



([https://www.dfrobot.com/category-](https://www.dfrobot.com/category-203.html)

[203.html](https://www.dfrobot.com/category-203.html))

Introduction

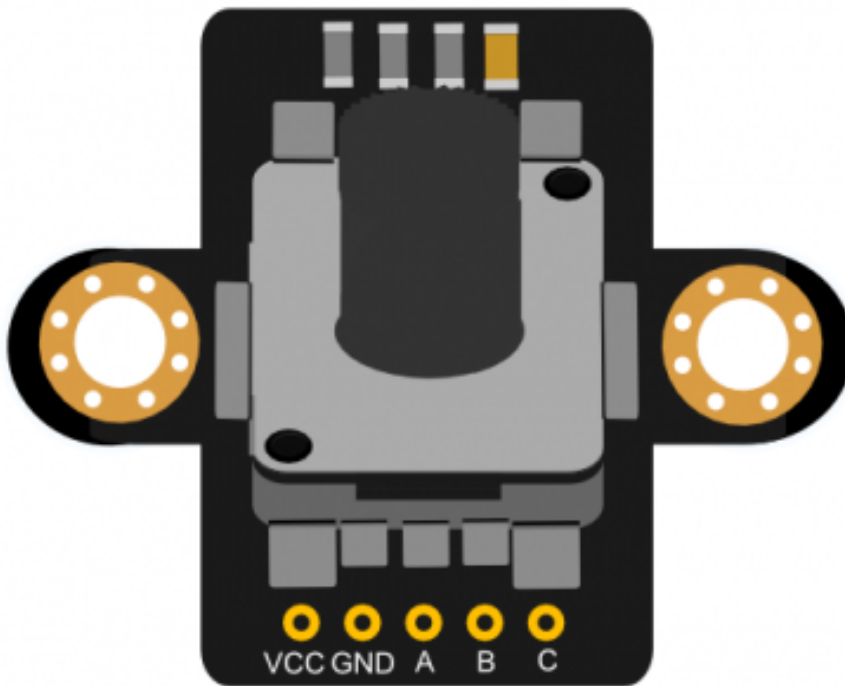
DFRobot 360 degree rotary encoder module is designed based on EC11 rotary encoder button. This module has three signal terminals: terminal A & B is encoder output; terminal C is button signal output. It is very suitable for applications such as volume knob, lighting adjustment. The rotary encoder module comes with XH2.54 bonding pad, easy to be used in prototyping projects, like automotive electronics, multimedia speakers, instrumentation, smart home and other fields.

Specification

- Operating Voltage: 3.3V ~ 5V
- Full Rotation Angle: 360 degrees (no stop point)
- Number of pulse: 20
- Rotation Life: 30000 ± 200 Cycles

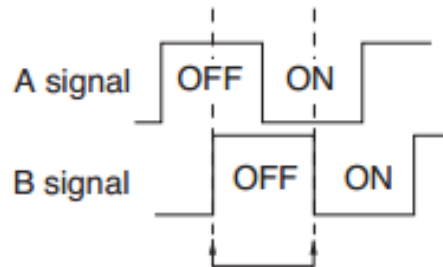
- ROTATION LIFE: 50000 ± 200 CYCLES
- Contact Resistance: $\leq 100\text{m}\Omega$
- Insulation Resistance: 100M Ω or more
- Oscillation: <10ms Press
- Working Temperature: -30 °C ~ 80 °C
- Module Size: 33.8 * 22.4 (mm) /1.3 * 1.1 (inches)
- Weight: 14g

Board Overview



Num	Label	Description
1	VCC	3.3~5V
2	GND	GND
3	A	Encoder-phase A
4	B	Encoder-phase B
5	C	Button

CW Direction: phase A B signal:

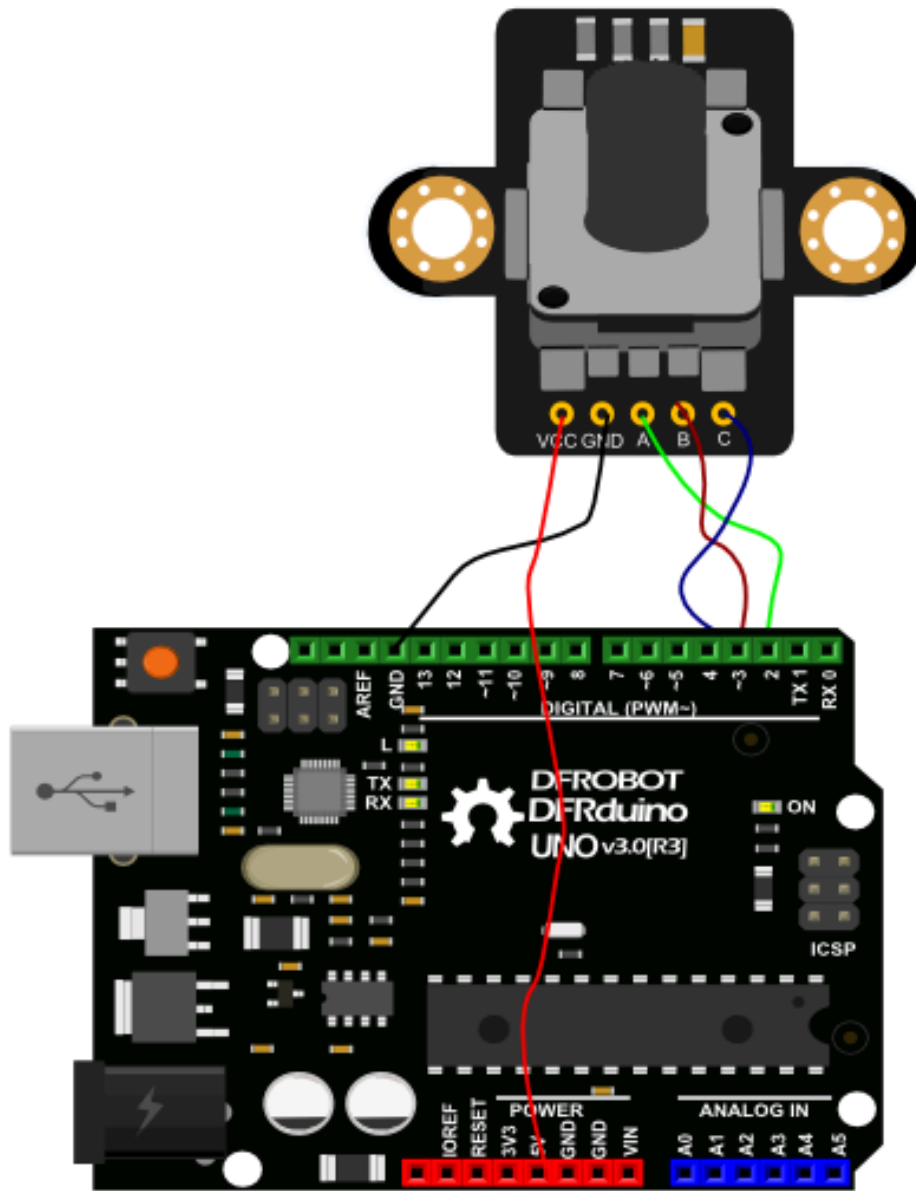


Tutorial

Requirements

- **Hardware**
 - DFRduino UNO (or similar) x 1
 - XXX x #
 - M-M/F-M/F-F Jumper wires
- **Software**
 - Arduino IDE (Version requirements: V1.6.?),
[<https://www.arduino.cc/en/software> (<https://www.arduino.cc/en/software>)]
Click to Download Arduino IDE from Arduino®]

