Holcombe Department of Electrical and Computer Engineering
Clemson University

# Final Report:
# Quadcopter System Capable of FLIR Vision and Payload Delivery

Fall, 2022

Team 5 – Project NightWatch

Andrew Burrell
Kyle Jenko
Brandon Kim
Humberto Ruiz

Holcombe Department of
ELECTRICAL AND COMPUTER ENGINEERING
Clemson University

# Executive Summary

Project Nightwatch is a quadcopter integrated with a FLIR camera and payload delivery system. The design pairs the portability of a handheld drone with the capability of low light vision and ability to drop a small payload. We chose the Holystone HS700D GPS Drone & Controller for its price and availability. Infrared vision is made possible by the Teledyne FLIR Lepton 3.5, a miniature camera about the size of a dime. This camera can be easily connected to a microcontroller via PureThermal 2 Breakout Board and a micro-USB to USB adapter. All image processing is handled by the Lepton and Breakout Board, no further signal analysis is needed. Live FLIR feed can be delivered to a smartphone via wifi thanks to a custom built program. Our payload delivery system consists of two 3-D printed motor driven clamps that can be opened and closed by remote control. The clamps can fit an average sized water bottle. All non flight related systems are connected to a Raspberry Pi 4, which is housed in a custom 3-D printed chassis that can be clipped to the belly of the Drone. Both the clamp and the adjustable camera mount can be controlled by a custom button assembly connected via RF signal. Total expected flight time is approximately twenty-two minutes. The Project Nightwatch artifact has a total cost of $681.44, under the proposed $1500.00 budget. The project did encounter several problems during production, both with the camera and with a weight limit. An older breakout board connected to the RPi via SPI ports could not be recognized as a camera and the quadcopter could not generate enough lift to fly with our first assembly. Several revisions later, we designed a lighter system with a newer board. In retrospect, the project can be deemed a success in spite of several setbacks. The final drone and chassis assembly can fly without disruption due to its light weight and the 3-D print's hexagonal lattice design. Our custom button assembly works as intended with the RF modules allowing for greater operational range, and FLIR Lepton video can be live streamed over wifi with its advertised resolution and minimal framerate drops. Future project designers will have to contend with the trade off between cost, performance, and portability. It is recommended that similar projects take into consideration weight limits of the drone. Power consumption was a minimal issue compared to the flight capabilities of the entire system. Although more portable, smaller models can only carry lightweight payloads and will have a shorter battery life. In contrast, bigger drones will cost more and will be harder to carry. Without resorting to a larger drone, possible solutions include more powerful motors or adding more rotary blades.

## Abstract

Modern drones provide a valuable reconnaissance tool in a light and portable package; most come with an onboard vision system that relay live feed back to its operator. Unfortunately, most small drone cameras are meant for day use only. Some models, like dedicated fishing drones and heavy-duty package drones, are even equipped to deliver a payload during flight but are far too large to be carried. Team 5 has designed a quadcopter system capable of nighttime operations and payload delivery in a small, compact package. The system consists of a custom built chassis that holds a FLIR camera on an adjustable mount and a motor driven clamp for payload delivery. This tool is optimal for Defense or Search and Rescue Applications where weight and low light vision are operational concerns.


## I. Introduction

Project Nightwatch is a drone outfitted with a custom built chassis that would allow for the utilization of a FLIR camera and a payload system. The FLIR video should be able to clearly detect infrared radiation and display it accordingly. This video should be able to work in low light areas. The FLIR video would be streamed to another device allowing it to be piloted from a small distance. This device can either be a phone or laptop. The camera would be attached to a motor allowing it to look down in preparation for payload release. The payload system involves a clamp attached to a motor where at the push of a button releases the payload. The payload is roughly the size of a small water bottle. It should be manually loaded before takeoff and remain attached until it is released by the pilot. This system will be controlled by a Raspberry Pi 4. The Pi will be able to receive a signal using a receiver/transmitter. The signals will be sent via push buttons attached to the drones controller. The goal of Project Nightwatch is to successfully deliver a payload using wireless streaming and communication. Skills from microcontrollers and coding classes were directly involved with the programming of the Pi. Electronics skills were used for setting up the receiver & transmitter. As well, knowledge of communication systems worked in conjunction with the video streaming.

## II. Problem Definition

We purchased the FLIR Lepton Breakout Board v2.0 in the hopes of easy integration. The 10-pin board was connected to the RPi 4 by SPI ports for image processing. With little knowledge of FLIR products or IR signal analysis in general, we scoured GitHub for code that allows us to display live video on a computer monitor via the pi. The team found a repository built by supplier GroupGets, from which we pulled necessary code to run. It resulted in a program that worked about 50% of the time. When working it displayed a small window of live infrared footage, otherwise it showed a small red box. Streaming was another issue: since the breakout board was connected via SPI and not USB, the pi recognized it as a peripheral not a proper camera. Unfortunately this meant we could not transmit video to a phone as designed. Thankfully the GroupGets vendor also sold another breakout board: the PureThermal 2, that connects by USB. This purchase worked on the first try, and allowed the team to work on other aspects of the project.

