

# STARTUP TRACKING SYSTEM

© EA3HMJ

Translated by CT1BYM

# INDICE

- STARTUP TRACKING SYSTEM .....2
- Encoders .....4
  - Elevation SOLAR-360 MB .....4
    - Verification .....6
  - Azimuth SE58Sx .....7
    - Verification .....7
- DriverDish .....8
- JPLastroserver .....12
- Verification of communications.....13
- SNserver.....15

# STARTUP TRACKING SYSTEM

This is an antenna tracking system that integrates dedicated hardware and software designed for amateur use in Earth-Moon-Earth (EME) communications, radio astronomy, amateur Deep Sky Network (DSN) and other Space Communications applications where an accurate and high precision tracking is needed.

Proper use of the system requires knowledge of electronics and some software skills. Therefore, it is recommended for users with previous experience in antenna tracking systems only.

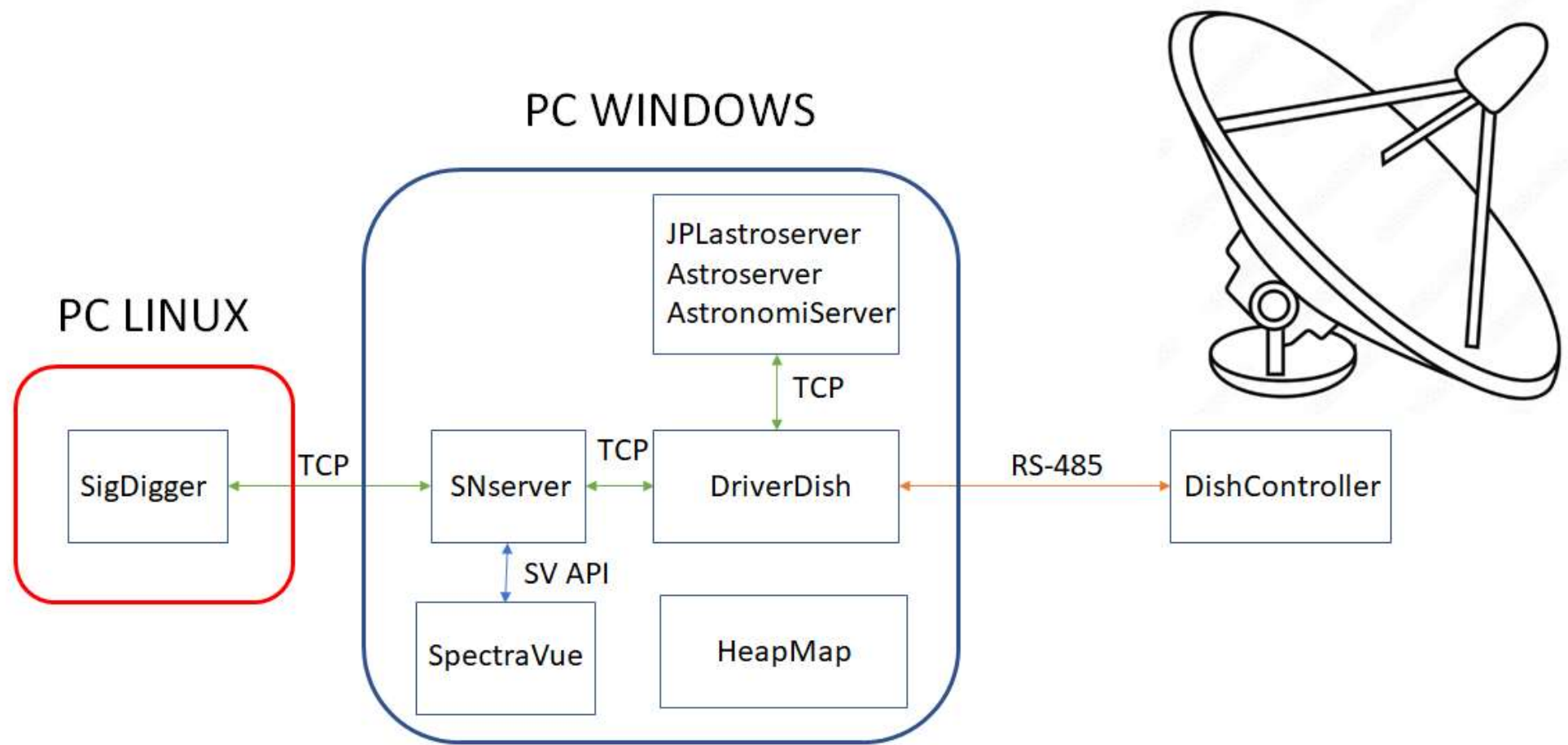
The hardware components of this system have been specifically selected to support the software applications that run on it. These components include a DishController unit that uses an ESP32 microcontroller (MCU), a DC motor driver board with high-performance, high-current control devices, RS485 input/output devices, and a DC step-up converter (versions <1.5). Other peripherals can be connected, such as a variety of digital absolute encoders and inclinometers to sense the motion and position of the motorized turning system, and a remote control unit.

The application software includes the DriverDish program, responsible for requesting data from the ephemeris server and for communicating with the DishController unit, and a suite of server programs (JPLastroserver, Astroserver, or Astronomyserver) that obtain the most accurate positioning data possible, are available from reliable sources.

The integration of dedicated hardware and software in such a system offers several advantages. First, it improves system performance and reliability by allowing the DishController unit to directly access and control specialized hardware components such as motor drivers and position sensors. This reduces overhead and improves response times. Second, it simplifies system design and development by providing a standardized set of hardware and software components optimized for narrow-beam antenna tracking. Third, unlike other antenna tracking system designs, motion and position sensors are located at the last link in the motion chain to actively monitor the aim of the antenna. In this way, the system is able to minimize the impact of mechanical tolerances and gear backlash.

The system is made up of three blocks that work together:

- Ephemeris servers (Astroserver, JPLastroserver or Astronomyserver)
- Control software, SO Windows (DriverDish.App)
- Antenna Controller (Controller Dish)



# EA3HMJ Antenna Tracking System

## Encoders

The first thing we must do is configure the encoders, both SOLAR-360 elevation and SE58xx azimuth.

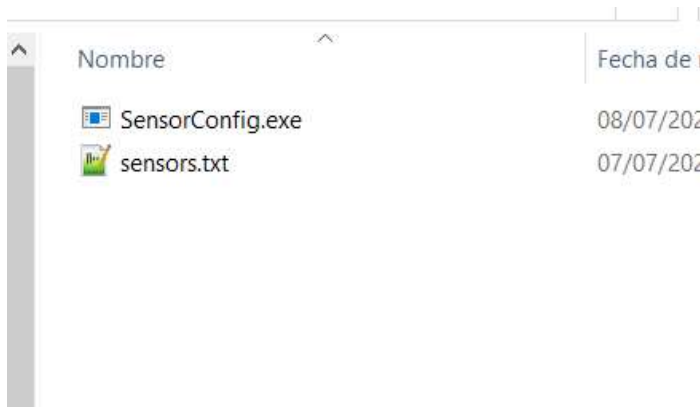
To simplify the process I have created a utility that does the job automatically, we just have to connect the sensor to the computer through the USB-RS485 adapter and power it.

