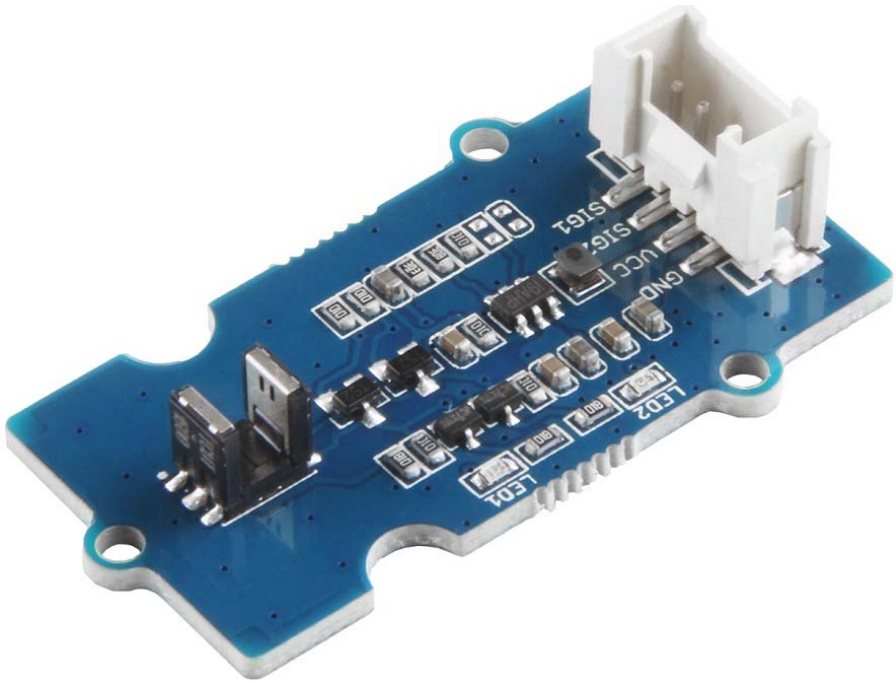


Grove - Optical Rotary Encoder(TCUT1600X01)

The Grove - Optical Rotary Encoder(TCUT1600X01) is a transmissive sensor that includes an infrared emitter and two phototransistor detectors. Usually, the infrared emitter emits infrared rays, the phototransistor detectors receives the infrared rays, then the phototransistor is turned on, both of the output is High, the on-board LED indicators light up. When there is an obstacle blocking, the phototransistor can not receive the infrared rays, so the phototransistor will be turned off and both of the output will be Low, the on-board LED indicators fade away.

You can use this sensor as a rotary encoder to detect the speed or rotation, and thanks to the two phototransistor detectors, you even can detect the rotation direction.

Features

- Double phototransistor detectors, can determine the direction of rotation
- On-board LED indicators
- Grove Interface

Specification

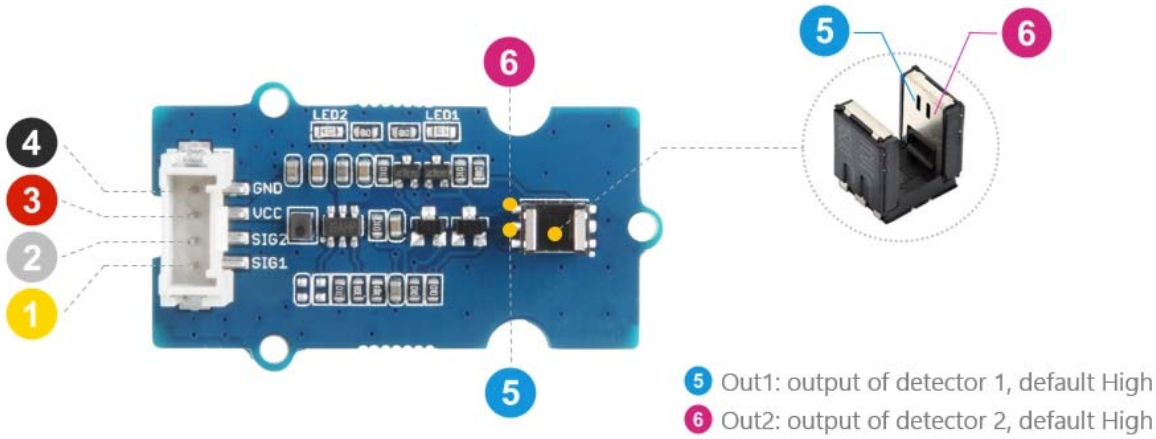
Item	Value
Operating voltage	3.3V / 5V
Operating temperature	-40°C to +105°C
Storage temperature Range	-40°C to +125°C
Emitter wavelength	950 nm
Gap	3 mm
Interface	Digital

