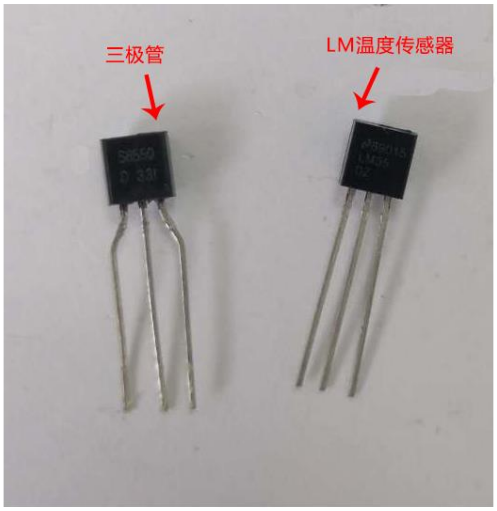
LM35 Temperature Sensor Experiment

Introduction

Many scenarios in real life require temperature measurement。In order to obtain an accurate temperature, we need to use a temperature sensor to measure it. Mercury thermometers are used for body temperature measurement. PT100/PT1000 are normally used to measure the temperature of industrial instruments. LM35 and 18B20 are commonly used temperatures in daily life. This experiment will measure temperature based on LM35.



LM35 is a temperature sensor consisted of a precision integrated circuit and its output voltage is linearly proportional to the Celsius temperature. Therefore, LM35 performance is much better than the absolute scale linear temperature sensor. The LM35 series sensor has been calibrated during production and the output voltage corresponds to the temperature in Celsius, so it is very convenient for users to use. The sensitivity of LM35 series sensor

is 10.0 mV/℃, the accuracy is 0.4℃-0.8℃ (temperature range of -55℃-+150℃), the reproducibility is high and the output impedance is low. It can work under single power supply or positive and negative power supply and it has the following functions:

- Can be directly calibrated in degrees Celsius
- + 10.0 mV /℃ Linear scale
- Can ensure the accuracy of 0.5℃ （25℃）
- Operating temperature range is -55℃-+150℃
- It can be used for long distances
- The operating voltage range is 4v-30v.
- Low power consumption, less than 60 uA.
- Low self-heating effect at rest temperature (0.08℃)
- Nonlinear data is only ±1/4℃
- When passing 1mA current and its output impedance is only 0.1Ω.

Experimental Principle

Raspberry Pi collects the output value of LM35 every 1 second. First of, we obtain the actual voltage via an A/D analog-to-digital converter.

$$V_R = \frac{Value}{2^{10}-1} \times V_{DD}$$, V = Vad*5 / 1023 /10(5V)

Secondly, according to the accuracy of the LM35 sensor: Temp = Vad (V) * 100 (℃/V), we can get the corresponding temperature value.

