



# Final Milestone

5/9/2022

Senior Software Design - Senior Capstone





# **Introduction to the Radio Telescope Project**

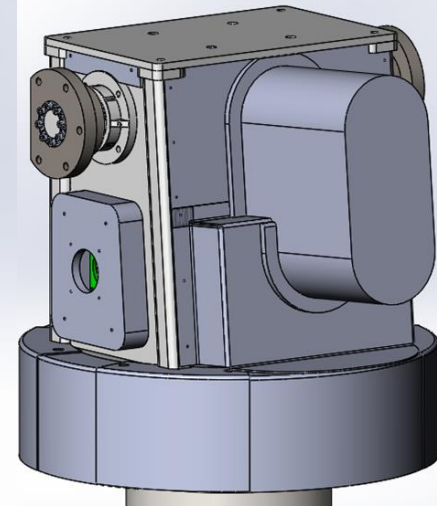
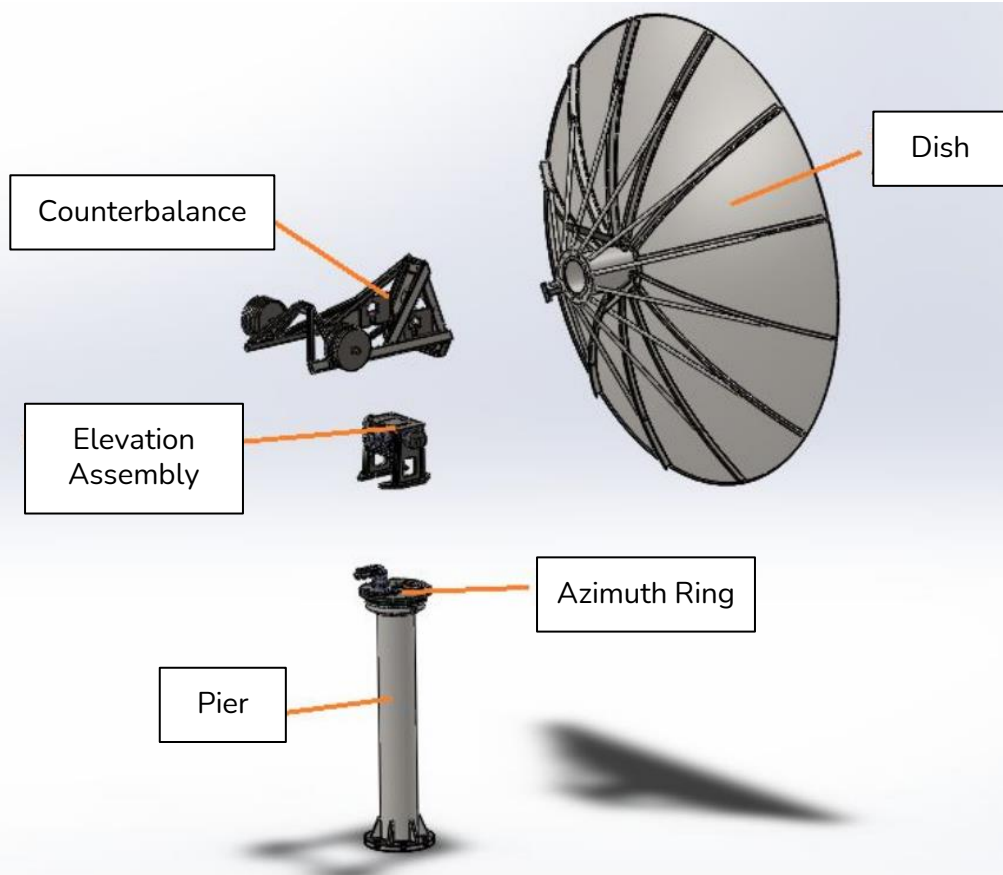


# What Is The Radio Telescope?

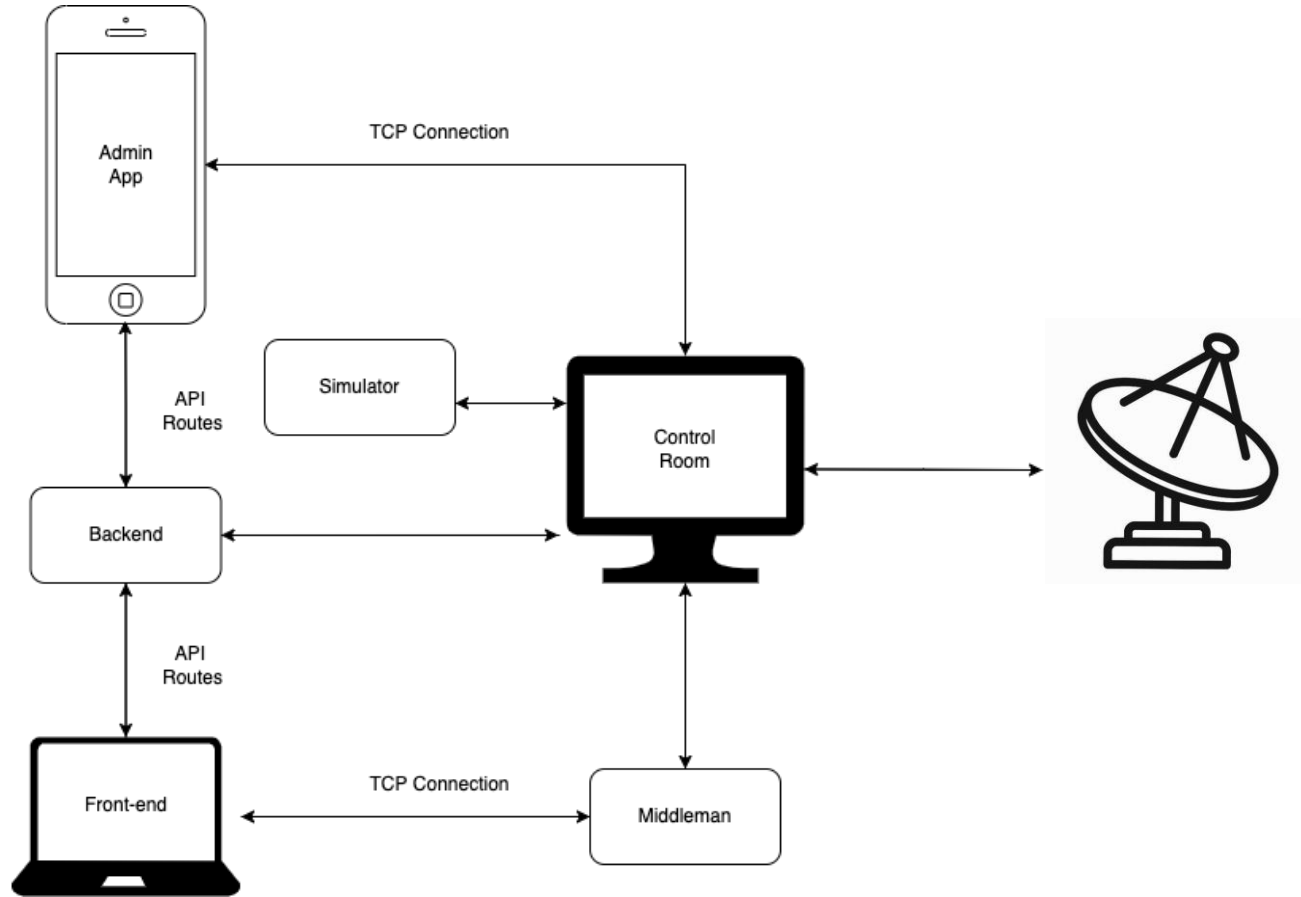
- 4.5m, remote controlled, auto-tracking, auto-locating Radio Telescope capable of scanning 1.42Ghz radio signals
- 4 years ago YCP was contracted to build a radio telescope for the York County Astronomical Society to be placed in John C. Rudy County Park, York
- The telescope is for educational and amateur astronomy research
- This has been an ongoing project, delayed by COVID-19, worked on by 77 students throughout 4 years
- The telescope is planned to be installed in the park



