Water Sensor Experiment

Introduction

Water depth sensor is specially designed for water quality detection. It is extensively used to sense rainfall, water level and even liquid leakage. The module is mainly divided into three parts: electronic connector, 1MΩ resistor and several bare wires.

The working principle of the sensor is to ground a series of exposed copper wires and interweave the sensing wires and ground wires. The sensor wire has a weak pull-up resistor of 1MΩ. The resistor pulls the sensor wire high until a drop of water shorts the sensor wire to the ground wire.

The Water Sensor (water level)is an easy-to-use, compact, lightweight and the high cost performance water level/droplet identification detection sensor. Low power consumption and high sensitivity are another major feature. It can easily complete the conversion of water volume to analog signal and then the output analog value can be directly applied by the function in the program. It can be used in experiments by connecting to the Raspberry Pi controller and then directly inserting the sensor expansion board.

Water Sensor Module

Pin Description

- "S"represent signal input
- "+"represent the power supply
- "-" represent GND

Feature

- Working voltage: 5V
- Working current: <20mA
- Interface: analog
- Detection width: 40mm×16mm
- Detection range: 4cm
- Working temperature: 10℃~30℃
- Weight: 3g
- Size: 65mm×20mm×8mm
- Low power consumption
- High sensitivity
- Output voltage signal: 0~4.2V

Experimental Purpose

- Understand the working principle of the water sensor module and learn to use the Raspberry Pi board to obtain the water level depth value measured by the water sensor and print the value on the computer terminal.

Component List

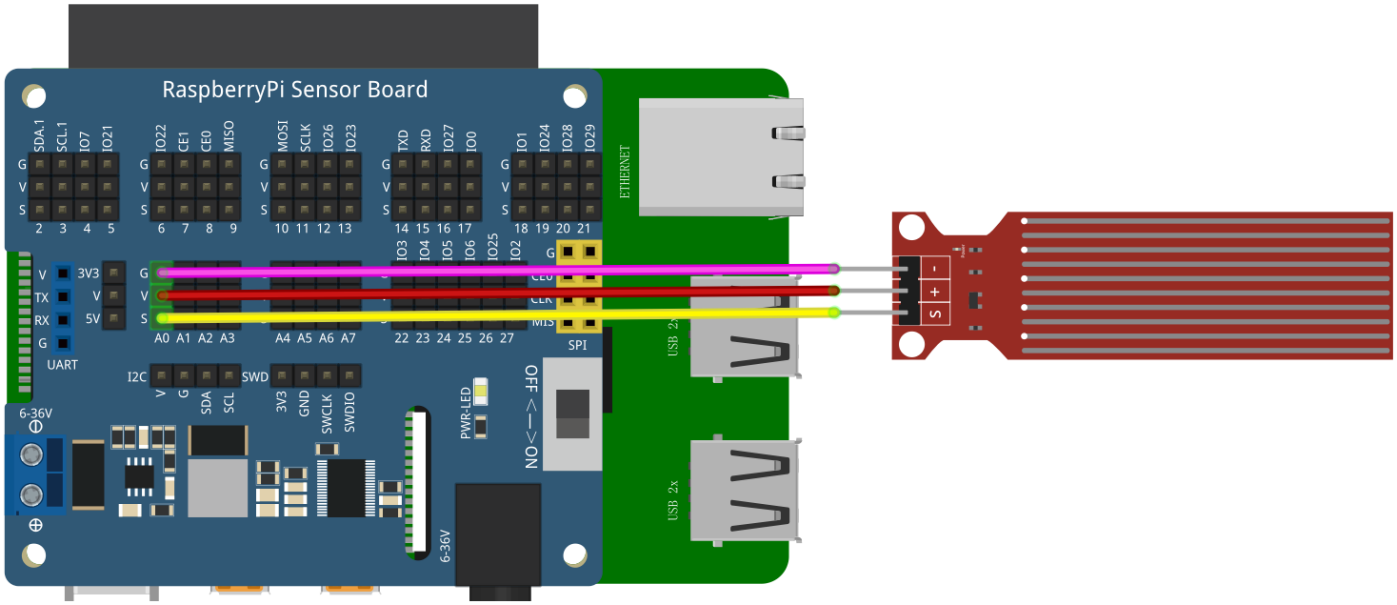
- Raspberry Pi main board
- Breadboard
- Cable
- Water Sensor Module * 1
- A cup of water
- Server jumper wires