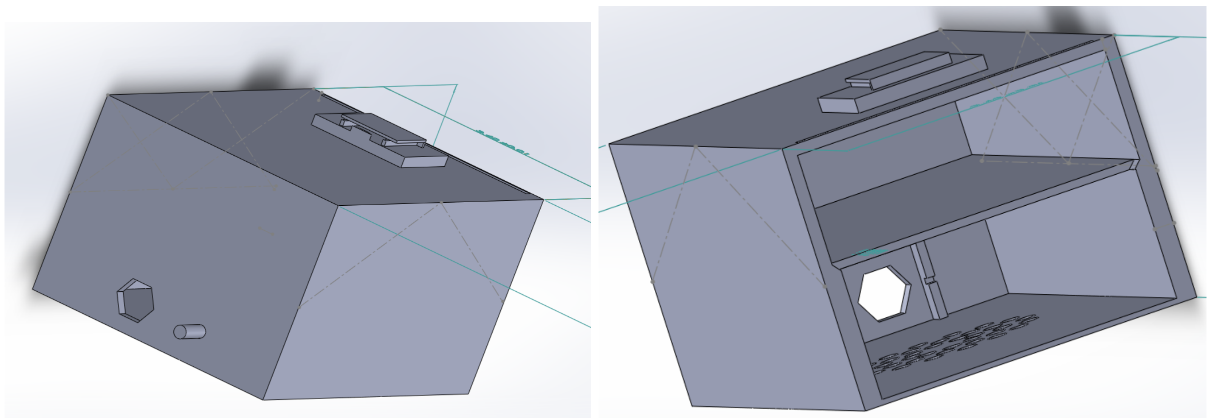

Figure 1. SolidWorks view initial chassis design

## III. Final Design

### *A. Drone*

Team 5 chose to use the Holystone HS700D GPS Drone & Controller for its relatively low price of ~$250 and availability at local superstores like Walmart. The HolyStone Drone is a quadcopter powered by a rechargeable 2800mAh battery that can provide about 22 minutes of total flight time and maximum distance of 3280 ft/900m. Flight is driven by four 1500kV brushless motors that are connected to its four removable blades. Its stands are also removable for easier packaging and transportation. The HS700D model is capable of autonomous flight with its internal GPS module; however, the team chose to ignore this feature in favor of manned flight. For anyone interested in the GPS, it can be accessed by the HolyStone Drone App. The HS700D also comes with a 4K Full High Definition 90° adjustable camera with 5G transmission straight to your phone. We also chose to scrap the provided camera to make room for our own FLIR mount. The entire drone system, excluding the controller, weighs about 630g and measures at 434 x 434 x 151 mm withouts its propeller guards. Its controller is powered by four AA batteries and consists of a power switch, an elevator control rod, and a throttle control rod. It also has an LCD display that shows battery life and GPS signal strength. There are additional buttons for system lock-out, photo mode, and a return to home feature that we chose not to use.  [1]

### *B. Custom Hardware*

Our custom hardware assembly is driven by a *Raspberry Pi 4* that is powered by a *Metecsmart 5V 5000mAh battery* via USB. For image processing, we used the *PureThermal 2 Breakout Board*, which connects to the Pi via a micro-USB to USB cable. As seen in Figure 2, it is a small package measuring at 22 x 30 mm. It has a 48-pin STM32F412 ARM microprocessor that executes all image processing without the need for an external system and additional 0.1" holes for UART, I2C, and GPIO extensions [4].

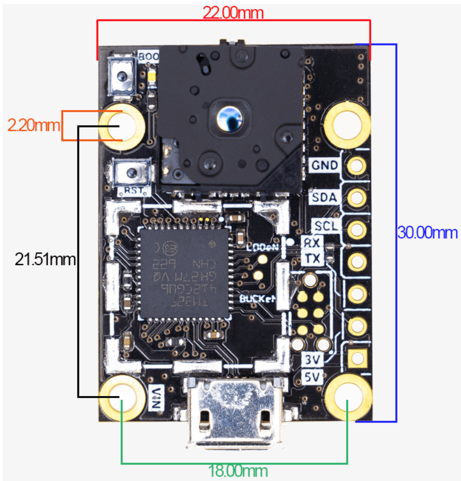


Figure 2. PureThermal 2 Breakout Board

Clicked inside the PureThermal 2 lies the *FLIR Lepton 3.5 camera*. Lepton 3.5 is a 32 pin package that measures at 11.8 x 12.7 x 7.2 mm, weighs 0.9g, and provides a resolution of 160x120 pixels. It operates at an average frame rate of 8.6 Hz and in the longwave infrared range of $8 - 14\mu m$. FLIR has two temperature modes: high-gain which is designed for lower temperature ranges, and low gain for higher temperature. Their 3.5 model can detect from -10°C to 140°C in high gain with an accuracy of +/- 5% and up to 450°C in low gain with an accuracy of +/- 10%. FLIR Lepton uses an **uncooled VOx microbolometer** as its thermal imaging detector. A bolometer is a digital thermometer whose resistance changes with temperature [6]. Figure 3 shows that a series of these components are arranged into a focal point array; each microbolometer "pixel" consists of a thin IR absorbing membrane called the bridge. The bridge is made of Vanadium Oxide (VOx), which is notable for its high temperature coefficient of resistance (TCR). It is suspended above a CMOS based readout integrated circuit (ROIC) connected by two narrow legs. The ROIC sends a current pulse at a constant framerate of 30 or 60 Hz to sense a change in the bridge's resistance [3].

