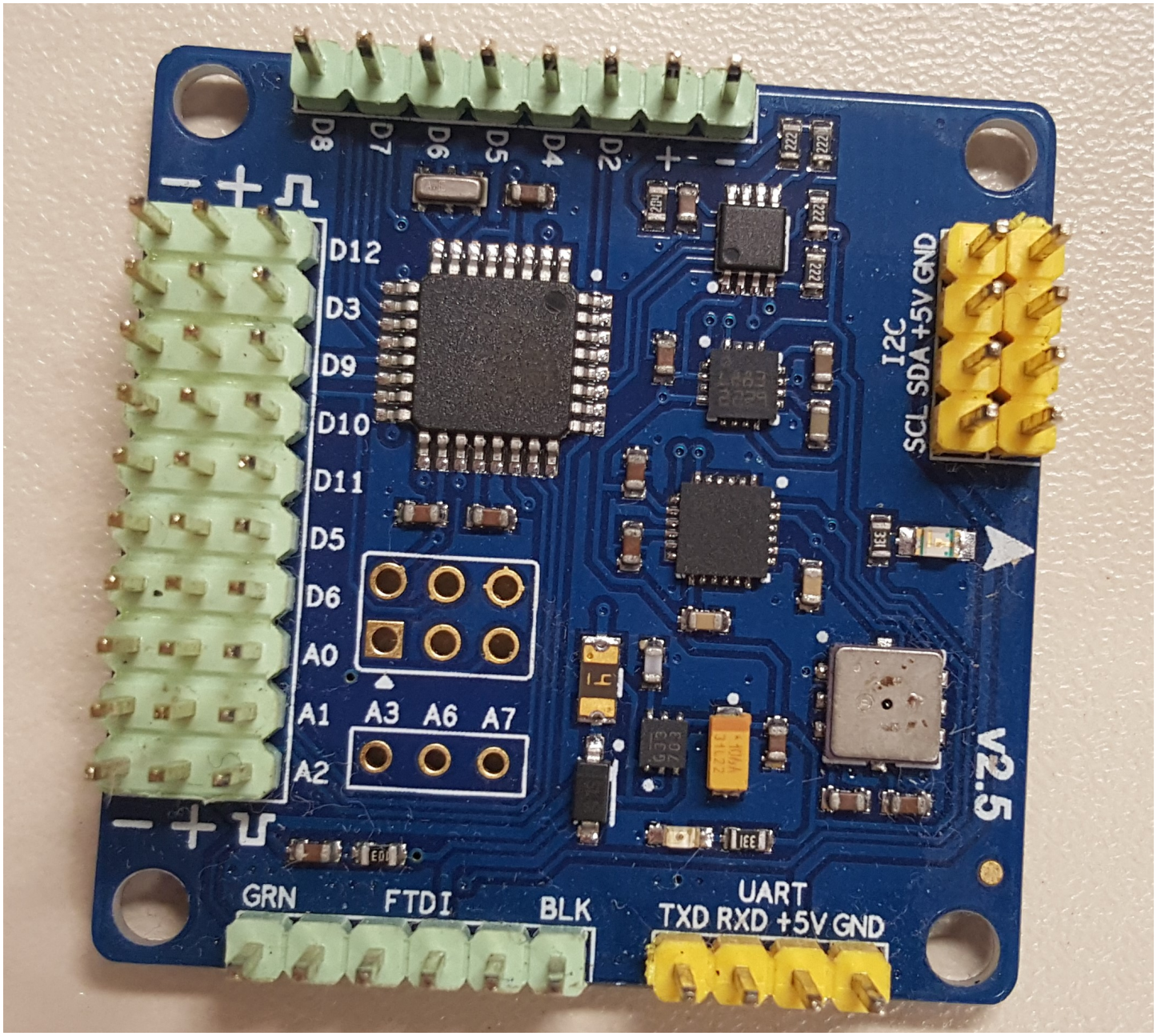
Setting-up quadcopters with the MultiWii v2.5 flight control board

This guide shows how to build and set-up a quadcopter with the MultiWii v2.5 board.

Table of contents

- Setting-up quadcopters with the MultiWii v2.5 flight control board
  - Our Parts
  - The flight control board
  - Flight controller configuration
    - The basic setup
    - Calibration
  - Connecting parts
    - Receiver
    - ESCs
    - Power
    - Connections summary
  - Remote controller and flight modes
    - Configuring the remote controller
    - Flight modes
  - PID tuning
  - Conclusion
- Files
- References



Our Parts

- 1 x Quadcopter frame
- 4 x Electronic speed controllers - SK-30A (with SimonK firmware)
- 4 x Brushless motors - A2830-11
- 1 x Remote controller - Turnigy 9x 9Ch Transmitter
- 1 x Transmission module - Turnigy RF9X-V2
- 1 x Channel receiver - Turnigy 9X8C-V2 8
- 1 x Flight control board - CRIUS MultiWii v2.5
- 1 x 14,8V Li-Po battery - Zippy 20C Series 4000
- 1 x FTDI basic breakout (needed only to connect the board to a PC)

It's not mandatory to use the same models and brands as ours, you can search for different options, but make sure that the components are compatible among themselves and with the weight of the structure.

The flight control board

There are plenty a lot of boards you can use to control your UAV, in our case we are using the CRIUS MultiWii v2.5, that features an ATMEL Mega 328P microcontroller, and in-built accelerometer, gyroscope, barometer and magnetometer. The board name comes from the MultiWii open source software project,

aimed to the control of RC controlled multi rotor flying platforms. Here you can find more information about the project, and download the codes and software that we will use in the board configuration.

