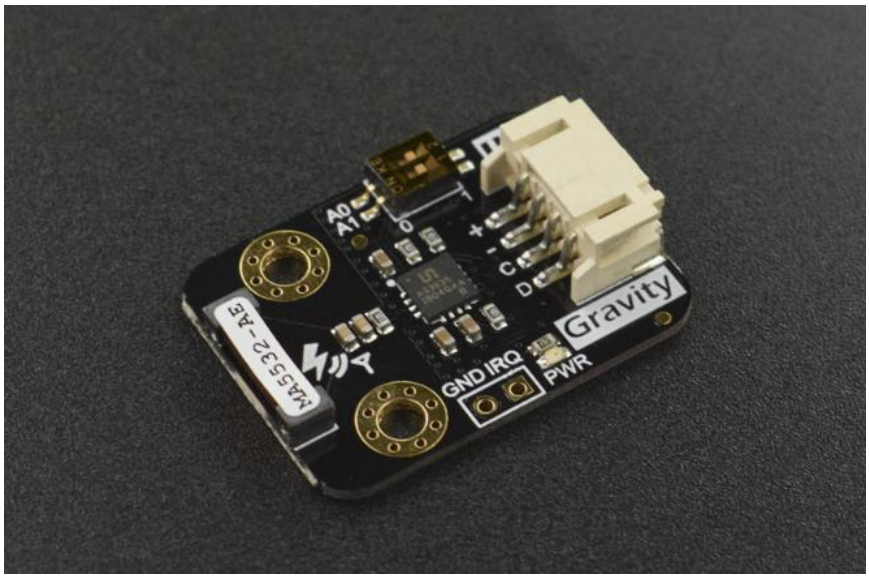


Gravity: Lightning Distance Sensor

SKU:SEN0290

Gravity: Lightning Sensor employs AMS innovative AS3935 Franklin lightning sensor IC and Coilcraft MA5532-AE dedicated antenna to detect lightning distance, intensity, and frequency within a radius of 40km both indoors or outdoors.

The embedded man-made disturber rejection algorithm can effectively avoid the electrical interference generated by various household appliances. In addition to allowing general weather enthusiasts to measure local thunderstorm data simply and efficiently, thanks to its compact size and wide detection range, it can also be embedded in various smart wearable devices for outdoor climber or people who work at heights. This provides early warning of thunderstorms that people can perceive, so that people can take precautions a step ahead. The sensor can also be embedded in the indoor protection device inside lightning-sensitive equipment, and automatically trigger these devices to switch to the backup power to isolate the power grid when lightning strikes. At the moment of lightning, the interrupt pin IRQ generates a pulse. This can be used to trigger the shutter to open, helping photographers to accurately capture the exciting moment of lightning.

