

Public Outreach Radio Telescope (PORT)

Graphical User Interface User Manual

Document Number: 3000

Revision: Release-1.0

Team 21039

Spring 2021

Contents

1	Revision History	2
2	Graphical User Interface (GUI) Overview	2
3	Dependency	2
4	GUI Layout	3
5	Main Control Function	3
5.1	Menu Bar	3
5.2	Three Main Buttons	3
5.2.1	Start Calibration	3
5.2.2	Telescope Stop	3
5.2.3	Stow Telescope	4
5.3	Coordinate Entry	4
5.4	System Status	4
5.5	Jogging	4
5.6	System Settings	5
5.6.1	Enable Telescope	5
5.6.2	Enable Servomotors	5
5.6.3	Enable AnalogFrontEnd	5
5.6.4	Advance Features	5
5.7	Data Output	6
6	Operating Procedure	6

1 Revision History

- 2021-04-28: Draft created
- 2021-05-09: Draft edited
- 2021-05-09: v1.0 released

2 Graphical User Interface (GUI) Overview

The Public Outreach Radio Telescope (PORT) Graphical User Interface (GUI) is the front-end for interfacing and controlling the PORT.

As of writing this manual, the GUI only has capability to perform basic telescope movement. Tracking and other improvement are future goals and were outside the requirement/scope of the Team 21039.

As of most other Capstone projects, this GUI is written in **Python** for simplicity and fast deployment. The GUI toolkit is **PySimpleGUI**, probably the easiest one to get started comparing to others.

Python3.7 was used during the development, newer version might be used but they are not tested and verified.

The GUI was developed under a Debian 10 GNU/Linux environment, so the rest of this manual assumes such. Since Python and the GUI toolkit are cross-platform, there should be no problem to fork the software to other platforms; only slight modification are needed such as the name of the USB-Serial port.

3 Dependency

The GUI is mainly written in Python, and Python3 and newer is required. To install python3 on Debian 10:

```
sudo apt install python3.7
```

To install other dependencies, Python's pip is recommended. To install pip on Debian 10:

```
sudo apt install python3-pip
```

With pip installed, check if package **PySimpleGUI** and **pyserial** exists using:

```
pip3 list
```

If not, install them using:

```
pip3 install PySimpleGUI
```

```
pip3 install pyserial
```

As of writing this manual, PySimpleGUI is on 4.30.0, and pyserial sits on 3.5

In an GNU/Linux environment, and after cloning the repo, the GUI can be opened via command line: `python3.7 main.py`

4 GUI Layout

If the GUI is opened correctly, there should be four windows, the main control, two position dials, and an output window.

The main control is where essential operations take place; position dials simply keep track of the position of the telescope in a visual way. The output window shows (error) messages generated by the software for debugging.

Despite the output window mostly providing informational messages, it does output error and debug messages. Keep in mind that currently the outputs will not be saved upon exiting the GUI.

5 Main Control Function

This section documents the various functions of the main control GUI window. This window has the title: **Public Outreach Radio Telescope Control**.

5.1 Menu Bar

The menu bar contains three buttons: File, Edit, and Help.

From the file button you can exit the GUI.

The Edit button has the subbuttons of Edit, Copy, and Paste, which you can use when entering in coordinates.

The Help button will show a link to this instruction manual.

5.2 Three Main Buttons

The three main buttons at the top of the GUI are **Start Calibration**, **Telescope Stop**, and **Stow Telescope**.

5.2.1 Start Calibration

The **Start Calibration** button should be used after **Enable Telescope** and **Enable Servomotors** are selected. This button runs a function which moves the telescope from its stow position to zenith.

The idea is that, since the dish is off-axis, it's easy to find a spot on the counter-weight arm or the ALT arm and calibrate it to zenith using a level or angle meter.

After the ALT arm is calibrated to zenith, the software can simply move it 30° over and the telescope dish will point straight up to zenith, and therefore the ALT servomotor can be zeroed and use that position as its absolute zero.

5.2.2 Telescope Stop

The Telescope Stop button is available for users if wrong coordinates are entered, or if there is a need for the telescope to be stopped immediately. This does not take the place of the emergency

stop button. If a hard stop is needed, the physical E-STOP should be used to disable the servomotor drive and interrupt the motion.

In the case of an incorrect coord. entry, or any other reason, one can use the Telescope Stop button to stop the telescope immediately without interrupting the power or closing the GUI.

5.2.3 Stow Telescope

Stow Telescope brings the telescope back to its stow position. It only works if **Enable Telescope** and **Enable Servomotors** are both selected.

The telescope stowing position is 30° pass zenith (relative to the ALT or counter-weight arm).

Once the user has finished using the PORT, they **MUST** stow the telescope by clicking this button before shutting down the GUI. This is to minimize the cross section of the dish to help reducing the wind load when stowed.

5.3 Coordinate Entry

The coordinate entry inputs are in terms of ALT/AZ. Therefore, the user must input only azimuthal and altitudinal values. The user simply has to enter a value between 15-120 degrees for the altitude, and a value between 0-360 for the azimuth axis. Then, you click the **Read** button. The PORT then moves to the inputted coordinates. If the user enters a value that exceeds the mechanical limit, there is a software limit which will not allow the telescope to move past either 120 degrees altitudinally (stow position), or past horizon.