Connection Diagram



Sample Code

```
int encoderPinA = 2;
int encoderPinB = 3;
int buttonPin = 4;

volatile int lastEncoded = 0;
volatile long encoderValue = 0;

long lastencoderValue = 0;

int lastMSB = 0;
int lastLSB = 0;

long readEncoderValue(void){
    return encoderValue/4;
}

boolean isButtonPushDown(void){
    if(!digitalRead(buttonPin)){
        delay(5);
        if(!digitalRead(buttonPin))
            return true;
    }
    return false;
}

void setup() {
    Serial.begin (9600);

    pinMode(encoderPinA, INPUT);
    pinMode(encoderPinB, INPUT);
    pinMode(buttonPin, INPUT);

    digitalWrite(encoderPinA, HIGH); //turn pullup resistor on
    digitalWrite(encoderPinB, HIGH); //turn pullup resistor on
```

```
//call updateEncoder() when any high/low changed seen
//on interrupt 0 (pin 2), or interrupt 1 (pin 3)
attachInterrupt(0, updateEncoder, CHANGE);
attachInterrupt(1, updateEncoder, CHANGE);

}

void loop(){
  //Do stuff here

  if(isButtonPushDown()){
    Serial.println("you push button down!!!");
  }
  Serial.println(readEncoderValue());
  delay(50); //just here to slow down the output, and show it will work even
}

void updateEncoder(){
  int MSB = digitalRead(encoderPinA); //MSB = most significant bit
  int LSB = digitalRead(encoderPinB); //LSB = least significant bit

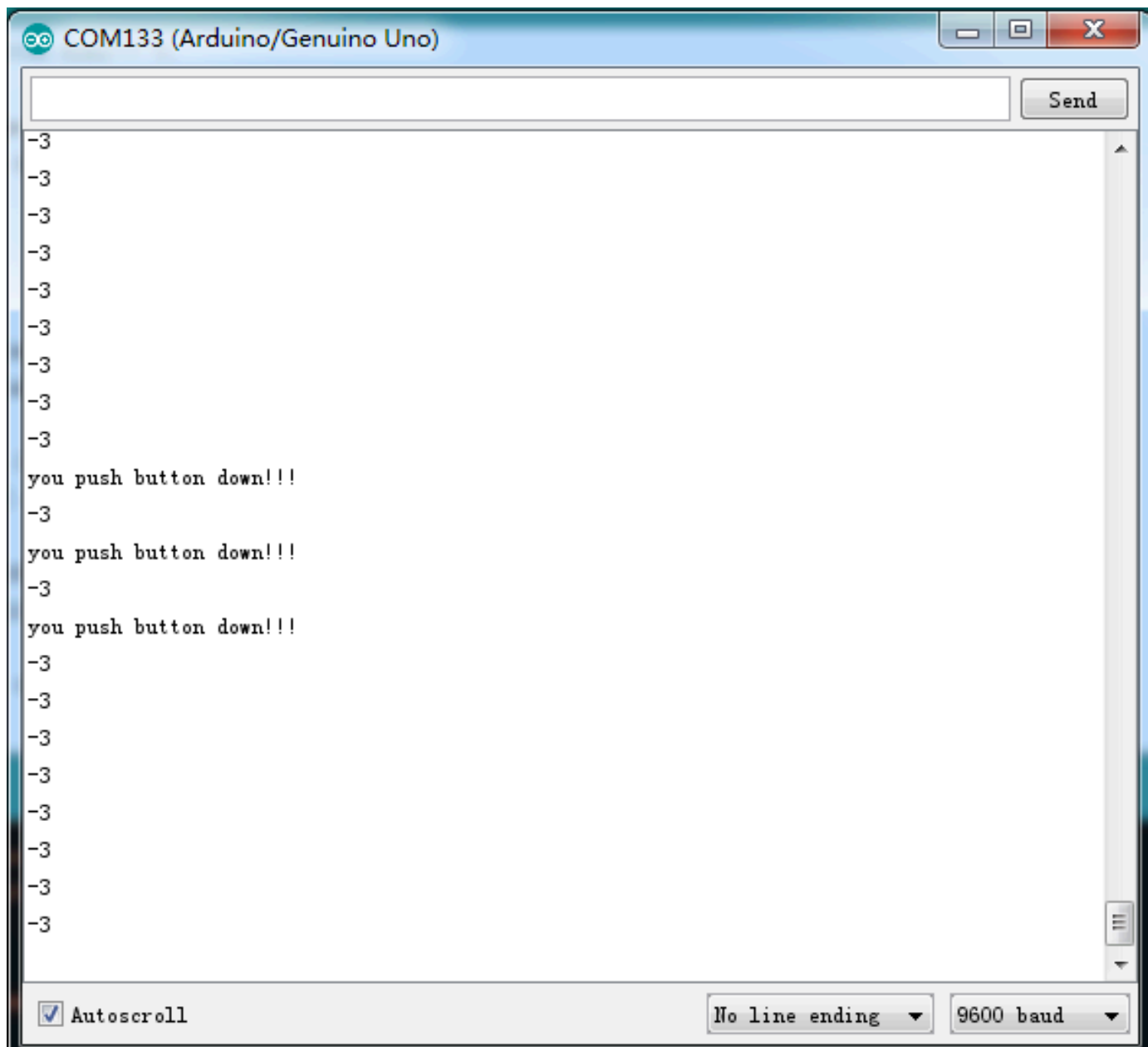
  int encoded = (MSB << 1) | LSB; //converting the 2 pin value to single number
  int sum = (lastEncoded << 2) | encoded; //adding it to the previous encoded

  if(sum == 0b1101 || sum == 0b0100 || sum == 0b0010 || sum == 0b1011) encoder
  if(sum == 0b1110 || sum == 0b0111 || sum == 0b0001 || sum == 0b1000) encoder

  lastEncoded = encoded; //store this value for next time
}
```

Expected Results

- Clockwise Rotation: 1
- Anticlockwise Rotation: -1
- Press the Button: you push button down



FAQ

For any questions, advice or cool ideas to share, please visit the **DFRobot Forum** (<https://www.dfrobot.com/forum/>).

More Documents

- Schematic

(https://github.com/Arduinolibrary/DFRobot_SEN0235_EC11_Rotary_Encoder_Module/raw/master/EC11%20Rotary%20Encoder%20Module%20Schematic%20_V1.0_.pdf)

- Dimension

(https://github.com/Arduinolibrary/DFRobot_SEN0235_EC11_Rotary_Encoder_Module/raw/master/EC11%20Rotary%20Encoder%20Module%20Dimension%20_V1.0_.pdf) EC11 Forward Direction Series Specification.pdf
(<https://dfimg.dfrobot.com/nobody/wiki/0385d189fd3e6dec54fda6e6658039e6.pdf>)