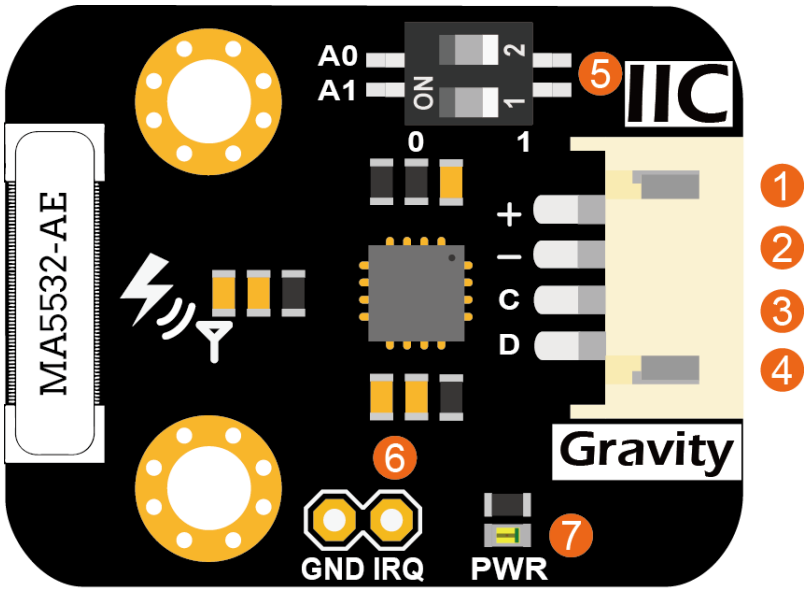
Features

- Lightning detection within 40km in 15 steps
- Lighting intensity detection
- Used both indoors or outdoors
- Embedded man-made disturber rejection algorithm
- Embedded man-made disturber rejection algorithm

Specification

- Input Voltage : 3.3V~5.5V
    - Maximum Detection Range: 40 km
    - Distance Detection Resolution: 1~4 km
    - Maximum Detection Range: 40 km
    - Intensity Detection Resolution: 21 bits, i.e. 0 ~16777201
    - I2C Address: Three options 0x03, 0x02, 0x01
    - Interface: Gravity I2C (logic level: 0-VCC)
- Dimension: 30.0mm*22.0mm
- Weight: 10g

Board Overview



No.	Label	Description
1	VCC	Power VCC（3.3~5.5V）
2	GND	Power GND
3	SCL	I2C clock signal
4	SDA	I2C data signal
5	ADDR	I2C address DIP switch
6	IRQ	Lightning alarm interrupt pin
7	PWR	Power indicator (red)

About Lightning Distance and Intensity Estimation

- The maximum estimated distance of lightning strike is 40km. Limited by the inherent measurement method and algorithm, the distance estimation resolution is 1~4km, 40 km in 15 steps. This is shown in the following table:

No.	Distance (km)
1	40
2	37
3	34
4	31

No.	Distance (km)
5	27
6	24
7	20
8	17
9	14
10	12
11	10
12	8
13	6
14	5
15	Storm is Overhead !

- The estimated lightning intensity is 21 bits, i.e. 0 ~ 16777201. This value is just a pure number and has no physical meaning, but it can be used to compare the relative strength between different lightnings.

Arduino Tutorial

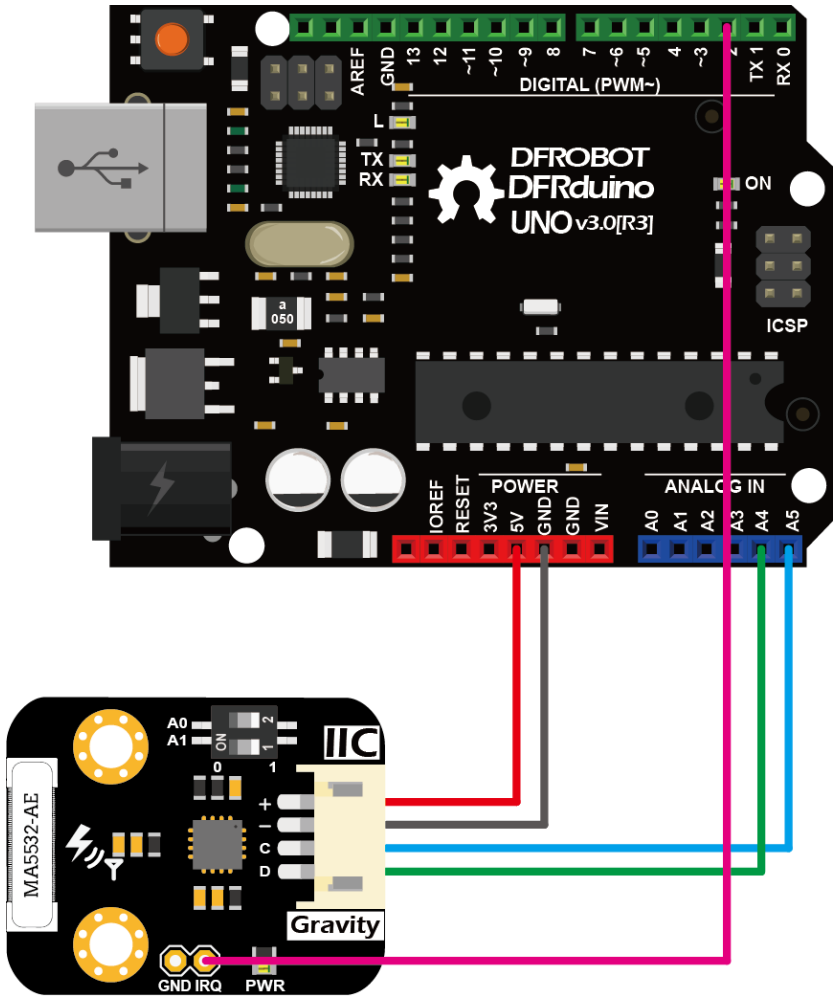
This tutorial presents a basic usage of the module with Arduino UNO.

