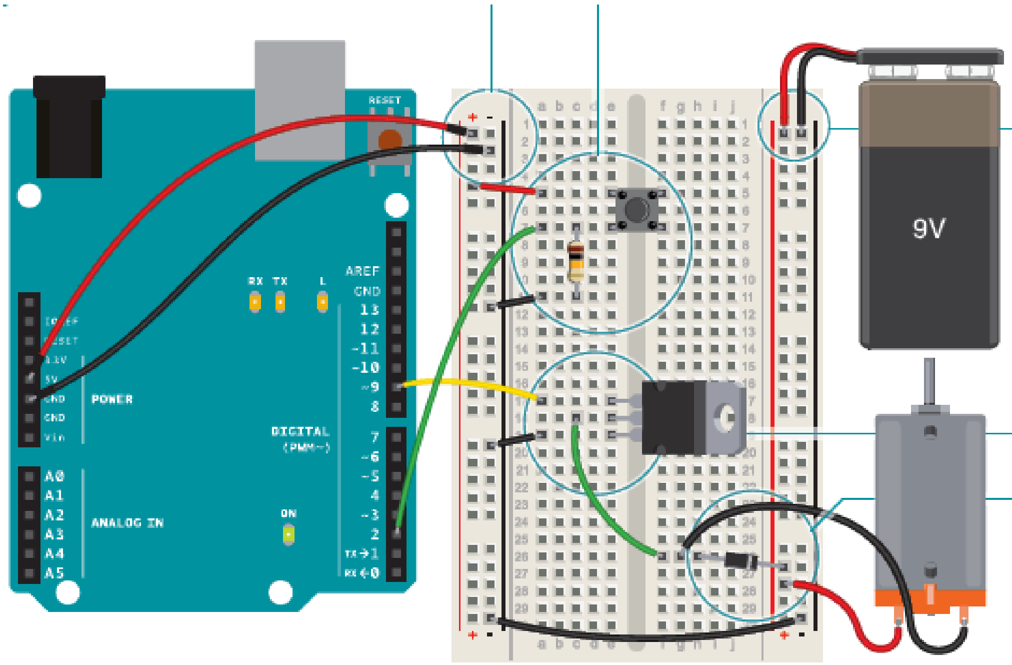
9.DC brushless Motor

Motors are a type of inductive device. Induction is a process by which a changing electrical current in a wire can generate a changing magnetic field around the wire. When a motor is given electricity, a tightly wound coil inside the housing of copper creates a magnetic field. This field causes the shaft (the part that sticks out of the housing) to spin around.

Motors typically require more current than the Arduino can provide. Some motors require a higher voltage as well. To start moving, and when it has a heavy load attached, a motor will draw as much current as it can. The Arduino can only provide 40 milliamps (mA) from its digital pins, much less than what most motors require to work. Transistors are components that allow you to control high current and high voltage power sources from the low current output of the Arduino. You will use a MOSFET (IRF520). A motor can generate electricity when the shaft is spun around. Try attaching an LED to the two leads of your motor, then spin the shaft with your hand. If nothing happens, spin the shaft the other way. The LED should light up. You've just made a tiny generator out of your motor. When you stop supplying energy to a motor, it will continue to spin, because it has inertia. When it's spinning, it will generate a voltage in the opposite direction than the current you gave it. You saw this effect when you made your motor light up an LED. This reverse voltage, sometimes called back-voltage, can damage your transistor. For this reason, you should put a diode (IN4007) in parallel with the motor, so that the back voltage passes through the diode. The diode will only allow electricity to flow in one direction, protecting the rest of the circuit.

