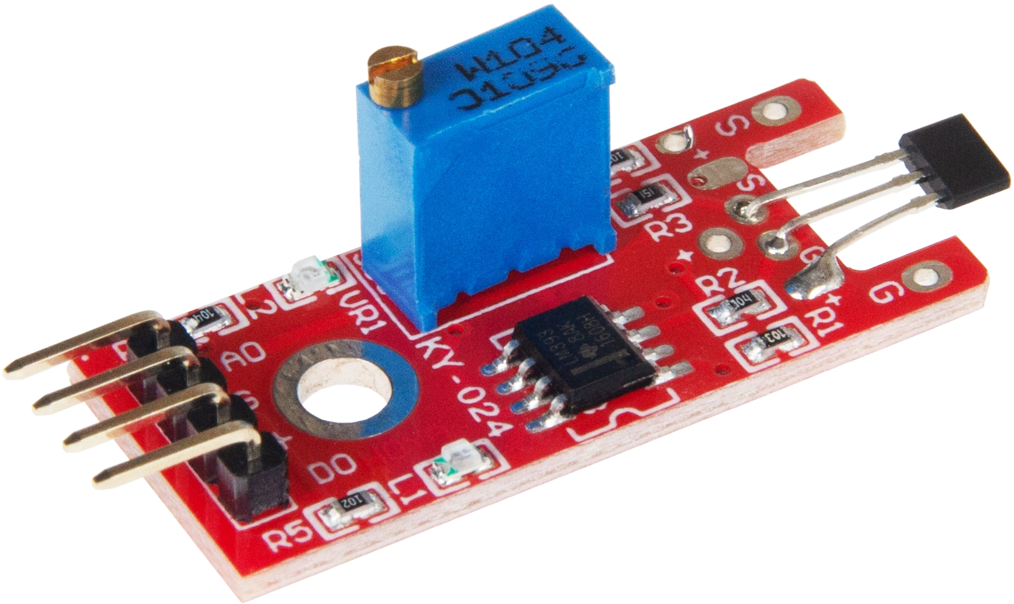
HALL MAGNETIC FIELD SENSOR

SEN-KY024LM



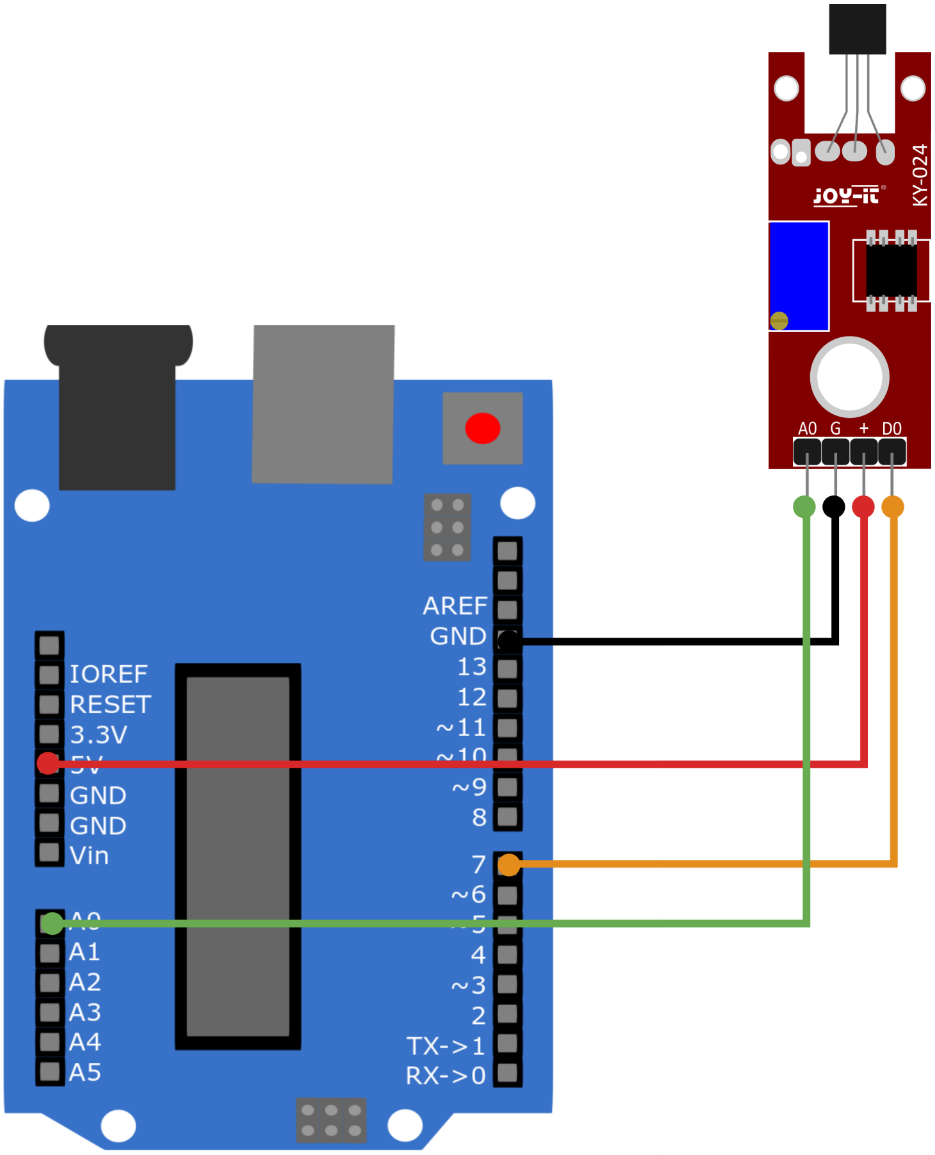
1. GENERAL INFORMATION

Dear customer,
thank you very much for choosing our product.
In following, we will introduce you to what to observe while starting up and using this product.
Should you encounter any unexpected problems during use, please do not hesitate to contact us.

2. USAGE WITH THE ARDUINO

Wiring

Connect your sensor to your Arduino as shown in the diagram and table below.



Arduino	KY-024LM
Pin A0	A0
GND	G
5 V	+
Pin 7	D0

Copy the following code example and upload it to your Arduino. Make sure that you have selected the correct board and the correct port in your Arduino IDE.
The programme will show you the current analog value of the sensor in the serial monitor and whether the limit has been reached.

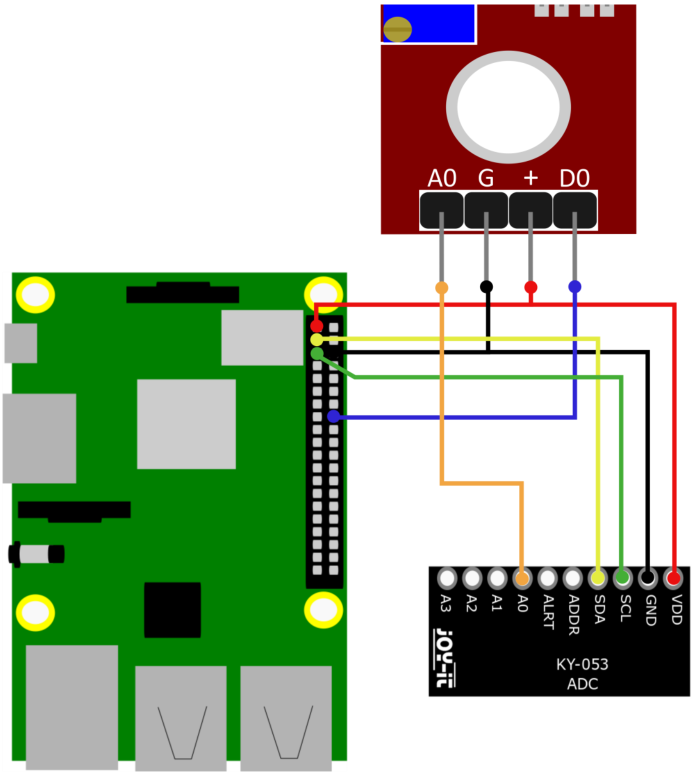
```
// Declaration and initialisation of the input pins
int Analog_Eingang = A0; // Analog-Pin
int Digital_Eingang = 7; // Digital-Pin

void setup ()
{
  pinMode (Analog_Eingang, INPUT);
  pinMode (Digital_Eingang, INPUT);

  Serial.begin (9600); // Serial output mit 9600 bps
}

void loop ()
{
  float Analog;
  int Digital;

  //Current values are read, converted to the voltage value...
  Analog = analogRead (Analog_Eingang) * (5.0 / 1023.0);
  Digital = digitalRead (Digital_Eingang);

  //... and printed at this point
  Serial.print ("Analog Voltage:");
  Serial.print (Analog, 4);
  Serial.print ("V, ");
  Serial.print ("Limit:");

  if(Digital==1)
  {
      Serial.println (" reached");
  }
  else
  {
      Serial.println (" not reached");
  }
  Serial.println ("-------------------------------");
  delay (1000);
}
```

3. USAGE WITH THE RASPBERRY PI

This guide was written under Raspberry Pi OS Bookworm for the Raspberry Pi 4 and 5. It has not been checked with newer operating systems or hardware.