Applications

- Automotive optical sensors
- Accurate position sensor for encoder
- Sensor for motion, speed, and direction
- Sensor for “turn and push” encoding

Hardware Overview

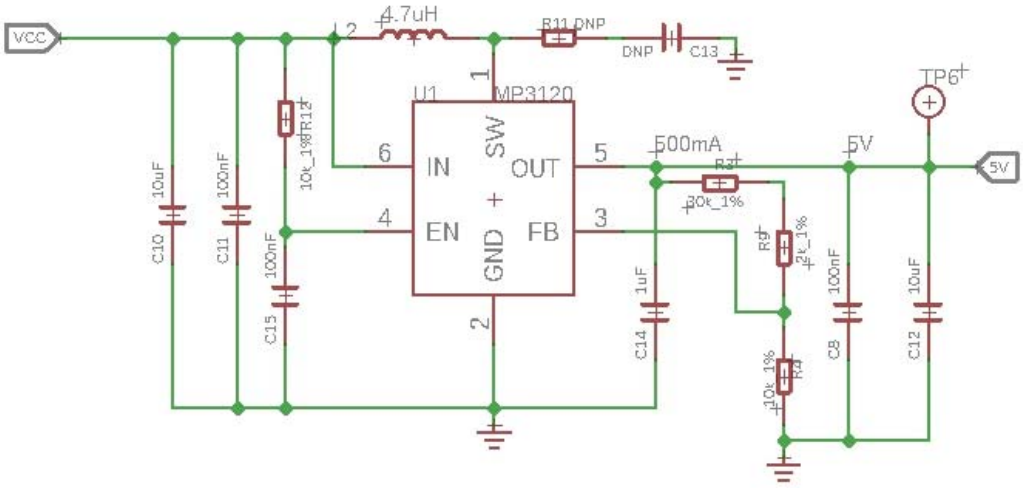
Pin Map



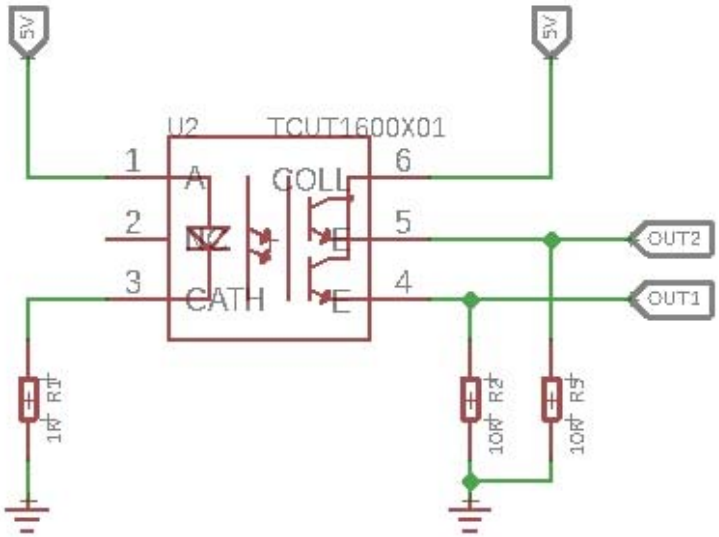
4 GND: connect this module to the system GND
3 VCC: you can use 5V or 3.3V for this module
2 SIG2: default High, output of channel 2, which connect to phototransistor detector 2
1 SIG1: default High, output of channel 1, which connect to phototransistor detector 1

