



**TRIBHUVAN UNIVERSITY  
INSTITUTE OF ENGINEERING  
PULCHOWK CAMPUS**

**B-05-BAS-2018/23**

**Design and Construction of Satellite Tracking Rotator and Ground Station for  
Low Earth Orbit Satellites**

by

Amulya Bagale (075AER005)

Ankit Adhikari (075AER006)

Sagar Thapa (075AER035)

A PROJECT REPORT

SUBMITTED TO THE DEPARTMENT OF MECHANICAL AND AEROSPACE  
ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR  
THE DEGREE OF BACHELOR IN AEROSPACE ENGINEERING

DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

LALITPUR, NEPAL

MARCH 2023

## **COPYRIGHT**

The author has agreed that the library, Department of Mechanical and Aerospace Engineering, Central Campus Pulchowk and Institute of Engineering may make this project report freely available for inspection. Moreover, the author has agreed that permission for extensive copying of this project report for scholarly purpose may be granted by the professor(s) who supervised the work recorded herein or, in their absence, by the Head of the Department wherein the thesis was done. It is understood that the recognition will be given to the author of this project report and to the Department of Mechanical and Aerospace Engineering, Central Campus Pulchowk and Institute of Engineering in any use of the material of this project report. Copying or publication or the other use of this project report for financial gain without approval of the Department of Mechanical Engineering, Central Campus Pulchowk, Institute of Engineering and author's written permission is prohibited.

Request for permission to copy or to make any other use of this project report in whole or in part should be addressed to:

Head of Department  
Department of Mechanical and Aerospace Engineering  
Pulchowk Campus, Institute of Engineering  
Pulchowk, Lalitpur  
Nepal

TRIBHUVAN UNIVERSITY  
INSTITUTE OF ENGINEERING  
CENTRAL CAMPUS PULCHOWK

DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

The undersigned certify that they have read and recommended to the Institute of Engineering for acceptance, a project report entitled "DESIGN AND CONSTRUCTION OF SATELLITE TRACKING ROTATOR AND GROUND STATION FOR LOW EARTH ORBIT SATELLITES " submitted by Amulya Bagale, Ankit Adhikari and Sagar Thapa in partial fulfillment of the requirements for the degree of Bachelor of Aerospace Engineering.

---

Supervisor Asst. Prof Sudip Bhattra (Ph. D.)  
Department of Mechanical and Aerospace  
Engineering

---

Supervisor Asst. Prof. Sanjeev Maharjan (Ph. D.)  
Department of Mechanical and Aerospace  
Engineering

---

External Examiner, Er. Jiten Thapa  
Engineer  
Orion Space

Date: \_\_\_\_\_

---

Committee Chairperson  
Department of Mechanical and Aerospace  
Engineering

## **ABSTRACT**

The report entitled “DESIGN AND CONSTRUCTION OF SATELLITE TRACKING ROTATOR AND GROUND STATION FOR LOW EARTH ORBIT SATELLITES” presents the design and construction of a ground station and rotator utilizing a helical antenna that is compatible with the SatNOGS database. The ground station comprises a support structure, a helical antenna, a rotator and a radio receiver. The selection of a helical antenna is justified by its directional pattern and high gain, making it optimal for satellite tracking. A rotator is integrated to monitor the satellite's movement across the sky while the radio receiver decodes the signals. The SatNOGS database provides a comprehensive catalogue of satellite orbits and schedules that can be used to plan observations. Building a ground station with a helical antenna and rotator allows for satellite tracking and communication, contributing to the network of satellite tracking stations worldwide. This report delves into the details of the ground station's design and construction, including the necessary components, support structure and software required for optimal operation. Additionally, the use of the SatNOGS database for scheduling observations and monitoring their progress is explained. The report concludes with a discussion of the potential applications of the ground station, including future scope and satellite operations.

***Keywords:*** *Ground station, Rotator, Helical antenna, SatNOGS database, Satellite Tracking, Software-Defined Radio.*

## **ACKNOWLEDGEMENT**

We are delighted to present our project report entitled "DESIGN AND CONSTRUCTION OF SATELLITE TRACKING ROTATOR AND GROUND STATION FOR LOW EARTH ORBIT SATELLITES ". We would like to take this opportunity to express our sincere gratitude to the individuals who provided invaluable support throughout the project. Their guidance and encouragement have been instrumental in making this project a great success.

We would like to express our sincere gratitude to our project supervisor Dr. Sudip Bhattraai and Dr. Sanjeev Maharjan for their guidance and support throughout the duration of this project. Their knowledge and expertise have been invaluable in the design of the ground station and rotator with helical antenna with SatNOGS database.

We would also like to express our gratitude to Asst. Prof and Department Deputy Head Kamal Darlami for his assistance and guidance throughout the project. Additionally, we would like to acknowledge the special help and coordination provided by Er. Rakesh Chandra Prajapati, Er. Jiten Thapa and Er. Rishav Dhungel from Orion Space. Their contributions from the start of the project to the end weather at the time of resources unavailability or at the time of emergency their help been instrumental in the success of this project.

Finally, we would like to thank all of our family members and the entire 2075 Aerospace batch mates for their time and expertise in assisting us with this project at various mode and level. Your encouragement and support have been invaluable and we could not have completed this project without your assistance.

# TABLE OF CONTENTS

<b>COPYRIGHT</b> .....	<b>ii</b>
<b>ABSTRACT</b> .....	<b>iv</b>
<b>ACKNOWLEDGEMENT</b> .....	<b>v</b>
<b>LIST OF FIGURES</b> .....	<b>ix</b>
<b>LIST OF ACRONYMS AND ABBREVIATIONS</b> .....	<b>xi</b>
<b>1 CHAPTER ONE: INTRODUCTION</b> .....	<b>1</b>
1.1 Background .....	1
1.2 Problem Statement .....	2
1.3 Objectives .....	3
1.3.1 Primary Objective .....	3
1.3.2 Secondary Objectives .....	3
1.4 Application .....	3
1.5 Features .....	4
1.6 Feasibility Analysis .....	4
1.6.1 Economic Feasibility .....	4
1.6.2 Technical Feasibility .....	5
1.6.3 Operational Feasibility .....	5
1.7 System Requirements .....	5
1.7.1 Hardware Requirements .....	5
1.7.2 Software Requirements .....	6
<b>2 CHAPTER TWO: LITERATURE REVIEW</b> .....	<b>7</b>
2.1 Ground Station .....	7
2.2 Rotator Design .....	8
2.3 Antenna Design .....	9
2.4 Tracking software and Controller .....	11
<b>3 CHAPTER THREE: RELATED THEORY</b> .....	<b>13</b>

3.1	Frequency Bands .....	13
3.2	Standing Wave Ratio and Impedance Matching .....	13
3.3	Two Line Element .....	14
<b>4</b>	<b>CHAPTER FOUR: METHODOLOGY .....</b>	<b>16</b>
4.1	Ground station .....	16
4.2	Flowchart for rotator .....	18
4.3	Design Parameters for Antenna.....	19
4.4	Basic Geometry of the Antenna .....	19
4.5	Steps for the Design of Helical Antenna.....	20
4.6	Equations for the Helical Antenna .....	21
4.7	Hardware used.....	21
4.7.1	RTL-SDR.....	21
4.7.2	Raspberry Pi.....	23
4.7.3	Arduino Pro Nano .....	24
4.7.4	Stepper Motor .....	24
4.7.5	A4988.....	25
4.7.6	LM2596.....	25
4.7.7	N-type RF Connectors .....	26
4.7.8	Nano VNA .....	26
4.8	Software Used .....	27
4.8.1	GNU Radio .....	27
4.8.2	SatNOGS Client.....	28
<b>5</b>	<b>CHAPTER FIVE: RESULTS AND DISCUSSION.....</b>	<b>29</b>
5.1	Antenna Design .....	29
5.2	Rotator Design.....	34
5.3	Rotator Controller .....	35
5.4	SatNOGS Client Setup.....	36

5.5	Assembly.....	39
5.6	Observations from Ground Station .....	41
5.6.1	Observation of CELESTA .....	41
5.6.2	Waterfall diagram of CELESTA .....	42
5.6.3	Observation of the DUCHIFAT-3 .....	43
5.6.4	Waterfall diagram of DUCHIFAT-3 .....	44
5.7	Limitations .....	46
5.8	Problems faced .....	46
<b>6</b>	<b>CHAPTER SIX: CONCLUSION AND FUTURE ENHANCEMENTS .....</b>	<b>47</b>
6.1	Conclusion.....	47
6.2	Future Enhancements .....	47
<b>7</b>	<b>REFERENCES.....</b>	<b>48</b>
<b>8</b>	<b>APPENDICES .....</b>	<b>50</b>
	APPENDIX A: BUDGET ANALYSIS .....	50
	APPENDIX B: CONSTRUCTION OF ROTATOR.....	51
	APPENDIX C: CONSTRUCTION OF ANTENNA.....	52
	APPENDIX D: ASSEMBLY OF ROTATOR AND ANTENNA .....	53
	APPENDIX E: ARDUINO CODE.....	54



## LIST OF FIGURES

Figure 3.1: Two Line Element [11] .....	15
Figure 4.1: Block Diagram of Ground station .....	16
Figure 4.2: Flowchart for Rotator .....	18
Figure 4.3: Comparison of ground station with and without SDR. [7].....	22
Figure 4.4: RTL-SDR .....	22
Figure 4.5: Raspberry Pi Model B+ .....	23
Figure 4.6: Arduino Pro Nano .....	24
Figure 4.7: Stepper motor .....	24
Figure 4.8: A4988 motor driver .....	25
Figure 4.9: LM2596 Buck Convertor .....	25
Figure 4.10: N-type RF connectors.....	26
Figure 4.11: Nano VNA.....	26
Figure 5.1: Helical Antenna Design.....	29
Figure 5.2: Helical Antenna in Free Space (left) with voltage source given in 4nec 2 (right) .....	29
Figure 5.3: Swr and Rdf Performance of an Antenna.....	30
Figure 5.4: CAD model of Helical Antenna .....	31
Figure 5.5: Making of Helical Antenna (left) and Completed Helical Antenna (right) .....	31
Figure 5.6: Brass Plate, wire and N -type connector being soldered (Left) Impedance Matching Being Checked (Right) .....	32
Figure 5.7: Verification of SWR and Frequency of Antenna .....	33
Figure 5.8: CAD assembly model of azimuthal rotator (left) and elevation rotator (right) .....	34
Figure 5.9: Assembly of rotator (left) and completed rotator (right).....	35
Figure 5.10: Schematic of the circuit board of rotator controller .....	36
Figure 5.11: Front View Of Rotator Controller PCB (Left), Back View of Rotator Controller PCB (Right) .....	36
Figure 5.12: Installation of SatNOGS Imager on Raspberry Pi.....	37
Figure 5.13: Basic Configuration for SatNOGS-Client.....	38
Figure 5.14: Ground Station as seen in SatNOGS Network.....	39

Figure 5.15: Final View of the Ground Station .....	40
Figure 5.16: Observation summary of CELESTA satellite .....	41
Figure 5.17: Waterfall diagram of CELESTA .....	42
Figure 5.18: Observation summary of DUCHIFAT-3 satellite .....	43
Figure 5.19: Waterfall diagram of DUCHIFAT-3 .....	44

## **LIST OF ACRONYMS AND ABBREVIATIONS**

COTS	Commercially off-the-shelf
DB	Database
DC	Direct Current
GEO	Geostationary Earth Orbiting
GPIO	General Purpose Input/output
GPS	Global Positioning System
HF	High Frequency
IIC	Inter-Integrated Circuit
IR	Infra-Red
LEO	Low Earth Orbiting
M&C	Monitoring & Control
MEO	Medium Earth Orbiting
NEC	Numerical Electromagnetic Code
OS	Operating System
PCB	Printed Circuit Board
RF	Radio Frequency
RPi	Raspberry Pi
RTL	Register Transfer Level
SatNOGS	Satellite Networked Open Ground Station

SISO	Single Input Single Output
SDR	Software Defined Radio
SLL	Side Lobe Level
SPI	Serial Peripheral Interface
SWR	Standing Wave Ratio
TEC	Total Electron Content
TNC	Terminal Node Controller
TT&C	Telemetry, Tracking & Command
TTL	Transistor - Transistor Logic
UART	Universal Asynchronous Receiver-Transmitter
UHF	Ultra-High Frequency
USB	Universal Serial Bus
VHF	Very High Frequency
VNA	Vector Network Analyser

# CHAPTER ONE: INTRODUCTION

## 1.1 Background

Satellite communication is a rapidly expanding field that has numerous applications, such as weather monitoring, global positioning and television broadcasting. Satellites orbiting the Earth are divided into three categories: LEO, GEO and MEO. LEO satellites, which orbit the Earth at altitudes ranging from 160 to 2000 km and complete an orbit around the Earth in about 90 minutes, require ground stations with rotators and helical antennas to communicate with them. [1]

A ground station consists of two segments: the user segment and the ground segment. The user segment comprises the satellite and the equipment required for signal transmission and reception. The ground segment includes a range of equipment such as antennas, receivers, transmitters and rotators, which are crucial for satellite communication. The ground station receives and transmits signals to and from the satellite. Ground stations with rotators and helical antennas capture the satellite signal, track the satellite's movement and process and decode the signal. [2]

Ground stations with rotators and helical antennas are essential for applications that rely on satellite communication. LEO satellites are widely used for remote sensing, weather monitoring and environmental research, among other things. [3] Ground stations with rotators and antennas are critical for tracking and communicating with small satellites in Low-Earth orbit, such as CubeSat, which are becoming increasingly popular for scientific research, education and technology development.

The aim of this project is to develop a low-cost ground station with readily available components that interfaces with the SatNOGS database, a global network of ground stations that track and receive satellite signals. By designing and implementing this ground station, the project aims to demonstrate the practicality and usefulness of ground stations with rotators and helical antennas for LEO satellites, as well as provide a blueprint for others interested in building similar ground stations.

This report provides a detailed description of the design and implementation of the ground station with a rotator and a helical antenna using the SatNOGS database. The report covers the design of the antenna, the rotator and the electronics used to control the rotator and communicate with the satellite. Additionally, the report discusses the significance of ground stations with rotators and antennas for LEO satellites and their potential applications in satellite communication.

Ground stations with rotators and antennas play a vital role in real-time monitoring and control of LEO satellites, making them an essential component of space exploration and research. The report explores the benefits and potential applications of ground stations with rotators and helical antennas and their significance for the field of satellite communication. The project aims to demonstrate the importance of ground stations with rotators and helical antennas for communicating with LEO satellites, as well as the potential for low-cost, readily available components to enable broader access to satellite communication technology.

## **1.2 Problem Statement**

The following challenges may be part of the problem statement for designing and building a ground station and satellite tracking rotator for low Earth orbit satellites.

