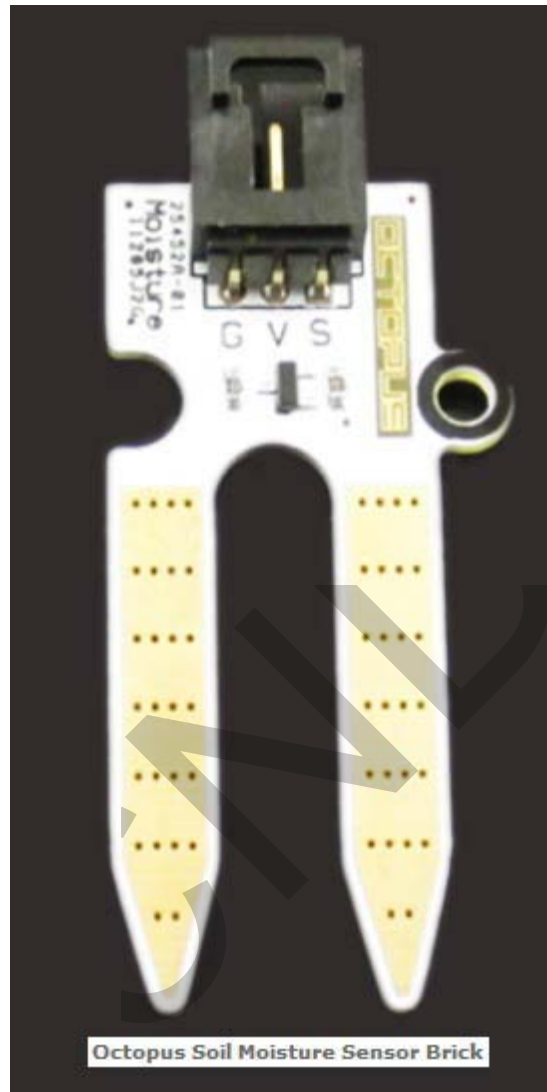


■ Soil Moisture Sensor Brick



1. 사양

1.1. Description

Soil Moisture Sensor Brick 은 토양 환경에서 현재 수분의 양을 읽을 수 있다. 이 센서는 흙 속에 설치하는 2개의 프로브를 사용하며, 수분의 단계를 확인할 수 있는 저항 값을 읽는다. 더 많은 수분은 더 쉽게 전기가 전도될 수 있도록 하여 저항을 줄이고, 마른 토양은 전기가 전도되기 어려워 저항이 커진다.

3.3V와 5V 전원에서 동작하며, 0에서 4.2V의 전압 신호를 출력한다.

Soil Moisture Sensor Brick

1.2. Feature

Power supply: 3.3v or 5v

Output voltage signal: 0~4.2v

Pin definition:

Analog output(Yellow wire)

GND(Black wire)

Power(Red wire)

1.3. 구성

1 x Soil Moisture Sensor

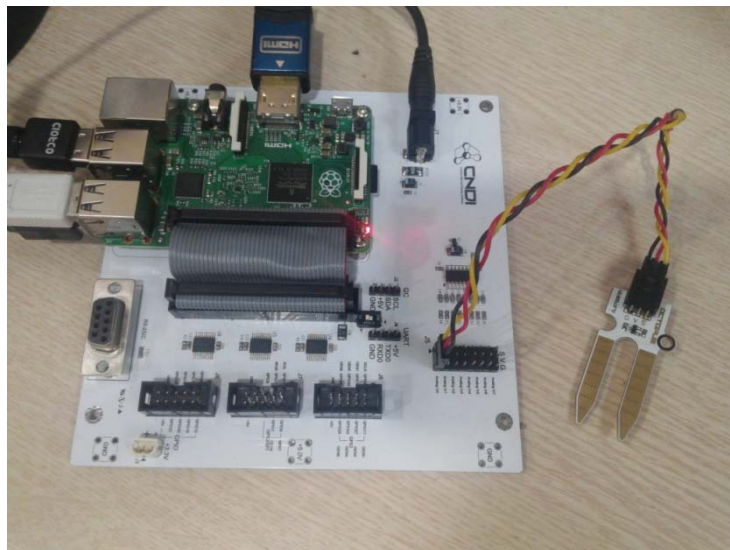
1 x Analog Sensor Cable

2. User Guide

2.1. 결선

모듈의 결선은 아래의 그림과 표를 이용하여 결선한다.

