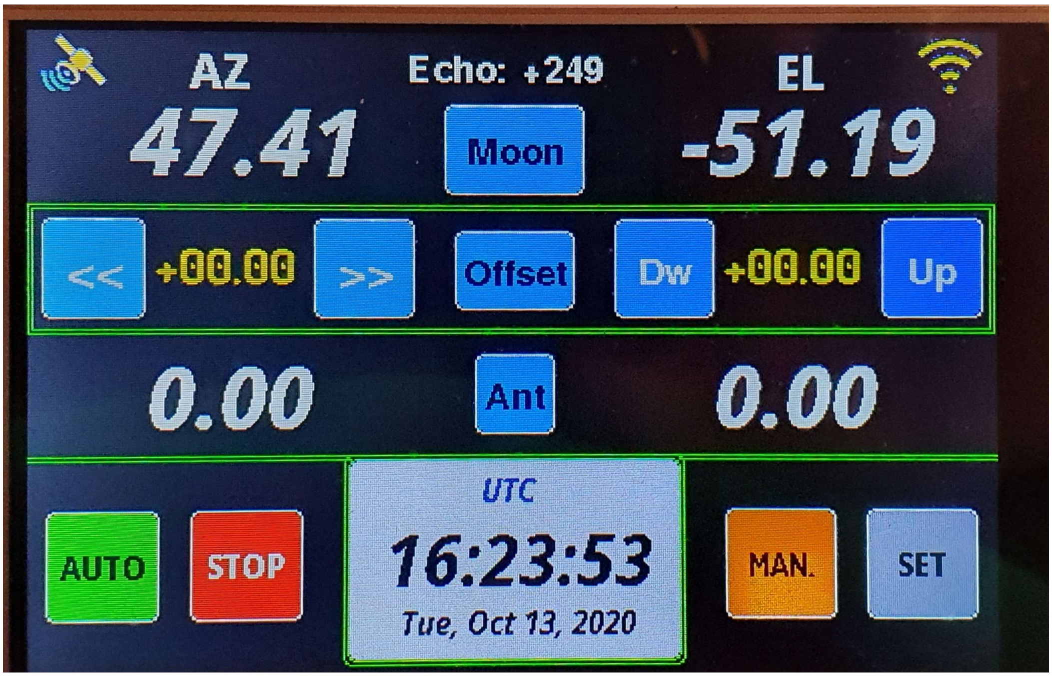
AnTrack–Pro©
an EME Antenna Controller
and more…..

HB9DRI



October 2024

v1.14

(under construction)

History and Design theory:

1.1 History:

16 years ago, during the 13th EMe Conference in Florence, Italy, I had the opportunity to personally meet Hannes OE5JFL. In the area dedicated to exhibitors, Hannes had a small table where he was presenting his new antenna controller. The great novelty of this controller was the use, for the first time, of low-cost absolute encoders such as the MAB25.

In HB9Q, we used the first version of that antenna controller. The software was practically the same; the interface was a 20x4 LCD screen. But the great novelty was the use of these low-cost encoders with an impressive 12-bit resolution, something reserved for the A2-S-S encoder from US Digital. The big difference is that for an antenna to implement US Digital's absolute encoders, it would cost an average of 1,000 USD (even today due special cables and programming splitter) when 2 MAB25 cost less than 80.00 dollars.

I became very interested in the unit and asked Hannes to sell me a complete set. Unfortunately, Hannes did not have any systems available and told me that he had no intention of going into production. He did not like the idea of having to respond to customer demands. He offered to give me all the information, the Gerbers files to produce the PCBs, as well as the BOM list of a supplier in Germany where he bought his parts.

He suggested that to reduce my final cost I produce at least 5 sets of PCB's and order parts to make 5 antenna controllers.

This is how the DRIACS-G2 was born. The only condition that Hannes proposed to me was that I keep its callsign OE5JFL on the welcome screen, and that's what I did, always giving 100% of the design credit to him; my job was simply to reproduce, sell and provide support.

So, two years later, in 2010, at the 14th EME Conference in Dallas, I made the official presentation of the DRIACS-G2.

My contribution to Hannes's work was not only to produce and market, but together with him we produced and improved the FW of the controller on several occasions. I wrote extensive notes on how to program it and how to solve some noise problems in the data line of the encoders and published all the information so that whoever wanted to produce it on their own could do so. I also produced software to control the unit remotely from the PC.

Unfortunately, the software was hacked, and many “good friends” did not buy the software from me but downloaded the patch from the network. Andy, G4JNT helped me with the development of a GPS interface. It was the first time that a GPS unit was incorporated into an antenna controller to maintain the time, date and geographic position with high precision.

It didn't take long for copies of the DRIACS-G2 to appear. Some developments were only PCBs, other equipment was complete but of quite low quality in its parts and assembly. The DriTrack software to manage the controller from the PC was offered free of charge to anyone who had purchased the unit from me. Unfortunately, many copies were sold stating that they were produced by me, which generated a large complaint from some who felt scammed.

Having sold more than 600 units and starting my journey through South Africa and Thailand, I decided that it was time to retire the DRIAC-G2 and stopped producing it.

However, I always wanted to be able to design my own controller, where I could integrate many new functions and where above all I could have direct control of the FW, this is how the idea of Antrack-Pro that I describe here was born.

The central idea of this new controller is focused on its high operability and multifunction at the lowest possible cost on the market.

As you will see, the central part of the Antrack-Pro is a set of PCBs that will be sold at cost plus shipping costs and where the FW will be totally free for those who buy the LinkRF PCBs. It has been several years of work, debugging the software, creating own libraries with some long periods of inactivity due to my other work commitments and COVID crisis but it’s finally here.

Considering that price is an important factor in the world of amateur science, I can guarantee that the Antrack-Pro has an unmatched cost/benefit compared with similar products.

73 de Alex, HB9DRI

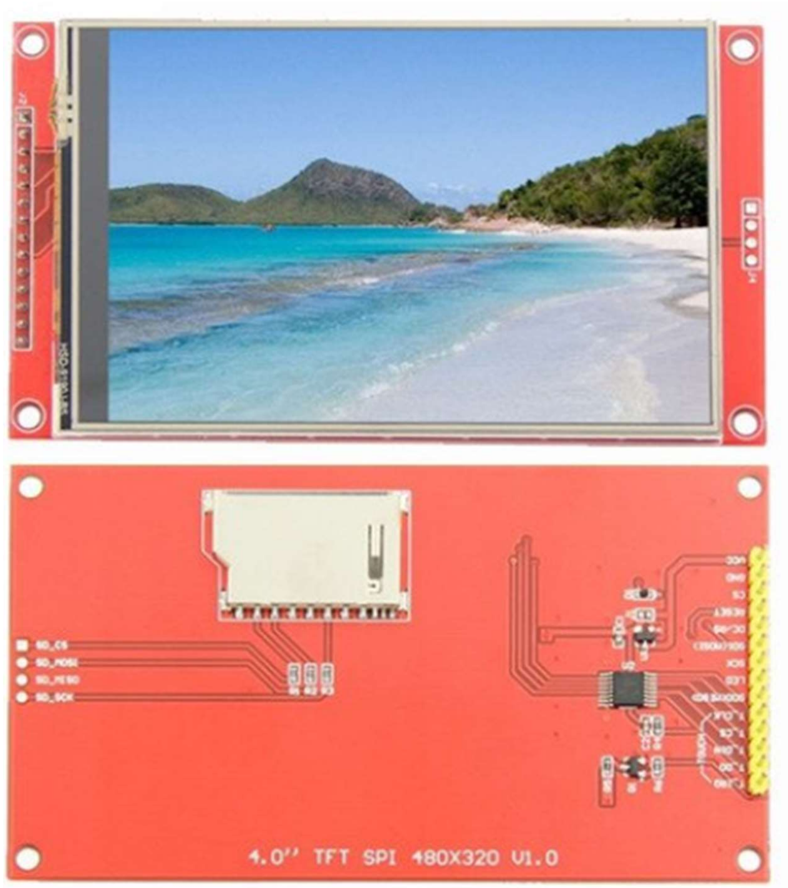
2 Characteristics:

