University of Colorado
Department of Aerospace Engineering Sciences
ASEN 4018

Conceptual Design Document
Optical Thermal Regulation System
(OTheRS)

Monday 1st October, 2018

1. Information

Approvals

Role	Name	Affiliation	Approved	Date
Customer	Zachary Koch/Matthew Yavorsky	General Atomics		
Course Coordinator	Jelliffe Jackson	CU/AES		

Project Customers

Name: Zachary Koch Email: Zachary.Koch@ga.com Phone: 720-255-0903 (work)	Name: Matthew Yavorsky Email: Matthew.Yavorsky@ga.com

Team Members

Name: Justin Alvey Email: jual6332@colorado.edu Phone: 313-310-2704	Name: Ryan Bennett Email: rybe0897@colorado.edu Phone: 303-885-0339
Name: Emma Cooper Email: emma.cooper@colorado.edu Phone: 720-833-8782	Name: Sean Ellingson Email: sean.ellingson@colorado.edu Phone: 720-301-8107
Name: Pierre Guillaud Email: pigu4637@colorado.edu Phone: 310-425-9039	Name: Nash Jekot Email: naje0055@colorado.edu Phone: 720-347-9915
Name: Jacob Killelea Email: jaki3198@colorado.edu Phone: 650-305-9455	Name: Micah Svenson Email: misv9521@colorado.edu Phone: 970-409-9333
Name: Kendall Worden Email: kewo3780@colorado.edu Phone: 972-310-8124	

Contents

2. Project Description

2.1. Purpose

Thermal regulation is important for successful operation of any system and is even more important in space systems, which are subjected to extreme and widely ranging thermal environments. As electronics have taken over almost every aspect of modern spacecrafts, they have also increased the risk of mission failure due to their high thermal sensitivity. Current thermal regulation systems although simple, have introduced complexity in wiring, harnessing, and bus inputs as reliance on electronics increases.

The OTheRS project seeks to provide proof of concept for an innovative non-contact thermal regulation solution using IR cameras. This solution would reduce bus inputs, wiring and harnessing complexity, and provide greater thermal detail of the avionics electronics stack being imaged. This system will provide thermal profile data through a serial output as well as on/off commands to heaters or other representative components in the stack.

In practice this project will determine the feasibility of such a system by constructing a test bed, which will model the dimensions and materials used in the satellite and produce a random thermal profile to be imaged and analyzed. The test bed will also be fitted with traditional thermal sensors in order to establish a baseline for data comparison.

2.2. Objectives

The OTheRS must be able to map and accurately sense the temperature of a mock electronics stack within the test bed and provide spatially correlated temperature data through a serial output. Additionally, a controller must receive temperature data then process and switch on/off commands to heaters or LED's to simulate temperature regulation of the stack. The data from the OTheRS will then be compared to traditional thermistors, which will also be located on the stack, as our reference data and means of validation.

2.3. CONOPS

The test bed used to validate the design of the OTheRS will be fabricated to resemble the avionics bay of a General Atomics (GA) small satellite with the cameras being mounted on the inner floors or ceiling. The cameras will image the outside of the stack, which is made up of tightly stacked aluminum trays, and process the images to produce a thermal profile of the stack. The image processing will include thermal and geometric calibration of the cameras, spatial mapping, and proper temperature scaling. The thermal data derived from image processing will then be sent to the satellite bus via serial output. In the test bed case, the data will be output to a computer for data logging. Additionally, the system will use the thermal data to process and send on/off control signals to a series of LED's, which will represent a heating mechanism or component off-command to regulate the stack temperature. Finally, this data will be validated by comparing to thermistor measurements from the stack.

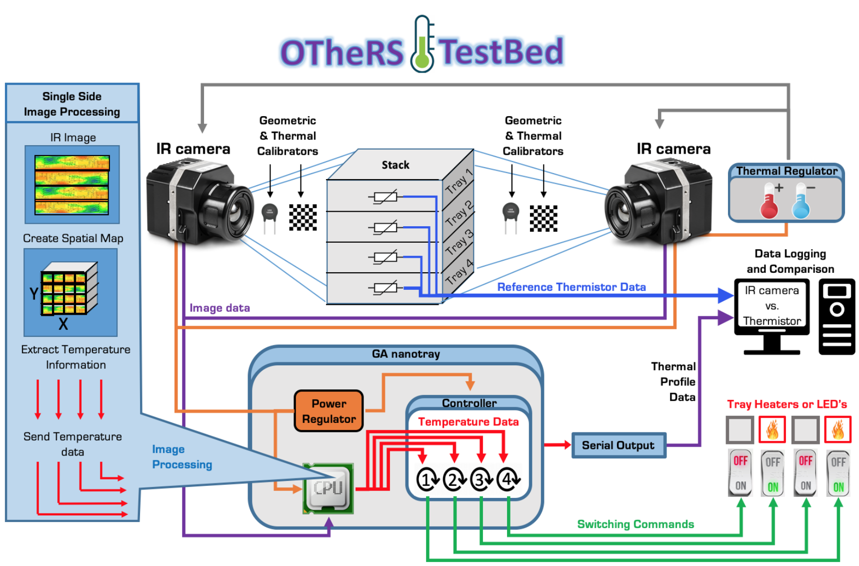


Figure 1. OTheRS TestBed CONOPS

2.4. FBD

The satellite bus, electronics stack, and heaters are not project elements, but will supply the system with an unregulated 28V power source and receive a serial output from the system. These elements may however be replaced by computers, power supplies, fabricated mock ups, etc. in the test bed setup.

The overall system itself contains five elements: a power management system, one or multiple thermal cameras, a processing unit, a calibration unit, and a thermal management system. The power management system is responsible for regulating and transforming the 28V volts into a desired voltage and amperage required by the other components. The IR camera(s) receive a thermal image of the avionics stack and outputs the image data to the processing unit which will process the images and produce temperature data for serial output to the satellite bus. A thermal management system is required to keep the processing unit and the thermal camera(s) at operating conditions. The calibration unit will be used to calibrate the temperature and spatial measurements of the thermal camera by measuring the temperature at one or more points in the camera's Field Of View (FOV) in order to accurately identify spatial and thermal information in the image data.

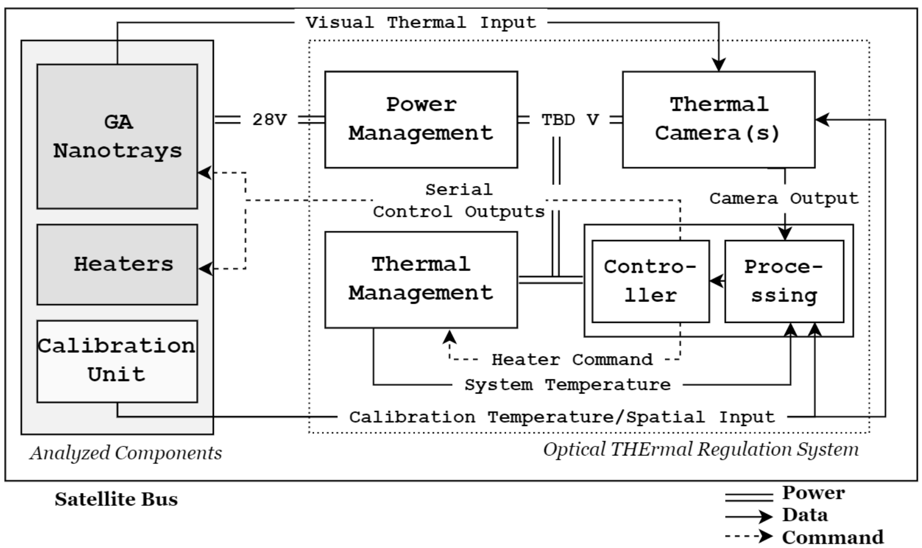


Figure 2. Major functional elements of the system

2.5. Functional Requirements

FR 1 System shall return thermal data map for multiple components between -30°C and 60°C with 0.1°C temperature accuracy.
Motivation: This requirement is given by customer as primary design driver for the developed system. The thermal map allows thermal control of components that are viewed by the system.
Validation & Verification: Test - Thermal ambient testing will verify if the system can sense this temperature range and resolution. Also, verification of DR-CAM 1 can be used.

FR 2 System shall provide regulatory commands when components are outside -20°C to 50°C.
Motivation: This requirement is given by customer as primary design driver for the developed system. Regulatory commands outside of this range will allow for possible control of stack electronics.
Validation & Verification: Test - During thermal ambient test, successful verification of DR-CONT 1 and 2.

FR 3 System shall operate on 28V unregulated power provided by the spacecraft.
Motivation: The satellite provides power at this voltage. Therefore this is a customer defined requirement that allows integration of this system to a mock satellite.
Validation & Verification: Test - Verify system is operational when power is provided by a power supply at the given voltage.

FR 4 Supporting systems electronics shall fit within a standard GA nanotray with TBD dimensions.
Motivation: Customer derived requirement to allow for system integration into GA models and tests.
Validation & Verification: Demonstration - The final system can be measured to show the size is within the required dimensions.

FR 5 System shall be able to switch a 2.5A load as needed to control an externally powered heater or representative indicator.
Motivation: Customer derived as the heaters used in the satellite operate of this specification.
Validation & Verification: Demonstration/Test - An ammeter can be used to verify the output signal meets this requirement.

FR 6 The thermal camera(s) shall be placed such that it can image the stack.
Motivation: Customer derived as the purpose of this project is to monitor the temperature of the electronics stack in the satellite.
Validation & Verification: Inspection - An image of the stack can be inspected to verify that this requirement is satisfied. Also, verification of DR-CAM 3.1 can validate this requirement.

FR 7 System shall regulate its own temperature.
Motivation: Thermal regulation of the OTheRS will allow for components and subsystems to function properly in a harsh thermal environment.
Validation & Verification: Test - Testing in a thermal ambient chamber will show if the thermal regulation is successful. Validation of this requirement can also use a thermal model and successful verification of DR-THER 2 and 2.1.

FR 8 Test bed shall mimic the GA satellite.
Motivation: This requirement is derived from customer requirements to show feasibility of the system to operate in a mock satellite.
Validation & Verification: Inspection - The test bed can be inspected and compared to the provided CAD file to verify dimensions and other key specifications. Design requirements in the Test Bed subsystem can be successfully verified to validate completion of this requirement.

FR 9 System shall communicate with TBD communication protocol.
Motivation: The OTheRS is desired to communicate with the satellite in a serial manner so a communication protocol is needed to define how the information is transferred.
Validation & Verification: Demonstration - View the output of the system through a logic analyzer to verify the correct data.

FR 10 OTheRS shall not include ITAR or EAR restricted articles.
Motivation: The project is published in public domain so the University requires this.
Validation & Verification: Inspection - Checking all parts used in the OTheRS prior to purchasing or installation.