We will use two programs, one to configure and the other to verify.

To configure: <https://ea3hmj.net/download/SensorConfig.rar>

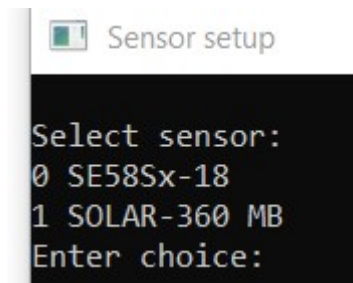
To verify: <https://ea3hmj.net/download/ModRTUClient.rar>

Unzip the file to a directory and run SensorConfig.exe

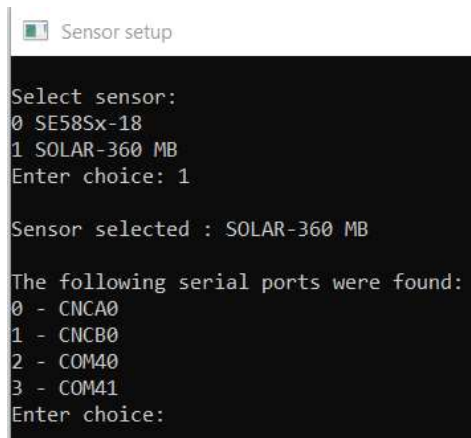


## Elevation SOLAR-360 MB

We connect the encoder and feed it.



Option 1 ask wich port it is connected



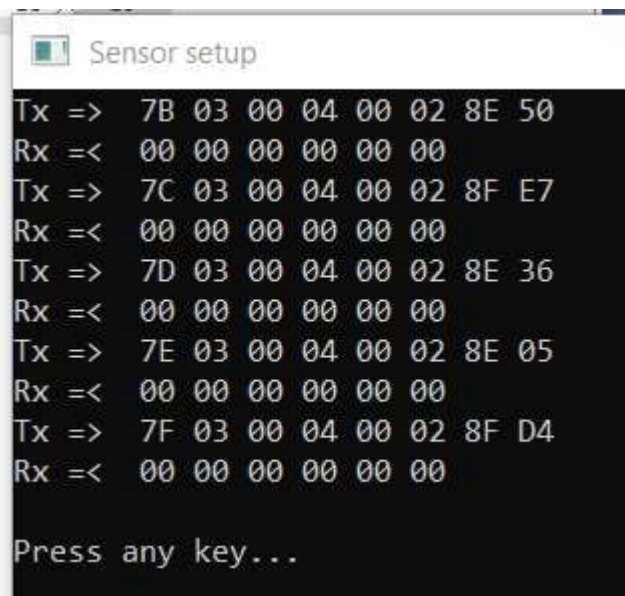
We select the port and data will begin to appear on the screen.

```
Select sensor:
0 SE58Sx-18
1 SOLAR-360 MB
Enter choice: 1

Sensor selected : SOLAR-360 MB

The following serial ports were found:
0 - CNCA0
1 - CNCB0
2 - COM40
3 - COM41
Enter choice: 2
Select port : COM40
Verifying correct parameters
Tx => 02 03 00 04 00 02 85 F9
Rx =< 00 00 00 00 00 00
Check 19200 bps
Tx => 02 03 00 04 00 02 85 F9
Rx =< 00 00 00 00 00 00
Tx => 01 03 00 04 00 02 85 CA
Rx =< 00 00 00 00 00 00
Tx => 02 03 00 04 00 02 85 F9
Rx =< 00 00 00 00 00 00
```

It ends with the following screen



```
Sensor setup

Tx => 7B 03 00 04 00 02 8E 50
Rx =< 00 00 00 00 00 00
Tx => 7C 03 00 04 00 02 8F E7
Rx =< 00 00 00 00 00 00
Tx => 7D 03 00 04 00 02 8E 36
Rx =< 00 00 00 00 00 00
Tx => 7E 03 00 04 00 02 8E 05
Rx =< 00 00 00 00 00 00
Tx => 7F 03 00 04 00 02 8F D4
Rx =< 00 00 00 00 00 00

Press any key...
```

If the system is unable to find the encoder, change the order of cables A and B, maybe they are reversely connected.

When everything has gone correctly, it presents us with the following screen.

```

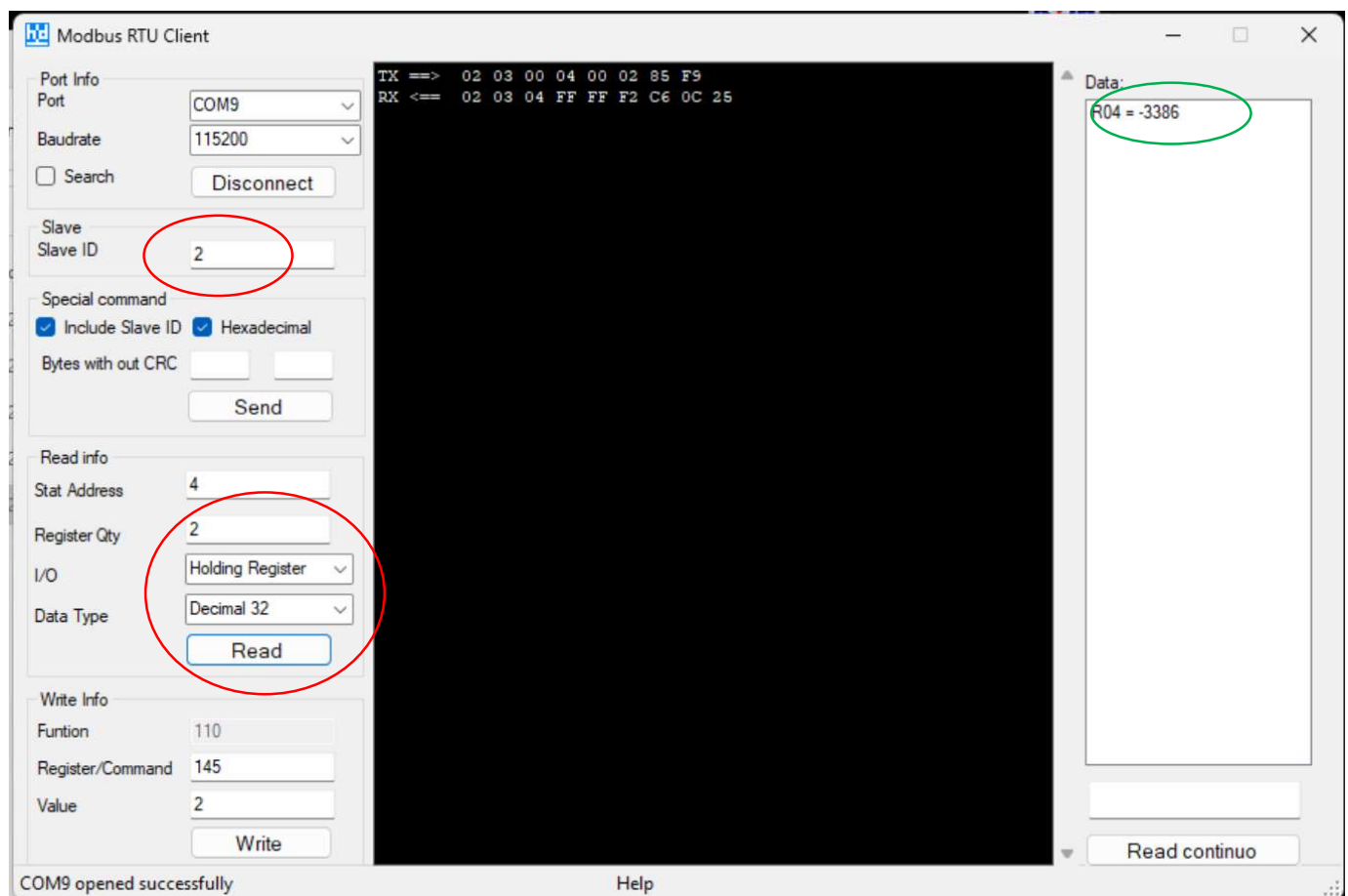
Tx => 61 03 00 04 00 02 8C 6A
Rx =< 00 00 00 00 00 00
Tx => 62 03 00 04 00 02 8C 59
Rx =< 00 00 00 00 00 00
Tx => 63 03 00 04 00 02 8D 88
Rx =< 00 00 00 00 00 00
Tx => 64 03 00 04 00 02 8C 3F
Rx =< 64 03 04 00 02 14 DD A1 AC

Found slave 100 at 38400 bps
Changing address 100 to 2
Tx >= 64 6E 91 02 92 98
Rx =< 64 6E 91 00 13 59 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Changing bauds 38400 to 115200
Tx >= 02 6E 8F 07 45 B3
Rx =< 78 80 00 00 00 00
Changing filter to 0.5 Hz
TX => 02 06 00 09 00 03 19 FA
RX =< 02 06 00 09 00 03 19 FA
Press any key...

