



[Type text]



Hydrogen-Alpha Exploration with Light Intensity Observation Systems (HELIOS) II **Final Science Report**

December 13, 2013

Caleb Lipscomb¹, Jonathan Sobol², Anthony Lima³, Kristen Hanslik⁴, Jorge Cervantes⁵, Ashley Zimmer⁶, Austin Bennett⁷, Glenda Alvarenga⁸, Greg McQuie⁹

*University of Colorado Boulder, Boulder, CO, 80309
Colorado Space Grant Consortium*



**Project HELIOS II
flew on the HASP 2013 mission and captured high-resolution images of the Sun in the Hydrogen Alpha
wavelength. HELIOS II was designed and built by students at the
University of Colorado at Boulder.**

¹ Project Manager

² Systems Engineer

³ Optical system lead,

⁴ Command and Data Handling system lead

⁵ Electronic Power System lead

⁶ Structure & Thermal lead

⁷ Structural Engineer

⁸ Student Advisor, Member of HELIOS I

⁹ Student Advisor, Member of HELIOS I



[Type text]



Nomenclature

- SWIS* = Solar Wavelength Imaging System, or Optic System
- ADCS* = Attitude Determination and Control System
- CDH* = Command and Data Handling System
- EPS* = Electronic Power System

Table of Contents

Section	Page Number
I. Mission Overview	3
II. Mission Background and Theory	3
III. Requirements	4
IV. HELIOS II Design	6
V. Mission Results and Failure Analysis	23
VI. Lessons Learned	33
VII. Demographics	36
VIII. Conclusion	36
XV. Acknowledgements	37
X. References	37



I. Mission Overview

The mission of Hydrogen-Alpha Exploration with Light Intensity Observation System (HELIOS) II was to capture high-resolution images of the Sun in the Hydrogen Alpha light wavelength. HELIOS II was designed as a payload on the High Altitude Student Platform (HASP). Team HELIOS II designed and constructed a Solar Wavelength Imaging System (SWIS) to view the Sun in the Hydrogen Alpha wavelength and capture high resolution images in that wavelength using an Attitude Determination and Control System (ADCS) to locate the Sun and orient the SWIS towards the Sun on-board a HASP flight.

Mission Objectives:

1. Observe and capture images of the Sun in Hydrogen Alpha wavelength using the SWIS system.
2. Design and implement an ADCS system to locate the Sun in the sky and orient SWIS towards the Sun.
3. Prove the viability of high altitude balloon solar observation during a Colorado Space Grant Consortium (COSGC) sponsored HASP flight.

II. Mission Premise

A. High Altitude Balloon Observation

Currently, the majority of solar observations are performed using ground or orbit based telescopes. These two methods of observing the Sun have several drawbacks. Ground observations face issues with interference from atmospheric filtering; effectively lowering the quality of solar images and reducing the ability to gather accurate scientific data from those images. Orbiting observatories have a high cost, limiting the quantity of such solar imaging missions in space. The lower quantity of orbital missions causes reduced access to the solar images taken by orbital observatories. With these restrictions in mind, an alternative method to image the Sun is through the use of high altitude balloon observatories. High altitude balloons are a relatively inexpensive platform. The HASP platform shall travel above 99.5% of earth's atmosphere, mitigating the effect of atmospheric interference during solar observations.

The Colorado Space Grant Consortium (COSGC) at the University of Colorado at Boulder (CU) has a history of high altitude observatory experiments. DIEHARD (2008) determined the viability of high altitude observatories by collecting diurnal and nocturnal images of celestial bodies to determine atmospheric turbulence and light intensity due to residual particles in the atmosphere. This was done using photometers mounted 45-degrees from the horizon. BOWSER (2009) further determined the practicality of high altitude observatories by examining certain wavelengths of cosmic light and took corresponding diurnal images and light intensity readings of the sky. BOWSER also measured platform stability in order to determine the conditions in which future HASP missions will fly. SPARTAN-V (2010) worked towards the goal of supporting precise photometry from balloon based pointing systems and telescopes. SPARTAN-V focused on characterizing atmospheric scintillation and extinction to support the practicality of observing exo-planets from a high altitude balloon.

In 2012, the University of Colorado Boulder HASP team, HELIOS, flew a similar mission to test the viability of solar observation on a high altitude balloon platform. However, their mission was hindered by several issues. HELIOS II improved on the HELIOS I design by:

1. Designing, building, and flying a functioning ADCS system to actively track the Sun
2. Fly an ADCS camera to characterize the accuracy of the ADCS system
3. Improving the magnification and resolution of the science camera
4. Designing and using a more robust Electronic Power System (EPS).

B. Photometry

The Solar Wavelength Imaging System (SWIS) captured images in the Hydrogen-Alpha (656.3 nanometer) wavelength. What is considered "visible light" can be separated into two categories. One: light that is seen with the naked eye and appears in images as white light. Two: the filtered spectrum of this white light, which can narrow



down specific wavelengths. This filter system is used to observe details of the Sun that would be obscured by the white light from the Sun’s photosphere.¹⁰

H-Alpha filters allow the camera to detect only a small bandwidth of visible light around the wavelength of 656.3 nanometers. This wavelength is absorbed and re-emitted by the Hydrogen in the Sun’s atmosphere. It is one of the most useful wavelengths in which to observe the Sun because it eliminates most light from the photosphere, allowing high visibility of solar features. This interaction between emitted light and Hydrogen in the Sun’s atmosphere predominantly highlights surface features.¹¹ Major solar features in this wavelength are solar prominences, sunspots, and coronal mass ejections. Therefore, H-Alpha imaging of the Sun provides incredible amounts of comprehensive data to be taken on solar activity.¹²

The atmosphere of the Earth has shifting air pockets, which distort the view of ground telescopes despite scientific advancements in telescope design. The Earth’s atmosphere has a considerable amount of hydrogen, despite its small percentage of the total composition. This Hydrogen also interferes with ground-based solar observation. New ground telescope technology has been able to correct for the atmospheric distortion to some extent but there is not a way of seeing the wavelengths blocked by the atmosphere. Solar observation will achieve the best clarity only if it is done above the atmosphere. HELIOS II ascended above 99.5% of Earth’s atmosphere, which allowed SWIS to observe in these desired wavelengths without interference from the Earth’s atmosphere.¹³

III. Requirements

In order to complete all mission goals, Team HELIOS II followed all requirements specified by HASP, in addition to all requirements derived from mission objectives. The HELIOS II Requirements were as follows:

Level	Requirement	Derived
0.1	Observe and capture images of the Sun in Hydrogen Alpha Wavelengths	Objective
0.2	Design and implement an ADCS (Attitude Determination and Control System) to locate the Sun in the sky and orient SWIS (Solar Wavelength Imaging System) toward the Sun	Objective
0.3	Prove the viability of high altitude balloon solar observation within a COSGC (Colorado Space Grant Consortium) sponsored HASP flight.	Objective

Level	Requirement	Derived
0.1.1	SWIS shall implement an infrared filter allowing imaging of 656.28 nm wavelengths	0.1
0.1.2	SWIS shall implement a camera capable of gathering high resolution images	0.1
0.1.3	SWIS shall use enough magnification to be able to resolve sunspots on the Sun’s surface.	0.1
0.1.4	One barrel of SWIS shall have a large field of view with low resolution	0.1
0.1.5	One barrel of SWIS shall have a small field of view with high resolution	0.1
0.1.6	SWIS shall be insulated and isolated from all other systems’ thermal footprint	0.1
0.1.7	SWIS data storage system shall be capable of connecting to primary control computer	0.1

¹⁰ "Observing the Sun in H-Alpha." *The Prairie Astronomy Club*. The Prairie Astronomy Club, n.d. Web. 11 Dec. 2013. <<http://www.prairieastronomyclub.org/resources/solar-observing/observing-the-sun-in-h-alpha/>>.

¹¹ *Ibid*. 2013.

¹² *Cool cosmos*. (n.d.). Retrieved from http://coolcosmos.ipac.caltech.edu/cosmic_classroom/multiwavelength_astronomy/multiwavelength_museum/sun.html

¹³ Eisenhamer, J. (2012, March 02). *Hubblesite*. Retrieved from http://hubblesite.org/the_telescope/hubble_essentials/



[Type text]



Level	Requirement	Derived
0.2.1	ADCS shall monitor the Sun's position along x and y axis	0.2
0.2.2	ADCS shall use motors to orient SWIS in the direction of the Sun	0.2
0.2.3	SWIS shall be capable of orienting the cameras towards the Sun throughout the entire mission	0.2
0.2.4	ADCS shall be designed with consideration to thermal effects on materials	0.2

Level	Requirement	Derived
0.3.1	HELIOS II shall comply with all HASP requirements outlined in RFP	0.3
0.3.2	HELIOS II shall comply with all budget and schedule constraints dictated by COSGC and HASP	0.3
0.3.3	HELIOS II shall maintain temperatures required for proper operation of all systems	0.3

Level	Requirement	Derived
0.2.1.1	ADCS shall use photodiodes to measure the intensity of light hitting the photodiode arrays	0.2.1
0.2.1.2	ADCS shall use a photodiode array shaped such that it allows the light intensity readings from the photodiodes to locate the Sun.	0.2.1
0.2.1.3	ADCS shall use a photodiode array to locate the Sun on the x and y axis	0.2.1
0.2.2.1	The ADCS shall use motors that are able to point the arrays to within 0.25 degrees of a specified position	0.2.2
0.2.2.2	ADCS shall use motors with enough torque to move the SWIS housing	0.2.2
0.2.3.1	HELIOS II Camera housing shall be tall enough to allow the cameras to be raised to 70° above the azimuth on the y-axis	0.2.3
0.2.3.2	ADCS shall be able to rotate the SWIS camera housing 360 degrees on the x-axis	0.2.3

Level	Requirement	Derived
0.3.1.1	Payload volume shall not exceed 38x30x30 cm	0.3.1
0.3.1.2	Payload shall resist the effects of up to 10 g vertical force and 5 g horizontal force	0.3.1
0.3.1.3	Payload shall utilize a twenty-pin EDAC 516 interface to HELIOS II system power and analog downlink channels	0.3.1
0.3.1.4	Payload shall not draw more than +30 VDC or 2.5 amps and shall split the provided +30 VDC to voltages necessary to operate payload	0.3.1
0.3.1.5	Payload shall enable six discreet command functions from HASP using EDAC 516-020 interface.	0.3.1
0.3.1.6	Payload shall allow serial downlink functioning at 4800 baud	0.3.1
0.3.1.7	Serial up-link shall allow for 2 bytes per command	0.3.1
0.3.1.8	Payload shall use a DB9 connector, RS232 protocol, with pins 2, 3 and 5	0.3.1
0.3.1.9	Payload shall transmit data packages showing health statuses utilizing unique header identification	0.3.1
0.3.1.10	Payload shall be incorporated into existing HASP platform mounting design	0.3.1

Level	Requirement	Derived
0.3.2.1	All receipts and proofs of purchase shall be retained	0.3.2
0.3.2.2	Schedule shall include weekly deadlines for each phase of design, assembly, and integration process	0.3.2
0.3.2.3	Schedule shall include all design document revisions; including relevant presentations	0.3.2
0.3.2.4	Schedule shall include all weekly team meeting dates and times	0.3.2



Level	Requirement	Derived
0.3.3.1	SWIS structure shall be insulated to minimize thermal footprint of other systems	0.3.3
0.3.3.2	All internal components of all systems shall remain within operating temperatures within the range of -80 to 60 °C	0.3.3
0.3.3.3	All systems shall utilize heat sinks to mitigate thermal footprint effects on SWIS	0.3.3

IV. HELIOS II Design

A. Solar Wavelength Imaging System (SWIS)

The final design of the Solar Wavelength Imaging System, SWIS, used two cameras (Science Camera and ADCS Camera). Each camera had a different mission: the ADCS Camera was designed in order to observe the performance of the active tracking system, and the Science Camera was designed to see features on the surface of the Sun.

The Science Camera (see Figure 4.1.0) used a large magnification to observe the surface of the Sun in high resolution. This resulted in a small field of view. The design of the Science Camera changed to use just one lens instead of two. The first design of the science camera magnification system had an error in the calculation of the magnification of the two-lens system.

The simplest way to resolve this error was to remove the second lens and only use the objective lens. This configuration produced the correct magnification. The final design used one lens with a focal length of 250 mm and had a 25 mm aperture. Unfortunately, this made the barrel excessively long and exceeded the vertical space allowance. The objective lens was located at the front of the C-mount tube while the filters were located at the rear. The filters were placed in the following order: one dichroic Longpass filter (filters UV light), one hot mirror (filters IR light), one narrow bandpass filter with a 10 nanometer bandwidth (passes light at 656 nanometers), and one neutral density filter with an optical density of three. See Figure 4.1.1. The filters and lens were placed in a tube of black anodized aluminum with standard C-mount threads. Because standard parts were used, No parts needed to be machined. Additionally, the anodized aluminum minimized outgassing. The actual camera used also changed. A camera using a CMOS chip was used instead of the CCD chip in the preliminary design. The new chip had more megapixels and considerably increased resolution. The CMOS chip had a resolution of 2592 by 1944 pixels, larger than the resolution of 1600 by 1200 pixels of the CCD chip. Additionally, the CMOS camera was much smaller than the original camera.



Figure 4.1.0 Science Camera

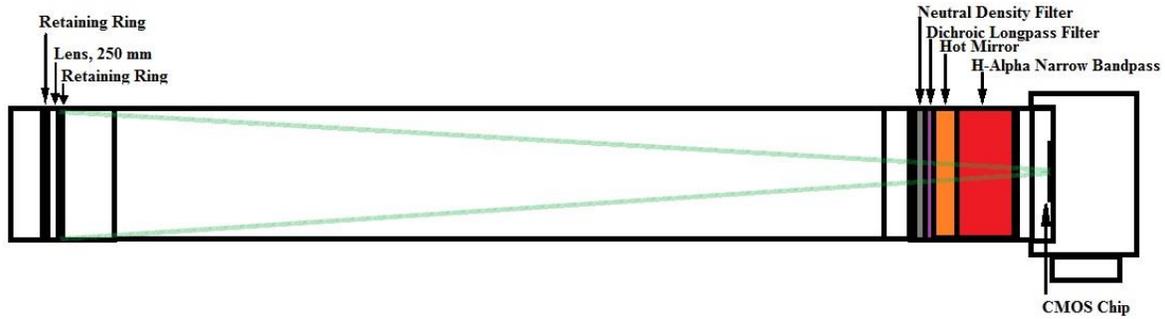


Figure 4.1.1 Science Camera

The ADCS Camera (see Figure 4.1.2) used the CCD chip as specified in the initial design. The objective lens was located at the rear of the C-mount tube while the filters were located at the front. This order is reversed in reference to the Science Camera because of the difference in focal lengths. The focal length used by the ADCS camera was too small to avoid light shifts. If the filters were placed after the lens, their filtration wavelengths would have been shifted. Also, there was insufficient space behind the lens to fit the filters. The filters were placed in the following order: one dichroic Longpass filter (filters UV light), one hot mirror (filters IR light), and two neutral density filter with an optical density of three. The tube which held all of the optical components was also black anodized aluminum with standard C-mount threads.