- TFT interface 480x320 (color touch screen) = 153,600 pixels
- 32-bit Dual Core 240 Mhz CPU up to 16MB flash data memory
- Graphic interface design using GuiSlice under MIT Open License
- Mathematical calculations and some dependency libraries are design and implemented by LinkRF (FW is NOT and open-source project)
- Concept of an encoder hub from ON4CDU testing jig.
- GPS U-Blox Neo M8N providing GNSS (GPS, Galileo, Glonass, Beidu)
- RTC (real-timeclock) on board, non-volatile for 5 years with CR2032.
- WIFI module included with the ESP32 board.
- Ethernet 100 MB when W5500 board is added.
- NTP Stratum 1 Server included based on the GPS data and RTC clock, you can synchronize your PC clock just pointing the NTP client software on your PC's to the Antrack-Pro IP address. No need of internet connection for that.
- NTP Client to synchronize the Antrack Pro time/date if GPS and/or internet is not available.
- You can track by predefine touch buttons Sun, Moon, Sagittarius, Cassiopeia, Taurus, Leo and Pictor.
- You can track any position introducing RA and DEC values.
- Display your home echo frequency when Moon target is selected and the properly band is selected.
- Include a novel "Till Correction Table" with 60 AZ positions (1 entry each 6 AZ degrees) to correct any error in the verticality antenna installation, very use full for big dishes and/or high MW operations.
- Soft start and Soft Stop based in Pulse Width Modulation (PWM)
- Support inexpensive IBT_2 Hybrid Bridges up to up to 20amps 34 vdc or more to control AZ and EL electric DC motors.
- Pre-define offset in AZ and EL to easy install the encoders.
- Automatic FW update over internet OAT (Over the Air), no need any special software to upload new FW and reprograming.
- Support Absolute encoders from 9 to 16bits (or more)
- Tracking calculations double precision FP, ESP32 decimal precision verified to 15 decimals, displaying 0.01 REAL accuracy.
- Support MAB25, 28, ETS25, HH12, HH12 inclinometer, Posital AC360 inclinometer, A2-S-S US digital and the new BAX-16 with 0.01 accuracy and 0.005 degrees repeatability.

- Other kind of encoders can be supported with the properly code modification on the encoder hub software.

3 Hardware Modules:

The Antrack-Pro is a DIY antenna controller built using a custom ‘mother board.’



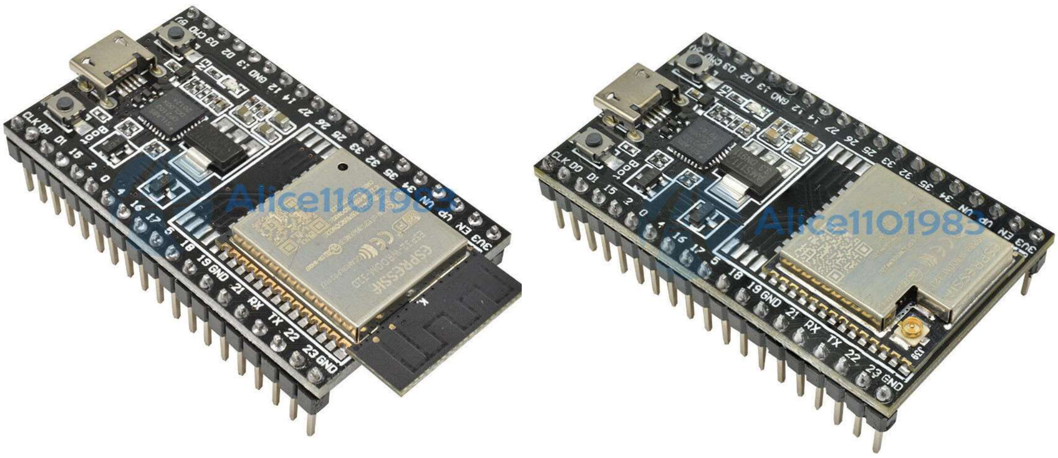
and off the shelf modules. The modules reduce the problems with component sourcing as well as the inevitable solder issues. The Antrack-Pro was built around specific modules, and the user must purchase these boards to have the Antrack-Pro to function.
All the modules used in development were purchased from Ebay or AliExpress.

No links to specific providers are given but the images included show the boards you MUST use if you build the Antrack-Pro.

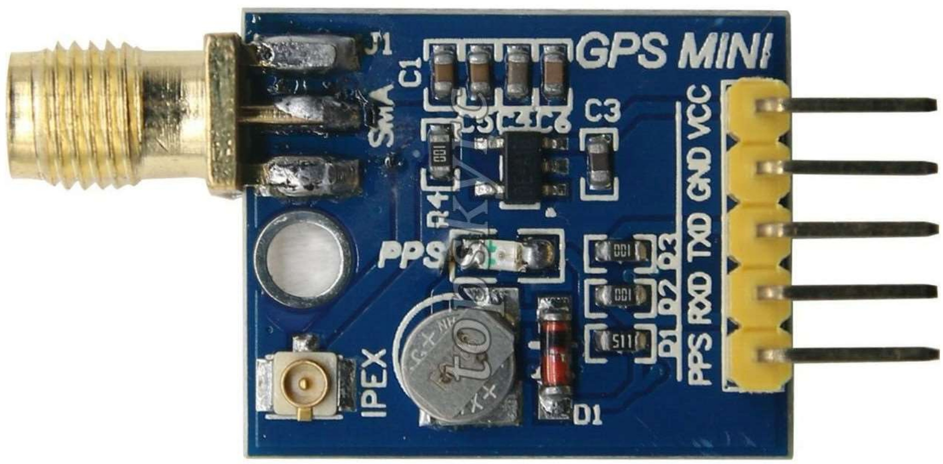
- Main board

- 4” LCD + touch screen. The GUI display is designed around a 480 x 320 with resistive touch screen using a ST7796S graphics driver. You are free to source this card from another provider. We recommend to buy the v1.1

- ESP32-DevKitC. Use only Official Espressif development kits with 39pins. There are 2 versions.
    - Internal antenna, ESP32 DevkitC with 32D
    - External antenna, ESP32 DevkitC with 32U



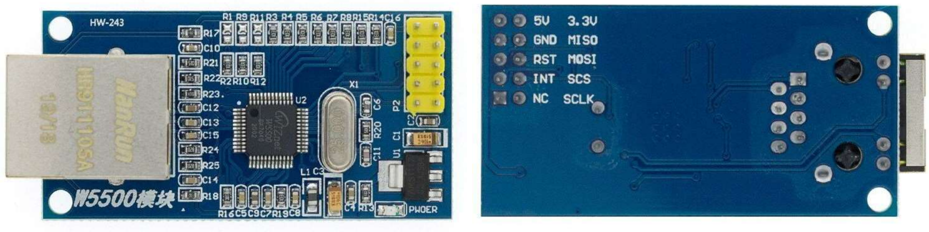
- uBlox M8N GPS module
-



https://www.ebay.com/itm/353021205696?itmmeta=01J0H6X9M7SAX50RARDFEGGTGZ&hash=item5231b41cc0:g:YUwAAOSwDQdeeyAP&itmprp=enc%3AAQAJAAAA4JbVfPzbug5YWrbE6PgunbBJeNY9NFvJri3HvycVqUincVaJDYQsXaeNakXHPMuKdEqb5j%2B6iP2hzG