Flight controller configuration

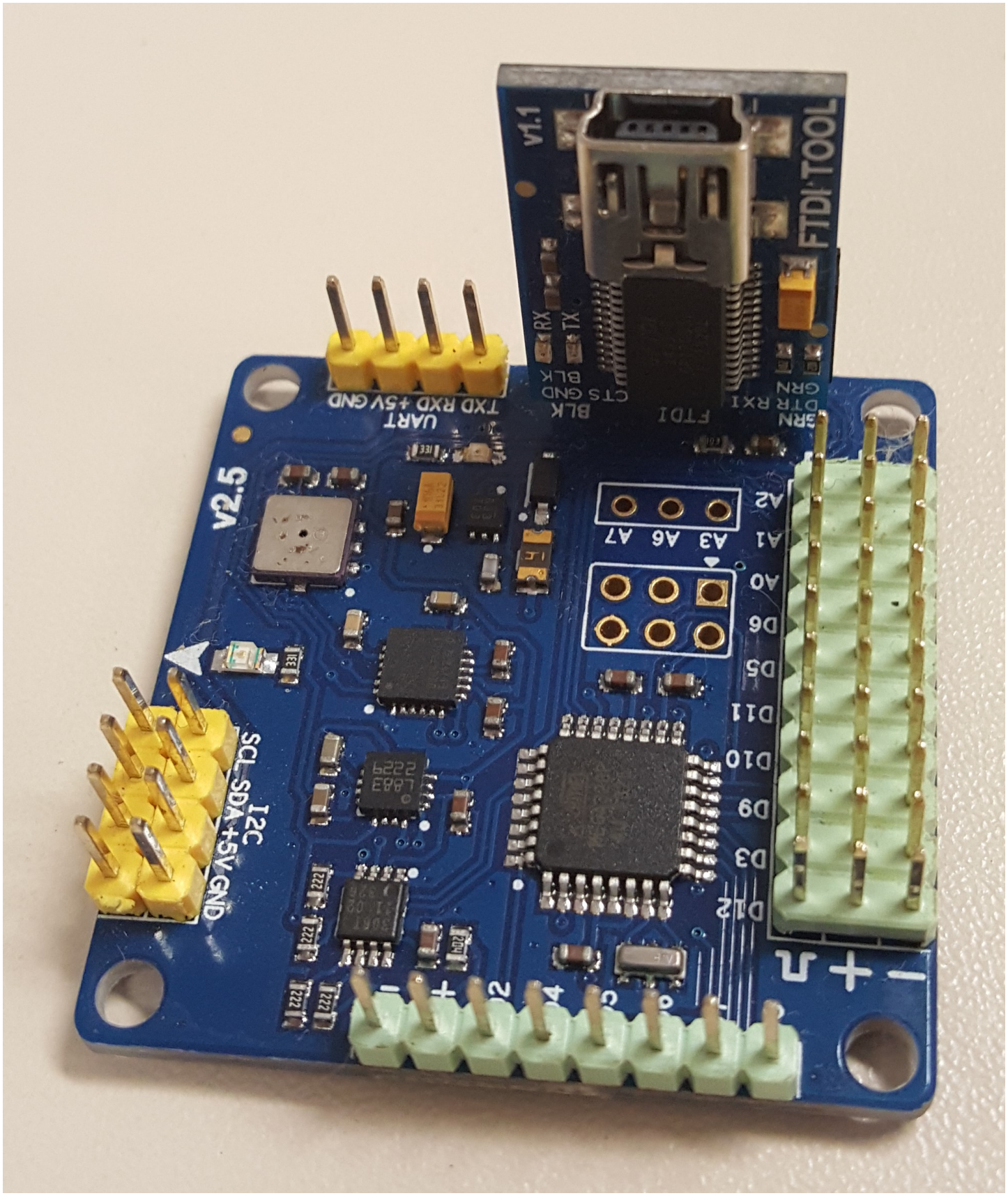
For this step we'll need the MultiWii codes and GUI (that can be found here). We're using the 2.4 version, that is the last available at this time. To edit and flash the software to our board we will use the Arduino Software. Under the "MultiWii 2.4\Multiwii" folder, that were downloaded before, you'll find the MultiWii.ino file, use it to open the code files on the arduino software.

The basic setup

Open the codes on the arduino software and look for the config.h tab, there will be the configurable parameters of the software that will be uploaded to the board. In this file we'll just worry with the first section of the configurable parameters. To select an option you just have to uncomment its line. These are the options that we might have to change:

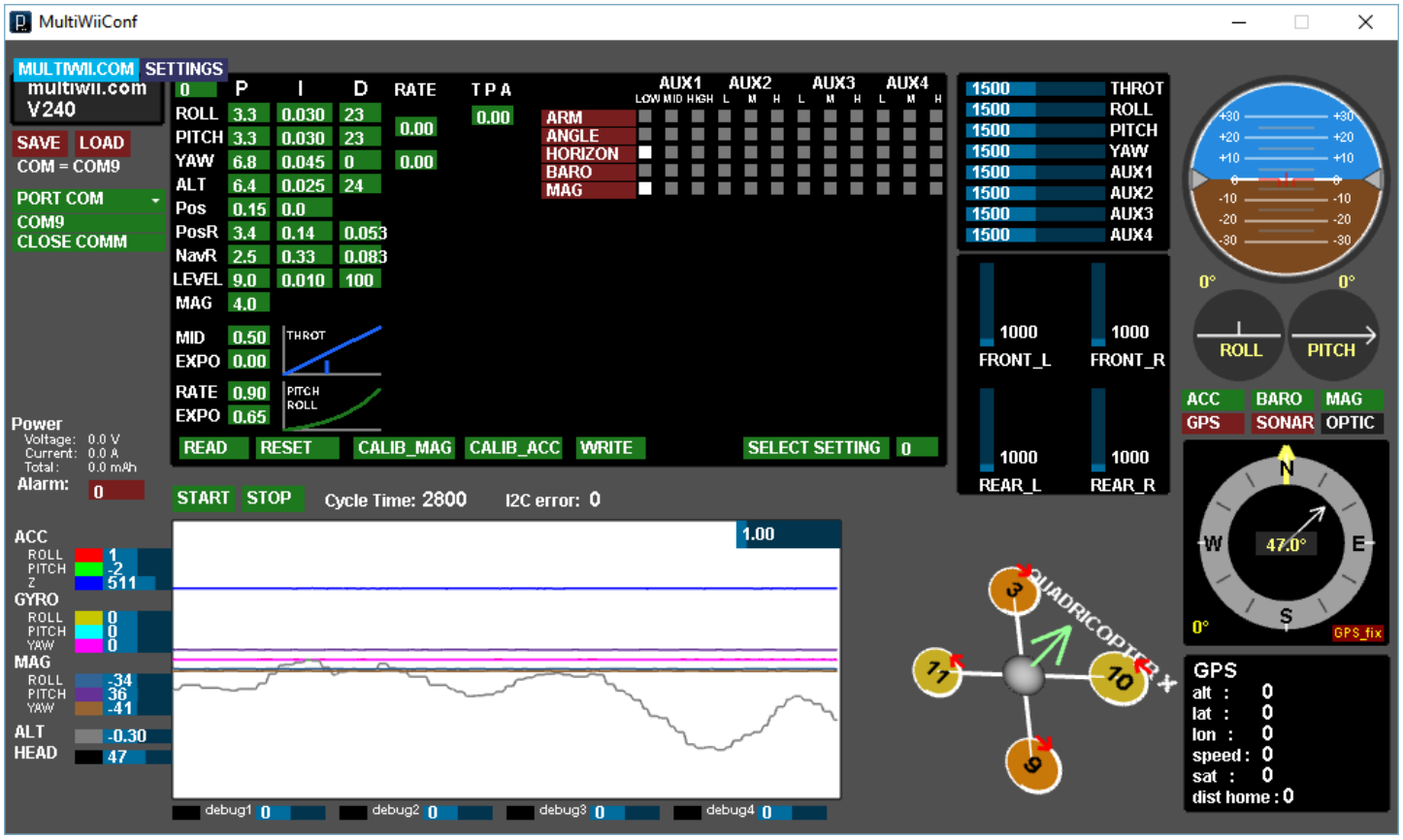
- The type of the multicopter: here you can define which type of multicopter you wish to control with the board. In our case it's a quadcopter, and we opted to use the X configuration. To select it you just need to uncomment the line containing "#define QUADX".
- Motor minthrottle: this information sets the minimum throttle command sent to the ESC. If you have ESCs with the SimonK firmware you should use the special ESC option "#define MINTHROTTLE 1064 // special ESC (simonk)", otherwise you can choose the one that fits your ESCs or keep the default option.
- Boards and sensor definitions: the choice here will depend on your control setup. If you have a flight control board like us, you'll probably find it on the list, or a similar one, and if you're using independent sensors, you must select them too. The option that matches with our flight controller is the "#define CRIUS_SE_v2_0 // Crius MultiWii SE 2.0 with MPU6050, HMC5883 and BMP085".