Wiring

Since the Raspberry Pi has no analog inputs, we have to use an external analog / digital converter. In this example we use the Com-KY053ADC from Joy-it.
Connect your sensor to your Raspberry Pi as shown in the diagram and table below.



Raspberry Pi	KY-024LM	ADC
3,3 V	+	VDD
GND	G	GND
Pin 16	D0	
	A0	A0
SCL (Pin 5)		SCL
SDA (Pin 3)		SDA

To be able to use the ADC with the Raspberry Pi, you must first activate I2C. To do this, enter the following command in the console:

```
sudo raspi-config
```

Now go to **3 Interface-Options** -> **I4 I2C** and enable the I2C interface.

Now install pip3 with the following command:

```
sudo apt install python3-pip
```

The next step is to set up the virtual environment. To do this, enter the following commands:

```
mkdir project-name && cd project-name
```

```
python -m venv --system-site-packages env
```

```
source env/bin/activate
```

We use the Adafruit_CircuitPython_ADS1x15 library for our code example, which was published under the MIT license. Use the following commands to download and install this library.

```
pip3 install adafruit-circuitpython-ads1x15
```

Now you can download our prepared sample code:

```
wget https://joy-it.net/files/files/Produkte/SEN-KY024LM/SEN-KY024LM.zip
```

```
unzip SEN-KY024LM.zip && rm SEN-KY024LM.zip
```

Now you can execute the code example with the following command:

```
python3 SEN-KY024LM.py
```

Alternatively, you can manually insert the code example on the next page into your Python file.

```python
# -*- coding: utf-8 -*-
import time
from gpiozero import DigitalInputDevice
import board
import busio
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn

# Create the I2C bus
i2c = busio.I2C(board.SCL, board.SDA)

# Create the ADC object using the I2C bus
ads = ADS.ADS1115(i2c)

# Create single-ended input on channels
chan0 = AnalogIn(ads, ADS.P0)

delayTime = 1      # Time between measurements
Digital_PIN = 23   # Digital Pin (BCM)

# Using gpiozero for digital inputs
sensor = DigitalInputDevice(Digital_PIN, pull_up=False)

while True:

    voltage = chan0.voltage  # Conversion to voltage

    # Output
    if sensor.value == 0:  # Check sensor value
        print("Voltage: {:.3f} V, Limit: not reached".format(voltage))
    else:
        print("Voltage: {:.3f} V, Limit: reached".format(voltage))
    print("---------------------------------------")

    time.sleep(delayTime)
```

4. ADDITIONAL INFORMATION

Our information and take-back obligations according to the Electrical and Electronic Equipment Act (ElektroG)

Symbol on electrical and electronic equipment:

This crossed-out dustbin means that electrical and electronic appliances do not belong in the household waste. You must return the old appliances to a collection point.
Before handing over waste batteries and accumulators that are not enclosed by waste equipment must be separated from it.

Return options:
As an end user, you can return your old device (which essentially fulfils the same function as the new device purchased from us) free of charge for disposal when you purchase a new device.
Small appliances with no external dimensions greater than 25 cm can be disposed of in normal household quantities independently of the purchase of a new appliance.

Possibility of return at our company location during opening hours:
SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn, Germany

Possibility of return in your area:
We will send you a parcel stamp with which you can return the device to us free of charge. Please contact us by email at Service@joy-it.net or by telephone.

Information on packaging:
If you do not have suitable packaging material or do not wish to use your own, please contact us and we will send you suitable packaging.

5. SUPPORT

If there are still any issues pending or problems arising after your purchase, we will support you by e-mail, telephone and with our ticket support system.

Email: **service@joy-it.net**

Ticket system: https://support.joy-it.net

Telephone: +49 (0)2845 9360-50 (Mon - Thur: 09:00 - 17:00 o‘clock CET,
Fri: 09:00 - 14:30 o‘clock CET)

For further information please visit our website:

www.joy-it.net

Published: 2024.04.24