Schemaitc

Power

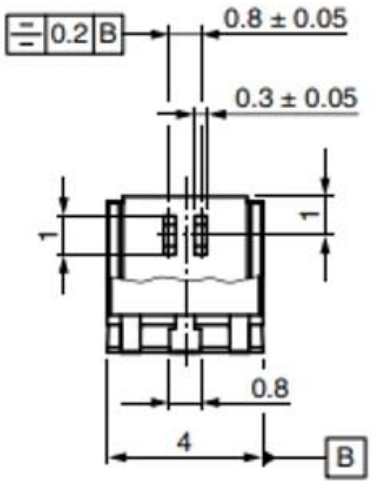


The typical voltage of TCUT1600X01 is 5V, so we use the MP3120 current mode step-up converter to provide a stable 5V. The input of MP3120 ranges from 0.8V to 5V, so you can use this module with your Arduino both in 3.3V and 5V.

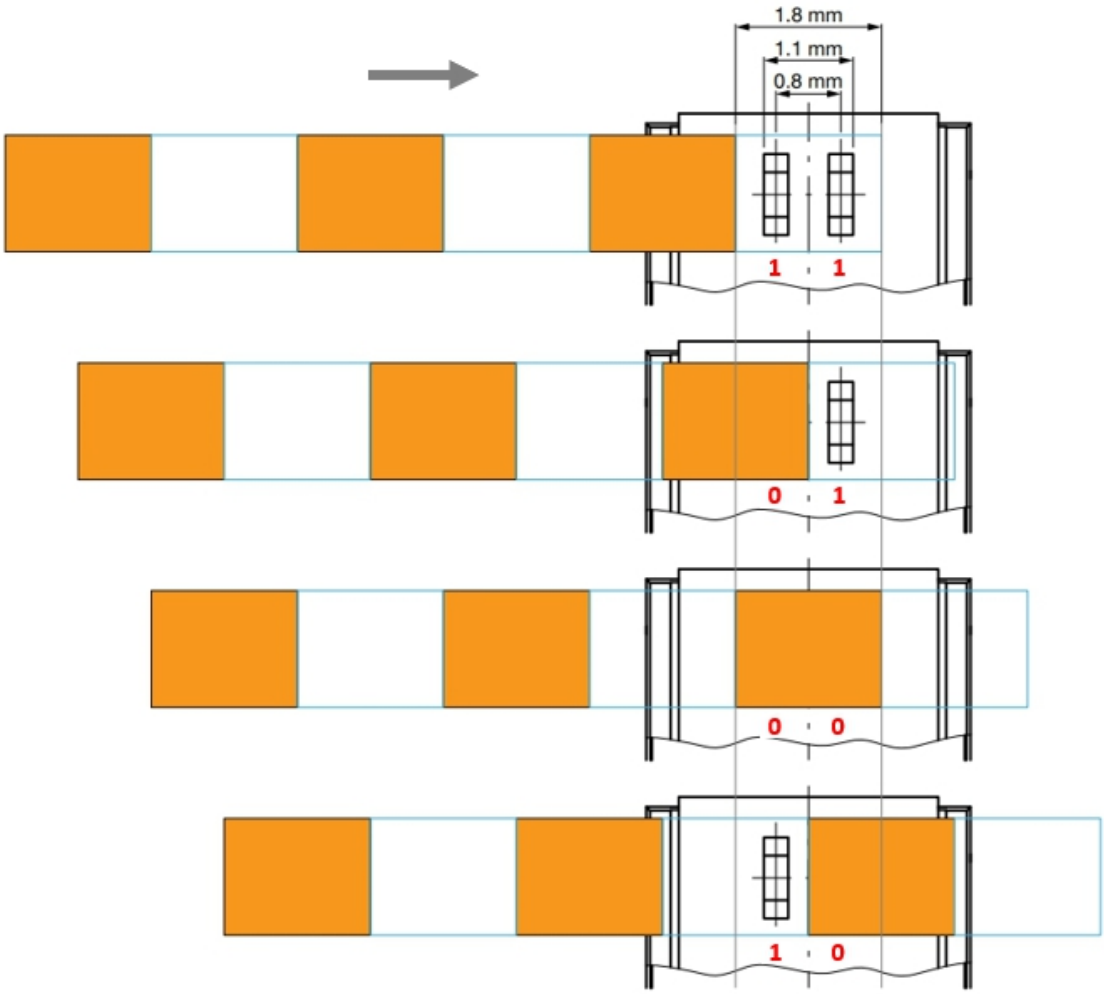


When the phototransistor detectors receive the infrared signal, the output should be High, and when the obstacle blocks the infrared, the OUT1 and OUIT2 should be Low. However due to the leakage current, it won't be 0V. The leakage voltage varies with the input voltage.

Mechanical Drawing



Directional Detection



Tip
Thanks to the two phototransistor detectors, we can detect the moving direction. If the obstacle moves from the left to right, The output states change should be **11** → **01** → **00** → **10**; in the same way, if the obstacle moves from the right to left, it should be **11** → **10** → **00** →**01**.

Platforms Supported

Arduino	Raspberry Pi	BeagleBone	Wio	LinkIt ONE
				

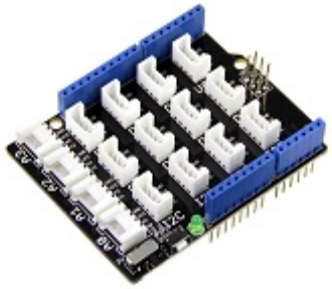
Caution
The platforms mentioned above as supported is/are an indication of the module's hardware or theoritical compatibility. We only provide software library or code examples for Arduino platform in most cases. It is not possible to provide software library / demo code for all possible MCU platforms. Hence, users have to write their own software library.

