

Hello friends in this tutorial I am showing you how to control Brushless dc motor aka BLDC motor

with Arduino and joystick

Step 1: HOW a BLDC WORKS

A brushless DC motor (also known as a BLDC motor) is an electronically commuted DC motor which does not have brushes. The controller provides pulses of current to the motor windings which control the speed

These types of motors are highly efficient

Brushless DC motor has two basic parts: the rotor and the stator. The rotor is the rotating part and has rotor magnets whereas stator is the stationary part and contains stator windings.

Step 2: ESC Aka ELECTRONIC SPEED CONTROLLER

An electronic speed control follows a speed reference signal (derived from a throttle lever, joystick, or other manual input) and varies the switching rate of a network of field-effect transistors (FETs) By adjusting the duty cycle or switching frequency of the transistors, the speed of the motor is changed. The rapid switching of the transistors is what causes the motor itself to emit its characteristic high-pitched whine, especially noticeable at lower speeds.

Different types of speed controls are required for brushed DC motors and brushless DC motors. A brushed motor can have its speed controlled by varying the voltage on its armature. (Industrially, motors with electromagnet field windings instead of permanent magnets can also have their speed controlled by adjusting the strength of the motor field current.) A brushless motor requires a different operating principle. The speed of the motor is varied by adjusting the timing of pulses of current delivered to the several windings of the motor.

Step 3: Components Needed for This

BLDC motor

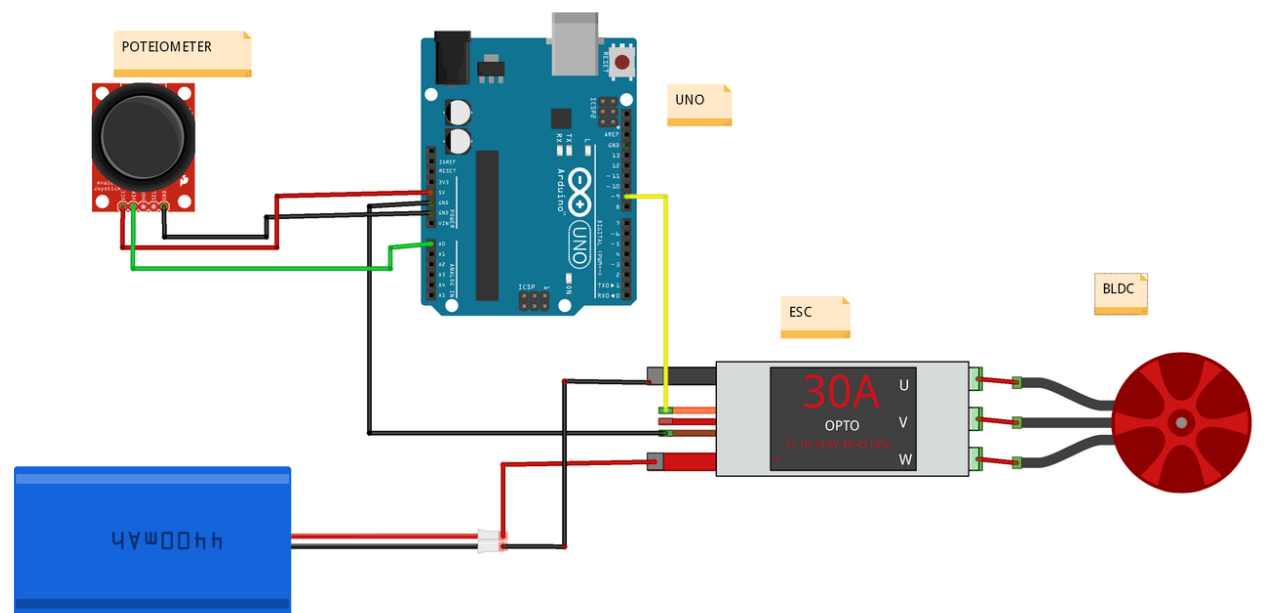
ESC

7.4V battery

Arduino

joystick

Step 4: Circuit Diagram



fritzing

Step 5: Code

```
#include <Servo.h>

Servo ESC; // create servo object to control the ESC

int potValue; // value from the analog pin

void setup() {

  // Attach the ESC on pin 9

  ESC.attach(9,1000,2000); // (pin, min pulse width, max pulse width in microseconds)

}

void loop() {

  potValue = analogRead(A0); // reads the value of the potentiometer (value between 0 and 1023)

  potValue = map(potValue, 0, 1023, 0, 180); // scale it to use it with the servo library (value between 0
and 180)

  ESC.write(potValue); // Send the signal to the ESC

}
```