- i. Low Earth orbit satellites must be able to be precisely tracked by the rotator as they move across the sky.
- ii. The rotator and ground station must be durable and capable of withstanding extreme weather.
- iii. Cost-efficient and reasonable construction should be used in the design.
- iv. Multiple satellites and communication protocols should work with the rotator and base station.
- v. The design should make it simple to carry and set up in various locations.

## **1.3 Objectives**

### **1.3.1 Primary Objective**

The main objective of the project is to design and construct a satellite tracking rotator and ground station for Low Earth Orbit Satellites

### **1.3.2 Secondary Objectives**

- i. To receive input from a LEO satellite and decode the information in order to obtain information from the satellite being tracked.
- ii. To design and simulate the antenna for receiving and tracking data from satellites at a specified range of an antenna.
- iii. Using Arduino and with the help of 3D printed parts, creating a two-axis rotating system rotor enabling azimuthal and elevation antenna movement.

## **1.4 Application**

- i. For operations like amateur radio, remote sensing and satellite photography, the rotator and ground station can be used to locate and connect low Earth orbit satellites.
- ii. Data from low Earth orbit satellites can be gathered using the rotator and ground station for a variety of purposes, including weather forecasting, environmental monitoring and resource management.
- iii. The rotator and ground station can be utilized for educational and outreach initiatives, showing the general public and students the capabilities of satellite tracking and communication.
- iv. For situational awareness and collision avoidance in space, the rotator and ground station can be combined to track and monitor space debris.

## **1.5 Features**

- i. A lightweight and effective method of tracking low Earth orbit satellites can be achieved by incorporating helical antennas into the rotator and ground station designs.
- ii. Helical antennas offer a high gain and a wide bandwidth, which can boost the signal and enhance the system's tracking performance
- iii. It is possible to build the user interface for the rotator and ground station so that they are simple to use and manage.

## **1.6 Feasibility Analysis**

### **1.6.1 Economic Feasibility**

A complete examination of the project's costs and benefits and risk analysis can be used to assess the project's economic feasibility up to this point.

The project's initial capital costs, which comprise the price of system design, building and installation, are Rs. 25,000. The anticipated Rs. 15,000 in continuous operational expenses, which include personnel, repair and utilities, would be needed up until the project is finished. The project is expected to cost Rs. 40,000 in total over the course of a year.

Since the project is being conducted for educational and scientific goals, its advantages center on providing academics and students with simple access to satellite tracking and data collecting improves research opportunities and helps students comprehend themes related to orbital determination.

Once the ground station and rotator mechanism is installed at the Pulchowk campus and is up and running, the following mechanism can be employed to generate data from low earth orbit satellites at various locations inside the valley or even at IOE's other constitutional campuses.



Additionally, it might produce income based on the demand of data subscribers or data clients. The proposed project has strong predicted returns on investment indicating that it is economically feasible.

### **1.6.2 Technical Feasibility**

To ensure that the project can be implemented successfully, the project's resources, including personnel, equipment and funding, have been assessed. Potential risks related to the technology, such as breakages, compatibility problems and security risks, have been identified and discussed. The technological feasibility of the proposed project is favorable, with the required technology being readily available, the required resources in place and the potential risks manageable according to the results of this evaluation, which show that the risks associated with the technology are low and can be effectively managed through the implementation of appropriate mitigation strategies.

### **1.6.3 Operational Feasibility**

The project structure has been evaluated to ensure that the necessary resources are available to support the project's objectives. The needs for the project's process, such as those for planning, executing and monitoring the project, have been identified and evaluated. The project can be supported by the current procedures and any additional processes that are required may just be added. The overall evaluation indicates that the project can be supported successfully by the team's current structure with pure dedication, that the delay up to this point can be fixed in the coming days with few modifications needed and that it can also be completed within the allotted time period with little disruption to ongoing activities.

## **1.7 System Requirements**

### **1.7.1 Hardware Requirements**

The following are the hardware required for our project:

- Raspberry Pi 3B+

- Arduino Nano
- Stepper Motor (Nema-17)
- Stepper holder bracket
- Motor driver (A4988)
- Speed sensor (LM393)
- RTL – SDR
- Buck Converter
- Aluminum pipes/rods
- Aluminum T-slots
- Hidden connectors
- Coaxial cables (50 & 75 ohms)
- Threaded rod (M5)
- Belt and stepper pulley
- Bearings ( 6008zz & 625zz )
- Nuts & Bolts
- SMPS Power Supply
- 3D Printed parts

### **1.7.2 Software Requirements**

The following are the software requirements needed for our project.

- Raspbian OS
- Autodesk Fusion 360
- Catia V5
- 4nec2
- SD Card Formatter
- Win32 Disk Imager
- Arduino IDE
- SatNOGS DB

## CHAPTER TWO: LITERATURE REVIEW

### 2.1 Ground Station

Recent development in space industry has led to more research and advancement in ground station. A paper published by Tuli et al. [4] shows a generic design for a low cost ground station used for communicating with nanosatellite called CanX-2. This ground station uses S band/UHF/VHF band frequencies and can receive and transmit signals from ground station. The Yagi-uda antenna with 435 MHz frequency in UHF range used in this ground provides bi-directional functionality meaning it can both uplink and downlink the signals whereas the 2.08m S-band dish antenna is used for downlink proposes only. The rotator used in this ground station is a COTS product manufactured by AlfaSpid Radio. It provided a 180-degree travel in the elevation axis and has limit switches that cut power to the motor in case of over travel. The rotator is controlled by a custom microcontroller board that offers greater flexibility, reliability and cost-effectiveness compared to commercial alternatives. It uses an ATmega32, a simple 8-bit microcontroller, to interface with the rotator position switches, ground station TNC computer for accepting commands and the motor controllers driving the rotator. The central component of the ground station is a TNC based on a GFSK modem and a single-board ARM computer running Linux, responsible for (de)modulation, hardware control and orbital tracking. The GUI software running on standard PCs serves as the satellite's operating environment and allows remote control of experiments onboard the satellite. The software and hardware design is modular to allow for easy modification and upgrades for future missions. The ground station has been tested with successful tracking and receiving signals from other satellites and is ready for full operation after CanX-2's launch.

Recent studies published by Ubong et al. [5] outlines a low-cost ground station for nanosatellite missions that aims to promote educational advancement and interest in space engineering among students and serve as a laboratory for future space technology development. In this study, the ground station is designed to track and receive signals from amateur band satellites in the VHF and UHF range, utilizing Yagi-Uda antennas, a Low Noise Amplifier, a rotator, an IC-910H transceiver and a computer for

automation. The paper also highlights the use of LNB which increased the system performance and decreased the signal noise and corrected the effective orientation of antennas for capturing the signals from satellite. It also discussed about the implementation of the system in an economical point of view to provide learning opportunities for students in the development of future satellite communication systems and space technology.

A paper published for a university satellite missions from Georgia Tech [6] highlights a uniform ground station design for small satellite missions at Georgia Tech, which will consist of three distinct ground stations, utilizing commercially available operations software, software-defined radios and open-source tracking software. It also utilizes different frequency antennas for uplink and downlink facilities like S-band, VHF and UHF located at different places. The station includes Yagi-Uda antenna for VHF/UHF and S-band parabolic Dish for S-band frequency. This station has a COTS rotator that moves the antenna in azimuth and elevation direction. Its uses different hardware like band pass filters, SDR dongle, LNA for signal processing. The signal processing software used is GNU Radio and tracking software used is Gpredict. Therefore, the ground station architecture at Georgia Tech is designed to be adaptable to different missions by utilizing customizable operations software, software defined radios and open source tracking software that is user-friendly.

A paper [7] published in an international conference talks about the potential of SDR in antenna and space applications and examines its use in demodulating signals received from Low Earth Orbit (LEO) satellites in a ground station. It evaluates the feasibility of implementing SDR technology for space purposes by building a cost-effective VHF/UHF ground receiving station to support university missions. The report presents the results of experiments carried out on OSCAR microsattellites and the MARSAT-1 satellite to demonstrate the effectiveness of the SDR technology in space applications.

## **2.2 Rotator Design**

This paper [8] talks about building a low cost amateur rotator for an economical ground station. This project was realized due to sheer high cost of existing commercial rotators like YAESU G5500, SPID Electronic RAS, OR2800P and MT3000 which cost more

than \$1800. It also talks about manual rotators which are steered using hand and automatic rotators. The automatic rotator used here is a RC\_FNQ. This is a custom made rotator that has an azimuthal and elevation motors and a series of metal gears used for positioning the antenna. It also talks about the difficulties faced for adjusting a support for the bearing as they were not available commercially. It showed that a low cost rotator can be built that cost over 8 times less than the commercial one and works just as good as the latter.

This study [9] focuses on the development of an automatic antenna pointing system that utilizes a compass sensor (CMPS10) for feedback and predicts satellite movement through TLE calculations. The system was tested by tracking the NOAA 19 satellite and showed an average deviation of 5 degrees from the predicted angle. The rotator system was made using two gears and two DC motors with custom frames and underwent successful testing. This report also outlined the results of three tests performed on the rotator system: power supply circuit, elevation and azimuth power and mechanical tests. The rotator was also tested for different satellite tracking software which produced different results for various software.

This research [10] focuses on the design of Azimuth/Elevation Rotor and the interface design of its Schematic Controller. The research also talks about the design and set up an automatic antenna tracking system for communication with amateur radio satellites (OSCAR) for voice and data transmission. The rotator system in this paper uses a 24 V-DC motor with a gear ratio of 10:1 on the gear drive shaft. The rotator uses an ATmega328 microcontroller for providing elevation and azimuth angles. The microcontroller board controls the rotator and provides an improved option with increased flexibility, reliability and cost-effectiveness compared to other options. The tracking software used was Orbitron software as it was open source.

### **2.3 Antenna Design**

A cylindrical metal conductor that has been twisted in the shape of a helix makes up the standard helical antenna for ground stations. The "helix," a cylindrical conductor that serves as the radiating component and is often composed of conductive metals like copper or aluminum. [11] The coaxial wire that connects to the helix's core and feeds

the antenna with radio signals does so by feeding it. Circular polarization, which is preferred for satellite communication, is provided by the helical antenna, which operates in the UHF or VHF frequency band.

The communication with the satellites was done by the means of two antennas, one for VHF communication and another for UHF communication. The antenna for VHF communication uses a  $\lambda/4$  ground plane antenna that specifically communicates at 145 MHz band. In other hand, the UHF one uses a  $3\lambda/4$  parasitic circularly polarized vertical antenna that precisely communicates at 436 MHz band. Both the antennas are connected to a Low noise amplifier through a RG58 cable. Additionally, two extra antennas are also used for 2400 and 10500 MHz as mentioned in work done A.Done [12].

Since, a sole antenna cannot transmit in all frequency bands, various types of antenna are used for different stations. In this paper written by Choi et al. [13] three different antennas are used for three different stations. The CCRF station uses an S-band antenna system. It communicates in the range from 2200 to 2310 MHz. The antenna also includes a low noise amplifier designed as a receiver, high power amplifier designed for transmit. It is mounted on top of rotator adept of maneuvering the antenna at the speed of  $60^\circ/\text{s}$  in azimuth and elevation direction. These all are combined into an antenna pedestal. The VL station uses a VHF band yagi antenna system. It is a circularly polarized antenna system. It is an uplink system capable of communicating in the range of 144-148 MHz of VHF band. The uplink signal is amplified two times via two distinct amplifiers since it was more economical than using a single large amplifier. The antenna also includes a band pass filter for filtering of signals. Likewise, the MK station uses the same components as the transmitter station (VL) but it uses an UHF frequency band. It is a downlink system capable of communicating in the range of 430-438 MHz of UHF band. This station operates two yagi uda antennas which are connected as a phase array for increment in antenna gain.

Helical antennas were designed and made with a high-gain in mind, which makes them perfect for long-distance communications. In comparison to other kinds of directional antennas, helical antennas are generally compact in size, making them simple to integrate into systems. They are also relatively easy to build. The most recent version

of a well-known numerical electromagnetic code program (4nec2) was used to do several iterations and optimization for the design simulations for helical antennas. To get the required performance out of antennas, design parameters were first computed and then optimization algorithms were made to fit their needs. [11]

Radiation efficiency and size/weight are crucial factors to consider when selecting an antenna. In the case of helical antennas, radiation efficiency depends on design parameters and the quality of materials used. High radiation efficiency means that more power supplied to the antenna is radiated as electromagnetic energy, resulting in more efficient transmission and reception of signals. Helical antennas are relatively small and lightweight, making them ideal for portable or mobile applications. This combination of high radiation efficiency and compact size makes helical antennas a popular choice for satellite tracking rotator and ground station systems for low Earth orbit satellites [12]

## **2.4 Tracking software and Controller**

After the ground station is constructed, it becomes a stationary part most of the time and will have limited resources, thus the usage must be carefully planned for it to cover the necessary needs of the mission. The antennas can only point at one point in the sky and so tracking multiple satellites in different regions is not possible at the same time. Using orbital propagation software, the access time of the satellite must be predicted in advance for the ground stations. Conflicts between different satellites shall be solved in advance. Nowadays, there are many commercial software that helps with the planning of the ground station and provides information on the constraints of the system such as the satellite to be tracked and some data about the ground station. And it can also help the operator show the conflicts that need to be solved during operation of the ground station. [13]

Since all the activities that are to be performed by the satellite are planned in advance, it also includes tuning the frequency to communicate with the ground station. The procedures to be executed shall be decided and prepared beforehand for each communication opportunity. It also includes the generation of the telecommand sequences that is to be sent to the ground station and the amount of telemetry that is to

be retrieved during communication with the ground station. And thus the antennas will point to the predicted acquisition of signal (AOS) azimuth calculated from the TEL. [14]

Regarding a tracking software, there are many choices to choose from. Most of them are open source and inexpensive with respectable performance [9]. As specified previous in Fischer & Scholtz [9], the signal processing segment utilizes a mission specific TNC (Terminal Node Controller) which also include a modulator. It specially designed for uplink purposes. The conversion to UHF is done via a radio transceiver. For downlink purpose, a commercially available Datum Systems PSM-500 modem is used where it does the demodulation portion and then sends the digital data to the TNC

A PEB20525 Serial Communications Controller (SCC) takes the core of the control panel that gives interfaces to the dual synchronous HDLC using a PC/104 bus. It also has an interface for rotator namely “RS-422” that communicates with the rotator controller that provides veering commands. Additionally, it has a modem interface for each frequency bands providing a path for TNC software to direct commands to spacecraft and also accounts for the Doppler shift or trigger PTT. It also has Scrambler\descrambler to make sure that adequate bit transitions are in the received data stream. In case of software, an application called Nanosatellite Interface Control Environment (NICE) which is based on the Nanosatellite Protocol which itself is built on the AX.25 link layer protocol which is most widely used in the satellite communication community which is printed in the paper published by Tuli et al [4].