Getting Started

Play With Arduino

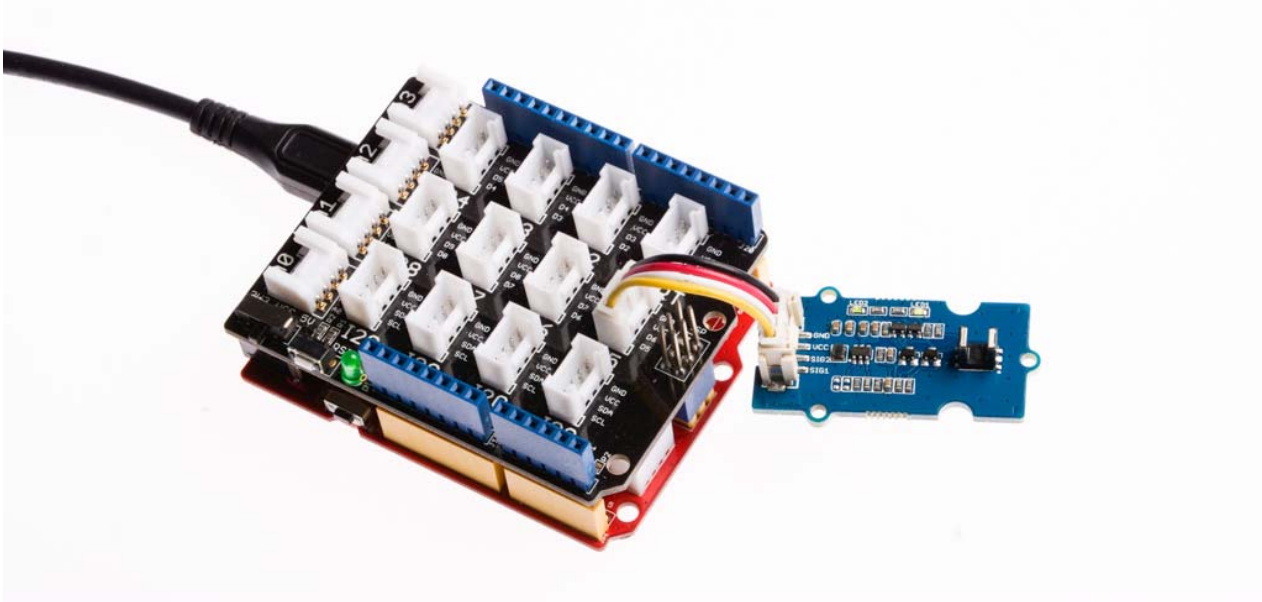
Hardware

Materials required

Seeeduino V4.2	Base Shield	Grove - Optical Rotary Encoder
		

Note
1 Please plug the USB cable gently, otherwise you may damage the port. Please use the USB cable with 4 wires inside, the 2 wires cable can't transfer data. If you are not sure about the wire you have, you can click here to buy
2 Each Grove module comes with a Grove cable when you buy. In case you lose the Grove cable, you can click here to buy.

- **Step 1.** Connect the Grove - Optical Rotary Encoder to the **D5** port of the Base Shield.
- **Step 2.** Plug Grove - Base Shield into Seeeduino.
- **Step 3.** Connect Seeeduino to PC via a USB cable.



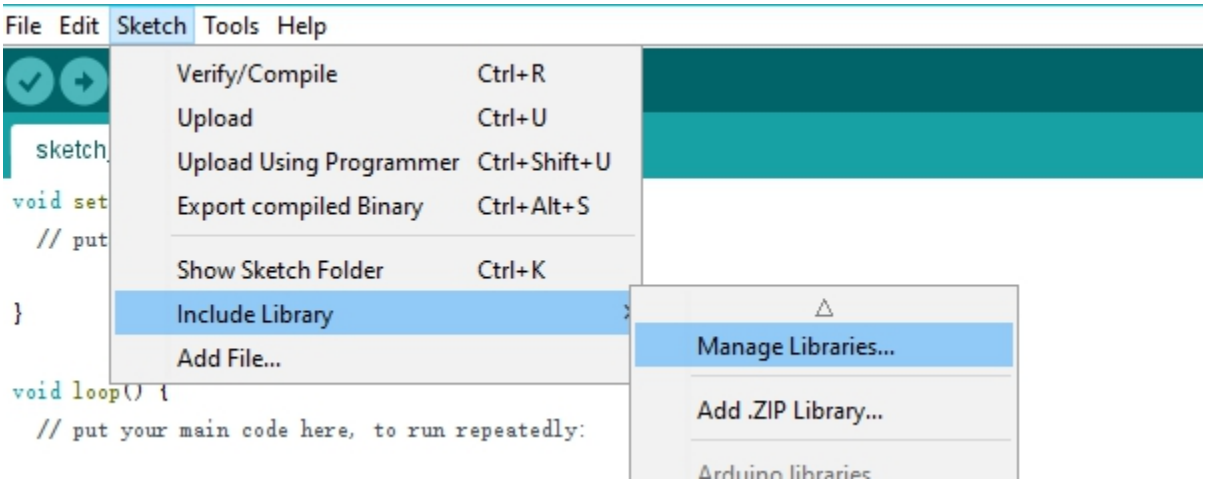
Note
If we don't have Grove Base Shield, We also can directly connect this module to Seeeduino as below.