3. Design Requirements

3.1. Camera

DR-CAM 1 The camera subsystem shall be sensitive to IR radiation, measuring between 9 and 12 microns in wavelength.
Motivation: The wavelengths corresponding to thermal radiation in the monitored thermal range.
Validation & Verification: Demonstration - Imaging of a object with a known temperature can verify the sensitivity to IR radiation in this range.
Related Requirements: FR 1

DR-CAM 2 Images of the stack shall be taken at TBD intervals.
Motivation: Imaging the stack at intervals can decrease the data processed while still viewing the changing temperature of the stack.
Validation & Verification: Demonstration - Checking the number of images taken in a defined time period will give the frequency of data capture.
Related Requirements: FR 1

DR-CAM 3 The camera subsystem shall be placed between 100mm and 140mm from the stack.
Motivation: This is a spatial constraint from the satellite design.
Validation & Verification: Inspection - Measuring the placement of the camera in the test bed.
Related Requirements: FR 6

DR-CAM 3.1 The TBD field of view (FOV) of the camera shall contain the stack.
Motivation: The stack must be fully visible at the distance in DR CAM 3. The FOV is a characteristic of the lens on the camera which will allow full visibility of the stack.
Validation & Verification: Demonstration - An image of an object at a given distance will verify if the FOV will contain the stack.
Related Requirements: FR 6

3.2. Processing

DR-PROC 1 The thermal map shall differentiate between objects in stack based on satellite and test bed configurations.
Motivation: This is customer derived, as being able to differentiate between objects in the stack will allow for OTheRS to control the temperature of these objects.
Validation & Verification: Successful verification of DR-PROC 1.1.
Related Requirements: FR 1

DR-PROC 1.1 The thermal map shall distinguish between TBD number of objects in the stack.
Motivation: OTheRS will need to tell how many objects are in the stack to correctly identify the object.
Validation & Verification: Test/Inspection - Output thermal map will be analyzed to verify the map identifies the tray regions.
Related Requirements: FR 1

DR-PROC 2 The camera subsystem shall be both spatially and thermally calibrated.
Motivation: The calibration of the system is instrumental in the image processing as without calibration, OTheRS will not be able to identify what it is looking at correctly.
Validation & Verification: Test - Thermal ambient testing with control surfaces or devices will show if the requirement is satisfied.
Related Requirements: FR 1

DR-PROC 3 Processing subsystems shall control internal thermal regulation of the OTheRS.
Motivation: The internal thermal environment must be controlled so the components can operate as they are designed.
Validation & Verification: Demonstration - This can be shown in the thermal ambient testing along with a thermal model of the OTheRS.
Related Requirements: FR 7

DR-PROC 4 Temperature data shall be extracted from thermal map.
Motivation: Thermal data being extracted from the map is a primary function of the system and needed for level 1 success.
Validation & Verification: Inspection - The output from the processing subsystem will be checked against actual location thermal data provided by a sensor.
Related Requirements: FR 1, DR-PROC 1

DR-PROC 4.1 Temperature data shall be extracted in TBD time before next image is taken of stack.
Motivation: The processing needs to extract data prior to another image being given to the subsystem to prevent a backlog of data occurring in a queue which will prevent accurate monitoring of the stack.
Validation & Verification: Demonstration - Checking the output time of the OTheRS and direct comparison to the frequency of image capture.
Related Requirements: DR-CAM 2

3.3. Power Management

DR-POW 1 The power management subsystem shall provide power distribution to the subsystems within OTheRS.
Motivation: Power will need to be allocated or given to subsystem to allow for operation of all subsystems.
Validation & Verification: Inspection - Successful verification of the power outputs of the power subsystem, through ammeter or voltmeter readings, will verify if the output to all subsystems is correct and satisfies the required power of all subsystems.
Related Requirements: FR 3

DR-POW 2 Possible electrical connections on GA nanotray shall be used to transfer power from the spacecraft bus to the OTheRS, if any exist.
Motivation: If the GA nanotray has electrical connections built into the tray, OTheRS will use the connections to ease future system integration.
Validation & Verification: Inspection/Demonstration - Verify power connections or data transfer is successfully with oscilloscope or logic analyzer.
Related Requirements: FR 4

3.4. Control

DR-CONT 1 A control decision shall be communicated by the OTheRS to turn a heater on or a representative indicator.
Motivation: This is customer defined and critical to the thermal control of the stack.
Validation & Verification: Demonstration/Test - In the thermal ambient testing, an actual or representative system, can be used to verify OTheRS is sending the correct command for the tray.
Related Requirements: FR 2

DR-CONT 2 A control decision shall be communicated by the OTheRS to turn a component off or a representative indicator.
Motivation: This is customer defined and critical to the thermal control of the stack.
Validation & Verification: Demonstration/Test - In the thermal ambient testing, an actual or representative system, can be used to verify OTheRS is sending the correct command for the tray.
Related Requirements: FR 2

3.5. Internal Thermal Regulation

DR-THER 1 The internal temperature of the OTheRS shall be monitored by a thermistor.
Motivation: The internal temperature of OTheRS will need to be monitored to ensure safe operating conditions for all subsystems.
Validation & Verification: Demonstration - The thermal component can be verified at room temperature to show the component is reading the correct temperature.
Related Requirements: FR 7

DR-THER 2 The minimum operating temperature of the OTheRS shall determined by highest minimum operating temperature of all subsystems or components.
Motivation: The highest minimum operating temperature of all components will be used to define the minimum operating temperature of the OTheRS to ensure full operation of the system.
Validation & Verification: Test - Successful verification of DR-THER 1 and 2.1 along with a thermal ambient test will show this requirement is met.
Related Requirements: FR 7

DR-THER 2.1 The minimum operating temperature shall include 10°C Factor of Safety.
Motivation: A factor of safety will need to be used as a safeguard against possible sensor failure or errors that read the internal temperature of OTheRS.
Validation & Verification: Test - Successful verification of DR-THER 1 and a thermal ambient test will show this requirement is met.
Related Requirements: FR 7

3.6. Communication

DR-COM 1 The OTheRS shall use serial communication to transfer information, signals, etc. with the spacecraft bus.
Motivation: Serial communication is desired by the customer, thus making it a requirement for any communication in or out of OTheRS.
Validation & Verification: Demonstration/Inspection - Verify all outputs of the system are serial using a personal computer.
Related Requirements: FR 5

DR-COM 2 OTheRS shall communicate with TBD heater communication protocol (if any).
Motivation: Heater specifications may be provided by the customer so output must conform to this heater.
Validation & Verification: Demonstration/Inspection - Checking the output designated for the heater will verify if this requirement has been satisfied.
Related Requirements: FR 5

3.7. Test Bed

DR-TEST 1 The outer dimensions of the test bed shall be 183.5mm x 515mm x 547mm.
Motivation: The test bed is being used to simulate the actual satellite so the dimension should match the system for a good replication of actual integration.
Validation & Verification: Inspection - Verify measurements taken on the completed test bed for direct comparison to the CAD model.
Related Requirements: FR 8

DR-TEST 1.1 The outer dimensions of the stack in the test bed shall 183.5mm x 290mm x 318mm.
Motivation: Similar to DR-TEST 1, the interior stack will need to be representative of the stack in the satellite which is the primary object being tested.
Validation & Verification: Inspection - Verify measurements taken on the completed test bed for direct comparison to the CAD model.
Related Requirements: FR 8

DR-TEST 1.1.1 The stack surface material shall be 5mm thick.
Motivation: The thickness of the surface will determine the thermodynamics of the heat transfer through the wall thus a value representative of the actual satellite will be used.
Validation & Verification: Inspection - This value can be directly measured the verify the requirement is satisfied.
Related Requirements: FR 8

Spec-TEST 1.2 The stack surface material shall have between 0.1 - 0.4 absorptivity coefficient.
Motivation: The thermal properties of the stack will need to replicate the true environment of the satellite so the system can be modelled correctly. The material value will be used to eliminate testing that is outside of the scope.
Related Requirements: FR 8

Spec-TEST 1.3 The stack surface material shall have 0.05 - 0.1 emissivity coefficient.
Motivation: The thermal properties of the stack will need to replicate the true environment of the satellite so the system can be modelled correctly. The material value will be used to eliminate testing that is outside of the scope.
Related Requirements: FR 8

4. Key Design Options

In order to ensure the design requirements are met the OTheRS can be broken down into six different subcategories. These subcategories include the Camera, Software, Electronics, Control System, Thermal Control of OTheRS, Thermal Model, and Test Bed. Camera design options include not only the camera model, but also the setup and number of cameras that will be used. The software category only corresponds to the image processing done by OTheRS used to extract temperatures from the images taken by the camera subsystem. The electronics subsystem refers to how power will be supplied to all different parts of OTheRS including camera(s), the microprocessor, and the microcontroller. The thermal model will be used to analyze the temperatures produced by OTheRS to ensure that it stays in the bounds given by the functional requirements. It will also predict the temperatures produced by scatter off of the material covering the stack. The testbed section refers to all different designs that the OTheRS could be used with in order to ensure the system can accurately sense temperatures of a fabricated stack as well as output serial commands. The control system refers to how OTheRS will output commands to turn satellite heaters on or off depending on the realized temperature of the stack. Finally, the Thermal Control system refers to how OTheRS will thermally regulate itself in order to keep it within operating temperatures. This includes the heating or cooling of the camera(s) and other subsystems in order to make sure they stay withing their operational temperature bounds. In order to approach the design problem of OTheRS, studies were performed on each of the subcategories listed above. These design studies produced different viable design alternatives for each category, which interface with all options and fulfill the design requirements.

4.1. Camera Type

As per the requirement given by the customer, OTheRS will be using an infrared camera to monitor the electronics stack. This type of camera is critical to the design as it will be reading the infrared energy emitted from the stack and converting that into temperature data. The IR camera uses infrared thermography, the study of infrared energy as it is emitted from an object. This energy is then converted into temperature, and then displays this temperature data in an infrared image. Basic signal processing in an IR camera is outlined in the figure below.

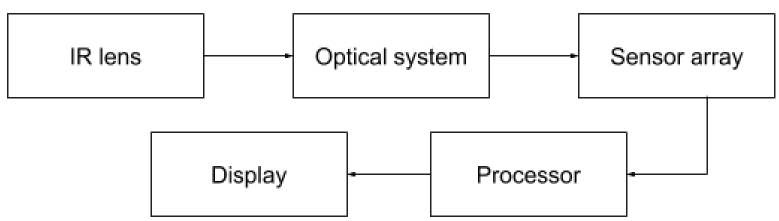


Figure 3. Flow diagram of general infrared camera

Beyond the lens of the IR camera, there is an optical system that focuses on a sensor array, comprised of hundreds of pixels. The camera sensor measures the emitted photons hitting each pixel. This information is then converted into an electrical signal. All of the signals are then processed and an infrared image is created from the results. Certain restrictions are going to limit the camera selection available to us. The biggest restriction is that the thermal camera can't be under International Traffic in Arms Regulations (ITAR). Another restriction is that the camera needs to be longwave infrared (LWIR), not shortwave infrared (SWIR) nor midwave infrared (MWIR). This is because the LWIR is the only type that will be able to read the wavelengths involved in this project (8.70 to 11.93 microns).

4.1.1. FLIR Lepton

One option for thermal cameras is the FLIR Lepton. This camera is not under ITAR regulations. This camera comes with a breakout board which is beneficial because the breakout board allows for the Lepton to be connected to a single-board computing platform. The operating temperature for this camera is -10°C to +80°C. The temperature sensing range is -10°C to +120°C and the thermal sensitivity is 0.05°C.

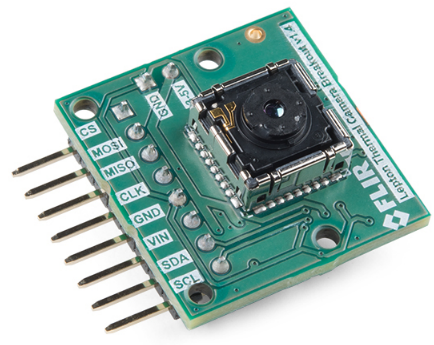


Figure 4. FLIR Lepton thermal camera on breakout board

Pros	Cons
Non ITAR	One camera might not be able to image one complete side of the stack due to FOV
LWIR	Smaller thermal sensing range
Comes with a breakout board (to connect other necessary platforms)	
A lot of helpful information in the datasheet	

4.1.2. Tamarisk Precision

Another option for thermal cameras is the Tamarisk Precision. Although it has a wider operating temperature range and thermal sensing range, no export information was provided, so it is unknown if this camera is under ITAR or not. The temperature sensing range is -40°C to +80°C and the thermal sensitivity is 0.05°C. The operating temperature range is -20° to +80°C. The price of this infrared camera is unknown.