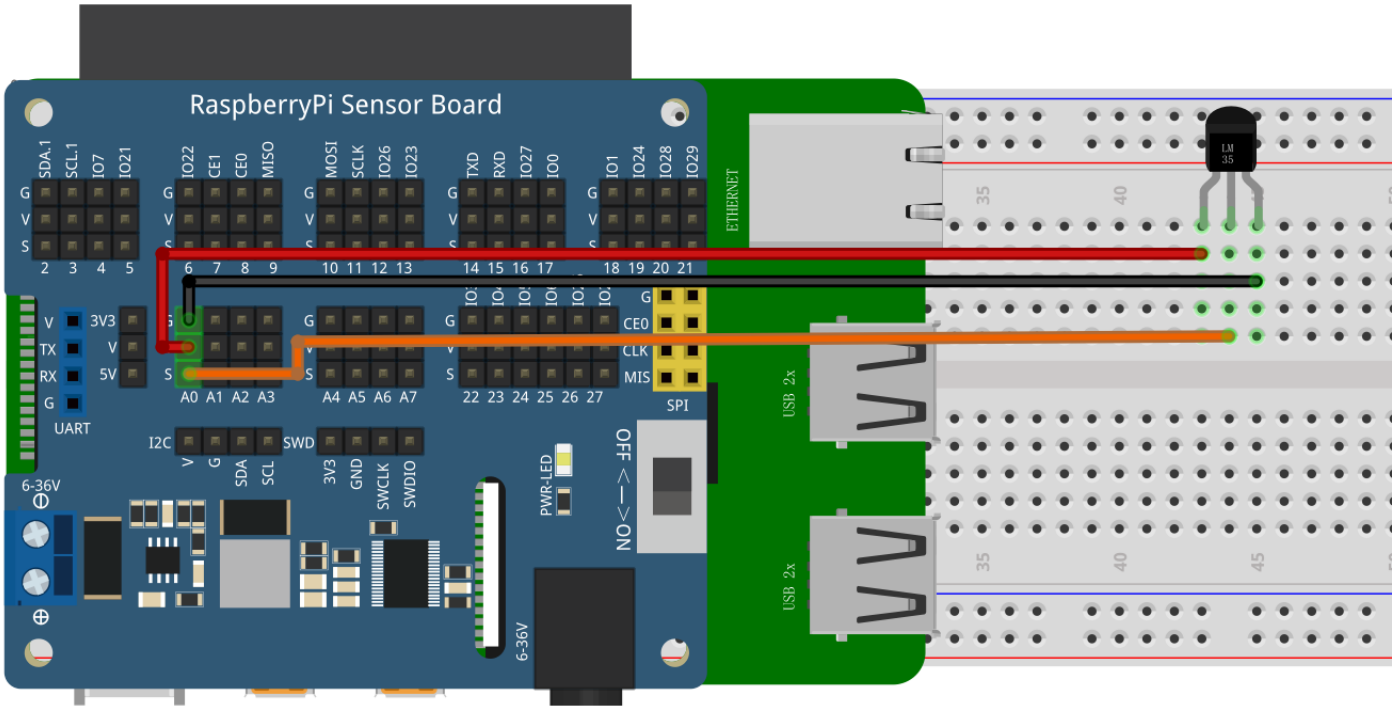
Component List

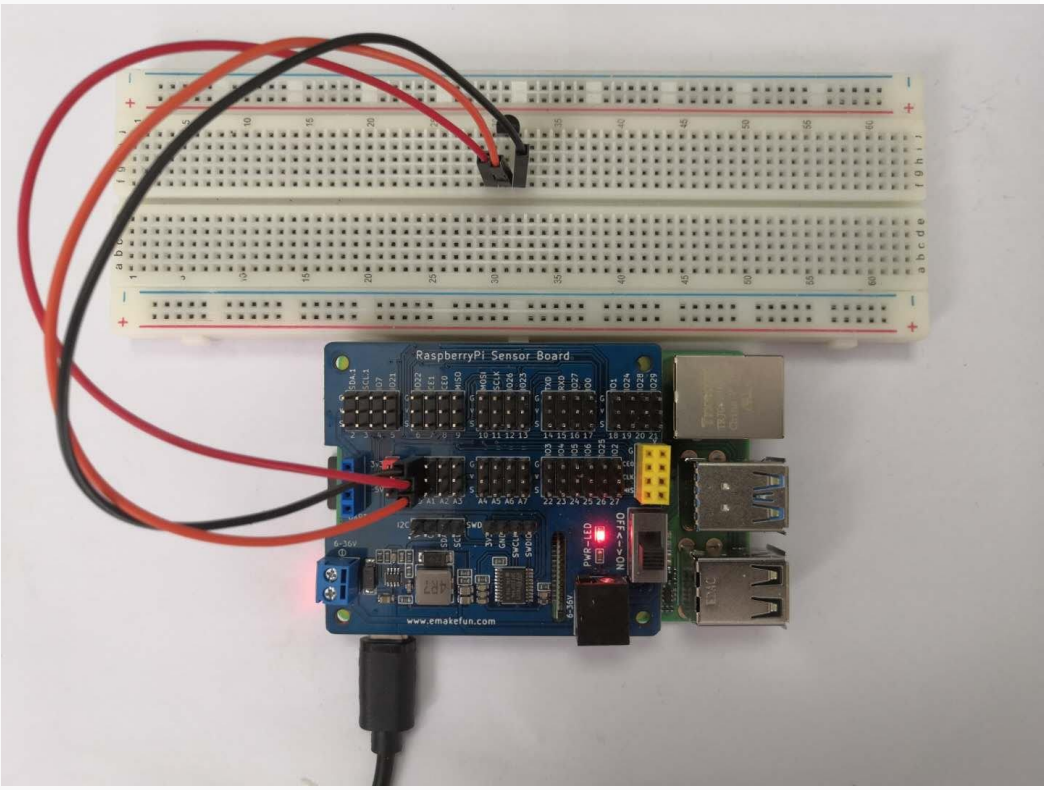
- Raspberry Pi main board
- Raspberry Pi expansion board
- USB Data Cable
- LM35 Temperature Sensor *1
- Several jumper wires

Wiring

Raspberry Pi	LM35

GND	GND
VCC	VCC
A0	DATE（OUT）





C++ program

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

int  value = 0 ;
int a;
float voltage = 0.0 ;

int main()
{
    wiringPiSetup();
    wiringPiI2CSetup(0x04);
    while(1)
    {
        value =  wiringPiI2CReadReg16(0x04, 0x10);
        voltage = ( ( float )value ) / 1023 / 10 ; // Obtain actual voltage through A/D
```