HK5xvE4vEsHYIu49Z4NqCI3w1JruXRfVcR%2BVcK7pgA9p2gZl1fntIFTOX47qV5dwVPgLIpWkHxhsYowZ7ZtEQmWnqy%2Bm%2B7moD%2Bvh5lwLNlgQ8x0LO8fDFM1GmVyk7aen14s34mo8lKAi7nGtBAdq70rofTf%2BnJI%2BPPoeeuUg4LfcgOywnMTmUAftjoJ0ES6XBPH6hqfrpvXUsLUVlattDyBCOoIYGg%7Ctkp%3ABFBMqJr1poRk

W5500 Ethernet module (optional)



AnTrack-Pro Operation

Operate the controller is a straightforward task, you don’t have extra push button with multifunction per button to operate the controller.
All configuration and operation can be done via the touch screen (and/or) via the web interface, later more about the web interface.

The controller consists of 3 main parts:

1.- The controller attached to the touch screen.
2.- The encoder board
3.- The LDMOS power modules

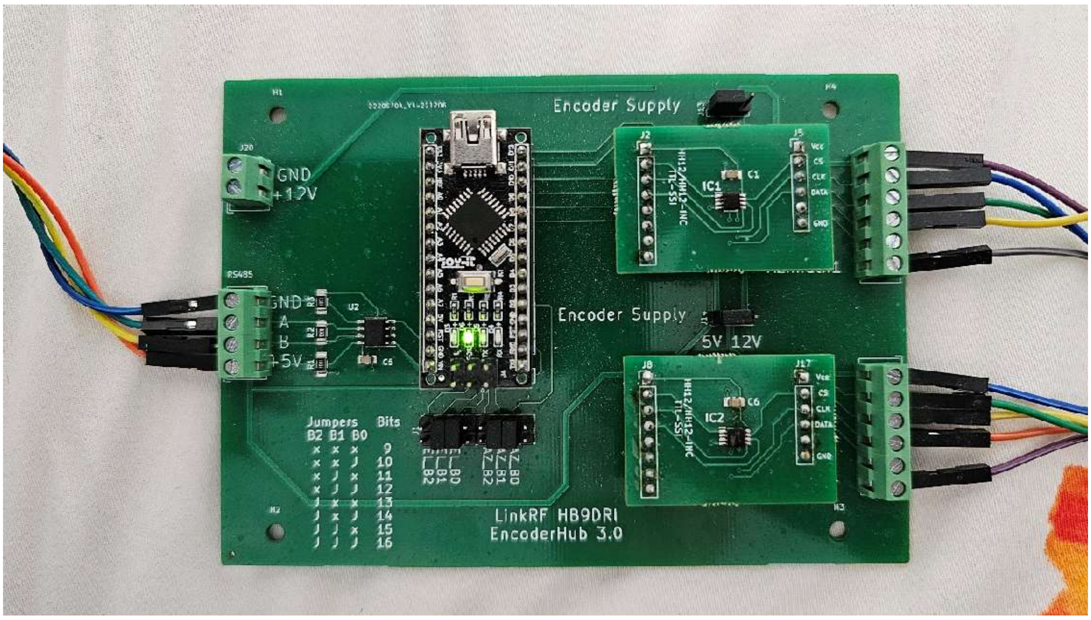
We adopt the concept of an encoder board based on the work done by ON4CDU, this approach made much easy to interface any kind of encoder, absolute or incremental, from 8 to 16 bits, if necessary, SSI, SEI, binary etc.

This encoder board will be installed near the antenna, the encoders will be attached to that board. Due the big number of encoders options its impossible to include from the beginning support for all kind of, we will provide access to the necessary part of the code if you want to include any kind of specific encoder

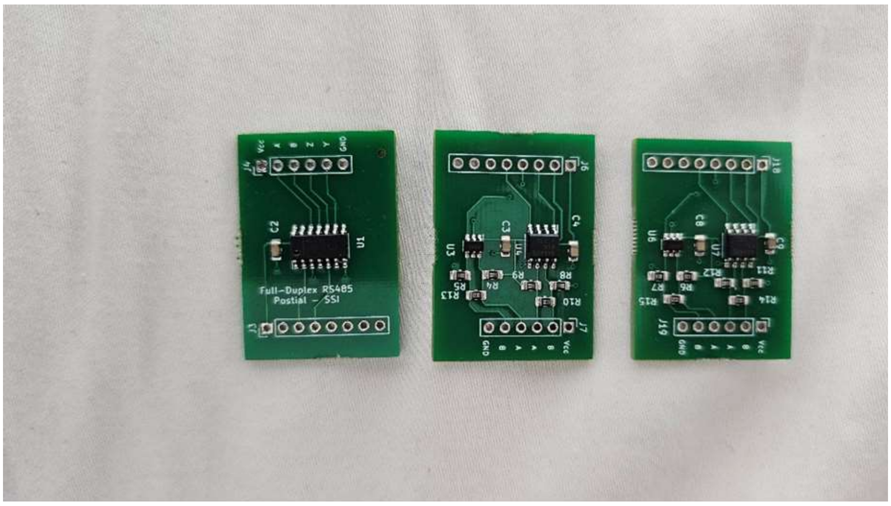
The configuration about the bit resolution is done via a set of jumpers. This encoder board runs an Arduino NANO and is connected to the controller board via two twisted pairs cable, a third twisted pair will provide 5vdc to the encoder board, the connection is done via RS485, that permit up to 1.2km distance between the controller and the encoder board.

For the time been encoders like the MAB25, 28, ETS25, HH12, HH12 Inclinometer, Posital AC360 inclinometer, A2S US digital are supported. The new LinkRF BAX-16 encoder with an accuracy of 0.01 degrees and a repeatability of 0.005 degrees is included in the support list.

To interface the encoders in a properly way the encoder hub will have small plug in pcb’s with the appropriate electronic interface, 6 of this pcb’s are included in DIY kit. The main reason to use this plugin boards refers to the different ways encoders communicate, for example US digital, Posital encoders and the new BAX-16 needs full duplex interface when the other encoders just need half duplex. A simple resistive and capacitive divider is an approach used by some ones, but we discover this cheap way has a trade off in stability, the distance you could manage and the reliability of the lectures.



When the encoder connected to the encoder hub works with 5vdc the power is provided via the twisted pairs cable coming from the controller. In case your encoders work with 12VDC you must attach to the encoder hub the 12vdc via a separate wire pair and move the DC jumpers near the plug-in boards to the 12vdc position.