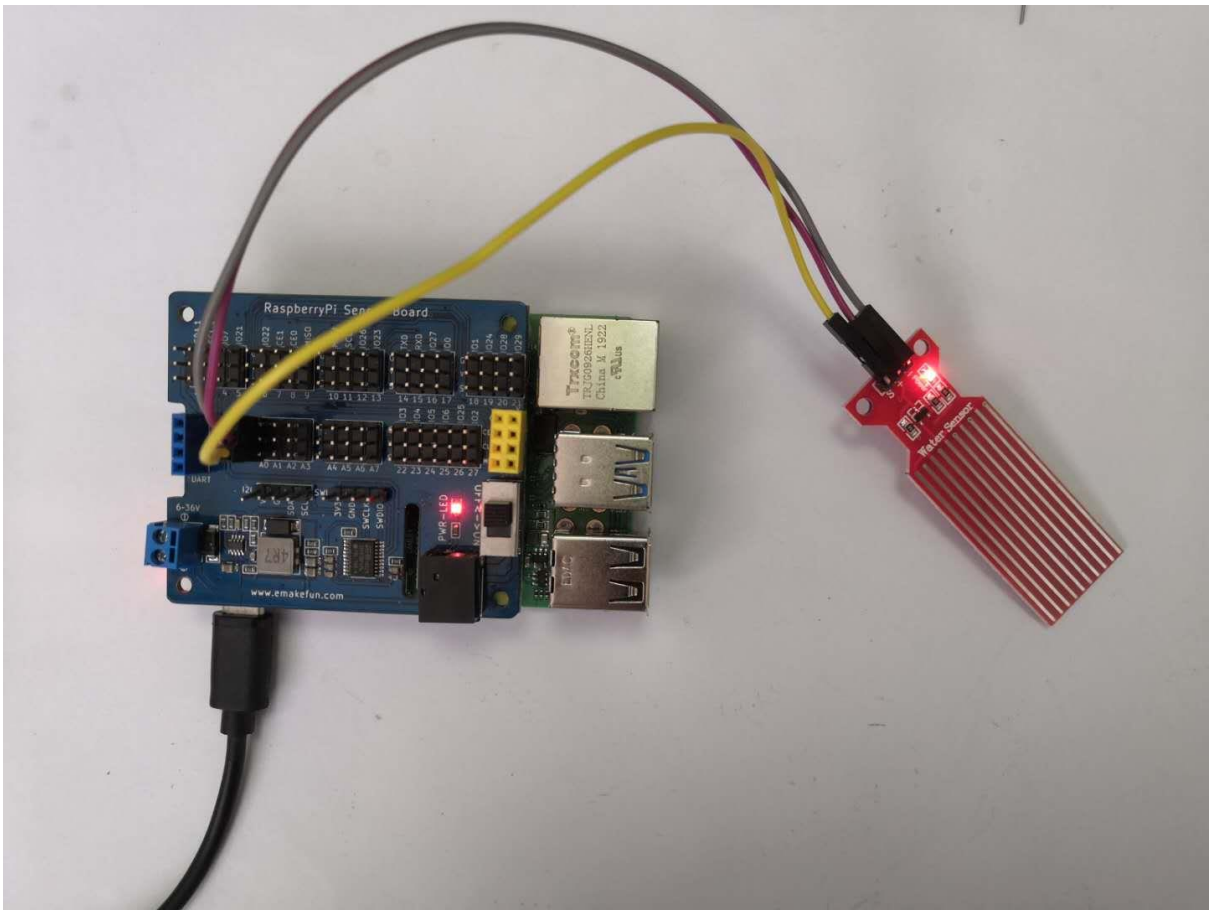
Wiring

Water Sensor Wiring

Water Sensor Module	Raspberry Pi
S(signal pin)	A0

-（GND）	GND
+ （power）	5V





C++ program

```
#include <stdio.h>//Import the basic library
#include <wiringPi.h>//Import the Raspberry Pi WiringPi encoding IO control library
#include <wiringPiI2C.h>//Import the Raspberry Pi WiringPi coding I2C control library
```

```
int  LEDPIN = 24; //LED light is connected to IO0 port
int  value = 0 ;
float voltage = 0.0 ;
int main()
{
    wiringPiSetup();
    wiringPiI2CSetup(0x04);
    pinMode(LEDPIN, OUTPUT);
    while(1)
    {
        value =  wiringPiI2CReadReg16(0x04, 0x10);
        voltage = float(value)/6500*4;
        printf("%f\n", voltage); // Print Value
        delay(1000); // Delay for 1 second
    }
}
```

Python program

```
import time
import smbus as smbus
import RPi.GPIO as GPIO

ADC=smbus.SMBus(1)#Declare to use I2C 1
GPIO.setmode(GPIO.BCM)

while True:
    ADC.write_byte(0x04, 0x10)#Write a byte to the slave
    val = ADC.read_word_data(0x04, 0x10);
    val = val /6500 * 4
    print(val)#Raspberry Pi reads the data returned by the expansion board and prints it
out
    time.sleep(1)#Delay 1 second
```

