maintain the position of the ratchet with respect to the rib joint, which is crucial because the hinge cannot be actuated unless the hinge joint and the ratchet joint are in line. This ratchet solves most of the major possible problems with antenna stowage. Because it fits entirely underneath the mesh, the ratchets do not require any access from the front side of the dish. Additionally, the ratchets are connected to the joint itself, so they do not suffer from the lever-arm problem described above. The individual ratchets operate in ~10° increments - this is sufficiently small to prevent significant cross-axis torques in adjacent ribs. The ribs can be actuated sequentially, allowing for pseudo-simultaneous stowage. Because each rib joint can be actuated individually and will stay in place without additional outside interference, this also allows for the stowage process to proceed in a slow and controlled manner which is ideal for managing the mesh to prevent entanglement. By proceeding in a circular fashion around the dish, the outer ribs can be brought to a nearly-stowed position using these ratchets. However, because of the nature of the connection to the rib joint, they prevent full stowage of the outer ribs. To achieve full outer rib stowage, the ratchets are removed and replaced with single-piece form-fitting clips on the end, as seen in Figure 5. These are rapid prototyped in order to create the form-fitting profile. Once the outer ribs are locked into the stowed position with the clips, the inner ribs can be easily folded inward and the entire rib/hub/mast assembly can be lowered down into the canister, at which point the clips can be removed.

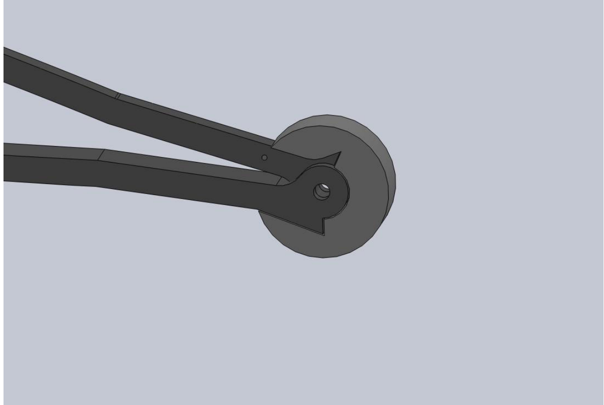


Figure 5: Endclip

Antenna Deployment

Deployment of the antenna is achieved using springs and a nylon burn wire. There are 62 total springs in the system - 60 springs on the rib joints (1 spring per joint, 2 joints per rib, and 30 ribs), 1 spring to deploy the mast, and 1 spring to deploy the entire hub assembly out of the canister. The entire system is initially held with the springs in tension - the ribs are constrained from deploying by the canister walls and the mast and hub are held in the stowed position with nylon monofilament in tension. This monofilament has a section which is wrapped in a nichrome wire which will, when provided current, cause the monofilament to melt. This will allow the spring to force the hub out of the canister. Once the antenna ribs clear the canister walls, the springs in the joint will force the dish to open. The springs are set up to provide torque in the deployed position - the final, correct shape is determined by hard stops in the hardware.

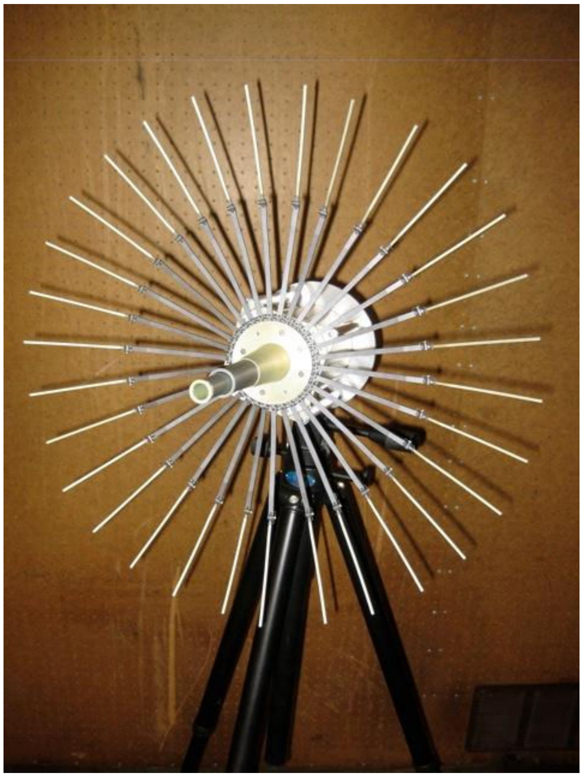


Figure 6: Antenna Qualification Unit

LINK DESIGN

There are 3 communication channels on Aeneas. Two of these are on the bus, allowing command and telemetry to the flight software. The third is the previously described parabolic antenna, used by the primary payload for its tracking mission. The design of these three links follows.