## **CHAPTER THREE: RELATED THEORY**

### **3.1 Frequency Bands**

UHF (Ultra High Frequency): UHF is a radio frequency range between 300 MHz and 3 GHz. It is commonly used for satellite communication because it can penetrate the Earth's atmosphere and is less affected by weather conditions than VHF.

VHF (Very High Frequency): VHF is a radio frequency range between 30 MHz and 300 MHz. It is also used for satellite communication but is more affected by atmospheric conditions than UHF.

Other Frequency Bands Used for Satellite Communication: Different frequency bands are used for different types of satellite communication. For example, the L-band (1-2 GHz) is commonly used for mobile satellite communication, while the S-band (2-4 GHz) is used for satellite-to-satellite communication and some telemetry and remote sensing applications. The X-band (8-12 GHz) and Ku-band (12-18 GHz) are commonly used for satellite communication in fixed, broadband and direct-to-home (DTH) applications. The Ka-band (26.5-40 GHz) is used for high-throughput satellite (HTS) communication and other data-intensive applications.

Understanding these frequency bands and their applications is important for selecting the appropriate frequency range for a satellite tracking rotator and ground station for LEO satellites. The choice of frequency band will depend on various factors, including the type of satellite, the application and the available technology. [14]

### **3.2 Standing Wave Ratio and Impedance Matching**

SWR (Standing Wave Ratio) is a measure of how well an antenna is matched to the transmission line and the transmitter/receiver. It is defined as the ratio of the maximum amplitude of the standing wave pattern to the minimum amplitude. A higher SWR indicates a poorer match between the antenna and the transmission line, which can result in signal loss, reduced efficiency and potential damage to the transmitter or receiver.

VSWR (Voltage Standing Wave Ratio) is a related concept to SWR, but it is expressed in terms of voltage instead of power. It is defined as the ratio of the maximum voltage to the minimum voltage along the transmission line. A VSWR of 1:1 indicates a perfect match between the antenna and the transmission line, while a higher VSWR indicates a poorer match. [15]

Impedance matching is the process of adjusting the impedance of the antenna to match the impedance of the transmission line and the transmitter/receiver. This is important because if the impedance of the antenna and the transmission line are not matched, then some of the energy from the transmitter will be reflected back towards the transmitter, leading to reduced efficiency and potential damage. There are various methods of impedance matching, including using a matching network or adjusting the physical dimensions of the antenna. [16]

### **3.3 Two Line Element**

Two-Line Element (TLE) is a standard data format used to convey orbital information for Earth-orbiting objects, such as satellites and debris. It consists of two lines of data, each containing a specific set of parameters that describe the object's orbit. The first line contains information about the object's name or number, its international designator and its epoch (the time at which the TLE was generated). The second line contains information about the object's position and velocity, including its inclination, eccentricity, argument of perigee, right ascension of the ascending node, mean anomaly and mean motion. It is generated using a mathematical model of the object's orbit, which is typically updated every few days based on observations from ground-based or space-based tracking systems. The accuracy of TLE degrades over time due to the complex nature of orbital mechanics and external forces such as atmospheric drag. They are used to track the position of a satellite in real-time, to predict when it will pass over a particular location and to determine its orbital characteristics, such as its altitude, velocity and orbital period. They are also used to predict and prevent collisions between satellites and debris in Earth's orbit. [11]

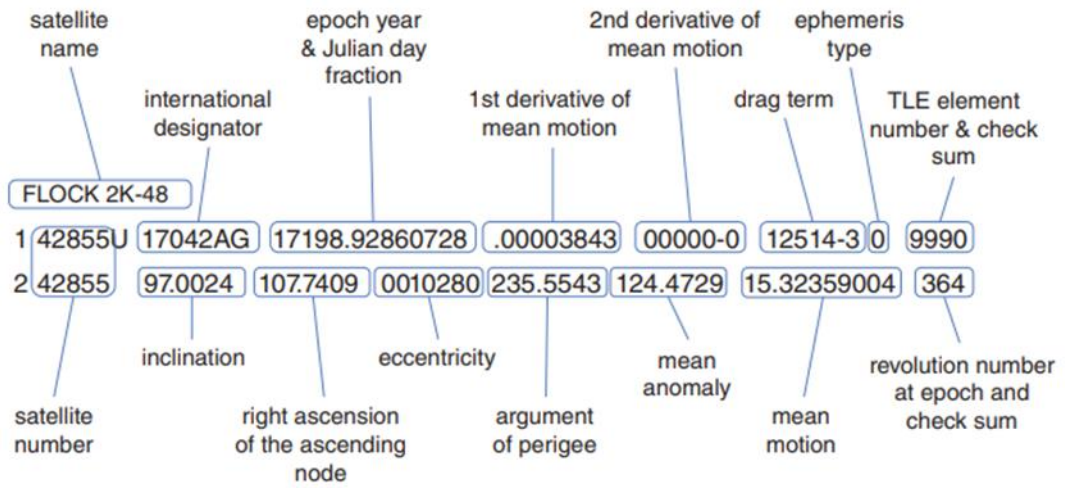


Figure 3.1: Two Line Element [11]

# CHAPTER FOUR: METHODOLOGY

## 4.1 Ground station

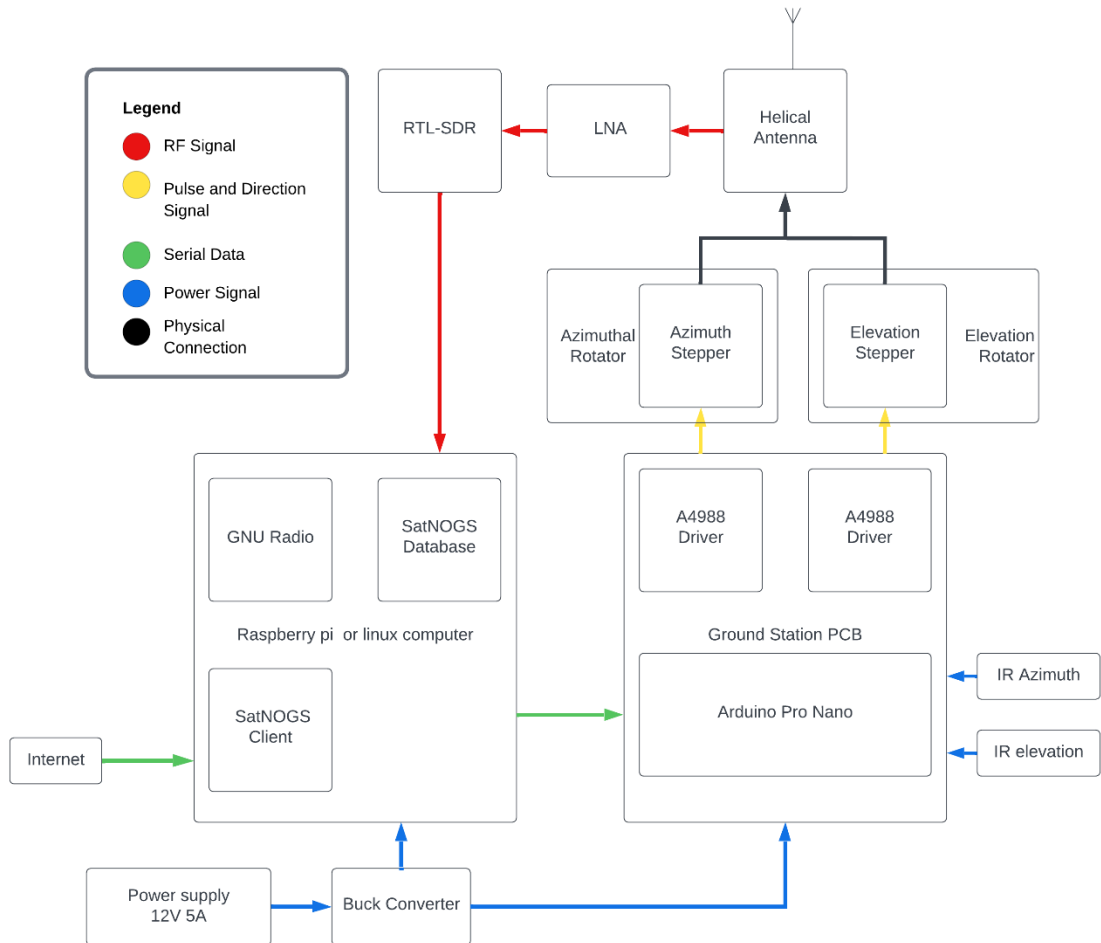


Figure 4.1: Block Diagram of Ground station

The above diagram depicts the design of a satellite ground station that utilizes a helical antenna. The antenna operates at 437 MHz frequency and has right-hand circular polarization. The design process and specifications are provided below.

In order to receive signals from passing satellites, the antenna must be pointed in their direction. This requires tracking the satellite and adjusting the antenna's position accordingly. Two stepper motors (12 volts each) are used for this purpose: one for azimuthal movement and the other for elevation movement. The amount of movement is determined through the SatNOGS database, an open-source software that tracks

satellites and provides information such as azimuth and elevation angles. To access this information, the ground station must be registered on the database and the commands must be loaded onto a Raspberry Pi. Once the Raspberry Pi is connected to the internet and the satellite ground station is registered in the SatNOGS online database, the SatNOGS site provides information on the various satellites in the area, including their UTC time and visibility. Users can then select the satellite they want to track and the SatNOGS database provides its elevation, azimuth angle and path. This information is then downloaded into the Raspberry Pi and passed on to the Arduino, which uses the motors to adjust the antenna and track the selected satellite. Once the antenna has been designed, the SWR and impedance are measured using an antenna analyzer. The output of the antenna is then fed into a Nooelec SDR which uses software defined radios to process the signals. It is connected to a Raspberry Pi. On the Raspberry Pi, the GNU radio companion software is installed and used to demodulate the signal, producing the desired result, which is saved and displayed on the screen.

## 4.2 Flowchart for rotator

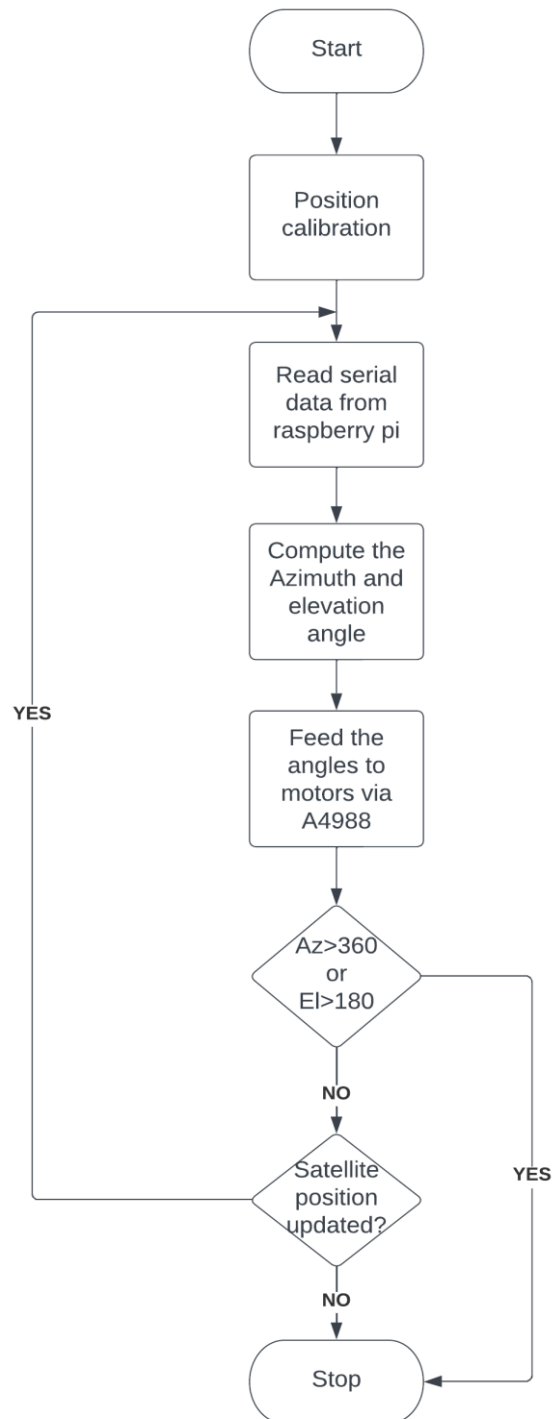


Figure 4.2: Flowchart for Rotator

The flowchart depicts the process by which the Arduino directs the stepper motors to track a satellite in the sky. The Raspberry Pi receives the azimuth and elevation angles

from the SatNOGS database and sends the information to the Arduino Pro Mini through serial communication via a USB cable. The Arduino calculates the necessary steps and direction for the motors to move and sends pulses to the A4988 motor driver to control the motors and track the satellite. The endstop limit switches attached to the rotator stops the motor from operating if the azimuth and elevation angle overshoots the limit. This process continues until the satellite is no longer visible.

### 4.3 Design Parameters for Antenna

- i. Centered Frequency ( $f$ ) = 437 MHz
- ii. Number of turns ( $n$ ) = 8
- iii. Wavelength ( $\lambda$ ) = 686 mm = 0.686m
- iv. Spacing between turns ( $S$ ) = 171.5 mm  $\approx$  17.2cm
- v. Axial length =  $nS$  = 137.6
- vi. Gain Antenna ( $G$ ) = 13.8
- vii.  $Z$  (Characteristic Impedance) = 150 Ohms
- viii. HPBW (Half Power BW) = 36.8°
- ix. Winding Diameter ( $D$ ) = 220 mm = 0.22m
- x. Circumference of helix =  $C = \pi D = 3.14 \times 220 = 690\text{mm}$
- xi. Winding Length = 5660mm
- xii. Boom Length = 1400mm [16]

### 4.4 Basic Geometry of the Antenna

- Three CPVC pipes each of 140cm is marked as A, B & C respectively. Top edge of pipe A is considered as  $\pm 0$ .
- The winding location is then marked along pipe A, creating a spacing of 17.2 cm for each turn, while pipe B is initially offset by a third of 17.2 cm from pipe A's initial position and then 17.2 cm are marked along pipe B.
- Similar to Pipe C, Pipe C is offset by  $2/3$  of 17.2 from Pipe A's initial position, presuming that Pipe C's uppermost point has been marked with 17.2 cm for 8 rounds.