Seeeduino	Grove - Optical Rotary Encoder
5V	Red
GND	Black
D6	White
D5	Yellow

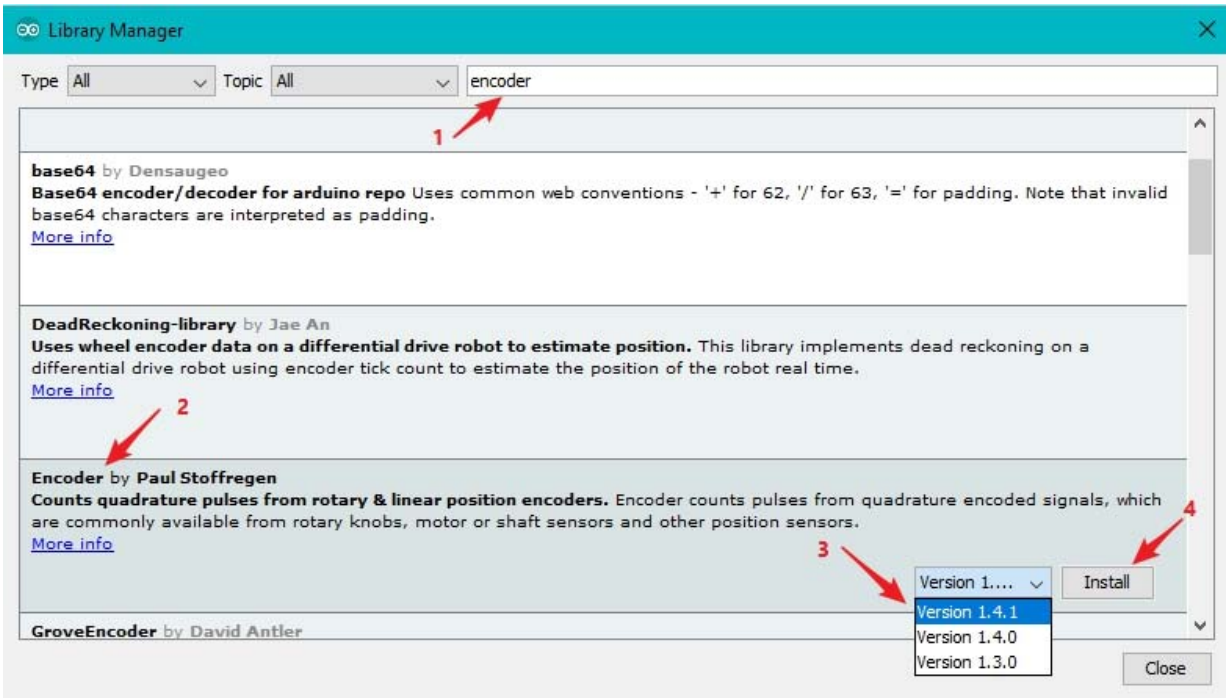
Software

Note
If this is the first time you work with Arduino, we strongly recommend you to see Getting Started with Arduinobefore the start.