```

## Verification

We open the Modbus RTU Client program, select the port that we have connected to the inclinometer and speed 115200 and Slave ID 2. We press Connect.



We configure the parameters that are shown on screen and press the Read button at the top, it indicates the value of the reading.

## Azimuth SE58Sx

We will use the same procedure and, if everything is correct, it will ask us the direction of rotation.

```
Tx => 01 03 00 00 00 02 C4 0B
Rx =< 00 00 00 00 00 00
Tx => 01 03 00 00 00 02 C4 0B
Rx =< 01 03 04 00 02 00 00 5B F3

Found slave 1 at 19200 bps
Changing bauds 19200 to 115200
Tx >= 01 CC 11 B5 0C
Rx =< 01 CC 11 B5 0C
Set rotation direction
  1 CW
  2 CCW
Enter choice:
```

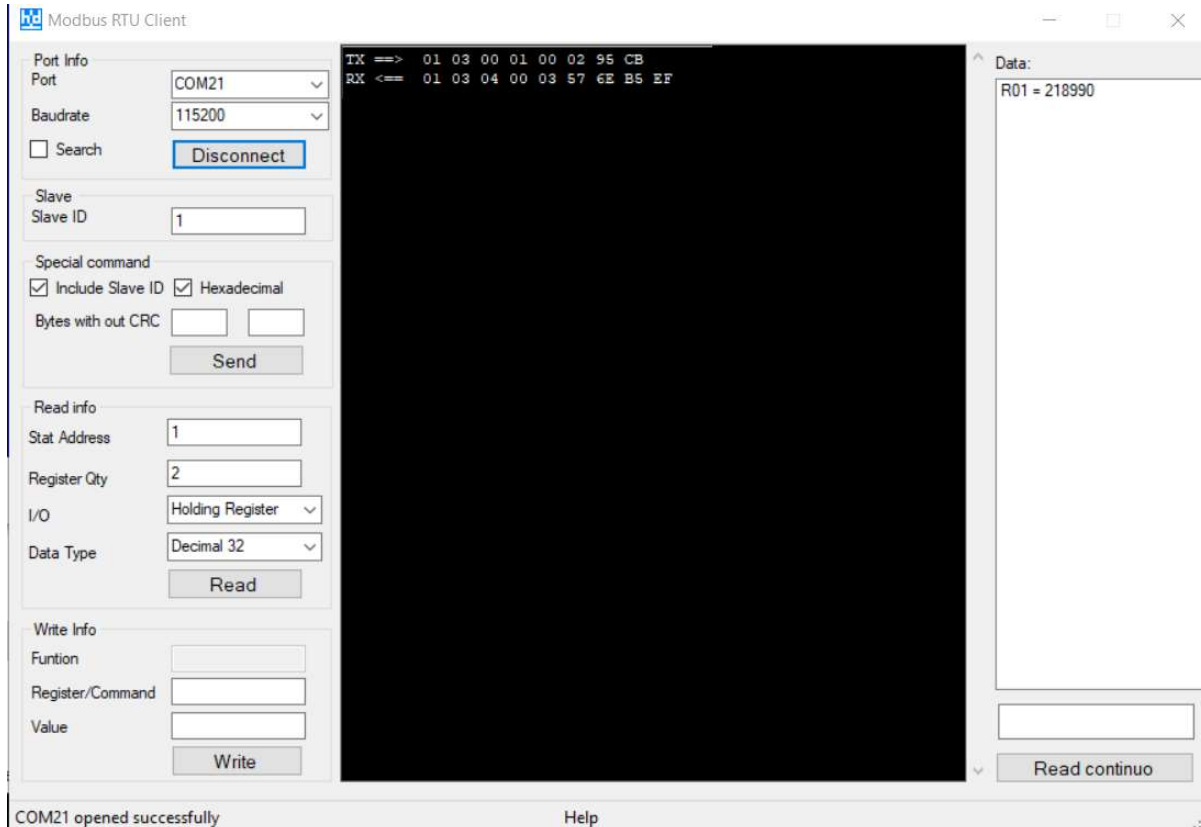
Select address

```
Set rotation direction
  1 CW
  2 CCW
Enter choice: 1
Changing rotation direction to CW
Tx >= 01 CC 01 B4 C0
Rx =< 01 CC 01 B4 C0
Press any key...
```

The encoder is already configured.

## Verification

We open the Modbus RTU Client program, select the port that we have connected to the encoder and speed 115200 and Slave ID 1. We press Connect.