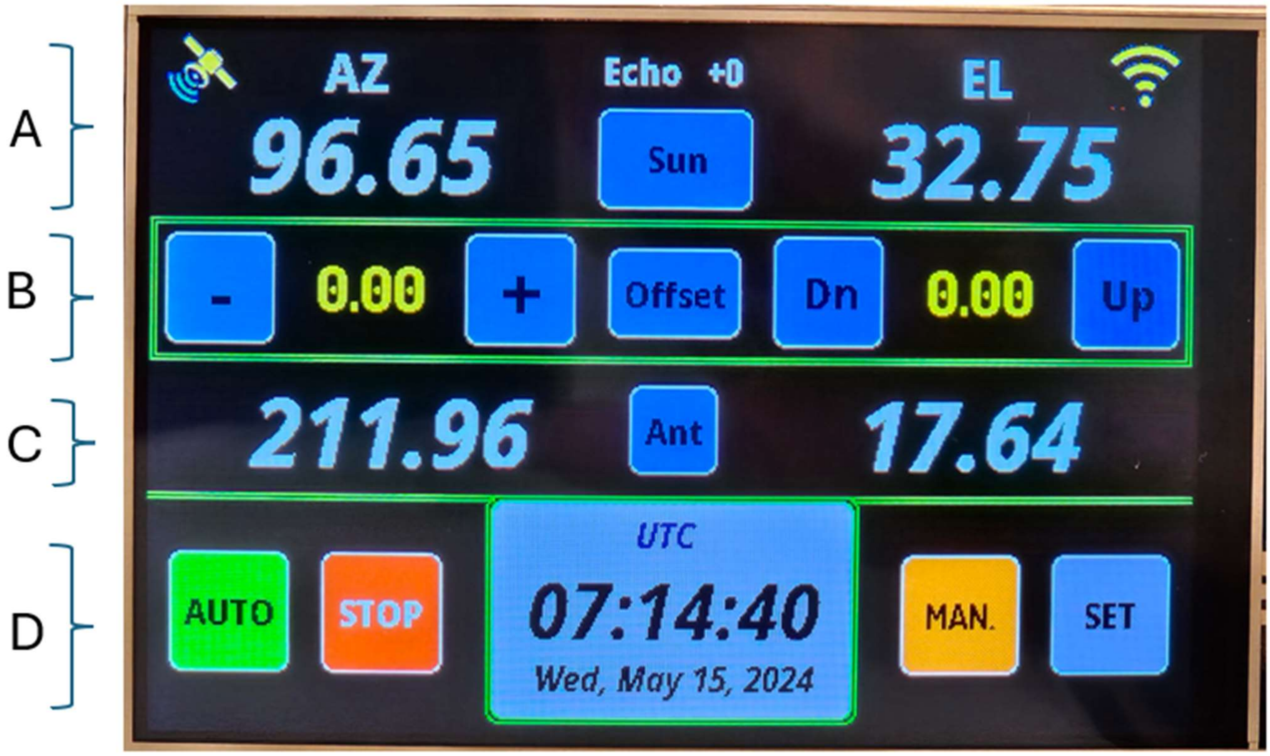
Last photo shows some plug-in boards used for different kind of encoders.

The third part of the controller is the LDMOS power modules, that part manage the DC for your AZ and EL electric engines. We opt for a generic and well proved power modules extensively used in robotics like the ITB-2 H bridge or similar:

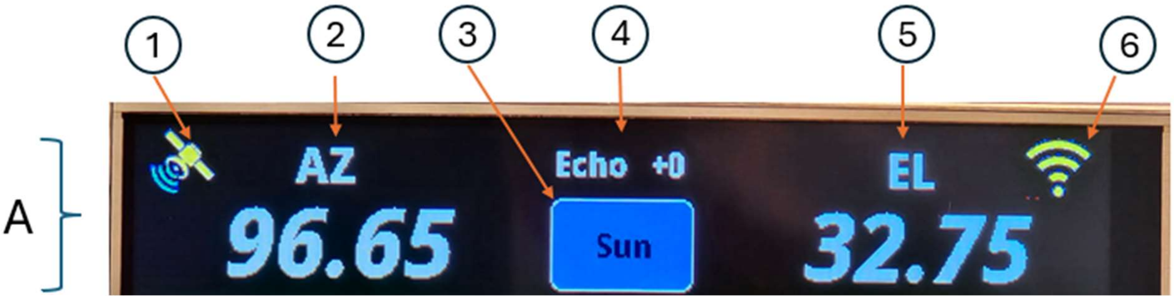
Those bridges are based on the BTS7960B chips and manage up to 34VDC and 40amps, nevertheless the alu dissipator included is too small for currents over 15amps but you can replace for a more convenient colling element depending on your demands. One per movement are needed and they support the PWM coming from the Antrack Pro to manage the soft start and soft stop.

Configuration Screens

The main screen is divided into 4 horizontal sections (A, B, C and D). For a better understanding, we are going to explain the different functions by sections. The sections A, B and C are aligned with respect to the vertical axis of the indication AZ (Azimuth) and EL (Elevation) in such a way that the 3 sections reflect the 2 movements AZ, EL with respect to the position of the target (section A), corrections in the antenna (section B) and absolute position of the antenna (section C). Section C represents the absolute reading of the encoders, this means "where the antenna points"



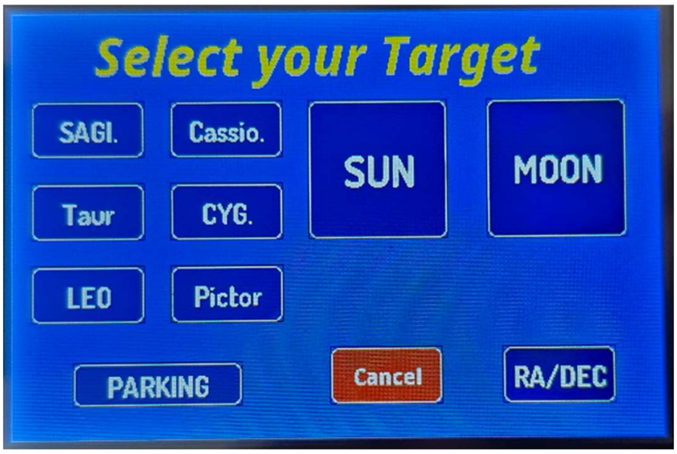
SECTION A (Target, Echoe, GPS and Ethernet indicators)



(1) GPS Status: if the icon is in yellow means a successful lock to at least 3 GPS satellites happens, if red means GPS enable BUT not lock.

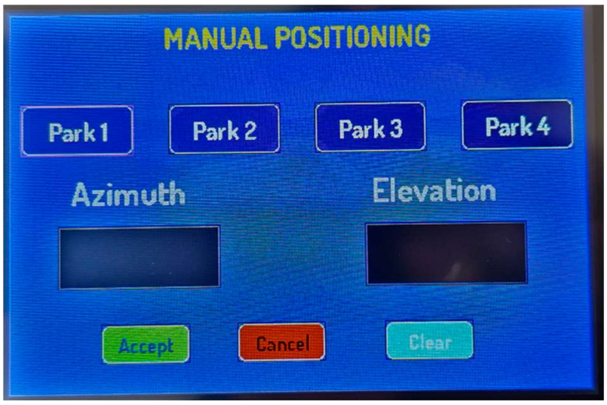
(2) AZ, Azimuth positions of the target, the numbers below represent the AZ position of the target selected in (3)

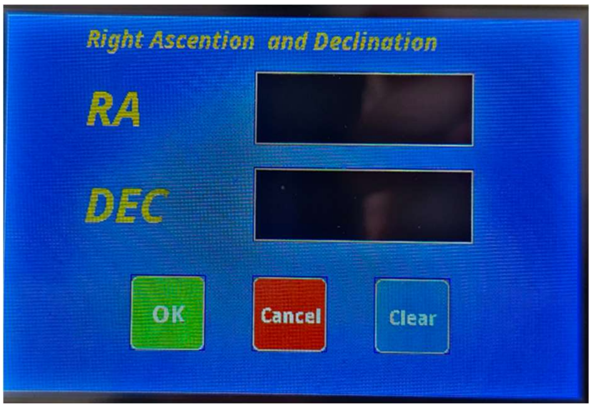
(3) SUN in this case the SUN is selected as a target, if you touch button (3) you will move to the target selection screen:



This window is self-explanatory, the SUN and MOON buttons are larger considering those objectives as the main ones, on the left side we can select the 6 most traditional ones.

