FEEG2001: Quadcopter Electronics Guide

Table of Contents

Introduction

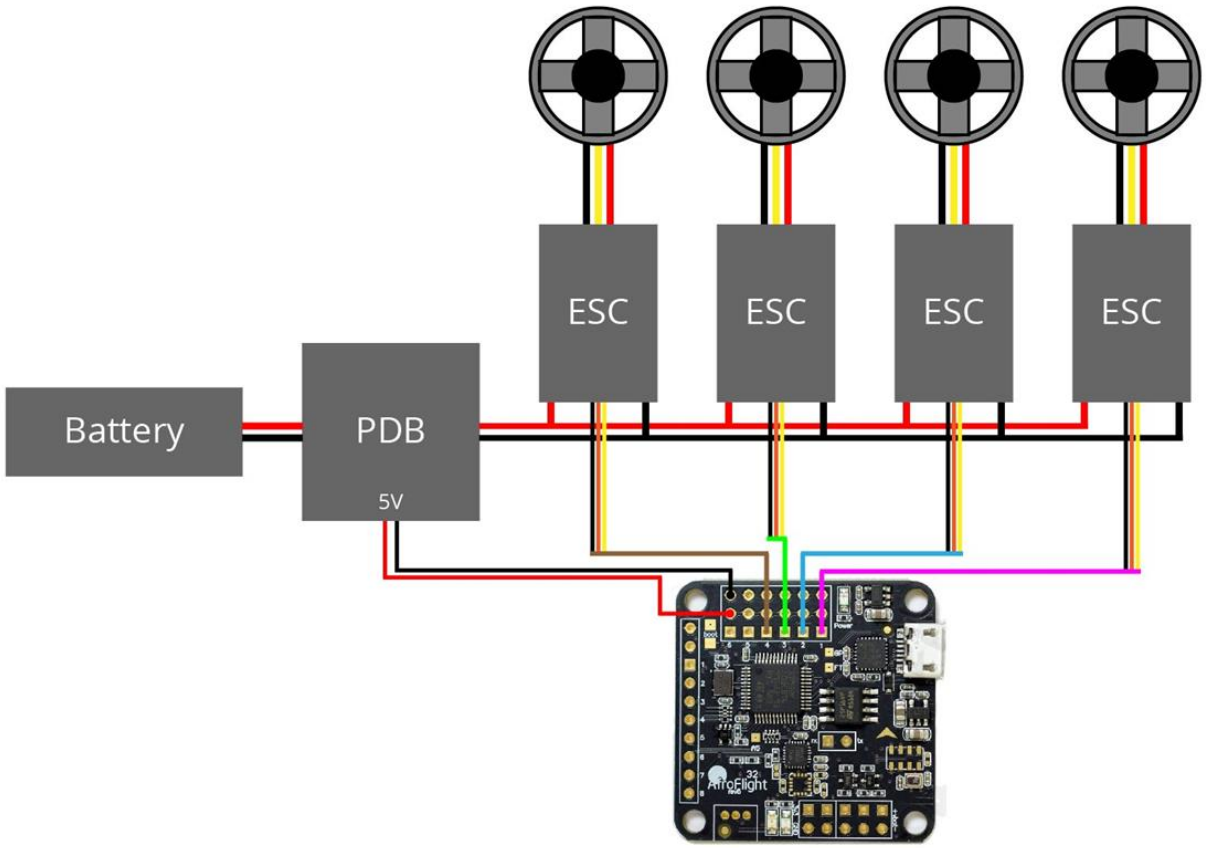
The following document aims to serve as a guide to setting up the wiring loom for a basic quadcopter with optional Arduino control for those who have never built a quadcopter before and who are unsure of how to connect the provided components together. The below information can be modified depending on the sensor configuration or Arduino board used etc. Please note that this basic guide uses the NAZE32 flight controller and the locations of pins etc. may be slightly different in derivative controllers e.g. the Skyline32 or SP Racing. Please refer to the manual for the specific board for the location of the correct pins.