Figure 4.1.2 ADCS Camera

Both cameras were focused by placing the lens one focal length away from the chip. This was verified by recording pictures of the Sun and making minor variations in the distance. The pictures were then used to calculate field of view and resolution metrics in order to ensure the accuracy of the metrics. The Sun's diameter is approximately 1.39×10^6 kilometers and is on average 1.50×10^8 km from the Earth. At this distance, the Sun's angular diameter is about 0.536 degrees. These two values were used to calculate the scales by pixel for each camera. Figure 4.1.3 displays a test image captured by the science camera on the ground. The dark spots on the sun highlighted in the red circles were identified as

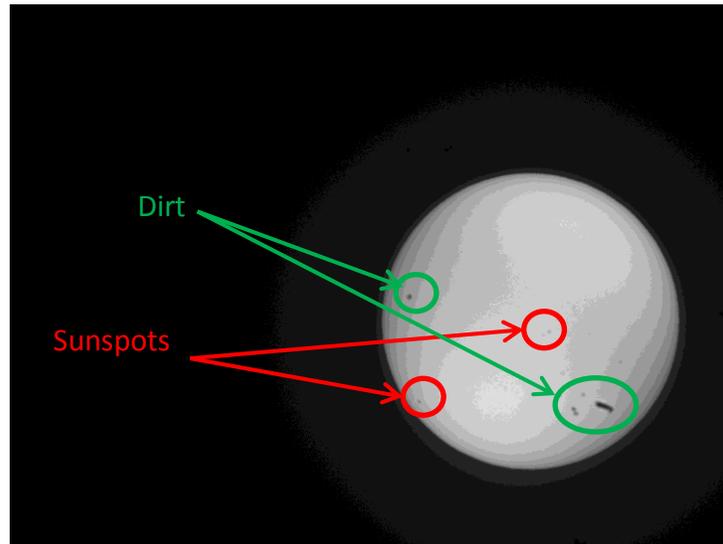


Figure 4.1.3 Test Science Camera image captured on ground

the science camera on the ground. The dark spots on the sun highlighted in the red circles were identified as



[Type text]



sunspots. The dark spots highlighted in the green circles are contamination on the lens of the camera. The dirt contamination was a result of the cameras not being cleaned properly before the test. Actual sunspots are differentiated from the dirt by multiple images of the sun. When comparing multiple images, the contamination would remain in the same location in the images and the sunspots would remain in the same location on the sun. The sunspots would appear to “move” with the sun as the sun changed location in multiple pictures. The contamination would stay in the same location in every image.

The Science Camera produced an image of the Sun with a diameter of 1056 pixels. The angular size of the image is about 0.000507 deg/px according to the angular diameter of the Sun stated above. Similarly, the scale of the image is approximately 1320 km/px according to the linear diameter of the Sun. This value means that each pixel is approximately 1320 km in side length, which is about the size of one granule. The total angular field of view of the Science Camera was 1.315 by 0.9864 degrees. This accuracy met the required target accuracy of 0.125 degrees for the tracking system.

The ADCS camera had a total field of view of 15.88 by 11.91 degrees to allow it to monitor the precision of the tracking system. It produced an image of the Sun with a diameter of 54 pixels. The angular size of the image is about 0.00992 deg/px according to the angular diameter of the Sun stated above. Moreover, the scale of the image is approximately 25,800 km/px according to the linear diameter of the Sun.

B. Structure

The entire structure was originally proposed to be of the dimensions 34cm X 28cm X 24cm; the final structure ended up being 41cm X 32cm X 37.45cm. It was originally proposed to include an electronics box, tensioned gear system, camera swing and top and side photodiode arrays, each containing 14 diodes. The final structure included an electronics box with a truss pattern cut into it to reduce the weight of the structure, a chain and sprocket system for positioning, camera swing with holes to reduce weight, and top and side photodiode arrays, each containing four diodes.

The proposed thermal system included insulating the electronics box, painting the external components of the structure white and heat sinking all the components of the payload directly to the HASP platform using a thermal epoxy. The final design entailed insulating the electronics box, the camera swing and the external side motor, painting the structure white to reflect as much heat as possible, and heat sinking the power board and motor drivers to HELIOS II using thermal paste and a putty epoxy. Figure 4.2.0 through Figure 4.2.4 display mechanical drawings of the HELIOS II structure. Figures 4.2.5 and Figure 4.2.6 display images of the final assembled HELIOS II structure



[Type text]

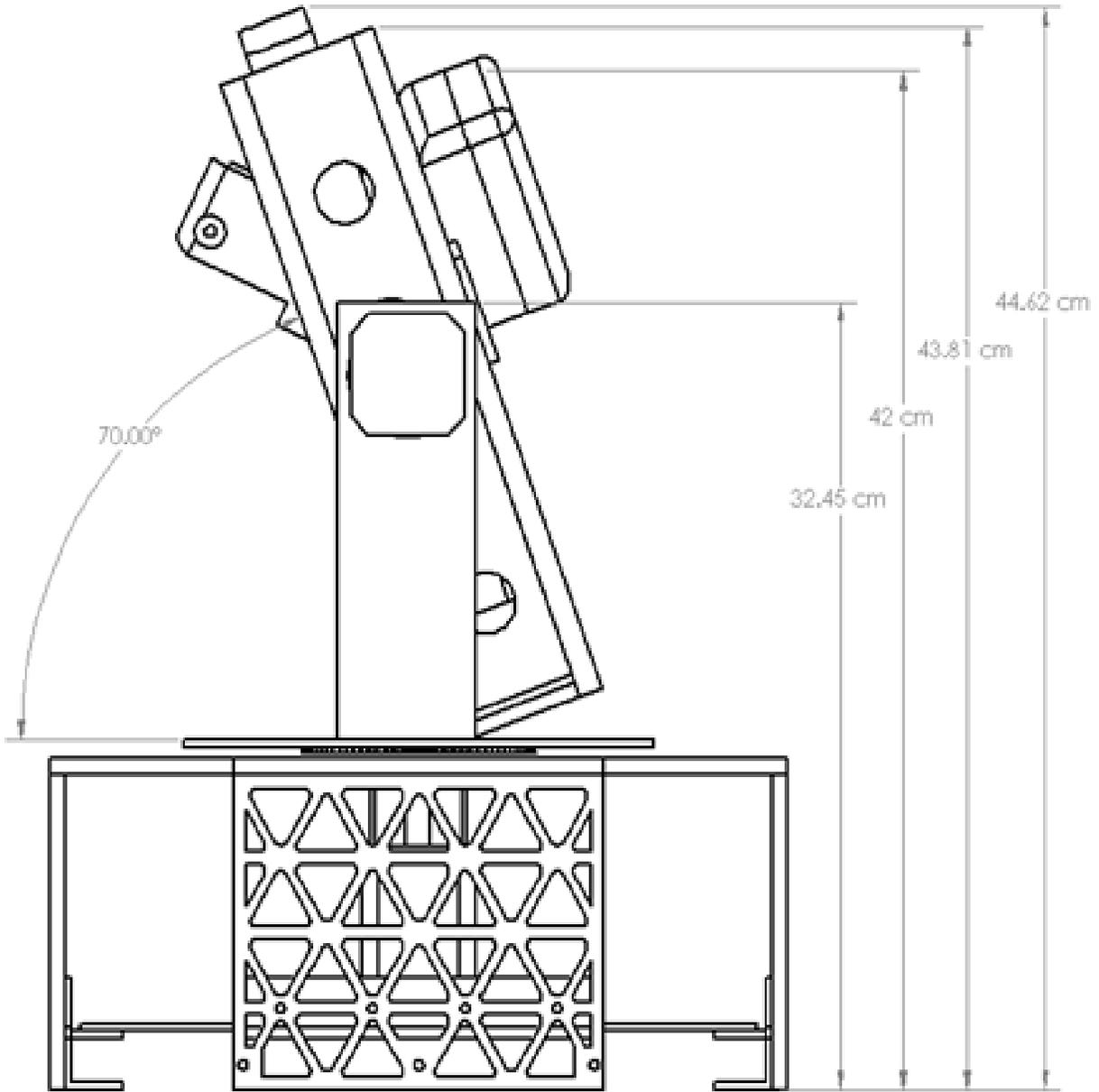


Figure 4.2.0 HELIOS II Final Assembly Angled to 70°



[Type text]

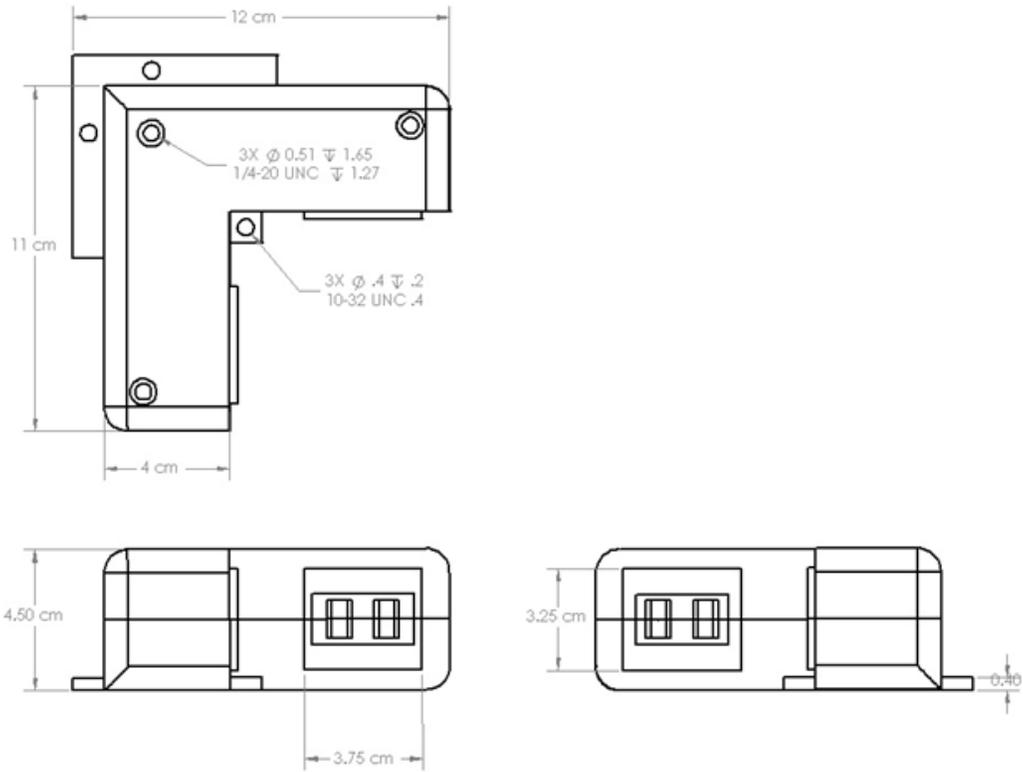


Figure 4.2.1 ADCS Photodiode arrays



[Type text]



- 1.) y-axis photo diode array
- 2.) x-axis photo diode array
- 3.) Science Camera
- 4.) ADCS camera
- 5.) y-axis motor
- 6.) x-axis motor

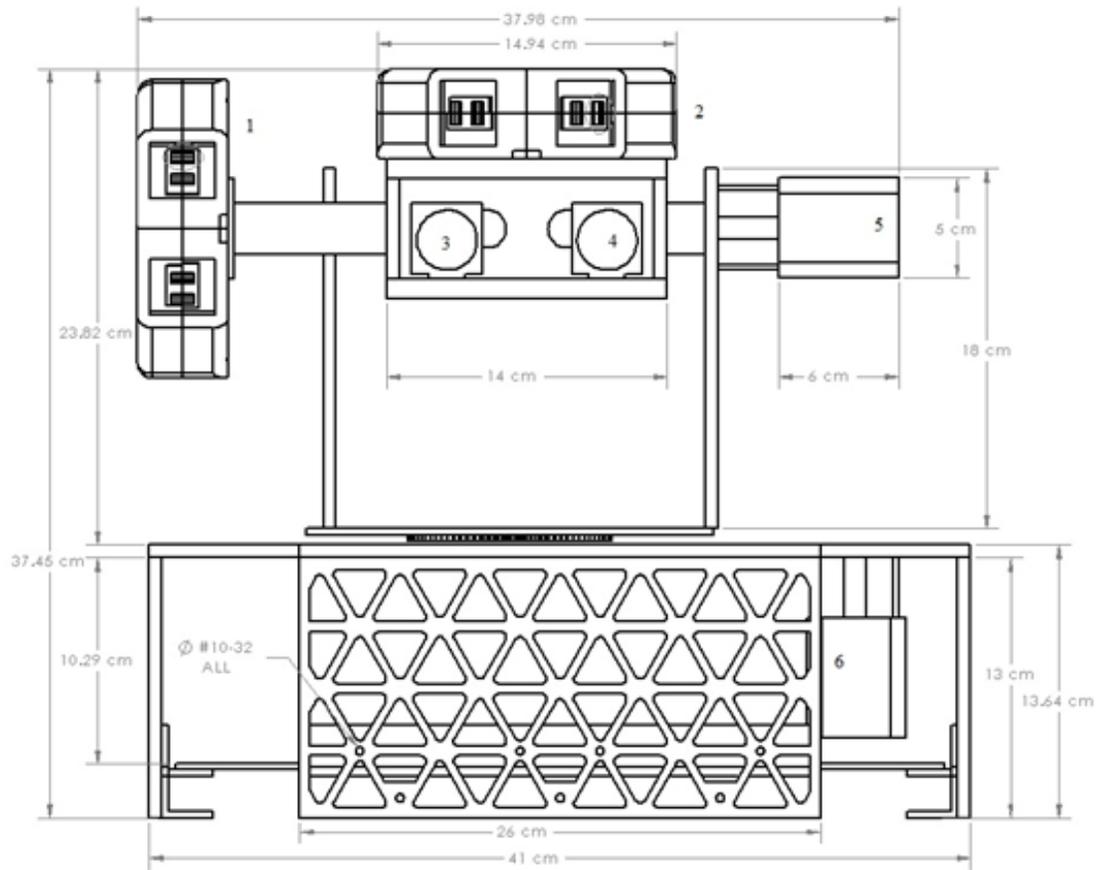


Figure 4.2.3 HELIOS II Final Assembly Front View



[Type text]