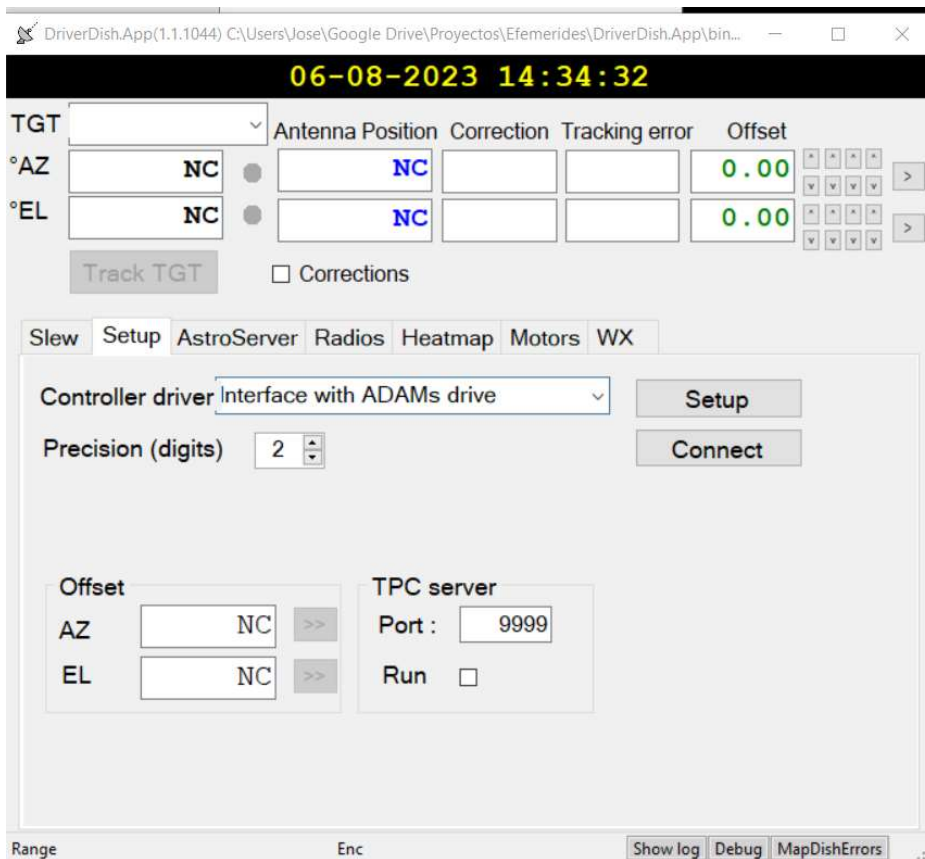
We configure the parameters, the same that are shown on screen, and press the Read button at the top, it indicates the value of the reading.



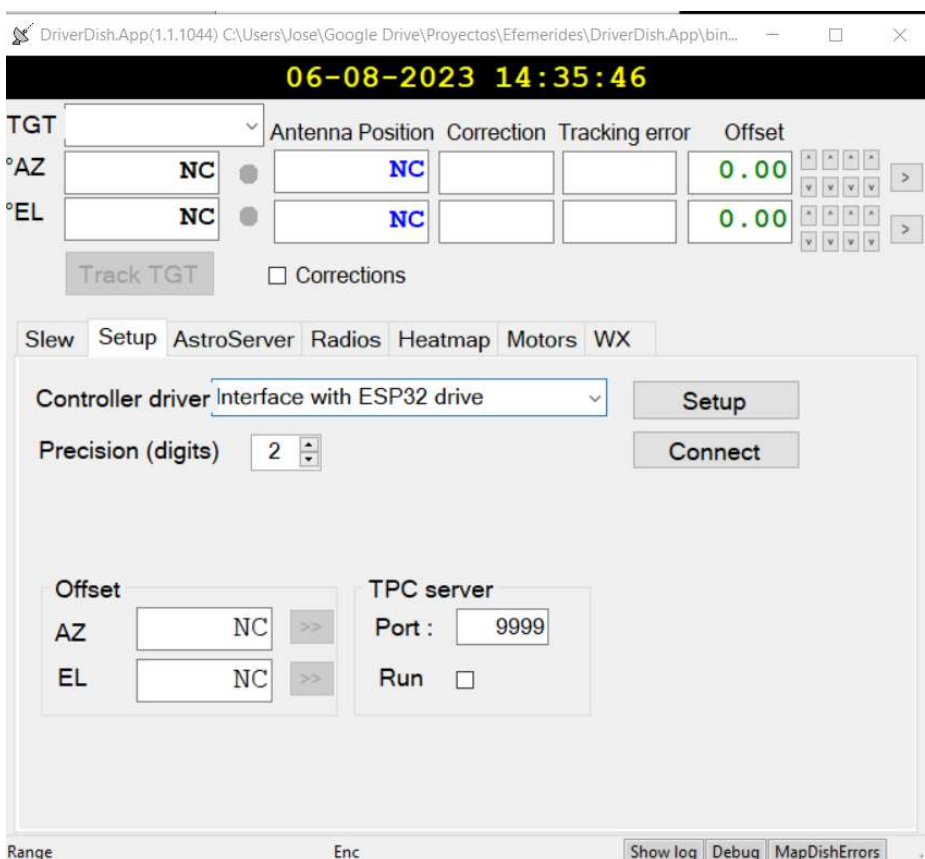
# DriverDish

We connect the encoders to the controller and the controller to the PC and restart the program.

We select the setup tab.



In controller driver we select **Interface with ESP32 drive**.



Press **setup**.

SetupDialogForm

Comm Port:  Connect Debug version

Baud rate: 500000

Correction file:

Azimuth Elevation Sensors Test speed

Offset:  >> Type: <Press All>

PWMHighSpeed:  >> Turn: <<  >>

° change SlowSpeed:  >> Bits:  >>

PWM Slowspeed:  >> Multi:  >>

Parking: 180.000 Encoder: <<

Accuracy:  >>

Precision (digits): 2

Limit min:  >>

Limit max:  >>

Timeout (s): Reset  >>

Default << All Reset

OK Cancel

We select the COMM Port and press connect, until the controls that are now gray are enabled.

SetupDialogForm

Comm Port: COM25 Disconnect Debug ControllerDish version 2.4.159

Baud rate: 500000

Correction file:

Azimuth Elevation Sensors Test speed

Offset:  >> Type: <Press All>

PWMHighSpeed:  >> Turn: <<  >>

° change SlowSpeed:  >> Bits:  >>

PWM Slowspeed:  >> Multi:  >>

Parking: 172.000 Encoder: <<

Accuracy:  >>

Precision (digits): 2

Limit min:  >>

Limit max:  >>

Timeout (s): Reset  >>

Default << All Reset

OK Cancel

In the azimuth tab we select type: **ENCODER BINARY**.