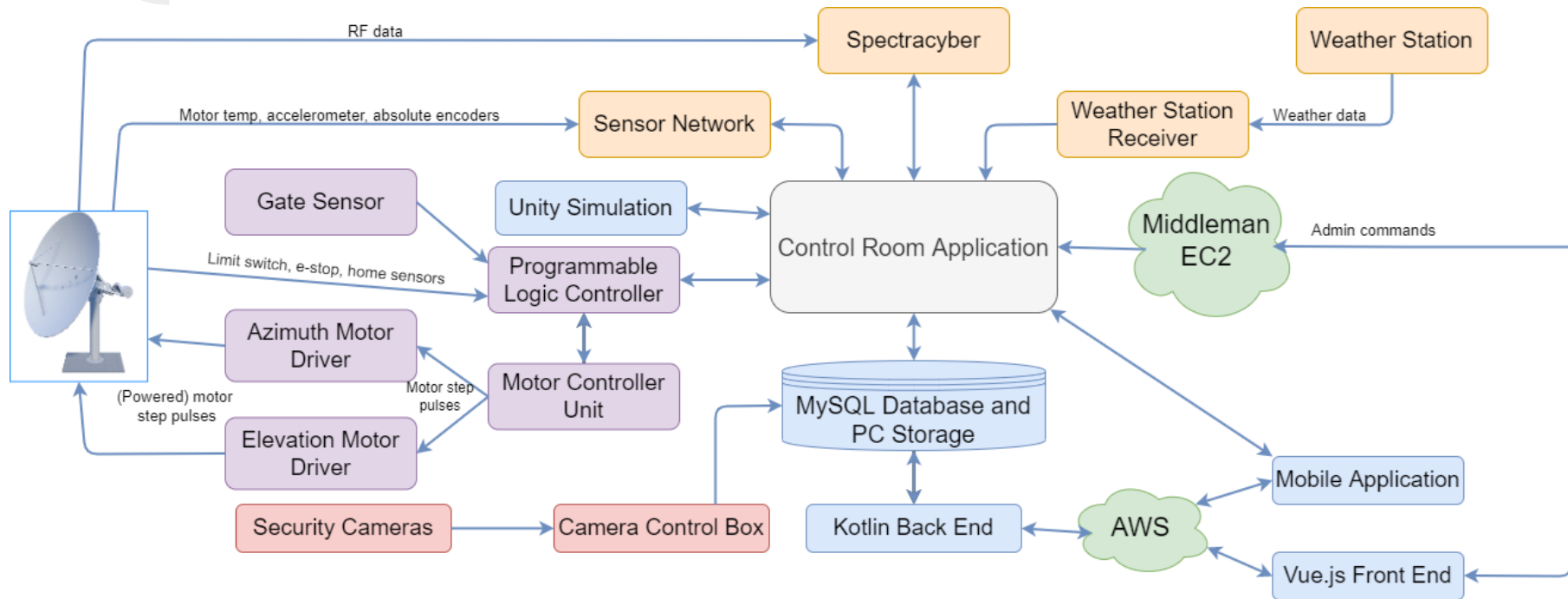
# Major Components



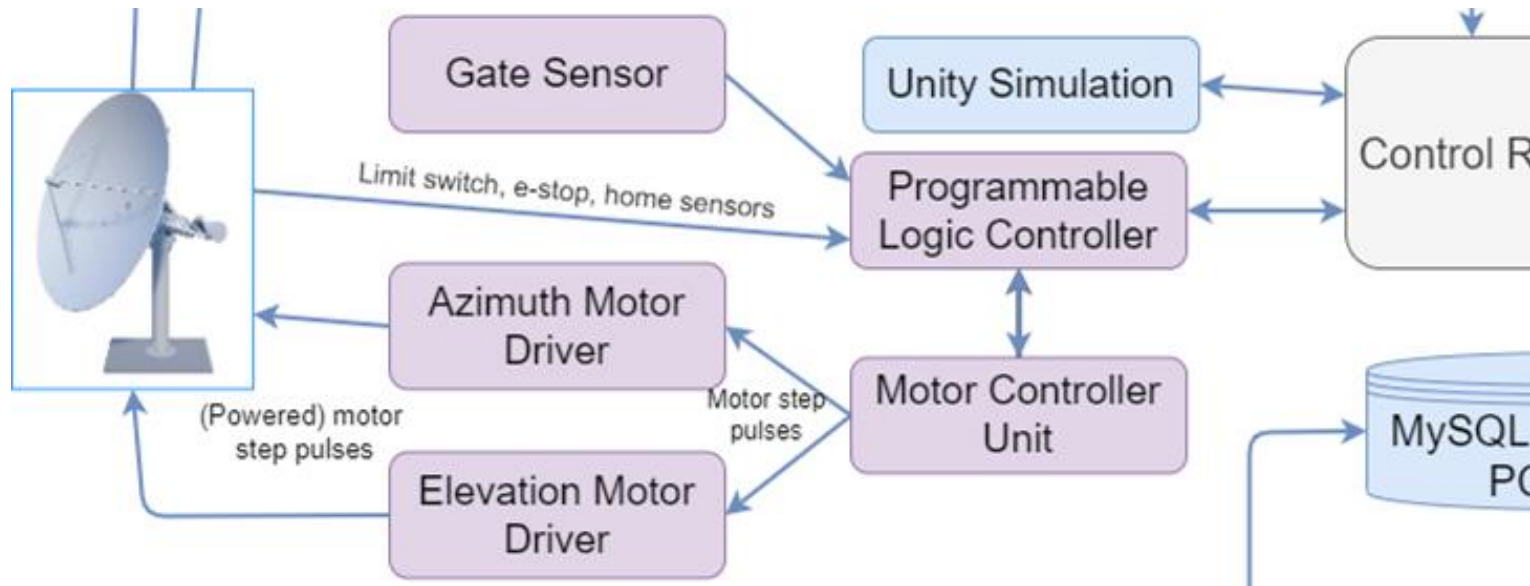
# Telescope Overview



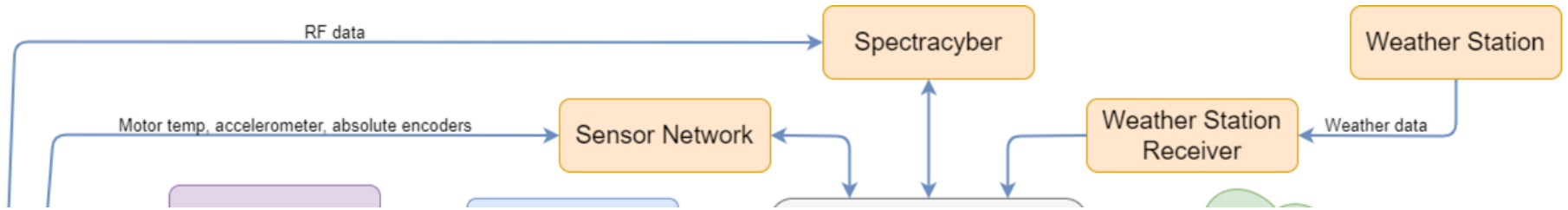
# Detailed Telescope Overview



# Detailed Telescope Overview - Control

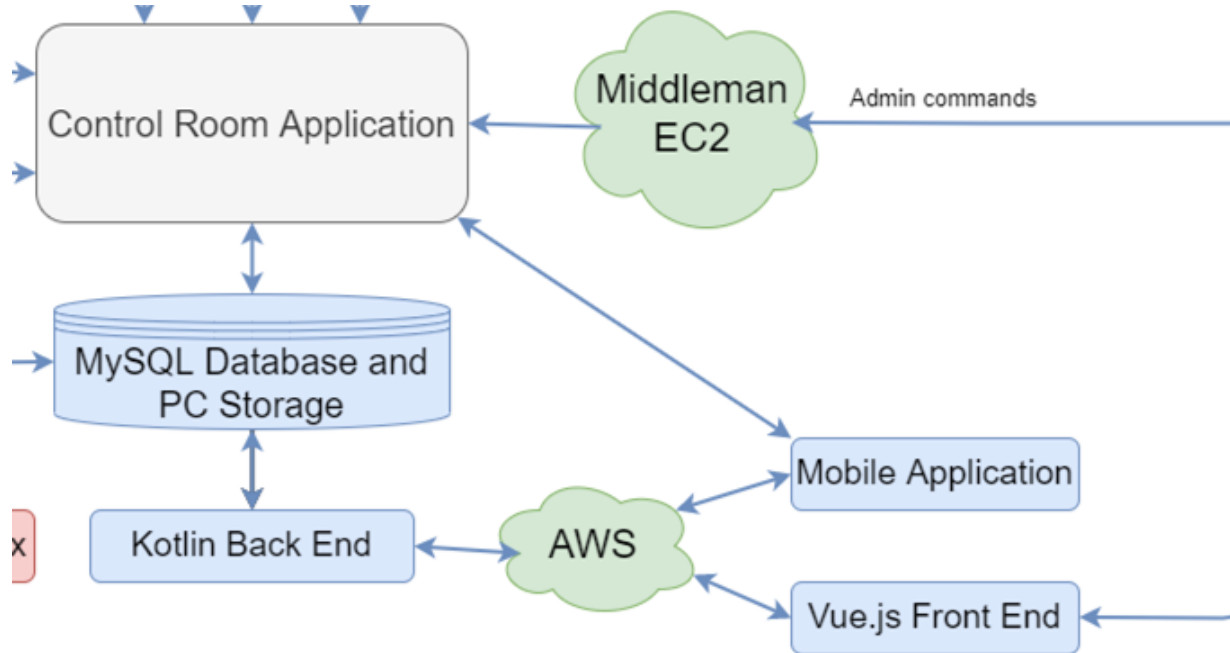


# Detailed Telescope Overview - Collection

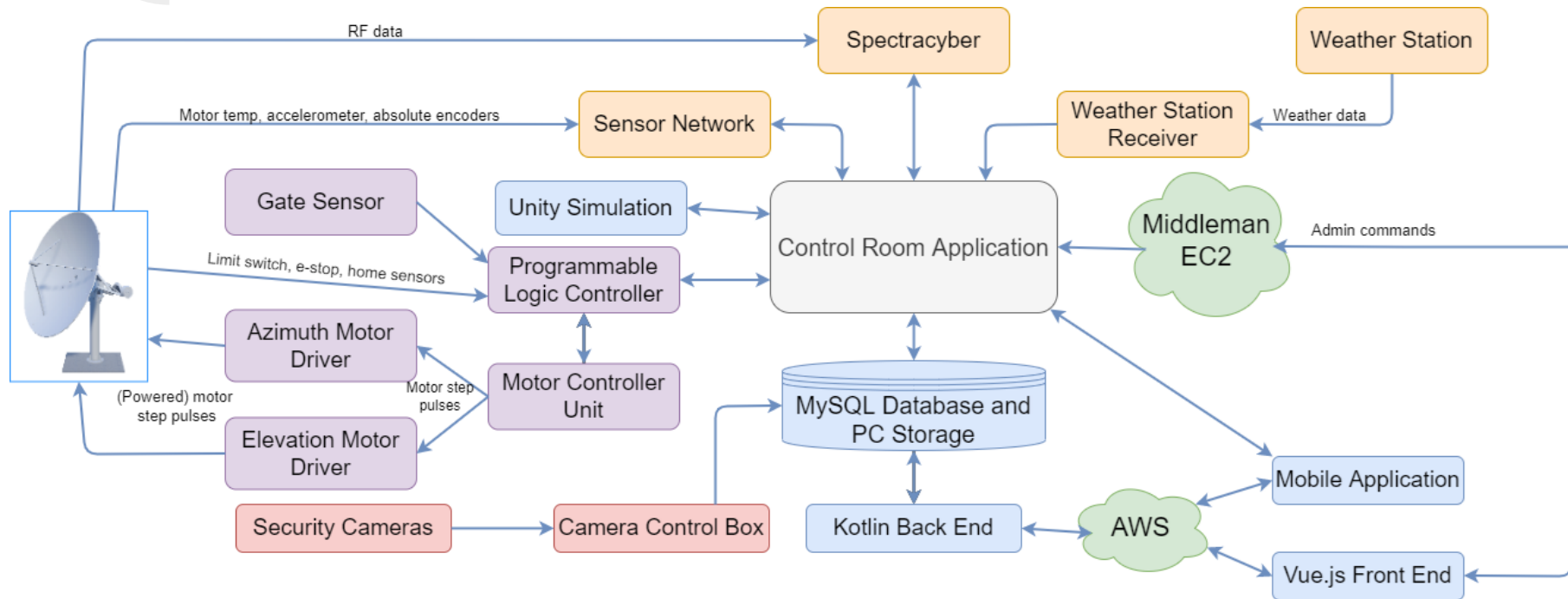




# Detailed Telescope Overview - Software



# Detailed Telescope Overview - Misc



# How will this be used?



- YCAS will have a website that the general public can create accounts and set appointments
- YCAS admins can use the control room software to monitor and move the telescope
- YCAS admins will have access to the mobile app allowing them to control the telescope remotely
- YCAS will have access to a VR game version of the park and telescope that can be shown off to students



**Where we started**

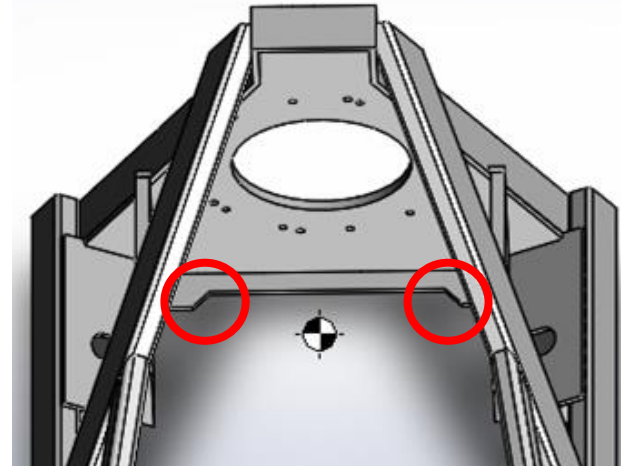
# The Hub and Counterbalance



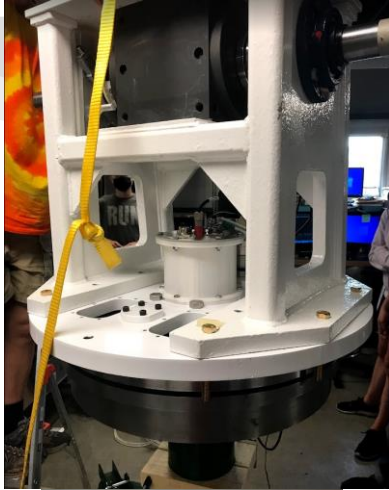
New  
Holes



Original  
Holes



# The Elevation Mount



## Elevation Mount Issue:

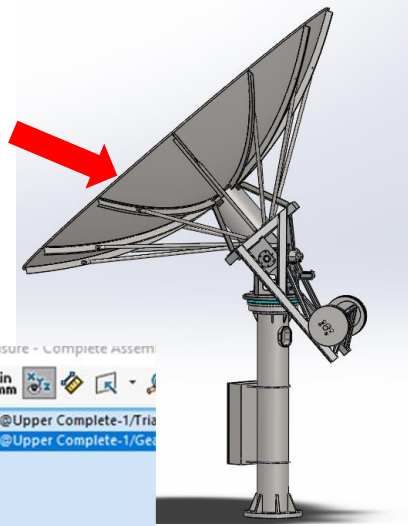
- Mating the elevation mount to azimuth ring due to misaligned bolt holes
  - Likely occurred in manufacturing

## Solution:

- Bore out elevation mount bolt holes to allow enough clearance to reach azimuth ring bolt holes.
- Size up washers to account for new hole size and safety

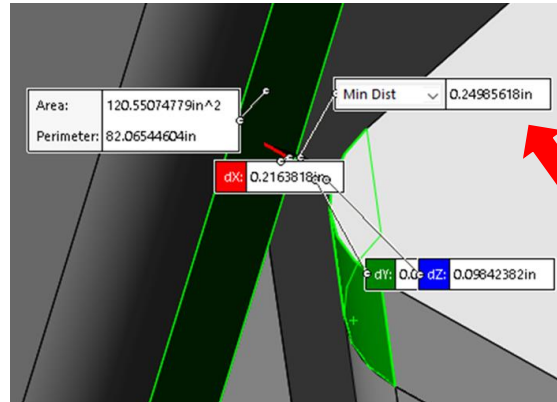
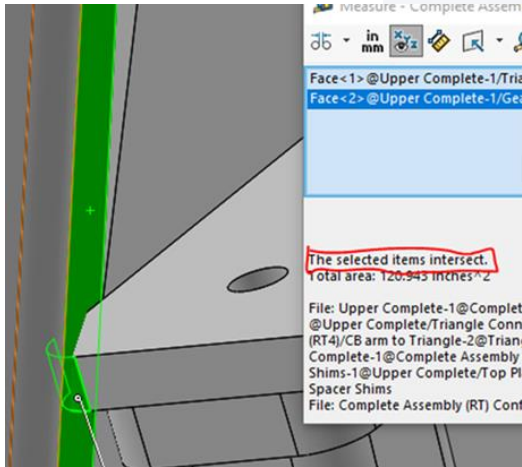
# Fixing issues (cont.)

Angle of telescope with largest intersection between counterbalance and the top plate



Top Plate Issue:

- Front corners of the top plate collide with counterbalance frame
  - The two parts intersect by about  $\frac{1}{4}$  inch from the corners



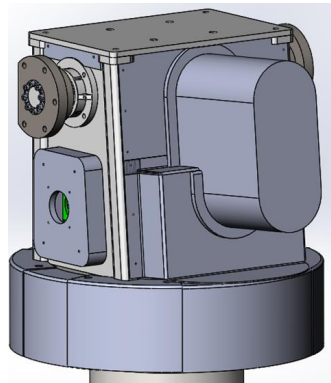
Gap Shown after making top plate flush with elevation mount

# Cover

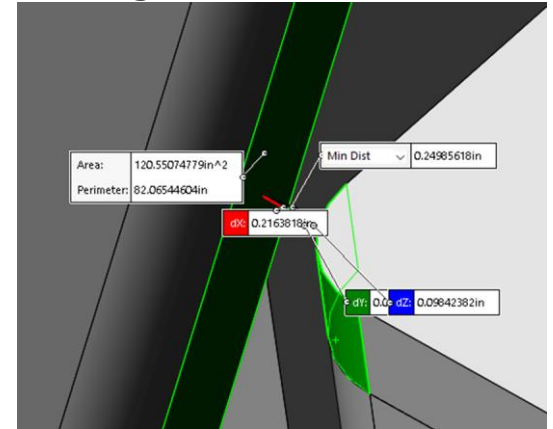
## Major Specifications the Needed to be Met:

- Maintain seal around elevation frame and azimuth ring
- Be lightweight
- Allow access for wiring to cross cover boundaries