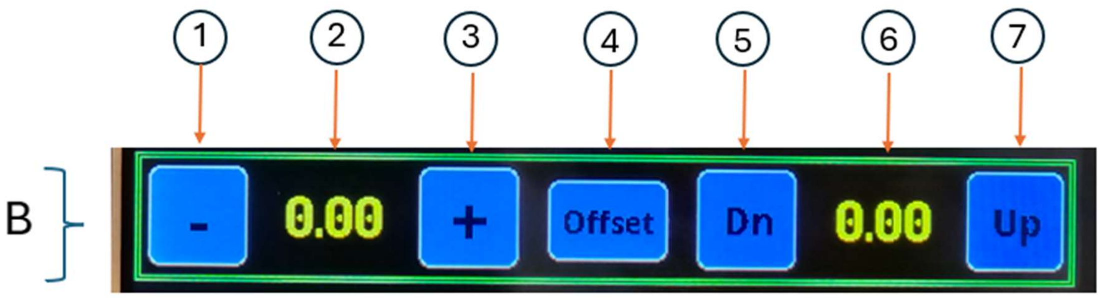
Additionally, the PARKING button moves us to another window where we can enter up to 4 predefined parking positions in AZ and EL. The same for the RA/DEC button, we can point the antenna at a predefined target by entering the data in RA (right ascension) and DEC (declination) format.





(4) Echo, indicate +/- in how many Hz we can listen our Echo, this doppler calculations is based on the band selection and is present only when the target selected is MOON, in the example above we can see Echo + +0 Hz because the target is the SUN
(5) EL, Elevation positions of the target, the numbers below represent the EL position of the target selected in (3)
(6) WiFI, is yellow means a success connection to an Access Point happens (including DHCP IP assignation for the controller) BUT doesn't mean Internet connection, the user must verify the access point access internet properly. If the icon is red means and access point is detected BUT no connection due probably bad password

SECTION B (antenna correction, tilt table and AZ/EL corrections in 0,01 deg step if your encoder support that accuracy.

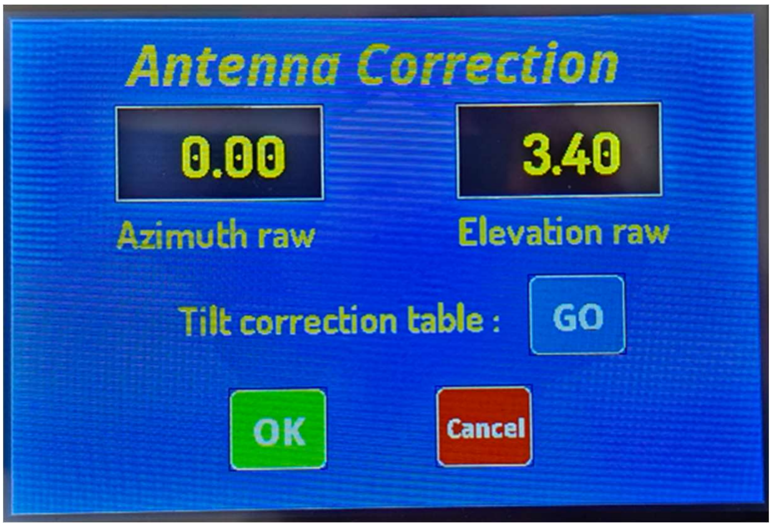


We use section b to introduce corrections in both AZ and EL, these corrections in steps of 0.01 degrees are introduced directly to the antenna, this helps us fine-tune the position of the antenna with respect to the objective to obtain the best signal. For example, when we test our echoes

FOR AZ

(1) – correction for AZ
(2) 0.00 total amount of correction applied to the antenna
(3) + correction for AZ
(4) Offset, (please have a look at the bottom of this section)
(5) Dn, down correction for EL
(6) Up, Up correction for EL

The Offset button (4) : The offset button moves us to a new screen called antenna Correction, here we can enter RAW values for each encoder so that we calibrate its absolute position via software without having to physically move the encoder.



In the example shown we see that we are applying +3.40 degrees of Elevation correction, this means that when we put the antenna at 0 (zero) degrees of elevation without any correction our encoder is reading not 0.00 but -03.40, to compensate for this misalignment we apply the correction " positive" of +03.40 degrees to obtain the required 0.00.
With this help, when installing the antenna and the encoders, we do not need to install the antenna perfectly in AZ and EL, we can do an approximate installation, for example to the geographical North and then introduce the corrections in this window. A good practice is to measure the Sun noise, so we can find how many

degrees of offset we have in each movement, then we will apply this correction in this section and the antenna will be perfectly calibrated for the other celestial objects.

Now from the same last window (Antenna Correction) we have the button GO for the Tilt correction table.

AZ	EL	AZ	EL	AZ	EL	AZ	EL	AZ	EL	AZ	EL
000	0.0	060	0.0	120	0.0	180	0.0	240	0.0	300	0.0
006	0.0	066	0.0	126	0.0	186	0.0	246	0.0	306	0.0
012	0.0	072	0.0	132	0.0	192	0.0	252	0.0	312	0.0
018	0.0	078	0.0	138	0.0	198	0.0	258	0.0	318	0.0
024	0.0	084	0.0	144	0.0	204	0.0	264	0.0	324	0.0
030	0.0	090	0.0	150	0.0	210	0.0	270	0.0	330	0.0
036	0.0	096	0.0	156	0.0	216	0.0	276	0.0	336	0.0
042	0.0	102	0.0	162	0.0	222	0.0	282	0.0	342	0.0
048	0.0	108	0.0	168	0.0	228	0.0	288	0.0	348	0.0
054	0.0	114	0.0	174	0.0	234	0.0	296	0.0	360	0.0

This table is designed to enter up to 60 correction positions in EL every 6 degrees of AZ. Installing an antenna with a perfect verticality is almost impossible, the very weight of the antenna, the size, the weight of the feed a few meters outside the central axis make the verticality suffer an error, this is particularly critical at frequencies above 5Ghz. With this table we can plot the 360 degrees by introducing correction factors every 6 degrees. The corrections tend to be minimal but have adverse results at high frequencies if they are not taken into consideration.

Initial Program Load

- Purchase of an AnTrack-Pro board set will include an ESP-32 developement board with the latest firmware programmed. Future updates to the firmware can be accomplished using the builtin OTA update function. (see: AnTrack-Pro Software Updating section)

Remote Console

The AnTrack-Pro was originally designed to operate as a standalone antenna controller and can still operate in this mode. Once the hardware was designed and software was progressing the idea of a web page capable of supporting a sub-set of the AnTrack-Pro functions was proposed.

The ESP32 is an IOT device with built-in WiFi and it seemed a simple task to support this feature. (..programmers comment.. I have an aversion to embedded web design and it was <u>not simple</u> to implement) Eventually a functioning web interface was created.