Requirements

- **Hardware**
  - DFRduino UNO R3 (or similar) x 1
  - DFRobot Gravity: Lightning Sensor x 1
  - Gravity 4P sensor wire (or Dupont wires) x 1

- **Software**
  - Arduino IDE
  - Download and install the **DFRobot_AS3935 Library** (About how to install the library?)

Connection Diagram



Sample Code

- Connect the module to the Arduino according to the connection diagram. The I2C address defaults to 0x03, which corresponds to “AS3935_ADD3” in the code. If you need to modify the I2C address, you can change it to 0x01 or 0x02 through the DIP switch on the module, and modify the macro definition of the I2C address in the software "#define AS3935_I2C_ADDR AS3935_ADDx", where x can be 1, 2, 3.
- Install DFRobot_AS3935 library.
- Open Arduino IDE, upload the following sample code to the Arduino UNO.
- Open the serial monitor of the Arduino IDE and set the baud rate to 115200.

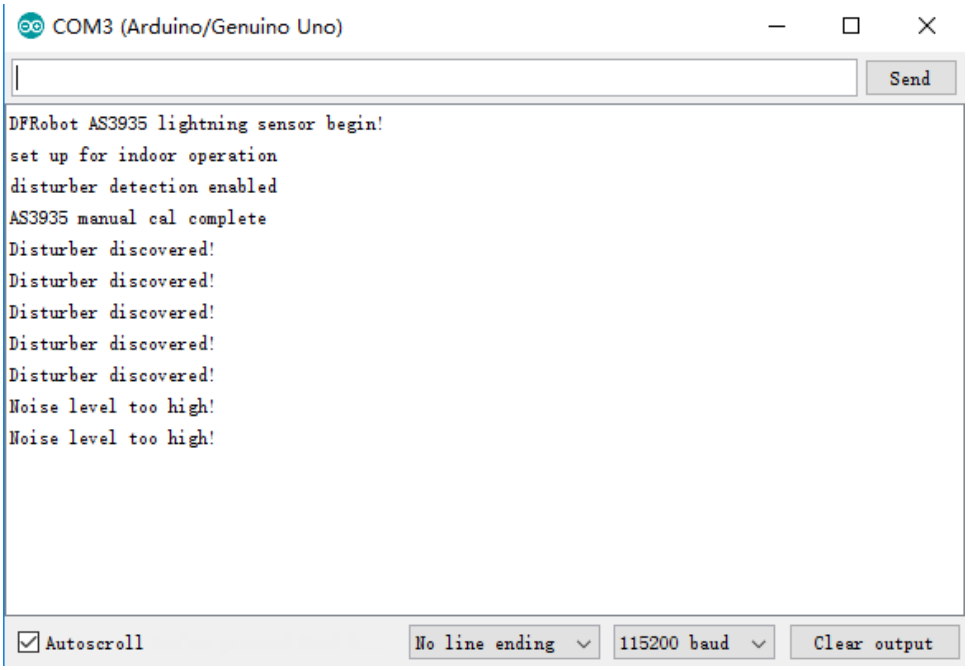
```
/*!
   file DFRobot_AS3935_lightning_I2c.ino

   SEN0290 Lightning Sensor
   This sensor can detect lightning and display the distance and intensity of the
lightning within 40 km
   It can be set as indoor or outdoor mode.
   The module has three I2C, these addresses are:
   AS3935_ADD1  0x01   A0 = High  A1 = Low
   AS3935_ADD2  0x02   A0 = Low   A1 = High
   AS3935_ADD3  0x03   A0 = High  A1 = High

   Copyright    [DFRobot](http://www.dfrobot.com), 2018
   Copyright    GNU Lesser General Public License

   version  V0.2
   date  2018-10-08
*/

#include "I2C.h"
#include "DFRobot_AS3935_I2C.h"

volatile int8_t AS3935_ISR_Trig = 0;

#define IRQ_PIN             2

// Antenna tuning capcitance (must be integer multiple of 8, 8 - 120 pf)
#define AS3935_CAPACITANCE   96

// Indoor/outdoor mode selection
#define AS3935_INDOORS       0