Figure 5. Tamarisk Precision thermal camera

Pros	Cons
LWIR	No information on export control
Wide thermal sensing range	One camera might not be able to image one complete side of the stack due to FOV
	Large in size
	Doesn't come with a breakout board

4.1.3. MLX90640

A third option for thermal imaging is using the MLX90640 FIR Sensor. The sensor is composed of an array of 32x24 of thermopiles. Outputs of all sensors are stored in internal RAM. The FOV for this configuration is 110°x75°. The operating temperature range is -40°C to +85°C. The thermal range is -40°C to +300°C, however the thermal sensitivity is 1.5°C, which is weaker than that of a thermistor (0.1°C). Each individual sensor costs $48.87, however if they are bought as a pack of 10, the cost of each reduces to $45.70.

Figure 6. MLX90640 FIR Sensor

Pros	Cons
Non ITAR	Larger FOV
Cheap ($50)	Lower thermal sensitivity
Integrates well with a breakout board	Low resolution

4.2. Sensor Layout

4.2.1. Multiple Cameras

The layout of the cameras is also very important to ensure OTheRS can view the entire avionics stack. In order to see how many cameras are needed to view each side of the stack, the dimensions of the stack were found as well as the distance between the walls and the stack. The dimensions are given below.

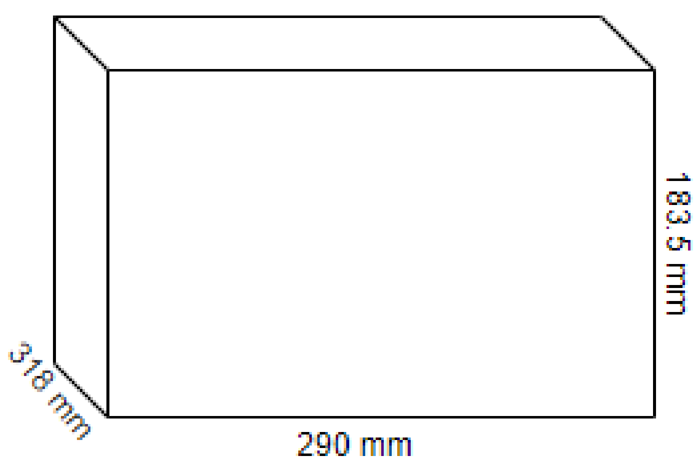
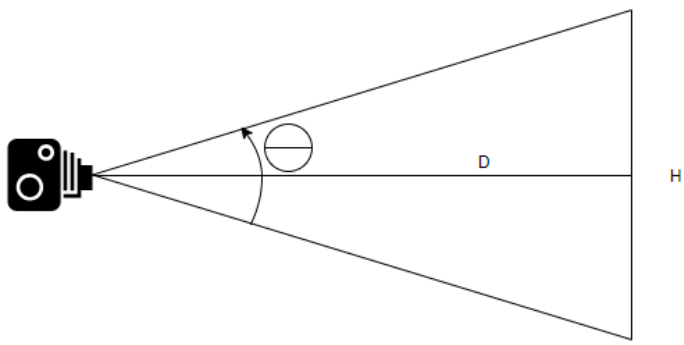


Figure 7. Dimensions of the Avionics Stack

The distance between the stack and the walls where the cameras will be located is 102 or 105 mm. In order to find out how many cameras are needed to capture the entire height of a single side of the stack the following equation was used.

$$tan\frac{\theta_v}{2} = \frac{\frac{H}{2}}{D} \tag{1}$$

This equation is based off of the following diagram.



In this equation θ_v refers to the vertical field of view of the camera, H refers to the height of the stack that will be captured, and D refers to the distance between the camera and the stack. A similar equation was also used to find out how many cameras are needed to capture the width of a single side of the stack.

$$tan\frac{\theta_h}{2} = \frac{\frac{W}{2}}{D} \tag{2}$$

Typical values for the Horizontal and Vertical field of views are $\theta_v = 37°$ and $\theta_h = 51°$. After inserting typical values for the Horizontal and Vertical fields of view of the cameras traded on it was found that 3 cameras would be needed to cover the entire height of the stack, 3 cameras would be needed to cover the width of 290 mm, and 4 cameras would be needed to cover the entire width of 318mm. This means in order to cover the entire stack 30 cameras would be needed. In order to stay within the budget of the project this design decision would be used to only measure one side of the stack in order to have a proof of concept. Another way to to use the multiple cameras is to mount them in the corners of the satellite. This increases the distance from the camera to the stack to 146.73 mm. Using the Law of Cosines it was found that only four cameras would be needed to view an entire side of the stack. Which means the entire stack could be covered by 16 FLIR Lepton Cameras. However, at $300 per camera this would contain our entire budget.

Pros	Cons
Non-complex	Expensive
Gives Proof of concept	Can only cover a single side of the Stack while staying under budget.
	Will consume a lot of power

4.2.2. Optical Manipulation

Another design option for the camera layout is to add lenses to the cameras in order to increase their FOV's. In order to find the FOV's needed for a single camera to capture a single side of the stack the following equations were used

$$\theta_v = 2tan^{-1}\frac{\frac{H}{2}}{D} \tag{3}$$

$$\theta_h = 2tan^{-1}\frac{\frac{W}{2}}{D} \tag{4}$$

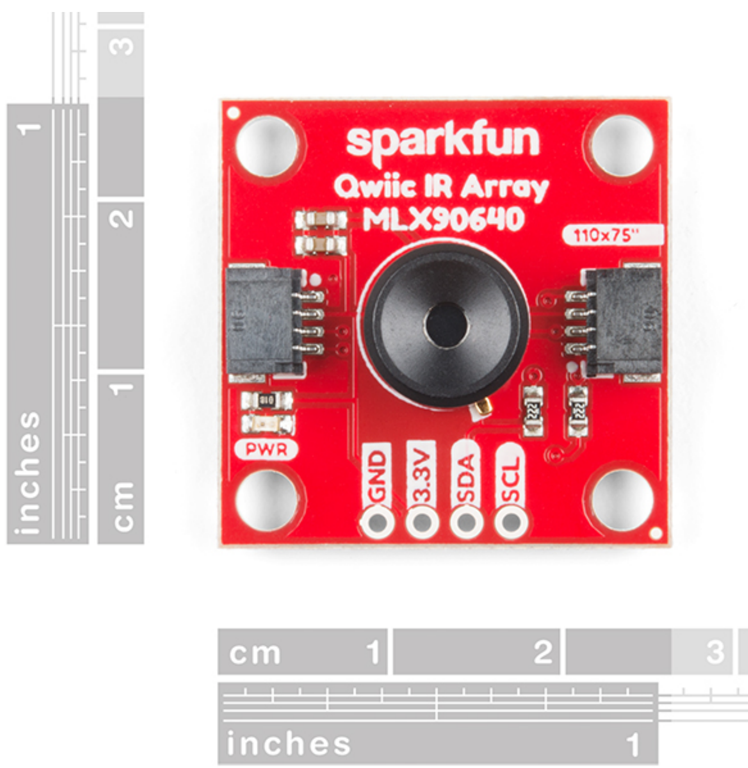
Where H and W are the the Height and Width values of the satellite. Using a value of 183.5 mm for the height of the stack and 318 for the value of the width of the stack the field of views were found as $\theta_v = 90°$ and $\theta_h = 120°$. This would cause the project to only need fewer cameras for each side of the stack which would allow OTheRS to view the entire stack during testing. However, different complications can come from adding in the lenses. To add the lenses to the micro cameras used in OTheRS a mounting platform would have to be machined for each camera meaning the system wont be made of only COTS parts. Adding lenses will also increase system complexity since the calibration system will have to take the lenses into account when determining temperatures of the stack. In order to ensure this will be a viable solution to the problem much more research will need to be performed on not only how lenses will interface with our chosen camera but also how they will affect temperature readings.

Pros	**Cons**
Inexpensive	Non-COTS Parts will need to be used
Can cover the entire stack	Increased system complexity
Minimizes mass and size of camera system	

4.2.3. Sensor Array

The third sensor option is an MLX90640 IR thermopile array. This sensor has low resolution and accuracy, but has a FOV of 110°x 75°Which would mean only two sensors would be needed to cover an entire side of the avionics stack. This sensor also has a temperature measurement and operational temperature range which fits the requirements given by GA. The sensor has a price of only \$60 per sensor as well and would only require 0.5184W from the electronics system to view the entire avionics stack. The area of each sensor array is also only $676mm^2$. A major issue with this sensor is its low resolution which is only about half of the resolution given by the FLIR Lepton and only has an accuracy of 1.5C. This sensor array also requires a micro controller with 20000 bytes or more of RAM to turn the raw pixel data into temperature data. Therefore modifications on the sensor array would have to be made to increase accuracy and usability in our system. Setting two of these sensor arrays up to view each side of the avionics stack would be extremely cheap but would also cause issues in accuracy.

Pros	**Cons**
low cost	Low accuracy and resolution
Only takes two sensor arrays to cover a wall of the stack	requires 20000 bytes of RAM
Small size and power requirements	



4.3. Camera Calibration

In order to ensure that the camera measurements are accurate, the camera must be calibrated. This is accomplished by contrasting the camera surroundings to a known point or points. How the system uses the data point(s) is the calibration methodology. The calibration can only be as accurate as the given data point(s). Therefore, it is also important to consider how these data point(s) are measured in the calibration system. Below gives an overview of some calibration methodologies as well as some different point determination options.

4.3.1. Calibration methodologies

One-Point calibration: The calibration method knows the SI value of a single location on the camera (in the case of OTheRS, this is temperature). The point is used to linearly fit a gradient of expected ideal responses near the single point. This calibration method is particularly powerful because only a single reference point is needed to calibrate the system. This type of system is generally implemented only for systems that are known to vary with a well known linear value, such as an aging thermocouple. The one point calibration can also be used as a drift check to observe when constant values are changing with time. Below lists a simple pro-con list for the one point calibration.

Pros	Cons
Simplicity (only one point is needed)	Low range
Accurate calibration for expected linear data	Only accurate for linearly changing systems

Two-Point Calibration: This calibration method uses the value of two known temperature points to linearly fit a gradient of expected values near the points. This method differs from the one point calibration because the slope is now calculated instead of specified. This is critical because it allows for the correction of both offset and and slope discrepancies. Much like the one point calibration, this method is generally restricted to linearly changing systems. Below lists a simple pro-con list of the two point calibration method.

Pros	Cons
Larger range	Added complexity with more than one data point
Can correct for slope and offset errors	Only accurate for linearly changing systems

Multi-point Curve Fitting Calibration This calibration method uses the value of multiple known temperature points to non-linearly fit a gradient of expected values near the points. This method differs from the other two in that the multi-point calibration will generate a non-linear fit to the points. This is powerful because it allows for the calibration of more complex systems that alter non-linearly. Below is a simple pro-con list of the multi-point calibration.