Cover Design



## Design Clearance issues:





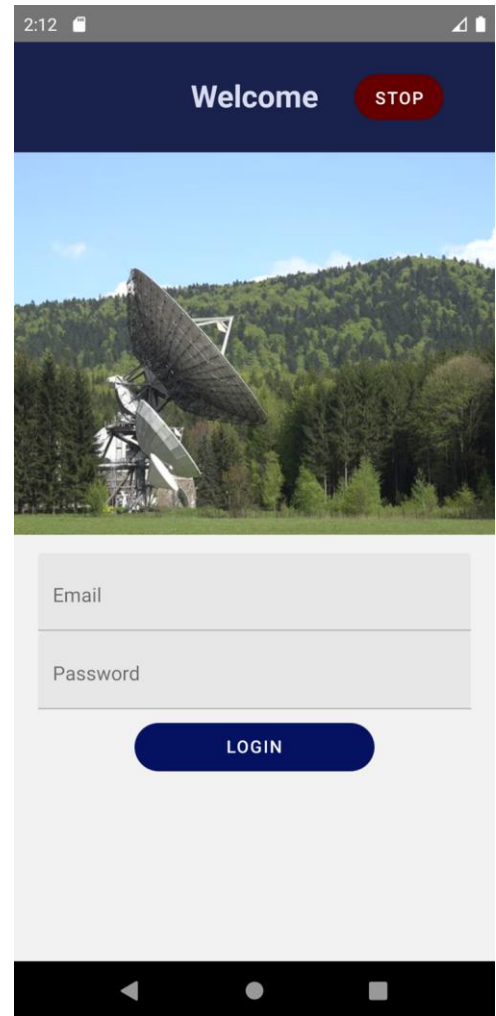


# Life of a Command



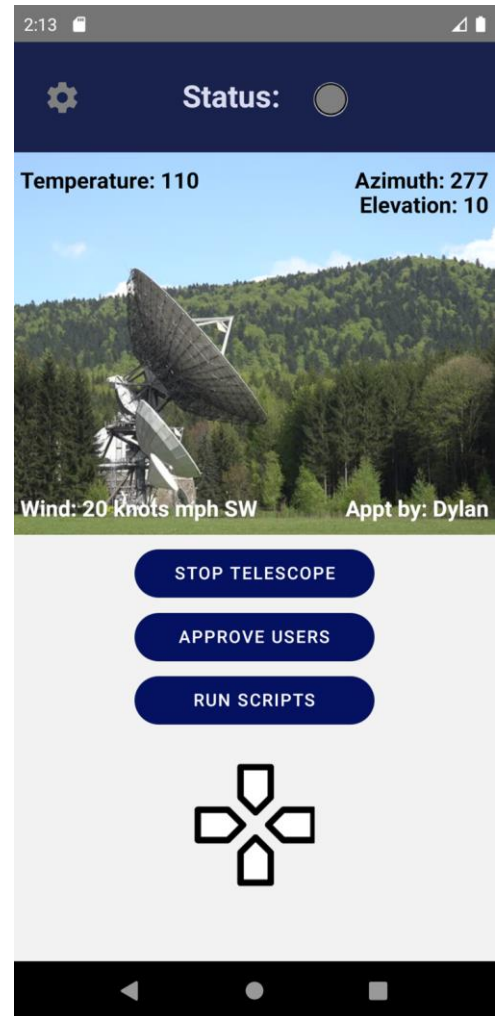
# Login

- Admins will be given a login to be used when installing the app
- Their information will be saved for future use
- This login will be used to access the APIs
- A stop button is added for safety
- After login the admin will be directed to the home page



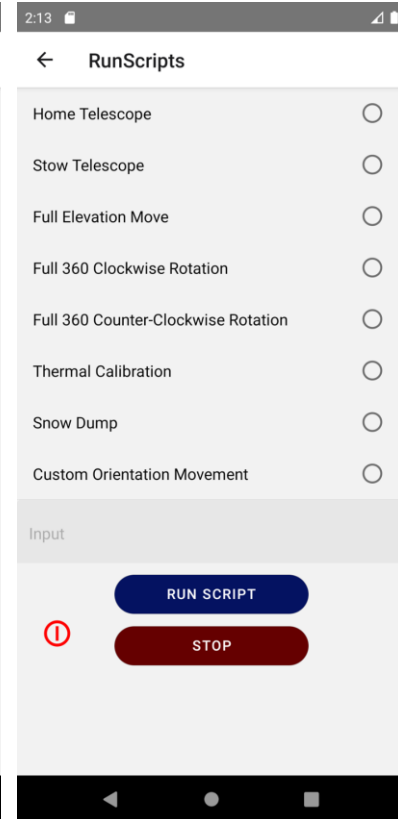
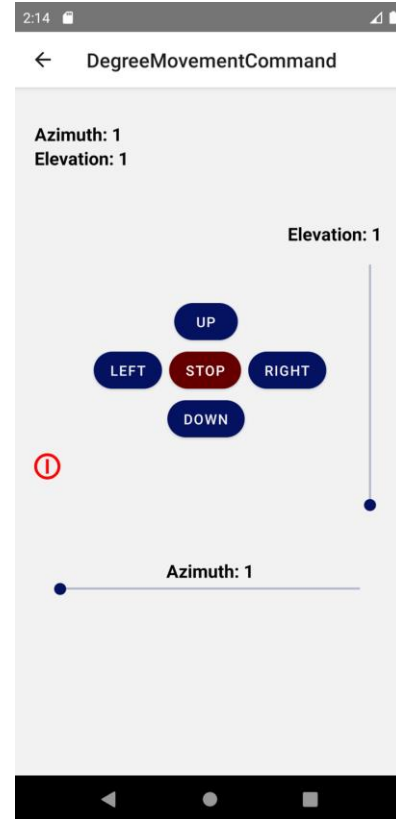
# Home

- From here the admin can navigate to:
  - Scripts
  - Degree Movement
  - Statuses
  - Weather Data



# Scripts and Commands

- Degree Movement Command Page
  - Allows relative movements
- Run Scripts Page
  - All Control Room scripts
  - Custom Orientation Movement
    - Absolute position



# Sending Command to Control Room

- Command is concatenated depending on the selection
- Encrypt the concatenated command
- Append the TCP version number for control room decryption
- Connect to TCP and Port specified by the constants file or edited by the admin on the settings page

Concatenated string

**<Version NUM> | <Type> | <Name <values>> | UTC TIME**

Encrypted string

**<Version NUM> | <ENCRYPTED COMMAND>**

# Control Room Overview

- “Brains” of the telescope
- 3 Forms
  - Main Form
  - Diagnostics Form
  - Control Form

The Main Form interface is divided into several sections. At the top left, there is a table for RT configuration:

ID	PLC IP	PLC Port	MCU Port	WS Port
1	127.0.0.1	8082	8083	222

Below this table are simulation settings for Weather station (222), Spectra Cyber (777), Simulated Weather Station, Simulated PLC (127.0.0.1), and Remote Listener (80). The right side of the form contains system configuration fields for MCU IP Address (127.0.0.1), PLC port (8082), and MCU Port (8083). It also includes fields for Sensor Network Sever and Client, and checkboxes for simulation options. At the bottom, there is a 'Radio Telescope Control' section with 'Shutdown RT' and 'Start RT' buttons.

Main Form

The Diagnostics Form provides a comprehensive overview of the system's status. It features several data panels:

- Sensor Data:** A list of sensors with their status (all False).
- Temperature Conversion:** A toggle between Celsius and Fahrenheit, currently set to Fahrenheit.
- Weather Sensor Data:** Real-time weather information including Wind Direction (N), Wind Speed (12.86 MPH), Daily Rainfall (3.4 Inches/Day), Rain Rate (0.32 Inches), Inside Temperature (69.68 Fahrenheit), Outside Temperature (62.3 Fahrenheit), and Barometric Pressure (29.64 Inches/Hg).
- Absolute Motor Positions and Temperatures:** Displays Azimuth Position (139.99 Degrees), Elevation Position (45.88 Degrees), Azimuth Motor Temp (66.2 Fahrenheit), and Elevation Motor Temp (73.4 Fahrenheit).
- Elevation Ambient Temperature and Humidity:** Shows Ambient Temp (65 Fahrenheit), Ambient Humidity (32%), and Ambient Dew Point (34.46 Fahrenheit).
- Motor Controller Status:** Shows MCU Status as 'Running' with a 'Reset MCU Errors' button.

At the bottom, there are three graphs showing Accelerometer Sensor Data for X, Y, and Z axes, and Counterbalance Accelerometer data.

Diagnostics Form

The Control Form is used for manual operation and configuration of the radio telescope. It includes:

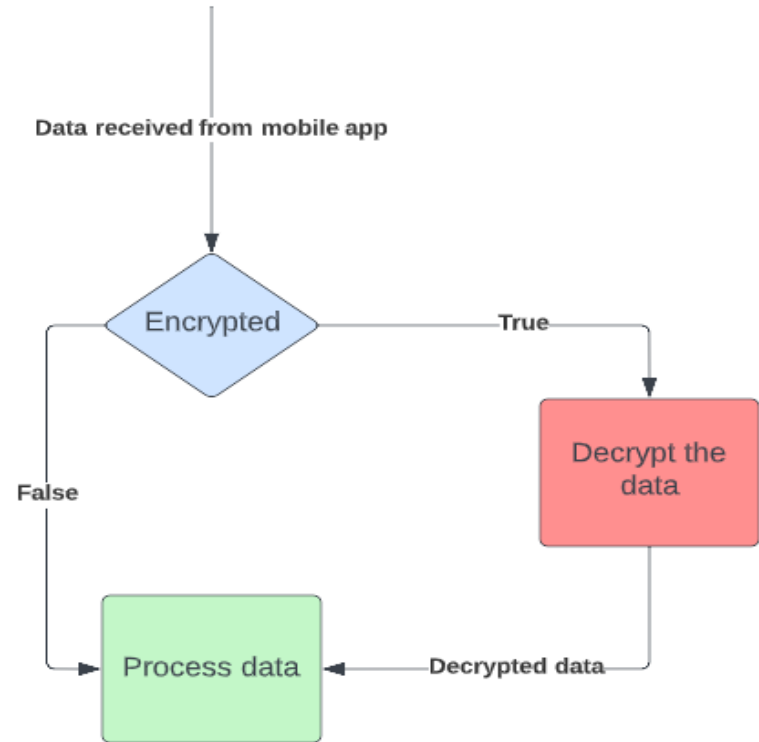
- Position Information:** Fields for Target and Actual Position (Right Ascension and Declination) and a 'STOP Telescope' button.
- Control Scripts and Spectra:** A dropdown for 'Radio Telescope Control Scripts' and a 'Run Script' button. Below it are settings for Spectra Cyber (Scan Type, Frequency) and DCGain (Gain, IF Gain, Offset Voltage, Integration Step, Int Step) with 'Start Scan' and 'Stop Scan' buttons.
- Edit Target Position:** Input fields for Right Ascension and Declination increments (0.25, 1, 5, 10) and manual control buttons (+ RA, - RA, + Dec, - Dec).
- Manual Control:** A large 'Activate Manual Control' button, along with '+ Ela', '- Ela', 'CCW Jog', and 'CW Jog' buttons. It also displays 'Current Elevation: 0.00' and 'Current Azimuth: 0.00'.

At the bottom right, there is a 'Speed (RPMs)' slider and a 'Free Control for Radio Telescope 1' indicator.

Control Form

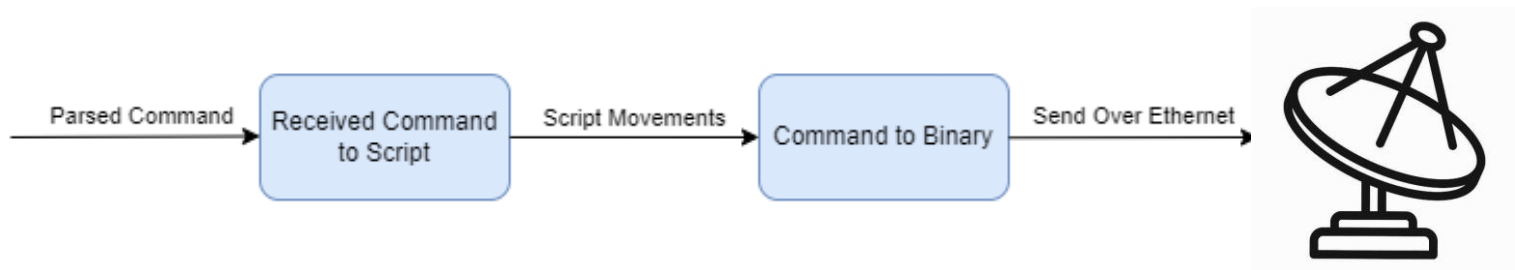
# Receiving command in Control Room

- When the command is received by the control room, it's checked to see if it's encrypted before processing the command
  - Uses the **<Version NUM>** tag from command
- The control room uses the AES-256 encryption standard
- The diagram on the right visualizes the process



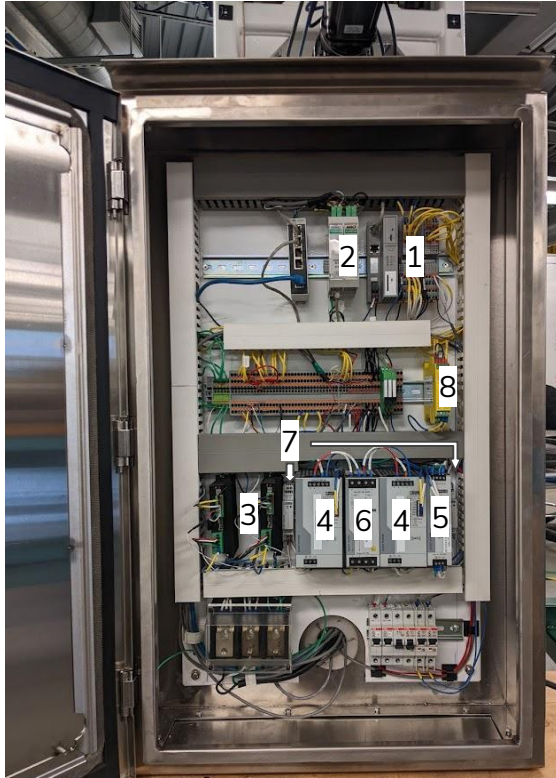
# Commanding the Telescope

- Commands are interpreted as scripts
- Scripts are a series of movements
- Different movements types to send to MCU
  - Absolute movements
  - Relative movements
  - Jog movements
- Binary instructions are assembled for the move and sent via ethernet





# Pier Electrical Panel (PEP)



## Key Components:

1. Programmable Logic Controller (PLC)
2. Motion Control Unit (MCU)
3. 2 Motor Stepper Drivers
4. 2 48VDC, 10A Power Supplies for Azimuth and Elevation Motors
5. 24VDC Control Panel Power Supply
6. Diode Redundancy Device
7. 5VDC, 12VDC power supply for the embedded sensor system
8. Safety Relay

# Pier Electrical Panel (PEP)



## At a glimpse:

1. Programmable Logic Controller (PLC)  
Part: Phoenix Contact Axioline 1050 Controller  
Role: Sends environment variables (inputs) to the control room software and outputs movement commands to the MCU if internal logic deems status as good.
1. Motion Control Unit (MCU)  
Part: Advanced Micro Controls ANF2 2 Axis Servo/Stepper Controller  
Role: Communicates to the PLC to get movement command through Modbus serial communication. Creates a 2 axis motion profile to send to the motor stepper drivers through signal and direction signals.
1. 2 Motor Stepper Drivers  
Part: Applied Motion Products STR8  
Role: Generates the phase current switch according to the MCU signal pulses. Higher the frequency of switching = faster motor speed

# Pier Electrical Panel (PEP)



## At a glimpse:

4. 48VDC, 10A Power Supplies for Azimuth and Elevation Motors

Part: Phoenix Contact QUINT4-PS/1AC/48DC/10 - 2904611

Role: Converts 120VAC to 48VDC, 10A to pass through the diode redundancy devices and supply to the motor stepper drivers.