SetupDialogForm

Comm Port: COM25  
Baud rate: 500000  
Correction file:

Disconnect Debug ControllerDish version 2.4.159

Azimuth Elevation Sensors Test speed

Offset  >> Type: ENCODER BINARY  
PWMHighSpeed  >> Turn <<  >>  
° change SlowSpeed  >> Bits  >>  
PWM Slowspeed  >> Multi  >>  
Parking: 172.000 Encoder <<  >>  
Accuracy  >>  
Precision (digits): 2  
Limit min  >>  
Limit max  >>  
Timeout (s) Reset  >>

Default << All Reset

OK Cancel

In the elevation tab we select type: **SOLAR360MB**

SetupDialogForm

Comm Port: COM25  
Baud rate: 500000  
Correction file:

Disconnect Debug ControllerDish version 2.4.159

Azimuth Elevation Sensors Test speed

Offset  >> Type: SOLAR360 MB  
PWMHighSpeed  >> Encoder X <   
° change SlowSpeed  >> Encoder Y <   
PWM SlowSpeed  >>  
Parking: 21.000  
Accuracy  >>  
Precision (digits): 2  
Limit min  >>  
Limit max  >>  
Timeout (s) Reset  >>

Default << All Reset

OK Cancel

Click OK to exit to the main screen.

We can now verify that we have already information about antenna position.

DriverDish.App(1.1.1044) C:\Users\Jose\Google Drive\Proyectos\Efemerides\DriverDish.App\bin...

06-08-2023 14:41:39

TGT

▼

Antenna Position

Correction

Tracking error

Offset

°AZ

NC

●

171.994

0.000

0.003

▲

▼

↶

↷

↵

°EL

NC

●

20.970

0.000

0.009

▲

▼

↶

↷

↵

Track TGT

☐ Corrections

Slew

Setup

AstroServer

Radios

Heatmap

Motors

WX

Controller driver

Interface with ESP32 drive

Setup

Precision (digits)

3

Disconnect

Offset

AZ

8.651

>>

EL

17.340

>>

TPC server

Port :

9999

Run

☐

Range

Enc 0/143201

Show log

Debug

MapDishErrors

DriverDish.App(1.1.1044) C:\Users\Jose\Google Drive\Proyectos\Efemerides\DriverDish.App\bin... — □ ×

**06-08-2023 14:41:39**

TGT ▼ Antenna Position Correction Tracking error Offset

°AZ NC ● 171.994 0.000  0.003 ▲ ▲ ▲ ▲

°EL NC ● 20.970 0.000  0.009 ▼ ▼ ▼ ▼

Track TGT ☐ Corrections

Slew Setup AstroServer Radios Heatmap Motors WX

Controller driver Interface with ESP32 drive Setup

Precision (digits) 3 Disconnect

Offset

AZ 8.651 >>

EL 17.340 >>

TPC server

Port : 9999

Run ☐

Range Enc 0/143201 Show log Debug MapDishErrors

# JPLastroserver

We start the program

JPLastroserver.App(2.0.423)

**8/6/2023 2:51:44 PM**

AZ	<input type="text" value="249.4577"/>	Spacecraft	<input type="text" value="Sun"/>	Home Latitude	<input type="text" value="41.5695334"/>
EL	<input type="text" value="45.2654"/>	Freq (MHz)	<input type="text" value="0.000000"/>	Home Longitude	<input type="text" value="2.013942"/>
		RxFreq	<input type="text" value="0.000000"/>	Home Height asl (Km)	<input type="text" value="0.32"/>
	<input type="text" value="0.000000000"/>	Drift	<input type="text" value="0.000000000"/>	<input checked="" type="checkbox"/> Server	<input checked="" type="checkbox"/> SD
				SigDigger	

```
2023-Aug-06 15:42:36.000 * 136.400213252 16.643675429 1.5173549944E+08 0.0349076 260.058877314
36.083949355
2023-Aug-06 15:43:35.000 * 136.400862159 16.643485266 1.5173550153E+08 0.0356967 260.244681916
35.902599866
2023-Aug-06 15:44:34.000 * 136.401511094 16.643295095 1.5173550366E+08 0.0364807 260.429911483
35.721149573
2023-Aug-06 15:45:33.000 * 136.402160056 16.643104917 1.5173550583E+08 0.0372596 260.614573329
35.539600681
2023-Aug-06 15:46:32.000 * 136.402809046 16.642914731 1.5173550805E+08 0.0380334 260.798674698
35.357955368
2023-Aug-06 15:47:31.000 * 136.403458062 16.642724538 1.5173551032E+08 0.0388020 260.982222768
35.176215796
2023-Aug-06 15:48:30.000 * 136.404107106 16.642534337 1.5173551263E+08 0.0395654 261.165224646
34.994384106
2023-Aug-06 15:49:29.000 * 136.404756178 16.642344129 1.5173551499E+08 0.0403237 261.347687371
34.812462422
2023-Aug-06 15:50:28.000 * 136.405405277 16.642153913 1.5173551739E+08 0.0410768 261.529617918
34.630452845
2023-Aug-06 15:51:27.000 * 136.406054404 16.641963690 1.5173551984E+08 0.0418246 261.711023196
34.448357460
Data download finished
3601 points added
Start server
Start timer
Start server in 8888 port
```

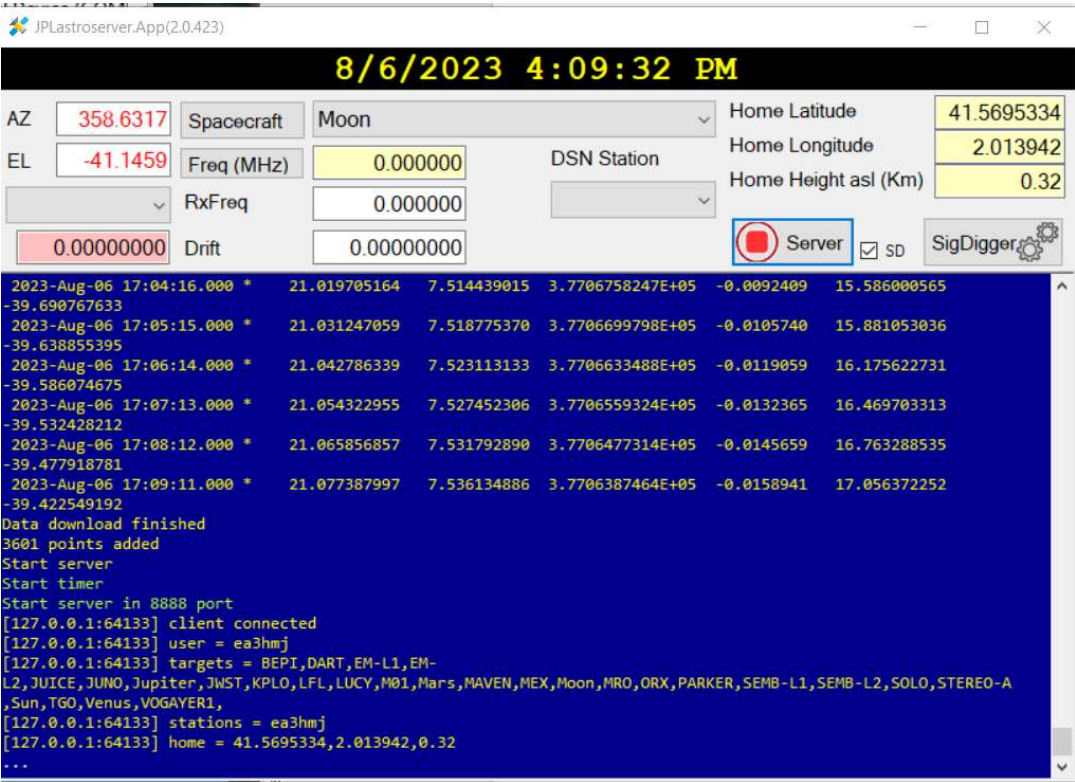
We must put the appropriate values of latitude, longitude and height of our location.