Pros	Cons
Non-linear systems are an option	Great complexity because of the multitude of points
Less dependency on accuracy of each single point	Tends to deviate at the extremes

4.3.2. Point determination

All three of the above methods have a strong dependency on the accuracy of the measured point(s). This section will consider the options of either utilizing a black body radiator or thermistor to get the needed data point(s) for calibration.

Black Body Radiator: The black body radiator as a point determiner essentially heats up an assumed black body to a specific temperature (typically a large temperature) by supplying an electric current for a specific time that is known to heat the object up to a known temperature. The calibration methodologies use this known temperature as an assumed truth value to calibrate the system. A simple pro-con of this point determination method is given below.

Pros	Cons
Very accurate temperature readings	Requires a large power consumption to heat the object to a known value
Useful for calibrating large ranges of temperature differences	Model dependent

Thermistor: In the method of point determination, a thermistor is connected to a known location in the FOV of the camera and feeds the calibration system the temperature of that point. The calibration method then assumes that the received value is accurate for the location and calibrates accordingly. This is beneficial because the thermistor requires a smaller amount of energy to measure a temperature rather than to heat up an object to a specific temperature. Below gives a simple pro-con list of the thermistor point determination method.

Pros	Cons
Power inexpensive	Calibration can only be as accurate as the thermistor
Relatively cheap	Calibration is system dependent

Although there appear to be many options above, many are not sufficient as an option for OTheRs. The selected calibration methodology and point determination option will be discussed below in baseline design selection section.

4.4. Image Processing Software

The image capturing hardware used for mapping the nanotrays (camera) will require extensive adjacent software components that will locate these objects within the image and extract temperatures to determine possible overheating in the GA electronics bay. When given an image of input, the on-board image processing software will be able to image and map the nanostacks. Once this is completed, extract temperature metadata from the image at verified spatial locations. A temperature range will be determined of the electronics components that conduct heat to the nanotray and plotted vs spacebay. For validation and proof of concept, the image processing software will be calibrated both on its ability to accurately map the stack relative to the thermal camera and based on the temperature data extracted from the image. This will be done based on actual measured locations of electronics bay nanostacks and additional software will need to be implemented to perform a black-body object temperature test to calibrate the temperature calculations of the image processing software. The specific software technologies being considered to perform the on-board image processing and determine temperature data in this trade-off are OpenCV in C++, OpenCV in Python, and Matlab.

SimpleCV in Python and scikit-learn in Python were also considered during research for solving this image processing problem but did not make the final cut. SimpleCV is an older method for image processing, portable to Python, that has been out of community use in scientific applications for some time. This library is Python-only but has weak documentation with little support compared to OpenCV which is currently an industry standard and has a thriving community of support. Scikit-learn is purely a machine learning platform that lacks the required relevant image processing functions and machine learning libraries.

4.4.1. OpenCV in C/C++

OpenCV is an extensive library for computer vision and machine learning applications. These tools, when utilized correctly, make image processing much simpler than in years past. OpenCV is a well-known and powerful library with a strong following of community developers and programmers. This platform was built on optimized code in C/C++ but can be used with Python, Java, and for some tools even MATLAB. For instance, interfacing with Python isn't difficult as the Python OpenCV functions are only wrappers for the original C/C++ optimized code which runs underneath. Therefore, a user can combine the accuracy of running OpenCV in C with an easier programming language such as Python. An example of OpenCV's image processing capabilities can be seen in Figure 8.

Figure 8. Example of OpenCV's technical prowess in image processing

A pros and cons breakdown for using OpenCV and C++ can be seen in Table 4.4.1.

Pros	Cons
Open source software	Intense learning curve for new programmers
Extensive optimized libraries for image processing (OpenCV)	Documentation is lacking
Fast runtime of software applications	
Large community of programmers	

4.4.2. OpenCV in Python

Python is an open-source and general-purpose programming language that has been extremely popular with developers since its inception. This programming language has a wide variety of uses from back-end web development, data analysis, artificial intelligence, and scientific computing. Python has been praised as a language simple to learn for beginning developers, more so than the core languages of Java and C/C++, while still allowing for the build of powerful software stacks, desktop applications, and much more. Common critiques with Python are that the software built in this language are not easy to maintain and performance overall can be lacking compared to a strictly compiled language like Java or C. A complete list of Pros vs Cons in using OpenCV Python for this project are listed in Table 4.4.2.

Pros	Cons
Open source software	Documentation is worse than OpenCV for C++
Code readability	Performance/runtime is lacking compared to that of a compiled language
Ease of software development	
Relatively fast runtime	
Extensive community for support	

As mentioned, Python is a great option when working with OpenCV as Python serves as a wrapper around the original and pre-compiled C code. This way, the optimized C code can be used on the back-end while Python operates on the front-end with benefits such as code simplicity and easier debugging without a drop-off in performance.

4.4.3. Matlab

MATLAB is a numerical computing environment and proprietary programming language provided by MathWorks. MATLAB is used extensively in scientific disciplines and academia, notably in aerospace applications. The necessary image processing and machine learning toolboxes are available in external MATLAB libraries for use of this project. Since MATLAB is a commercial product, it is itself an expensive software suite for hobbyists. CU students do possess software licenses to use MATLAB for academic purposes.

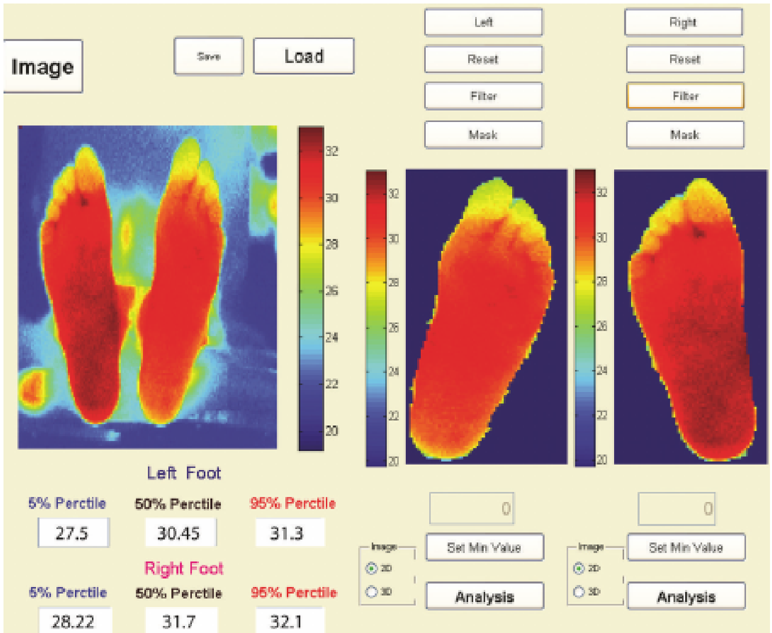


Figure 9. Example of MATLAB's image processing capabilities given a thermal image capture

Pros	Cons
Simple syntax	Slow runtime of software applications
Good documentation and dense community of programmers	Accuracy is lacking in floating point operations, compared to C pre-compiled code
Great library of matrix manipulating functions	Optimized image processing libraries are not extensive
Simulink stand-alone applications	Expensive software suite
OpenCV portability	

4.5. Processor

In the OTheRS electronics subsystem, 3 types of computational options were considered. These options were specifically considered because they have a vast heritage in aerospace applications. The OTheRS contemplated the utilization of a microcontroller (MCU), a microprocessing Unit (MPU), and a field-programmable gate array (FPGA), while discarding the options of using a Complex Programmable Logic Device (CPLD) or an Application Specific Integrated Circuit (ASIC) due to their inability to meet processing requirements. Since all three computational considerations have a deep history on satellite applications, all three could accomplish the electrical needs of the system. Because of this, the Trade study will focus heavily on a deeper analysis of available options.

4.5.1. Microcontroller:

The microcontroller can be thought of as a small computer on a single integrated circuit. What distinguishes this option from the other two is that the microcontroller has all the peripherals already connected to the unit. These peripherals include the memory, CPU, and I/O ports of the microcontroller. The code instructions for the microcontroller must be programmed in either high-level languages such at C, Python, and Ruby, or in low-level assembly using mnemonics. A simple pro and con list is given below for the microcontroller.

Pros	Cons
Prepackaged peripherals	Cannot process in parallel
Low cost	Programming changes depending on each different MCU
Software flexibility	

4.5.2. Microprocessor:

The MPU option is essentially just the computational part of a microcontroller. It is a clock based processor that executes centralized commands to be stored elsewhere. The big difference between the MPU and the MCU, is that the MPU does not have any inherent peripherals such as memory, I/O, timers, and clocks. These differences are highlighted in figure 10 below.

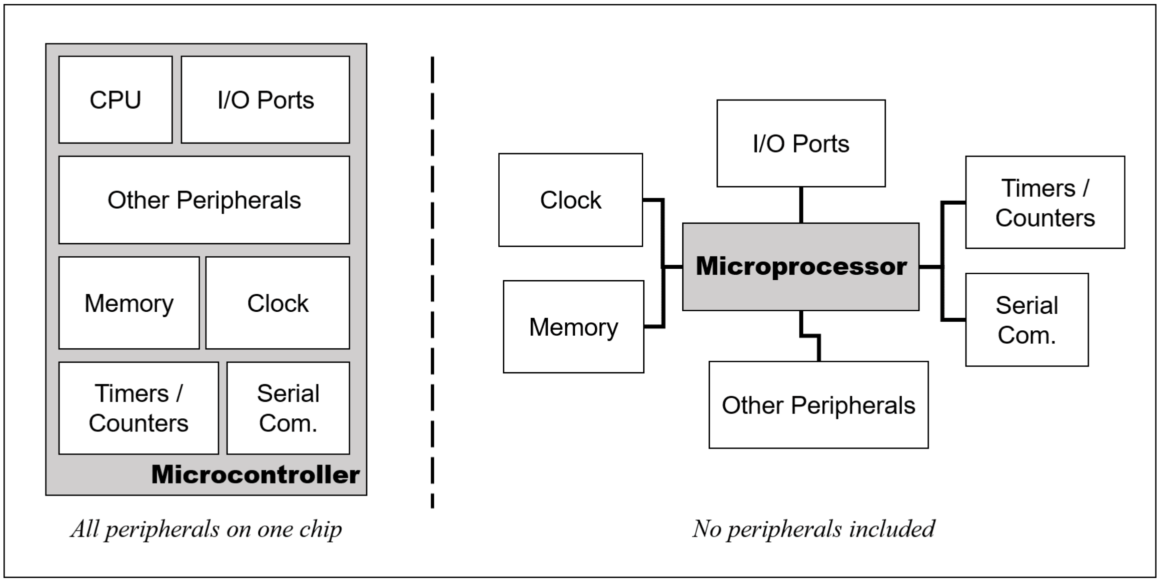


Figure 10. Peripheral difference between MCU and MPU

MPUs can handle both sequential and combinational digital logic. MPUs must execute code from the read-only memory, whereas MCUs generally load programs into RAM for execution. A simple pro and con list is given below for the microprocessing option.

Pros	Cons
Ability to select and choose peripherals on an as needed basis.	Reliance on external memory
High processing efficiency	Relatively higher cost

4.5.3. FPGA:

FPGAs differ greatly from both MPUs and MCUs. Specifically, FPGAs have hardware flexibility, they are designed for hardware configurations to be fluidly changed. The FPGA is simply in array of logic gates that must be specified. These gates are generally specified by using Hardware description language (HDL). A major difference between FPGAs and MCUs, is that the FPGA is able to process instructions in parallel. Figure 11 below shows the generic architecture for FPGAs.