For all the other options we'll keep the default values, but you can change them according to your needs. Now you can go into the Tools menu, on the arduino software, to select a compatible board and processor. For our MultiWii we are using the "Arduino Pro or Pro Mini" with the "ATmega328 (5V, 16MHz)". To connect the MultiWii to the computer we'll use the FTDI Tool, look at the BLK and GRN indications on both boards to connect it right, as in the image. Once you've done that use the "Verify" button to compile the sketch, and if everything is ok, use the upload button to upload it to the board.



Calibration

Once we've uploaded the software to the board, we are able to use the MultiWiiConf GUI, downloaded with the codes. Under the MultiWiiConf folder find your OS folder, and run the MultiWiiConf ("MultiWi 2.4\MultiWiiConf\YourOS\MultiWiiConf").
With the MultiWiiConf GUI open, select the COM that your board is connected (you can find it in the device manager), then click on READ, and on START after that. All information about the board is displayed on the GUI, like in the following image.



Now you must calibrate some of the board's sensors.

- Accelerometer:
  - Put your board on a flat surface, and then click on the CALIB_ACC button and don't move the board till the blue led goes off.
- Compass:
  - Click on the CALIB_MAG button and for the next 30 seconds rotate the board around all the axes.

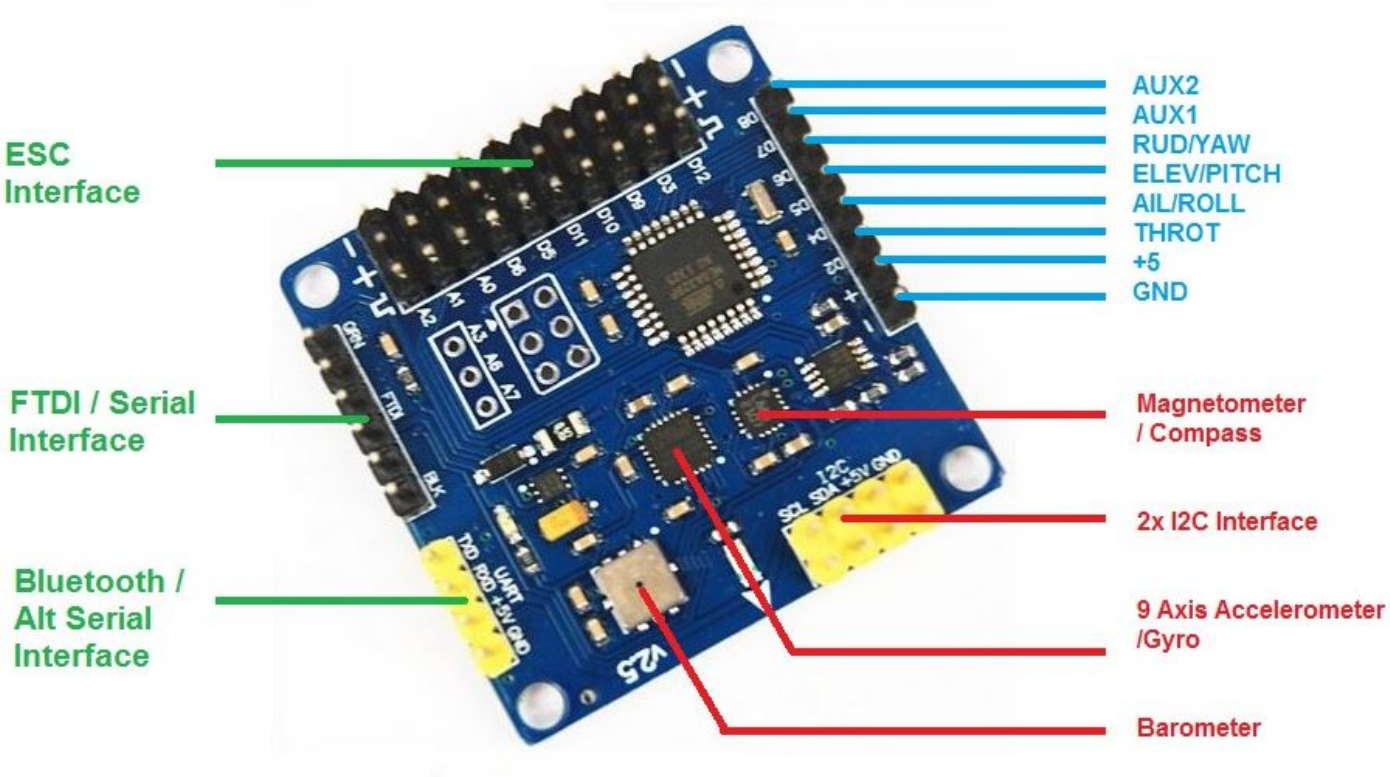
Later you can get back here to set the flight modes according to your will.

Connecting parts

You can start by attaching the motors and the ESCs to the frame, as shown in the image.