- 1.) x-axis photo diode array
- 2.) y-axis photo diode array
- 3.) y-axis motor
- 4.) Science camera
- 5.) ADCS camera

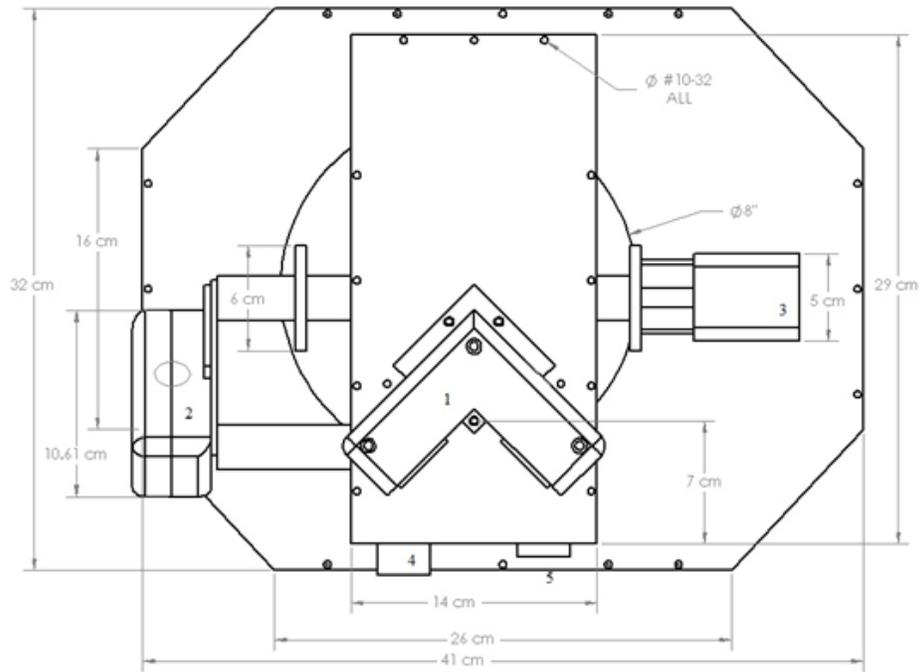


Figure 4.2.4 HELIOS II Final Assembly Top view



[Type text]

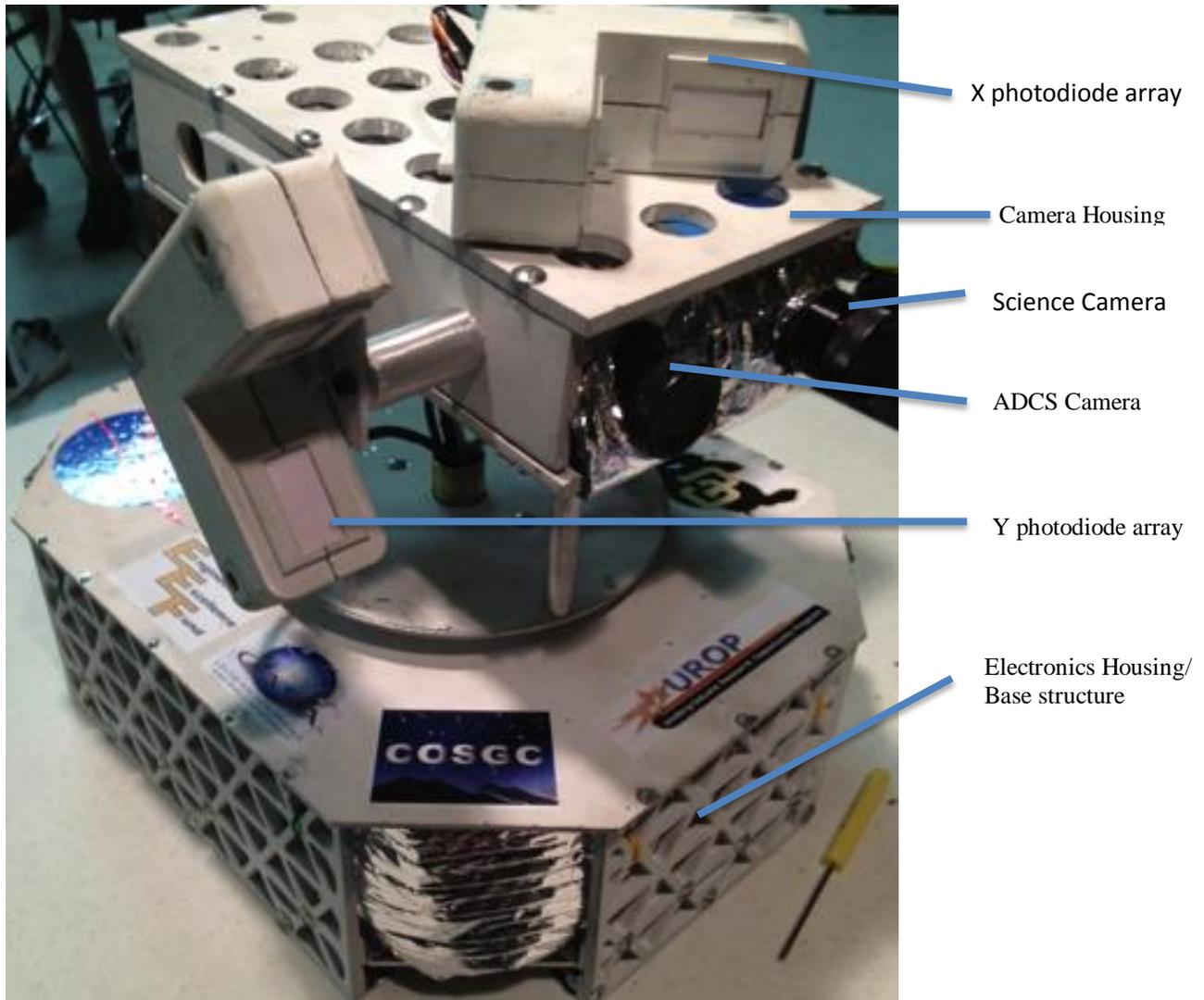


Figure 4.2.5 HELIOS II Final Assembly

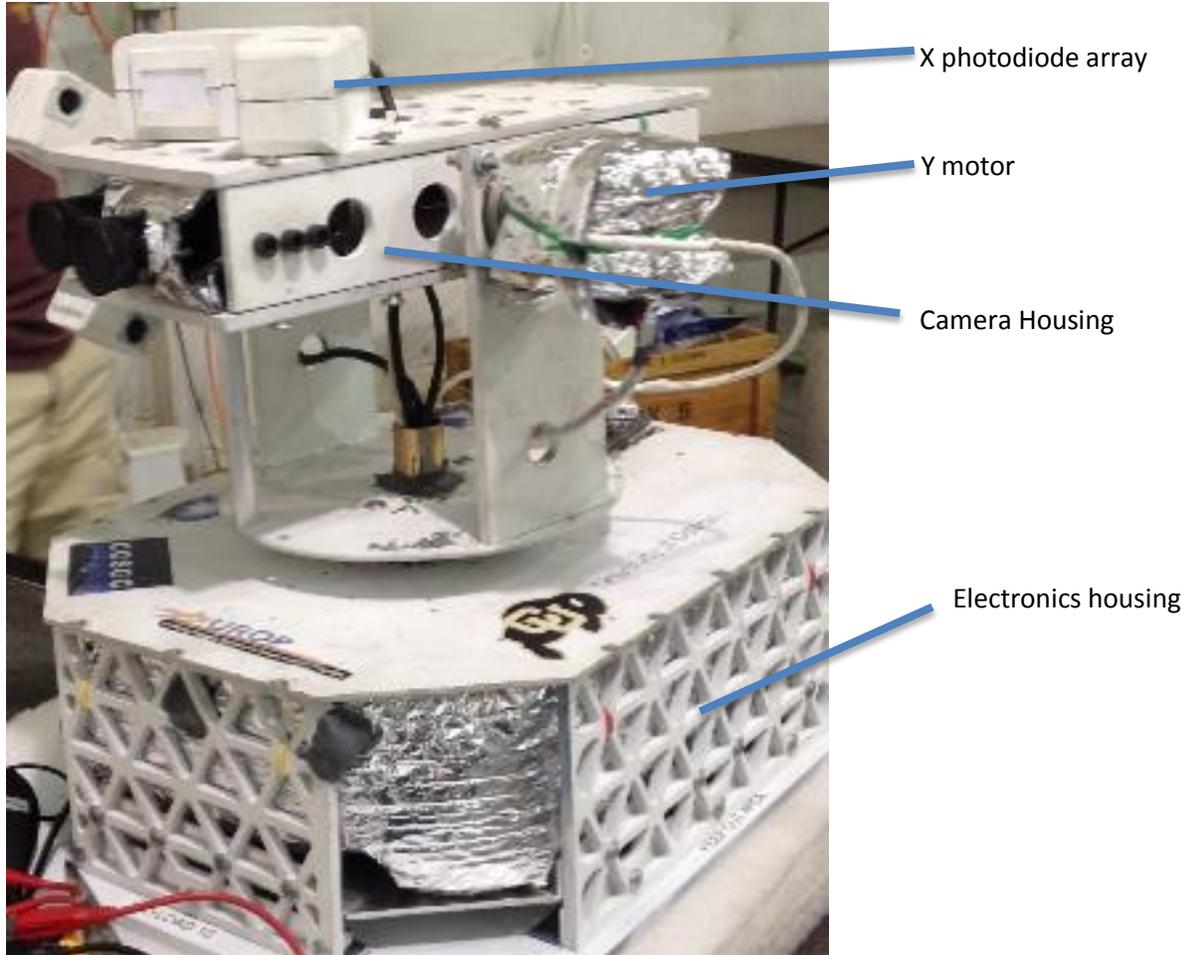


Figure 4.2.6 HELIOS II Final Assembly

C. Attitude Determination and Control System (ADCS)

The main job of the ADCS is to locate the Sun in the sky, and to orient the SWIS cameras towards the Sun. The ADCS uses two photodiode arrays to detect the Sun's position in the sky. The two photodiode arrays are attached to the SWIS camera housing. One photodiode array detects the Sun on the x-axis of the cameras. The second photodiode array detects the Sun in the y-axis of the cameras. Figure 4.3.0 shows the positioning of the photodiode arrays relative to the camera housing. An Arduino Mega microcontroller collects the photodiode readings. From the photodiode readings, the Arduino determines the Sun's position in the sky relative to the HELIOS II payload. Next, the Arduino commands two stepper motors to move the SWIS camera housing. One motor rotates the camera housing on the x-axis. The second motor rotates the camera housing in the y-axis. The Arduino commands the motors to rotate the camera housing until the cameras are centered on the Sun. The ADCS utilizes an active tracking system. That is, the ADCS continuously tracks the Sun. Constant tracking is necessary to counteract the rotation of the HASP platform.

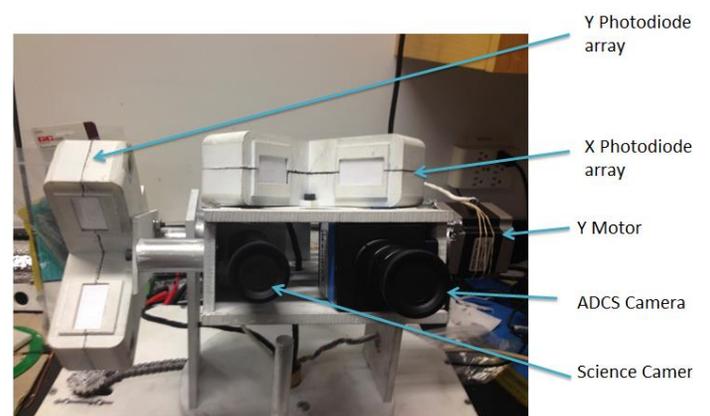


Figure 4.3.0 ADCS Photodiode arrays



1. Photodiode Arrays

Photodiodes were used to measure light from the Sun. Figure 4.3.1 displays a digital rendering of a photodiode. The ADCS used TEMD6010FX01 photodiodes in HELIOS II. The photodiodes output a current based on the amount of light measured by the sensor. As the light measured by the sensor increased, the photodiode output more current. If no light was detected by the photodiode, the photodiode output a “dark” current. The dark current for the TEMD6010FX01 photodiodes was about 0.1 nanoAmps. The current output by the photodiode was very small, outputting a max current of about 0.04 micro Amps. Because of this, Arduino Mega was unable to read changes in the raw output signal. An operational amplifier was used to increase the signal output by the photodiodes.

The ADCS used operational amplifiers, or op amps, to increase the output signal of the photodiodes. Figure 4.3.2 displays the printed circuit board for the op amps. The ADCS used CA3140A op amps in HELIOS II. After amplification, the photodiodes output 0 V in the dark and 2.8 V at full light saturation. The Arduino Mega is able to read changes in the amplified photodiode output signals.

When looking directly at the Sun, the photodiodes would become saturated. In order to solve this problem, light diffusers were placed in front of the photodiodes. The light diffusers consisted of 4 layers of white printer paper coated in scotch tape. With the filters in place, the photodiodes output 0.48 V while looking directly at the Sun. This is about 20% of their saturation value. This light filtration was done to ensure the photodiodes would not saturate above Earth’s atmosphere where sunlight is more intense. Additionally, this amount of filtration still offered very precise readings light intensity readings.

Two photodiode arrays were used in the ADCS. Figure 4.3.3 displays an image of the final photodiode array. One photodiode array was used to read light intensity in the x-axis. The second array measured the light intensity in the y-axis. Each photodiode array contained 4 photodiodes and 4 op amps. The arrays were arranged in an “L” shape, with two arms extending out at a 90 degree angle. Two photodiodes were placed on each arm in the photodiode array. The photodiode’s were grouped in “pairs”. The photodiodes were placed an equal distance away from the center of the array on opposite arms. Each array contained two photodiode pairs: a main photodiode pair and a backup photodiode pair.