Terminology & Definitions

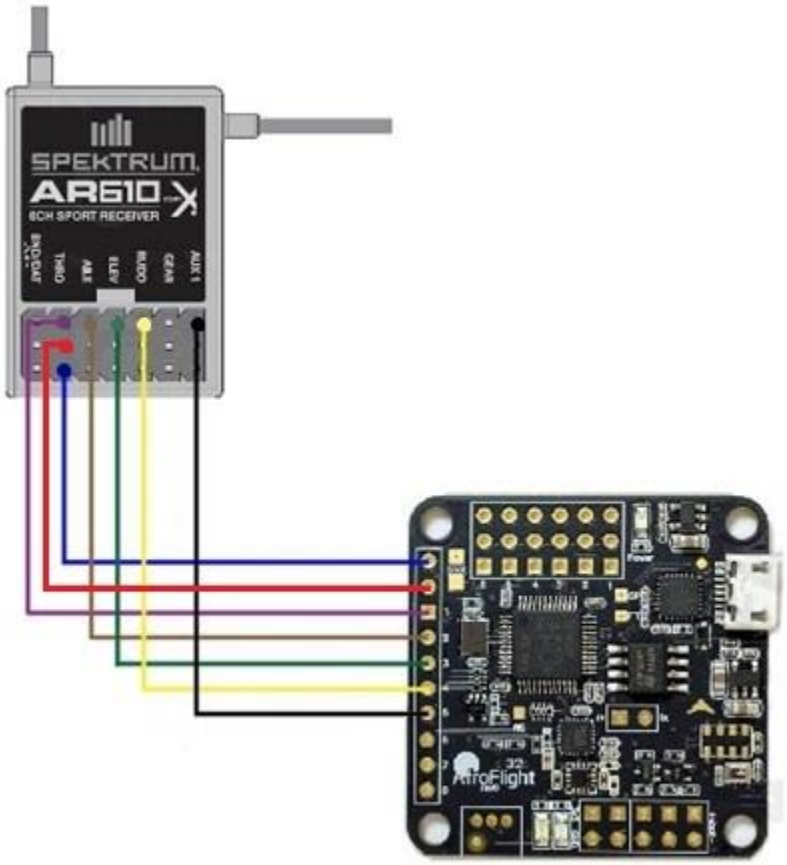
- **RX** (AR610 Receiver) – receives signals from the pilot's transmitter. This outputs PWM signals which can control servos, ESCs or be fed into the flight controller or Arduino.
- **FC** (Afro Flight Naze32 flight controller) – Takes the pilot's commands from RX or PWM commands from the Arduino and using an on-board accelerometer stabilizes the quadcopter. The flight controller outputs signals that drive each of the four ESC's. The FC therefore controls the speed of the motors in order to keep the aircraft level and prevent yaw. To stabilise the aircraft the FC has both an accelerometer and gyroscope built in.
- **ESC** (electronic speed controller) – Controls the rotational speed of a motor. Takes a PWM control input from the FC and drives the motor accordingly.
- **PDB** (power distribution board) – Connects directly to the battery and splits power to the four ESC's. Also has an on-board voltage regulator that provides a stable 5V or 12V output to power other electronic boards, such as the Arduino, FC and RX.
- **Arduino UNO** – Has a number of potential functions. Firstly, it can "listen" to the pilot's input from the RX to switch between operational modes using the "pulsein" command. Secondly, it reads the sensor data and runs and custom control algorithm(s). Finally, it records data to an SD card for post-flight analysis.

Motor Connection

The battery connects directly to the power distribution board (PDB). The power is then split and goes to the fours ESCs that are connected to the PDB. The PDB also provides 5V stable output that powers the flight controller. Each ESC connects to a motor using 3 cables. To change the rotation of the motor, switch the order of two arbitrary cables. The ESC signal wires connect directly to the flight controller pin header.



Arduino & RX Connection



The RX can be powered from the 5v and ground pins on the left of the FC (blue and red lines in the above image). The middle row of pins on the RX all accept a 5v input while the bottom row of pins are all ground. While the RX is powered using the throttle pins in the above image, any of the other sets of pins could be used. However, it's highly recommended the bind pins are not used.

The FC should NOT be used to power any other electronics otherwise the board is at risk of overloading and being destroyed. Instead, power the Arduino directly from the PDB.

Throttle, ailerons, elevator and rudder (purple, brown, green and yellow in the above image) should be connected to the corresponding input pins on the FC. The AUX 1 output on the RX is connected to the FC and used to arm/disarm the aircraft. No aircraft will be flown without first demonstrating the ability to arm and disarm.

The Arduino can be connected to the "Gear" channel on the RX and the "pulsein" command used to monitor the PWM value on this channel. The magnitude of this value can then be used to switch between modes on the Arduino. Alternatively, an interrupt can be defined on the Arduino for improved switching performance.