The idea of extending the web interface to an Ethernet device was floated. The ESP32 does support Ethernet but only a limited range of devices. Unfortunately almost every GPIO was in use and these multi-pin devices could not be used. Eventually a W5500 Ethernet module with a SPI interface was chosen. This brought with it a number of headaches including poor support in the ESP32 core software.

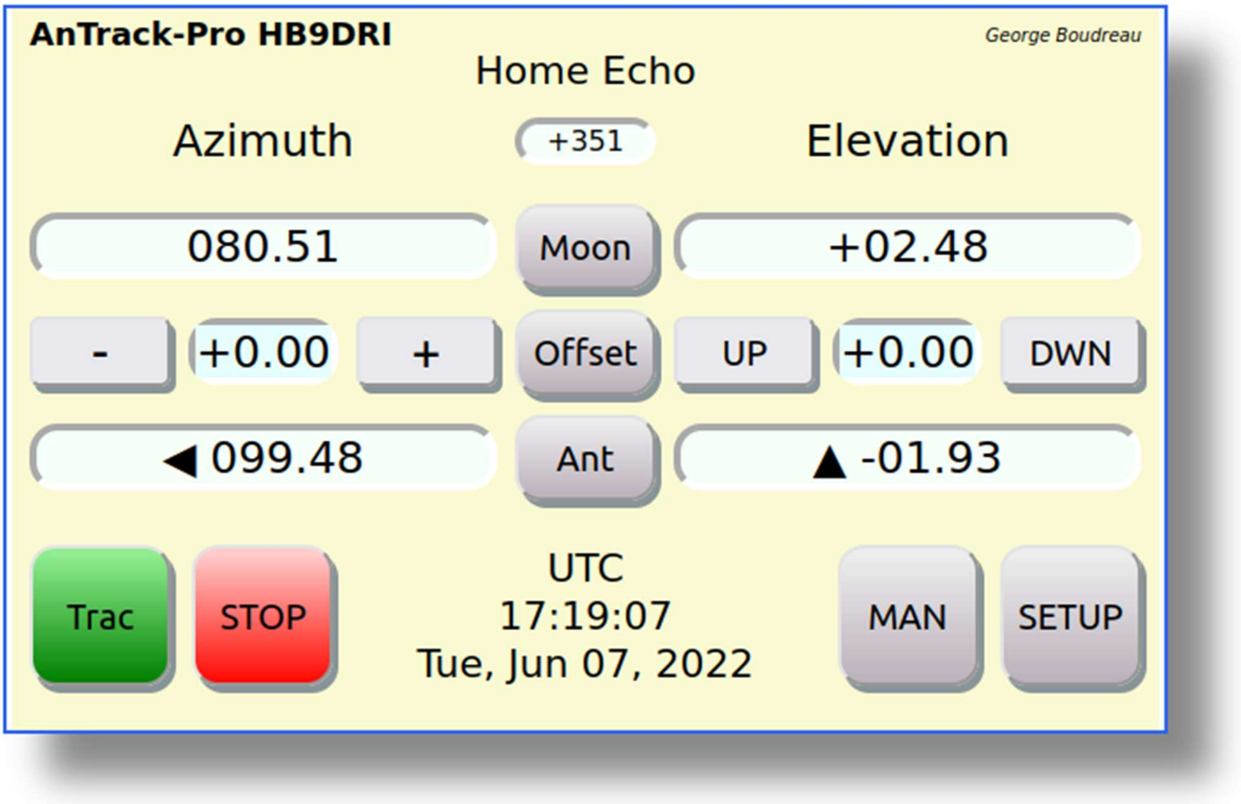
With executable space running low it was decided to rewrite the WiFi web server to be as generic as possible to allow for functions common between the WiFi and Ethernet servers to be extracted. The final code is compact and allows the same HTML page to be used for both WiFi and Ethernet.

--NOTE-- Both the WiFi and Ethernet web pages are hard coded in the compiled binary to eliminate the need for a SD card driver and the accompanying increase is code size.

::Operation::

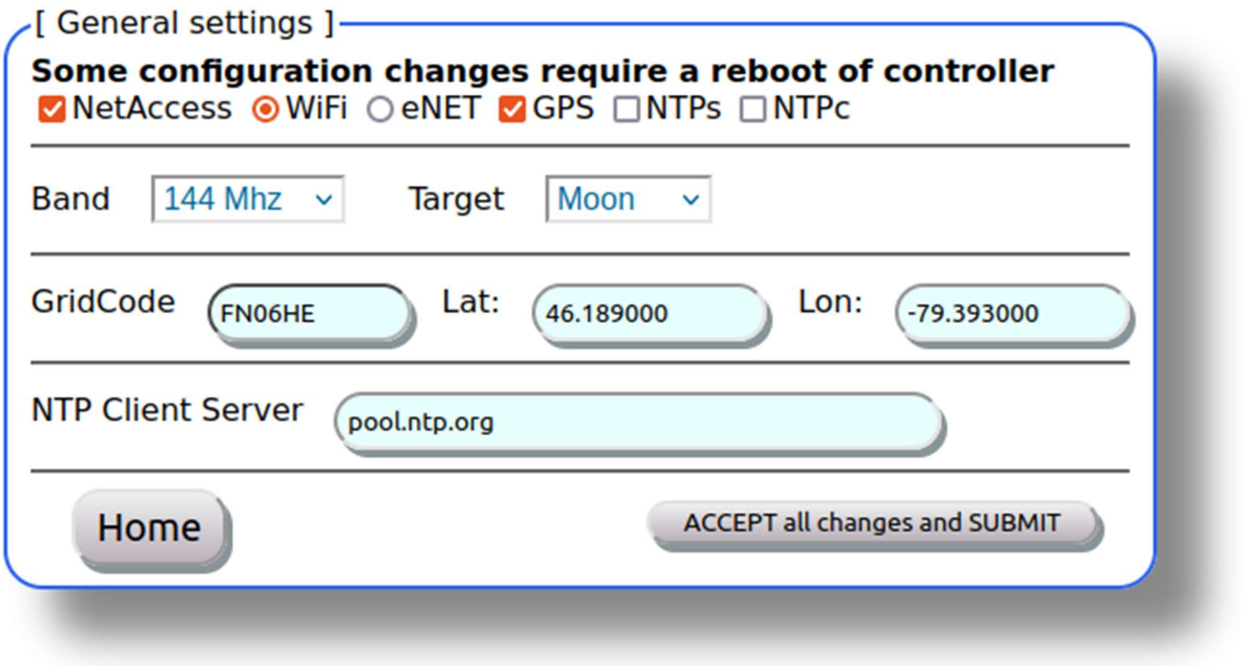
:::NOTE:::The remote console only supports a subset of the AnTrack-Pro configuration menus.

The home web page mirrors the touch screen display as closely as possible. Variable display elements are updated once per second with a call to the AnTrack-Pro server.