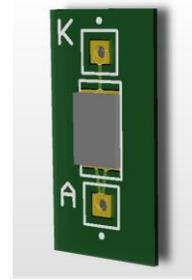


Figure 4.3.1 3D image of photodiode *K* is the cathode port and *A* is the anode port

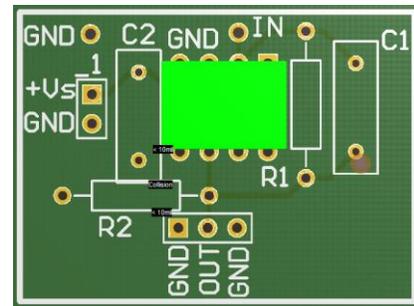
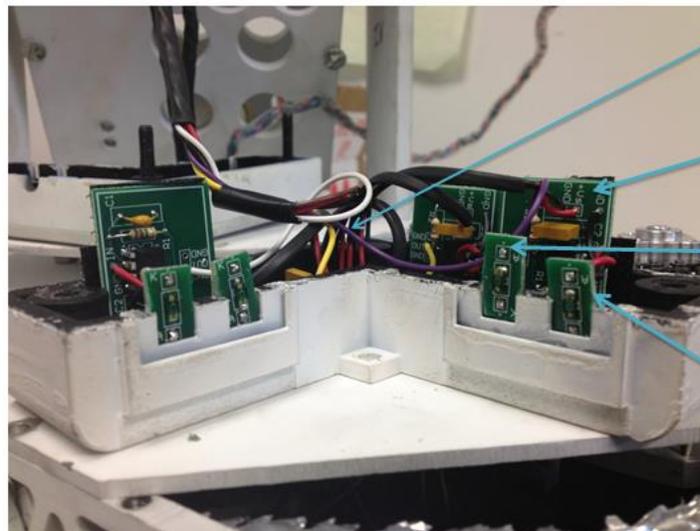


Figure 4.3.2 Op Amp Circuit



- Data & Power Lines
- Op Amp circuit
- Back-up Photodiode
- Main Photodiode

Figure 4.3.3 Cut away view of photodiode array.



The geometry of the array was designed to maximize the ability of the photodiodes to track the Sun. If the Sun was in the center of the array, each photodiode in the pair read the same light intensity. If one photodiode reads a greater light intensity, the Sun is off center, but still in the field of view of the photodiode array. The photodiodes were very sensitive to changes in angle. The photodiodes had a max sensitivity to angle changes at 45° from the normal. Based on the geometry of the array, when the Sun was in the center of the array, it would be at a 45° angle from the photodiodes. This is shown in Figure 4.3.4 displays the position of the photodiode array if the Sun was centered. The diodes were positioned in the array so that they would be at their max sensitivity to change when the Sun was near the center of the array.

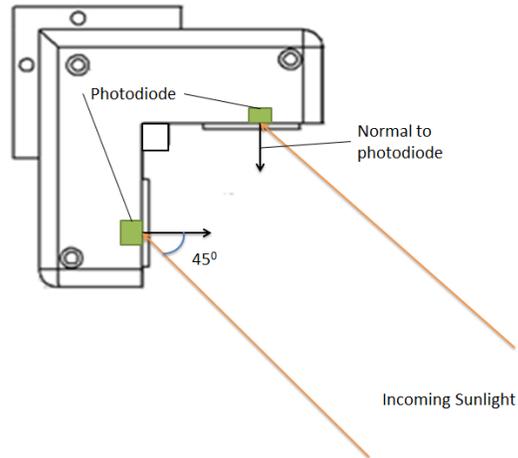


Figure 4.3.4 Incoming sunlight while Sun is at the center of the array.

2. Stepper Motors

Two stepper Motors were used to move the camera housing. One motor rotated the camera housing in the x plane and one motor rotated the camera housing in the y plane. A Big Easy driver was used to control the motor, and an Arduino micro controller sent commands to the drivers to control the motors. The Big Easy driver was used to power the motors, and received 7V of power from the EPS. The Big Easy driver had a built in current limiter. The current limiter was set to 1 A. Above 1 A, the drivers would overheat and behave erratically.

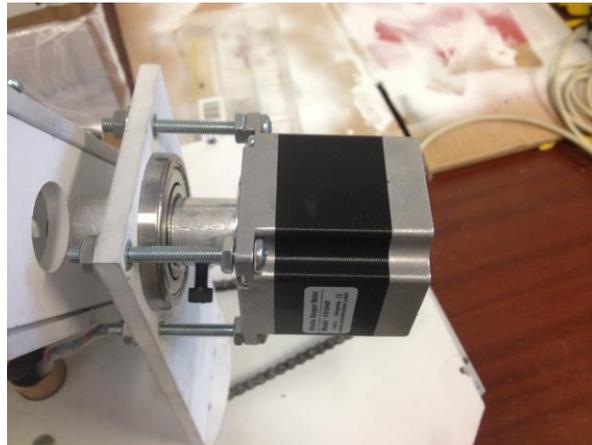


Figure 4.3.5 Y axis motor. 125 oz. stepper motor.
Motor was attached on the side of the camera housing

The Y motor was a 125 oz. stepper motor provided by SparkFun. The motor had a holding torque of 90 N.cm, and could move 200 steps per rotation. The Big Easy driver allowed the motors to micro-step. Micro-stepping was used in the final design. With micro-stepping, the motor could move 3200 steps per revolution. One step of the motor rotated the camera housing 0.1125 degrees. The Y motor was located on the side of the camera housing. This motor directly rotated the camera housing up or down based on the commands from the Arduino. The system was designed to move from 0° to 70° above the azimuth. However, the actual structure of HELIOS II only allowed for the camera housing to be raised to a max of 63° above the azimuth. If the motor tried to raise the camera housing above 63° , the camera housing would hit the circular mounting plate below

The X motor was a 68 oz. stepper motor provided by SparkFun. The motor has a holding torque of 48 N.cm and could move 200 steps per rotation. The Big Easy driver allowed the motors to micro-step. Micro-stepping was used in the final design. With micro-stepping, the motor could move 3200 steps per revolution before the gear ratio. A 3 to 1 gear ratio was used for the x axis motor. This allowed a smaller motor to be used to rotate the approximately 7 kg camera housing. After the gear ratio and micro stepping were applied, the X motor had a holding torque of 144 N.cm and could move 9600 steps per revolution. With this accuracy, the x motor was able to point the cameras to within 0.038 degrees.

3. Arduino Microcontroller

An Arduino Mega Microcontroller was used to read the photodiodes and give commands to the motors. The Arduino mega has 12 built in analog to digital converters and 3 TX/RX lines. The Arduino reads all incoming



signals as digital. The built-in analog to digital converters read a max of 3.3 V. The Arduino digital read is 10 bit. This means the Arduino reads from 0 to 1023 decimal. The conversion factor for volts to decimal was 1024 for 3.3 V. This simplified to 310.3 bits per volt. The Arduino Mega read the input of the photodiodes as digital. The Arduino Mega used the Big Easy driver to control the motors. The AccelStepper.h library was used to code the motor controls.

4. Control Algorithm

The ADCS control algorithm consisted of two main parts: the Macro Track Loop and the Micro Track Loop. The Macro Track Loop was used to find the Sun. The ADCS entered the Macro Track Loop immediately after it was powered on. The ADCS remained in the Macro Track Loop until the system determined that the Sun was in the field of view of the photodiodes. The Micro Track Loop was used to center the cameras on the Sun. The Micro Track Loop compared the readings of the photodiode pairs in the X and Y photodiode array. The Micro Track Loop then calculated how far the motors would move the array in order to center the Sun.

i. Macro Track Loop Algorithm:

1. Enter loop after system is powered on
2. Is the Sun in field of view of photodiode arrays?
 - a. Yes (If one photodiode reads above 50): enter Micro Track Loop
 - b. No: continue into Macro Track Loop
3. Rotate camera housing 360° clockwise
4. Rotate camera housing 360° counter clockwise
5. Repeat 3 & 4 until the Sun in FOV of photodiode arrays

ii. Micro Track Loop Algorithm:

1. Compare X diode readings
 - If Right > Left, move counter clockwise
 - If Right < Left, move clockwise
 2. Compare Y diode readings
 - If Top > Bottom, move down
 - If Top < Bottom, move up
- 1 decimal difference = 1 correction step
Larger difference:
- more steps per command
 - Faster motor speed



Figure 4.3.6 X axis motor. 68 oz. stepper motor. Motor was attached under the HELIOS II top plate



Figure 4.3.7 An Arduino Mega Microcontroller.

5. Differences from Proposal

HELIOS II has made several major design changes since the HASP proposal. The first major change was in the ADCS algorithm used for tracking the Sun. In the proposal, the ADCS used a passive tracking algorithm, sampling the photodiode readings several times before giving a movement command. The final design, the ADCS



used an active tracking algorithm. The final algorithm was constantly sampling the photodiodes and immediately giving movement commands based on the readings of the diodes. Additionally, the final tracking algorithm featured the ability to control the speed of the motors.

A second major design change was in the shape of the photodiode arrays. It was proposed that the ADCS would use a circular photodiode array on the X axis and a semi-circle photodiode array on the Y axis. In the final design, an “L” shaped array was used on both the X and Y arrays. A photodiode array is shown in figure 3.3.3.

A third design change that happened since the proposal was a reduced number of photodiodes. In the proposed ADCS, 14 photodiodes were used in the X array and 6 photodiodes were used in the Y array. The final design had 4 photodiodes in the X array and 4 photodiodes in the Y array. The final design used two main diodes and two back up diodes per array.

D. Electronic Power System (EPS)

The final design of the Electronic Power System, hereafter called EPS, was composed of two major parts. The first part regulated and distributed power, and the second part was an Arduino Mega interface capability. These two sections were fully integrated into a single power board. The board measured 27cm x 16.5cm and had a maximum height of 8.13 cm making it largest component in the payload housing. It was designed using the Altium PCB Design software and populated in-house at the Colorado Space Grant Consortium. The power board consisted of two general sections.

First, power was distributed and regulated using 4 power rails. Each rail had a buck converter to efficiently step down the voltage from 30 V to 9 V for the Arduinos, 9 V for the Y axis motor, 9 V for the X axis motor, and 6 V for the CDH processor. These voltages were then regulated with low dropout linear regulators for redundancy and in case of buck converter failure. The regulators dropped the voltage by another volt on each line.

The buck converters were a necessary component because of their high efficiency and their ability to output more current than what they received. In general, a buck converter will output the same amount of power it receives, with a small decrease due to efficiency less than unity. The power output from the buck converter can be found from

$$P_{in} = \eta * P_{out} \quad (1)$$

Where η is the converter efficiency. The current can be found from the power from

$$I = \frac{P}{V} \quad (2)$$

Substituting Eqn.(2) into Eqn.(1) and solving for the ratio of the input current to the output current gives

$$\frac{I_{in}}{I_{out}} = \frac{V_{out}}{V_{in}} * \eta \quad (3)$$

Eqn(3) that for a given efficiency, the current output can be larger than the current input if the voltage input is larger than the voltage output. For high efficiencies, a buck converter will draw only as much power as the load requires. This will minimize the energy lost to heat. On the other hand, linear regulators require as much current as they output. The same analysis can be done on linear regulators as for buck converters; in this case, the power is given by

$$P_{lost} = P_{in} - P_{out} \quad (4)$$

Where P_{lost} is the power lost to heat. In terms of current and voltage, where the current input equals current output, Eqn(4) becomes Eqn(5):

$$P_{lost} = IV_{in} - IV_{out} \quad (5)$$

Eqn. (5) shows that the energy lost to heat is proportional to the potential difference between the input and the output for a linear regulator.

These two considerations were very important when designing the power board. The highest voltage that any one of the components required was 7 V. This represents a potential difference of 23 V across a linear regulator. With a load current of 0.8 A – which is what the motors required - the power loss would have been 18.4 W. This number was significantly reduced to a 1.2 W maximum power loss by using the buck converters. The linear regulator still did serve a purpose, however. After the voltages were regulated by the buck converters, the linear regulators were placed in series to further regulate voltage, but with a minimal potential difference across them so the power loss was small: 1 W. The reason for the further use of the linear regulators was that they were capable of regulating power if the buck converters failed. Power regulation would not have been as efficient, and ultimately they would have also failed due to power dissipation limits and overheating, but they could increase chances of mission success if they sustained the mission even for a small period of time after the buck converters failed.

In addition to linear regulators and buck converters, a current sensor and an N-Channel MOSFET was implemented on each power rail for further power regulation. The current sensors measured a differential voltage from a resistor in series with the load and calculated the current across that resistor. The current sensors y then



output a signal to the Arduino Mega, which controlled the MOSFETS using the digital pins and their voltage output capability. The Arduino effectively “powered off” a component if it exceeded its current limit.

In addition to controlling the power MOSFETS, the Arduino Mega played the main role in the second section of the power board. The Arduino measured environmental statistics – pressure, temperature, humidity- as well as temperatures of certain components in the payload housing and acceleration of the payload on three axes. Additionally, the Arduino received time stamps from an external timekeeper, the ChronoDot. This clock was battery powered so that time would still be kept in the event that the payload was powered off. The Arduino then relayed the data to the CDH subsystem through serial communication, where it was stored for post flight analysis.

Fig. 4.4.0 demonstrates the general layout of the power board, not including components.

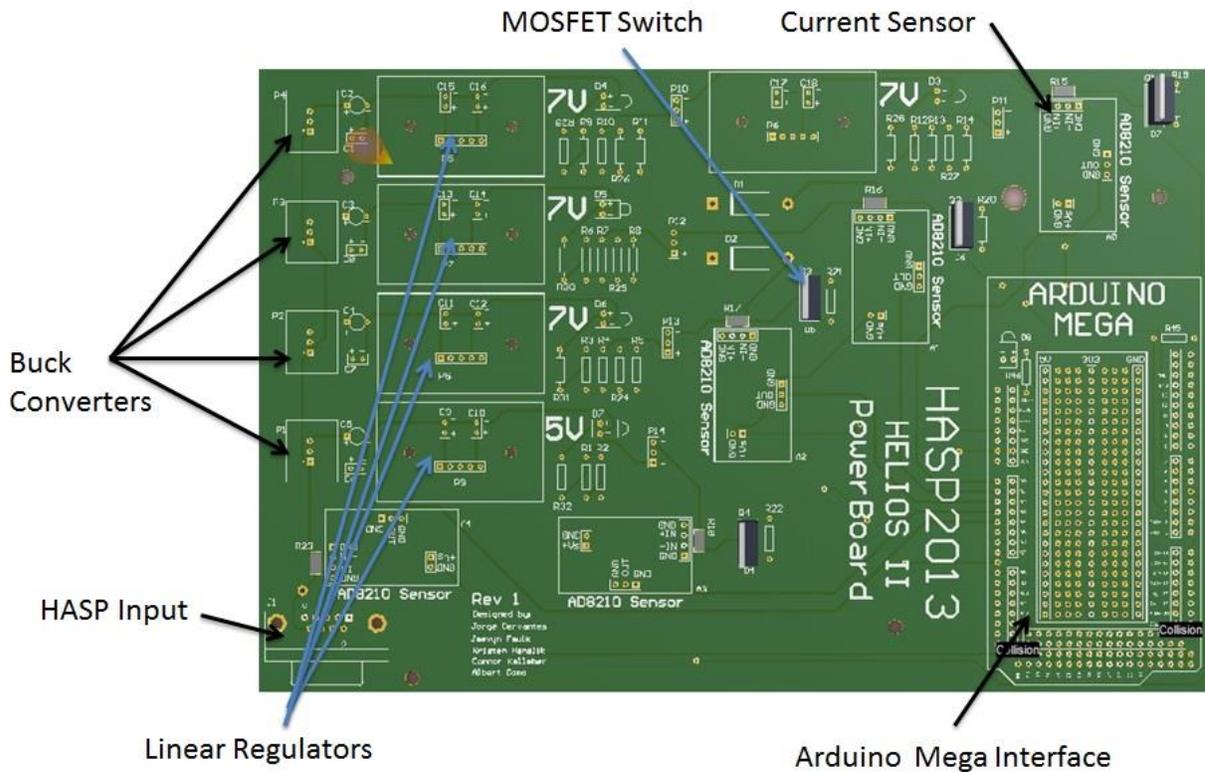


Figure 4.4.0. Final Design. The figure displays the final design for the power board. The buck converters are on the far left, followed by the linear regulators, current sensors, and MOSFETS. Also note the Arduino MEGA interface on the right side.