5. 24VDC Control Panel Power Supply

Part: Phoenix Contact QUINT4-PS/1AC/24DC/5 - 2904600

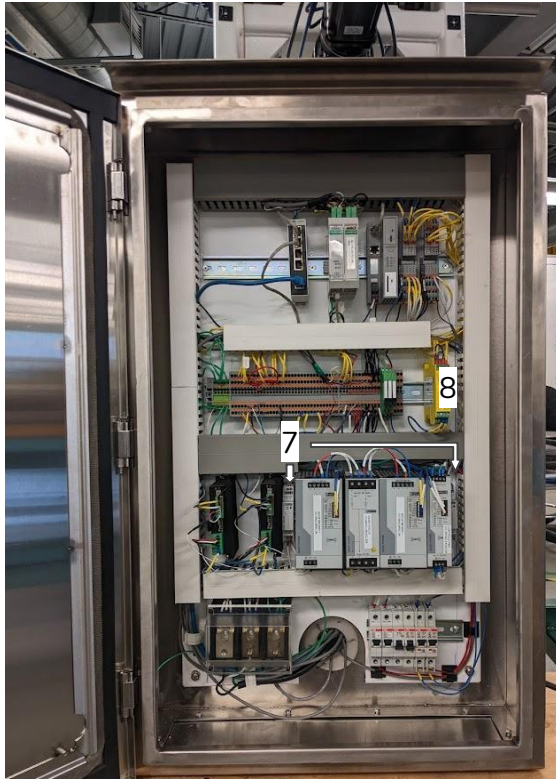
Role: Converts 120VAC to 24VDC, 5A to supply to the MCU, PLC, ethernet switch, and safety relay.

6. Diode Redundancy Device (DRD)

Part: QUINT4-DIODE/48DC/2X20/1X40 - 2907720

Role: Balances the load between the two 48VDC power supplies. In the event of a power supply failure, the DRD will allow the telescope to slowly return to the stow (upright) position.

# Pier Electrical Panel (PEP)



## At a glimpse:

7. 5VDC, 12VDC power supply for the embedded sensor system

Part: Phoenix Contact STEP-PS/1AC/5DC/2 & STEP-PS/1AC/12DC/1

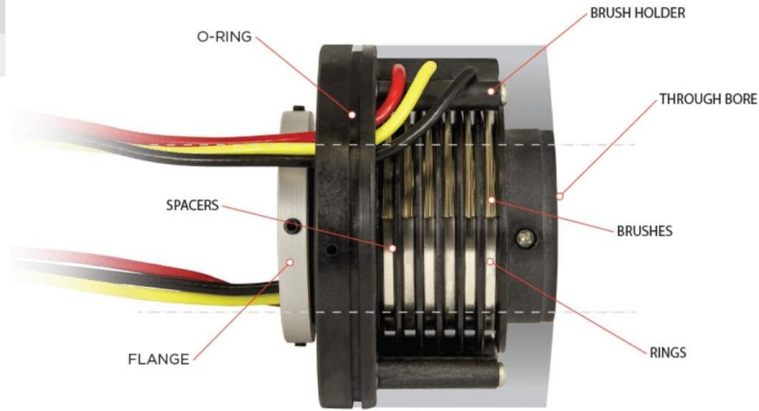
Role: Steps 120VAC down to supply 12VDC and 5VDC to the embedded sensor teensy microcontroller and the low noise amplifier (LNA).

8. Safety Relay

Part: Phoenix Contact PSR-MC72-2NO-1DO-24DC-SC

Role: Monitors system e-stops and sends status back to the PLC.

# Slip Ring Connections



- Bridges connections between the PEP and the ESS/upper telescope assembly
- Allows 360 degree motion of:
  - Motor Encoders
  - Motor Power
  - Home sensors
  - Limit Switches (signal, power)
  - ESS Power
  - Ethernet
- RF slip ring connects the 1.42 GHz signal

# Embedded Sensor System (ESS)



- The primary purpose of the embedded sensor system is to monitor a number of sensors and send data to the control room
  - Collect vibration and temperature data of both motors
  - Collect ambient temperature and humidity data of elevation enclosure
  - Collect angular position and vibration data of counterbalance
  - Collect position data of the azimuth top plate and elevation shaft
  - Command fan to prevent condensation in elevation enclosure

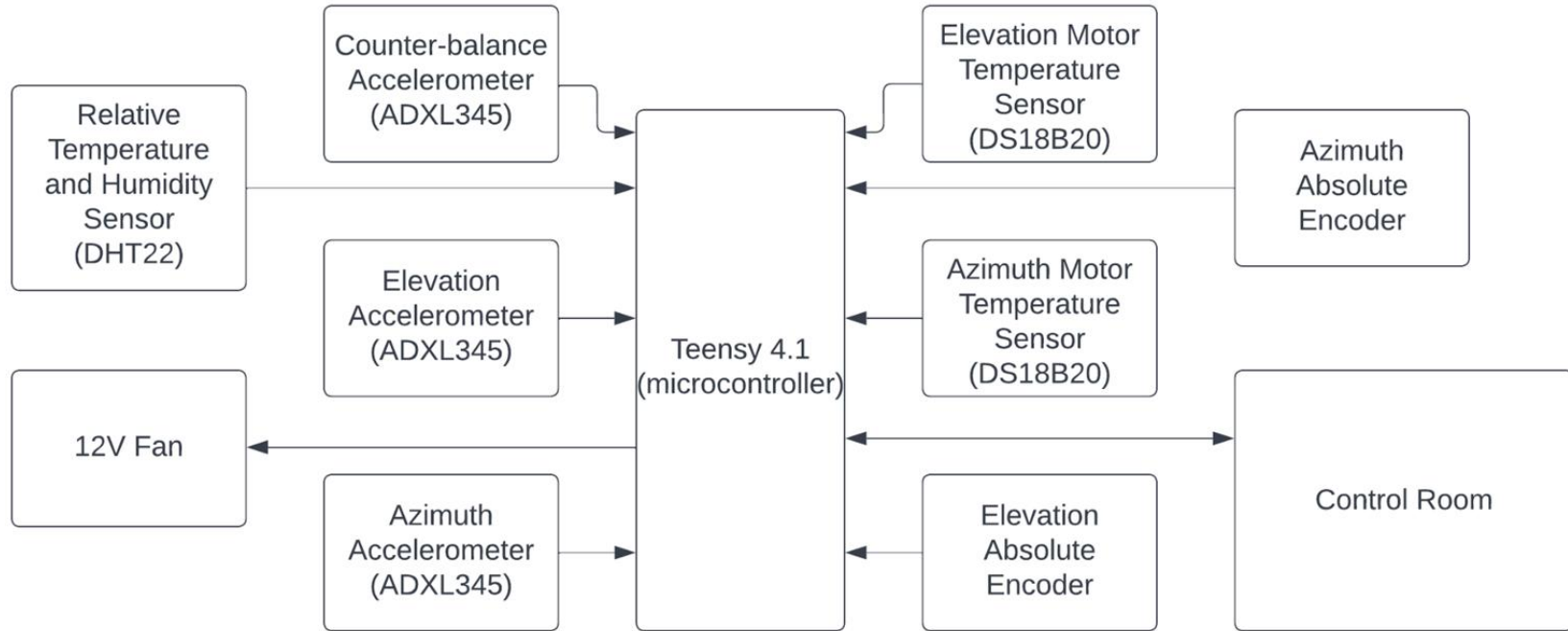
# PCB Revisions 5 and 6



- Primary changes from revision 4 include
  - Removing RJ45 and implementing 2.54mm and 3.96mm connectors
  - Implementing additional connector
    - Ambient temperature/humidity sensor, 12V fan
  - Addition of Bluetooth/Wifi module for OTA updating (Rev 6)
  - Addition of shunt capacitor (Rev 6)
  - Mounting hole changes (Rev 6)
  - Reducing size of PCB (Rev 6)

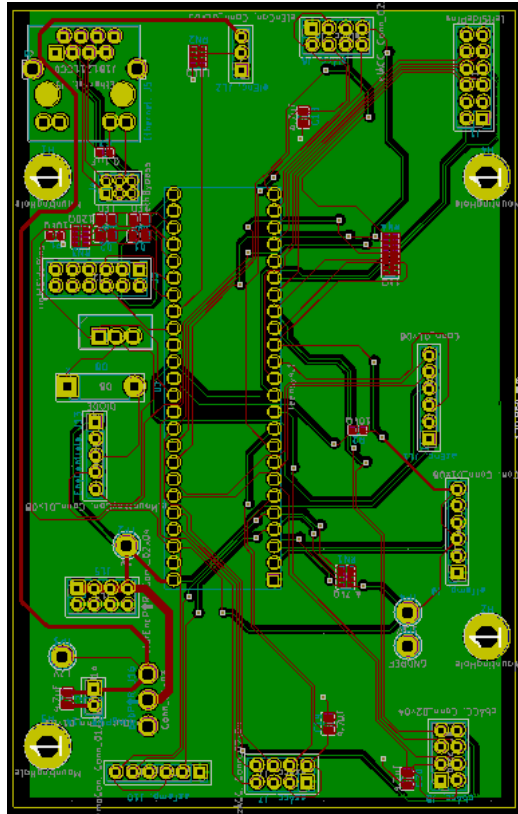


# Embedded Sensor System (ESS)

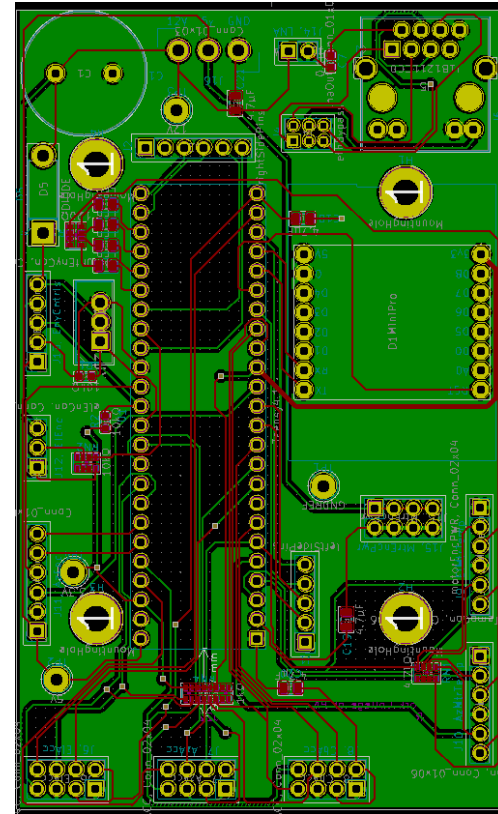




# Embedded Sensor System (ESS)

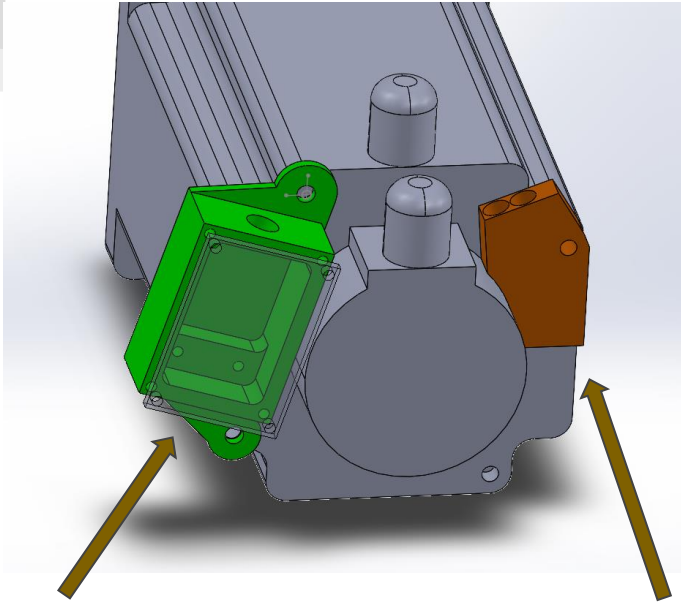


Revision 5



Revision 6

# Data Collection of sensors



## Elevation and Azimuth accelerometer

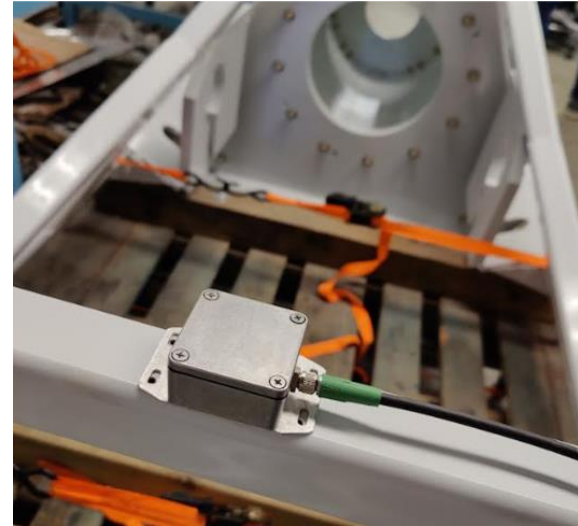
This is mounted to the Azimuth or Elevation motor, and must be tightly fastened in order for the vibrations of the system to transmit reliably to the accelerometer.

## Temperature Sensor

This is mounted to the Azimuth and Elevation motor, this allows the temperature of the motor to be monitored.

## Counterbalance Accelerometer

This is mounted to the counterbalance, its purpose is to transmit position and elevation angles to the operator.