Receiver

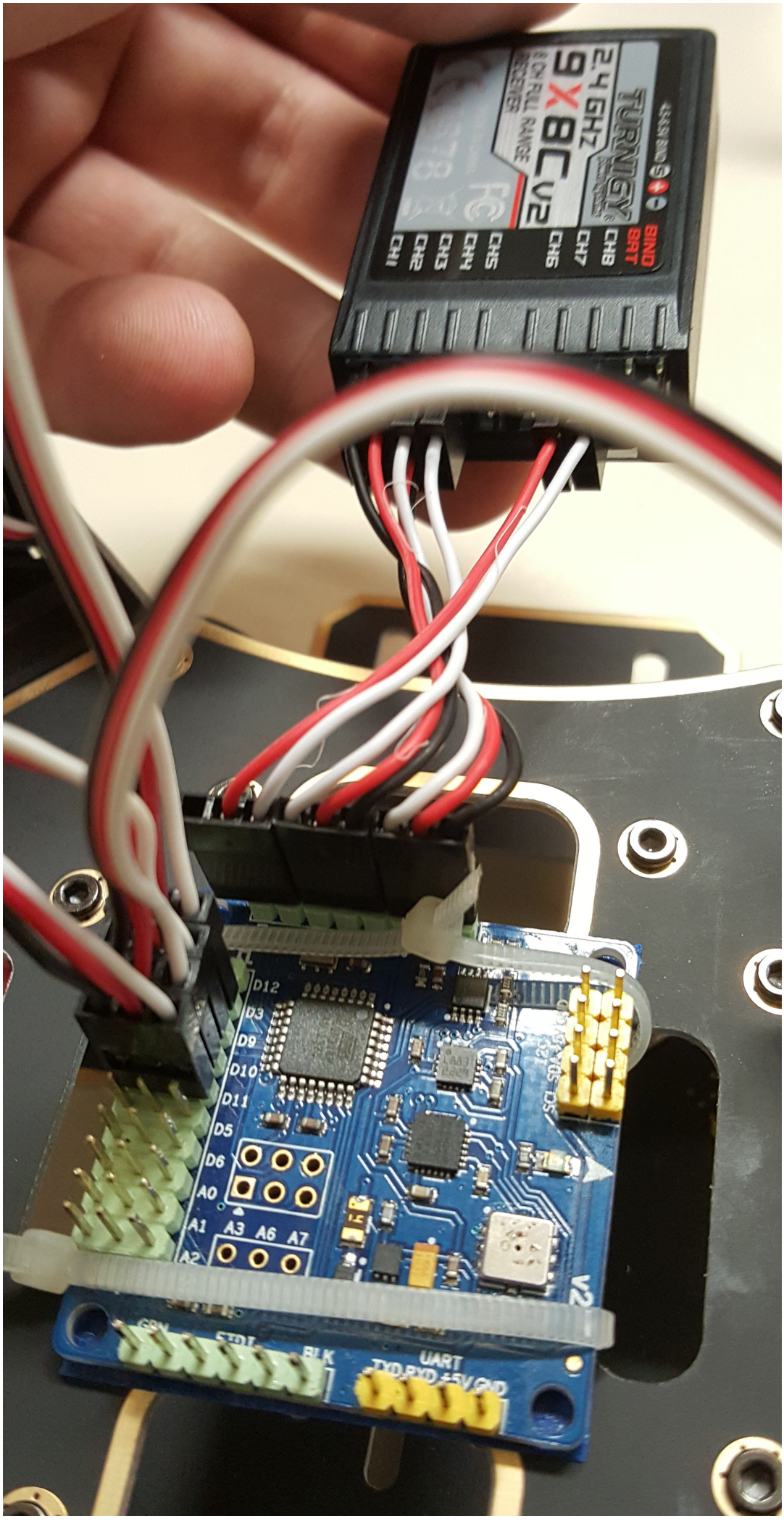
The next thing we'll connect is the receiver to the board. Take a look at the pins indicated with blue lines:

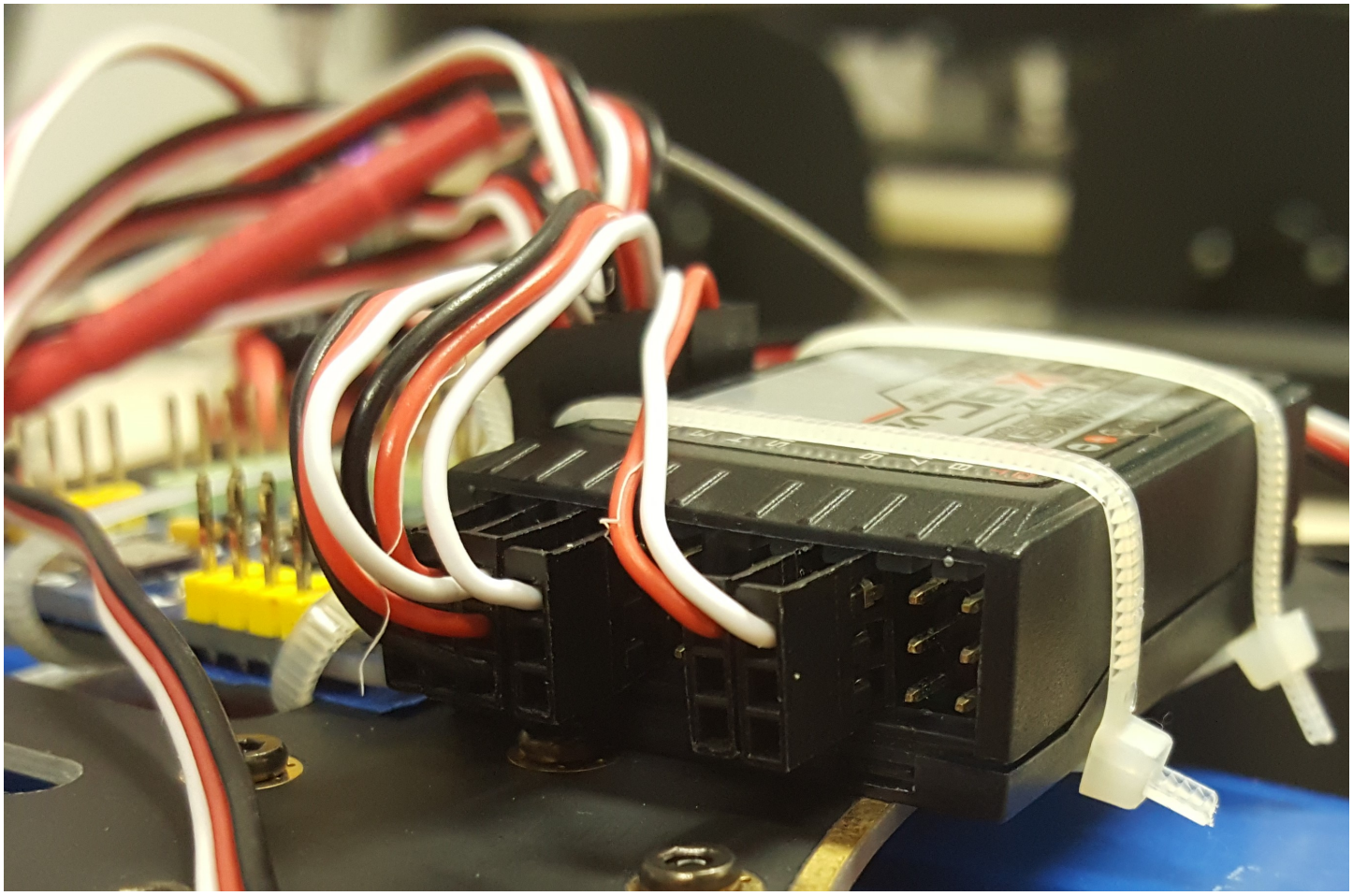


It shows where to connect each channel of the receiver. In our case, these are the receiver channels functions:

- CH1 - Aileron
- CH2 - Elevator
- CH3 - Throttle
- CH4 - Rudder
- CH5 - Gear
- CH6 - Auxiliar 1
- CH7 - Auxiliar 2
- CH8 - Auxiliar 3 (we used this with the board's auxiliar 1)