#define AS3935_OUTDOORS      1
#define AS3935_MODE          AS3935_INDOORS

// Enable/disable disturber detection
#define AS3935_DIST_DIS      0
#define AS3935_DIST_EN       1
#define AS3935_DIST          AS3935_DIST_EN
```

```
// I2C address
#define AS3935_I2C_ADDR       AS3935_ADD3

void AS3935_ISR();

DF_AS3935_I2C  lightning0((uint8_t)IRQ_PIN, (uint8_t)AS3935_I2C_ADDR);

void setup()
{

  Serial.begin(115200);
  Serial.println("DFRobot AS3935 lightning sensor begin!");

  // Setup for the the I2C library: (enable pullups, set speed to 400kHz)
  I2c.begin();
  I2c.pullup(true);
  I2c.setSpeed(1);
  delay(2);

  // Set registers to default
  lightning0.AS3935_DefInit();
  // Configure sensor
  lightning0.AS3935_ManualCal(AS3935_CAPACITANCE, AS3935_MODE, AS3935_DIST);
  // Enable interrupt (connect IRQ pin IRQ_PIN: 2, default)

  //  Connect the IRQ and GND pin to the oscilloscope.
  //  uncomment the following sentences to fine tune the antenna for better
performance.
  //  This will dispaly the antenna's resonance frequency/16 on IRQ pin (The
resonance frequency will be divided by 16 on this pin)
  //  Tuning AS3935_CAPACITANCE to make the frequency within 500/16 kHz ± 3.5%
  //  lightning0.AS3935_SetLCO_FDIV(0);
  //  lightning0.AS3935_SetIRQ_Output_Source(3);

  attachInterrupt(0, AS3935_ISR, RISING);

}

void loop()
{
  // It does nothing until an interrupt is detected on the IRQ pin.
  while (AS3935_ISR_Trig == 0) {}
  delay(5);

  // Reset interrupt flag
  AS3935_ISR_Trig = 0;

  // Get interrupt source
  uint8_t int_src = lightning0.AS3935_GetInterruptSrc();
  if (int_src == 1)
  {
    // Get rid of non-distance data
    uint8_t lightning_dist_km = lightning0.AS3935_GetLightningDistKm();

    if (lightning_dist_km == 1) {
```

