

# Hardware Assembly

## Power Distribution

Just like the human body has a circulatory system to carry oxygen-rich blood to wherever it is needed, the drone has a power distribution board to take the all-important battery power and send it to every component. In this step, you will mount 4 ESCs (electronic speed controllers), a BEC (battery eliminator circuit), a battery lead with XT60 connector, and a battery monitor lead to your PDB (power distribution board). ### Strip Wires Cut off all the connectors on the thick wires of the ESC as close to the connector as possible. Leave the PWM signal connector alone. Strip 0.5 cm from each of the wires you just cut. [ESC PRE POST SNIP] () Strip 0.5cm off your battery lead with XT60 connector. [XT60 PIGTAIL STRIPPED] () Inside the plastic box labeled, SKYLINE 32, find the battery monitor lead, cut off the two larger connectors, and strip 1cm. [SKYLINE BATTERY MONITOR PRE POST STRIP] () ### Tin PDB and Stripped wires. Tin all 20 ESC leads. Twist the battery monitor leads around the BEC leads and tin. Thoroughly tin the battery lead with XT60 connector. It is important that solder flows all the way through the exposed wire.

## Check your components

Before you start assembling your drone, it's important to make sure you have the right components. For the hardware assembly, you will need \* 4 ESCs (Electronic Speed Controllers). These convert DC power from the battery to three-phase AC, which drives the motor at the desired speed. [ESC PACKS] () \* 4 Motors. These brushless motors are driven by three phase AC (hence the three wires). On the bottom of the boxes, you will notice the code SE1806-xy, where x is either 11 or 13, and y is either CW or CCW. Make sure all your motors have the same x

## Powertrain Assembly

Solder Everything to PDB Stripping Combine Skyline 12v cable with Pi 5v Supply (BEC) Tinning Soldering Check motors Screw nuts onto motor Mount motors to frame Solder ESCs to motors Heat shrink Tinning Soldering Mount PDB Mount ESCS

1. Screw the nuts onto the correct motors so you know which motor goes where. Do this one box at a time so that you don't mix up the CW and CCW threaded motors. Red nuts screw CW.
2. Screw motors into frame in correct orientation. [Picture of direction motors should spin] Make sure to route the wire through a hole in the frame to

the underside of the frame.

3. Now, solder several components to the PDB. See the Soldering section for more details. Solder each of the four ESCs (the end with only two thicker wires) onto the PDB. This may require cutting off the ends of the wires and stripping them. Solder the BEC onto the PDB. Solder the battery connector to the PDB Solder the battery voltage checker [ref picture] to the same pads as the battery connector
4. Solder each motor to an ESC. Might have to cut off the end of the esc connector to strip the wire.
5. Attach PDB w/ standoffs
6. Route ESC wires nicely with zip ties. There is no single correct way to do this. We just need to ensure that the wires will not be cut by the spinning props. It is best to secure the ESCs and wires underneath the frame as in [picture]
7. Flash ESCs (battery required)
8. Attach the PWM cable to the Skyline
9. Attach the ESC PWM controls to the Skyline via the cable from 8. RR = 1, FR = 2, RL = 3, FL = 4
10. Attach the battery voltage check wire to the Skyline
11. Use ClearFlight to test the direction of the motors. [see image for correct rotation direction. Diagonals spin the same direction]. Mark which motors need to be reversed. (battery required)
12. For each motor that must be reversed, plug in its ESC PWM input to the ESC flashing device. Read ESC -> reverse -> write ESC [image(s)] (battery required)
13. Attach the Skyline to the “front deck” of the frame with double-sided sticky tape.
14. Screw the Pi mounting bracket onto the standoffs on the PDB. Now is a good time to make sure the ESC wires are mounted nicely and tucked inside of this mounting bracket [image]
15. Attach the Pi to the mounting bracket with two screws. The USB port on the Pi should be facing forward