Java Program

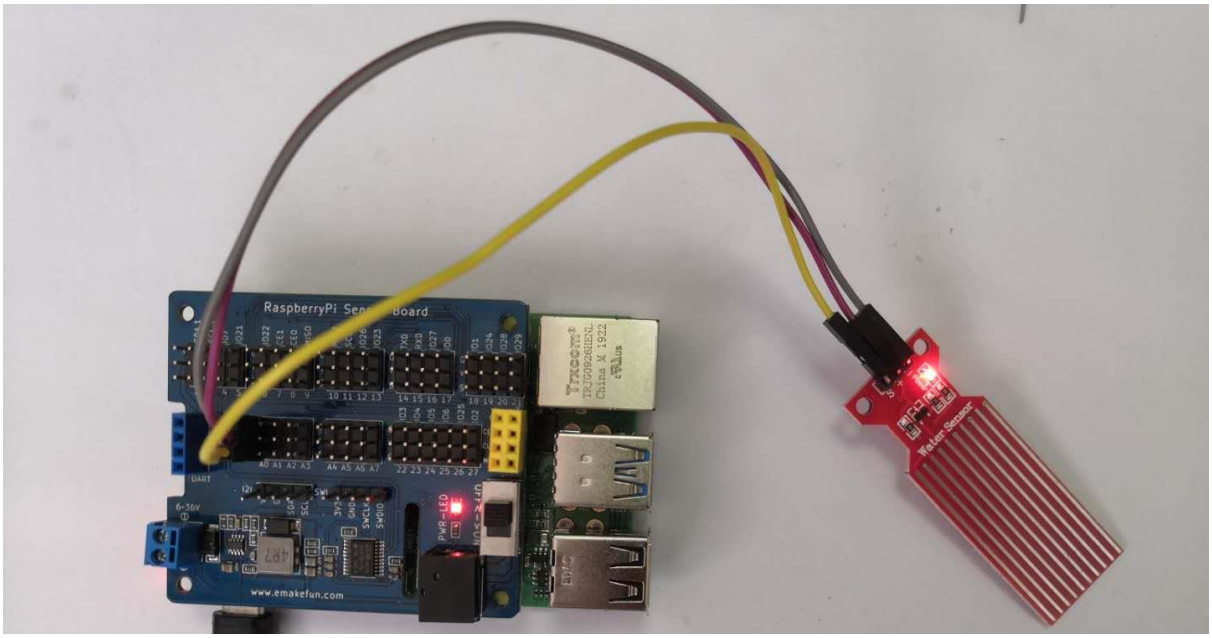
```
import com.pi4j.wiringpi.I2C;
import com.pi4j.wiringpi.Gpio;
import com.pi4j.io.i2c.I2CBus;
import com.pi4j.io.i2c.I2CDevice;
import com.pi4j.io.i2c.I2CFactory;
import java.lang.Math;
public class Water_depth {
```

```
    static int value = 0, fd;
    static float voltage;
    static {
        Gpio.wiringPiSetup();
        fd = I2C.wiringPiI2CSetup(0x04);
    }

    public static void main(String[] args){
        I2CDevice _device = null;
        I2CLCD _lcd = null;

        for ( ; ;) {
            value =  I2C.wiringPiI2CReadReg16(fd, 0x10);
            voltage = (float)(Math.round((1 - (float)value / 4095) * 100 * 100)) / 100;
            String s = String.valueOf(voltage);
            try {
                I2CBus bus = I2CFactory.getInstance(I2CBus.BUS_1);
                _device = bus.getDevice(0x27);
                _lcd = new I2CLCD(_device);
                _lcd.init();
                _lcd.backlight(true);
                _lcd.display_string_pos("deep: " + s +"cm", 1, 2);
            } catch (Exception ex) {
                System.out.println(ex.toString());
            }
        }
    }
}
```

Experimental Effect



While the water sensor module is not inserted into the water, the output value of the analog pin is 0. As the water sensor is gradually submerged in the water, the value of the analog pin becomes larger and larger. While it is completely submerged in the water, the output value of the analog pin is the maximum. Then gradually pull out the water sensor, the value of the analog pin will decrease accordingly. But while the water sensor is completely pulled out, the output value of the analog pin is not 0. I think it is the water left on the surface of the water sensor that affected the output value of the analog pin. After drying the surface of the water sensor with a dry paper towel and the output value of the analog pin will become 0.