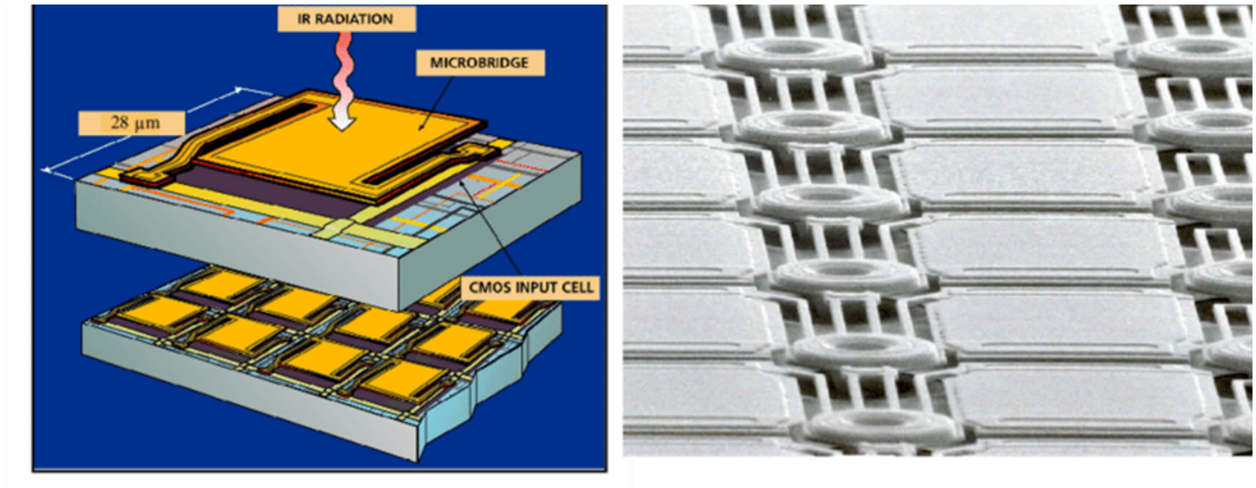


Figure 3. uncooled VOx microbolometer

Three prevailing materials used in microbolometers today: Vanadium Oxide (VOx), Amorphous Silicon (a-Si), and Barium Strontium Titanate (BST) [6]. They can be rated by their Noise Equivalent Temperature Difference (NETD) which specifies the amount of radiation required to produce an output signal. When comparing the NETD values of different cameras, FLIR recommends accounting for the f-number, or focal ratio, that expresses the diameter of the entrance pupil in terms of the effective focal length of its lens. Calculated with an f-number of 1, BST has a NETD of 0.1K while VOx has a NETD of 0.039K at a temperature of 25°C. The second standard of uncooled thermal cameras is determined by its **Johnson noise**, which can be expressed by the following equation:

$$E = V_{rms}\sqrt{4kTR\Delta f}$$

where

$$E = RMS\,Voltage\,Level$$

$$k = Boltzmann's\,constant\,(1.30 \cdot 10^{23})$$

$$T = temperature\,in\,Kelvin\,(Room\,temp\,300K)$$

$$R = Resistance$$

$$\Delta f = Circuit\,bandwidth\,in\,Hz$$

VOx detectors have an impedance of about $100k\Omega$ while a-Si detectors typically have a much higher impedance of $30M\Omega$. Assuming all other parameters are the same, the VOx based microbolometer will have a much lower Johnson noise level compared to its a-Si counterpart. With better performance compared to its peers, the VOx detectors currently make up about 70% of the uncooled detector market share [6]. While the camera assembly is connected by USB, we used the RPi 4 pinouts to connect the rest of the hardware components.

To control camera position and the payload clamps we used a *QIACHIP 433.92mHz RF Transmitter/Receiver*. The transmitter can operate anywhere from 3 to 24V and has a transmitting distance of 100m. RF signals are sent using Amplitude Shift Keying modulation [QIACHIP]. It is soldered onto an Adafruit Perma-proto board along with three momentary push buttons. This custom board is powered by a 3.3V and GND connection inside the HolyStone HS700D controller, as seein in Figure 4.

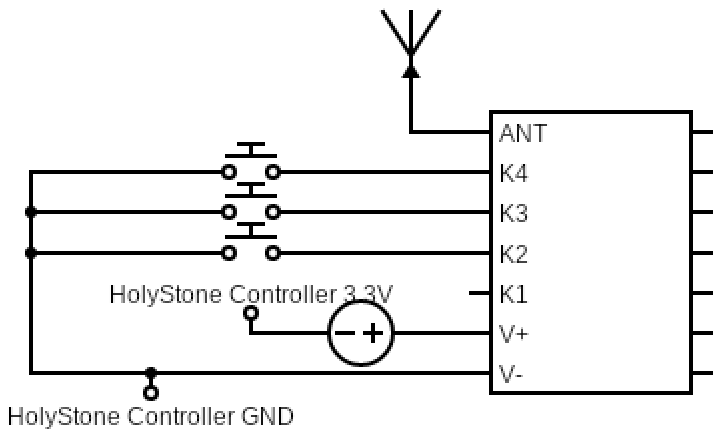


Figure 4. RF Transmitter Wiring

QIACHIP's receiver operates at voltage of 3.3-5V and has a receiving rate of 10KB/s [QIACHIP]. Figure 5 shows the receiver is connected by the following RPi pins: D0 to pin 16 (GPIO 23), D1 to pin 18 (GPIO 24), and D3 to pin 22 (GPIO 25). We used two *smraza micro servo 9g* motors, one for the clamp, and one to control the camera angle. Servo one is connected to pin 11 (GPIO 17) that sends a PWM signal to open and close the clamp. Servo two is connected to pin 15 (GPIO 27), whose PWM signal can rotate the camera angle up or down 90°.