1. Differences From Proposal

The final EPS design was actually quite different than the design in the proposal. The proposed design had 2 power rails, whereas the actual board had 4. Also, it was proposed to use relay switches to regulate power, but the final design contained N-Channel MOSFET switches. Finally, the EPS system acquired the Arduino interface and the ChronoDot timekeeper, neither of which were included in the proposal.

a. Power Rails:

The proposed EPS design only contained two power rails, whereas the final design included 4. This change was made so that power could be distributed more efficiently in terms of heat loss. For any given efficiency, a larger load on the buck converter represented a larger voltage drop across the converter. By adding more power lines, heat dissipation could be handled more effectively by decreasing the load on any single converter. In addition, it was important to add more power rails so that each component was on its own power rail. This made it easier to power off components individually since they were isolated from



all others. For example, if the ADCS Arduino needed to be restarted, there was no need to also remove power from the CDH CPU. Placing the CPU on a different power line than the Arduino allowed for removal of power to *only* the Arduino.

b. Switches:

It was proposed that relay switches be used to regulate power across each power rail. Instead, the final design included N-Channel MOSFETS as switches. These were chosen over relay switches because they are smaller than the relay switches; they physically require less space on the power board. Additionally, the MOSFETS are easily controllable by applying a 5V potential difference at the gate, which the Arduino could easily provide with the digital pins' output voltage capability.

c. Arduino Mega Interface:

The Arduino Mega interface was a design change that occurred after submission of the proposal. The Arduino Mega was not only used to control the switches and regulate power, but it also served as a central processing device for all environmental, acceleration, and component temperature data. The proposed design placed the aforementioned sensors on the CDH processor, but many of the sensors output analog signals, which the processor could not read without analog to digital (A/D) converters. It was decided that it was much simpler to read the sensors with the EPS Arduino, which had A/D converters built in, rather than add A/D converters to the CDH processor. Ultimately, the data was still handled and stored by the CDH processor because the Arduino transferred all the information to the CDH processor through serial communication.

d. ChronoDot Timekeeper:

A battery powered GMT timekeeper was added to the EPS system through the Arduino to facilitate post flight analysis. It was essential that the clock be battery powered so that time was kept even when the payload was powered off.

E. Command & Data Handling (CDH)

CDH communicates directly with the EPS and indirectly with the ADCS as displayed in Figure 4.5.0. Upon receiving a command from ground station, CDH verified that the command is valid by using a process based on code shown to the CDH team by Mike Stewart of the LSU HASP program. Once the command was verified, CDH passed it on to the EPS Arduino Mega via TX/RX. If the command was intended for EPS then EPS executed it. If the command was meant for ADCS, EPS passed it on to the ADCS Arduino Mega via TX/RX. Verification was sent back from whichever subsystem the command was intended for. If the command was intended for ADCS, the response was sent back through the EPS to CDH. If the command was intended for EPS, a response was sent directly back to CDH. This response was then sent to ground and stored in a data file. This process was tested using a program called Term232 suggested by Doug Granger of the LSU HASP team. Term232 allowed the CDH team to send a two byte hexadecimal command to the payload. This flight simulation was useful in the verification of CDH command processing and notification abilities. Data is received from both subsystems in a similar manner where photodiode readings from the ADCS are passed through the EPS to CDH for storage. The RTC (real time clock) and sensors are located on the EPS Arduino Mega. This data along with the ADCS data was sent via TX/RX to CDH for storage and downlink to ground.

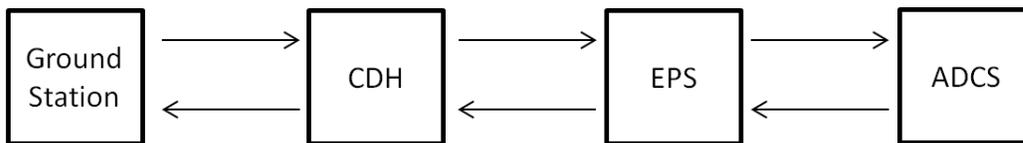


Figure 4.5.0 C&DH Communication



[Type text]

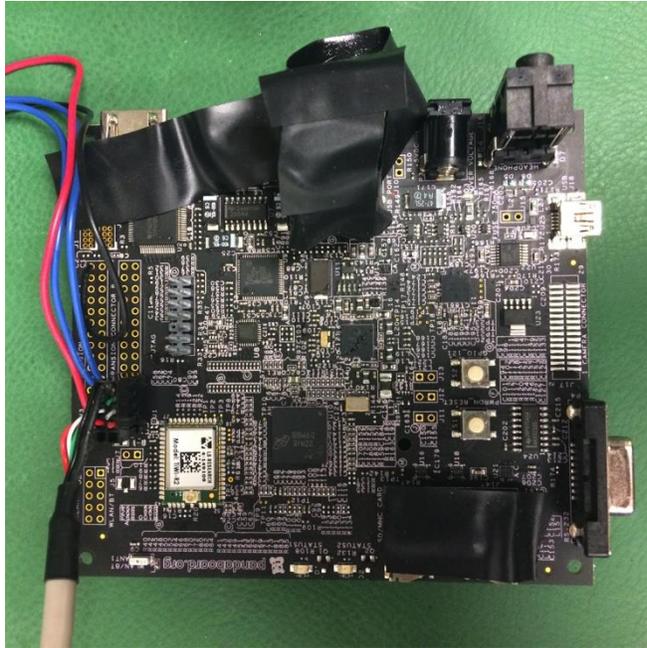


Figure 4.5.1 Pandaboard



Figure 4.5.2 Solid State Drive

IC	Use	Hex Value	Purpose
a	Switch to backup diode pair- x axis	1A,1B	In case of main diode pair failure on the x axis, ADCS shall switch to a backup diode pair on the same axis.
b	Revert to main diode pair- x axis	1C,1D	Revert to main diode pair on x axis.
c	Switch to backup diode pair- y axis	2A,2B	In case of main diode pair failure on the x axis, ADCS shall switch to a backup diode pair on the same axis.
d	Revert to main diode pair- y axis	2C,2D	Revert to main diode pair on x axis.
e	Initialize pan mode	3A,3B	In case of ADCS failure, initialize pan mode.
f	Revert to ADCS main function	3C,3D	Revert to ADCS main function.
g	Power off ADCS	4A,4B	Power off ADCS
h	Power on ADCS	4C,4D	Power on ADCS
N/A	Initialize backup camera mode	7A,7B	Switch to non-sun tracking mode.
N/A	Revert to main camera mode	7C,7D	Revert to sun tracking mode.

Figure 4.5.3 C&DH Serial Commands list



[Type text]



1. Outline Of Main Flight Code

1. Establish serial connection with the EPS
2. Establish serial connection with the HASP platform
3. Start thread to check for and receive data from the EPS (runs continuously)
 - a. Read from the EPS serial line
 - b. If length of reading is not 0 and is less than 1
 - i. Then response back from the EPS is what was received
 - ii. Verify which command was received
 - iii. Once verified, send notification of response received to ground
 - c. If length of reading is greater than 18
 - i. Then a data packet from the EPS is received
 - ii. Store data on the SSD
 - iii. Send data to ground
4. Start thread to check for commands from ground (runs continuously)
 - a. Read from the HASP serial line
 - b. If the length of the reading is not 0
 - i. If the command is valid
 1. Notify EPS of command
 2. Send notification to ground
 3. Write the notification to the main data file
5. Begin infinite while loop calling the camera function
 - a. Take ten ADCS camera pictures. Take one Science camera picture

2. Differences from Proposal

The final design proved to be somewhat different than proposed. All sensors were implemented on the EPS Arduino Mega instead of on an Arduino Uno specifically reserved for that purpose. EPS monitored system health from those sensors and responded accordingly instead of CDH as proposed. In the proposal it was stated that CDH would monitor the health of the SWIS subsystem but that statement proved to be not applicable. The ADCS subsystem used an Arduino Mega instead of an Arduino Due. Figure 4.5.0 displays a basic portrayal of the current communication lines between subsystems. These are the only major differences from what had been proposed.



V. Mission Results

A. Science Results

The Science Camera captured 852 pictures in total, only one of which contained the Sun. This yielded a success rate of 0.117%, much lower than expected. The ADCS camera also had a smaller success rate than anticipated. 1663 images were captured and only 106 contained the Sun. This put the success rate of the ADCS camera at 6.4%. The one image of the Sun recorded by the Science Camera contained no resolvable solar features. Fortunately, the pictures of the Sun taken by the ADCS camera were able to be used.

The images captured by the ADCS camera are distorted because the camera applied a high gain. The gain was manually set to the lowest value in the code, but the camera operated in auto-gain mode due to an error in the libraries that controlled the CCD camera. This error caused the effect that appears to be over saturation (see Figure 5.1.2). It is possible that this error also affected the Science Camera. However, the Science Camera image appears to be at an appropriate gain.

The image recorded by the Science Camera (see Figure 5.1.0) contains no visible sunspots. The image was compared against an image recorded by SOHO on the same day (see Figure 5.1.3) to verify that there were no sunspots in the quadrant that was captured. The surface of the Sun in the image recorded by HELIOS II appears to show granulation. One pixel of the Science Camera was roughly the diameter of the average granule.

This is not conclusive evidence that granulation was observed, unfortunately. The magnification of the Science Camera was not great enough to satisfy the Rayleigh criterion for granulation. The Science Camera had a magnification that allowed it to make a distinction between two objects that had an angular separation as small as 0.0018 degrees. The average angular separation of two granules is 0.00038 degrees, approximately 5 times smaller than the capability of the Science Camera. This means that the Science Camera is mathematically unable to view granulation. The Science Camera failed to view any solar activity.

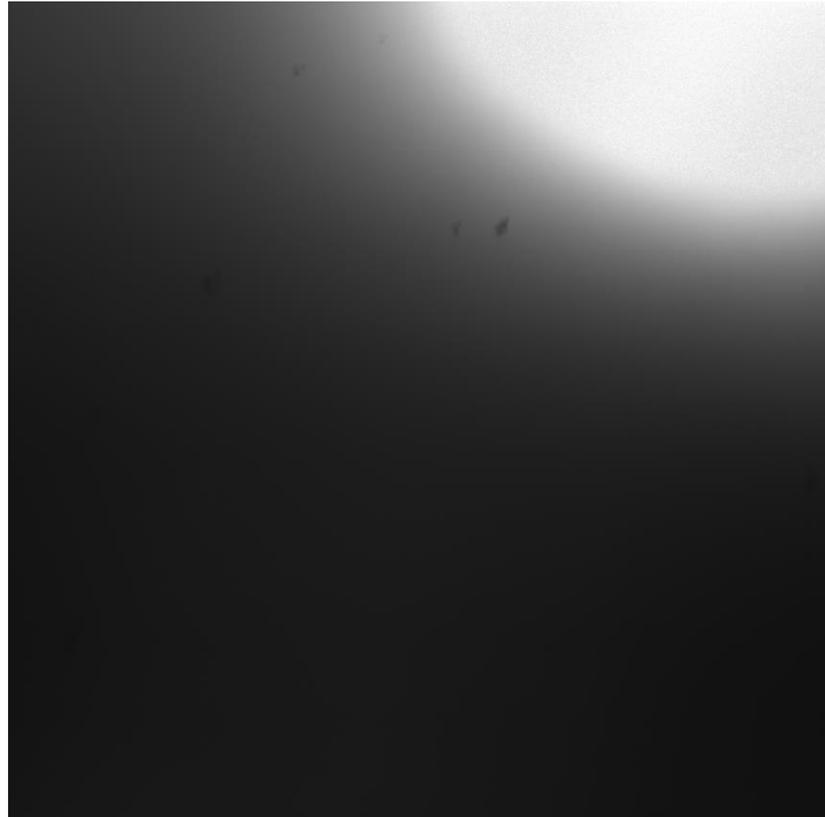


Figure 5.1.0 Science Camera image containing the Sun.