We close the program.



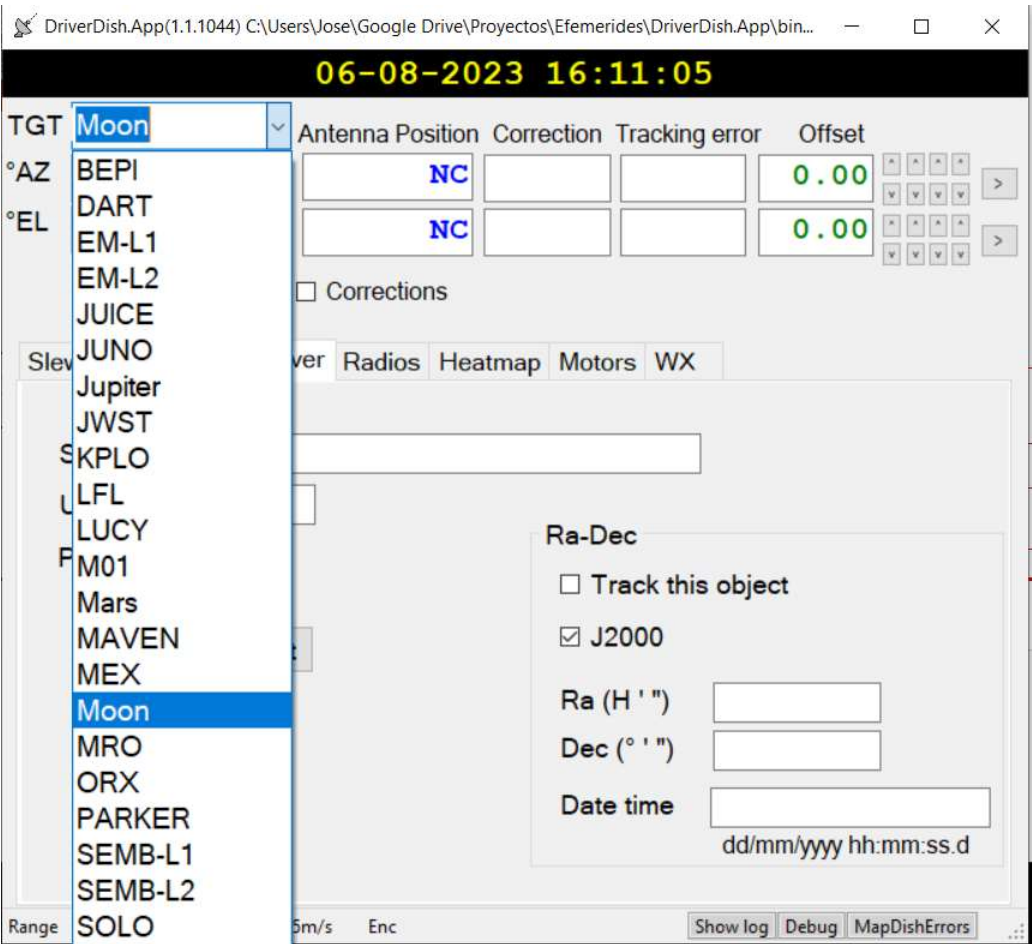
# Verification of communications

We open JPLastroserver first and then open DriverDish.



In the last lines of the log we can verify that driverdish has been connected.

We select an object in DriverDish.



The program already obtained ephemeris from the server and we can track the object selected.

DriverDish.App(1.1.1044) C:\Users\Jose\Google Drive\Proyectos\Efemerides\DriverDish.App\bin... — □ ×

**06-08-2023 16:12:44**

TGT **Moon** Antenna Position Correction Tracking error Offset

°AZ **359.64** **NC** **0.00**

°EL **-41.14** **NC** **0.00**

☐ Corrections

Slew Setup **AstroServer** Radios Heatmap Motors WX

Server

User :

Port :

**Ra-Dec**

☐ Track this object

☒ J2000

Ra (H ' ")

Dec (° ' ")