5.4 System Status

The system operator shall be able to know the current status of the telescope in order to perform the current operation. The System Status section fulfills such requirements.

The Current/Target Position indicates telescope position in Az-Alt coordinates.

Wind Speed is an important parameter, because the telescope needs to be stowed when the operating wind speed limit is exceeded. If the wind speed is above 5 m/s, a warning is shown. If the wind speed exceeds 25 m/s, a text is shown to stow the telescope immediately.

The voltage display of both servo motors gives the operator insight of the power supply. The motion accuracy of servo motors deteriorates as the operating voltage drops, which is why when voltage starts to drop, troubleshooting is advised.

5.5 Jogging

Jogging is a convenient way to position the telescope for small movements, as it is often required when performing maintenance/troubleshooting.

The operator must select **Enable Jogging** in System Settings section before the user can jog the telescope. The operator can choose one of the four available jogging steps for each axis, ranging from a 0.5 to 10° per click. This is useful if the telescope position is needed to be adjusted in small increments, especially when stowing the telescope. Simply click which axis you wish to move, and then click the increment (in degrees) that you want to move.

It is advised to deselect the **Enable Jogging** when not in use to avoid accidental movement of the PORT.

5.6 System Settings

It was decided to include the System Settings feature to prevent unintentional movement or use of the PORT. To be able to move the telescope in the first place, the user must enable that specific feature on the GUI, such as jogging. This design was included because this telescope will be used by the public, so any additional layer of protection is seen as beneficial.

5.6.1 Enable Telescope

The **Enable Telescope** and **Enable Servomotors** options must be selected to operate the PORT.

The **Enable Telescope** only effects the ability to select other options and doesn't effect the status of the radio telescope when it is toggled on or off. This can be deselected after selecting the **Enable Servomotors**, so that servomotors cannot be disabled accidentally.

5.6.2 Enable Servomotors

Selecting **Enable Servomotors** initiates the RS-232 serial connections to both ALT/AZ motors. According to servomotor's datasheet, the serial connection has parameters of 8N2, baudrate of 115200k.

This option needs to stay selected if telescope motions are to be performed. When the serial connection to any of the servomotors is lost, there may be error message show up on the **Output** window, and there is a slight chance that the GUI may crash.

5.6.3 Enable AnalogFrontEnd

Selecting **Enable AnalogFrontEnd** causes the GUI polls AnalogFrontEnd (AFE) data. At the of writing, only the wind speed reading is polled, to update the **Wind Speed** in **System Status**.

The physical AFE is an Arduino Nano, and is connected to the Raspberry Pi via USB. They communicate using RS-232 with 8N2, baudrate 57,600. Arduino pin A0 is used as ADC input for anemometer analog output.

It is planned that in the future AFE will be used to control the brake system, and sense the E-STOP that will be mounted on-site with the telescope.

5.6.4 Advance Features

Selecting this will enable 3 buttons, **Home AZ**, **Home ALT**, and **Restart**.

Home AZ and **Home ALT** set the current position as the home/zero absolute position for that servomotor. This could be useful to recover the telescope after an unexpected power interruption and the system fails stow.

Restart sends signal to both servo motors and they restart. Sometimes this is needed after an unexpected power interruption, where servomotors can throw *Hold Error* or something like that. This can also be used if **Jogging** is not working.

5.7 Data Output

The Data Output feature is a placeholder for future teams so that they can save data and create graphs for future information received from the PORT.

6 Operating Procedure

For the GUI to run correctly, the user must turn on the telescope in a specific order. The following outlines the steps needed to turn on the telescope correctly.

Note: This procedure will need to be updated as soon as the remote E-STOP system is deployed in the future.

1. Connect the power cord plug (NEMA 5-15) to wall/power outlet.
2. Check both USB-Serial cables are connected to servomotor DB15 cables. They are labelled Servo-AZ and Servo-ALT.
3. Check the USB end of the USB-Serial are connected to the Raspberry Pi. If not, connect the AZ first, then ALT.
4. Check the Arduino Nano is connected to the Raspberry Pi.
5. Clear the telescope surrounding.
6. Remove the lock-out hasp from the disconnect switch, and connect the disconnect switch. If done correctly, Power Indicator should light up. Ensuring you are following the LOTO procedure.
7. Reset the E-Stop.
8. Use RealVNC to remote control the remote Raspberry Pi.
9. Open the Terminal and type the following:
`cd SRT-GUI` (*change directory to the one has the GUI*)
`python3.7 main.py` (*use python3.7 to open the GUI*)
10. GUI should open. If not, go back over the steps and make sure you did not miss any connections.
11. After opening the GUI, the first action is to select **Enable Telescope**, **Enable Servomotors**, and **Enable AFE** in **System Settings**.
12. Click the **Start Calibration**.
13. Operate the PORT as you wish.
14. Once you are done using the PORT, you must stow the telescope. Do this by clicking **Stow Telescope**.
15. After stowing the PORT, you can now exit the GUI using **Exit**.
16. GUI log and last servomotor positions can be found within files `gui.log` and `absPos.dat`.
17. Turn the disconnect switch to **OFF** position and follow LOTO procedure, Power Indicator should go away in few seconds.