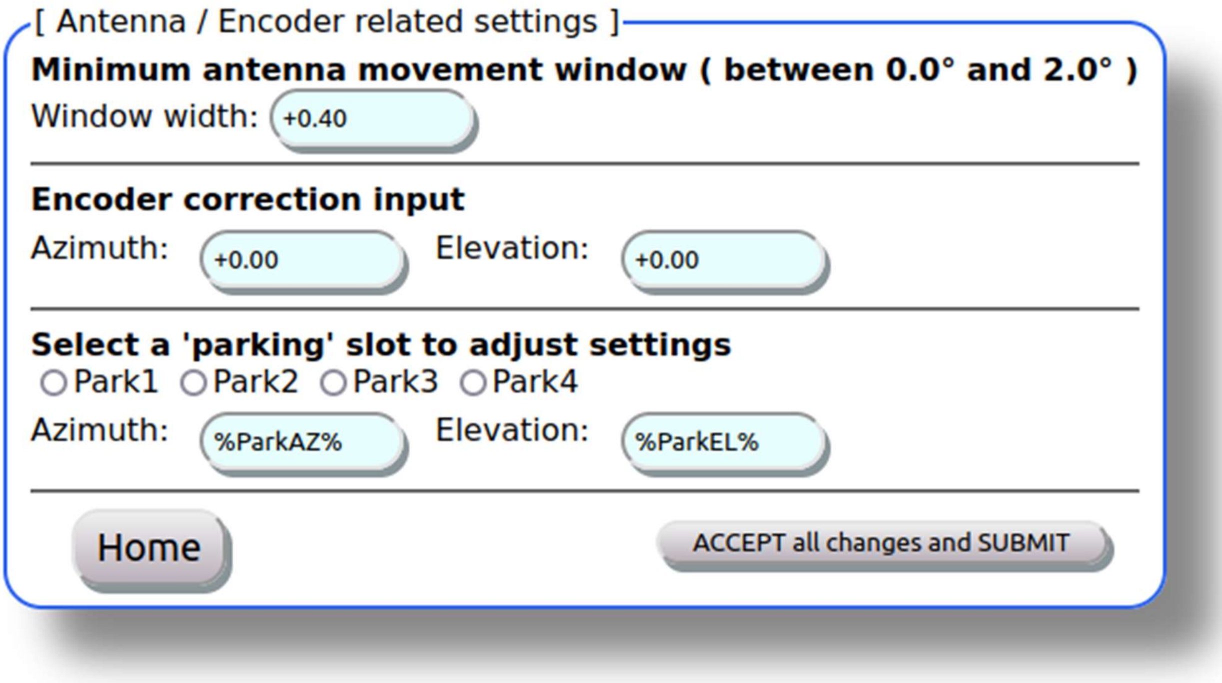
General Configuration

- Information Interfaces
- Band / Target
- Antenna location
- NTP server


[General settings]
Some configuration changes require a reboot of controller
NetAccess
WiFi
eNET
GPS
NTPs
NTPc
Band
144 Mhz
Target
Moon
GridCode
FN06HE
Lat:
46.189000
Lon:
-79.393000
NTP Client Server
pool.ntp.org
Home
ACCEPT all changes and SUBMIT

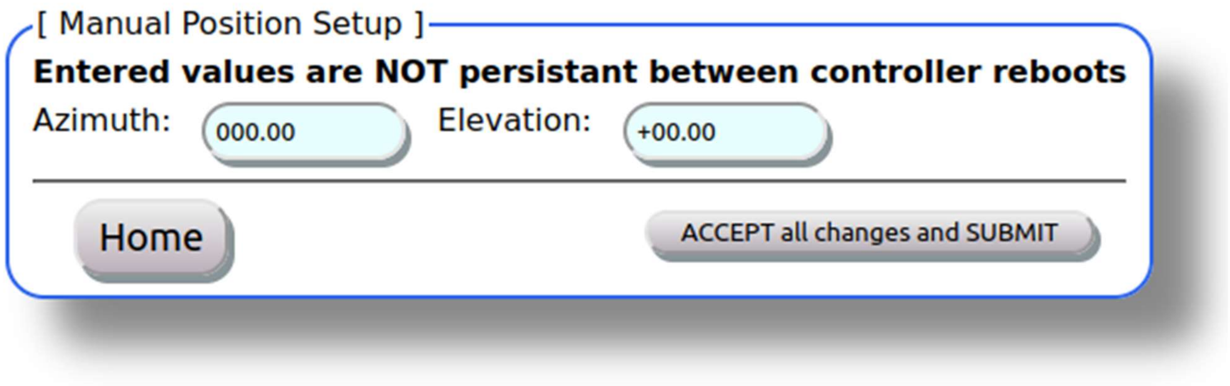
Antenna / Encoder related setting

- Minimum antenna movement window
- Encoder Correction Input
- Parking slot configuration



Manual Antenna Movement

-



DSN Tracking

Although not currently implemented DSN tracking should be possible. This function will only be possible if the AnTrack has an internet connection.

Preliminary JPL ephemeris collection experiments have been completed and an AnTrack webpage entry has been created but not integrated into the code base. Available memory limitations may be an issue.

A ‘dummy’ webpage is shown below.

AnTrack-Pro Software Updating

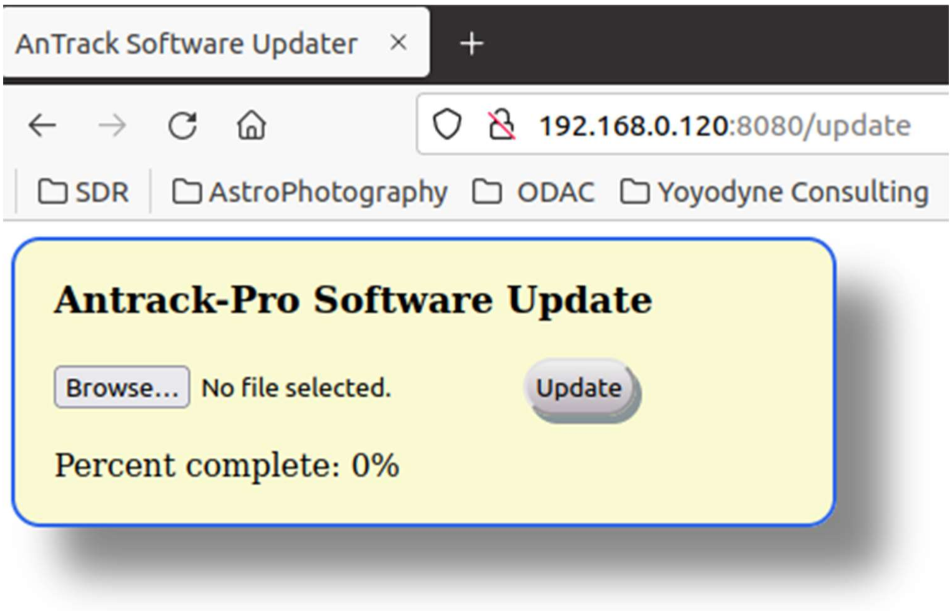
No piece of software can be guaranteed defect free and the software driving the AnTrack-Pro is no exception. If software errors of discovered or feature improvements, such as DSN, are added the AnTrack-Pro provides a painless method for updating the embedded software.

The ESP32 hardware supports direct software upload via the UART port. However, this requires an external software package(s) to be installed as well as access to the UART port and a compatible USB-to-serial device.