# Embedded Sensor System (ESS)

- On power-on, ESS enters a “setup” state; connection to the control room established
- ESS receives 45-byte configuration packet from control room
  - Contains sensor enabling, timer values, and accelerometer parameters
- Starts watchdog timer, initializes sensors, setting up ISRs, and performing accelerometer self-tests

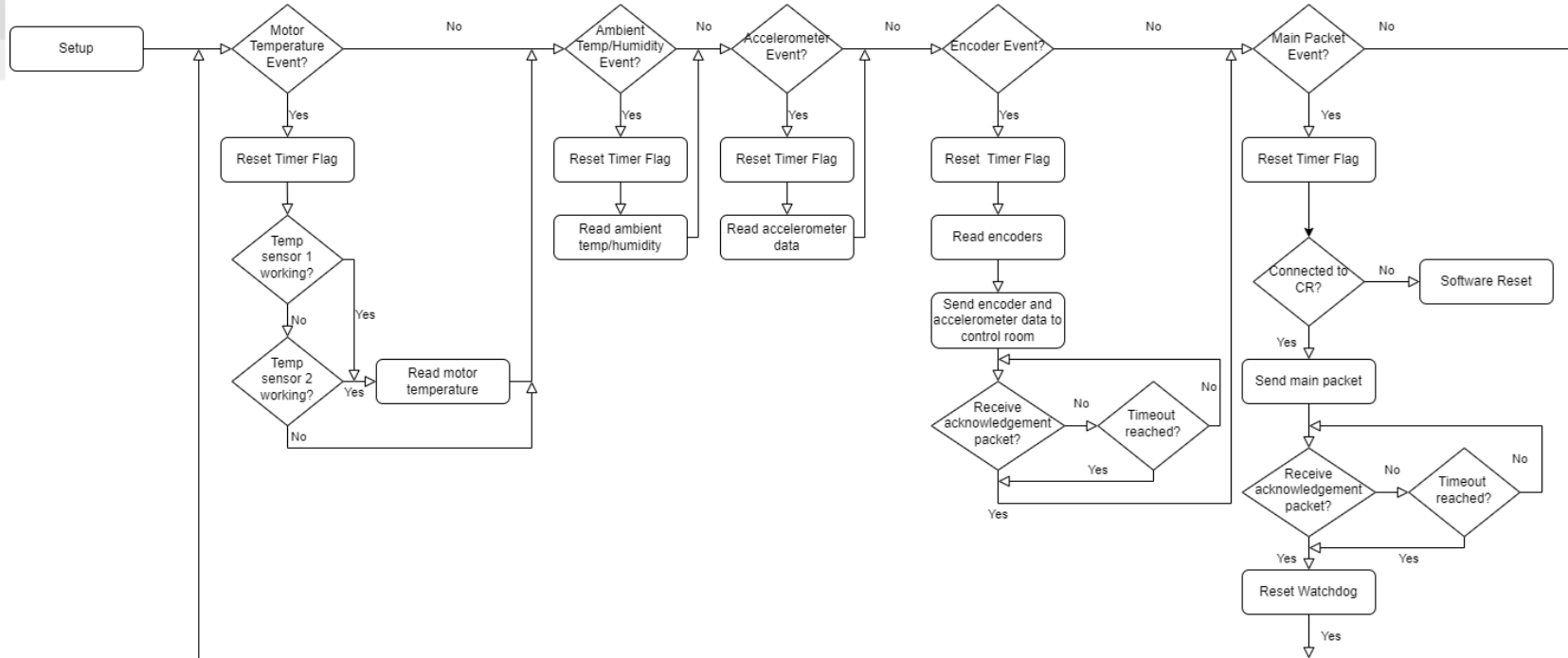


# Embedded Sensor System (ESS)



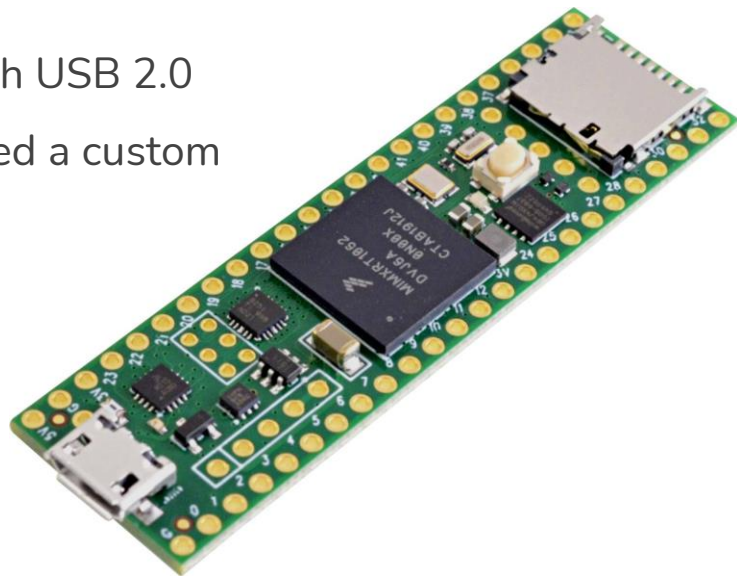
- On entering super-loop, ESS begins collecting data. On every iteration, sensor data is added to queue
- After a set amount of time, ESS sends contents of queue to control room
  - “Encoder” packet: Contains data from elevation and azimuth absolute encoders, as well as counterbalance accelerometer data
  - Main packet: Contains rest of the sensor data (accelerometers, temperature sensors)
  - After every packet sent, ESS waits for a reply from the control room

# Embedded Sensor System Superloop



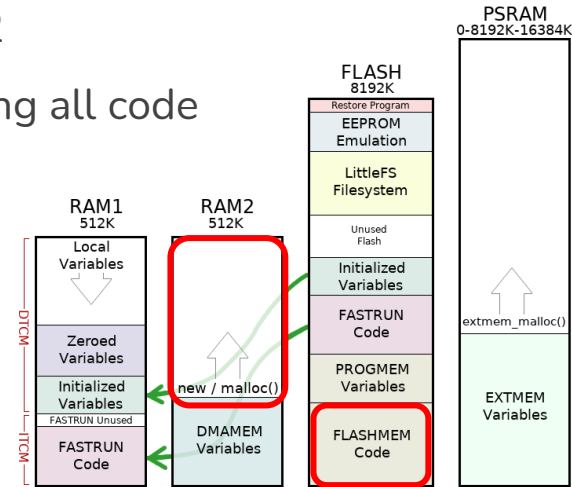
# Embedded Sensor System Updating

- Updates or changes to the ESS are performed through software updates to the Teensy
- Teensy 4.1 requires USB 2.0 for standard updating via the on-board micro USB port
  - The Slip Ring is not compatible with USB 2.0
  - Designed, developed, and integrated a custom Bluetooth OTA updating protocol



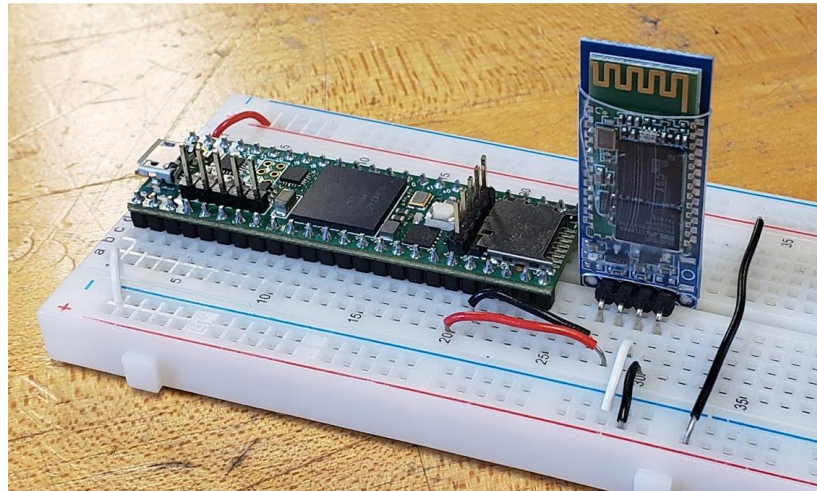
# Bluetooth OTA Updating Process

- User sends serial command to ESS via BT to initiate update
- Teensy pauses Superloop
- User sends new source code in Intel HEX format via BT
- Teensy receives source code and performs validation on each line
  - Teensy stores validated source code in RAM2
- Teensy prompts user for confirmation after receiving all code
- User sends command to reboot into new code
- Teensy flashes source code from RAM2 to FLASH
- Teensy reboots into new source code



# Physical Bluetooth Implementation

1. HC-06 Bluetooth module communicates with control room via BT COM port
2. Serial communication is opened with Tera Term emulation tool
3. HC-06 is powered with 5V on PCB
4. HC-06's data pair is connected to Teensy's Serial7 data pair





# Bluetooth OTA Updating Demo

The image displays a development environment for a Bluetooth OTA updating demo. It features a Visual Studio Code window with the following components:

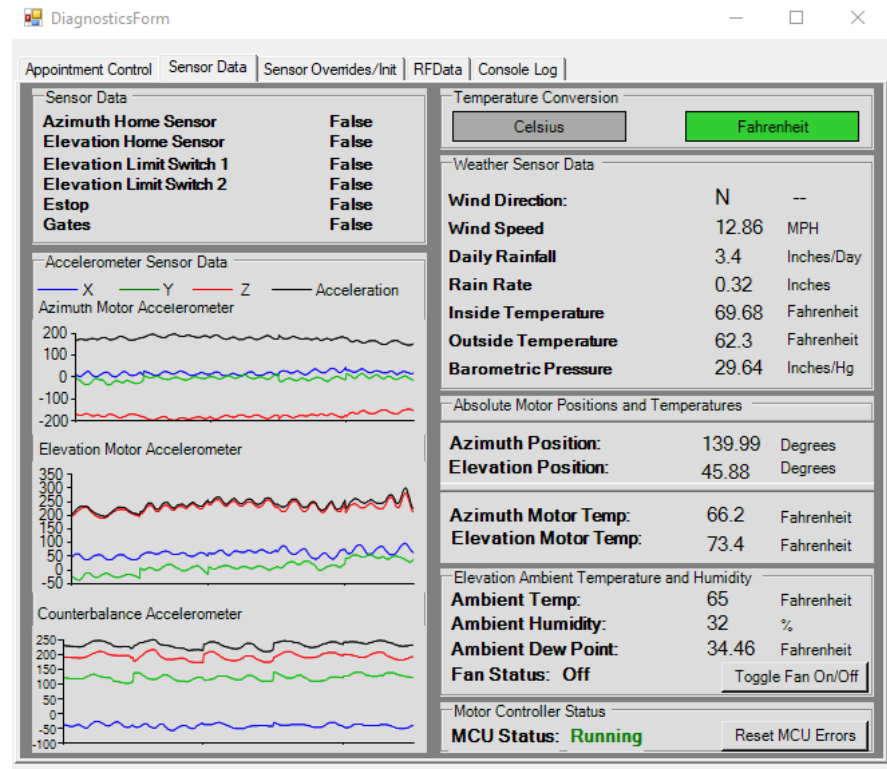
- EXPLORER:** Shows the project structure for 'FlasherXino', including files like `.pio`, `.vscode`, `c_cpp_properties.json`, `extensions.json`, `launch.json`, `settings.json`, `include`, `FlashBox.h`, `src`, `FlasherXino`, `FlashBox.cpp`, `.gitignore`, `platformio.ini`, and `README.md`.
- EDITOR:** Displays the `FlashBox.cpp` file with the following code snippet:

```
src > FlasherXino > loop()
79     }
80     else {
81         ((HardwareSerial*)serial)->begin( 115200 );
```
- TERMINAL:** Shows the compilation and linking process:

```
No dependencies
Building in release mode
Compiling .pio\build\teensy41\src\FlasherXino.cpp.o
Linking .pio\build\teensy41\firmware.elf
```
- Tera Term Dialog:** A 'New connection' dialog box is open, showing the 'Serial' connection type selected. The port is set to 'COM3: Intel(R) Active Management Te'. Other options like TCP/IP, TeJnet, and SSH are also visible.
- Inset Image:** A photograph of a physical microcontroller board (likely a Teensy 4.1) connected to a breadboard with various components and wires.

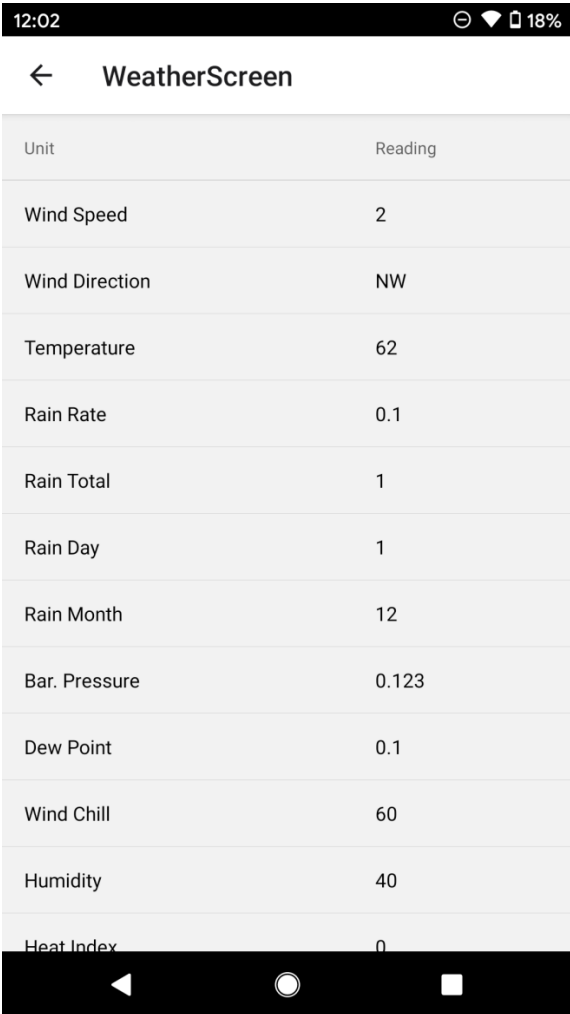
# Monitoring Movements - Control Room

- As telescope moves, sensor data from ESS and weather station is collected and displayed
- Decisions are made based on the data collected and the safety of the telescope
- Toggle fan if telescope is too hot or too humid



# Weather Data

- Weather data will show the current weather from the Control Rooms weather station
- When moving via jog this page will update every second
- Otherwise it updates every minute