- Brass plate is soldered at the end of the Winding. In between the N-type is positioned and the end of the Winding.' [12]

#### **4.5 Steps for the Design of Helical Antenna**

- The cutting and drilling of aluminum is done that is used to provide support to antenna and house the rotator mechanism.
- The PVC pipes which are used to support the helical coil of wire are cut to shape and drilled.
- The reflector is prepared with a wire grid cut to the above given diameter.
- 2mm thick copper wire is cut to length to make the winding for the helical antenna. The wire is configuring and aligned to the above given parameters.
- The reflector is attached to the base of the helical antenna and the rotator mechanism is attached.
- A SO-239 connector is placed at the base of the reflector and a 50-ohm co-axial cable takes the output from the antenna and passes it to the SO-239 connector.
- The SO-239 connector is connected to a RTL SDR that sends the received signal to the computer. [12]



## 4.6 Equations for the Helical Antenna

$$G = 10 \times \log_{10} \left( 10.8 + 10 \times \log_{10} \left( \frac{C^2 SN}{\lambda^2} \right) \right) \quad (3.1)$$

$$Z = \frac{150}{\sqrt{\frac{C}{\lambda}}} \quad (3.2)$$

$$HPBW = \frac{52}{\frac{C}{\lambda} \times \sqrt{n \times \left(\frac{S}{\lambda}\right)}} \quad (3.3)$$

Equation (3.1) describes the theoretical gain of the helical antenna in dBi. It is calculated by the help of the circumference, winding spacing and number of turns and the wavelength of the helical antenna [16]

Equation helps (3.2) in the calculation of the theoretical impedance of the helical antenna in ohm which is give with the help of the parameters: circumference and wavelength of the helical antenna. [16]

Similarly, equation (3.3) gives the half power beam width of the helical antenna in degrees which is calculated by the help of the winding circumference, wavelength, winding spacing and number of turns of the helical antenna. [16]

## 4.7 Hardware used

### 4.7.1 RTL-SDR

RTL-SDR is a radio communication system which implements the use of software components to replace the hardware components of the ground station. It uses an analog to digital converter close to the antenna and then sends the digitized data to the processing unit. It gives flexibility to change the modulation, signal and bandwidth making it compatible to use it for different missions. The signal processing is forwarded to the general-purpose processor and thus a wide range of radio protocols can be received and transmitted. RTL-SDR can help with filtering band selection, power

amplification for transmission, low noise amplification for reception and analog to digital conversion and vice versa. GNU Radio is a free open source processing software for missions that makes the development of the complete SDR system easier. [7]

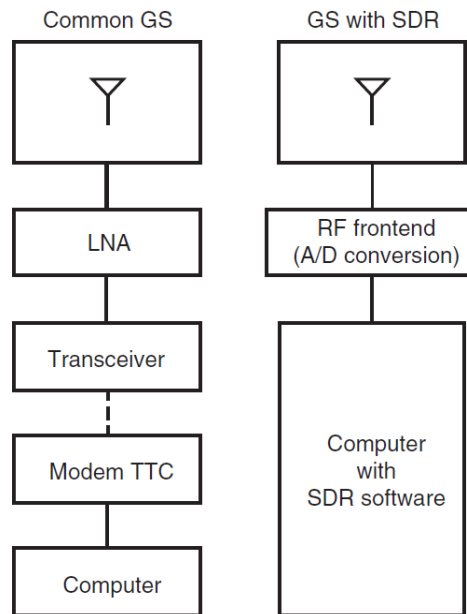


Figure 4.3: Comparison of ground station with and without SDR. [7]



Figure 4.4: RTL-SDR

## 4.7.2 Raspberry Pi



Figure 4.5: Raspberry Pi Model B+

Raspberry Pi is a microcontroller which helps to process the data obtained from the SatNOGS database and calculate the position of the satellite and get the necessary parameter of azimuth and elevation angle and send it to the Arduino board. We are using Raspberry Pi Model B+ for our project. The imager for the Pi is available from the SatNOGS and thus setting it up makes it easier because it comes pre-build and we only have to use the registered API key after creating a username in SatNOGS website and the setup becomes easier than it would have been without the help of SatNOGS.

### 4.7.3 Arduino Pro Nano

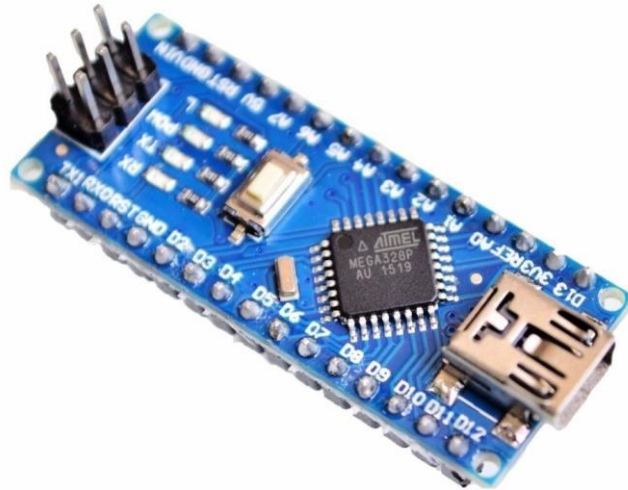


Figure 4.6: Arduino Pro Nano

Arduino Pro Nano is an open source electronic prototyping platform that consists of a physical programmable circuit board and an integrated development environment that is used to write and upload codes onto itself. The processed data from the Raspberry Pi is fed into the Arduino Pro Nano which is used to control the speed of the stepper motors that are used to move the antenna.

### 4.7.4 Stepper Motor

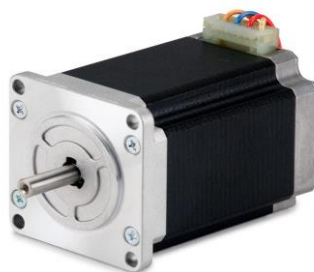


Figure 4.7: Stepper motor

Two 24 V stepper motor are used where one of them is used to move the antenna vertically and another is used to move the antenna horizontally

#### 4.7.5 A4988

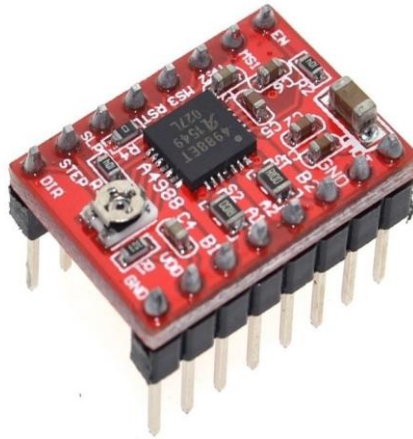


Figure 4.8: A4988 motor driver

A4988 motor driver is used to drive the stepper motor. It is used to control the direction and speed of the stepper motor.

#### 4.7.6 LM2596

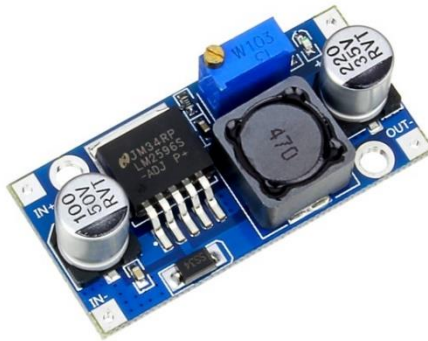


Figure 4.9: LM2596 Buck Converter

The LM2596 is a popular DC-DC step-down voltage regulator, also known as a buck converter. It is used to convert a higher DC voltage to a lower, regulated voltage. The LM2596 is capable of handling input voltages up to 40V and can output a regulated voltage between 1.25V and 35V with a maximum current of 3A.

### 4.7.7 N-type RF Connectors



Figure 4.10: N-type RF connectors

An N-type RF connector is a type of coaxial connector used for terminating coaxial cable and connecting radio frequency (RF) equipment, such as antennas, to transmitters or receivers. The N-type connector has a threaded coupling mechanism and is designed to operate over a wide frequency range, typically up to 11 GHz or more. It provides low loss and high shielding effectiveness, making it suitable for use in high-power and high-frequency applications.

### 4.7.8 Nano VNA

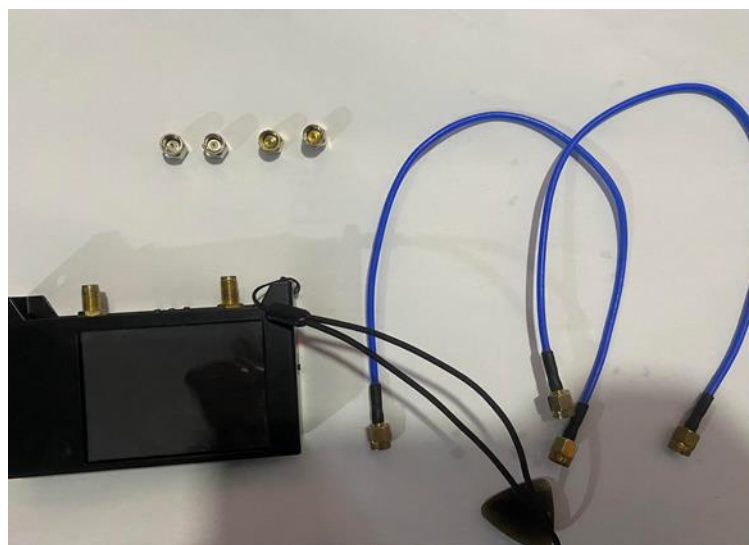


Figure 4.11: Nano VNA

A Vector Nano Analyzer (VNA) is a miniaturized version of a VNA designed specifically for measuring high-frequency properties of nanoscale electronic devices. The VNA measures the amplitude and phase of an RF signal as it passes through a device under test (DUT), as well as the amount of signal reflected back from the DUT. This information can be used to determine the complex impedance of the DUT at each frequency and calculate other important parameters, such as the scattering matrix or S-parameters.

## **4.8 Software Used**

### **4.8.1 GNU Radio**

GNU Radio is a free and open-source software development toolkit that offers signal processing blocks for creating software radios. By combining it with readily available, low-cost external RF hardware, software-defined radios can be developed. It handles all the signal processing and is utilized to write applications that transmit and receive data through radio hardware. GNU Radio has various blocks such as filters, channel codes, synchronization elements, equalizers, demodulators, decoders and more, which are commonly found in signal processing systems. It also includes a means of connecting these blocks and managing the flow of data from one block to another. In this project, the purpose of using GNU Radio is to demodulate the signal received by the satellite antenna, process the output data and store the data on a Raspberry Pi.

## 4.8.2 SatNOGS Client

SatNOGS Client is an open-source software tool used to control and operate satellite ground stations. The software is part of the larger SatNOGS project, which aims to create a global network of ground stations for receiving and transmitting data from satellites in orbit. The SatNOGS Client is designed to be used with a compatible ground station setup, which typically includes an antenna, a software-defined radio (SDR) and a computer. The client software runs on the computer and communicates with the SDR to receive satellite signals. The software also provides a user interface for controlling the ground station and managing data received from satellites.

Some of the features of the SatNOGS Client software include:

- **Schedule tracking:** The software can track satellite schedules and automatically adjust the ground station to point at the correct satellite at the right time.
- **Real-time data display:** The software can display real-time information about the satellite being tracked, such as signal strength, frequency and modulation.
- **Data storage and sharing:** The software can store and organize data received from satellites and users can share this data with others in the SatNOGS community.
- **Customization:** The software is open-source, so users can modify and customize it to suit their specific needs.



# CHAPTER FIVE: RESULTS AND DISCUSSION

## 5.1 Antenna Design

The helical antenna was initially CAD designed using FUSION 360 to assess whether or not the performance of the antenna satisfies the results that we need. Next, we run the simulation to examine the outcomes.

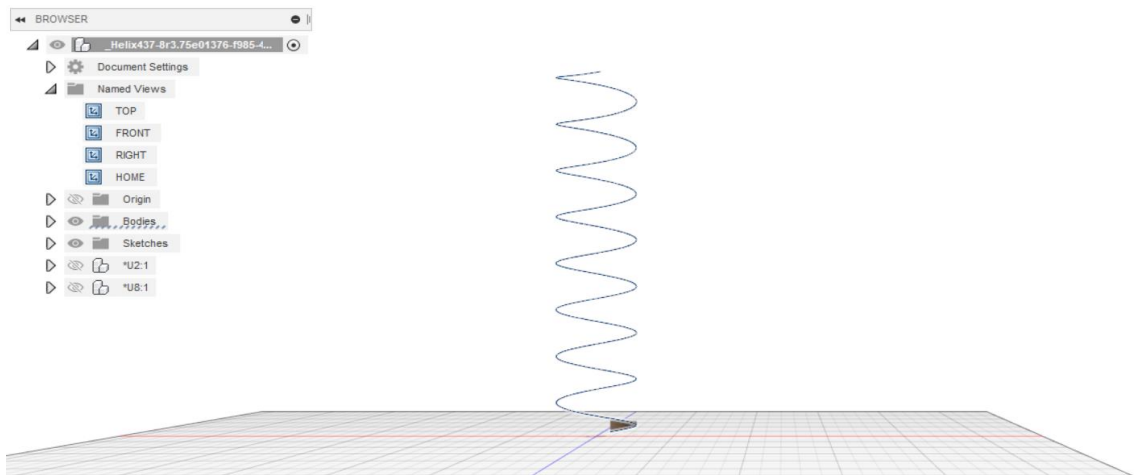


Figure 5.1: Helical Antenna Design

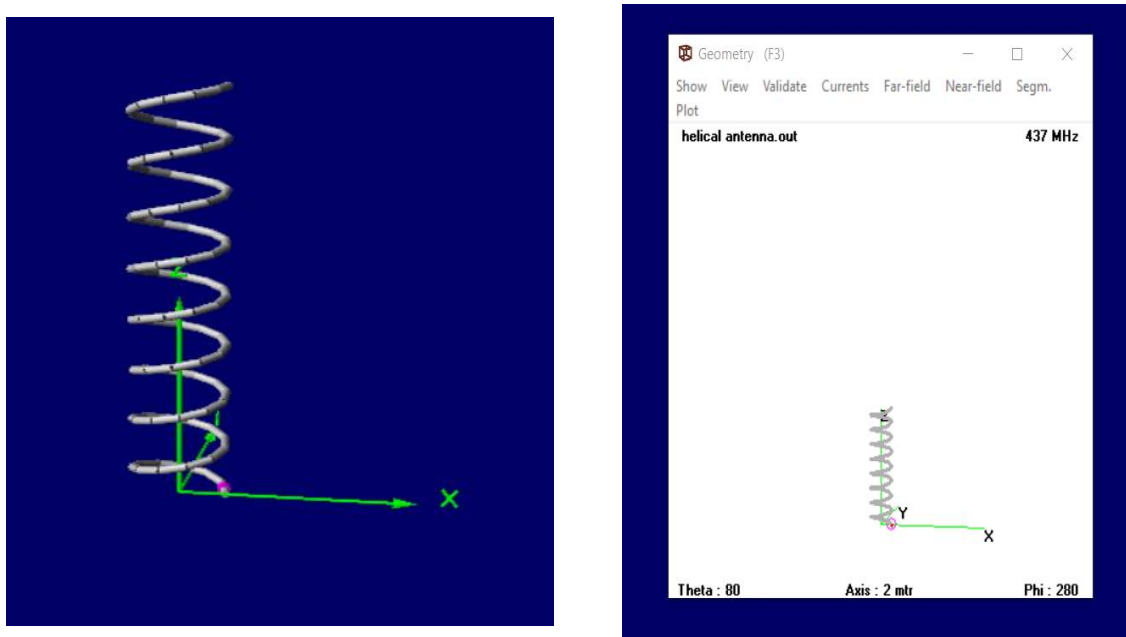


Figure 5.2: Helical Antenna in Free Space (left) with voltage source given in 4nec 2 (right)

