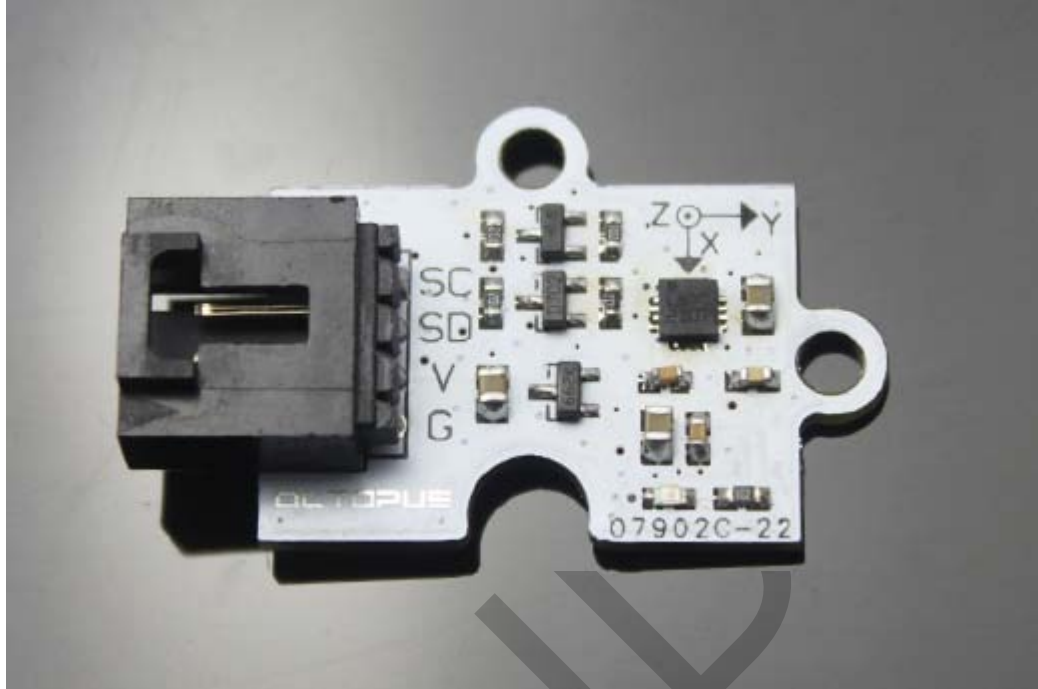


■ 3-Axis Digital Compass Sensor



1. 사양

1.1. Description

이 모듈은 쉽게 각 축에 기울기를 인식하는 센서로 I2C 인터페이스를 지원한다. HMC5883L 이라는 IC를 베이스하고 이 IC는 자기를 이용하여 각 축에 대한 기울기를 인식한다. 이 모듈은 은 주로 핸드폰이나 노트북, 네비게이션 등에 주로 사용된다.

1.2. Feature

- 12-bit ADC and low interference AMR sensor, able to achieve 5 milligauss resolution in ± 8 gauss magnetic field.
- built-in self-test function
- low voltage operation (2.16-3.6V) and low power (100uA)
- built-in drive circuit
- I2C digital interface

3-Axis Digital Compass Sensor

- wide range of magnetic field ($\pm 80\text{Oe}$)
- Power supply needs: 5V
- Simple drive circuit
- Fast response and High sensitivity
- Stable and long life

1.3. 구성

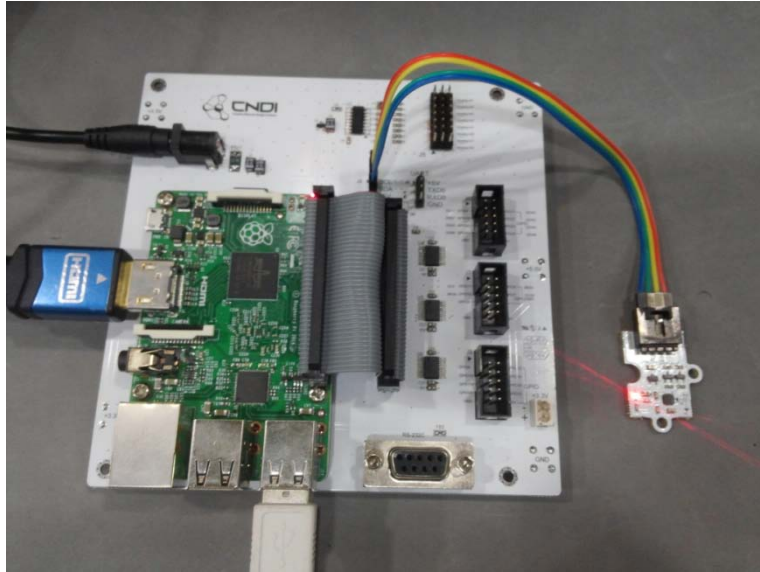
- 1 X 3-Axis Digital Compass Sensor
- 1 x Analog Sensor Cable