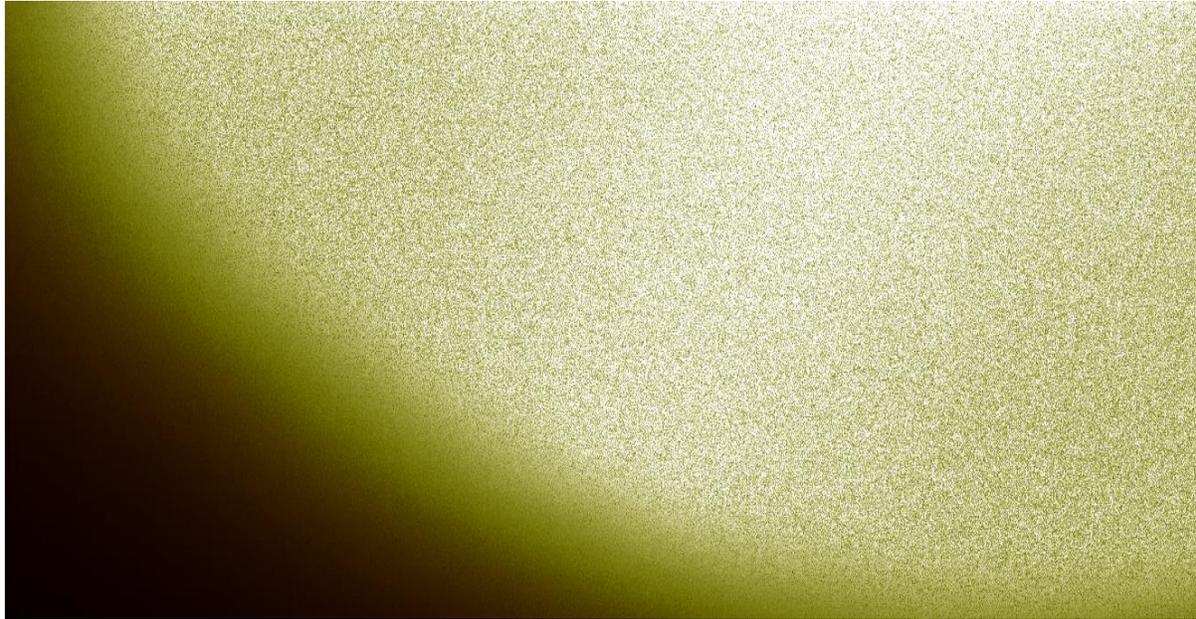


Figure 5.1.1 Science Camera image of Sun with false coloring and increased contrast

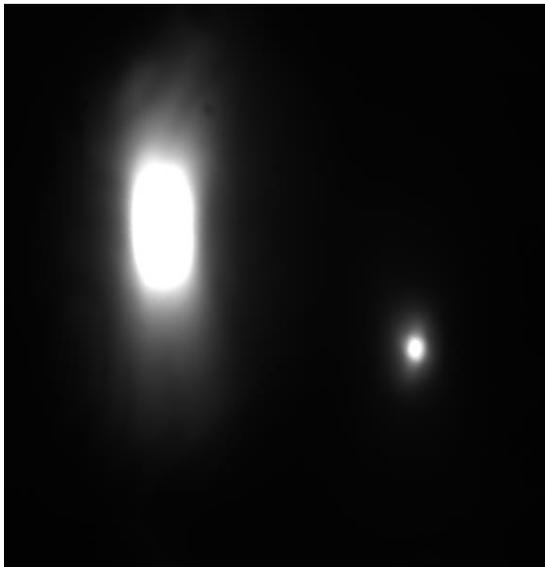


Figure 5.1.2 A typical ADCS Camera image. *It appears as though there are two suns in this image. The larger “sun” is the actual Sun. The smaller “sun” is a reflection about the center of the image. This reflection was captured because the Pandaboard was unable to turn off the auto-gain feature on the cameras.*

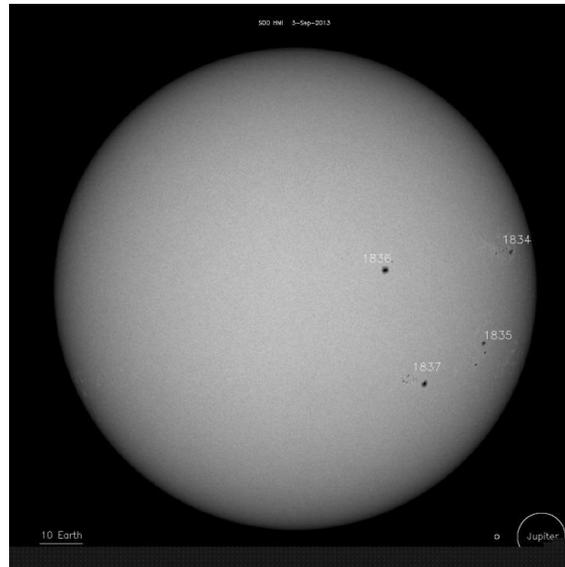


Figure 5.1.3 SOHO Image in H-Alpha. *This image was compared with the image captured by HELIOS II to verify the absence of sunspots in the small portion of the Sun that was observed. This image was taken on the day of flight, September 3, 2013.*

The image that HELIOS II captured was compared with an image from SOHO in the same wavelength. The image from SOHO verified that there were no sunspots in the quadrant observed. While this does not prove that SWIS was functioning properly, it is reassuring. If HELIOS II had captured a full image of the Sun, the results would be compared with SOHO to verify their accuracy.



B. ADCS Performance

The Science Camera captured 852 images and only of those images captured the Sun. This means the Sun was in 0.117% of the captured Science Camera images. The ADCS Camera captured a total of 1663 images. The Sun was in 106 of the ADCS images. This means 6.4% of the ADCS images contained the Sun. The images in the ADCS Camera that contained the Sun were used to characterize the accuracy of the ADCS while the Sun was in the field of view of the cameras. If the Sun was in the center of the ADCS camera image, the Sun would also be in the center of a Science Camera image captured at the same time and camera orientation.

The ADCS Camera was oriented sideways during the flight. Figure 5.2.0 contains an image showing the orientation of the ADCS camera. This orientation was used to calculate the Sun's distance from the center of the ADCS Camera.

Figure 5.2.1 is a graph showing the location of the center of the Sun in the ADCS Camera image. Each blue dot corresponds to the center of the Sun in a captured ADCS Camera image. The red dot displays the mean location of the center of the Sun. The location (0, 0) in Figure 5.2.1 corresponds to the center of the ADCS Camera.

From the ADCS images, it was calculated that the Sun's average position in the ADCS Camera was 3.94 degrees above the center of the image and 1.98 degrees to the right of the center of the image. The Sun's position on the Y axis had a standard deviation of 2.95 degrees. The Sun's position on the X axis had a standard deviation of 0.935 degrees.

1. Failure Analysis

The ADCS did not perform as well as it was designed to. Only 1 Science Camera contained an image of the Sun. The lighting environment on the HASP platform caused the inaccuracy in the ADCS. During the flight, shadows cast by cables connecting the balloon to the HASP platform would fall over HELIOS II. These shadows would cause the photodiodes in the ADCS system to read different values, and would cause inaccurate pointing of the ADCS.

Additionally, sunlight reflected off the HASP balloon caused the inaccuracy observed in the Y axis. The geometry of the Y photodiode array caused it to look at the Sun and the balloon for the majority of the flight. The balloon would be in the FOV of the bottom photodiode in the Y photodiode array when the Sun was at an angle of elevation of 45° or larger. Figure 5.2.1 depicts the balloon and the Y photodiode array when the Sun is at 0° angle of elevation. Figure 5.2.3 depicts the balloon and the Y photodiode array when the Sun is at a 45° angle of elevation. Because the balloon was in the FOV of the bottom diode, it caused the bottom diode to read slightly higher values

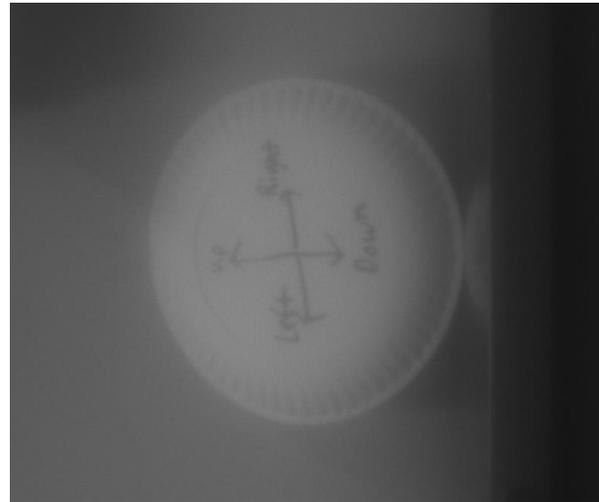
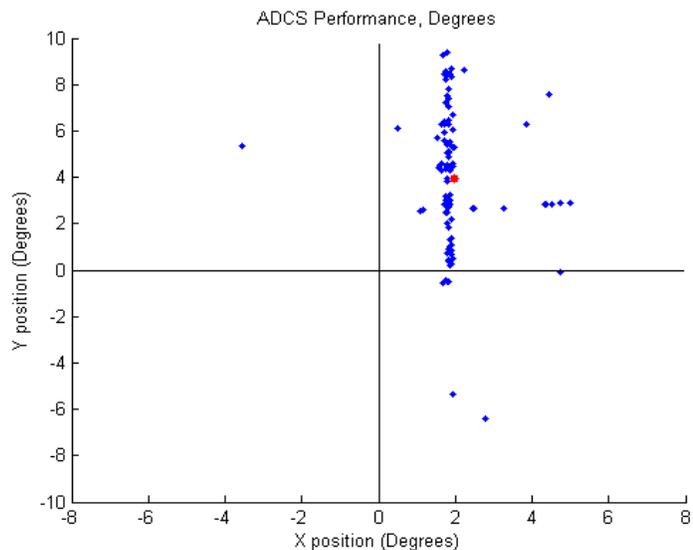


Figure 5.2.0 A test ADCS camera image. This image displays how the ADCS Camera was oriented on the final assembly of HELIOS II





that it would if the balloon were not in view. Because the bottom diode read higher values, the control algorithm commanded the arrays to move down in order for the values to be equal. The balloon caused the ADCS to think the Sun was lower in the sky than it truly was. This is seen in Figure 4.2.2. If the array was pointing below the Sun, the Sun's center would be in the top half of the image.

Additionally, the Y axis had a standard deviation that was about 3 times larger than the standard deviation of the X axis. The balloon was not constantly reflecting the same amount of light towards HELIOS II. The variations in the reflections caused variations in the readings of the bottom photodiodes. As a result, the Sun's position in the Y axis array had a larger standard deviation than the X axis.

C. EPS Results

Unfortunately, the CDH processor stopped communicating approximately 6 hours after launch, or about 4 hours after startup. Cosmo Cam feed demonstrated that the payload was still powered on, which implied that the CDH stopped working due to internal failure. Post-flight analysis provided insight as to the reason for the failure of the CDH processor.

Fig. 5.3.0 displays the temperatures of the components in the payload housing for the duration of the flight. The data in gray is data recorded by the LSU HASP team. The data was from the "Electronics Bay Free" on the HASP platform. Fig. 5.3.1 displays the same temperature profiles, not including the HASP data. In both figures, the temperature is only recorded until about 400 minutes, at which point the CDH processor stopped communicating with the ground due to reasons explained later. At the time that HELIOS II was powered on, the temperature was about -20 degrees Celsius. After startup the temperature only increased, until it reached a maximum recorded temperature of about 65 degrees Celsius. In addition, and as seen from Fig. 2, there are other times at which the CDH processor stopped communicating with the ground. These times are represented by the discontinuities in the temperature data. Another feature to note about Fig. 5.3.1 is that the temperature of the motor drivers decreased when the ADCS system was powered off. The temperature resumed its upward trend when the ADCS system was powered on again. In addition, the entire payload was powered off at mid-day, when the Sun was outside of the field of view of the camera housing. This is represented by the discontinuity in data, *in addition* to the drop in temperature. To explain, when the payload stopped communicating there were discontinuities in the data, but when communication was restored the temperature was higher than the last recorded temperature – payload

[Type text]

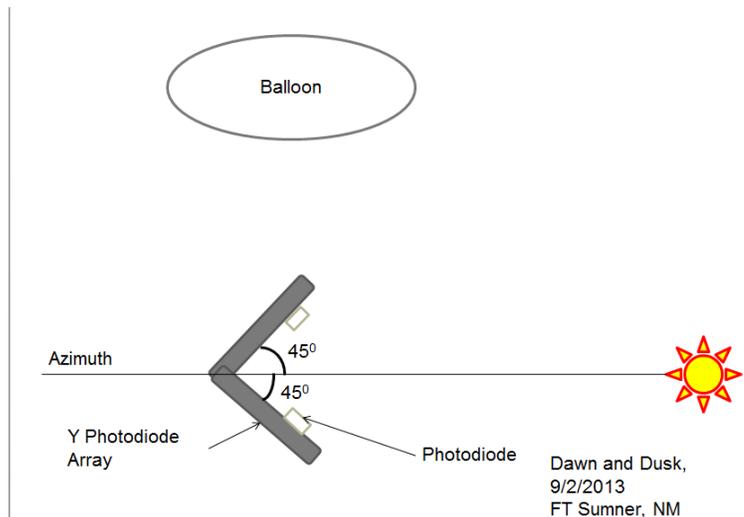


Figure 5.2.2 ADCS view of the sun and balloon when Sun is at an angle of elevation of 0°.

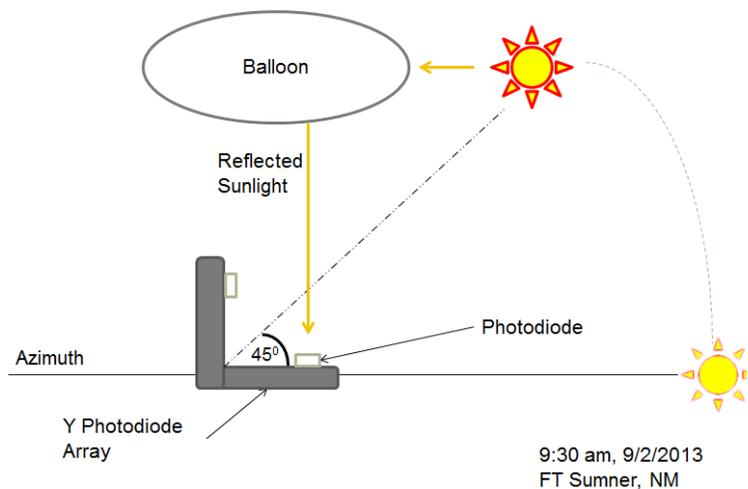


Figure 5.2.3 ADCS view of Sun and balloon when Sun is at 45° angle of elevation. The balloon is in full view of the bottom photodiode. Light reflected off the balloon is read by the bottom diode



[Type text]



components were still operating and dissipating heat. On the other hand, when the payload was powered off, the discontinuity in data was followed by a temperature smaller than the last recorded temperature- the payload components were powered off.