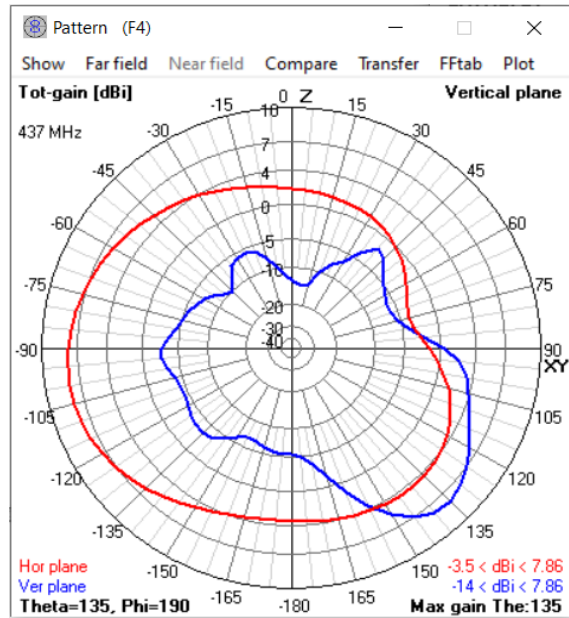


Figure 5.3: Swr and Rdf Performance of an Antenna

The simulation for an 8-turn helical antenna was carried out using the 4NEC2 software. Therefore, we acquire the vertical standing wave ratio of 4.32 and the rdf of 8.07 by using equal spacing procedures between the helix. A low SWR shows that the antenna and transmission line are well-matched, which leads to maximum power transfer and effective antenna radiation. In a similar vein, the helical antenna's lower Radiation Efficiency Factor value of 8.07 suggests that it is not very effective at converting the electrical power provided to it into radiated power. Thus, it is necessary to optimize the design, which includes, among other things, the diameter of the conductor, the height above ground, the size and shape of the helix, the number of turns and other factors.

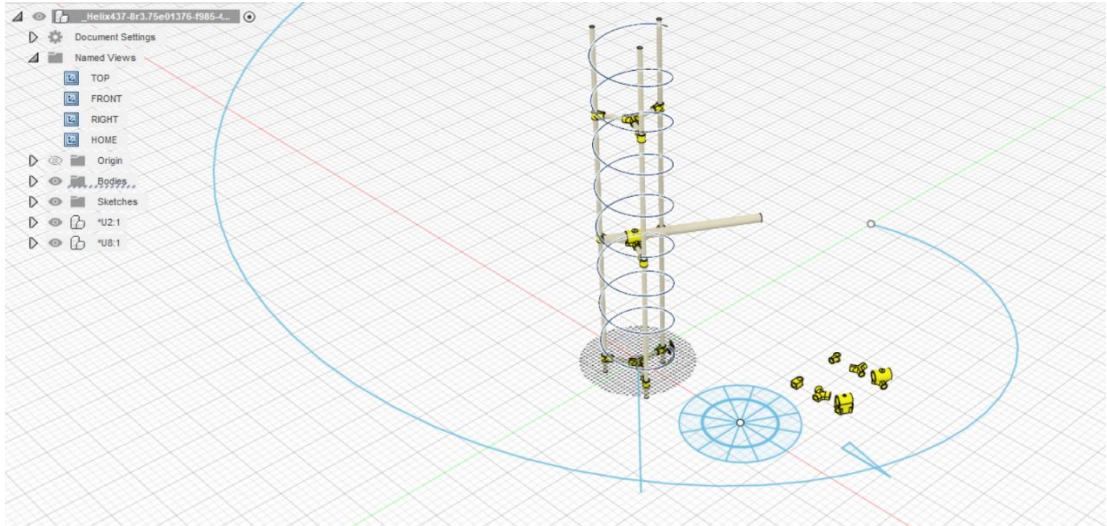


Figure 5.4: CAD model of Helical Antenna

Helical antenna was fabricated according to design specified in this report. The design was made in Fusion 360 software.



Figure 5.5: Making of Helical Antenna (left) and Completed Helical Antenna (right)

After completing the construction of the antenna, an N-type connector was affixed to the lower surface of the antenna, which is linked to the reflector. The attachment was performed by soldering the endpoint of the helical wire to a brass plate that had been cut into a triangular strip measuring 240cm by 45cm in base and perpendicular height

and then ensuring that the plate was properly connected to the end of the wire. The appropriate location for installing the N-type connector was then determined by referring to the CAD design and following the specifications provided by the SatNOGS database for antenna design. Finally, the N-type connector was installed close to the end of the reflector.



Figure 5.6: Brass Plate, wire and N -type connector being soldered (Left) Impedance Matching Being Checked (Right)

After installing the N type connector properly on the reflector, the RF test cable is used to connect the Nano VNA and an antenna. The cable helps to ensure that the RF signal being transmitted between the VNA and the device is not distorted or attenuated, which can affect the accuracy of the measurements. By using a properly matched and high-quality RF test cable, we can ensure that the measurements taken with the Nano VNA are accurate and reliable.

To calibrate the VNA, the Nano VNA is first connected to the calibration kit using the RF test cable. Then, the Nano VNA is turned on and the "Calibration" menu is accessed to select the desired type of calibration to perform, such as Open-Short-Load or through. The appropriate calibration standard, such as an open circuit, short circuit, or load, is connected to the Nano VNA. The on-screen instructions are then followed by pressing the "Calibrate" button and once the calibration is complete, the calibration file is saved for future measurements. It is important to ensure that the calibration process is performed accurately and carefully, as errors in calibration can lead to inaccuracies in measurements.

After calibration, the antenna is connected to channel one port of the VNA via RF cables and parameters such as reflection coefficient, Smith chart and SWR are observed. The reflection coefficient measures reflected signal due to impedance mismatch and the Nano VNA can use it to measure impedance, identify mismatches and optimize matching. The Smith chart is a graphical tool for analyzing transmission lines and impedance matching and the Nano VNA displays measurements that can be used to analyze impedance and identify mismatches. The SWR measures the efficiency of power transfer between the source and load and a high SWR indicates significant reflection due to mismatch.

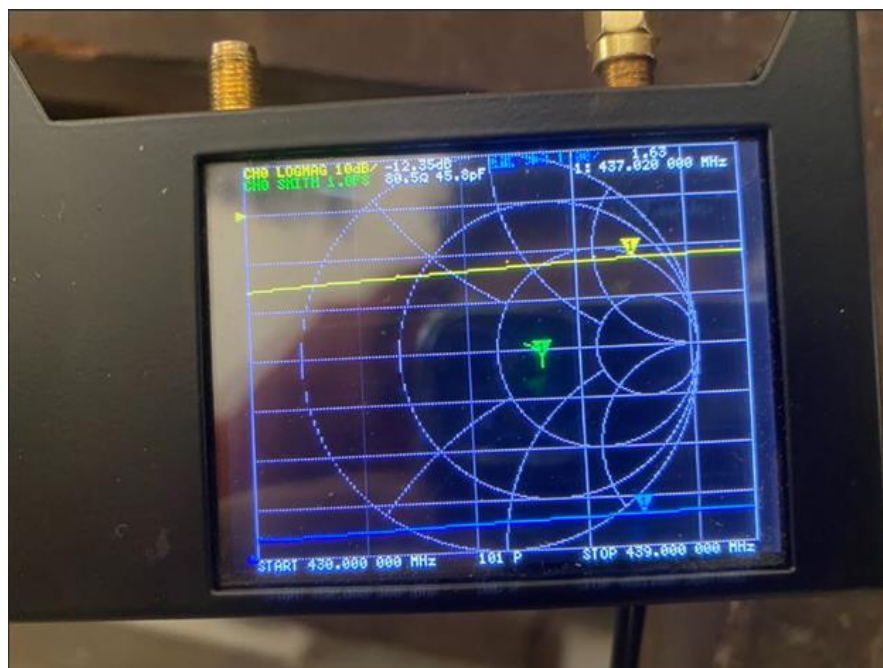


Figure 5.7: Verification of SWR and Frequency of Antenna

## 5.2 Rotator Design

The individual schematic designs for the Rotator were available in SatNOGS website. The CAD model of the assembly was created and the model was fabricated according to the design.

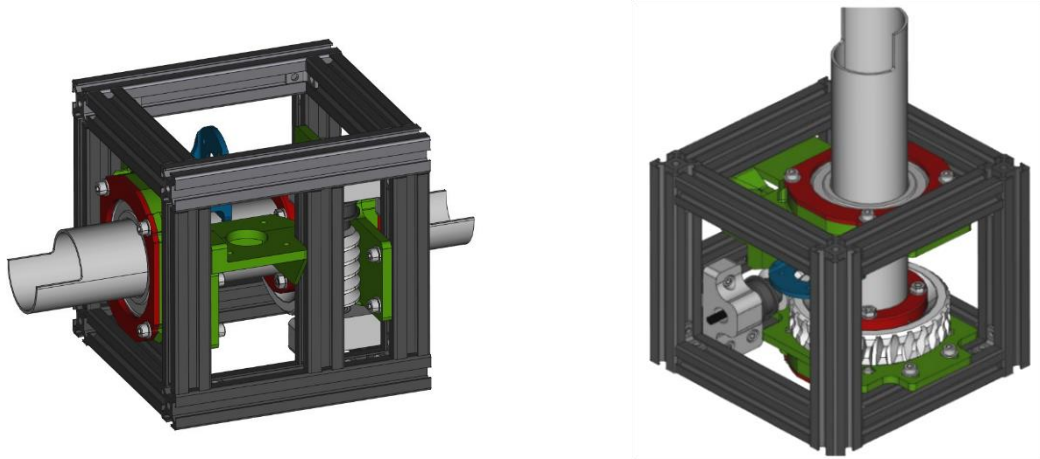


Figure 5.8: CAD assembly model of azimuthal rotator (left) and

For Rotator, the design were 3d printed using 3d printers available in our campus. Then using the 3d printed parts, the assembly began. The Azimuth and Elevation frames of the rotator were created using Aluminum T-slots which were joined together using hidden connectors to form the external frames. Ball bearings and counter 3D printed parts were assembled and fixed to the frame. The worm gear was assembled using a M5-sized thread rod and its lock, which was then inserted into both the azimuth and elevation frames. Other components such as stepper motors, pulleys, timing belts and IR speed sensors were added to the rotor frames. The stepper motors were positioned and the elevation axis frame was mounted on top of the azimuth frame using hidden connectors and PVC pipes. The PVC pipes needed for this assembly was of 40mm diameter but even though PVC pipes of 40mm diameter were commercially available, they were slightly bigger than 40mm. So, the lathe machine was used to bring the diameter back to 40mm. For testing the rotator, a basic Arduino code was given and the rotator successfully ran according to the code given

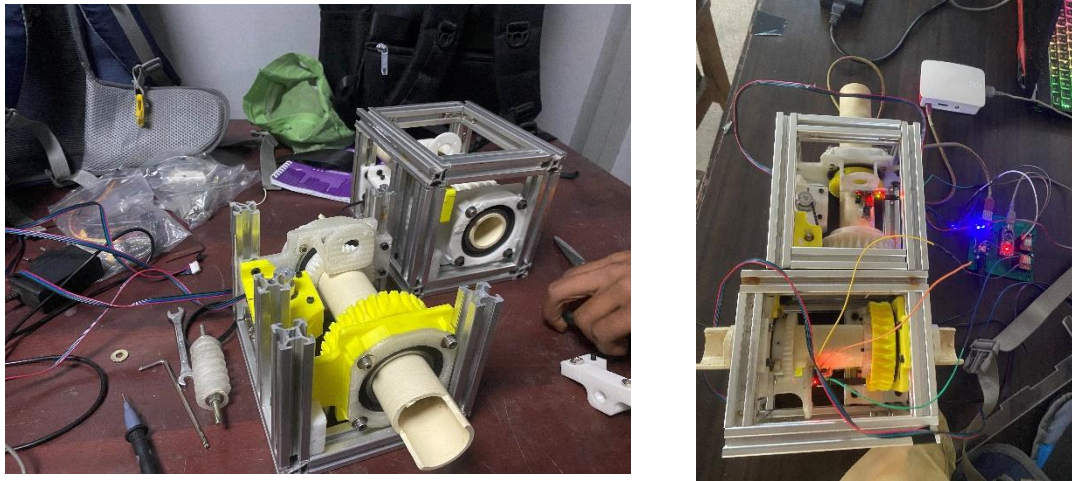


Figure 5.9: Assembly of rotator (left) and completed rotator (right)

### 5.3 Rotator Controller

The PCB for the rotator controller was provided by Orion Space. Each and every components need for the controller like Arduino Nano, A4988 drivers, Buck Converter, etc was assembled and soldered onto the PCB following the reference schematic design given below. A DC jack is used to give power to the board and a USB female connector is used to power the raspberry pi. The capacitors here used are of 25V 100 uf and resistors used are of 100k ohms. The polarity of the capacitors should be checked before soldering onto the board to prevent explosion of capacitors.