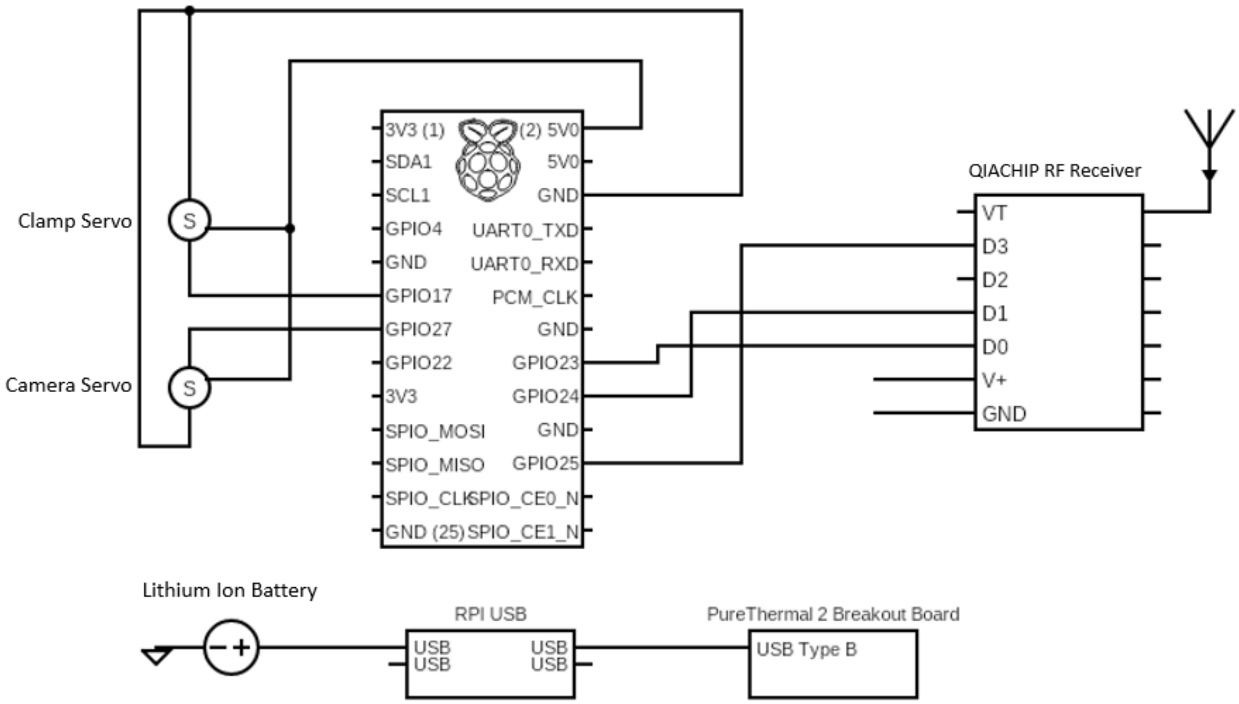


Figure 5. Raspberry Pi 4 Wiring

To house all of the components of the system, the chassis was designed using SolidWorks. As stated prior, the initial iteration caused a weight issue due its bulky design that prevented the drone from taking off. To address this issue, a new chassis was designed from the ground up. While retaining the same 9 cm width of its previous iteration, the height of the chassis was reduced from 7 cm to 4.318 cm. In addition, a hexagon lattice pattern was applied to the walls, top and bottom of the chassis, leading to a reduction of weight. When printing the new design, the infill setting used was reduced from 25% to 10-15% to further reduce weight as well as print time. The weight of the chassis's final design was reduced from 160g to 100g (Figure 6 & 7).

Because of the hexagonal lattice design, this allowed the original landing gear that came packaged with the drone to be attached to the bottom of the chassis, reducing the need of using the custom landing gear. A hook was created on the bottom of the interior to allow the camera mount to be easily inserted and secured. In addition, a pair of claws was designed for the payload system, one of the gears being larger than the other (Figure 8 & 9). A plate was also designed to hold the FLIR camera to the camera mount (Figure 10).

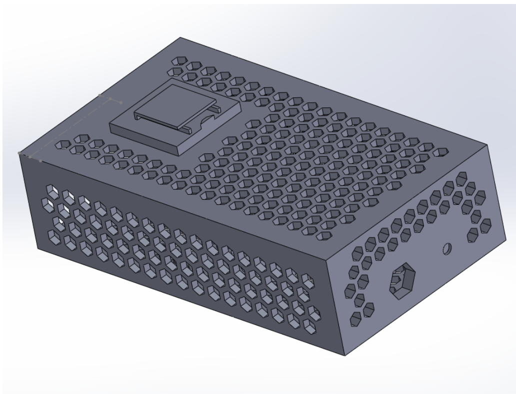
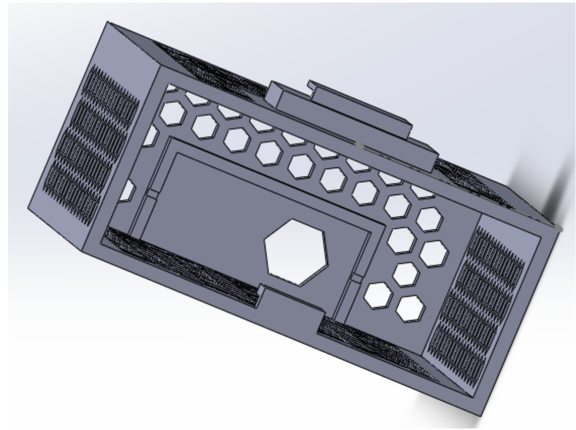

Figure 6 & 7. SolidWorks view of the new chassis

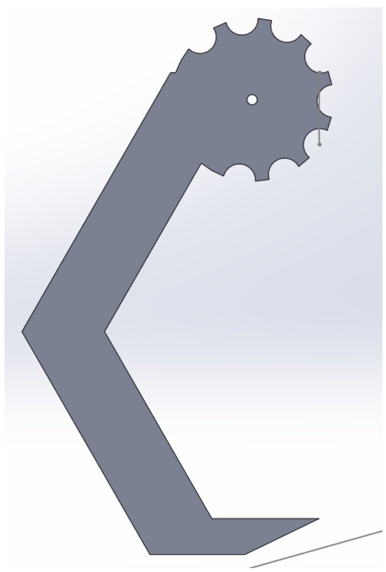
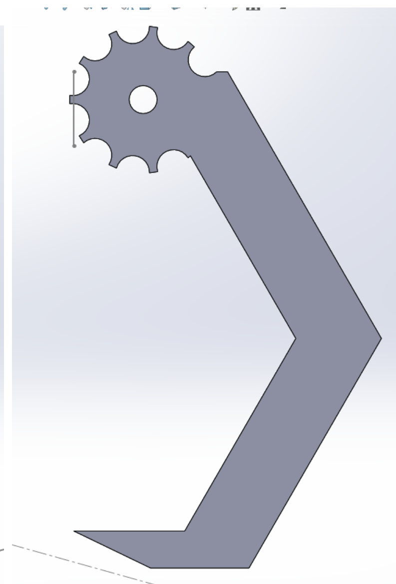