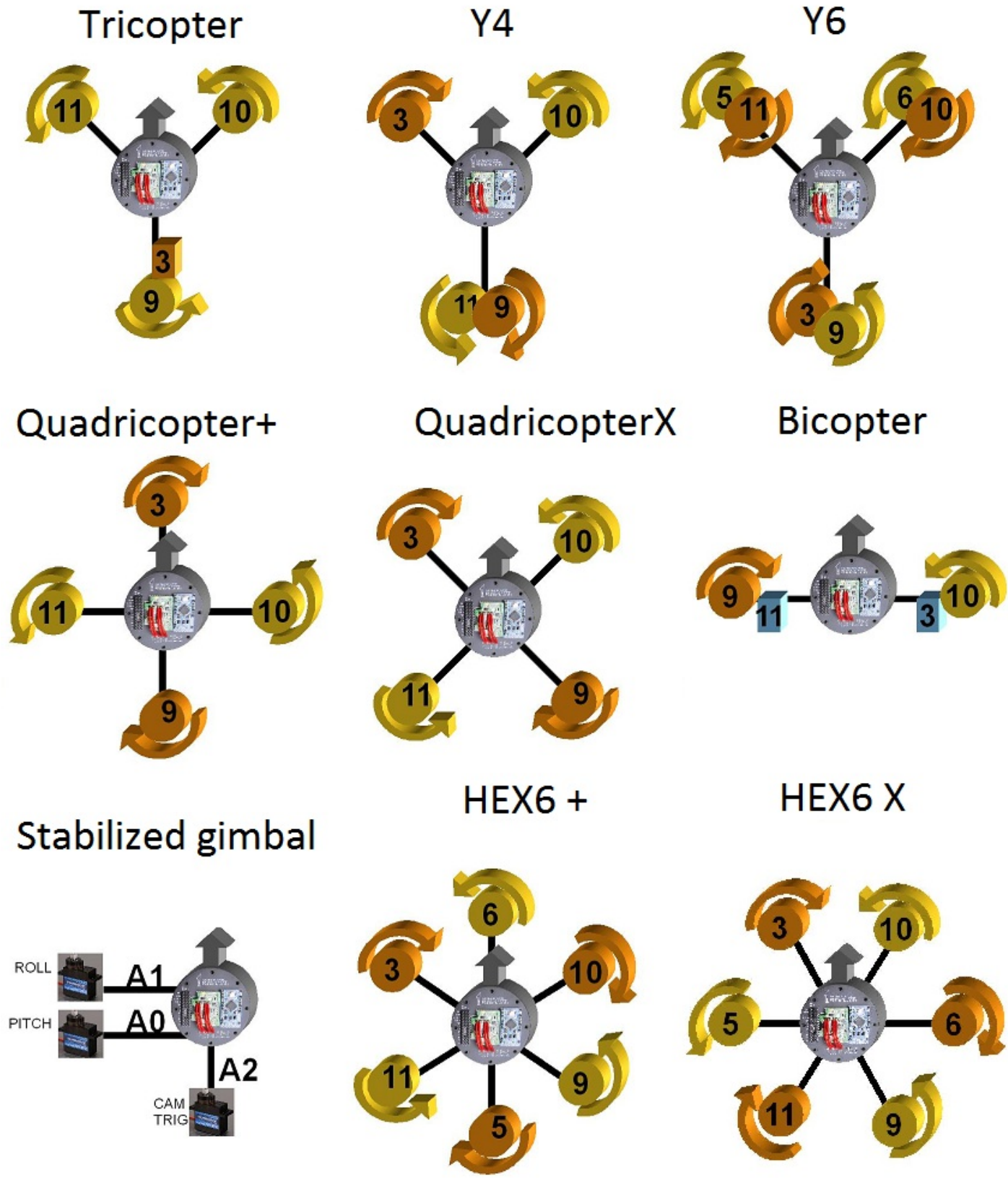
So you can connect each pin to its corresponding channel. In our receiver the second and third lines of pins are used for the energy supply, the second line is common 5V, connect it to de "+" pin on the board, and the third line is the common ground, corresponding to the "-" pin on the MultiWii.


2.4 GHZ
9X 8C V2
8CH FULL RANGE
FUTABA
BIND
BAT
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
D12
D3
D9
D10
D11
D5
D6
A0
A1
A2
A3
A6
A7
V2
GRN
FTDI
BLK
UART
TXD RXD +5V GND