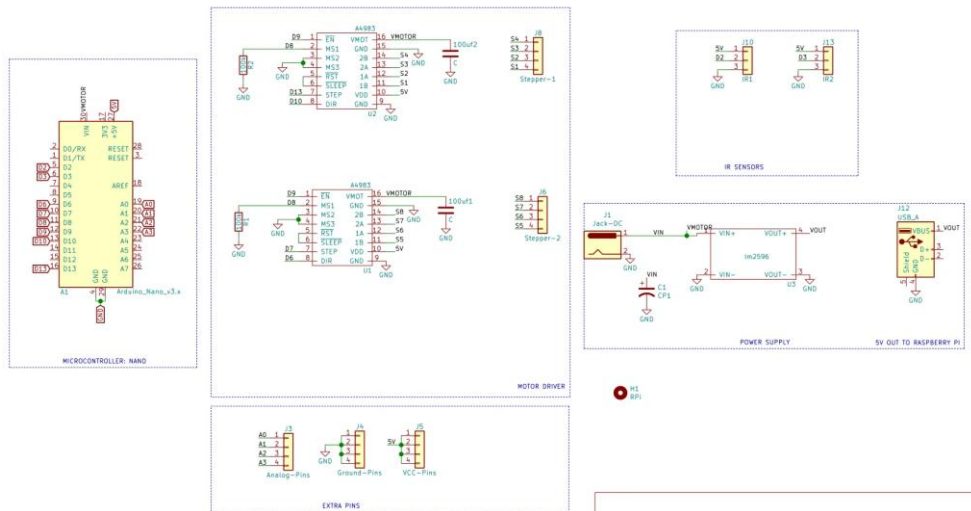


Figure 5.10: Schematic of the circuit board of rotator controller

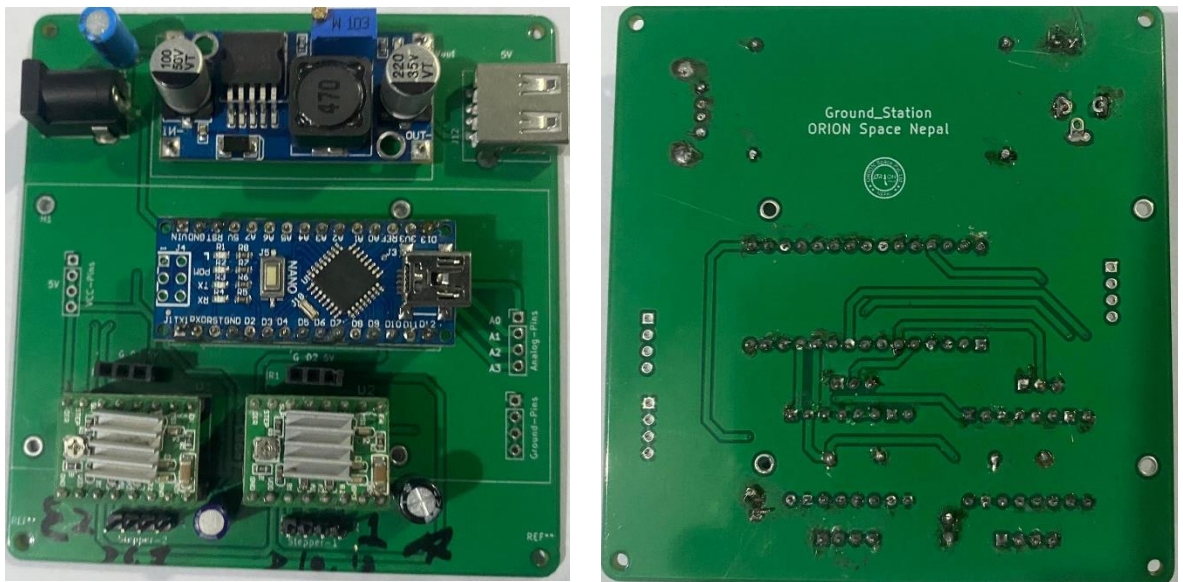


Figure 5.11: Front View Of Rotator Controller PCB (Left), Back View of Rotator Controller PCB (Right)

## 5.4 SatNOGS Client Setup

The client section consist of a Raspberry Pi with custom made OS by SatNOGS team. The raspberry pi was booted using the imager that was available on the SATNOGS website which had all the packages preinstalled in it. Then using an imager software, the imager was loaded to the raspberry pi via SD card and booted up. After configuring



basic settings at first initial boot-up, such as username and password, the pi was now ready to be used accordingly for the mission. The pi was made to connect to the WiFi and with the help of its IP address, it could be accessed remotely making it portable to places where there is a good internet connection.

An account on the SATNOGS network was created along with filling out specifications about the ground station hardware and other parameters such as the latitude, longitude, elevation and a station ID that is assigned to the ground station. Using this ID and an API key generated from the site, the basic configurations of SatNOGS Client was recorded using the following command:

```
$ sudo satnogs-setup
```

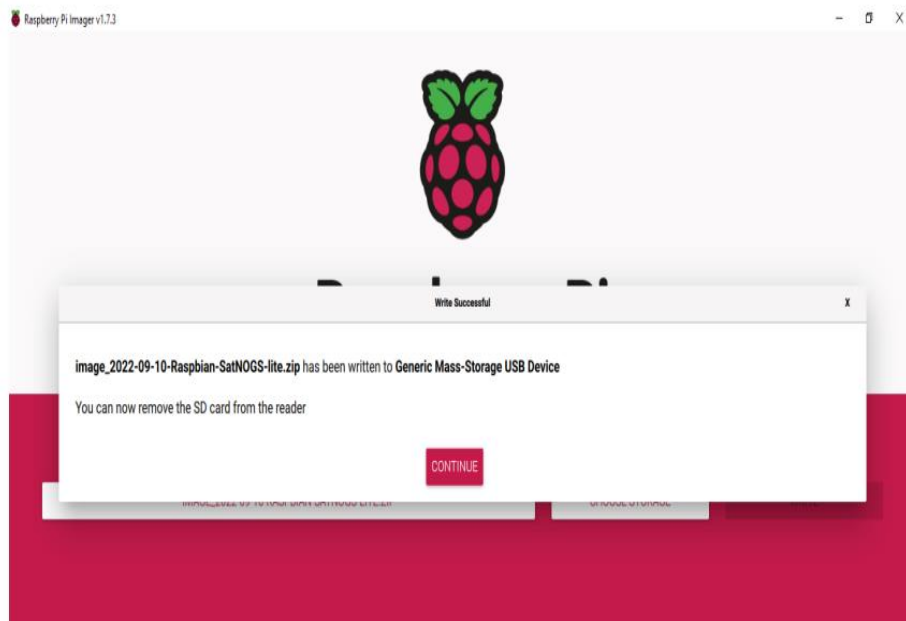


Figure 5.12: Installation of SatNOGS Imager on Raspberry Pi

```

|-----Basic configuration options-----|
x [UP], [DOWN] arrow keys to move      x
x [ENTER] to select                      x
x                                       k x
x x SATNOGS_API_TOKEN *Define API token [3fabd1ef114a8d 31aa6b] x x
x x SATNOGS_SOAPY_RX_DEVICE *Define Soapy RX device [driver=rtlsdr] x x
x x SATNOGS_ANTENNA *Define SatNOGS Radio Antenna [RX] x x
x x SATNOGS_RX_SAMP_RATE *Define RX sampling rate [2.048e6] x x
x x SATNOGS_STATION_ELEV *Define station elevation [1350] x x
x x SATNOGS_STATION_ID *Define station ID [3041] x x
x x SATNOGS_STATION_LAT *Define station latitude [27.682492] x x
x x SATNOGS_STATION_LON *Define station longitude [85.3207731] x x
x x x x x x
x m-----] x
|-----] x
x <select> < Back > x
|-----]

```

Figure 5.13: Basic Configuration for SatNOGS-Client

SATNOGS_API_TOKEN	3fabd1ef114a8dde161360a96c4ca5ed0a31aa6b
SATNOGS_SOAPY_RX_DEVICE	driver=rtlsdr
SATNOGS_ANTENNA	RX
SATNOGS_RX_SAMP_RATE	2.048e6
SATNOGS_STATION_ELEV	1350
SATNOGS_STATION_ID	3041
SATNOGS_STATION_LAT	27.682492
SATNOGS_STATION_LON	85.3207731

After entering all the basic parameters, the pi connected to the SATNOGS server and registered itself as a ground station in use to track the satellites.

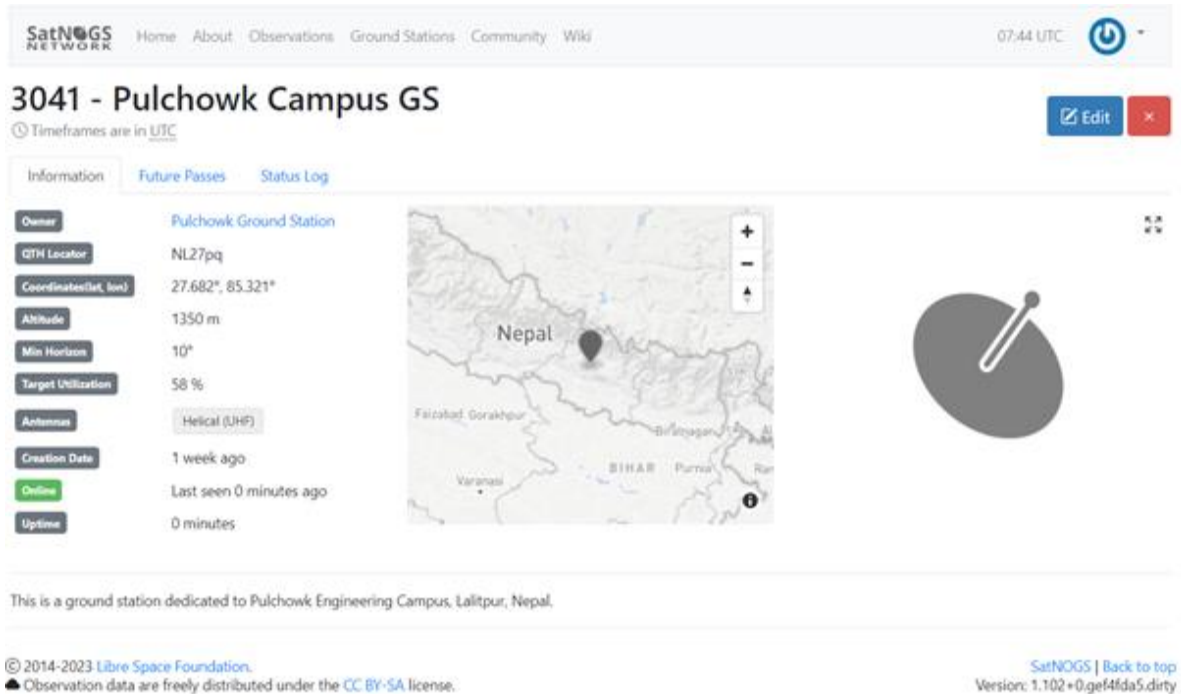


Figure 5.14: Ground Station as seen in SatNOGS Network

The following command was used for troubleshooting of the SatNOGS Client.

```
$ journalctl -f -u satnogs-client.service
```

After the SatNOGS client setup and SatNOGS network registry is completed, the satellites that are passing over the ground station with their signal strength could be seen while logging on the SatNOGS website. There the satellites can be selected to track and schedule for the ground station to connect to the satellite and collect the data from it. After the ground station has successfully established connection and collected the raw data from the satellite, it'll upload the data to the SatNOGS DB where the data is decoded as the raw data from the satellite is an encrypted data and every satellite has its own encryption key of which free satellites have their encryption key registered at SatNOGS. After the decoding is done, the results are processed and published on the SatNOGS network website, which anyone interested can view.

## 5.5 Assembly

After the completion and testing of the antenna and the rotator, the setup was taken to the terrace of the aerospace department. The view of the sky was clear with no tall buildings or trees blocking the pass of the satellite. The rotator was placed on a metal tripod stand and inserted the antenna into the rotator using a PVC pipe. The PCB and

other electronics were housed inside a box and attached to the stand. The helical antenna's center of mass shifted in a different direction because one side was heavy from the reflector, making it unable to rotate correctly. To keep the center of mass in the system's center, a counter-weight had to be hung in the opposite way.



Figure 5.15: Final View of the Ground Station

## 5.6 Observations from Ground Station

After setting up all the necessary parts and antennas, it was time to test the ground station to test the ground station to connect with some satellites. To do so, in the website of SATNOGS network, a satellite to track has to be scheduled and when the satellite passes over the ground station, the ground station will move accordingly to track and communicate with the satellite. During the tracking, the ground station got hold of two satellites responding to frequency of approximately 436.5 MHz.

### 5.6.1 Observation of CELESTA

CELESTA (ROBUSTA-1D) is a 1U CubeSat developed by the CSUM (platform) and CERN (payload). The mission of CELESTA is to test electronic components in space environment, the payload is a module including a radiation monitor “RadMon” and a SEL (Single Event Latch-up) experiment, both developed by CERN.