Primary Payload

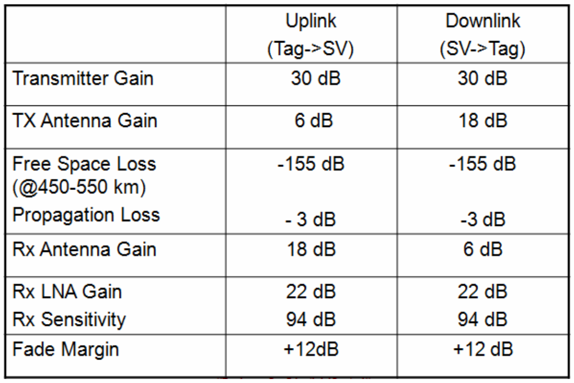
The primary payload – the parabolic deployable antenna and feedhorn-mounted transceiver – will communicate with both Earth-based transceivers called iTags and small array antennas tied to central receivers called iGates. Both the iTags and iGates are tied to the internet to provide remote monitoring. The transceiver as well as the entire tracking infrastructure is produced by iControl Incorporated and is in deployment in ports today.

Parabolic reflectors provide a significant advantage for long range communications in that they allow for higher gains at a given frequency relative to other antenna geometries suitable for nanosatellite application, including horn and lens antennas. Resultantly, parabolic reflectors operating at high gain will necessarily produce reduced beamwidths; the reflector diameter must exceed the transmission wavelength by a large margin for these high gains to be realized. The transceiver aboard the satellite and located at Earth stations communicate via a 2.4 GHz link. By Planck's light equation, this provides a wavelength of 0.09 m, about 50 times smaller than the actual reflector diameter as noted above. Thus, the reflector size requirements are easily satisfied.

The transceivers themselves feature a dual receive (RX)/transmit (TX) design. This eliminates noise produced by RX/TX switching within the antenna circuit. RX/TX antenna switch loss at the feed point is estimated at 4.5 dB. The dual integration design is a product of iControl Incorporated. Each dual integration transceiver PCB includes 2 Mb of memory and has store and forward capability. Communications are performed under 128 AES encryption, and the DSSS 2.4 GHz band is internationally legal. In terms of specific RX/TX functionality, the communication uplink performs tracking and alarm control while the downlink performs tag commissioning and command.

The downlink transmission – defined as the space vehicle to ground link – features 18 dB antenna gain, a larger figure than the 6 dB antenna gain associated with the uplink transmission. Additionally, the link includes a Noise Amplifier (LNA) device located aboard the nanosatellite payload. This multistage LNA includes filters to minimize out-of-band noise and provides a gain of 22 dB. Table 1 is an overview of the system link budget of the parabolic antenna.

Table 1: An overview of the AENEAS primary payload communications link budget



	Uplink (Tag->SV)	Downlink (SV->Tag)
Transmitter Gain	30 dB	30 dB
TX Antenna Gain	6 dB	18 dB
Free Space Loss (@450-550 km) Propagation Loss	-155 dB - 3 dB	-155 dB -3 dB
Rx Antenna Gain	18 dB	6 dB
Rx LNA Gain Rx Sensitivity	22 dB 94 dB	22 dB 94 dB
Fade Margin	+12dB	+12 dB

The free space loss calculations depicted above correspond to the ideal satellite range conditions. The 450-550 km altitude range translates to a directly overhead pass for the satellite at perigee. The full slant range extends to well over 2000 km for a horizon pass at apogee.

Telemetry and Command

The telemetry and command link between a satellite monopole whip antenna and Earth is closed using a ten-foot, azimuth-elevation controlled dish antenna with a secondary reflector and helical feed, mounted on a trailer and located as needed for good mission visibility. The fixed dish was retrofitted with sturdy articulation hardware and the helical feed was hand made. Together they have a measured gain of 19dBi at the frequency of the MHX425 – nominally 437 MHz. With primary characteristics that include a transmission power of 1 Watt and a horizon-limited elevation angle of 15 degrees, the ground station will be used to complete the primary communications link with the satellite. This power level provides a closed transmission link up to a satellite slant range of 1400 km.

A secondary communications channel is comprised of the AstroDev NE-1 9600 baud GMSK beacon. It uses a second whip antenna and broadcasts rotating health, status and mission information data at regular intervals. The 1 watt unidirectional link is closed with a large azimuth-elevation controlled Yagi antenna and an amateur radio receiver on the roof of the physics building on the University of Southern California campus. Amateur radio enthusiasts around the world will be employed as with previous missions to return beacon data throughout its orbit.

Accuracy Improvements

Although the satellite bus antennas are omnidirectional, both the ground dish and the Yagi antenna used to track them are not. In order to maximize the link time, accurate tracking information is required.