2. User Guide

2.1. 결선

모듈의 결선은 아래의 그림과 표를 이용하여 결선한다.

RaspberryPi	Compass Sensor
5V	V
GND	G
SDA	SD
SCL	SC



모듈의 장치 및 회로에 대한 상세한 내용은 데이터시트 및 회로도를 참고한다.

2.2. 예제프로그램

아래 예제 프로그램을 작성 후 컴파일 한 후 실행하면 3-Axis Digital Compass 센서에 입력 값을 디버그 메시지로 표시하는 동작을 한다.

I2C Load 명령 : **gpio load i2c**

컴파일 명령 : **gcc -o 3-Axis Digital Compass Sensor 3-Axis Digital Compass Sensor.c -l wiringPi**

실행 명령 : **./3-Axis Digital Compass Sensor**

- 3-Axis Digital Compass Sensor.c

```
#include <stdio.h>
#include <string.h>
#include <errno.h>
#include <unistd.h>
#include <stdlib.h>
#include <stdint.h>
#include <time.h>
#include <wiringPi.h>
#include <wiringPiI2C.h>
```

3-Axis Digital Compass Sensor

```
#define ADDRESS 0x1E
#define X_MSB_REGISTER 0x03
#define X_LSB_REGISTER 0x04
#define Y_MSB_REGISTER 0x05
#define Y_LSB_REGISTER 0x06
#define Z_MSB_REGISTER 0x07
#define Z_LSB_REGISTER 0x08

int main()
{
    int nCompass;
    if((nCompass = wiringPiI2CSetup(ADDRESS)) == -1)
    {
        printf("Unable to initialise I2C");
        return 1;
    }
    wiringPiI2CWriteReg8(nCompass, 0x02, 0x00);

    int i;
    uint8_t high[3], low[3];
    int result[3];
    while(1)
    {
        printf("nCompass = %d\r\n",nCompass);

        high[0]=wiringPiI2CReadReg8(nCompass,X_MSB_REGISTER);
        low[0]=wiringPiI2CReadReg8(nCompass,X_LSB_REGISTER);

        high[1]=wiringPiI2CReadReg8(nCompass,Y_MSB_REGISTER);
        low[1]=wiringPiI2CReadReg8(nCompass,Y_LSB_REGISTER);

        high[2]=wiringPiI2CReadReg8(nCompass,Z_MSB_REGISTER);
        low[2]=wiringPiI2CReadReg8(nCompass,Z_LSB_REGISTER);
```

3-Axis Digital Compass Sensor

```
        for(i = 0; i<3; i++)
        {
            result[i] = (high[i] <<8) | low[i];
        }

        printf("x = %i, y = %i, z=%i\n" ,result[0],result[1],result[2]);
        delay(1000);
    }

    return 0;
}
```

```
root@raspberrypi:/home/pi/work# gcc -o 3-Axis\ Digital\ Compass\ Sensor 3-Axis\
Digital\ Compass\ Sensor.c -lwiringPi
root@raspberrypi:/home/pi/work# ./3-Axis\ Digital\ Compass\ Sensor
x = 18, y = 65023, z=65170
x = 17, y = 65025, z=65171
x = 18, y = 65026, z=65169
x = 65422, y = 65058, z=65161
x = 65356, y = 65076, z=65163
x = 65527, y = 65027, z=65178
x = 65520, y = 65024, z=65179
x = 65521, y = 65024, z=65179
x = 65519, y = 65026, z=65182
x = 65522, y = 65022, z=65183
x = 65521, y = 65023, z=65182
x = 65525, y = 65021, z=65180
x = 242, y = 65047, z=65183
x = 12, y = 65023, z=65173
x = 65529, y = 65021, z=65180
x = 65526, y = 65025, z=65183
x = 65526, y = 65024, z=65183
```