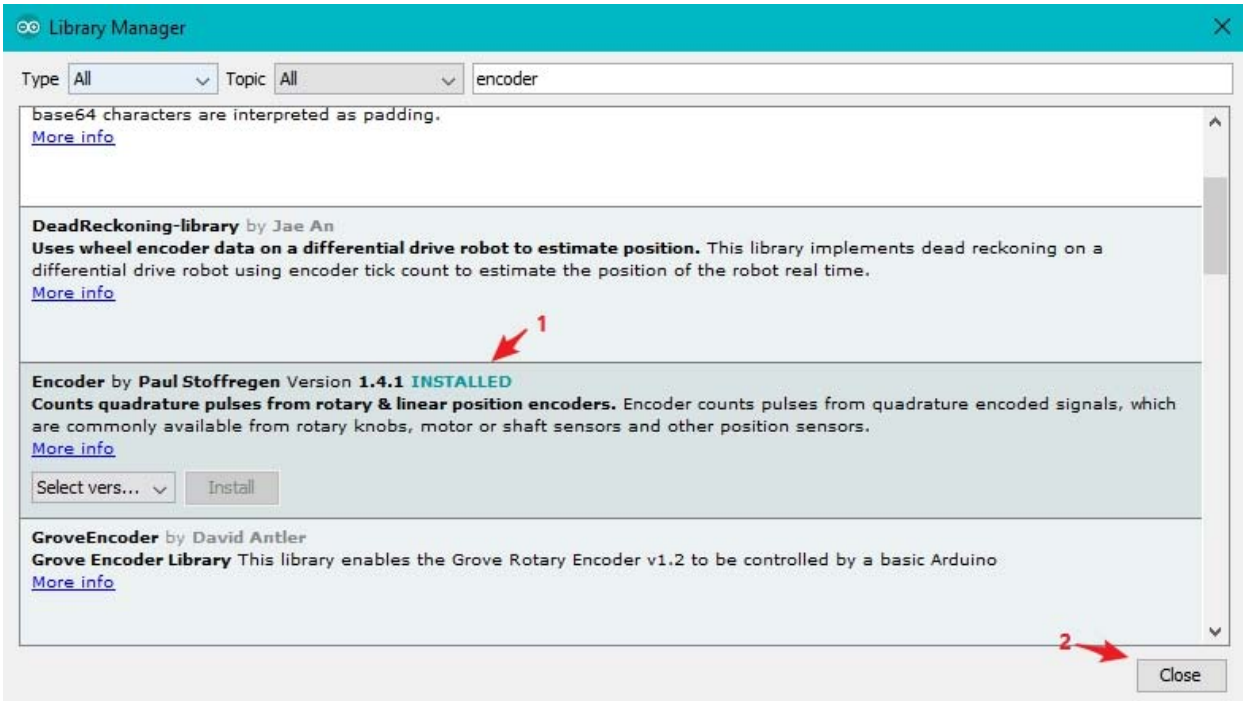
- **Step 1.** Install the **Encoder Library** in the Arduino IDE. You can find this library by the following path: **Sketch→Include Library→Manage Libraries**



Then search for the **encoder** in the pop-up window. Find the **Encoder by Paul Stoffregen**, choose the **Version1.4.1**, then click **Install**.