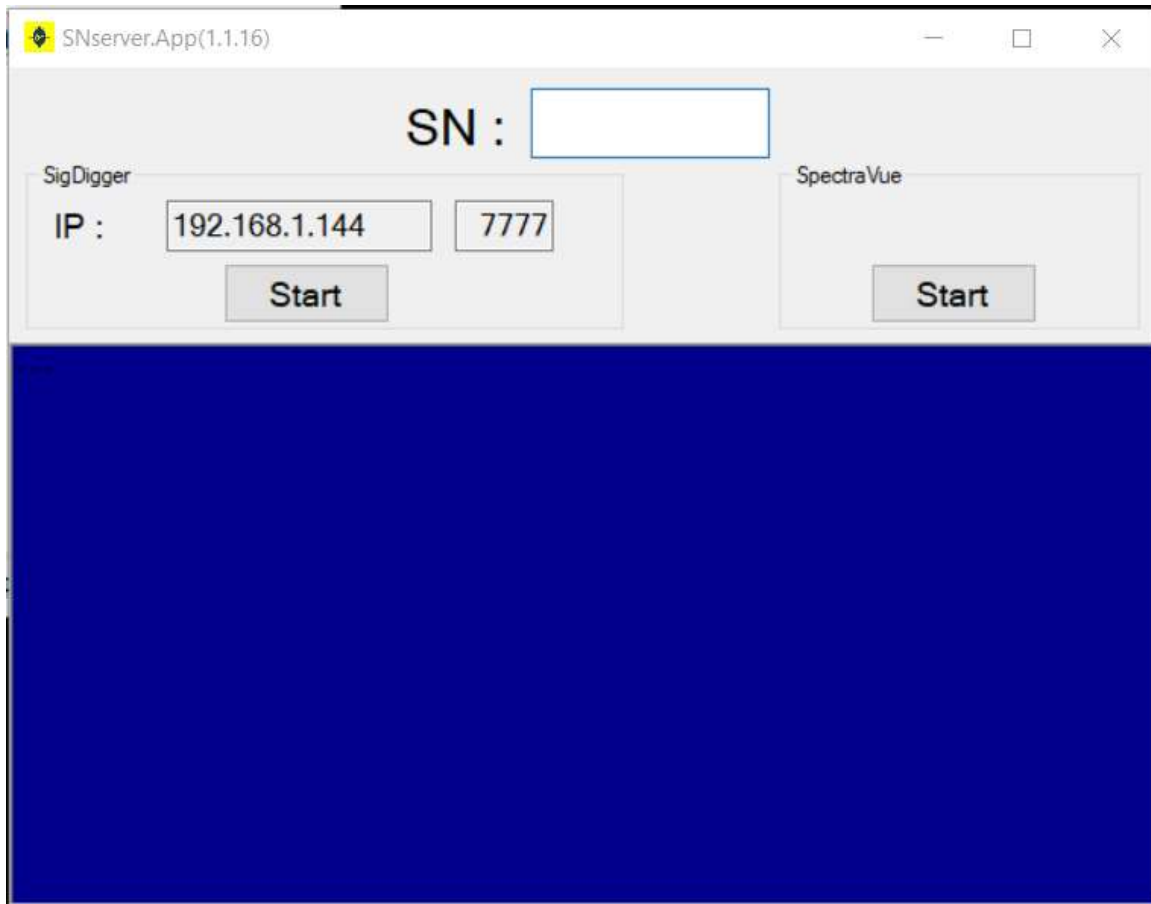
Date time

Range 376987 Kms 0.06176m/s Enc

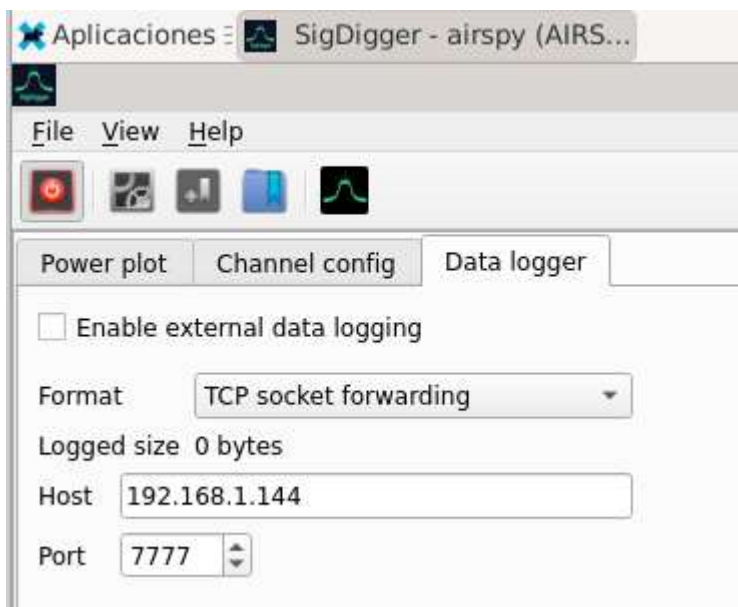
## SNserver

We can connect DishDriver with SpectraVue or SigDigger through SNserver.

This program acts as an interface between the information it receives from these programs and DriverDish.



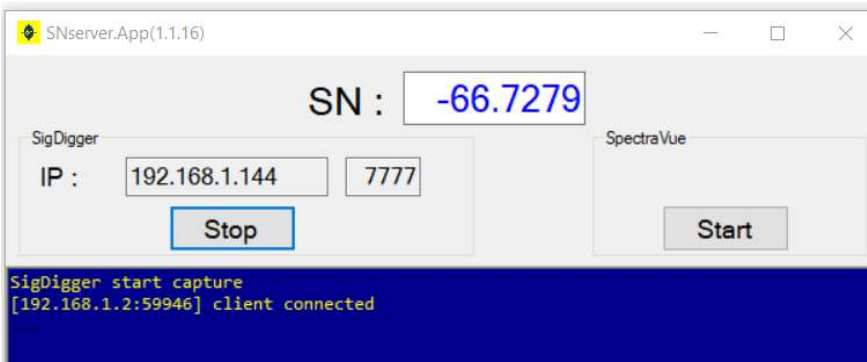
This screen gives us the information to configure SigDigger.



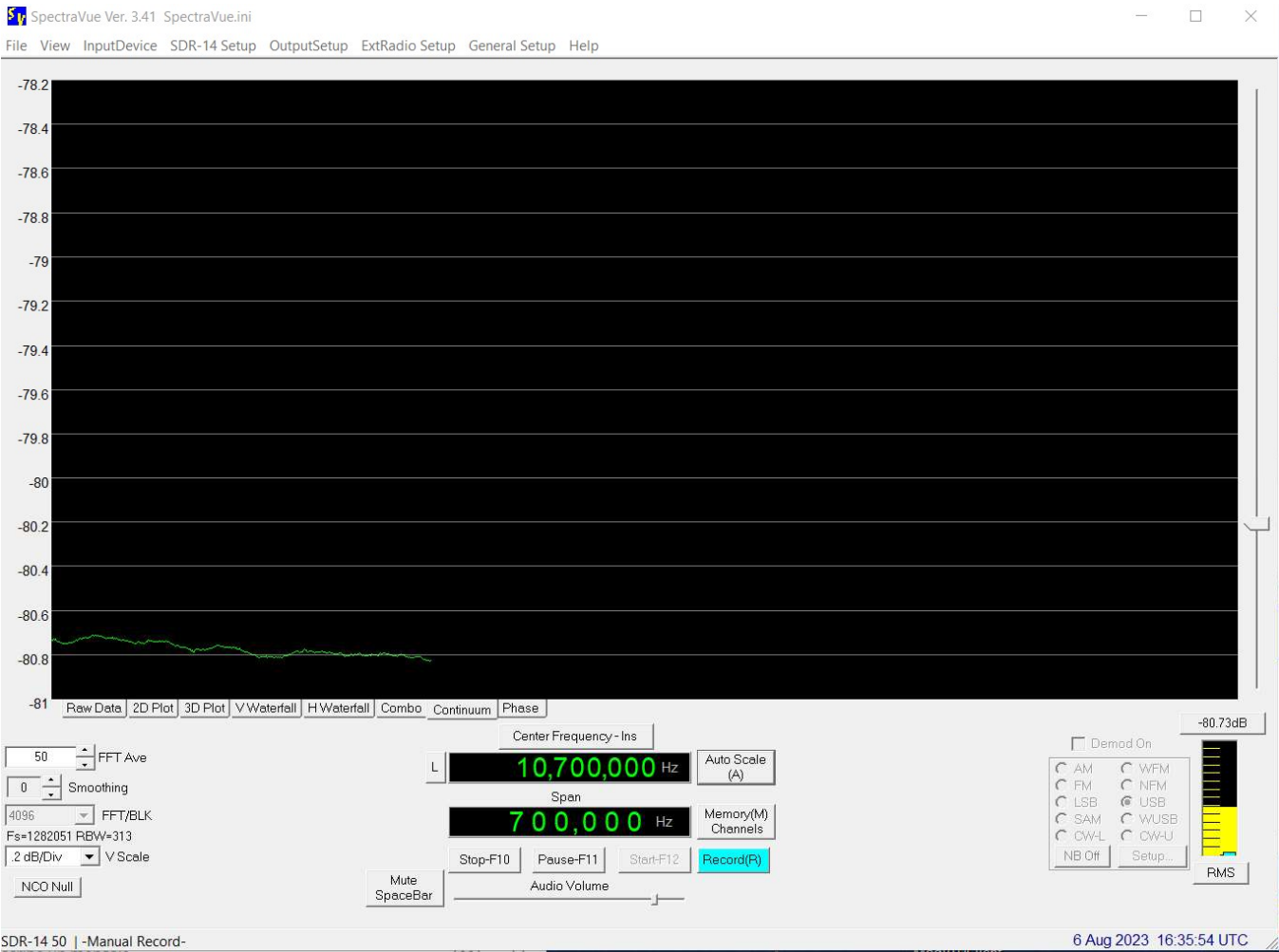
This information must be configured at SigDigger to receive information.