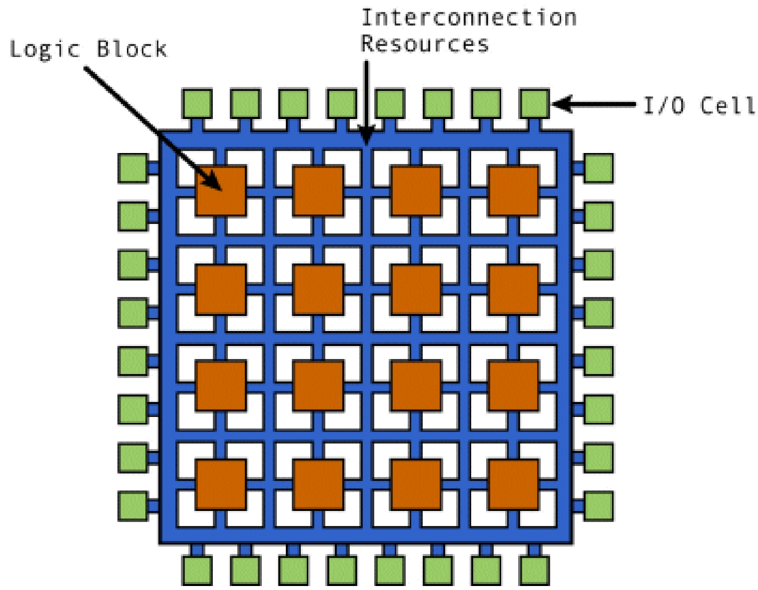


Figure 11. Generic Architecture of a FPGA

A simple pro and con list is given below for the FPGA option.

Pros	Cons
Hardware flexibility	High relative cost
Can process in parallel	Difficult configuration

4.6. Test Bed

The goal of the Test Bed is to use OTheRS on a fabricated aluminum avionics stack which is being heated up to or cooled beyond the given temperature range. Ideally the setup of the Test Bed will match the satellite dimensions given by GA so that it can properly replicate the satellite and show the scatter affects encountered within the actual satellite. The test bed will also incorporate a thermistor system in order to compare the accuracy of OTheRS against the accuracy of more common thermal measurement systems. A replicated fabricated avionics stack will be created in the machine shop complete with fabricated module trays. The avionics stack will then be coated in aluminum to mimic the real avionics stack used by GA. Outside the avionics stack, a metal frame will be used to mount the cameras on. A fabricated nanotray, whose specifications are also given by GA, will be created and have the processor placed within in order to show these components can fit on a standard GA nanotray. All of these parts will be expensive to create and will cause a dip in the budget. Resistors would then be placed inside the fabricated stack in order to heat the stack beyond the temperature range. The resistors will not be able to be precisely controlled but will be able to exceed the temperature range specified by GA. Once the OTheRS senses the temperature go above the specified maximum temperature, it will send a serial command to turn on a light bulb to show the functionality of the system. The recorded temperatures given by the thermistors will be logged onto a computer and will be compared with the temperature taken by OTheRS. This test bed design will demonstrate the viability of the overall system.

Pros	Cons
Test Bed Replicates the Satellite	High cost to build
Scatter effects are accounted for	Resistor temperatures cannot be completely controlled
Accuracy of OTheRS and Thermistors can be compared	

4.7. Control System

Our goal is to maintain the stack electronics between the temperatures specified in **FR 2**. Electronics are some of the most difficult items to keep working in a satellite. If they either get too cold or too hot, they fail. Failed avionics could compromise a satellite's mission. Electronics in a satellite consume electrical power and produce heat ($\dot{Q}_P$). That heat needs to be conducted ($\dot{Q}_{cond}$) or radiated ($\dot{Q}_{rad}$) away in order to stop the components from overheating. Solar and Earth-radiated (albedo) heat loads ($\dot{Q}_{ext}$) also cause periodic increases in heat flux each time the satellite comes around to the daylight side of its orbit. However, electronics that get too cold could also fail. Therefore, we need to be able to control the heat flux into and out of the electronics stack and apply electrical heat ($\dot{Q}_H$) if current heat loads are insufficient to keep the electronics warm. Several thermal control systems have been flown in space. Cutting edge research is ongoing in both active and passive control schemes. Because passive thermal controls frequently rely on a careful balance of environmental factors or complicated hardware and consumables, such as the ice sublimation cooling used on American space suits[4], these schemes are out of reach for this project. We are therefore left with active thermal control systems where we attempt to drive the heat of the avionics components to a target temperature. The only small satellite active control system currently designated Technology Readiness Level[6] (TRL) 9 by NASA uses electrical heaters[5].

On that basis, we traded on control schemes which use an electric heater to maintain a minimum temperature for the stack electronics and sufficient heat-sinking to stop potential overheating. We compared three different control schemes, listed below.

4.7.1. Threshold Based Control

Threshold based control is a form of closed loop control where the heater element is switched fully on or fully off. An upper and lower threshold set the temperatures at which the heater is switched on or off. If the lower threshold is reached, the heater is switched on. If the upper threshold is reached, the heater is switched off, similar to a household thermostat. The primary advantage of this control scheme is its simplicity and easy analysis.

At the coldest tolerable temperature, we turn on the heater. We then only need to check that the total heat flux into the electronics stack is positive. This comes down to the equation:

$$\dot{Q}_P + \dot{Q}_H + \dot{Q}_{ext} - \dot{Q}_{rad} - \dot{Q}_{cond} > 0 \tag{5}$$

At the hottest tolerable temperature, we turn the heater off and insure that the next heat flux into the electronics stack is negative:

$$\dot{Q}_P + \dot{Q}_{ext} - \dot{Q}_{rad} - \dot{Q}_{cond} < 0 \tag{6}$$

With these two points satisfied, we can be assured that we can keep the satellite inside our tolerable temperature range. However, in order to ensure good performance, we would probably utilize a larger than necessary heat sinks and heater. This control method would cause large and rapid swings in temperature - from one end of the acceptable scale to the other. We could mitigate such severe swings by setting tighter threshold temperatures. However, without changing out the heaters and heat sinks, we would simply oscillate faster over a shorter temperature range. This may not be an issue, but in the case of a mechanical switch such as a relay for the heater, this could cause premature failure.

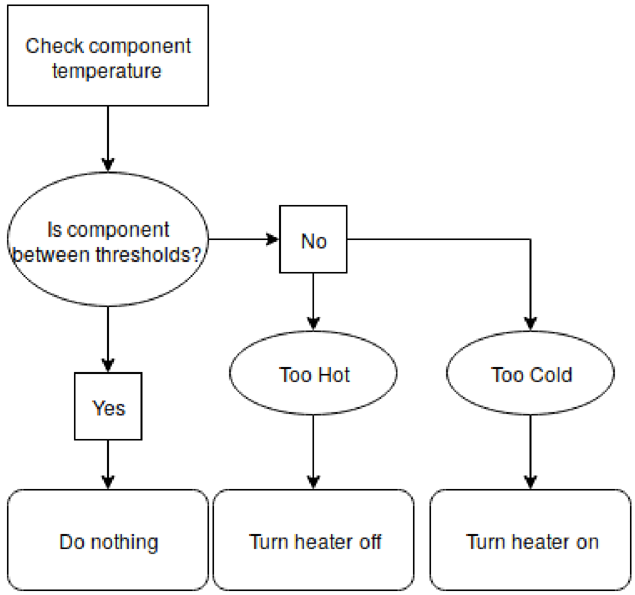


Figure 12. Threshold Based Control Flowchart

Pros	Cons
Easy to implement	Causes temperature oscillations
Hardware is simple	Hardware may wear out prematurely
Short implementation time	Constant temperature oscillations may cause premature failures in other components

4.7.2. Proportional Control

To stop the temperature oscillations, we might instead try and run the heater at some specific power setting in order to try and drive the components to a target temperature. In this case, we'd aim to balance the heat flux in and out of the system in order to maintain steady state.

$$\dot{Q}_P + \dot{Q}_H + \dot{Q}_{ext} - \dot{Q}_{rad} - \dot{Q}_{cond} = 0 \tag{7}$$

In the equation above, we would control $\dot{Q}_H$ and the problem turns to controlling this input value. One of the most common ways to control this value is using proportional control, where we drive $\dot{Q}_H$ according to the rule:

$$\dot{Q}_H = K_P(T_{target} - T_{actual}) \tag{8}$$

K_P is a *gain* value that we would either pick analytically or experimentally. It has units of Watts per degree Kelvin.

This system would give more stable and controllable temperatures. However, we would require electronic switching hardware to arbitrarily modulate $\dot{Q}_H$. This is unlikely to be a relay, like in the threshold based control. Instead, we would probably use a high frequency Pulse Width Modulation (PWM) signal to drive a MOSFET in order to supply the power to the heater. The MOSFET is a solid state switch that can be modulated much faster than a mechanical device with much less wear. This introduces some additional hardware and software complexity, as we'd need a way to calculate the difference between our goal and actual temperature, use the gain value K_P to determine the appropriate heater wattage, and then create a PWM signal of the appropriate duty cycle in order to produce that wattage from the heater. In addition, K_P may be difficult to determine. The rate of heat loss due to radiation from an object is $\dot{Q}_{rad,out} = \epsilon\sigma T^4$, where ϵ is the object's emissivity, σ is the Boltzmann Constant, and T is the object's temperature. Because T is raised to the fourth power, we may have difficulty using the analytic methods we have learned in class. Finally, proportional-only control does not damp small oscillations, which may persist and it does not account for environmental biases or improperly set values of K_P. Small errors in temperature setting could persist indefinitely.

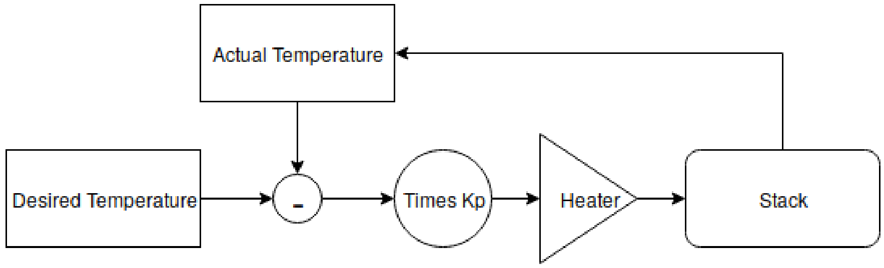


Figure 13. Proportional Based Control Flowchart

Pros	Cons
More stable than threshold control	Still prone to some oscillations and lingering bias
Allows us to set a specific target temperature	Will probably require solid state switching at high frequencies.
	Requires tuning to find a good gain

4.7.3. PID Control

A control scheme which solves the values associated with proportional control is proportional-integral-derivative control. The heater modulation law is changed to:

$$\dot{Q}_H(t) = K_P(T_{target}(t) - T_{actual}(t)) + K_I \int_0^t (T_{target}(t) - T_{actual}(t))dt + K_D \frac{d[T_{target}(t) - T_{actual}(t)]}{dt} \tag{9}$$

The new term $K_I \int_0^t (T_{target}(t) - T_{actual}(t))dt$ compensates for lingering error over time by progressively driving the average error to zero. The term $K_D \frac{dT(t)}{dt}$ damps oscillations.

PID control algorithms are common in industry wherever tight and repeatable automatic control is needed. They do not require any different hardware than proportional control scheme. However, they are more difficult to tune because in addition to K_P, the new gain variables K_I and K_D also need to be determined. Because the system response is coupled between all three variables, finding the correct gains by trial and error becomes substantially more difficult. In addition, our implementation must become *stateful*. We would need to keep track of the temperature at all points in the past in order to compute the integral and differential terms properly. However, well tuned PID control can keep a very stable temperature through many different environments.