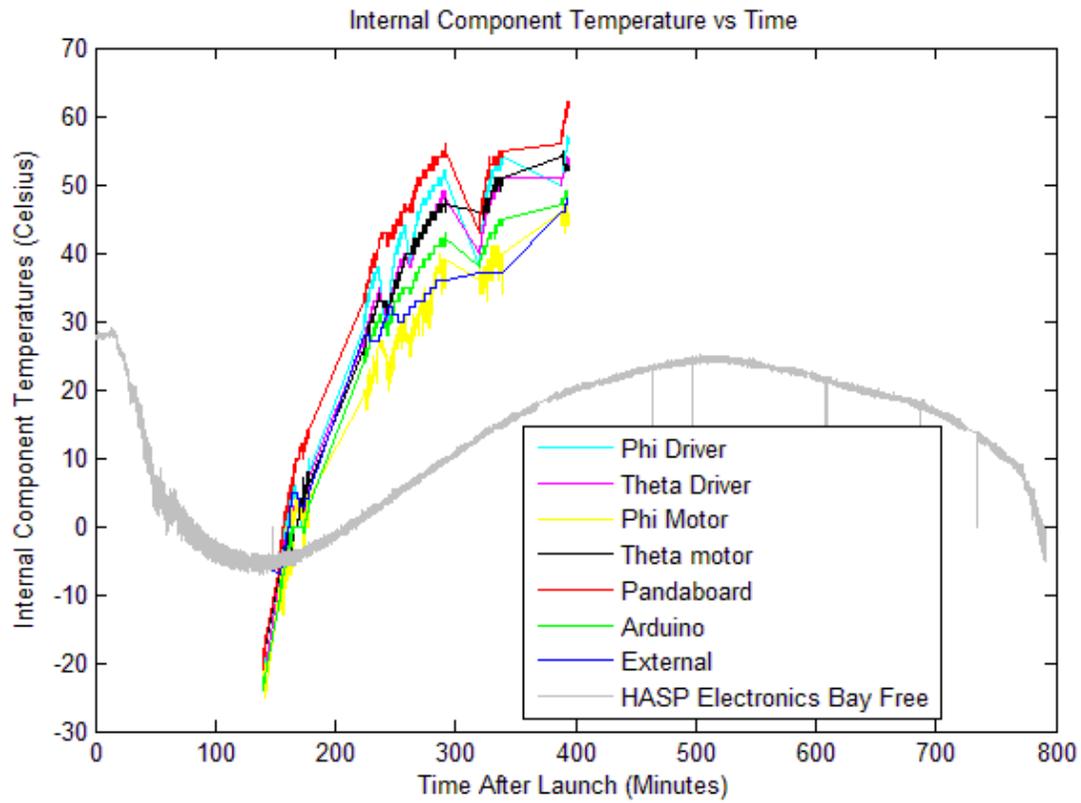


Figure 5.3.0 Temperature Profiles with HASP Data. The plot displays the temperature profiles of certain components in the payload housing (see legend, including the “Electronics Bay Free” from the HASP platform). Note the abrupt stop in recorded data from the HELIOS team.



[Type text]

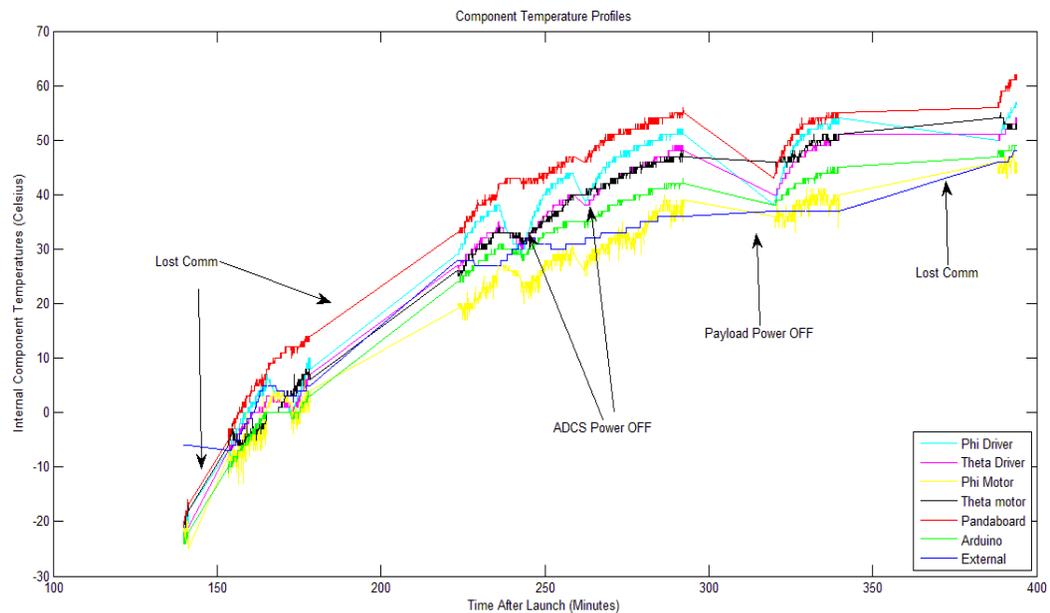


Figure 5.3.1 Temperature Profiles. *The plot displays the temperature profiles of certain components in the payload housing (see legend). Note the discontinuities in data, as well as the decrease in temperature of the motor drivers when the ADCS system was powered off.*

Fig 5.3.2 is a plot of the recorded pressure for the time that the CDH processor was still communicating. Pressure readings beyond this point are not available due to the communication loss of the CDH processor. Note that the scale is PSI to the negative 14th (effectively zero). The only thing we can really deduce from this plot is that HELIOS II was in near space. Note that the limited resolution of the pressure sensor makes the data jump from level to level, rather than creating a smooth curve. Also, the pressure appears to be less than zero at certain points during the flight. This is due to imperfect calibration of the pressure sensor, and noise in the Arduino power line. These deviations from zero pressure are negligible, however, because of the small magnitude of the variations.



[Type text]

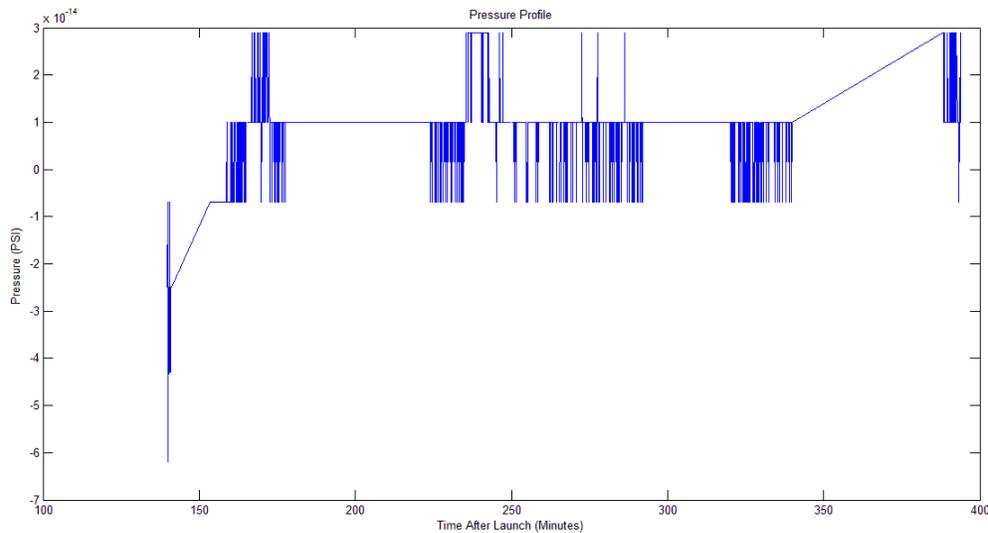


Figure 5.3.2 Flight Pressure. *The figure is a plot of the pressure during flight. Note that the scale is PSI raised to the negative fourteenth.*

The following are results that are more relevant to the performance of the EPS system. Fig. 5.3.3 is a plot of the current draw from the HASP platform as measured by the LSU HASP team, in blue, and the HELIOS II team, in red. Again, the data obtained by the HELIOS II team contains discontinuities when the CDH processor lost communication with the ground. Also, the current drops when the ADCS system was powered off. When the entire payload was powered off at mid-day the current drops to zero. The most important thing to note about this plot is that the current draw from the platform was always about 800 mA when the entire payload was being powered, meaning the requirement to maintain current draw levels below 2.5 A was met.

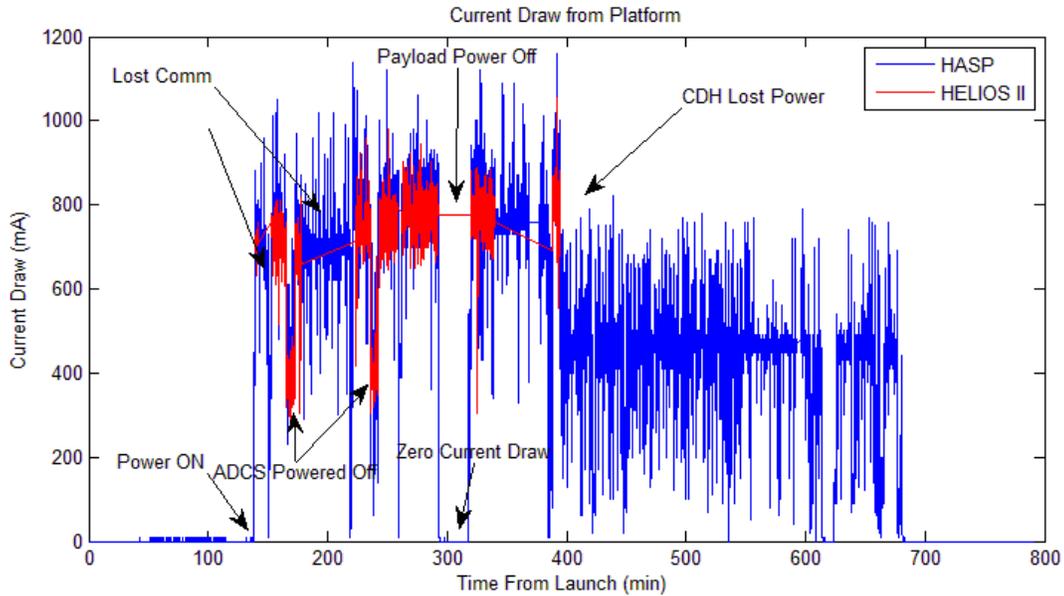


Figure 5.3.3 Current Draw From Platform. The plot displays the current draw from the platform, as measured by the LSU HASP team (blue) and the HELIOS II team (red).

The following plot, Fig. 5.3.4, is perhaps the most important in regards to the overall performance of the payload during flight; it is a plot of current draw from the CDH processor. As seen from the plot, the payload was successfully operating – and taking pictures- for four periods of time. Unfortunately, as compared to the length of the flight, this wasn't very much time. We were powered on for about 9.5 hours, but were only operational for about 3 hours. After about 6 hours after launch, the payload completely stopped communicating with the ground.

As a final result, and probably the most important, it was noticed that the CDH buck converter was severely heat damaged. The plastic that it was wrapped in was partially displaced and the paper label was completely gone. The buck converter itself was not operational. When connected to 30 V it was not providing power to the CDH voltage rail; it created an open circuit. Fig. 5.3.5 demonstrates the post flight condition of the buck converter.

1. Failure Analysis

Data analysis revealed that the *temporary* losses of communication were due to a cable failure that was detected before the flight. The cable was replaced by one of the team members, but miscommunication led to the cable being re-installed in the payload. Unfortunately, this was not obvious until after flight. The cable was unfit for data transfer and therefore created random and unpredictable errors that halted the CDH processor program. See the CDH section for more details on the matter. In addition, it was concluded that the ultimate failure of the CDH processor – the complete loss of communication after 600 minutes of flight – was due to the CDH buck converter failure. The buck converter failed due to high temperature conditions so it did not to provide power to the processor.

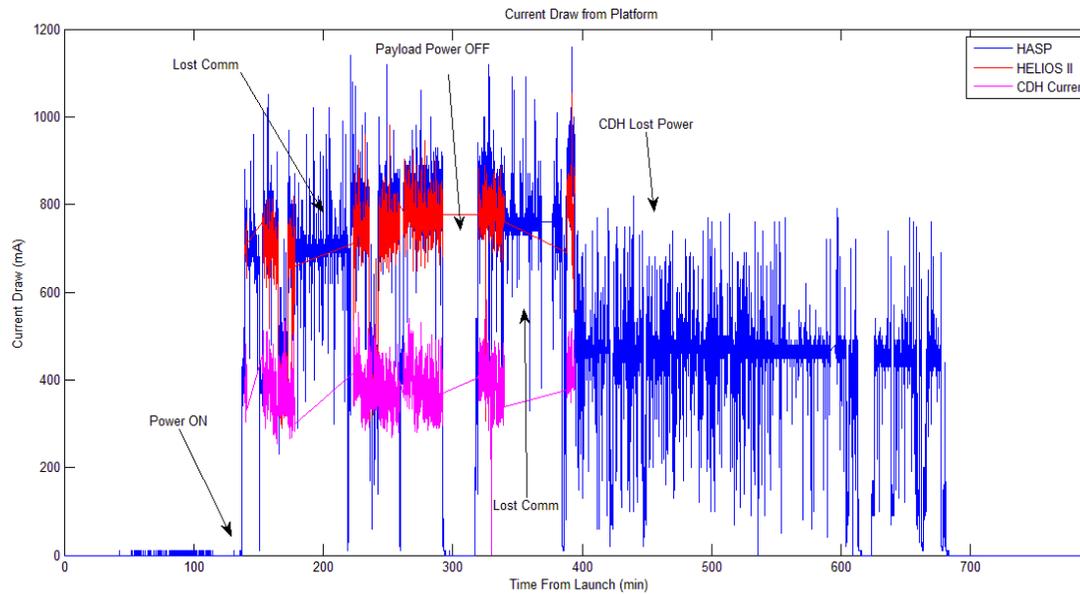


Figure 5.3.4 CDH Processor Current. The plot displays the current draw by the CDH processor (pink) as well as the current draw from the platform as measured by the LSU HASP team (blue) and the HELIOS II team (red).

This conclusion was supported by the fact that the buck converter was not operational post-flight. Further support for the conclusion is found from the temperature plots. Recall that the temperature for some of the components was at about 60 degrees Celsius when the processor stopped communicating. The maximum ambient operational temperature for the buck converter is 70 degrees C, based on the datasheet. Although there were no temperature sensors on the buck converters, it is safe to say that the entire payload housing was at about the same temperature, given that the component temperature values were very close to each other. In reality, the buck converters were most likely at a higher temperature because they were dissipating energy in the form of heat. This means that the buck converters would have been near their maximum temperature rating at the time that the CDH buck converter failed.

Another supporting detail is that, based on the datasheet, the buck converter efficiency decreases with lower voltage output. At a 9V output the efficiency is about 95%, whereas at 6 V, the efficiency drops to about 80% for the load being applied. This means that the buck converters for the Arduinos and the motors were running with a 95% efficiency, whereas the CDH buck converter was running with about 80% efficiency. Using Eqn (1) – and the information about the current and voltage for each line obtained from post-flight data – the input and output power for the buck converters was calculated. It was found that the CDH buck converter was dissipating almost twice as much heat as the other buck converters; it was dissipating 1.1 W, whereas the others were only dissipating 0.6 W. This explains why the other buck converters were not heat damaged like the CDH buck converter. They were outputting voltages that optimized the efficiency of the buck converter, but the CDH buck converter was outputting a voltage that was not quite as efficient.