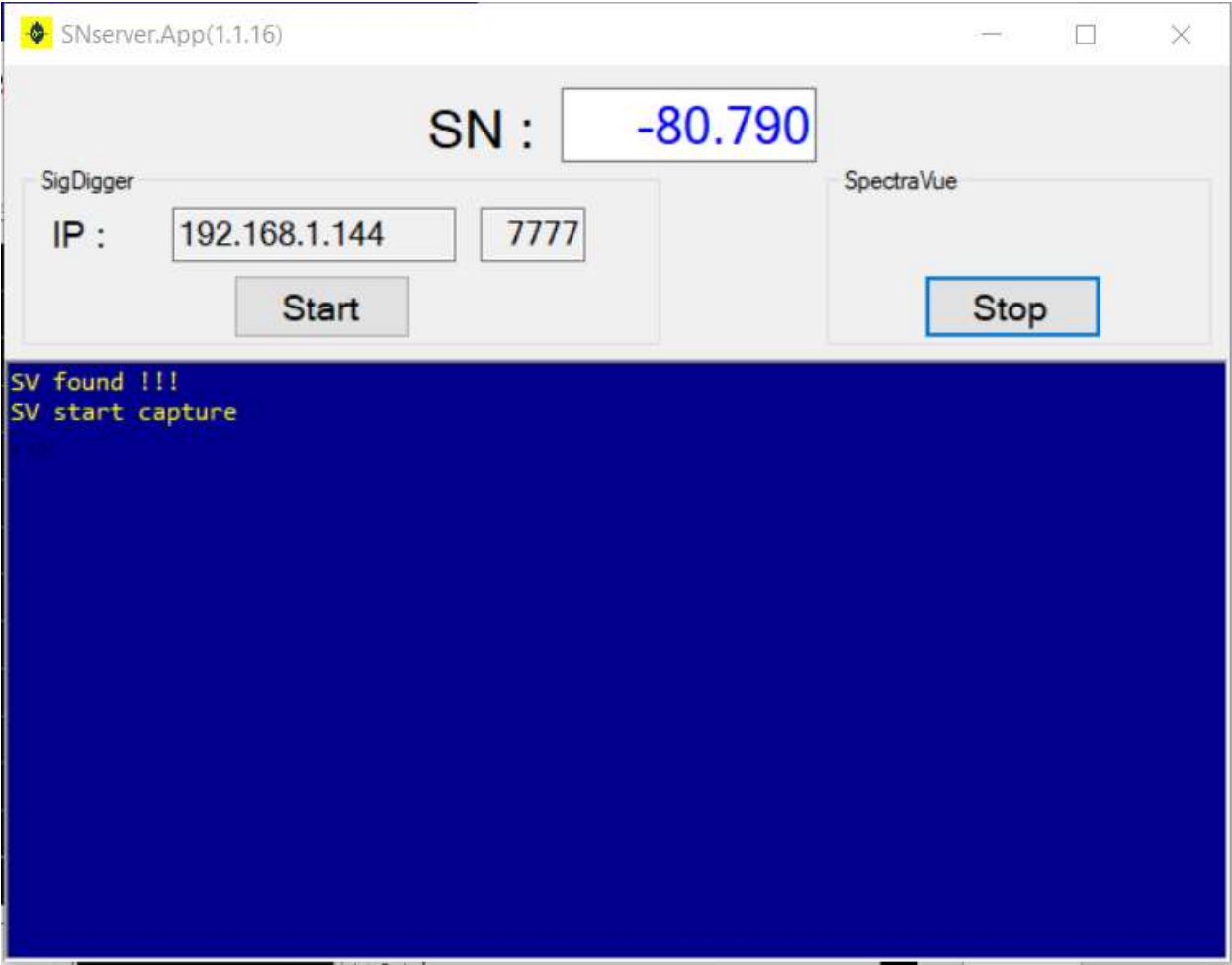
Press Start in SNserver at the SigDigger side.



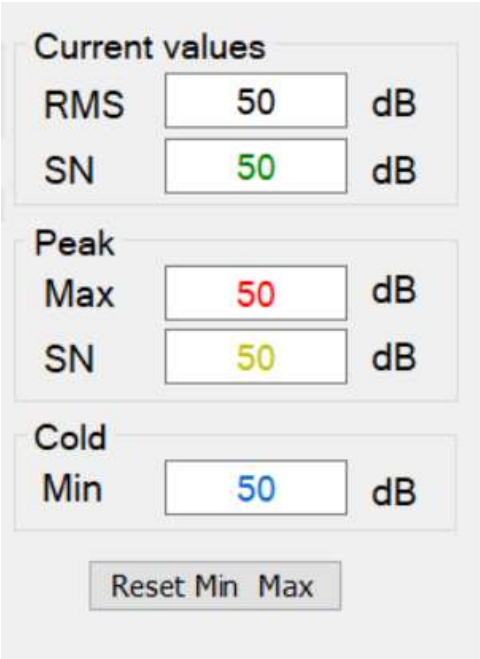
But if we want to connect through SpectraVue it is easier, we must have the program started.



Press Start in SNserver at the SpectraVue side.



If we have started DriverDish before and SNserver did not connect, we must do it manually by pressing the **Reset Min Max** button.



DriverDish.App(1.1.1044) C:\Users\Jose\Google Drive\Proyectos\Efemerides\DriverDish.App\bin...

## 06-08-2023 16:39:58

TGT Moon

°AZ NC

°EL NC

Antenna Position

Correction

Tracking error

Offset

NC

NC

NC

0.00

NC

NC

0.00

0.00

☐ Corrections

Slew Setup AstroServer Radios **Heatmap** Motors WX

**Configuration**

**Mode**

**Current values**

Slew degrees ± 1 °

Steps 0.1 °

Error degrees ± 0.01

Comment here

☒ Heatmap

☐ Autoposition

RMS -80.81 dB

SN 0.02 dB

**Peak**

Max 0.00 dB

SN 80.83 dB

**Cold**

Min -80.83 dB

Range Enc SV Run Show log Debug MapDishErrors

We already have the information.

If we first start SNserver when starting DriverDish it automatically connects.