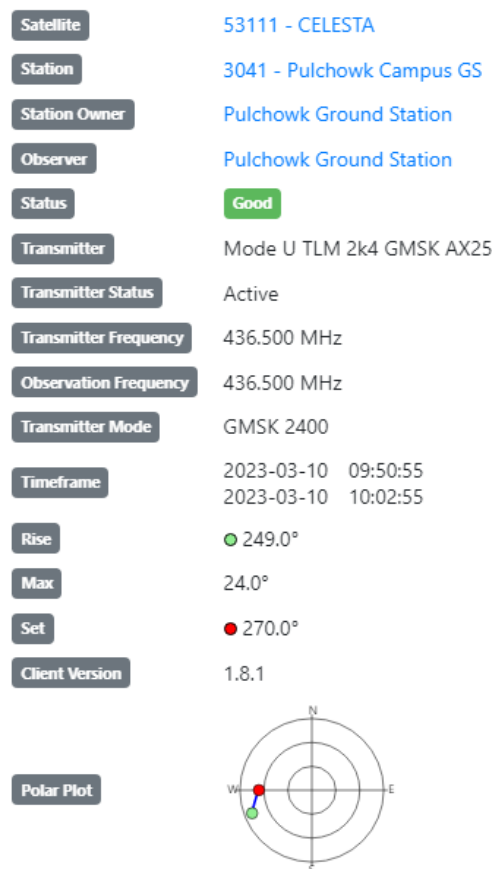


Figure 5.16: Observation summary of CELESTA satellite

## 5.6.2 Waterfall diagram of CELESTA

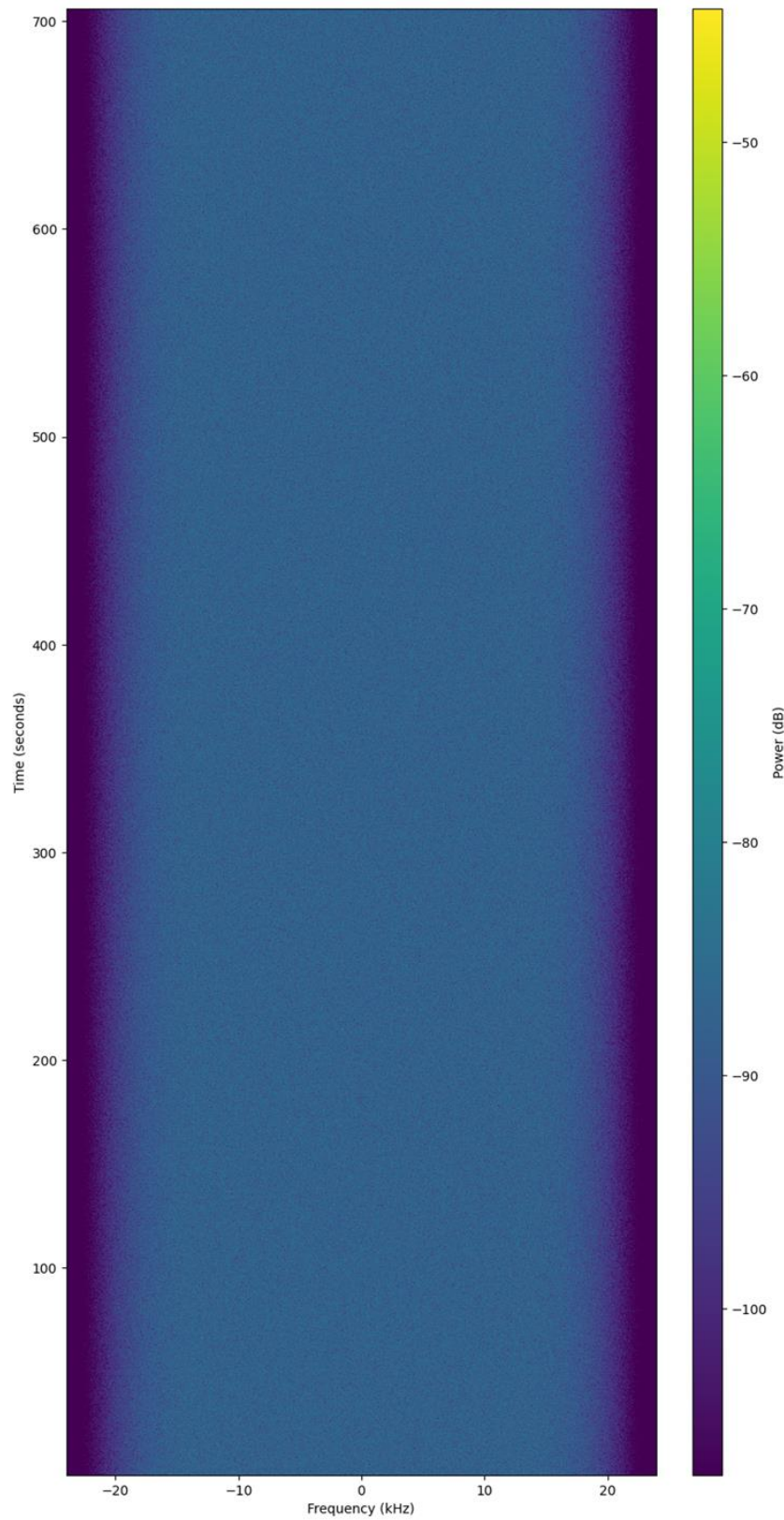


Figure 5.17: Waterfall diagram of CELESTA

The waterfall diagram is used to measure the good, medium or bad reception of the signal. In case of a bad reception of signal i.e. no signal is received, waterfall diagram is not present. If the signal is received but not properly, a black waterfall diagram is received. For good reception usually the packets of data received are visible in a waterfall diagram by seeing the form of bar-bursts. The tangent line in the waterfall diagram is the local noise and sometimes swinging lines are received which are the terrestrial noise. In the case of this waterfall diagram, neither any bar-bursts nor tangent lines were observed. Hence, it was concluded that the quality of the received signal was bad. Hence, later neither audio nor data was received from this observation.

### 5.6.3 Observation of the DUCHIFAT-3

Duchifat-3 is an Israeli student-built CubeSat that was developed by students at the Herzliya Science Center and the Sha'ar HaNegev High School, with support from the Israeli Space Agency and the Israel Ministry of Science and Technology. The satellite was launched on August 30, 2021, as part of a rideshare mission on a SpaceX Falcon 9 rocket. It is launched to demonstrate a new communication system that allows small satellites to communicate with each other without the need for a ground station.

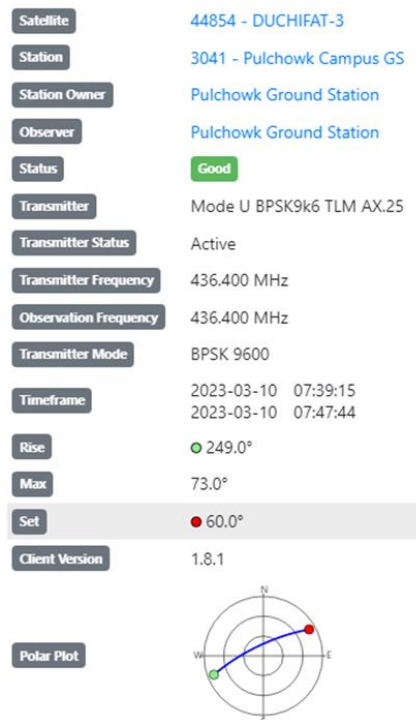


Figure 5.18: Observation summary of DUCHIFAT-3 satellite

### 5.6.4 Waterfall diagram of DUCHIFAT-3

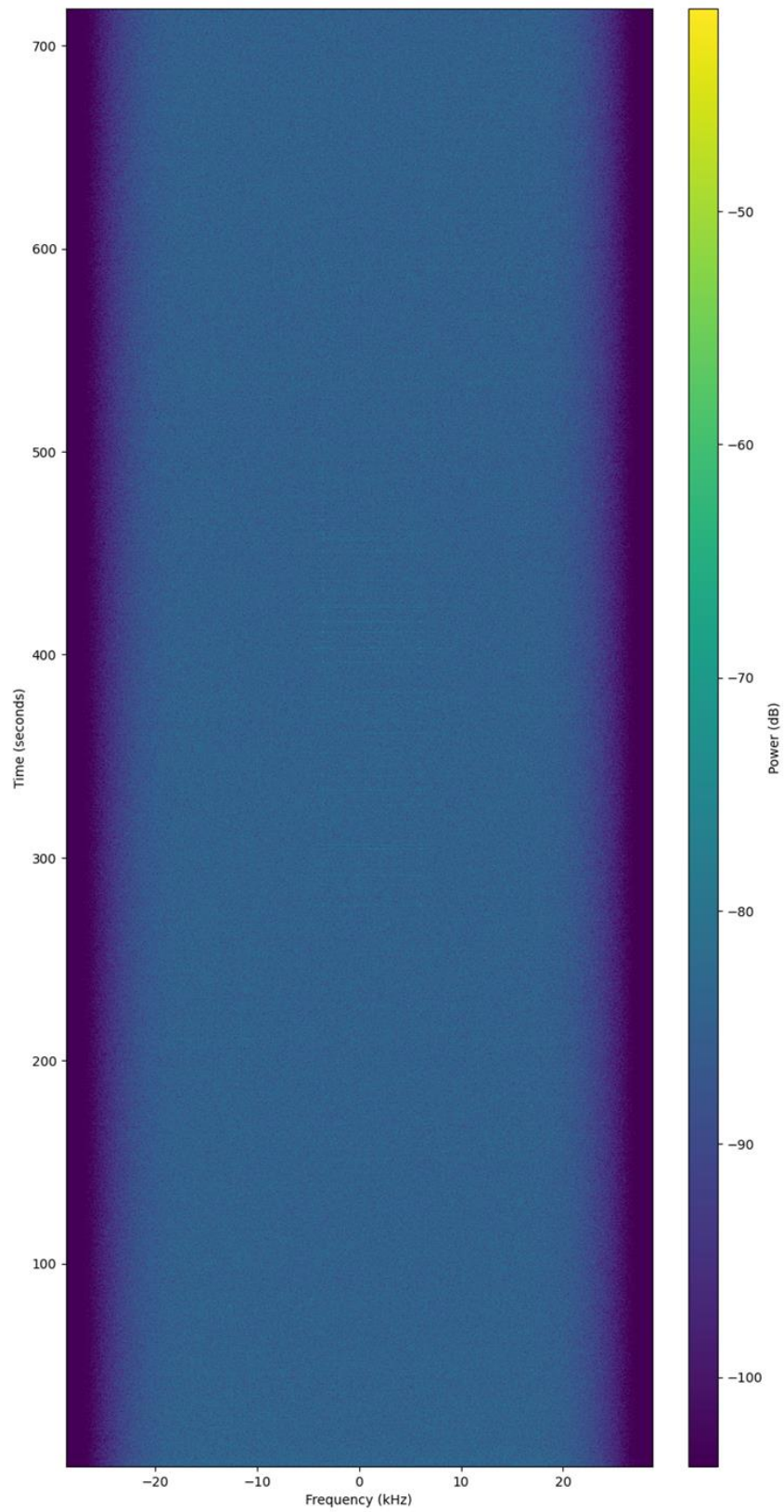


Figure 5.19: Waterfall diagram of DUCHIFAT-3



In the waterfall diagram of the DUCHIFAT-3 satellite, the satellite had communication with the ground station for a short time and it was successful in establishing connection with the satellite. Although faint burst of lines is seen in the middle part of waterfall diagram, no proper data was received on the ground station from the satellite.

The audio is also available at the SATNOGS network site as well as the waterfall diagrams. For any other satellites what are needed to be tracked and record the data, the passing over of the satellite over the ground station can be checked and scheduled for the observation of the desired satellite.

## **5.7 Limitations**

- i. The size and weight of the ground station and rotator is not considered.
- ii. It is only possible to detect LEO satellites.
- iii. Limited time for communicating with satellite..

## **5.8 Problems faced**

- i. Since the ground station incorporates mechanical as well as electronic components, problems occurring on either one of such part is limited by the knowledge of the operator on such parts.
- ii. The design and construction of a satellite tracking rotator and ground station gets constrained by the lack of parts, particularly specialized components or tiny variations in the dimensions of 3D printed parts.
- iii. The performance and dependability of the rotator and ground station gets affected by environmental conditions such as wind, rain and extremely high or low temperatures
- iv. Helical antenna compatibility was a crucial factor to take into account for both the rotator and the base station.

# **CHAPTER SIX: CONCLUSION AND FUTURE ENHANCEMENTS**

## **6.1 Conclusion**

In conclusion, designing and building a satellite tracking rotator and ground station is a challenging task that necessitates a lot of technical expertise and resources. The procedure requires taking into account a number of variables, including price, power needs, size and weight, environmental aspects, compatibility with helical antennas and parts availability. Despite these difficulties, a well-made ground station and tracking rotator for satellites can be a useful instrument for tracking and interacting with them. A correctly operating rotator and ground station can significantly improve the capacity to gather data and communicate with satellites, whether it is employed for amateur radio operations, as a component of a scientific investigation, or for commercial applications. A functioning and reliable satellite tracking rotator and ground station can be developed with careful planning, taking into account the constraints and issues encountered during the design and construction process and using high-quality components.

## **6.2 Future Enhancements**

A satellite tracking rotator and ground station's design and construction can be continuously upgraded to better suit users' needs. The list of prospective areas for future enhancement is as follows:

- i. The ground Station can be also be made as a transmitter station.
- ii. A proper external covering of rotator and antenna can be made.
- iii. Incorporating more robust materials instead of 3D printed parts and improved construction techniques can increase the durability and reliability of rotators and ground stations.
- iv. The demodulation of signal and data can be made faster.

## REFERENCES

- [1] M. Fischer and A. L. Scholtz, "Design of a Multi-mission Satellite Ground Station for Education and Research," 2010 Second International Conference on Advances in Satellite and Space Communications, 2010, doi: 10.1109/spacomm.2010.13.
- [2] Y. Du, "A Satellite Ground Station Control System," 2005.
- [3] T. ZHANG and Y. LI, "Design and Analysis of Micro-satellite Ground Station System," DEStech Transactions on Environment, Energy and Earth Science, no. icese, Apr. 2017, doi: 10.12783/dteees/icese2017/7894.
- [4] Tuli, Orr, Zee, T. Tuli, N. Orr and R. Zee, "Low Cost Ground Station Design for Nanosatellite Missions."
- [5] E. Ubom, "Ground Station Design for Satellite and Space Technology Development Alternative Energy View project Spectrum Management and harmonious coexistence View project," Feb. 2022.
- [6] T. Choi, T. Stevenson and E. G. Lightsey, "Reference Ground Station Design for University Satellite Missions with Varying Communication Requirements," 55th AIAA Aerospace Sciences Meeting, Jan. 2017, doi: 10.2514/6.2017-1334.
- [7] V. Dascal, P. Dolea, O. Cristea and T. Palade, "Low-Cost SDR-Based Ground Receiving Station for LEO Satellite Operations," Oct. 2013.
- [8] G. Woźniak and M. Stolarski, "Low cost amateur rotators for student's satellites and high altitude balloon application," SPIE Proceedings, Nov. 2012, doi: 10.1117/12.2000189.
- [9] S. Rama, H. Anggara, A. Wijanto, B. Prasetyo and Syihabuddin, "Automated Ground Station with Customized Rotator for Antenna Pointing using Compass Sensor," Nov. 2014.

- [10] N. Hongyim and S. Mitatha, "Building Automatic Antenna Tracking System for Low Earth Orbit(LEO) Satellite Communications," 2015.
- [11] Rogério Atem De Carvalho, J. Estela, M. Langer and J. Wiley, *Nanosatellites : space and ground technologies, operations and economics*. Hoboken ; Chichester: John Wiley & Sons Ltd, 2020.
- [12] A. Done, A.-M. Căilean, C.-E. Leşanu, M. Dimian and A. Graur, "Design and implementation of a satellite communication ground station." [13]A. de, J. Estela and M. Langer, *Nanosatellites*. John Wiley & Sons, 2020.
- [13] Ø. Karlsen, "Ground station considerations for the AMOS satellite programme," 2017.
- [14] S. S. Pattnaik and S. K. Panda, "Helical Antennas for Satellite Communication: A Review," in *International Journal of Engineering Research & Technology*, vol. 4, no. 7, Jul. 2015.
- [15] N. Gupta, R. Garg and I. Bahl, "Microstrip Antennas," in *Microstrip Antenna Design Handbook*, Artech House, 2001,
- [16] J. T. Bernhard, "Helical Antennas for Satellite Communication," in *Antenna Theory and Design*, John Wiley & Sons, 2017,
- [17] O. Ceylan et al., "Small Satellites Rock A Software-Defined Radio Modem and Ground Station Design for Cube Satellite Communication," *IEEE Microwave Magazine*, vol. 17, no. 3, pp. 26–33, Mar. 2016, doi: 10.1109/mmm.2015.2505700.
- [18] J. W. Cutler and A. Fox, "A Framework For Robust and Flexible Ground Station Networks," *Journal of Aerospace Computing, Information and Communication*, vol. 3, no. 3, pp. 73–92, Mar. 2006, doi: 10.2514/1.15464.
- [19] R. Leila, B. Nouredine and A. Hafid Belbachir, "Software and hardware implements for tracking low earth orbit (LEO) satellites Une nouvelle méthode de cartographie de la région d’Oran à l’aide de la télédétection multispectrale.