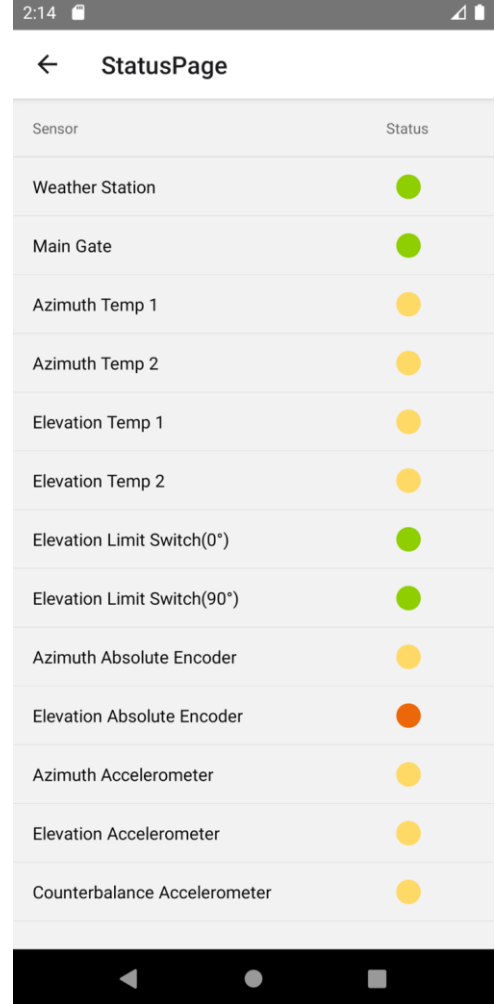
12:02 18%

← WeatherScreen

Unit	Reading
Wind Speed	2
Wind Direction	NW
Temperature	62
Rain Rate	0.1
Rain Total	1
Rain Day	1
Rain Month	12
Bar. Pressure	0.123
Dew Point	0.1
Wind Chill	60
Humidity	40
Heat Index	0

# Status

- Before doing any commands admins can check the sensor status page
- The most recent status for each sensor will be shown
- Admins may override statuses from this page

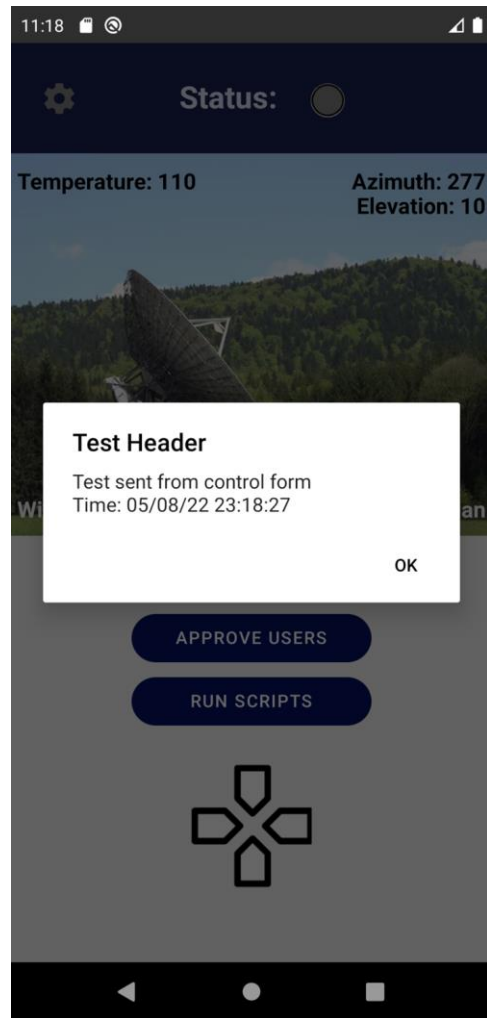


The screenshot shows a mobile application interface for a 'StatusPage'. At the top, there is a status bar with the time '2:14' and battery level. Below it is a navigation bar with a back arrow and the title 'StatusPage'. The main content is a table with two columns: 'Sensor' and 'Status'. The table lists 14 sensors with their corresponding status indicators (green, yellow, or red circles).

Sensor	Status
Weather Station	●
Main Gate	●
Azimuth Temp 1	●
Azimuth Temp 2	●
Elevation Temp 1	●
Elevation Temp 2	●
Elevation Limit Switch(0°)	●
Elevation Limit Switch(90°)	●
Azimuth Absolute Encoder	●
Elevation Absolute Encoder	●
Azimuth Accelerometer	●
Elevation Accelerometer	●
Counterbalance Accelerometer	●

# Push notifications

- Push notifications are sent from the Control Room via Google's Firebase Notification Service APIs
- Push notifications are sent when:
  - Sensors override
  - Sensors become critical
  - Disconnect from ESS



# Sensor Status/ Sensor Override

The image displays a web application interface for a radio telescope control system, split into a mobile app view on the left and a desktop control panel on the right.

**Mobile App View (Left):**

- Status:** A toggle switch is currently turned off.
- Temperature: 110**
- Azimuth: 277**
- Elevation: 10**
- Wind: 20 Knots mph SW**
- Appt by: Dylan**
- Buttons:** STOP TELESCOPE, APPROVE USERS, RUN SCRIPTS.
- Control:** A directional pad (joystick) for manual control.

**Desktop Control Panel (Right):**

**MapInfo**

Click on the IP address of the RT to open diagnostic tool

ID	PLC IP	PLC Port	MCU Port	WS Port
1	127.0.0.1	8082	8083	122

System IP Address and Port Numbers

MCU IP Address: 127.0.0.1  
PLC port: 8082  
MCU Port: 8083

Sensor Network Server: 127.0.0.1 1600  
Sensor Network Client: 127.0.0.1 1680

not back to simulator  Read Value for production

**Edit Settings**

**Radio Telescope Control**

**Shutdown RT** **Start RT**

**Diagnosiform**

Appointment Control | Sensor Data | Sensor Overrides | RFData | Console Log

Weather Station	Status	Upper	Lower
Weather Station	Offline	17.00	1.00
Gate Sensors	Online	100.00	00.00
Main Gates Sensors	Online	20.00	00.00

Temperature Sensors

- Azimuth Motor Temperature Sensor: Online
- Elevation Motor Temperature Sensor: Online
- Ambient Temperature Humidity Sensor: Online

Proximity Sensors

- Elevation Limit Switch 0°: Online
- Elevation Limit Switch 90°: Online

Encoders

- Azimuth Absolute Encoder: Online
- Elevation Absolute Encoder: Online

Accelerometers

- Azimuth Motor Accelerometer: Online
- Elevation Motor Accelerometer: Online
- Counterbalance Accelerometer: Online

Software Setup Limits

Ambient Temperature (F): 100.00 | G-Range: 16

Ambient Humidity (%): 20.00

**Update Thresholds**

Sensor Network Sensor Initialization

Accelerometer Settings: Counterbalance

Sampling Speed (Hz): 800 | G-Range: 16

FIFO Size: 32 | Offset X: 0 | Y: 0 | Z: 0

Full Bit Resolution

Timer Settings: Timer | Period (ms): 1

Elevation Encoder

Azimuth Encoder

Elevation Motor Temperature

Azimuth Motor Temperature

Ambient Temperature and Humidity

Azimuth Accelerometer

Elevation Accelerometer

Counterbalance Accelerometer

Data Retrieval Timeout (seconds): | Initialization Timeout (seconds): 10

Status: ReceivingData | Update Sensor Configuration

**Diagnosiform**

Appointment Control | Sensor Data | Sensor Overrides | RFData | Console Log

Hardware	Status	Current Appointment
SpectraCyber	Online	Start Time
Weather Station	Online	End Time
MCU	Offline	Status

Diagnostic Scripts: **Run Script**

Encoder Simulation

**Azimuth Encoder**

Degrees: 260.39 | Custom Value

Ticks: 0 | Custom Value

**Elevation Encoder**

Degrees: 0.81 | Custom Value

Settings

Set Error:   
Set Position:   
Set Bits of Precision:

**Test Notification**

**logi**

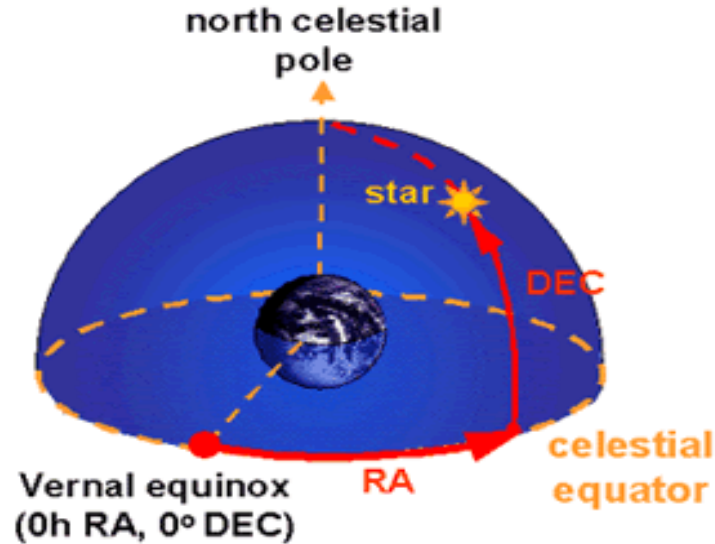


# Appointments



# Right Ascension(RA) and Declination(DEC)

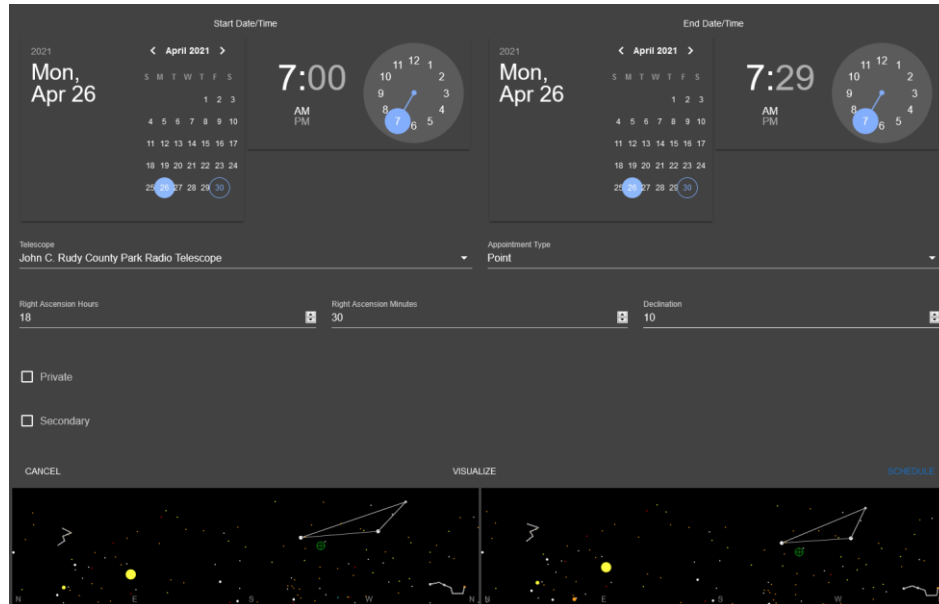
- A simple coordinate system for stars
- Right Ascension(RA) similar to X
- Declination(DEC) similar to Y





# Scheduling an Appointment

- User specifies appointment type and position on website
- Submitted appointment gets reviewed by system admin
- Approved appointments get scheduled to run



The screenshot shows a scheduling interface for a telescope appointment. It features two date and time pickers for the start and end of the appointment, both set for Monday, April 26, 2021, at 7:00 AM and 7:29 AM respectively. Below these are dropdown menus for the telescope (John C. Rudy County Park Radio Telescope) and the appointment type (Point). There are also input fields for Right Ascension Hours (18), Right Ascension Minutes (30), and Declination (10). Checkboxes for 'Private' and 'Secondary' are present. At the bottom, there are buttons for 'CANCEL', 'VISUALIZE', and 'SCHEDULE', along with two side-by-side star charts showing the telescope's field of view.

Start Date/Time

2021  
Mon, Apr 26

< April 2021 >

S M T W T F S

1 2 3

4 5 6 7 8 9 10

11 12 13 14 15 16 17

18 19 20 21 22 23 24

25 26 27 28 29 30

7:00

AM PM

End Date/Time

2021  
Mon, Apr 26

< April 2021 >

S M T W T F S

1 2 3

4 5 6 7 8 9 10

11 12 13 14 15 16 17

18 19 20 21 22 23 24

25 26 27 28 29 30

7:29

AM PM

Telescope

John C. Rudy County Park Radio Telescope

Appointment Type

Point

Right Ascension Hours

18

Right Ascension Minutes

30

Declination

10

Private

Secondary

CANCEL VISUALIZE SCHEDULE

# Appointment Execution



- General overview, an appointment will run some movements on the telescope and save radio frequency data
  - First, the appointment data is pulled from the database
  - Spectracyber is then configured to scheduler's settings
  - Once this is done, the telescope performs calibration readings
  - After calibration, the telescope moves to the orientation specified by the appointment for the specified duration
  - Once the appointment duration ends, the telescope performs calibration readings again, and an email with the results is sent to the user who scheduled the appointment

# Appointment Calibration



- Before an appointment starts, the telescope needs to be calibrated to ensure data accuracy
- The telescope is calibrated before and after every appointment
- The steps for the process are as follows (yes, we do scan a literal tree)
  - Home the telescope
  - Perform beginning calibration (Tree and Zenith)
  - Run the appointment
  - Perform end calibration (Zenith and Tree)
  - Compile the data and send it to a user