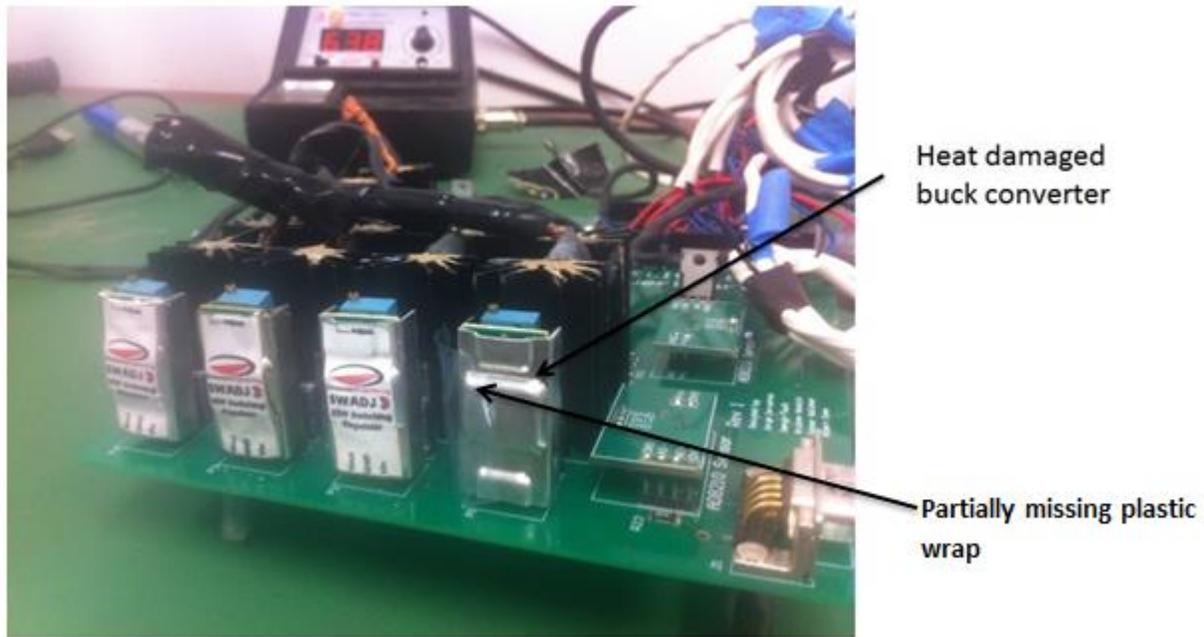


Figure 5.3.5 CDH Buck Converter. *The figure displays the condition of the CDH buck converter after flight. Note that the plastic wrap is partially gone, and the paper label is completely missing. Other non-damaged buck converters are on the left.*

2. Verification

The hypothesis was verified by replacing the damaged buck converter with one of the other buck converters, and functionality was restored to the CDH power line. Operation of the rest of the components was validated by operating the power board after flight. The board displayed correct values for all environmental data and the power rails were at the correct voltages.

C. CDH Results

During the course of flight, the CDH subsystem performed mostly as expected with the exception of few issues. CDH stored images and data on the solid state drive (SSD) as expected with minimal data corruption and downlinked data to ground during flight. There was, however, an issue with receiving and implementing commands. 5 commands were successfully verified by CDH and were registered as sent to EPS in the data file. These commands were intended for the EPS, telling the system to either power off or power on the ADCS. The commands in order of receipt were: power the ADCS off, power the ADCS off, power the ADCS on, power the ADCS off, and power the ADCS on. Although CDH confirmed that the commands had been verified and received, no indication of a response from EPS was found in the data file. Voltage readings, however, indicate that the ADCS was powered off and on twice during flight. There were instances in the data file that indicate that CDH began to process a command but deemed it invalid. It is not known if what was read from the HASP serial line was intended to be a command. While this command verification and notification did not work as expected, this process of notification was shown to work before flight. The cause of this issue has not been determined.

Based on sensor data, there were times during flight when the Pandaboard was receiving power but was neither storing nor transmitting data to ground. Issues with the Pandaboard and SSD configuration were realized post flight. The CDH team, after flight, was not able to get anything to store to the SSD when running the flight code or a basic test code. The CDH team noticed two issues with hardware as well. First, the connection between the USB cord and the housing that the solid state is mounted on is unstable. It was noticed that when the CDH team tried to attach the SSD via its housing and USB cable that it sometimes was not registered by computers running a Linux OS. Over time, its frequency of successful registering decreased. This may be due to the unstable connection.



When the SSD was removed from its housing and directly attached to a desktop computer, the SSD registered showing that the solid state drive was still operating.

The second issue noted post flight was a thin, long USB extension cable, that was used to more easily access the SSD while the Pandaboard and SSD were in the payload, had flown. Previously there were issues appearing in the flight code output regarding the cameras. After doing some research it was found that using certain cables with the cameras could cause that problem. In that particular case, once the cable switched out, the camera problem was eliminated. A similar problem may have occurred when this extension cable was not removed preflight but this was not verified whatsoever. This same configuration was not possible to test, due to the lack of output when trying to run the flight code post flight as mentioned previously. This may have affected the performance of CDH and more importantly the payload as a whole in some way but how or if was not determined.

E. Structural performance

The structure successfully supported the HELIOS II payload during flight and landing. None of the structural components appear to have been broken during flight. Part of the payload, specifically the left camera swing bracket (when viewed from the front), was disassembled following launch, presumably for packaging and shipping. Upon landing the payload struck the ground and bent the rod mounted to the circular plate. As shown in Figure 5.5.1 one can see the rod pulled the circular plate upwards, significantly bending the plate.

The thermal system did not perform. Nearing the end of the flight, the electronic components overheated, leading to a system failure. The Pandaboard overheated due to an insufficient thermal management system.

VI. Lessons Learned & Design Improvements

A. Science

The gain issue was caused by the faulty library used to control the cameras. The cameras should have been coded in a different language or operating system so that the libraries were supported and up to date. Another option would be to choose cameras with supported libraries in python. This should eliminate any apparent oversaturation of the image.

The Science Camera design was elongated significantly since the preliminary design. This caused a great deal of structural difficulty and exceeded the vertical space limitations. The design should be shortened using a better telescope configuration. The Science Camera could feature a Cassegrain configuration to shorten the total length of the tube by a factor of three.

B. ADCS

Several lessons were learned from the flight of HELIOS II. First is that the ADCS is able to locate and track the Sun using photodiodes. However, the current ADCS design is sensitive to the balloon's reflection of sunlight. Additionally, it was learned that the rotation of the balloon platform was not a major factor affecting the accuracy of the ADCS.

Based on these lessons, there are several recommended design changes. First is that photodiodes will only be used for the Macro Track Loop to find the Sun. The photodiodes can be used to locate the Sun and place the Sun within the FOV of the ADCS camera. The Second design change is that the ADCS camera will be used for Micro Track Loop. An algorithm should be written to locate the Sun in the ADCS camera images real-time. A microcontroller will then give commands to center the Sun using the information collected from the ADCS camera.



Figure 5.5.1 Bent circular plate. *The circular plate underneath the camera housing was bent during the HASP landing.*



C. EPS

There are some EPS items that the HELIOS II team would do differently if the payload was built again. Firstly, the choice of the buck converter would be different. It was found that the “Plug and Play” configuration that these buck converters provided was not robust. In addition to the problem in flight, these buck converters presented many issues prior to flight. Some would become damaged for no obvious reason and others would intermittently malfunction. Unfortunately, the problem was not apparent until later in the design phase when it was too late to redesign the power board. Overall, these buck converters were very prone to failure.

The alternative to these buck converters would be a step-down switching regulator. These provide the same high efficiencies as buck converters, but they are actually only a small chip around which the rest of the regulator is built using an inductor, a MOSFET, and other electrical components. These chips do essentially the same thing as the buck converters that were used in the final design, but they are more robust and can handle higher load currents. In addition, these chips are almost always small surface mount components. Such small components help reduce the size of the power board, especially given the large size of the current buck converters.

Finding smaller components would be another major design change to the system. Right now the power board spans about 10.6 in x 6.5 in, and this is because of the large size of the heat sinks on the linear regulators, the large size of the buck converters, and the accommodation needed for the Arduino Mega. The size of the power board could be decreased by using more surface mount components like the switching regulators mentioned above. Surface mount components can be placed on the bottom layer of the board more easily than through-hole components, and surface mount components are generally more compact than their through-hole counterparts. Surface mount components would allow the team to shrink the size of the power board in all three dimensions, making them an enormous consideration for a design improvement.

D. CDH

Two ways to improve CDH are to change how the data was stored and the format the data was stored in. Clarity and consistency within the flight output file would increase the ease and decrease the time of analyzing data post flight. It would be useful to have time kept by a RTC (real time clock) on the Pandaboard itself or request that information from the HASP platform instead of receiving that information from the EPS Arduino Mega as was done on HELIOS II. This would allow CDH to apply timestamp data to the images stored, as well as have a more accurate estimation of when HELIOS II powered on. Looking into what the payload system as a whole requires from its main controller and then choosing one based on those needs is recommended versus automatically choosing the Pandaboard again. It is recommended that a future team check cables if unexplainable errors appear in the output of code.

E. Structure

There are many things that could have been done differently when designing the structure. First, it would have been better to use a meshed gear system as opposed to the chain and sprocket system. This would allow for much more accurate positioning. During cold temperature tests it was noticed that the chain connecting the two gears contracted significantly. Although it seems that no problems arose due to this it is recommended for next year that they simply connect two gears rather than use sprockets and chain again. This action eliminates any potential contraction problems and the removal of a chain makes for an easier assembly as well.



Figure 6.5.0 Misaligned Screw Holes. *The top plate holes to connect to the side panels were made incorrectly*



[Type text]



Secondly, a better thermal system is needed. Had the Panda-board been heat-sinked, and the existing heat sinks been more efficient, the electronics may have been able to function longer. A possible solution could be heat sinking all the electronics to a piece of aluminum underneath the electronics box, not touching the bottom metal plate in order to store the heat away from the electronics.

Third, it is recommended to use thinner aluminum plates for the top and bottom of the camera structure to reduce the mass of the camera swing. While the $\frac{1}{4}$ " plates used provide a very strong structure they carry unnecessary mass. $\frac{1}{8}$ " plates on the top and bottom would not reduce structural integrity significantly but would reduce the load on the y-axis motor. Despite using a large motor on the y-axis the movement was sluggish and reducing the mass would allow for faster operations.

Fourth, the L brackets used on the bottom of the structure to connect the structure to the board should be flipped to the outside of the structure. Their current position made assembly extremely difficult due to the hard to reach locations of the nuts and bolts. Also because this would put the payload outside of the size restrictions the length and width of the structure need to be reduced accordingly.

Finally, manufacturing errors must be addressed. The Figure 6.5.0 depicts the top panel of the structure. Notice how the line drawn through the three holes does not run parallel with the edge of the panel itself, nor does the line run through all three holes. This mistake in machining meant the side panels did not fit properly, the panels bent outwards when assembled, and only one screw could be used to attach the side panel to the top plate on that side. Also the hole on the top plate which would mount the camera platform was not drilled wide enough. Placing the bearing in the hole required the use of a hammer. The design should have included a hole slightly larger than the bearing in order to actually accommodate the bearing.



VII. Demographics

Team HELIOS II consisted of many different students of various ages, academic levels, and backgrounds. Figure 7.0 contains a table representing the demographics of all students who worked on HELIOS II.

Student	Gender	Ethnicity	Race	Student Status	Disability
Caleb Lipscomb	Male	non-Hispanic	Caucasian	Undergraduate	No
Jonathan Sobol	Male	non-Hispanic	Caucasian	Undergraduate	No
Jorge Cervantes	Male	Hispanic	Hispanic	Undergraduate	No
Kristen Hanslik	Female	non-Hispanic	Caucasian	Student Status	No
Devon Connor	Male	non-Hispanic	Caucasian	Undergraduate	No
Austin Bennett	Male	non-Hispanic	Caucasian	Undergraduate	No
Ashley Zimmer	Female	non-Hispanic	Caucasian	Undergraduate	No
Jack Swanson	Male	non-Hispanic	Caucasian	Undergraduate	No
Rishav Banerjee	Male	non-Hispanic	Indian	Undergraduate	No
Star Pais	Male	non-Hispanic	Caucasian	Undergraduate	No
Gabe Frank	Male	non-Hispanic	Caucasian	Undergraduate	No
Ali Elhouderi	Male	non-Hispanic	Middle Eastern	Graduate	No
Albert Como	Male	non-Hispanic	Caucasian	Undergraduate	No
Jordan McNally	Female	non-Hispanic	Caucasian	Undergraduate	No
Jaevyn Faulk	Male	non-Hispanic	Asian	Undergraduate	No
Connor Kelleher	Male	non-Hispanic	Caucasian	Undergraduate	No
Dan Nowicki	Male	non-Hispanic	Caucasian	Undergraduate	No
Daniel DeWolf	Male	non-Hispanic	Caucasian	Undergraduate	No
Anthony Lima	Male	Non-Hispanic	Caucasian	Undergraduate	No

Figure 7.0 Demographics of HELIOS II

VIII. Conclusions

An image of the Sun was captured in the Hydrogen Alpha wavelength. No solar features were identified in the image; however the Science Camera would have been able to resolve sunspots or other large solar features if they had been present in the partial Science Camera image. The ADCS system was able to locate and track the Sun throughout the flight. The ADCS was able to track the Sun to within 4 +/- 3 degrees in the Y axis and 2 +/- 1 degrees in the X axis. However, the designed ADCS system was not accurate enough for HELIOS II to capture a large number of images in the Science Camera. Finally HELIOS II was able to capture many low resolution images of the sun and one partial high resolution image of the sun. All observation issues were a result of problems with the HELIOS II payload, and not with the HASP platform. If the technical issues HELIOS II faced had been resolved, HELIOS II would have been able to capture several high resolution images of the sun. This shows that high altitude balloons are a viable solar observation platform.



[Type text]



IX. Acknowledgements

Team HELIOS II would like to thank everyone who supported the project. Special thanks to:

- Dr. James Green
- Dr. Greg Guzik
- Doug Granger
- Mike Stewart
- Chris Koehler
- Mark Eaton
- Team HELIOS I

X. References

¹⁰ "Observing the Sun in H-Alpha." *The Prairie Astronomy Club*. The Prairie Astronomy Club, n.d. Web. 11 Dec. 2013. <<http://www.prairieastronomyclub.org/resources/solar-observing/observing-the-sun-in-h-alpha/>>.

¹¹ Ibid. 2013.

¹² *Cool cosmos*. (n.d.). Retrieved from

http://coolcosmos.ipac.caltech.edu/cosmic_classroom/multiwavelength_astronomy/multiwavelength_museum/sun.html

¹³ Eisenhamer, J. (2012, March 02). *Hubblesite*. Retrieved from

http://hubblesite.org/the_telescope/hubble_essentials/