## APPENDICES

### APPENDIX A: BUDGET ANALYSIS

S.No	Name of Components and Items	Quantity	Price (NRs.)
1.	Raspberry Pi 3 Model B	1	9500
2.	Arduino Nano	1	850
3.	Battery	1	5500
4.	Stepper Motor 24V	2	3100
5.	DRV8825	2	700
6.	RTL-SDR	1	5000
7.	Aluminum L-channel	1	2000
8.	PVC pipe 40 mm	6m	1200
9	N type Connector ,Adaptor & 50ohms co -axial Cable	1-1-1piece	4650
10	6008 Zz 625 Zz Bearing	2-2 piece	2000
11	(M3,M4 ,M5,M6 ,T5) Nuts ,Bolts & Washer	250	4000
12	PVC pipe 20 mm	6m	1000
13.	Miscellaneous		500
		Total	40,000

## APPENDIX B: CONSTRUCTION OF ROTATOR

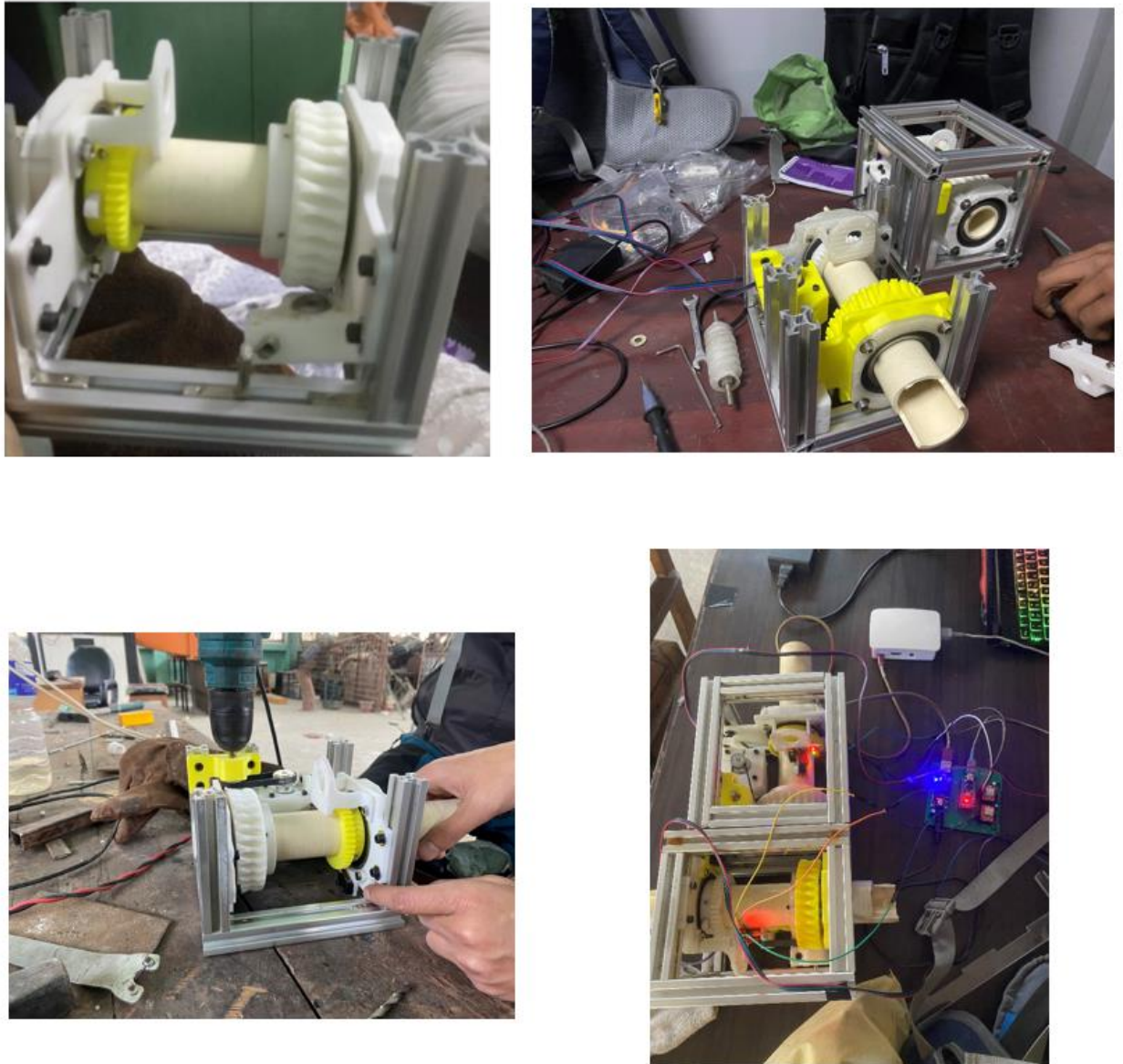


Figure B.1: Construction of Rotator

## APPENDIX C: CONSTRUCTION OF ANTENNA



Figure C.1: Construction of Antenna



## APPENDIX D: ASSEMBLY OF ROTATOR AND ANTENNA



Figure D.1: Final Setup of Ground Station on top of Aerospace Department

## APPENDIX E: ARDUINO CODE

```
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include <AccelStepper.h>

#include <SoftwareSerial.h>
SoftwareSerial bluetooth(5, 4); // RX, TX

#define DIR_AZ 10 //PIN for Azimuth Direction
#define STEP_AZ 13 //PIN for Azimuth Steps
#define DIR_EL 6 //PIN for Elevation Direction
#define STEP_EL 7 //PIN for Elevation Steps

#define MS1 8 //PIN for step size
#define EN 9 //PIN for Enable or Disable Stepper Motors

#define SPR 200 //Step Per Revolution
#define RATIO 65 //Gear ratio
#define T_DELAY 60000 //Time to disable the motors in
millisecond

#define HOME_AZ 2 //Homing switch for Azimuth
#define HOME_EL 3 //Homing switch for Elevation
/*The MAX_ANGLE depends of ANGLE_SCANNING_MULT and maybe
misbehave for large values*/
#define ANGLE_SCANNING_MULT 180 //Angle scanning
multiplier
#define MAX_AZ_ANGLE 360 //Maximum Angle of Azimuth for
homing scanning
#define MAX_EL_ANGLE 360 //Maximum Angle of Elevation for
homing scanning
```

```

#define HOME_DELAY 6000 //Time for homing Deceleration
in millisecond

/*Global Variables*/
unsigned long t_DIS = 0; //time to disable the Motors
/*Define a stepper and the pins it will use*/
AccelStepper AZstepper(1, STEP_AZ, DIR_AZ);
AccelStepper ELstepper(1, STEP_EL, DIR_EL);

void setup()
{
  /*Change these to suit your stepper if you want*/
  AZstepper.setMaxSpeed(150);
  AZstepper.setAcceleration(50);

  /*Change these to suit your stepper if you want*/
  ELstepper.setMaxSpeed(150);
  ELstepper.setAcceleration(50);

  /*Enable Motors*/
  pinMode(EN, OUTPUT);
  digitalWrite(EN, LOW);
  /*Step size*/
  pinMode(MS1, OUTPUT);
  digitalWrite(MS1, LOW); //Full step
  /*Homing switch*/
  pinMode(HOME_AZ, INPUT);
  pinMode(HOME_EL, INPUT);
  /*Serial Communication*/
  Serial.begin(19200);
  bluetooth.begin(115200);
  bluetooth.println("start");

  /*Initial Homing*/

```

```

    //bluetooth.println("Initial Homing");
    Homing(deg2step(-ANGLE_SCANNING_MULT), deg2step(-
ANGLE_SCANNING_MULT));
    bluetooth.println("Reached Home Position");

}

void loop()
{
    /*Define the steps*/
    static int AZstep = 0;
    static int ELstep = 0;
    /*Time Check*/
    if (t_DIS == 0)
        t_DIS = millis();

    /*Disable Motors*/
    if (AZstep == AZstepper.currentPosition() && ELstep ==
ELstepper.currentPosition() && millis() - t_DIS >
T_DELAY)
    {
        bluetooth.println("disable motors");
        digitalWrite(EN, HIGH);
    }
    /*Enable Motors*/
    else {
        bluetooth.println("enable motors");

        digitalWrite(EN, LOW);
    }

    /*Read the steps from serial*/
    bluetooth.println("I am above read");
    cmd_proc(AZstep, ELstep);
}

```

```

    /*Move the Azimuth & Elevation Motor*/
    stepper_move(AZstep, ELstep);
}

/*Homing Function*/
void Homing(int AZsteps, int ELsteps)
{
    /**/
    bluetooth.println("Move the steppers to initial
position/ Homing");
    int value_Home_AZ = HIGH;
    int value_Home_EL = HIGH;
    int n_AZ = 1; //Times that AZ angle has changed
    int n_EL = 1; //Times that EL angle has changed
    boolean isHome_AZ = false;
    boolean isHome_EL = false;

    AZstepper.moveTo(AZsteps);
    ELstepper.moveTo(ELsteps);

    while (isHome_AZ == false || isHome_EL == false)
    {
        value_Home_AZ = digitalRead(HOME_AZ);
        bluetooth.println(value_Home_AZ);
        value_Home_EL = digitalRead(HOME_EL);
        /* Change to LOW according to Home sensor */
        if (value_Home_AZ == HIGH)
        {
            AZstepper.moveTo(AZstepper.currentPosition());
            isHome_AZ = true;
            bluetooth.println("AZ is high " );
        }
        /* Change to LOW according to Home sensor */
        if (value_Home_EL == HIGH)

```

```

{
    ELstepper.moveTo(ELstepper.currentPosition());
    isHome_EL = true;
    bluetooth.println("EL is high " );
}
if (AZstepper.distanceToGo() == 0 && !isHome_AZ)
{
    n_AZ++;
    AZsteps = deg2step(pow(-1, n_AZ) * n_AZ *
ANGLE_SCANNING_MULT);
    if (abs(n_AZ * ANGLE_SCANNING_MULT) > MAX_AZ_ANGLE)
    {
        error(0);
        break;
    }
    AZstepper.moveTo(AZsteps);
}
if (ELstepper.distanceToGo() == 0 && !isHome_EL)
{
    n_EL++;
    ELsteps = deg2step(pow(-1, n_EL) * n_EL *
ANGLE_SCANNING_MULT);
    if (abs(n_EL * ANGLE_SCANNING_MULT) > MAX_EL_ANGLE)
    {
        error(1);
        break;
    }
    ELstepper.moveTo(ELsteps);
}

AZstepper.run();
ELstepper.run();
}
/*Delay to Decelerate*/

```

```

long time = millis();
while (millis() - time < HOME_DELAY)
{
    AZstepper.run();
    ELstepper.run();
}
/*Reset the steps*/
AZstepper.setCurrentPosition(0);
ELstepper.setCurrentPosition(0);
}

/*EasyComm 2 Protocol & Calculate the steps*/
void cmd_proc(int &stepAz, int &stepEl)
{
    /*Serial*/
    bluetooth.println("inside cmd_proc");

    char buffer[256];
    char incomingByte;
    char *p = buffer;
    char *str;
    static int counter = 0;
    char data[100];

    double angleAz, angleEl;

    /*Read from serial*/
    while (Serial.available() > 0)
    {
        bluetooth.println("data available");
        incomingByte = Serial.read();
        /* XXX: Get position using custom and test code */
        if (incomingByte == '!')
        {

```

```

        /* Get position */
        Serial.print("TM");
        Serial.print(1);
        Serial.print(" ");
        Serial.print("AZ");
        Serial.print(10 *
step2deg(AZstepper.currentPosition()), 1);
        Serial.print(" ");
        Serial.print("EL");
        Serial.println(10 *
step2deg(ELstepper.currentPosition()), 1);
    }
    /*new data*/
    else if (incomingByte == '\n')
    {
        p = buffer;
        buffer[counter] = 0;
        if (buffer[0] == 'A' && buffer[1] == 'Z')
        {
            if (buffer[2] == ' ' && buffer[3] == 'E' &&
buffer[4] == 'L')
            {
                /* Get position */
                bluetooth.println("getting data");
                Serial.print("AZ");

Serial.print(step2deg(AZstepper.currentPosition()), 1);
                Serial.print(" ");
                Serial.print("EL");

Serial.print(step2deg(ELstepper.currentPosition()), 1);
                Serial.println(" ");
            }
        }
    }
    else

```



```

{
  /*Get the absolute value of angle*/
  str = strtok_r(p, " " , &p);
  strncpy(data, str + 2, 10);
  angleAz = atof(data);
  /*Calculate the steps*/
  stepAz = deg2step(angleAz);

  /*Get the absolute value of angle*/
  str = strtok_r(p, " " , &p);
  if (str[0] == 'E' && str[1] == 'L')
  {
    strncpy(data, str + 2, 10);
    angleEl = atof(data);
    /*Calculate the steps*/
    stepEl = deg2step(angleEl);
  }
}
}
/* Stop Moving */
else if (buffer[0] == 'S' && buffer[1] == 'A' &&
buffer[2] == ' ' && buffer[3] == 'S' && buffer[4] == 'E')
{
  /* Get position */
  bluetooth.println("stop moving");
  Serial.print("AZ");

Serial.print(step2deg(AZstepper.currentPosition()), 1);
  Serial.print(" ");
  Serial.print("EL");

Serial.println(step2deg(ELstepper.currentPosition()), 1);
  stepAz = AZstepper.currentPosition();
  stepEl = ELstepper.currentPosition();
}
}
}

```

```

    }
    /* Reset the rotator */
    else if (buffer[0] == 'R' && buffer[1] == 'E' &&
buffer[2] == 'S' && buffer[3] == 'E' && buffer[4] == 'T')
    {
        /* Get position */
        Serial.print("AZ");

Serial.print(step2deg(AZstepper.currentPosition()), 1);
        Serial.print(" ");
        Serial.print("EL");

Serial.println(step2deg(ELstepper.currentPosition()), 1);
        /*Move the steppers to initial position*/
        Homing(0, 0);
        /*Zero the steps*/
        stepAz = 0;
        stepEl = 0;
    }
    counter = 0;
    /*Reset the disable motor time*/
    t_DIS = 0;
}
/*Fill the buffer with incoming data*/
else {
    buffer[counter] = incomingByte;
    counter++;
}
}

/*Error Handling*/
void error(int num_error)
{

```

```

switch (num_error)
{
  /*Azimuth error*/
  case (0):
    while (1)
    {
      Serial.println("AL001");
      delay(100);
    }
  /*Elevation error*/
  case (1):
    while (1)
    {
      Serial.println("AL002");
      delay(100);
    }
  default:
    while (1)
    {
      Serial.println("AL000");
      delay(100);
    }
}
}

/*Send pulses to stepper motor drivers*/
void stepper_move(int stepAz, int stepEl)
{
  AZstepper.moveTo(stepAz);
  ELstepper.moveTo(stepEl);

  AZstepper.run();
  ELstepper.run();
}

```

```
/*Convert degrees to steps*/
int deg2step(double deg)
{
    return (RATIO * SPR * deg / 360);
}

/*Convert steps to degrees*/
double step2deg(int Step)
{
    return (360.00 * Step / (SPR * RATIO));
}
```