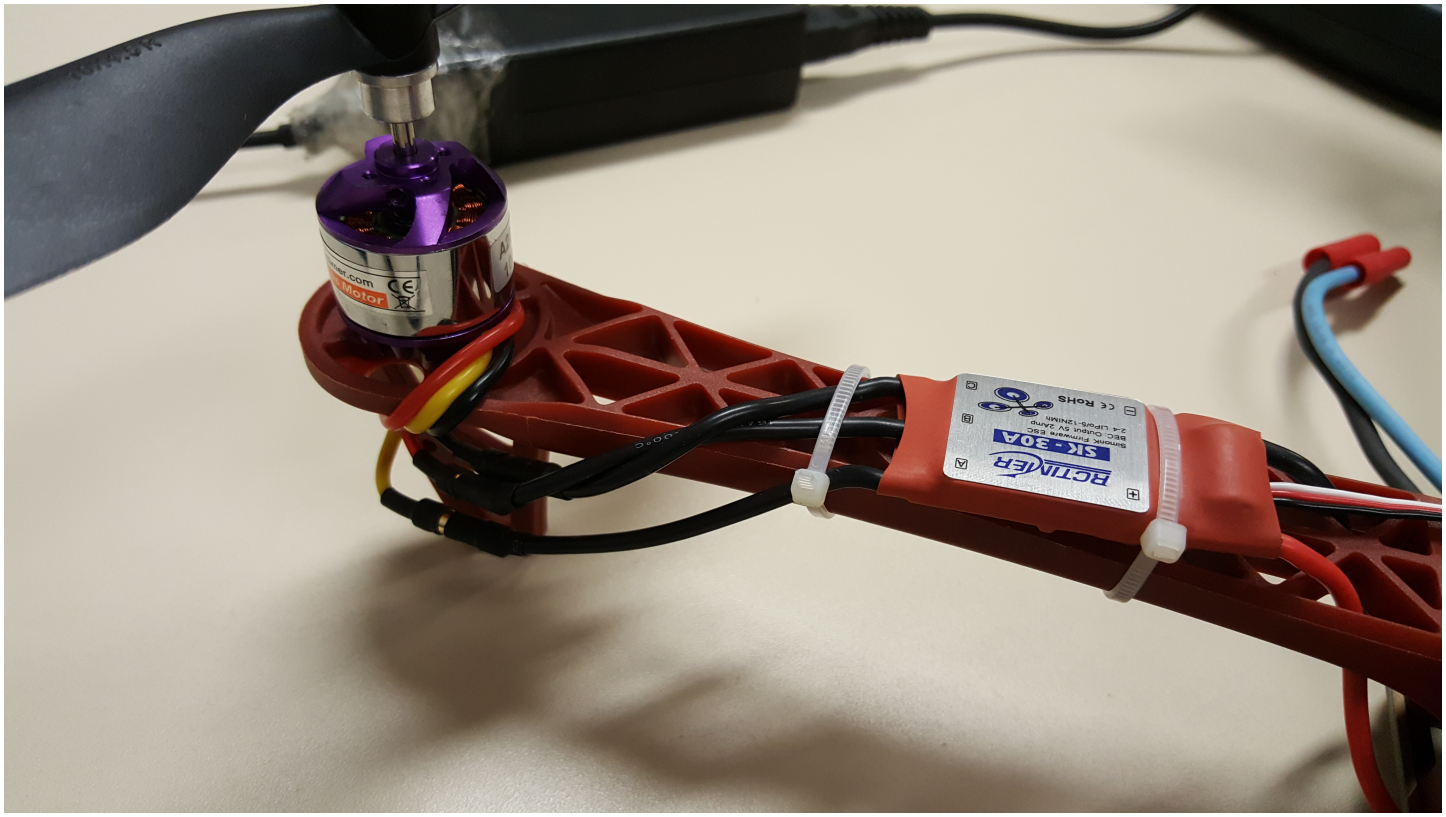
ESCs

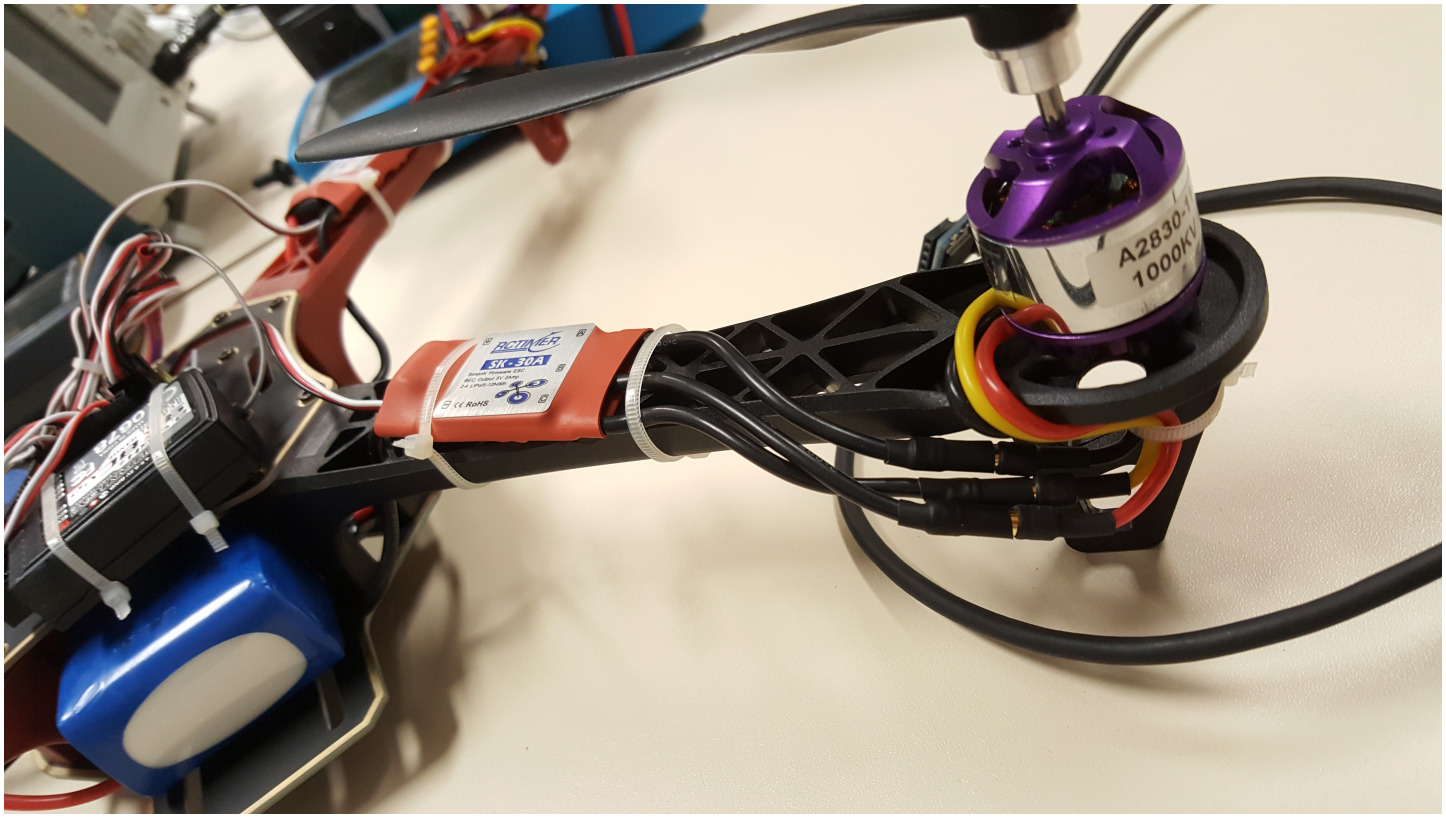
The ESCs are the next thing we'll connect to the board. How we'll do it will depend on the multirotor type. The following image show how to connect each ESC according to the type.



We are using the quad X configuration for our multirotor, so we just have to look at the configuration type image to see which ESC we must connect to each pin. For an example, if the motor is marked with 10 in the image, connect the corresponding ESC cable to the D10 pins on the MultiWii. Pay attention to the wires order in the pins, in our ESCs the black wires are the ground, the red ones are the Vcc, and the whites are for the square wave.