Figure 8 & 9. SolidWorks view of clamps

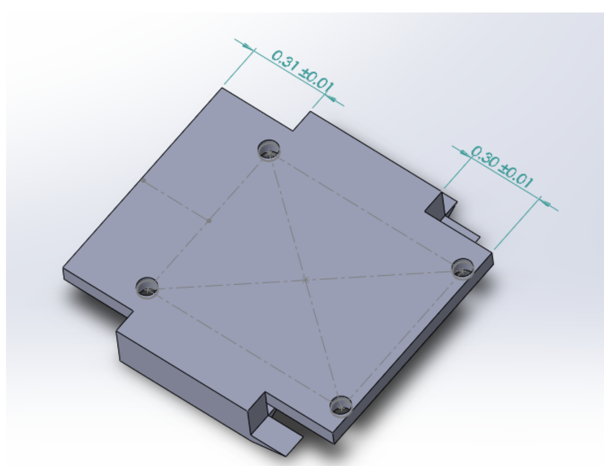


Figure 10. SolidWorks view of plate for FLIR Camera

## C. Software

The software requirements for this project are as follows: streaming the FLIR video to an outside devic, control the camera's servo motor, and operate the clamp servo motor. In order to complete these tasks, two python programs were created on the Raspberry Pi 4. Python was used because it was the simplest language to use on a Pi and there are many useful packages that can be installed. Originally, an Android application was going to be made to work in conjunction with these programs. However, it became clear that just using these programs was the simplest option while still meeting the basic requirements.

*1.)Streaming.py:* The first program of the project involves streaming the FLIR video wirelessly. Given the difficulties of wireless communication, this was the first task focused on. The first few designs of this program involved using an external transmitter to stream video. Though, through researching multiple methods, it was discovered that Raspberry Pi 4 has WiFi and bluetooth capabilities. It was in this research that the package "pyshine" came to light. This Python package allows for low latency video streaming of USB webcam video. The package "fswebcam" was needed so that the Pi can use video from a USB webcam. By using these packages along with Opencv, video can be streamed to a website using the URL: "http:// [Rasberry_Pi_4_IP]:[Given_Port]". The website is created with simple HTML code provided in the Python program.

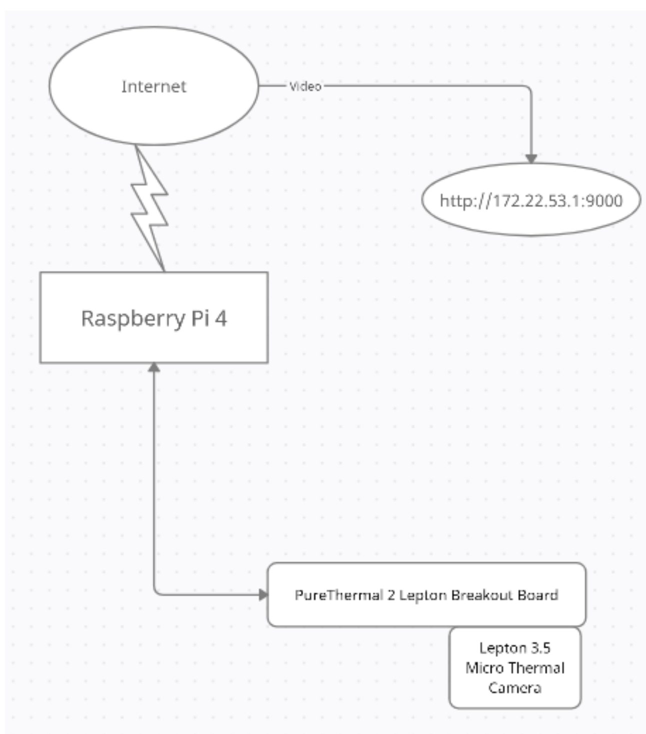


Figure 11. Streaming.py Flowchart

The figure above shows the flow of video when Streaming.py is running. All communication is over WiFi and is hosted by the Pi itself. The website can be accessed by any device with an internet browser (phones, laptops, etc.). An example of how the video looks can be seen in figure 12. Since the video can be accessed easily by multiple devices and wirelessly, it took away the need for an application. With the current program, all the application would do is open a URL. Which is redundant since most devices come with a browser installed. Also, this allows for any device to open the website since not all devices can use Android applications.


Figure 12. FLIR Video being Streamed

*2).Nightwatch.py*: The second program used for this project controls the servo motors for the camera and clamp. This program is a far simpler Python program using the GPIO pins on the Raspberry Pi. The basic function of the program is that it waits till a signal is received from the 433 MHZ wireless Receiver and signals the corresponding motor; this is all done using a while-loop and if-statements. The flowchart for this program can be seen below in Figure 13.