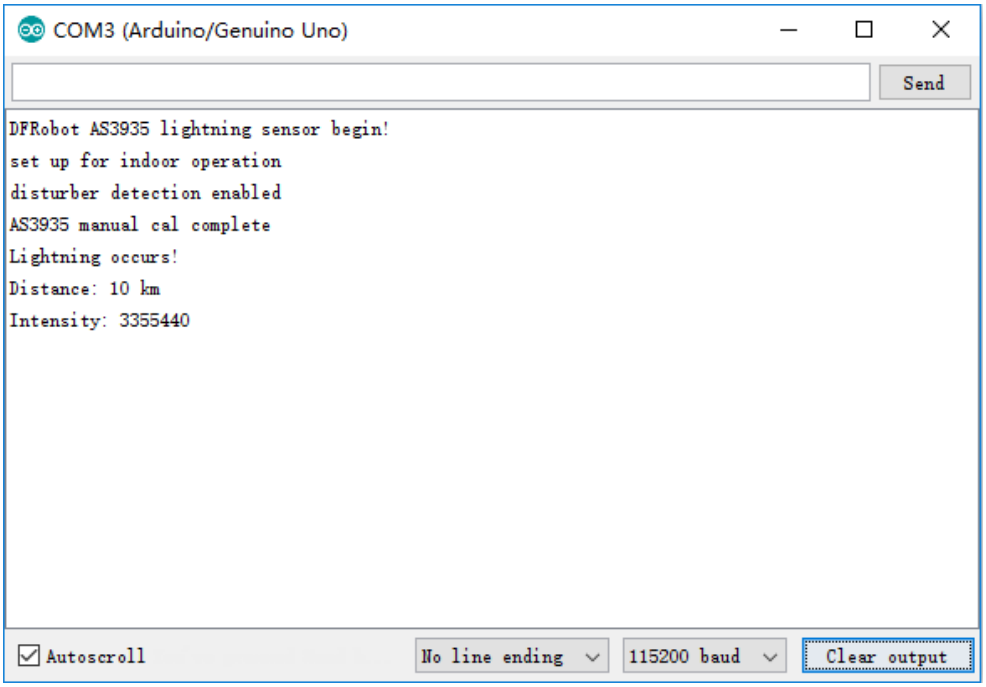
```
      Serial.println("Lightning overhead!");
    }
    else if ((lightning_dist_km >= 5) && (lightning_dist_km <= 40)) {
      Serial.println("Lightning occurs!");
      Serial.print("Distance: ");
      Serial.print(lightning_dist_km);
      Serial.println(" km");
    }
    else if (lightning_dist_km == 0x3F){
      Serial.println("Out of range.");
    }

    // Get lightning energy intensity
    uint32_t lightning_energy_val = lightning0.AS3935_GetStrikeEnergyRaw();
    Serial.print("Intensity: ");
    Serial.print(lightning_energy_val);
    Serial.println("");
  }
  else if (int_src == 2)
  {
    Serial.println("Disturber discovered!");
  }
  else if (int_src == 3)
  {
    Serial.println("Noise level too high!");
  }

}

//IRQ handler for AS3935 interrupts
void AS3935_ISR()
{
  AS3935_ISR_Trig = 1;
}
```

Results

- When lightning occurs, the interrupt alarm pin IRQ triggers the controller to display the lightning alarm "Lightning occurs!", and shows the estimated lightning distance and intensity.



- When there is electromagnetic interference near the module, the module will issue an alarm “Disturber discovered!”. If the interference noise is too large, the module will issue the alarm “Noise level too high!” .