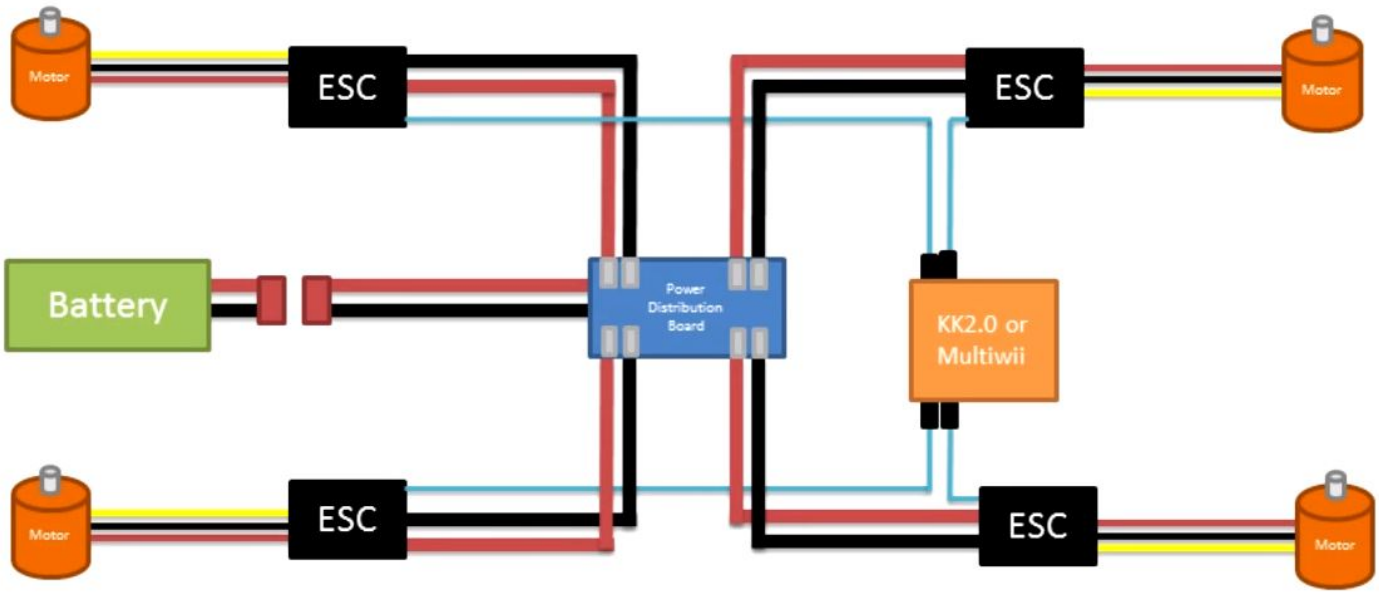
The configuration type image shows too the direction of each motor rotation, this tells you how to connect the motors to the ESCs, to change the rotation direction you must reverse two connections of the motor.



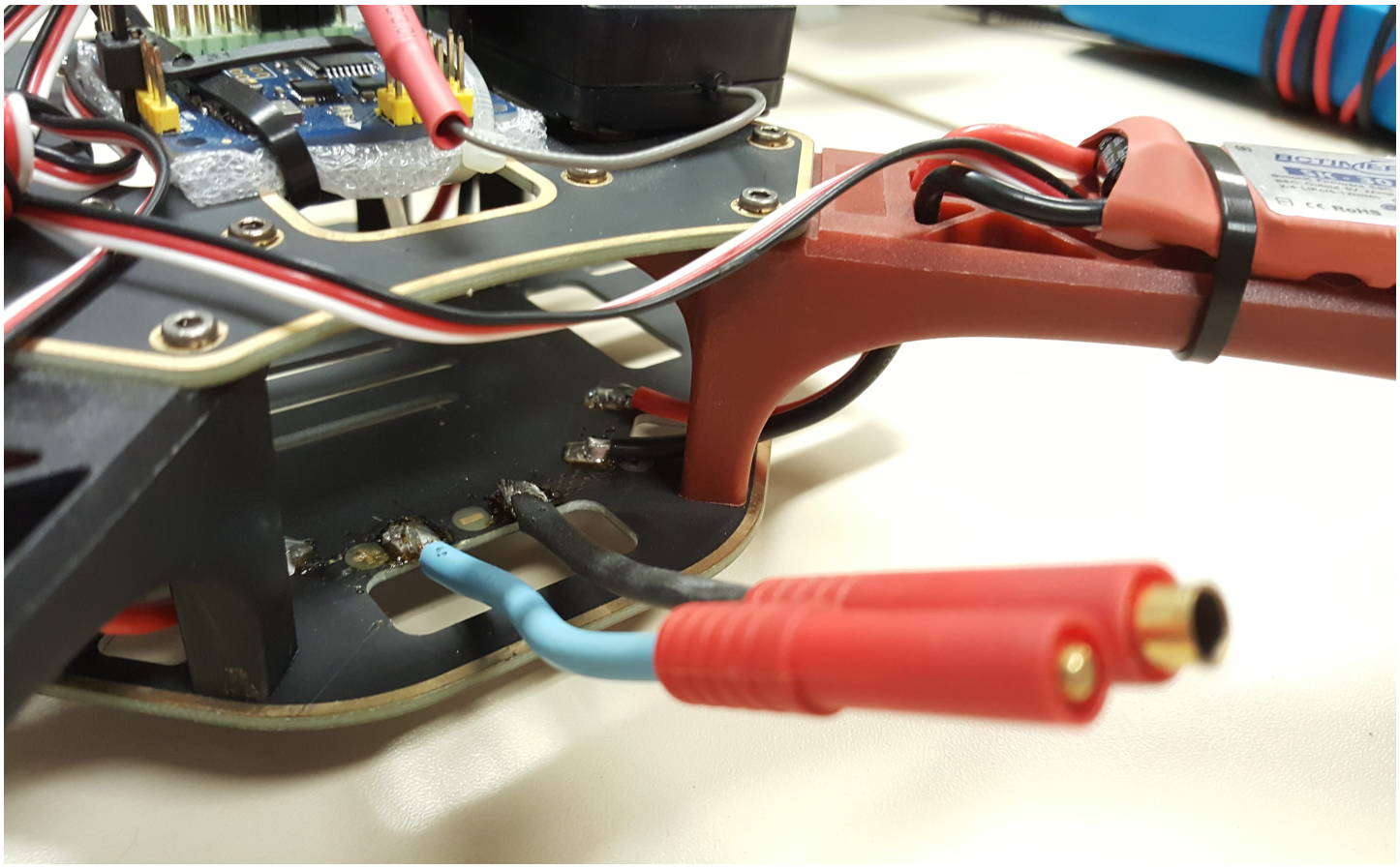


There we see the motors 11 and 3, respectively, note that the black and yellow connections are reversed from one to another. The connection of the motor 10 must be equal to the 11, while the connections of the 3 are equal to the 9, this for a quad X.
When every wire is at the right connection, you can attach it all to the frame, trying to keep the center of gravity in the center of the frame. We used plastic seals to fix the parts to the frame. The next image shows how it'll look with the battery and the propellers. Be carefull when attaching the propellers to the rotors, be sure that the propeller match the rotation direction of the rotor to produce thrust.

Power



The image shows how the ESCs are connected to the battery. Our frame works as the power distribution board, we just have to solder the ESCs power wires and the battery connector wires to the corresponding spots on the board.



Connections summary

Connections on the board:

Board	Receiver	ESCs
D2	CH3	
D4	CH1	

D5	CH2	
D6	CH4	
D7	CH8	
D8	CH7	
D3		Front left
D9		Rear right
D10		Front right
D11		Rear left

Power:

Battery	ESC
Vcc	+
Ground	-

ESCs connections to the motors, according to board connections:

	A	B	C
D3	Y	R	B
D10	B	R	Y
D11	B	R	Y
D9	Y	R	B

A, B and C corresponds to the ESCs connectors, and Y, R and B represents respectively the yellow, red and black wires of each motor.