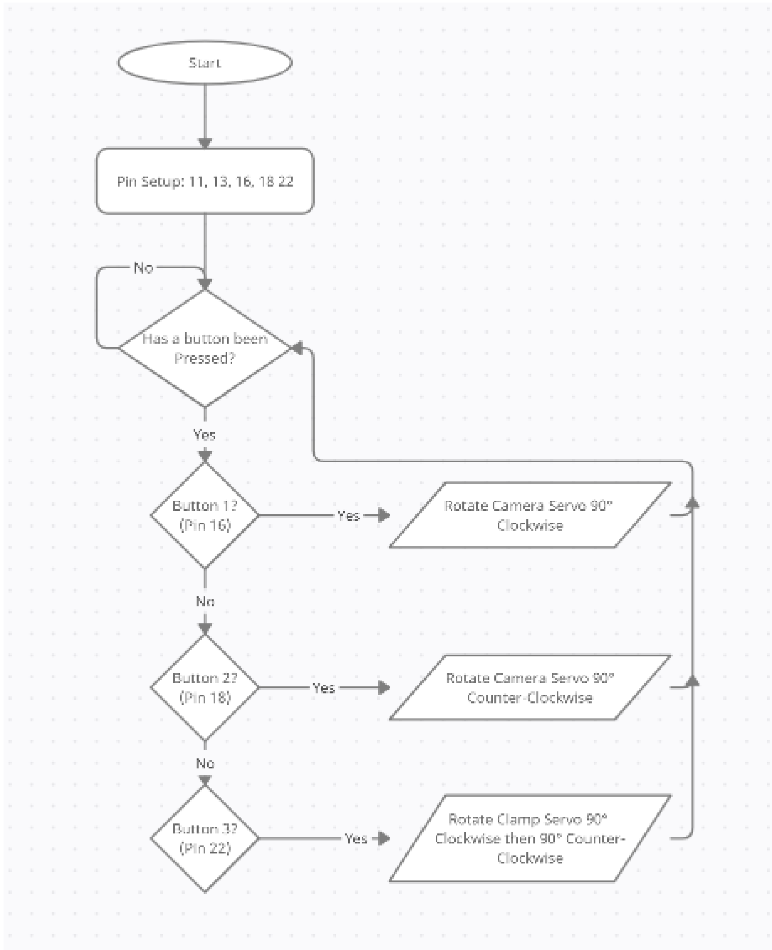


Figure 13. Nightwatch.py Flowchart

The flowchart in figure 13 shows what happens in each iteration of the while-loop. When a button is pressed, it sends a signal to the Pi to drive the motor at the designated duty cycle. For the Smraza Micro Servo 9g S51, 5.5 rotates 90° clockwise and 10.5 rotates 90° counter-clockwise. The list of pins used in the program are as follows:

- 11: PWM for Clamp servo
- 13: PWM for Camera Servo
- 16: Button 1; Camera Looks Forward
- 18: Button 2; Camera Looks Down
- 22: Button 3; Clamp opens then closes

The simplicity of Nightwatch.py is another reason why an Android Application was not used. Since the only signals being communicated were button presses. It was easier and faster to solder push buttons and use a basic receiver/transmitter. Additionally, these buttons could be attached to the controller making it easier for the pilot to use them. Overall, this program does complete the required task of motor operation via wireless communication because it works independently from Streaming.py. This is to lessen the chance of stream interference and allows for modulation of the system.

## IV. Performance Characterization

### *A. Weight*

Weight was a very big issue for this project and the bulk of the quantifiable data. Every component that was used had to fit a specific weight limit. For the Holystone, the weight capacity is roughly 415g. This number was gathered by doing multiple weight tests on the drone and seeing if it can take off.

Table 1: Mass of Components

| **Component** | **Mass (g)** | **Used** |
|---|---|---|
| FLIR Camera & Mount | ~34 | Yes |
| Micro Servo | ~19 | Yes |
| Parallax Feedback 360 iirc | ~40 | No |
| Raspberry Pi 4 (no case) | ~90 | Yes |
| Raspberry Pi Zero (case) | ~75 | No |
| 12 V 6Ah powerbank | ~371 | No |
| mlady 5000mAh powerbank | ~101 | Yes |
| 443 MHZ Receiver + breadboard | ~50 | No |
| 443 MHZ Receiver soldered breakout | ~30 | Yes |
| First Housing | ~166 | No |
| Second Housing | ~100 | Yes |
| Stock Drone Legs | ~19 | Yes |
| Printed Legs 25% infill | ~93 | No |
| Printed Legs 15% infill | ~73 | No |

Table 1 shows the components we weighed and if it was used in the final design. Most of the components not used were because they weighed too much. For example, the first battery pack used was the 12V one. The 5V battery pack could easily power the Pi just as well and weighed significantly less. Therefore, the 5V battery pack was used in the final implementation. Almost every decision followed this same pattern of picking the lighter option except for the Pi. Every other decision was between components that functioned the same but the Pi's were very different. At the time of deciding, all the programming was done on the Pi 4 and not the Zero. This led to the Pi 4 being favored. Also, the Zero did not have a USB port for the camera, rendering it unable to meet one of the requirements. After assembly, the entire system weighed ~364g. This means that the drone would be able to comfortably lift a payload of 10g-15g.

### *B. Size*

While taking measurements during the design phase of the chassis, we were focused on having enough interior space to fit all of the components while maintaining a compact form factor. While this goal was achieved, it did bring up an issue regarding weight distribution. Only the micro-servo motor has a dedicated slot for within the chassis, leaving the rest of the components loose. This poses the potential risk of components moving around, causing the weight to shift towards one side. Another possible cause would be the weight of the system would already be shifted due to the positioning of said components.

### *C. RF Transmitter & Receiver*

QIACHIP advertise their RF Transmitter and Receiver as operating at 433mHz. We used a lab provided 35M-4400M Spectrum Analyzer to determine its accuracy. As Figure 14 shows, when the buttons are pushed a 433.50mHz signal is transmitted. Occasionally the signal spiked to 500mHz, but it was rare.