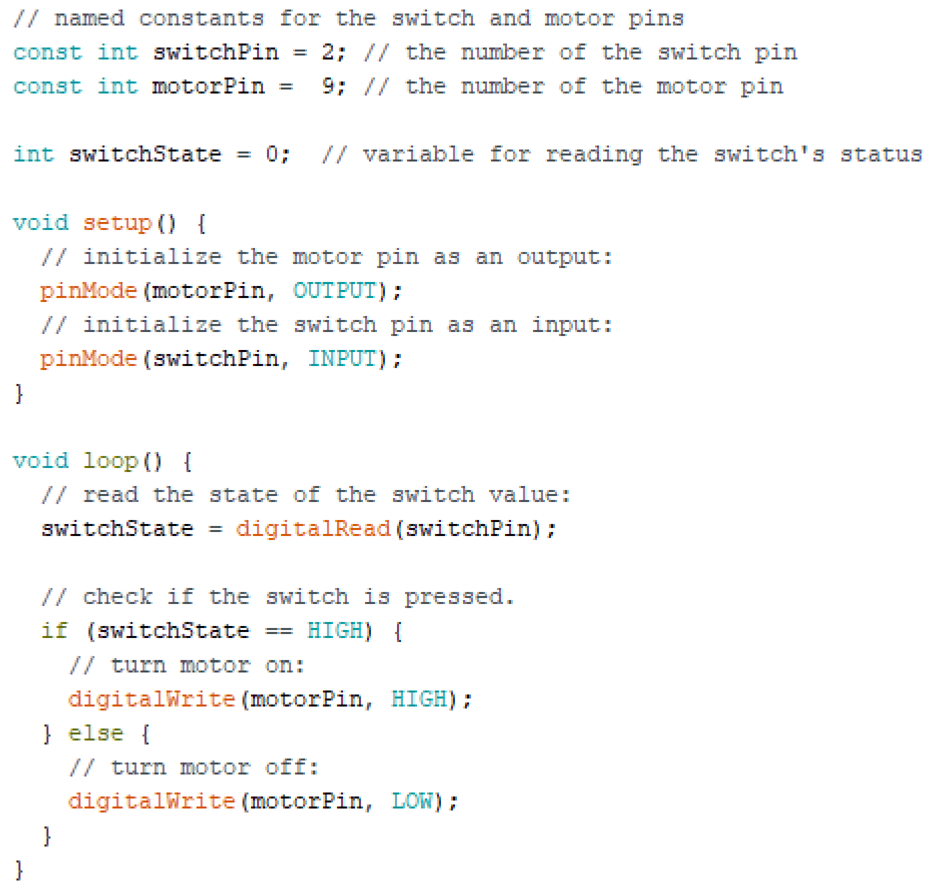


provide a common ground

Add a momentary switch to the board, connecting one side to power, and the other side to digital pin 2 on the Arduino. Add a 10-kilohm pull-down resistor to ground on the output pin of the switch.

Place the transistor on the board. Look at the component so that the metal tab is facing away from you. Connect digital pin 9 to the left pin (gate). A change in voltage on the gate makes a connection between the other two pins. Connect one end of the motor to the middle pin (drain). When the Arduino activates the transistor by supplying voltage to the gate, this pin will be connected to the third pin, called the source. Connect the source to ground.

Notice that the diode has a stripe on one end. That end is the negative end, or cathode. The other end is the anode. Connect the anode of the diode to the ground of the motor and the cathode of the diode to the power of the motor.



```
// named constants for the switch and motor pins
const int switchPin = 2; // the number of the switch pin
const int motorPin =  9; // the number of the motor pin

int switchState = 0;  // variable for reading the switch's status

void setup() {
  // initialize the motor pin as an output:
  pinMode(motorPin, OUTPUT);
  // initialize the switch pin as an input:
  pinMode(switchPin, INPUT);
}

void loop() {
  // read the state of the switch value:
  switchState = digitalRead(switchPin);

  // check if the switch is pressed.
  if (switchState == HIGH) {
    // turn motor on:
    digitalWrite(motorPin, HIGH);
  } else {
    // turn motor off:
    digitalWrite(motorPin, LOW);
  }
}
```

The code is remarkably similar to the code you first used for turning on an LED. First of all, set up some constants for the switch and motor pins and a variable named switchState to hold the value of the switch.

In your setup(), declare the pinMode() of the motor (OUTPUT) and switch (INPUT) pins.

Your loop() is straightforward. Check the state of the switchPin with digitalRead().

If the switch is pressed, turn the motorPin HIGH. If it is not pressed, turn the pin LOW. When HIGH, the transistor will activate, completing the motor circuit. When LOW, the motor will not spin.