Remote controller and flight modes

When the receiver is properly connected to the board, and the power supply is ready, we can try to turn the multirotor on. On the MultiWii, to do so, you must keep the throttle down, and put the yaw full to de right:

If you try to do that and it works, your remote controller is probably ready to go, that was not the case with our Turnigy TGY 9X.

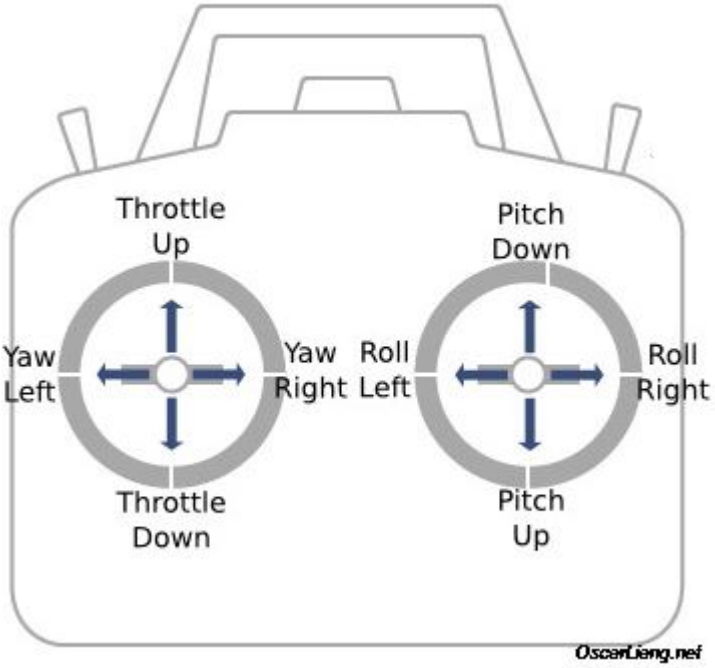
Configuring the remote controller

These are the steps to get the TGY 9X to work:

- click on the MENU button

-
  - enter on the SYSTEM SETTING menu
  -
    - Go into type select
      - select HELI and then HELI-4
      - EXIT
    - enter in STICK SET
      - select MODE2
      - click EXIT when it asks you "THRO REVERSE", we don't want it.
    - EXIT
  - enter the FUNC SETTING menu
  -
    - enter E. POINT
      - set the throttle DN (down) to 120%
      - set the rudder L (left) and R (right) to 103% (move the rudder stick to alternate between L and R)
      - click MENU to save

The STICK SET in MODE2 corresponds to that:



Now the remote controller must be able to turn the board on. If you have a different remote controller you might have to search for different orientations, but the numbers should be the same for our MultiWii. In the TGY 9X going into "MENU > FUNC. SETTING > AUX-CH" you can designate a switch or trim to a channel, we are using the HOV. THROT for CH8, with that we can put 3 different states in MultiWii auxiliar 1 (remember that we connected the board auxiliar 1 to the receiver CH8).

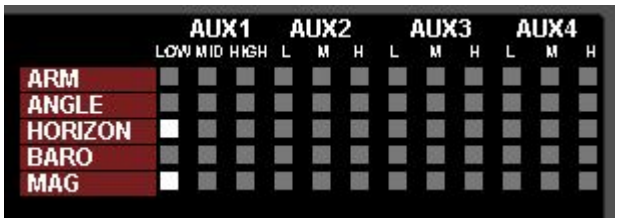
Flight modes

The MultiWii offers you different flight modes, depending on which sensors it's using, and your needs while controlling your multirotor. The following table shows which mode makes use of each sensor (x = required hardware, o = recommended hardware):

MODE	Gyroscope	Accelerometer	Barometer	Magnetometer	GPS
Acro/Gyro only	x				
Stable/Angle	x	x			
Horizon	x	x			
HeadFree	x	x	x	x	o
Altitude Hold	x	x	x		
GPS return home	x	x	o	x	x
GPS waypoint	x	x	o	x	x
GPS position hold	x	x	o	x	x
Failsafe	x				

- Arm (Acro Mode): indicates if the multirotor is armed or not for flying. If nothing else is enabled, you are automatically using the acro mode. The angle sent by your RC transmitter tells how fast the UAV wil rotate around its axes, and when you drop the sticks it'll continue to move.
- Angle (Self Level Mode): In this mode the flight controller will use the accelerometer and the gyroscope to maintain the multicopter level while there is no control input.
- Horizon: a combination between Arm and Angle modes. It'll behave as angle while calm usage of the RC transmitter sticks, and as acro when the RC sticks are at full load.
- Mag: with mag enabled the flight controller will use the magnetometer to keep the multirotor pointing to the same direction when there's no yaw input.
- BARO: uses the barometer to keep the plane at a certain height.
- GPS hold: hold the current position using GPS and BARO.
- GPS return home: use the GPS and magnetometer to return the multirotor to the initial point.

We'll set our quadcopter for the horizon and mag modes, in order to keep our multicopter self-level while the controller sticks are off, and have a smooth control of the attitude. For more detailed information about the flight modes, refer to this this. To set a mode connect the board to the PC, open the MultiWiiConf GUI, select the COM, click READ, START, and check the boxes matching the desired mode and auxiliar state. The modes can be mixed between the auxiliar states to satisfy different needs.



PID tuning

The PID tuning is made through the MultiWiiConf GUI, and it is the fine adjust of your UAV. It'll depend on the quadrotor behavior when it's taking off, landing and the attitude when flying. Here you'll find more information about what is the PID, how it works, how its coefficients values will affect the attitude of your VTOL and how can you tune it.
The MultiWii software have its owns standards values.

To change a value click and hold it, then slide the pointer to the right or left, to respectively increase or decrease the value.

Conclusion

The above instructions tells the basics of how to set-up a multirotor using a MultiWii flight control board. Despite that we're using quadcopters in X configuration, the steps can be easily adapted to other build types. There are plenty a lot of material on the internet about this subject, and in the end of this page you'll find a lot of links leading to where the information here comes from.

The finished quadcopter:

The battery plugged, and the board armed:

Files

MultiWii 2.4.zip - MultiWii v2.4 files. Taken from the MultiWii open source project page

References

Quadcopter beginner guide - A good read for beginners

MultiWii - MultiWii open source project page

MultiWii wiki - MultiWii project Wiki

Flight modes - More information about each flight mode and its sensors

PID Tuning - Information about how a PID works, and how its coefficient values will affect the attitude of the multirotor

Youtube playlist with tutorials and tips - This link leads to a youtube playlist that contains a guide to set-up a MultiWii multirotor and a lot of information and tips related to the subject

Arduino software - Link to the Arduino software, that we use to edit the MultiWii codes and send the software to the board

Tutorial for the MultiWiiConf GUI - More detailed information about the MultiWiiConf GUI and its usage