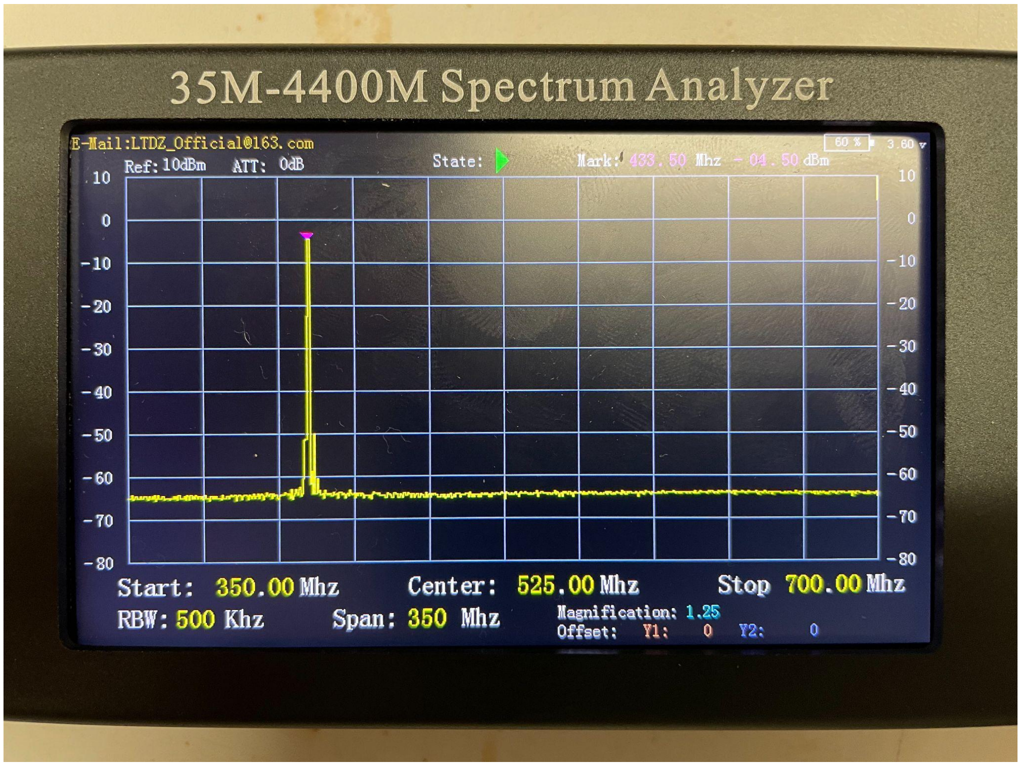


Figure 14. QIACHIP RF Transmitter Signal

## V. Cost Accounting

At conception, Team 5 projected a total cost of approximately $1500 for project completion. To date, Project Nightwatch has a total development cost of $815.02 for which $423.98 was reimbursed for parts. Development cost includes all material purchased for the project, whether they were used or not and the Clemson University Department of Electrical and Computer Engineering was kind enough to reimburse us for some of our purchases. A total of $305.06 was donated to the team by Clemson University's Dr. Raza, which includes the Raspberry Pi 4, all batteries and motors, and the PCB boards. Expected cost of the final artifact is $681.44, which excludes the Talentcell Rechargeable battery and FLIR Lepton Breakout Board v2.0. All cost calculations ignore minor costs like the screws used and University equipment which includes the 3-D Printer and filament. Currently, this project stands at $778.57 under budget.

## VI. Postmortem

### *A. Technical*

*1.)Drone:*The most important piece of this project had the most issues. Our current drone lacks the power to lift our payload assembly resulting in the team scrambling the cut weight. The Drone itself is easy to operate and fly but wasn't designed to carry much more than a small camera. Due to the limited power our project is limited to a very low altitude of about one meter. The small power output and small propeller size has resulted in barely any vertical lift. We spent most of our time trying to find a new drone after our initial drone failed to fly. In purchasing this new drone, we failed to consider weight requirements. Given the opportunity to review this project it was the correct decision to look for a second drone because our first drone was not designed to hover and was challenging to pilot. In the search for a second drone more time should have been spent in researching a drone that would be capable of effectively lifting the weight required for this project.

*2.)Hardware:*The hardware portion of this project has been one of the stronger portions of this project due to its reliability and flexibility. Our biggest issues with hardware in this project has been due to our drone's lack of power, requiring the change to various components due to weight issues. One of the small issues we've had with hardware for this project is three push buttons used for camera tilt and payload delivery is unreliable. This is due to the lack of a pull up/down resistor in our design. The only real constraint for the hardware in this project was weight and image processing. We chose to use a Raspberry Pi 4 for its processing power and ease of use. Implementing hardware and hardware changes never seemed to be an issue; most of our hardware was "plug and play".

*3.)Software:* The software for this project works well: all programs run smoothly without any known bugs or delays. Our final program is simple in design and implementation; it does not require much memory or processing power. All required tasks are completed and can work in conjunction with the hardware. We made each program work independently for easier implementation and easier individual component testing. The only drawback is each program needs to be manually run before assembly, it lacks an automatic startup. The startup script was going to be made but wasn't due to time constraints. The streaming program can be inconsistent and cut streaming because it operates over a local wifi network. Since the streaming platform was one of our last tasks, we chose wifi over other communication methods because it was the simplest method. Other than a few hiccups, Nightwatch.py works exactly how it is supposed to and completes the required task. The Software for this project was made to be simple in implementation and functionality which has resulted in limited capability.

### *B. Nontechinal*

All nontechnical issues were handled well by the group. Tasks were evenly distributed throughout the group and everyone knew what tasks they were assigned.  There weren't many scheduling issues as every member of the group had time to work.  However, availability of each member wasn't completely aligned.  Either way, this never posed a problem as most of the tasks were independent from another in the beginning.  Near the end of the project, every group member's availability became more open given the precedence of the final.

One large nontechnical issue that caused problems was ordering components. For the few starting weeks, all work was put to a stand still as desired components were needed. Granted that research and preparation were still being done, it put a bit of strain on the project. Another issue was having to reorder components when issues arose with the existing components. Further research and testing could have prevented this. The time management of the group could have been better if components were decided on/delivered sooner.

The project management of the group suffered from the previous issue. Due to a late start with components, a lot of the design process was rushed. Examples of this are neglecting to test the drone's weight limit and underestimating the difficulty of setting up the FLIR to stream. These issues negatively impacted the final design. However, the group still handled them well by putting in extra work hours to ensure a final design could be made to meet all the requirements.

### C. *Future Direction*

*1.)Advance Streaming/Video:* Given how simplistic the streaming is for this project, the obvious direction would be to work on the video processing. Rather than relying on WiFi, a new transmitter/receiver could be made to improve video quality and latency. As well, motion detection could be added on top of the stream with the use of video processing techniques. This would add more functionality to the system. If machine learning were to be implemented, it could even be able to identify people based on shape & temperature.

*2.)Delivery:* An implementation direction for the project could be delivery services. Since the drone is outfitted with a camera and a payload system, it lends itself nicely to this sort of work. However, a GPS and AI Autopilot system would be needed. The GPS would allow the drone to know where it is going and the AI gets rid of the need for a pilot. Machine learning could even be used to help identify people/recipients so that high-value packages won't be delivered incorrectly. If this direction is pursued, then a stronger drone would need to be used as the current drone can only carry ~400g.