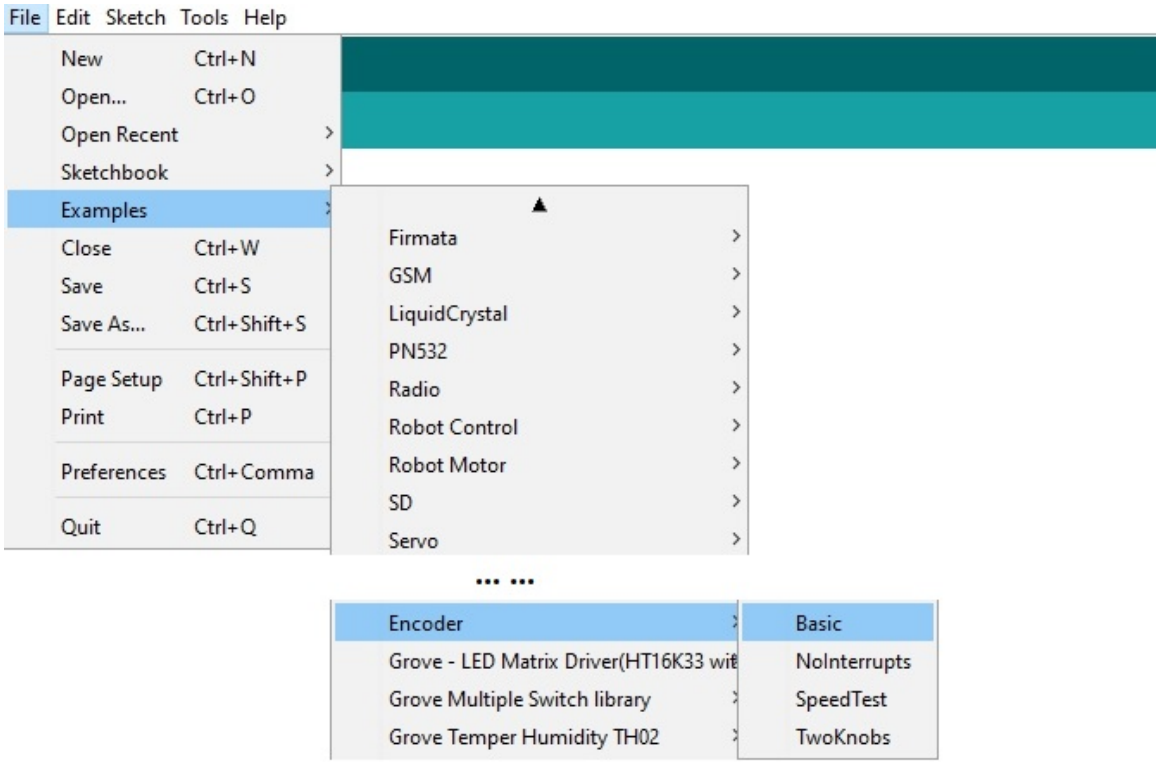


When the library is installed you will see **INSTALLED**, click **Close** then.

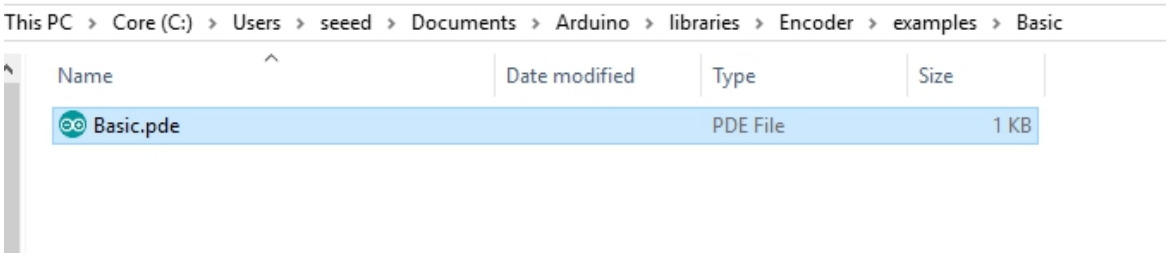


Thanks for Paul for his splendid library.

- **Step 2.** Restart the Arduino IDE. Open the example, you can open it in the following three ways：
  a. Open it directly in the Arduino IDE via the path: **File → Examples → Encoder → Basic**.



b. Open it in your computer by click the **Basic.pde** which you can find in the **xxxx\Arduino\libraries\Encoder\examples\Basic**, **XXXX** is the location you installed the Arduino IDE.



c. Or, you can just click the icon in upper right corner of the code block to copy the following code into a new sketch in the Arduino IDE.

```
/* Encoder Library - Basic Example
 * http://www.pjrc.com/teensy/td_libs_Encoder.html
 *
 * This example code is in the public domain.
 */

#include <Encoder.h>

// Change these two numbers to the pins connected to your
encoder.
//   Best Performance: both pins have interrupt capability
//   Good Performance: only the first pin has interrupt capability
//   Low Performance:  neither pin has interrupt capability
Encoder myEnc(5, 6);
//   avoid using pins with LEDs attached

void setup() {
  Serial.begin(9600);
  Serial.println("Basic Encoder Test:");
}

long oldPosition  = -999;

void loop() {
  long newPosition = myEnc.read();
  if (newPosition != oldPosition) {
    oldPosition = newPosition;
    Serial.println(newPosition);
  }
}
```

Tip

You can change two numbers to the pins connected to your encoder, for the Best Performance: both pins have interrupt capability, so you can change the code line 13 into Encoder myEnc(2, 3);, meanwhile, you should connect this sensor to the **D2** of the baseshield.

- **Step 4.** Upload the demo. If you do not know how to upload the code, please check How to upload code.
- **Step 5.** Open the **Serial Monitor** of Arduino IDE by click **Tool-> Serial Monitor**. Or tap the Ctrl + Shift + M key at the same time. Set the baud rate to **9600**.

Success
If every thing goes well, you will get the result. When you move the obstacle from left to right, the count value will increase by 1; when you move the obstacle from right to left, the count value will be decremented by 1.

```
Basic Encoder Test:
0
1
2
3
4
3
2
1
0
-1
-2
-3
-4
```