# SpectraCyber & Data Collection

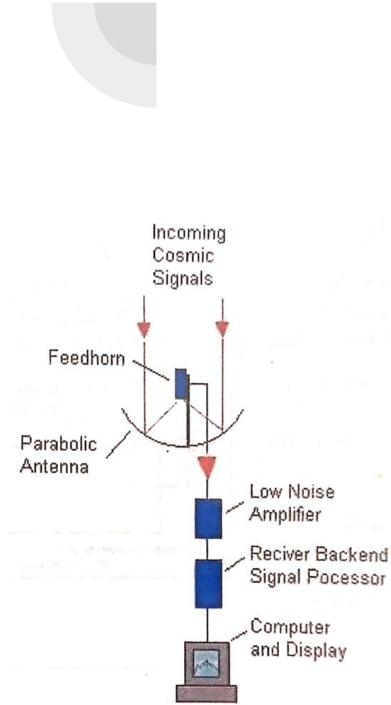


Figure 2. System diagram



Figure 1. Feedhorn Assembly

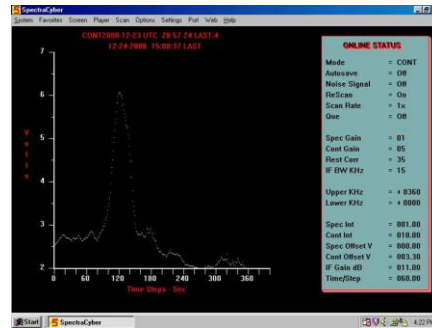
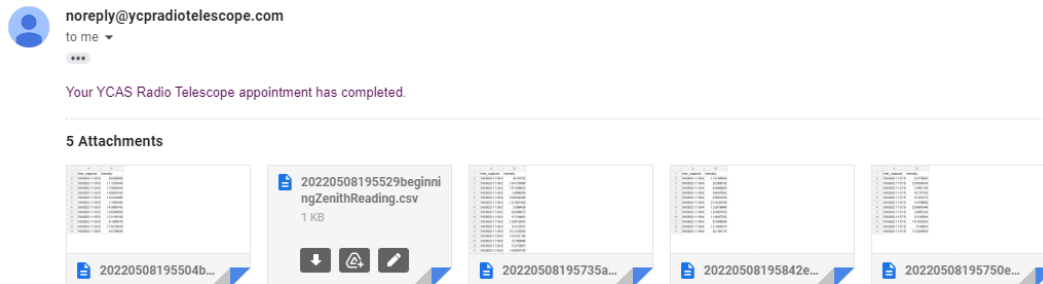


Figure 3. Sample SpectraCyber reading display

1. Parabolic antenna (the dish) reflects the RF signal to a focal point within the waveguide horn
1. 12VDC, 100mA is used to amplify the RF signal using a Low Noise Amplifier (LNA)
1. Signal is transferred through LMR 400 coaxial cable to the SpectraCyber
1. SpectraCyber processes the signal and displays it on the control room computer

# Data Collected and Sent to Users

- During the course of an appointment we collect radio frequency data
- We need to collect calibration data both before and after an appointment
  - Helps with discovering any discrepancies
- We collect data during the course of an appointment
- All data is sent to CSV files which are added to an email that gets sent to the user once appointment completion is finished





# Inertial Testing



# Inertial Testing Purpose

- Simulate the weight and momentum of the complete radio telescope assembly.
- Evaluate the motors ability to handle impulses and accelerations in both the azimuth and elevation directions
- Test adjustments to the motors



# Inertial Testing Calculations

- Verify structural integrity of wooden support arms and steel U bolts
- Check the difference in velocity required for the inertia test frame to have the same momentum as the counterbalance and hub assembly
  - Inertia test frame has the same momentum when spun at 1.61x the rate of the counterbalance and hub assembly

D	D/2	1+D/2 (rc)	ro		
0.6291	0.31455	1.31455	1.6291	in	
Rn		Area			Moment (lb-in)
1.295456	in	0.310835	in <sup>2</sup>		525.82
e	co				
0.019094	0.333644	in			
σ	σy				
18144.57	36300	psi			
FS					
2.000598					

Inertial Testing Calculations:

$$I = m_1 r_1^2 + m_2 r_2^2 \quad m_1 = m_2 \quad r_1 = r_2$$

$$= 2(m_1 r_1^2)$$

$$= 2(400 \text{ lbs} \cdot 4 \text{ ft}^2)$$

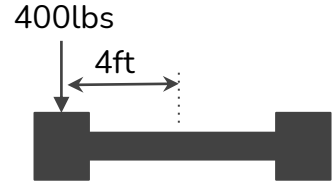
$$= 12,800 \text{ lb} \cdot \text{ft}^2$$

Momentum replication

$$L = I\omega$$

$$I_1 \omega_1 = I_2 \omega_2$$

$$\frac{\omega_2}{\omega_1} = \frac{I_1}{I_2} = \frac{20,600 \text{ lbft}^2}{12,800 \text{ lbft}^2} = 1.61 \text{ times faster}$$





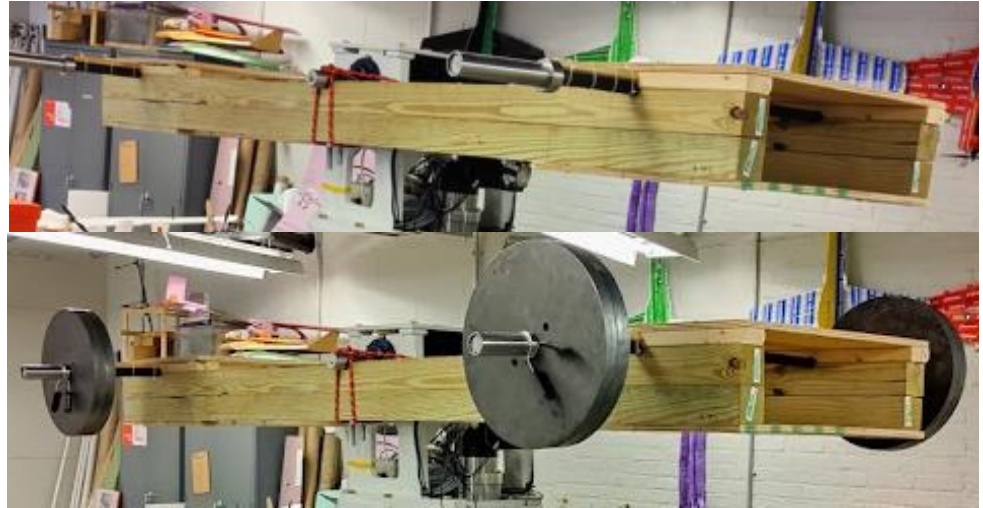
# Inertial Testing Frame Construction

- Wooden support arms had circular cutouts for the elevation shaft and for barbells
- Steel U bolts were heated and bent into shape



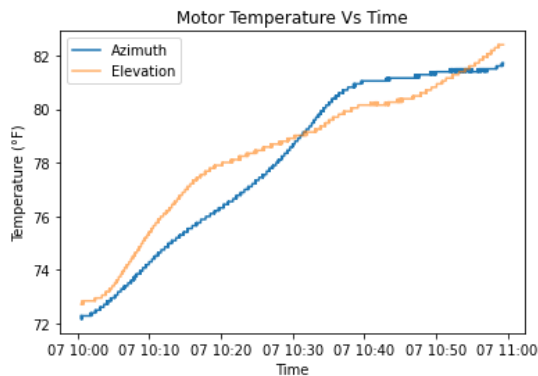
# Inertial Testing Process

- Frame only, 400 lbs added, then 800 lbs added
- All tests run
  - Various controlled speeds
  - Stopping methods

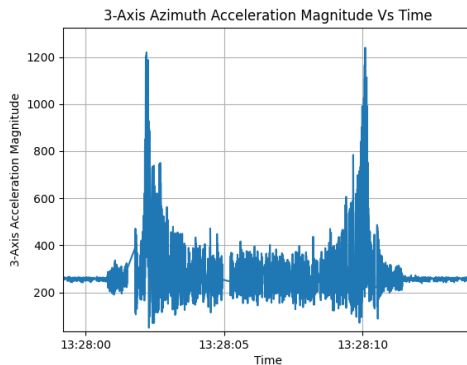


# Inertial Testing Software Tools

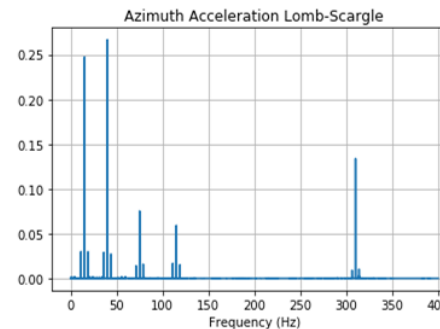
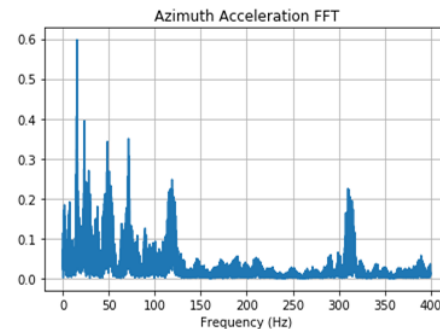
- How data was collected
  - (Accel, temp, position, encoder positions)
- Analyzed with Python tools



Temperature



Acceleration



Vibration Frequencies

# Inertial Testing Results



- Stored all data
- Immediate stop is too immediate
  - Too damaging to the gearboxes
  - Entire telescope shakes
  - Caused azimuth readings to be off
- Motors give out after Control Room disconnection
- Both motors increase in temperature even though one is moving



# Simulation and VR Game



# The Sim and Its Importance

- Useful for software testing
- Reads commands from control room
- Functionally the same as the real telescope

13:3 ms (75 fps)

MCU IP Address  
MCU Port  
Auto Fill for Sim  
Start Sim

Test 1  
Test 2  
Test 3  
Test 4

Azimuth  
Elevation  
Speed  
Test Movement

Unity Azimuth: 0  
Unity Elevation: 15  
Sim Azimuth: 0  
Sim Elevation: 0  
Input Azimuth: 0  
Input Elevation: 0  
Target Azimuth: 0  
Target Elevation: 0  
Azimuth Speed: 0  
Elevation Speed: 0

Current command: simulation initialization

MSW command	LSW command	MSW data	LSW data	MSW speed	LSW speed	acceleration	Display base: 16
Incoming Azimuth: 0	0	0	0	0	0	0	deceleration 0
Incoming Elevation: 0	0	0	0	0	0	0	0

MSW status bits	LSW status bits	MSW steps	LSW steps	MSW encoder	LSW encoder	heartbeat
Outgoing Azimuth: 0	0	0	0	0	0	0
Outgoing Elevation: 0	0	0	0	0	0	0

# Example Sim Operation

PLC port:  
MCU Port:  
Sensor Network Sever: 127.0.0.1  
Sensor Network Client: 8083

Loop back (for simulation)

Control Form

Position Information

Target Position	Actual Position
Right Ascension 23.26	Right Ascension 7.63
Declination 90	Declination 44.04

Radio Telescope Status:

Enable Software Stops

Edit Target Position

Right Ascension Increment: 0.25, 1, 5, 10

Declination Increment: 0.25, 1, 5, 10

Manual Control

Deactivate Manual Control

Current Elevation: 35.89  
Current Azimuth: 30.91

Speed (RPMs): 2

Buttons: +Ela, -Ela, CCW Jog, CW Jog, -RA, +RA, +Dec, -Dec

Control Scripts and Spectra

Radio Telescope Control Scripts

Run Script

Spectra Cyber

Frequency (kHz)

Scan Type

Finalize Settings

DCGain (dB) IFGain (dB) Offset Voltage Integration Step

Gain Int Step

Start Scan Stop Scan

Display 1: 16:19 Scale 1x Maximize On Play

10.5 ms (96 fps)


127.0.0.1  
8083  
Auto Fill for Sim  
Start Sim

Elevation: Test 1, Test 2, Test 3, Test 4, Test C, Test Movement

Azimuth:

Speed:

Unity Elevation  
Unity Azimuth  
Elevation Input  
Elevation Target  
Azimuth Target  
Azimuth Elevation

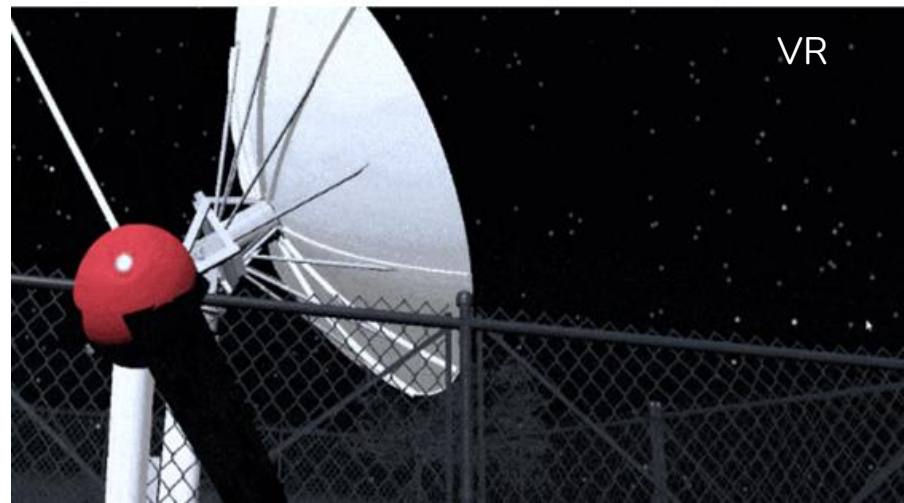


TELESCOPECONTROLLER: Negative Azimuth jog



# PC and VR Educational Visualizations

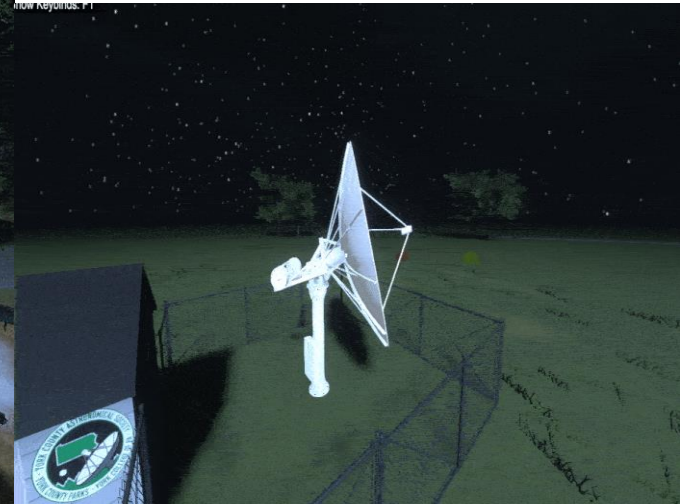
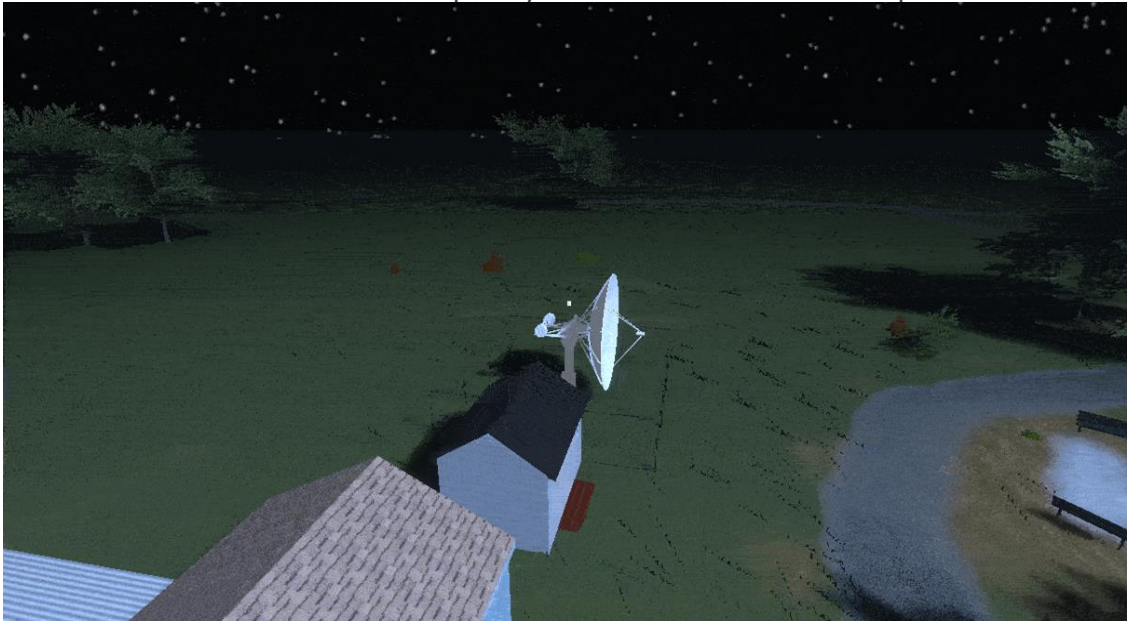
- Developed using Unity and C#
- Demonstrates the function and general construction of the telescope
- PC demonstration can be easily shown in classrooms
- VR demonstration effectively shows scale of telescope