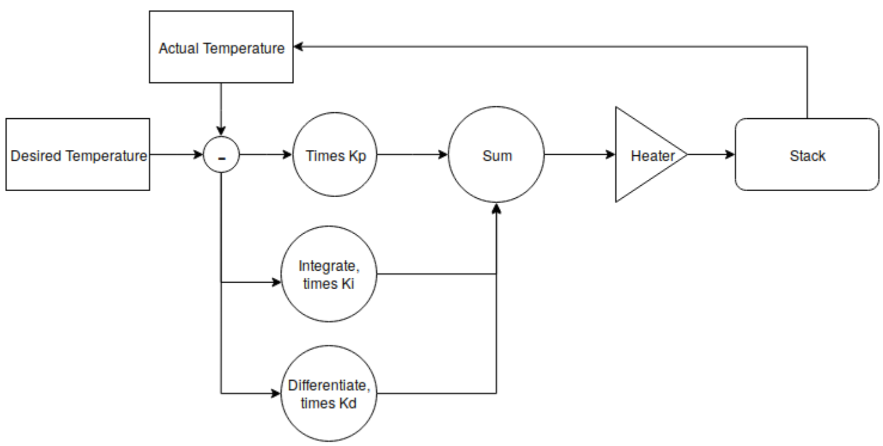


Figure 14. PID Based Control Flowchart

Pros	Cons
Can successfully damp out oscillations and remove bias	Much harder to analyze
Can hold a target temperature	Much harder to tune
No further hardware requirements over proportional control	Software requirements are more complex
	Have to store historical temperature information

Note: some clarification is needed around the term 'high frequency.' An electrical engineer's idea of high frequency is very different from an aerospace engineer's. In this context, high frequency only means a few hundred Hz. While this is a far from radio frequency signaling, it should be all we need. We expect the time scales associated with temperature changes to be in the seconds or tens of seconds. An Arduino Uno generates a PWM signal at $500Hz$. The time scales differ by two or three orders of magnitude. For the goal of avoiding temperature fluctuations and component wear caused by the constant on-off of a heater element, this is sufficient.

4.8. Thermal Control

In order to ensure that the various components of OTheRS remain within their operational temperature bounds, the OTheRS design solution must include the ability to measure its own temperature; only with this data can OTheRS fulfill requirement **FR 7**. The difference between section **4.7** and **4.8** is that the thermal control section refers to methodology used to accomplish the above task. However, the subsections below follow the same logical control considerations made above in **4.7**.

Sections *4.8.1* to *4.8.3* describe the self-monitoring methods of a multiple camera system, thermistors, or a combination of the two. Furthermore, sections *4.8.4* to *4.8.5* describe heat control techniques such as turning heaters on or off, or switching to low-power state modes. Turning systems on or off may be considered for the final solution, but comes with few benefits or costs. Physical systems were not considered here to keep the scope of the project consistent with customer requests.

4.8.1. Multiple Cameras

Using a camera-in-the-loop control system for our own thermal makes sense given our goal of replacing thermistors with thermal cameras. They would provide accurate and presumable, total coverage of our own electronics. However, the cameras themselves are active power consuming components with the associated failure rates and monitoring requirements. We would need to have each camera be monitored by another. That monitoring scheme would place constraints on our possible camera layouts and has the potiential to limit our ability to image our primary target: the avionics stack.

Pros	Cons
Increased image fidelity	Increased system complexity
System failure back-up	Increased cost & weight
Increased mounting options	Mounting requirements
	More data handling/processing required

4.8.2. Thermistors

Thermistors are the current and proven technology for temperature monitoring of avionics. However, we are trying to replace the thermistors with thermal cameras. While they are cheap and easy to monitor, they require analog to digital conversion electronics and increased number of wires from each component, as well as a separate converter channel or a signal mixer in order to handle a large number of sensors.

Pros	Cons
Lowest cost	Increased wiring complexity
No calibration needed	More data handling/processing required
High data rate	Old tech
Reliable and proven	We are attempting to eliminate thermistors from the design.

4.8.3. Combination

Using a combination of thermistors and cameras would not completely eliminate the downsides of each. Some of the cameras would still need to image each other, and others would still need thermistors. However, multiple data sources would increase the robustness of the overall design.

Pros	Cons
Highest system accuracy	Highest wiring complexity
Highest system redundancy	Highest cost and weight
Most data provided	Most extensive mounting requirements

4.8.4. Heater Control

In order to respond to thermal information about our own system, we need a method of thermal control. Only two reliable methods exist without modification of the satellite design. We can artificially heat components with electrical heaters or we can stop heat production by switching components off. Despite limited options, we need to consider this in order to ensure that the cameras and other electronics keep functioning at peak performance and reliability.

Pros	Cons
Simple to test and calibrate	Increased software implementation difficulty
Cost effective	Increased power away from electronics tray

4.8.5. Power-State Control

For processors and components that support it, we could instead switch them to a low-power state. This has the advantage of being better supported by some components than simply cutting power. However, it is not a universal feature and even components that are in standby or sleep modes still consume some amount of power.

Pros	Cons
Provides cooling without turning components on and off	Increased software implementation difficulty
Cost effective	Not all components may support this

4.9. Thermal Model

The thermal model refers to the software demonstration of heat transfer between the electronics components due to conduction & radiation within the electronics bay. Software used for thermal modeling will depend on many factors and some will not be immediately applicable to our project. The three software platforms traded on to perform accurate thermal modeling will be: user-designed heat transfer in a rod, SolidWorks with additional thermal Flow Simulation package, and Thermal Desktop.

4.9.1. Heat Transfer in a Rod

The first option is to design a thermal model of heat transfer in a rod, which will be built by the user. This option is by far the simpler model for testing, but will lack sophistication when it comes to validation. Real heat transfer that will take place between components will be three-dimensional, and will require some level of complex modeling.

Pros	Cons
Simplest Model	Not as applicable to our application
Introductory model	Inaccurate results for our application
Proof of concept on an accelerated timeline	Only two-dimensional analysis

4.9.2. SolidWorks Flow Simulation

This second option is to use an accompanying package to SolidWorks known as Flow Simulation, as opposed to similar modeling packages in which students have no experience. SolidWorks has license provided by the university and can produce an adequate analysis of the electrical components in question, based on the CAD model provided by GA and further model simplification done by the OTheRS team. SolidWorks is great for modelling, but is poorer in simulation compared to a software such as ANSYS.

Pros	Cons
Easy to use	Poor simulation properties
Simple setup (single toolbox)	Moderate learning curve
Great for modeling	Implementation fidelity
3D Analysis	
University-provided license	

4.9.3. Thermal Desktop

Pros	Cons
Very accurate model	Cost is astronomical
Great simulation	Complicated software
	High learning curve

The cost of a single thermal desktop license ($40,000) is the biggest detriment to this option and is thus is an easy throw out for the thermal model trade study.

4.10. Thermistor Selection

Due to constraints given by the customer, the goal in thermistor selection is to maximize resolution and flight-readiness while considering cost and wiring complexity. As such, one of many Measurement Specialties choices was selected - the GA30K5A1IA 30K NTC series thermistor provides reasonable component cost (at $6.15 per) while still fulfilling the system requirement of $-50 - 60 \deg C$, within the proposed system dimension requirement. Traditional thermocouple systems include complex programming/formatting requirements, as well as too many data processing requirements to remain relevant. Lower fidelity thermistors and thermocouples could compromise this data, in terms of both data rates and flight-readiness. Historically, NASA flight readiness level 9 (FR9) has been difficult for companies to achieve for all flight components - temperature management for the test bed aspect of this project is extremely important, as it validates the performance of all other design considerations of the project. Based on this, industry standard performance, flight-readiness, and ease of implementation allow for simple and early choice of the thermistor component.

5. Trade Study

5.1. Camera

Multiple cameras had to be researched in order for OTheRS to decide which would be best for this project. The metrics taken into account to make this decision were cost, size, power, resolution, operating temperature range, thermal sensing range, and HFOV. Table 1 below describes why each metric is weighted as it is.

Table 1. Metric Definitions

Metric	Weighting	Justification
Cost	0.2	The hard limit of the budget for the entire project is $5,000. Not all this money can go towards the thermal camera, so minimizing the cost of the camera while making sure the camera is still of good quality is ideal.
Size	0.15	There are 102mm between the stack and the outer wall of the shell where the camera will be mounted. The depth of the camera size needs to be minimized in order to maximize the field of view.
Power	0.05	The customer is giving us 28V of input power for the entire system. This must be distributed between the camera and all of the electrical components.
Resolution	0.25	The thermal camera must sense the temperature variation across each electronics tray. In order to do this, it must have a high enough resolution to distinguish between the trays.
Operating Temperature Range	0.1	The customer would like the camera to be operational between -30°C and +60°C. However, the limits of the non ITAR cameras available could constrict this range.
Thermal Sensing Range	0.05	The thermal camera needs to be sensitive to infrared radiation ranging from -30°C to +60°C.
FOV	0.2	We want to minimize the number of cameras per side of the stack while still being able to image the entire side

Table 2 below defines how each metric was broken up in order to get a raw score for each of the cameras involved in the trade study.

Table 2. Scoring Matrix

	Scoring Criteria				
Metric	**1**	**2**	**3**	**4**	**5**
Cost	> $1,000	> $700, < $1,000	> $400, < $700	> $100, < $400	< $100
Power	>28V	>8V, <28V	>5V, <8V	>3V, <5V	<3V
Size	> 30mm	< 30mm, > 25mm	< 25mm, > 20mm	< 20mm, > 15mm	< 15mm
Resolution	<1,000 pixels	<2,000 pixels, >1,000 pixels	<3,000 pixels, >2,000 pixels	<3,000 pixels, >4,000 pixels	>4,000 pixels
Operating Temperature Range	< +10°C to > +20°C	< 0°C to > 30°C	< -10°C to > +40°C	< -20°C to > +50°C	< -30°C to > +60°C
Thermal Sensing Range	< +10°C to > +20°C	< 0°C to > 30°C	< -10°C to > +40°C	< -20°C to > +50°C	< -30°C to > +60°C
FOV	< 30°	> 30°, < 60°	> 60°, < 90°	> 90°, < 120°	>120 °

Table 6 below shows the selected raw scores for each metric for each camera. These values are then multiplied by the corresponding weighting of each metric and summed together to produce the weighted totals at the bottom of the table.

Table 3. Trade Matrix

Metric	Weighting	FLIR Lepton RawScore	Tamarisk Precision RawScore	MLX90640 RawScore
Cost	0.2	4	1	5
Size	0.15	5	1	5
Power	0.05	5	4	4
Resolution	0.25	5	5	1
Operating Temperature Range	0.1	3	4	5
Thermal Sensing Range	0.05	3	5	5
FOV	0.2	2	2	4
Weighted Totals	1	**3.9**	**2.6**	**3.75**

From the results of the trade study, it is apparent that the FLIR Lepton camera should be used for the project. General Atomics has provided OTheRS with one FLIR Lepton Breakout Board v1.4. The specific camera that GA is providing will have to be tested to make sure it is in proper working condition because it has been inside of a thermal vacuum chamber during operation.