Raspberrypi	Soil Moisture
Gnd	GND(검정)
5V	VCC(빨강)
Analong In 0	SIGNAL(노랑)



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

2.2. 예제프로그램

아래 예제 프로그램을 작성 후 컴파일 한 후 실행하면 Soil Moisture센서에 입력 값을 디버그 메시지로 표시하는 동작을 한다.

컴파일 명령 : **gcc -o Soil Moisture Soil Moisture.c -lwiringPi**

실행 명령 : **./Soil Moisture**

● Soil Moisture.c

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#include <string.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <wiringPi.h>
#include <wiringPiSPI.h>

#define CS_MCP3208      8                //GPIO 8
#define SPI_CHANNEL      0                //SPI Channel
#define SPI_SPEED      1000000          //spi speed

#define MAX_BUFFER 30

unsigned int gunSensorValue;

int ReadMcp3208ADC(unsigned char adcChannel)
{
    unsigned char buff[3];
```

Soil Moisture Sensor Brick

```
int nAdcValue = 0;

digitalWrite(CS_MCP3208,0);           //low cs Active

wiringPiSPIDataRW(SPI_CHANNEL, buff, 3);

//8bit data
buff[1] = 0x0F & buff[1];
nAdcValue = (buff[1]<<8) | buff[2];

//spi chip Select command
digitalWrite(CS_MCP3208, 1);

return nAdcValue;
}

int main(void)
{
    int nAdcChannel = 0;
    int nAdcValue = 0;
    int nServerSockfd, nClientSockfd;
    int nClientLen, n;
    char buff[3];

    buff[0] = 0x06 | ((nAdcChannel & 0x07) >> 2);
    buff[1] = ((nAdcChannel & 0x07)<<6);
    buff[2] = 0x00;

    if(wiringPiSetupGpio() == -1)
    {
        fprintf(stdout,"Not start wiringPi: %s\n",strerror(errno));
        return 1;
    }
}
```

```
//wiringpi init
if(wiringPiSPISetup(SPI_CHANNEL, SPI_SPEED) == -1)
{
    fprintf(stdout, "wiringPiSPISetup Failed: %s\n", strerror(errno));
    return 1;
}

pinMode(CS_MCP3208, OUTPUT);

while(1)
{

    nAdcValue = ReadMcp3208ADC(nAdcChannel);
//sensorRead
    printf("Soil Moisture Sensor Value = %d\n", nAdcValue);
    delay(1000);

}
return 0;
}
```

Soil Moisture Sensor Brick

```
root@raspberrypi:/home/pi/work# gcc -o Soil\ Moisture Soil\ Moisture.c -l wiringPi
root@raspberrypi:/home/pi/work# ./Soil\ Moisture
Soil Moisture Sensor Value = 4095
Soil Moisture Sensor Value = 4095
Soil Moisture Sensor Value = 0
Soil Moisture Sensor Value = 1734
Soil Moisture Sensor Value = 0
Soil Moisture Sensor Value = 568
Soil Moisture Sensor Value = 1024
Soil Moisture Sensor Value = 4011
Soil Moisture Sensor Value = 0
Soil Moisture Sensor Value = 983
Soil Moisture Sensor Value = 4027
Soil Moisture Sensor Value = 1337
Soil Moisture Sensor Value = 0
Soil Moisture Sensor Value = 4011
Soil Moisture Sensor Value = 1024
Soil Moisture Sensor Value = 471
Soil Moisture Sensor Value = 1666
Soil Moisture Sensor Value = 2885
Soil Moisture Sensor Value = 769
```