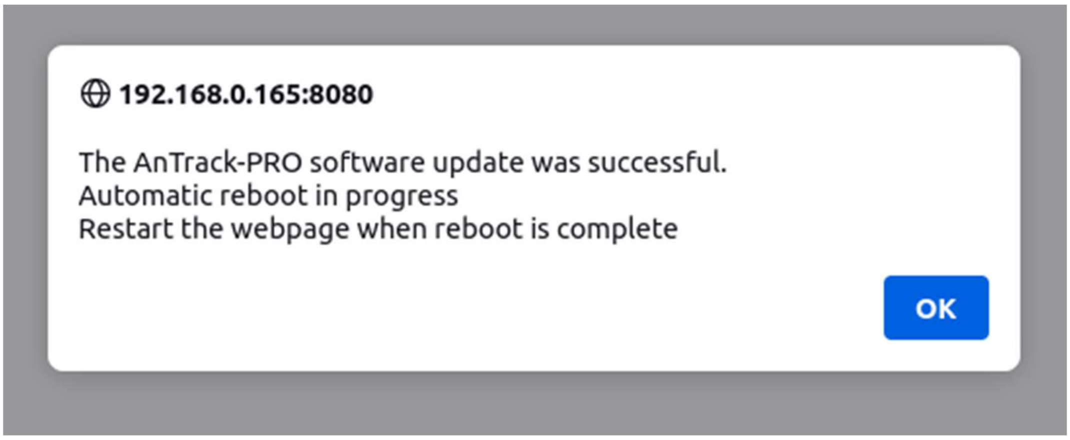
Espressif provides a small collection of routines to allow OTA updating, direct writing to the program flash. The developer must first create a flash partition to allow for the new code to be written to. Once the new code passes the download validation a boot pointer is adjusted to point to the new code. The new code will run on the subsequent reboots.

For OTA or “Over The Air” software updates of the AnTrack-Pro a small ‘update server’ was embedded in the main control software. This uses the Espressif update routines and allows the new code to be directly downloaded into the ESP32 module. No external software packages or hardware dongles are required. Unfortunately the support for OTA updates comes with a high cost and that is the reduction of available program memory. Program memory is allocated at 50% for the current code and 50% for the updates.

New code, if necessary, will be retrieved from the LinkRF.ch site and placed in an accessible directory on your PC. :::NOTE::: This PC MUST be on the same router as the AnTrack-Pro. The update server is accessed by appending **:8080** to main AnTrack-Pro remote console page (see image below). You then navigate to the directory where the new AnTrack-Pro software was downloaded to, select the new bin and press the ‘Update’ button. The download progress is given as an increasing percentage


AnTrack Software Updater
192.168.0.120:8080/update
SDR
AstroPhotography
ODAC
Yoyodyne Consulting
Antrack-Pro Software Update
Browse...
No file selected.
Update
Percent complete: 0%

A small popup window will let you know whether the download was successful.



If the download was successful the AnTrack-Pro will automatically reboot with the new code which takes approximately 60 seconds. Once a successful reboot has been accomplished you must re-entry the URL of the AnTrack-Pro remote console. :::INFO::: Updating the software will not change any configuration parameters.

Software Packages

AnTrack-PRO

Arduino IDE 2.3.2

Espressif board library 2.0.14

GUIslice + Builder 0.17.0

TFT_eSPI 2.5.0

RTClib 2.0.3

Ethernet 2.0.0

NTPserver (custom)

NTPclient (custom)

Astral calculations (custom)

WebServers Ethernet & WiFi (custom)

EncoderHUB

Arduino IDE 2.3.2

Arduino AVR Boards 1.8.6

APPENDIX

Encoder Resolution calculations

Most encoders output a binary value. The angle value of the LSB of an encoder is calculated as.
encoder value (360.0 / 2^Bits)
The encoder cannot resolve an angle less than this angle and movement is detected in increments of this value.

- HH12, MAB25, ETS25 etc. resolution is 12bits = 4096
- encoder value of 1 gives a deg value of: 1 * (360.0/4096) = 0.0879 deg
- encoder value of 2 gives a deg value of: 2 * (360.0/4096) = 0.1757 deg

- Posital Inclinometer resolution is 13bits = 8192
- encoder value of 1 gives a deg value of:(1 * (360.0/8192) = 0.0439 deg
- encoder value of 2 gives a deg value of: (2 * (360.0/8192) = 0.0878 deg

A theoretical 16bit encoder = 65535
- encoder value of 1 gives a deg value of: 1 * (360.0/65535) = 0.00549 deg
- encoder value of 2 gives a deg value of: 2 * (360.0/65535) = 0.01098 deg

  NOTE:: Use of a 16bit encoder would require extremely tight mechanical movement as well as a very rigid antenna.

Display resolutions of 0.01 would only be valid for encoders with 15 bits or greater resolution. Displaying of resolution or movements less that the LSB of an encoder is for visualization only.

Info:: Internally all astal calculations are done using double precision floating point to maintain precision.

AnTrack Antenna Movement Window

The variable 'AntennaMovementResolution' defines the size of the hysteresis or dead-zone for movement of the antenna. The window created is +/- around the current calculated astral target value [T]

```
      Hs      Ls
  L1 ----[L2 --- T --- L2]
```

L2 = AntennaMovementResolution + 1-bitDegree : Target Window. Note: a window of '0' still has a 1-bitDegree hysteresis window.

L1 = Target + 3.0deg is hardwired in code.

```
Hs Motors running at high speed.
Ls Motors running at low speed.
```

Only at startup or target change will the antenna position exceed 3.0deg and the motors will run at full speed. During normal operation the motors ramp up to half speed.

- antenna position is < L1 - run the motors at high or full speed until position is >= L2
- antenna position is >= L2 - ramp motor speed to low or tracking speed.
- antenna position is = T+1-bitDegree - ramp to stop.

The antenna does not move until delta between actual and Target (T) exceeds L2, the hysteresis limit.

Encoder Hub Data Format

The EncoderHub reads the Elevation/Azimuth encoders, formats the data and sends it to the resulting packet to the AnTrack over a RS485 serial cable.

A data frame consists of:

- FRAME_START
- Bit resolution of each encoder packed as nibbles.
- binary value read from each encoder padded to 16 bits
- FRAME_END

Note:: Encoder bit resolution is limited to values between 9 and 16 bits when using the EncoderHub.

With binary data having a range between 0x00 and 0xFF there is no way to differentiate directly between the reserved characters FRAME_START / FRAME_END and valid data.

To encode the frame a modified version of COBS, Consistent Overhead Byte Stuffing, was implemented to handle the reserved character. A full explanations of COBS algorithm can be found at : https://en.wikipedia.org/wiki/Consistent_Overhead_Byte_Stuffing

The customized COBS implementation changes are:

- Limiting the possible packed length to <254 bytes.
- Input frame is modified in place to save coding.
- Encoding starts at the end of the frame and steps backwards. This reduces some coding complexities.

The resulting data frame sent to the AnTrack-Pro has the following format:

```
DATA[7] = { FRAME_START,            // non-zero value
            EncBITS,                //AZ_bits << 4, EL_bits &0xFF
            EL_hi_byte, EL_lo_byte,
            AZ_hi_byte, AZ_lo_byte,
            FRAME_END };            // 0x00
```

The AnTrack-Pro accepts this frame and searches for the frame_end delimiter character, processes the incoming characters until it detects another frame delimiter. If there are the sufficient bytes it them decodes the frame.

Frame decoding applies the COBS algorithm to restore the received frame to its original format.

The encoder data is then converted to decimal degrees using the encoder bit resolution and raw binary value for use by the astral calculations.
If your encoder does not output an absolute value you can write your own interface to the AnTrack-Pro as long as you support the interface packet. For example pulse encoders could be converted to an absolute value, packaged and sent to the Antrack-Pro. You would be responsible for maintaining the calibration of your system but it would be possible for you to create a custom interface using a small MCU with an RS485 interface.

For more information on writing your own encoder interface contact HB9DRI who will put you in contact with the programmer.