The FLIR Lepton Breakout Board v1.4 is the fourth model of the FLIR Lepton Longwave Infrared (LWIR) thermal imaging camera. It can detect waves anywhere between 8-14 micrometers. It is meant to be operational in the temperature range of -10°C - +65°C and non-operational in the temperature range of -40°C to +80°C. This camera has a diagonal field of view (FOV) of 63.5 deg and a horizontal FOV of 51 deg. The depth of field is the field between the closest and farthest object in view that are visually sharp. The field for this FLIR Lepton is from 10 cm to infinity. This thermal camera is able to detect temperatures with an accuracy of 0.050 deg. This will produce a crisper image than a majority of different thermal cameras, as the average sensitivity is 0.1°C with the higher range <0.08°C and the lower range of 0.2°C. A block diagram can be seen in figure 15 below.

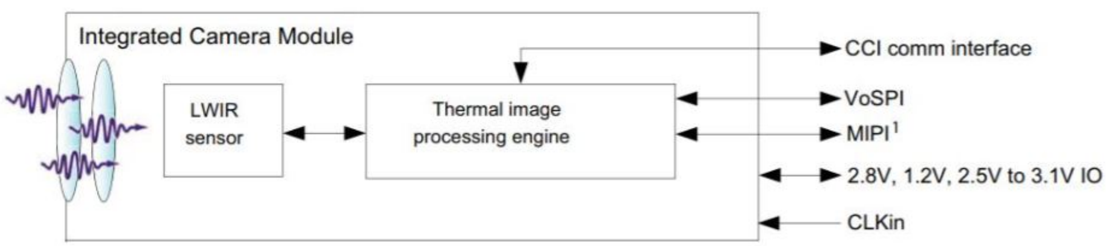


Figure 15. Block diagram of FLIR Lepton camera

5.2. Camera Setup

The metrics taken into account for the camera setup include cost, power, system complexity, coverage, and weight. The table below describes the reasoning for each metric.

Table 4. Metric Definitions

Metric	Weighting	Justification
Cost	0.2	The hard limit of the budget for the entire project is $5,000. This money will have to be spread out between all systems in order to ensure every system is properly funded.
Size	0.05	The requirements given by General Atomics were to minimize size and mass of the system. The entire setup that will be chosen must be able to fit in between the walls of the satellite and the given avionics stack. The size metrics given in the following table represent the amount of space taken up by the camera system. Since the cameras used will be relatively small this is a pretty small metric.
Power	0.15	The customer is giving us 28V of input power for the entire system. This must be distributed between the camera system and all of the electrical components. The power metrics given for the following table represent the power that will be used for the camera system on each wall while it is active
Coverage	0.3	At least an entire side of an avionics stack must be covered in order to show a proof of concept of OTheRS. Ideally though the entire avionics stack will be able to be captured by OTheRS to show how using a system similar to OTheRS is more efficient than using thermistors
System Complexity	0.3	The Customer wanted this to mainly be made of COTS parts. If parts need to be machined or different means have to be used to to help increase the coverage of the camera, this will add complications to calibration of the system and can hurt the accuracy. This also defines the experience the team has with optics. This metric is quite high since the team has low formal experience with optics.

Table 5. Scoring Matrix

	Scoring Criteria				
Metric	**1**	**2**	**3**	**4**	**5**
Cost	> $4000	> $2500	> $1500	> $1000	< $500
Coverage	System can only view less than half of a Stack Wall	System can only view half of a Stack Wall	System can only view a single stack wall	System can view at least two stack walls	System can view the entire stack
Size	$> 20000mm^2$	$> 10000mm^2$	$> 1000mm^2$	$> 500mm^2$	$< 500mm^2$
Electrical	$> 3000mW$	> 2000 mW	$> 1500mW$	$> 500mW$	$< 200mW$
Complexity	All Parts must be machined and Group has no experience with the hardware	Mostly non COTS parts used, team has under 20 hours of experience with the hardware	More than half the parts used are COTS. Team has between 20-60 hours with hardware.	Most Parts used are COTS, team has between 60 and 100 hours of experience with the hardware	All Parts used are COTS and Group has over 100 hours of experience with the hardware.

The scores for each different setup based on these given metrics are given below.

Table 6. Trade Matrix

Metric	Weighting	Multiple Camera Setup	Optical Manipulation	MLX Sensor Array
Cost	0.2	3	4	5
Size	0.05	4	5	4
Power	0.15	4	5	5
Coverage	0.3	3	5	5
Complexity	0.3	5	3	4
Weighted Totals	1	**3.8**	**4.2**	**4.60**

From this trade study it is clear that using two IR Arrays with MLX90640 sensors would be the best setup to take temperatures of the avionics stack. Using IR arrays with MLX sensors would allow the entire stack to be measured and modeled instead of just one side of the stack which would occur in the multiple camera approach. This will also cause a massive reduction in size and power compared to the multiple camera approach. Compared to the Optical Manipulation design decision this will be much easier to implement and will still ensure all parts used are COTS. One thing to notice though is the inconsistency between the camera setup and camera trade studies. The Camera trade study states to use a FLIR Lepton while the Camera setup states to use MLX90640 sensor arrays to image the stack. This is due to not knowing whether coverage or accuracy is more important for the customer. With the MLX90640 sensor array the entire avionics stack can be measured easily but at a much lower accuracy than the FLIR Lepton gives. However, if accuracy is the most important facet of the project for the customer then a setup of FLIR Leptons would be used on a single side of a fabricated stack. In either case, adding lenses to the FLIR Lepton to increase its Field of View will not be used due to the effect that new lenses will have on the Leptons accuracy. Accuracy is included in the camera trade study but not the camera setup trade study since the only setup that affects the the original accuracy of the system is the Optical Manipulation design, and how the accuracy is affected is not quantitatively known.

5.3. Image Processing

A trade study of image processing methodologies was completed to determine which platform between OpenCV Python, OpenCV C++, and Matlab would be best suited for the image processing design requirements. The metrics chosen for this trade study are programming language difficulty, level of documentation (additionally, community support), accuracy of floating point operations, and the libraries of IP functions available. The metric definitions for image processing can be found in Table 7.

Table 7. Metric Definitions

Metric	Weighting	Justification
Programming language difficulty	0.2	The difficulty of programming in a specific language can affect the speed at which software is developed. Experience of the developer also plays a factor as a simple algorithm implemented poorly can lead to inefficient and wasteful code. In the case of programming in C/C++, lazy programming can lead to system memory leaks and further damage. Some languages like Python and MATLAB are high-level and do not require as much tedious work in developing software components.
Runtime	0.3	All image processing will need to be performed on-board and within the electronics bay of the proposed GA system. For this reason, the image processing software requires a true level of runtime sophistication to perform the necessary duties with each image in the time between image captures.
Level of Documentation	0.1	Documentation is a key metric for this trade study, as the image processing software tools available are extensive. A community of like-minded programmers for troubleshooting/debugging support is also helpful when critical software micro-services fail. Community and support available during software development, depends mostly on the programming language of choice.
Accuracy of floating point operations	0.2	Fairly accurate calculations are required to track pixel temperature and when otherwise moving pixel data. This is a key metric for image processing in general which requires high latency and correct calculations to achieve accurate results. Accuracy of floating point operations will factor more into locating electronics components, due to locating and moving pixels in the image RGB stack.
Libraries of IP functions	0.2	The libraries focused on in this trade study are all expansive and require deliberate planning before actual coding takes place. OpenCV can be ported to several different languages including Python, C++, Java. All Python functions use pre-compiled C++ code. Therefore, C++ can be used minimally to develop the underlying image processing, but can be used later on to alter these same functions.

Note that the largest weighting for this trade study is spent on the runtime of the programming language and therefore weighted heavily based on performance relating to image-processing. As an aside, OpenCV Python and C++ both use pre-compiled C code that is highly optimized, while MATLAB has lacking performance when completing simple tasks such as calling external libraries. MATLAB programming language is itself a scripting language, but is compiled to machine code using the JIT compiler. However, performance can be increased many more times in C++ compared to MATLAB vectorized code.

Table 8. Scoring Matrix

	Scoring Criteria				
Metric	**1**	**2**	**3**	**4**	**5**
Programming language difficulty	Software is extremely difficult to use. Team has little experience using this product.	Software is more difficult to use than not. Team has less experience than not in using this software.	Software is fairly easy to use. Team has a decent amount of experience.	Software is easier to use than not. Team has more experience than not in using this software.	Software is extremely easy to use. Team has generous experience using this product.
Runtime	Terrible runtime. Non-ideal case.	Sub-average runtime.	Average runtime.	Better than average runtime. Bordering on ideal.	Impressive runtime. Ideal case, pre-compiled code.
Level of Documentation	Level of documentation and community support is lacking in all regards.	Level of documentation and community support is worse than fair.	Level of documentation and community support is fair.	Level of documentation and community support is better than fair.	Level of documentation and community support is impressive in all regards.
Accuracy of floating point operations	Software lacks the necessary tools to produce any accurate results, even when developed optimally.	Software has necessary tools to produce at-best sub-average accuracy.	Software has necessary tools to produce average accuracy.	Software has necessary tools to produce better than average accuracy.	Software has the tools necessary to produce extremely accurate results when developed optimally.
Library of IP functions	Minimal library of IP functions, nearly no tools required are present. Nearly all components must be custom-made and developed.	Less extensive library of IP functions than not.	Fair library of IP functions. Average amount of custom-made components need be developed.	More extensive library of IP functions than not.	Extensive library of IP functions, all tools required are present. Few custom-made components need be developed.

Table 9. Trade Matrix

Metric	**Weighting**	**OpenCV Python Raw Score (1-5)**	**OpenCV C/C++ Raw Score (1-5)**	**Matlab Raw Score (1-5)**
Programming language difficulty	0.20	4	3	5
Runtime	0.30	5	5	2
Level of Documentation	0.10	3	2	4
Accuracy of floating point operations	0.20	4	4	2
Library of IP functions	0.20	5	5	3
Weighted Totals	1	**4.3**	**4.0**	**3.0**

A few additional justifications follow. Since OpenCV uses the same underlying compiled code in C, OpenCV in Python is at least as-efficient when it comes to run time and accuracy of floating operations when this image processing platform is used. MATLAB, although used extensively in aerospace applications and known well enough by all group numbers, was not chosen as seen by the trade study results. It was determined that on-board image processing would not be able to be handled using MATLAB based on the run time requirements for this specific project application. From this trade study, OpenCV Python is clearly the front-runner image processing platform of choice for this project. Additional development in C/C++ will proceed when the need inevitably arises for custom software components, with a focus on maintaining an optimal-performance software stack.

5.4. Processor

Trade studies were conducted to determine which processor, between the MPU, the MCU, and the FPGA, would be used to meet the processing design requirements. The metrics used to make that determination were hardware integration, software experience and complexity, size and weight, power consumption, processing power, and cost.

Table 10. Metric Definitions

Metric	Weighting	Justification
Software Experience and Complexity	0.20	This metric defines how much experience the team has with the software and the software language required to integrate with the specific processor. The desire to create an efficient software to run the image processing and other parts of this project results in a high weight for this metric.
Hardware Integration	0.20	Processors require a variable amount of electronic modifications to be able to run the desired software, changing required cost and workload, and therefore has a high weight.
Power Consumption	0.15	This metric takes into account the fact that satellites have only a limited amount of power,and the processor should not have too large of a power consumption. Processors usually meet that requirement, and this metric will therefore not have the highest weight.
Processing Power	0.20	The processor needs to run the software at the TBD desired rate. This is critical and therefore has higher weight.
Size and Weight	0.15	The size of the system is a design requirement while the customer specified to minimize weight, and will therefore have the higher weight as a metric.
Cost	0.10	The budget of the project is limited, but the price ranges of most processors are well within the budget, and this metric will therefore be weighted at a minimum.

