Gyro

The first step in setting up the gyro is to solder stub pins into the eight port connections of the gyro. For this I suggest placing the gyro in the centre of the breadboard and pushing the stub pins though each of the connection holes until they are in as far as the breadboard will allow. Then take the soldering iron and solder and carefully place a small amount of solder on each to hold the pins in place. Then, remove the gyro and pins carefully from the breadboard (for example use a small screwdriver to ply the gyro off). The turn the gyro over and solder the bottom to secure the connection and pins in place.

Following this, the gyro can be placed back on the breadboard, roughly in the centre. Then take the Arduino Uno and you can connect the wires following the diagram in Figure \*\*\* or using the steps given below.

1. Place one end of a red wire in the breadboard pin directly adjacent to the Vin port of the gyro and the other end in a 5V port of the Arduino Uno.
2. Again take a red wired and place one end in the pin directly adjacent to the A0 port of the gyro and the other end in a 5V port of the Arduino Uno.
3. Take a black wire and put one end in the pin directly adjacent to the Gnd port of the gyro and the other in a ground port of the Arduino Uno.
4. Take two other wires of different colour for the I2C connection to the gyro. With one wire place one end in the pin directly adjacent to the \*\*\* port of the gyro and the other in the A0 port of the Arduino Uno. This will enable the clock to the gyro ensuring the microcontroller and peripheral are synchronised. For the second wire place one end in the pin directly adjacent to the \*\*\* port on the gyro and the other in the A1 port of the Arduino Uno. This allows data transfer between the microcontroller and the peripheral.

Image

Figure \*\*\*

The power for the Arduino Uno is provided from your laptop or computer, using the USB cable provided. First open up the Arduino software and load in the Gyro\_Configuration\_Test software. It is then advised to go through each stage of the code and familiarize yourself with the workings of the code. Each stage is explained below.



Following this load compile the code and upload to the board as is demonstrated in Figure \*\*\*. You can then see on the Serial monitor the data feedback from the gyro as is shown in Figure \*\*\*. Then try rolling the gyro and you should see a positive roll rate generated on monitor, and the same for pitch and yaw.

Image

Figure \*\*\*

Image

Figure \*\*\*

After this you are advised to make a copy of the code and modify and test out one version. This will improve your familiarity with how the gyro works.

Receiver and Transmitter

The next project stage is to set up the transmitter and receiver. First place the batteries in the back of the transmitter and turn it on to test that you have placed the batteries in correctly, a green light should appear on the front. You can now turn it off again and get the receiver. You need to bind the transmitter and receiver for them to communicate with each other. To do this complete the following steps.

1. Place the bind plug in the BAT port of the receiver.
2. In channel 1 of the receiver place one female end of a male to female red wired on the pin furthest away from the labels.
3. In channel 1 of the receiver place on female end of a male to female black wire on the pin in the middle of the three available.
4. Plug the red wire into a 5V port of the Arduino Uno and the black wire in a ground port of the Arduino Uno.
5. Power the Arduino Uno either through your laptop using the USB cable or using the power supply provided. You should see a light on the receiver.
6. Then holding the bind button on the front of the transmitter turn it on. The light should change on the receiver.
7. Turn off the transmitter.
8. Unplug the power, ground and bind plug on the receiver.

Step by step images are provided below.

Image

Figure \*\*\*

Next, connect a male to female cable from port 8 on the Arduino Uno to the pin closest to the label in channel of the receiver. Then the same, but a different colour wire, from port 9 on the Arduino Uno to the pin closest to the label in channel 2. Again, repeat this for port 10 to channel 3 and port 11 to channel 4. Then connect a red wire in the pin furthest from the label in the BAT port of the receiver to a 5V port of the Arduino Uno. Then connect a black wire in the pin in the middle of the BAT port of the receiver to a ground port of the Arduino Uno. Then connect the USB cable to your laptop or computer and the Arduino Uno to provide power and turn on the transmitter.

Then open up the Arduino software and load in the Recevier\_Transmitter\_Test code. You should then follow the following steps to familiarize yourself with the code.