The initial two-line element sets (TLEs) given by the launch provider yield accurate on-orbit tracking information in the early phases after launch. However, as the cubesat drifts away from this measured TLE (due to drag conditions), it becomes more and more difficult not only to track the cubesat but to tell it apart from other cubesats sharing the same launch. Using the measured doppler shift from our beacon and a current TLE, we are able to generate an updated TLE' that can help us keep track of our cubesat as orbit conditions change. The work is the subject of ongoing research and shows promise. The limiting factor seems to be rejecting noise on experimentally captured doppler shift data. Details of our method are provided in Hsu et al.[2]

CONCLUSION

The primary mission for Aeneas poses a great challenge, not only on its technical merits, but on the ability of the Space Engineering Research Center to effectively coordinate and utilize students in pursuit of difficult goals. By focusing on hands-on experience, SERC is exposing students to the tools and methods necessary to build the next generation of hi-tech satellites. In addition, the miniaturization of capability allows these students to address national concerns with university budgets. If successful, Aeneas will be the first cubesat to perform three-axis surface tracking, and will put the capability to use by demonstrating a revolutionary new communications system and space-qualifying a next-generation processor. All on a spacecraft the size of a loaf of bread.

In this paper we described the attitude determination and control software design and simulation. This subsystem is on track: on-orbit nulling of gyro drifts is being addressed, drag and MOI with respect to control authority are being driven to known quantities via refined mass models and planned experiments, and our previous launch experience has provided great risk reduction and grown our knowledge base and infrastructure.

The deployable parabolic reflector is generating interest as well as forcing careful design modifications as we head into qualification testing (vibration and thermal vacuum). Initial RF testing showed near-theoretical gains for a fixed version, boosting confidence in our fabrication procedures.

The team looks forward to completing the testing of the primary payload link, delivering an operational spacecraft by years end and performing a successful flight in June of 2012. By pressing the Colony I platform to its limits, we are leveraging small investments to execute big missions.

ACKNOWLEDGEMENTS

We thank Joe Kunc, Director of SERC, Dan Erwin, Astronautics Department Head, and Mike Gruntman, Founder of the USC Astronautics Program, for providing the best students that money can’t buy.

We thank the tireless efforts of our current and former staff: Will Bezouska, Jeff Sachs, John Smolik.

We recognize the many contributions of our student team: Omair Rahman, John Razzano, Emin Vartanians, Melissa Jawaharlal, Sara Gramling, Rahul Karkhanis, Chunyih Chu, Dayung Koh, Neha Rathore, Carson Vogt, Erin Fowler, Amparo Romero, Desiree Webster, Xiaodan Wu, Maria Guzman, Amit Shah, Pezhman Zarifian, Mike Zarem, Michael Martin, Desiree Webster, and Joe Parsons.

We thank iControl Inc. (Fred Tubb, Diane Quick and Brian Tubb) for the use of MSTI software, for implementing the gyro-equipped PPMs and of course for the primary payload and asset tracking infrastructure.

We thank Brian Davis of Space/Ground System Solutions, Inc. (SGS) and Stephen Arnold of the Naval Research Laboratory (NRL) for their technical advice.

We are thankful to Dave Williamson, Dutch Schultz and others that attended our CDR & TRR.

We thank the Information Sciences Institute staff for providing infrastructure and operational space.

We thank Talbot Jaeger of NovaWorks who gave us the opportunity to combat test our prototypes.

Finally, we give credit and a great deal of thanks to David Barnhart, former Associate Director of SERC, who conceived the mission and whose unbridled optimism enabled it to happen.

REFERENCES

1. Puig-Suari, J., Turner, C. and Ahlgren, W. “Development of the Standard CubeSat Deployer and a CubeSat Class PicoSatellite”, Proceedings of the 2001 IEEE Aerospace Conference, Big Sky, MT, 2001.
2. C. Hsu, D. Koh, O. Rahman, T. Barrett. “Low-Cost TLE Updating Method for Nanosatellites using Doppler Signature” AIAA Space 2011 accepted paper/poster
3. D. Barnhart, J. Feig, and E. Grigsby. “Miniature sensor technology integration (MSTI): small space platform program.” Proc. SPIE 1940, 174 (1993)
4. F. Tubb, R. McEwen, J. Farazian, A. Waddell. "MSTI3 Attitude Control Software Development Using Automatic Code Generation," AIAA Small Satellite Conference, 29 Aug 1994, Utah State University, Logan UT.
5. R. McEwen. "Overview of the Miniature Sensor Technology Integration Attitude Control System," AAS Conference on Guidance and Control, 2 Feb 1993, Keystone CO.