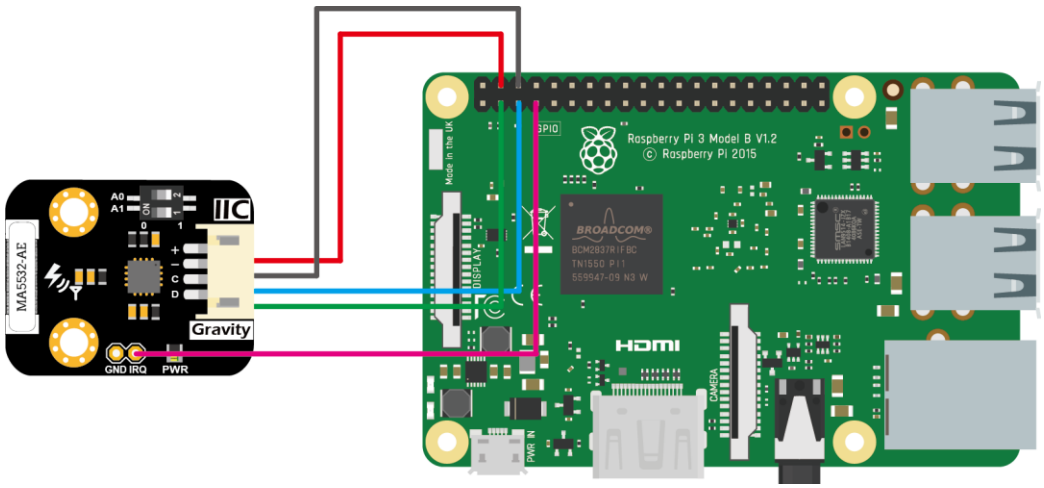
Raspberry Pi Tutorial

This tutorial presents a basic usage of the module with Raspberry Pi.

Requirements

- **Hardware**
  - Raspberry Pi 3 Model B (or similar) x 1
  - DFRobot Gravity: Lightning Sensor x 1
  - Gravity 4P sensor wire (or Dupont wires) x 1
- **Software**
  - RASPBIAN
  - Download and install the **DFRobot_AS3935 RaspberryPi library**.

Connection Diagram



Installation

- Start the I2C interface of the Raspberry Pi. If it is already open, skip this step. Open Terminal, type the following command, and press Enter:

```
pi@raspberrypi:~ $ sudo raspi-config
```

- Then use the up and down keys to select "5 Interfacing Options" -> "P5 I2C" and press Enter to confirm "YES". Reboot the Raspberry Pi.

- Installing Python libraries and git (networking required). If it is already installed, skip this step. In the Terminal, type the following commands, and press Enter:

```
pi@raspberrypi:~ $ sudo apt-get update
```

```
pi@raspberrypi:~ $ sudo apt-get install build-essential python-dev python-smbus git
```

- Download the driver library and run it. In Terminal, type the following commands, and press Enter:

```
pi@raspberrypi:~ $ git clone https://github.com/DFRobot/DFRobot_AS3935.git
pi@raspberrypi:~ $ cd ~/DFRobot_AS3935/RaspberryPi/python
```

```
pi@raspberrypi:~/DFRobot_AS3935/RaspberryPi/python $ python DFRobot_AS3935.py
```