Table 11. Scoring Matrix

	Scoring Criteria				
Metric	**1**	**2**	**3**	**4**	**5**
Software Experience and Complexity	Team has no previous experience with the required software.	Team averages under 20 hours of experience for required software	Team averages between 20 and 60 hours of experience for required software	Team averages between 60 and 100 hours of experience for required software	Team Averages over 100 hours of experience for the required software
Hardware integration	Hardware requires configuration of the specific board to be used as a processor and requires external components to run software.	Hardware requires external components and peripherals to run software.	Hardware comes with some of the components to run software but requires other external peripherals to interface with the rest of the system.	Hardware comes with all the required peripherals to interface with the system, but does not contain the more specific controllers.	Hardware has all necessary components and peripherals already integrated.
Processing power	Processor cannot run software at required frequency.	Processor can run software sequentially to meet image processing rate requirement but requires other hardware to run the internal thermal control software.	Processor can run both the image processing and the internal thermal control sequentially.	Processor can run the image processing in parallel but not while running the internal thermal control software.	Processor can run both the image processing and the thermal control software in parallel.
Power Consumption	Processor requires more than the system's TBD allowed power budget.	Processor operates on average close to the maximum power budget.	Processor can run intensive software without passing power budget	Processor does not need to be constantly supplied with power.	Processor does not require any power.
Size and Weight	Processor cannot fit in a nanotray.	Processor has a fixed size and can fit in a nanotray assuming minimum added peripherals.	Processor has a fixed size and can fit in a nanotray without other parts being affected.	Processor size can be adjusted.	Processor does not require space on the nanotray.
Cost	Average cost (including required components and peripherals) above $500	Average cost (including required components and peripherals) $300-$500	Average cost (including required components and peripherals) $200-$300	Average cost (including required components and peripherals) $100-$200	Average cost (including required components and peripherals) below $100

Table 12. Trade Matrix

Metric	Weighting	MCU Raw Score (1-5)	MPU Raw Score (1-5)	FPGA Raw Score (1-5)
Software Experience and Complexity	0.20	5	5	2
Hardware Integration	0.20	4	3	1
Power Consumption	0.15	3	3	3
Processing Power	0.20	3	3	4
Size and Weight	0.15	3	4	3
Cost	0.10	5	4	3
Weighted Totals	1	**3.80**	**3.65**	**2.60**

By looking at the total score of each processor alternatives, the microcontroller is chosen.

Microcontroller Unit (MCU)

The microcontroller uses coding language (C/assembly) in which the team members are very familiar with, and can easily integrate other languages. Furthermore, the amount of documentation available is quite extensive to integrate the software to the hardware. On one chip, the MCU contains most the peripherals necessary to effectively interface with the system, with the exception of specialized controllers, such as a CAN microchip if CAN is used for communication. The MCU runs its software sequentially, but will be able to run all the softwares required to meet all our requirements. While the MCU's power consumption, cost, size, and weight vary for each specific models, it can on average meet all the specific power budget and size requirements at a low cost.

Microprocessor Unit (MPU)

The microprocessor and microcontroller are very similar in terms of software complexity, requiring the same languages. However, the microprocessor does not possess all the peripherals necessary to run the software and interface with the rest of the system, increasing hardware complexity and cost. The processing power and power consumption are similar to the microcontroller, but often has a bigger size, due to the need to integrate other components independently, instead of having everything built on one chip. This also affects the weight of the processor system.

Field-Programmable Gate Array (FPGA)

The FPGA is not a familiar processor to anyone in the team, requiring languages such as HDL which can require a large amount of extra workload. Furthermore, the FPGA has to be configured to be able to run the desired softwares specifically and does not contain any peripherals required to interface with the rest of the system. The FPGA, however, can run all the softwares in parallel, allowing for faster and more efficient image processing. The power consumption is higher than the MCU and MPU, but can still fit inside the power budget. Similarily, FPGAs tend to be of a higher size and weight than their specialized, already configured counterparts, and are on average at a higher cost.

5.5. Control System

Table 13. Metric Definitions

Metric	Weighting	Justification
Mathematical Simplicity	0.4	A complicated system will take up much more of our time to analyze than a simple one, and time is at a premium.
Hardware Simplicity	0.1	Hardware options are fairly fixed. We either can only turn a heater on or off, or we can modulate power as required.
Thermal Stability	0.3	The longevity of satellite components depend, in part, on how well they're treated. Keeping temperatures very constant might be important.
Advance Knowledge Required	0.2	Some control systems have to be carefully tuned, which requires the whole system to be well modeled. Others might be very simple and not require the same amount of modeling and knowledge about the operating environment.

Table 14. Scoring Matrix

	Scoring Criteria				
Metric	**1**	**2**	**3**	**4**	**5**
Mathematical Simplicity	System requires in depth mathematical analysis to function	Performance is very poor without in depth mathematical analysis	Mathematical analysis improves performance	Mathematical analysis adds benefit but the same results can be accomplished without it	No mathematical analysis required
Hardware Simplicity	*unused*	*unused*	Solid state switching at high frequency needed	*unused*	Simple electronic on/off switch used
Thermal Stability	Oscillations likely less than $50K$	Oscillations likely less than $10K$	Oscillations likely less than $2K$	Oscillations likely less than $0.5K$	Oscillations likely less than $0.1K$
Advance Knowledge Required	Must fully characterize system and environment for all possible scenarios	Must fully characterize system, environmental factors can be neglected	Good characterization of system needed for stable performance	Good characterization of system not needed, trial and error can be used instead	No characterization needed

Table 15. Trade Matrix

Metric	Weighting	Threshold RawScore	Proportional RawScore	PID RawScore
Mathematical Simplicity	0.4	5	4	2
Hardware Simplicity	0.1	5	3	3
Thermal Stability	0.3	1	4	5
Advance Knowledge Required	0.2	4	4	2
Weighted Totals	1	**3.6**	**3.9**	**3**

6. Baseline Design Selection

In the sections below, the baseline design option for each subsystem or critical system component is laid out. These options we chosen based on the trade study results, extensive research, and some back of the envelope calculations based on our functional and design requirements.

6.1. Camera

The winning camera choice is the FLIR Lepton camera on the basis of its low power and small size. Depending on optical designs, it may be necessary to utilize multiple cameras in order to image the entire stack area. The space inside the satellite bus is limited and the field of view of one Lepton camera is too narrow to view the entire stack from inside the satellite. The camera choice is still contested as other infrared sensors have wider fields of view, but have other drawbacks such as lower resolution and accuracy. Our trade studies identified that multiple MLX90640 thermopile sensors give more effective coverage of the stack, but would limit the accuracy and resolution of the thermal profile. The final decision may change in the future as the needs and layout of our test bed is reconsidered.

6.2. Camera Layout and Test Bed

The baseline testbed is designed to approximate the avionics stack of the satellite. The mock avionics stack will be made up of stacked trays using the same material as the GA satellite bay. The exact dimensions of the outer walls will depend on the final camera configurations we select for our test. The baseline Camera Layout will use a wall of FLIR Lepton's to take the temperature of a single wall of a fabricated avionics stack. While the MLX90640 sensor array won the trade study there was also an explanation between whether coverage or accuracy is more important in the design. The baseline design will use a wall of FLIR lepton cameras in order to maximize accuracy unless the customer tells us differently. This is also the most accurate setup as the FLIR Leptons original lens has better accuracy than the MLX90640 sensor array with an added lens to increase the FOV.

6.3. Camera Calibration

For the purposes of OTheRS, a two point calibration system with one of the points coming from a black body radiator and the other point measured from a thermistor is sufficient. The linear thermal slope is unknown so one point calibration is not sufficient. Multi-point fitting would be too large to implement because of the limited require space near the stack. This leaves two-point calibration as the most viable method. As far as point determination, two thermistors would be ideal, however, by using two thermistors, it cannot be ensured that the required range of -30°C to 60°C is accurately captured by the two measured points. Thus, the optimal selection for point determination is one thermistor that measures a point near the stack that is expected to remain relatively thermally constant. The other point shall be assumed from a black body radiator that heats up to a temperature that is sufficiently far from thermistor measurement to increase the effective range of the calibration.

6.4. Image Processing

The image processing subsystem will involve the programming required to capture images via the camera, identify the stack's physical dimensions/positions, and extract temperature data of these components. This subsystem will interface with the camera and control systems, while being used in benchmark form during the test bed analysis. With the temperature data in hand, the control system will know which heaters to turn on/off in order to manage the thermal needs of the OTheRS. Regardless of camera choice, we will need to extract information from the images. If we use multiple cameras, we will likely need to perform advanced image processing in order to merge the images and create the thermal map. We chose the OpenCV image processing framework in Python on the basis of easy language constructs and the plethora of IP functions/paradigms available through OpenCV. This software stack will combine the optimized and high-performance pre-compiled C code available as the foundation of OpenCV with the easy

6.5. Processor

The Camera(s) will be connected to a microcontroller chip to process the images and relay data. The microcontroller will then control the temperatures of the electronics in the test bed stack switching electrical heaters according to proportional control law, or it will pretend to do so using LEDs as an indication of heater power. We will process and output data over a serial link as required in order to build a 'thermal map' of the avionics stack.

6.6. Control System

If we chose to include full power heaters, they will be switched according to proportional control law. This provides an optimum between stability and implementation difficulty. We will use a MOSFET or other similar transistor to switch up to $2.5A$ at high frequency using PWM modulation. If we chose to only use indicator LEDs then the transistors will switch those.

6.7. Thermistors

For performance comparisons we will also include thermistors or thermocouples of with an accuracy of $0.1°C$, which is industry standard. This requirement derives from the basic functionality requirement that our system be self-monitoring and self regulating. The test bed will focus on accuracy rather than a flight ready system.

References

[1] DigiKey, MLX90640 FIR Sensor, DigiKey Available: https://www.digikey.com/en/product-highlight/m/melexis/mlx90640-fir-sensor?&utm_adgroup=Supplier_Melexis&slid=&gclid=EAIaIQobChMIyeCD5Jbm3QIVVrjACh3LtwAzEAAYAiAAEgLF5_D_BwE

[2] Fluke, Thermography Fundamentals, Fluke Available: https://www.fluke.com/en-us/learn/best-practices/measurement-basics/thermography.

[3] Jackson, Jelliffe. "Conceptual Design Document (CDD)", University of Colorado–Boulder, Retrieved September 23, 2018, from https://learn.colorado.edu/d2l/home

[4] National Aeronautics and Space Administration, "Cool Spacesuits", Retrieved September 24, 2018, from https://www.nasa.gov/pdf/544886main_LS5_Cool-Space-Suits_C1.pdf

[5] National Aeronautics and Space Administration, "State of the Art of Small Spacecraft Technology", Retrieved Sept 24, 2018, from https://sst-soa.arc.nasa.gov/07-thermal

[6] National Aeronautics and Space Administration, "Definition Of Technology Readiness Levels ", Retrieved Sept 29, 2018, from https://esto.nasa.gov/files/trl_definitions.pdf

[7] Sierra Olympic Technologies Inc., Tamarisk Precision High Performance Thermal Cores, Sierra Olympic Technologies Inc. Available: https://www.sierraolympic.com/products/details/tamarisk-precision.

[8] Spark Fun, FLIR Radiometric Lepton Dev Kit, Spark Fun Available: https://www.sparkfun.com/products/14654.