Finally, upload the code and look at the serial monitor. You should then see data values displayed representing movements on your transmitter. This should be similar to that shown in Figure \*\*\*.

Image

Figure \*\*\*

Electronic Speed Convertors

The next stage is to set up the ESC’s. These provide pulses to the motors which determine the motor RPM. The signal pulse length is determined from both your input from the transmitter and the gyro to aid stabilisation.

The first stage is to configure the ESC’s so they respond directly to transmitter input. This will also act as a check that both the ESC’s and motors are working. You need to have the frame build as described in the 1. \*\*\* Hardware section. You are going to need the battery for this section, so as it takes a while to charge you should begin this now. Firstly get the battery out and the battery charger; plug in the battery charger and connect the battery as is demonstrated in Figure \*\*\*. This involves connecting the red and black wires and the \*\*\* cable. Next, follow these steps to program the battery charger to the correct settings.



Then mount the motors onto the frame, again, use section 1. \*\*\* Hardware. Next, sit the power distribution in the centre of the frame and connect from each of the four ESC’s the red and black cables to the power distribution board as is shown in Figure \*\*\*. Following this connect the three wires red, black and brown \*\*\* from the ESC to the three black wires of the motor; use Figure \*\*\* to connect these in the correct directions, making note of the direction of the quadcopter to ensure CW and CCW motors are correct.

Image

Figure \*\*\*

Image

Figure \*\*\*

Next, take four male to male wires and strip one end of each of them. Then take one female to female wire and strip on end of this. The result should be five wires which look as follows in Figure \*\*\*.

Image

Figure \*\*\*

Then, take the soldering iron and change the end by unscrewing it and replacing it with the pointed end. This should look like that demonstrated in Figure \*\*\*.

Image

Figure \*\*\*

Following this, turn on the soldering iron and adjust it to be approximately 250oC. Take some solder from the real and solder the four male ended wires together, this should look like that shown in Figure \*\*\*. Then take the female ended wire and applying a small amount more solder if required push it in-between the four wires so that you are left with a 1 female to 4 male wire as is shown in Figure \*\*\*.

Image

Figure \*\*\*

Image

Figure \*\*\*

Then you can take this and connect the four male ends up to the remaining connection to the ESC’s. You should connect the male ends in line with the brown wires for the ESC’s. This is demonstrated for one ESC in Figure \*\*\*.

Image

Figure \*\*\*

Then connect the female end with the receiver. This should be connected into channel three in the pin closest to the label. From one ESC, using a red male to female wire, connect the red wire on the ESC to the pin on the transmitter which is in the BAT channel and furthest away from the label. This is demonstrated in Figure \*\*\*.

Image

Figure \*\*\*

You should now be able to connect the battery if it is fully charged and hear a collection of beeps from the ESC’s. Then turning on the transmitter with the throttle held in the maximum location before moving it to the lowest location will calibrate the ESC’s and this is signified by another collection of beeps. The stick movements are shown in Figure \*\*\*. Applying movements in the throttle should now activate the motors. NOTE: you should not attach the propellers at this stage!

Image

Figure \*\*\*

The next stage is to control the ESC’s with the transmitter through the flight controller. For this take the flight controller and place wires from the brown connections of ESC’s 1, 2, 3 and 4 to pins 8, 9, 10 and 11 of the flight controller respectively. This is demonstrated in Figure \*\*\*.

Image

Figure \*\*\*

Next, connect wires from pins 4, 5, 6 and 7 to pins on the receiver closest to the label representing channels 1, 2, 3 and 4 respectively. This is again demonstrated in Figure \*\*\*.

Image

Figure \*\*\*

Load the Arduino software and load up the ESC\_Test\_Software. You should then go through this step by step to familiarise yourself with the workings of the code. The following steps explain the main points of the code.



After this execute the code using the USB cable to connect your laptop or computer to the flight controller. You should then be able to control the motors using the receiver through the flight controller. Using the serial monitor, you will be able to observe the pulse lengths sent to the motors. Further, you are now able to control the basic motions of flight for the quadcopter as throttle, pitch, roll and yaw are all controllable and the respective pulse lengths sent from each ESC is observable.

Flight Controller