Run Sample Code

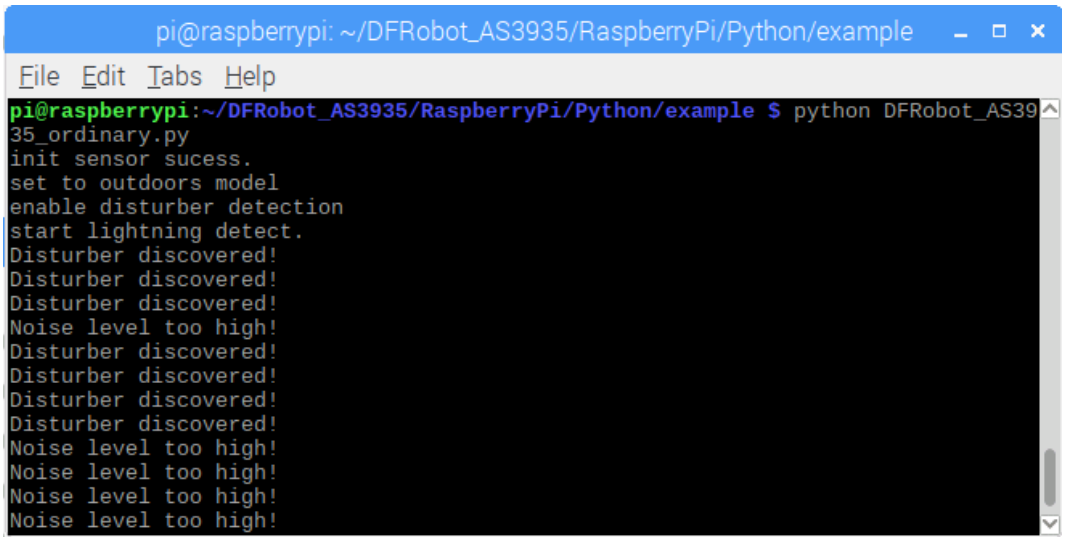
- Connect the module to the Arduino according to the connection diagram. The I2C address defaults to 0x03, which corresponds to “AS3935_ADD3” in the code. If you need to modify the I2C address, you can change it to 0x01 or 0x02 through the DIP switch on the module, and modify the macro definition of the I2C address in the software "#define AS3935_I2C_ADDR AS3935_ADDx", where x can be 1, 2, 3.
- Install DFRobot_AS3935 Raspberry Pi library.
- In the Terminal, type in the following commands and press Enter to run the sample code:

```
pi@raspberrypi:~/DFRobot_AS3935/RaspberryPi/python $ cd readAndInt
```

```
pi@raspberrypi:~/DFRobot_AS3935/RaspbeeryPi/python/readAndInt $ python readAndInt.py
```

Results

- When lightning occurs, the interrupt alarm pin IRQ triggers the controller to display the lightning alarm "Lightning occurs!", and shows the estimated lightning distance and intensity.
- When there is electromagnetic interference near the module, the module will issue an alarm “Disturber discovered!”. If the interference noise is too large, the module will issue the alarm “Noise level too high!” .

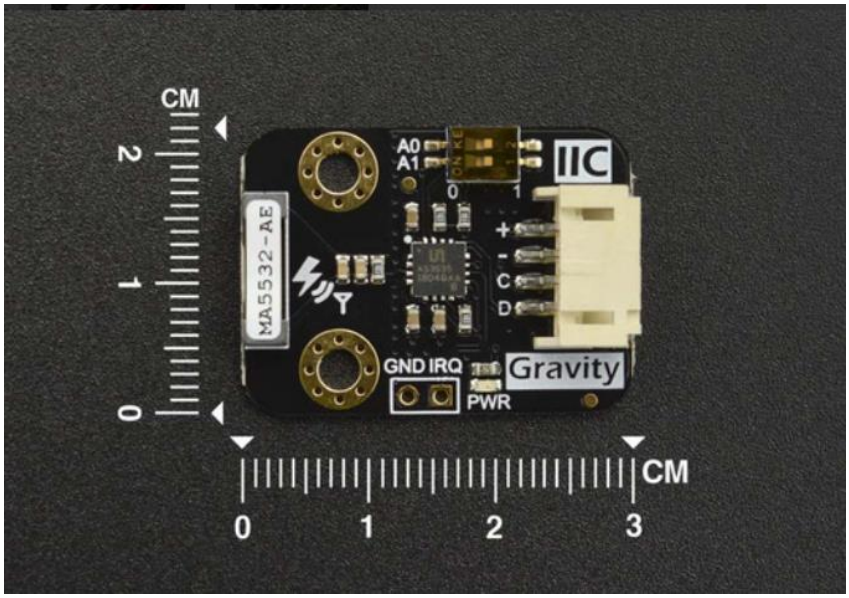


FAQ

For any questions, advice or cool ideas to share, please visit the **DFRobot Forum**.

More Documents

- Schematic
- Layout & Dimension
- AMS AS3935 Datasheet
- Coilcraft MA5532-AE Datasheet



https://www.dfrobot.com/product-1828.html?search=SEN0290/7-11-19