# PC Mouse & Keyboard Visualization

- 3D program that demonstrates telescope model and operations
- Compatible with most computers
  - High quality version for high-end computers
  - Low quality version for low-end computers
- List of User Interactions:
  - Move telescope with arrow keys
  - Interact with star system data points
  - Interact with telescope parts

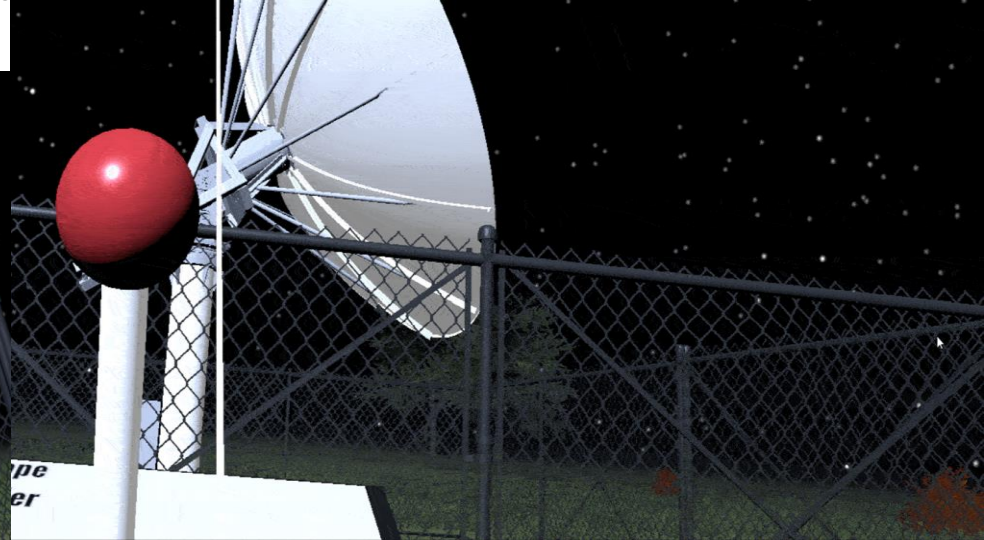


# Virtual Reality Visualization

- Similar to PC version, but is compatible with all VR devices
- User can gain sense of scale of telescope
- List of User Interactions:
  - Move telescope with joystick
  - Interact with star system data points
  - Interact with telescope parts



Dish: The dish antenna is in the form of a parabola and is the primary radio energy collecting element of the radio telescope. Optically it is equivalent to a parabolic mirror used in a reflecting telescope. When the dish antenna is pointed to a radio source the incoming radio waves are collected and reflected off the curved surface of the antenna and brought to a focus where it enters the feedhorn.



# Mock Replica of John C. Rudy County Park

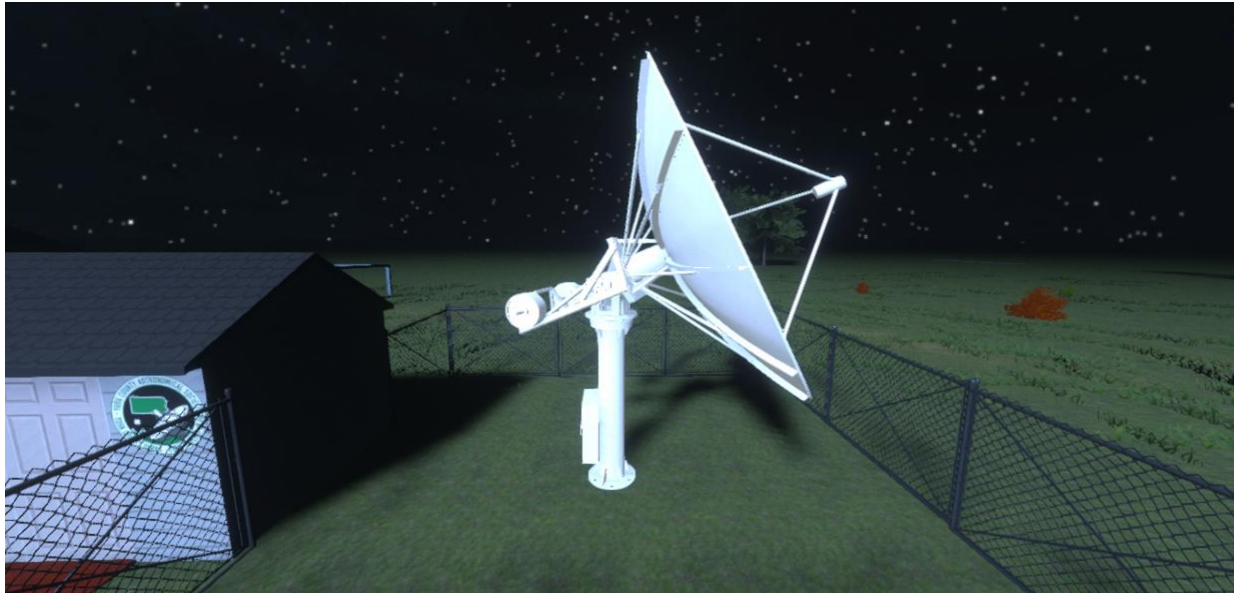
- Visualization scene is modeled after John C. Rudy Park
- Surrounding buildings and telescope are scaled accurately to real life
- Terrain, foliage, and benches added to simulate real life environment





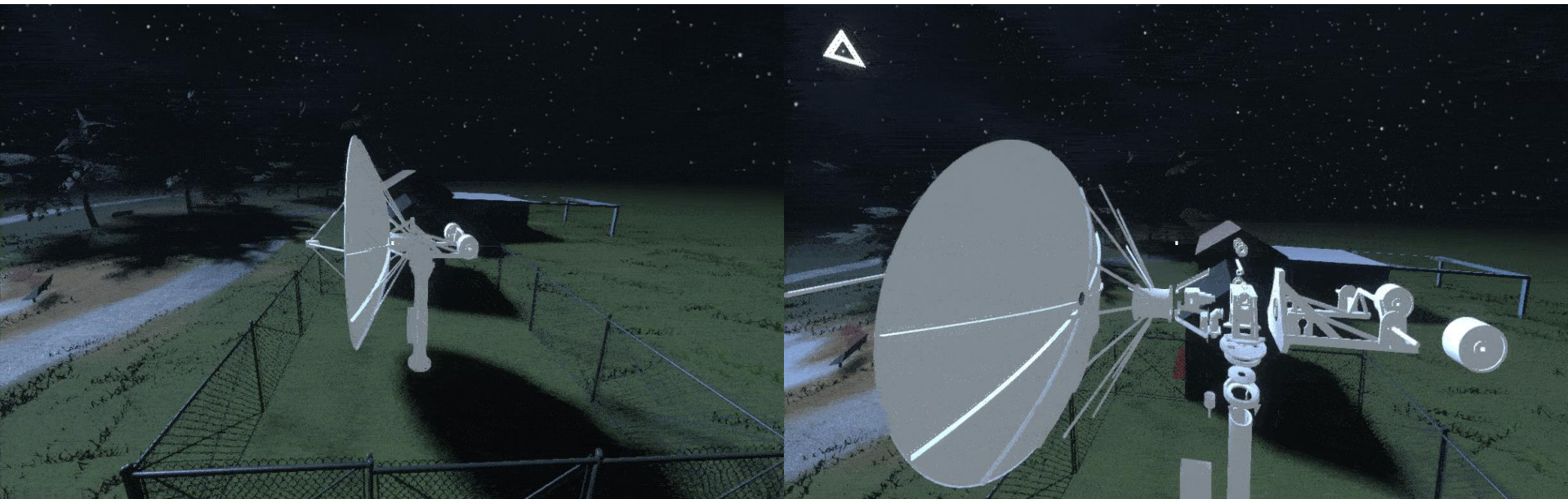
# Display of Telescope Model and Parts

- Collaboration between CS and ME students
- Imported Solidworks models into Unity
- Telescope model fully updated to include all parts
- Descriptions added to each telescope part



# Telescope Parts Expansion and Interaction

- Telescope can be expanded to show detailed composition of parts
- Interaction with a part will display its description



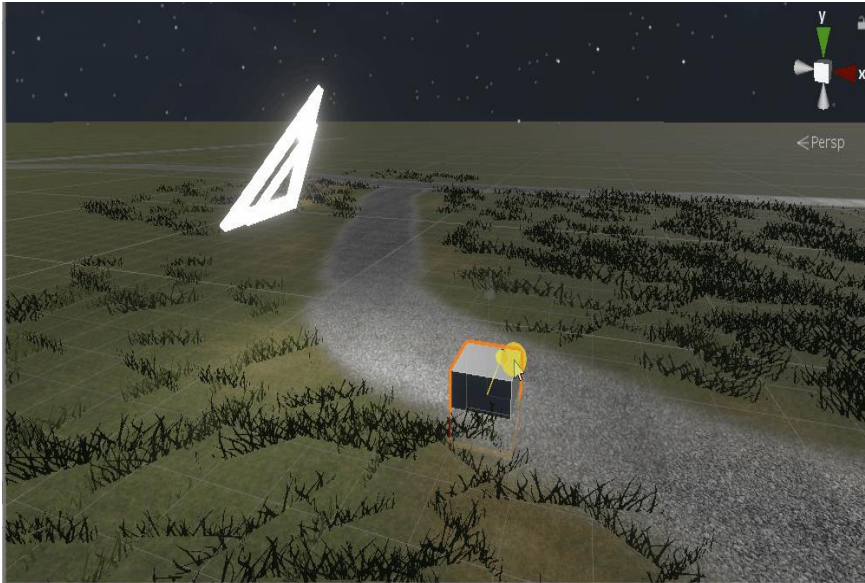
# Accurate Star System

- Brightest stars are imported from public database
- Placed using (X, Y, Z) spherical coordinates
- Stars are animated



# Star Interaction System

- Used to display telescope data points
- Triangle object is spawned on each data point
- Interaction shows Label, Description, and Image of data point





# Visualization Accessibility and Compatibility

- Using SteamVR API for VR compatibility
- Currently supporting Oculus Quest 2 and HTC Vive
- Separate Oculus Quest 2 APK demonstration

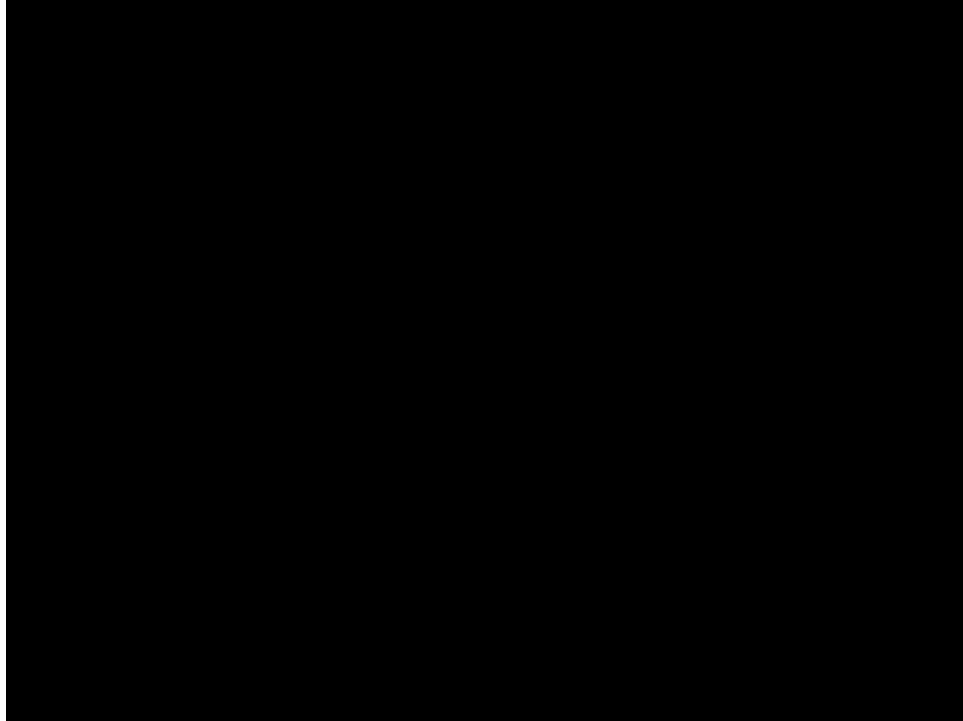






**Demo**

# Appointment Calibration Demo





**Questions?**