*3.)Defense:* Combining the first two directions would lead to a more defense oriented direction. Using advanced streaming/video processing and a better delivery system, a drone could be outfitted to assist in frontline combat. Drones are far cheaper to produce than most military hardware and take the place of a soldier. The FLIR camera would be ideal for picking out heat signatures no matter the time of day. Complications arise in how effective it would be. Drones produce a lot of noise and are not agile; this would make them an easy target. If this direction is pursued, sound reduction would need to be implemented to the drone while also making it more agile.

## VII. Acquiring New Knowledge

i. Andrew Burrell

1. I have learned wireless interfacing, Drone design, and Thermal imaging techniques this semester.
2. My strategy to learn wireless interfacing was to use the knowledge from my 3810 class. My strategy to learn drone design was to use the research during this project along with the drone I built previously. My strategy to learn thermal imaging techniques was the research to understand FLIR and its founding principles.

ii. Kyle Jenko

1. I have learned Python basics, Thermal Imaging Electronics, and Soldering this semester
2. My strategy to learn Python has been following online tutorials and GPIO programming. My strategy to learn Thermal Imaging Electronics has been relying on my class ECE 4340 Optoelectronics & Photonics. My strategy for learning Soldering has been working experience in the Spark Space.

iii. Brandon Kim

1. I have learned Python basics,  Video Streaming, and GPIO pin programming
2. My strategy for learning Python basics relied heavily on my prior knowledge of C and another class I was taking this semester, ECE-4420 Knowledge Engineering . Understanding Video Streaming was rather difficult but manageable through hefty research . GPIO pin programming was far easier than I expected. Much like Video Streaming, my strategy was to research and implement

iv. Humberto Ruiz

1. I have leaned CAD design, product design, and optimization of 3D printing this semester
2. My strategy to learn computer aided design was done through the use of SolidWorks and relied on trial and error. I had very little experience on using the software, though I was able to acquire some knowledge on how to use it by researching tutorials online. Product design was fairly easy to grasp, as I was tasked to design a product (i.e. chassis) that would fit a series of criteria. While I was fairly experienced in 3D printing, I learned on how to optimize these prints to ensure they do not weigh down said product.

## VIII. Technical Standards Used

IEEE C 95.2-2018 Standard for Radio-Frequency Energy and Current-Flow Symbols

OSHA 1910 Subpart S - Electrical

ASME Y14. 46 Product Definition for Additive Manufacturing

FAA 14 CFR §107 Small Unmanned Aircraft Systems

## References


[1] "HS700D GPS drone with 4K camera," *Holy Stone*, 2022. [Online]. Available: https://www.holystone.com/en/Drones/Premium/HolyStoneHS700D.html. [Accessed: 30-Nov-2022].

[2] "Lepton LWIR Micro Thermal Camera Module," *Teledyne FLIR*. [Online]. Available: https://www.flir.com/products/lepton/?model=500-0771-01&vertical=lwir&segment=oem. [Accessed: 30-Nov-2022].

[3] M. Kohin and N. R. Butler, "Performance limits of uncooled VOx microbolometer focal plane arrays," *SPIE Proceedings*, 2004.

[4] "PureThermal 2 - Flir Lepton Smart I/O module by GetLab," *GroupGets*, 2022. [Online]. Available: https://groupgets.com/manufacturers/getlab/products/purethermal-2-flir-lepton-smart-i-o-module. [Accessed: 30-Nov-2022].

[5] "QIACHIP RX480E TX118SA-4 433.92mhz wireless wide voltage coding transmitter module and decoding receiver module 4CH RF EV1527 2262 Encoding Learning Module DIY kit," *QIACHIP*. [Online]. Available: https://qiachip.com/collections/diy-part-modules/products/qiachip-tx118sa-4-rx480e-kit-433-92mhz-wireless-wide-voltage-coding-transmitter-module-and-decoding-receiver-module-kit-4ch-rf-ev1527-encoding-learning-module-diy-receiver-kit. [Accessed: 30-Nov-2022].

[6] T. FLIR, *Uncooled Detectors for thermal imaging cameras*. Teledyne FLIR, Teteringen - Breda, Netherlands, 2020.

# Appendix A

| | Project Budget | Projected Price |
|---|---|---|
| ELECTRICAL AND COMPUTER ENGINEERING | $ 1,500.00 | $ 815.02 |

| Item Description | Part Number | Vendor | Quantity | Total Cost | Payment Type | Used in Final Project |
|---|---|---|---|---|---|---|
| HolyStone HS700D GPS Drone | | Walmart | 1 | $ 149.99 | RB | Y |
| FLIR Lepton 3.5 | 500-0771-01-ND | Digikey | 1 | $ 164.00 | RB | Y |
| PureThermal 2 Breakout Board | 2077-PURETHERMAL-2- | Digikey | 1 | $ 109.99 | RB | Y |
| FLIR Lepton® Breakout Board v2.0 | | Teledyne FLIR | 1 | $ 60.00 | RB | N |
| 3 Sets of 433mhz RF Transmitter and Receiver | TX118SA-4 & RX480-E4 | Amazon | 1 | $ 25.98 | RB | Y |
| Raspberry Pi 4 | | | 1 | $ 173.00 | D | Y |
| Adafruit Quartersized PCB 3 set | | Adadfruit | 1 | $ 8.50 | D | Y |
| Metecsmart 4500mAh Small Portable Phone Charger | | Amazon | 1 | $ 29.99 | D | Y |
| Talentcell Rechargeable 12V 6000mAh/5V 12000mAh DC Output Lithium ion Battery Pack | | Amazon | 1 | $ 39.99 | D | N |
| smraza micro servo 9g motor | SG90 | Amazon | 1 | $ 19.99 | D | Y |
| 1-inch screws | | | 4 | | D | Y |
| High Speec Continuous Rotation Servo | ADA3614 | ThePiHut | 1 | $ 33.59 | D | N |