```
analog-to-digital converter
        value =   voltage * 5 * 100 ;            // Sensitivity is 10.0 mV/℃, use formula to
calculate Value
        printf("%d\n",value); // Print Value
        delay(1000); // 1 second delay
    }
}
```

Python program

```
import time
import smbus as smbus

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

while True:
    ADC.write_byte(0x04,0x20)#Write a byte to the slave
    val = ADC.read_word_data(0x04,0x20);
    temp = val / 10 / 1023 * 5 * 100  // Obtain actual voltage through A/D analog-to-digital
converter
    print(temp)#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.wiringpi.GpioInterrupt;
import com.pi4j.wiringpi.GpioInterruptListener;
import com.pi4j.wiringpi.GpioInterruptEvent;
import com.pi4j.wiringpi.GpioUtil;

public class LM35 {
    static int  value;
    static float voltage;

    public static void main(String args[]) throws InterruptedException{

        int fd = I2C.wiringPiI2CSetup(0x04);
        for (;;) {
            value =  I2C.wiringPiI2CReadReg16(fd, 0x10);
            voltage = ( ( float )value ) / 1023 / 10;
```

```
            value =   (int)(voltage * 100 * 5);
            System.out.println("当前温度: " + value +" C");
            Gpio.delay(1000);
        }
    }
}
```

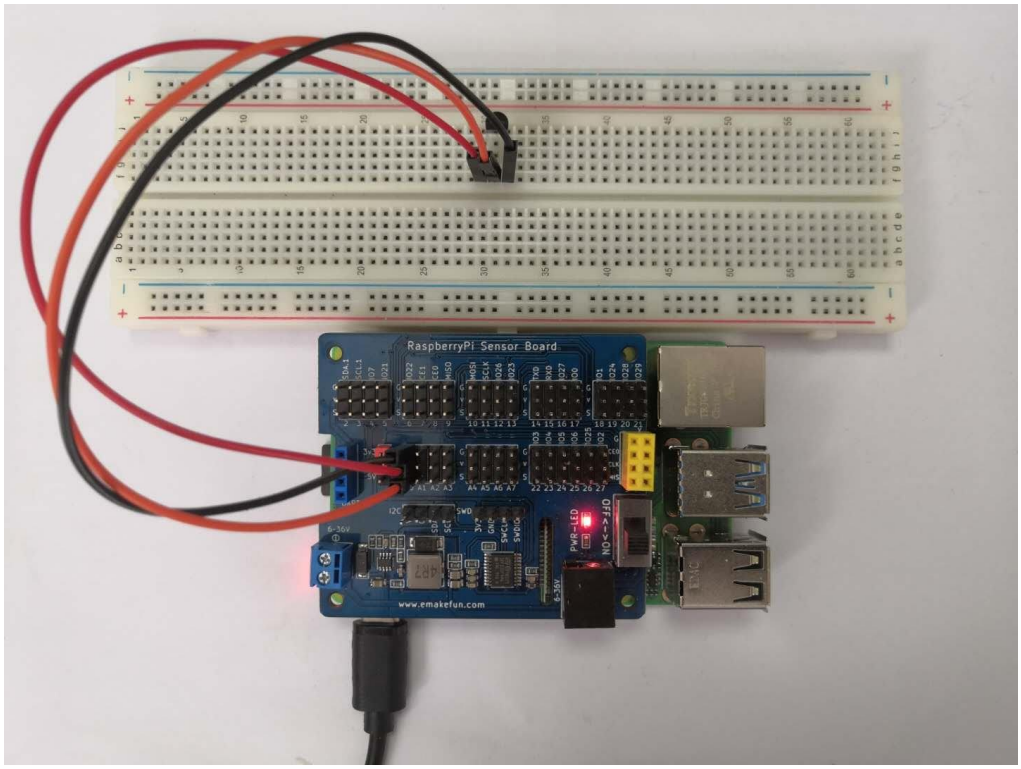
Experimental Effect

lm35.py

```
1  import time
2  import smbus as smbus
3
4  ADC=smbus.SMBus(1)#Declare to use I2C 1
5
6  while True:
7      ADC.write_byte(0x04,0x20)#Write a byte to the slave
8      val = ADC.read_word_data(0x04,0x20);
9      temp = val / 10 / 1023 * 5 * 100
10     print(temp)#Raspberry Pi reads the data returned by the expansion board and prints it out
11     time.sleep(1)#Delay 1 second
```

Shell

```
22.14076246334311
22.287390029325515
24.095796676441836
23.65591397849462
```



By controlling the Raspberry Pi to read the LM35 to obtain the temperature analog value. This experiment requires us to understand how to use the LM35 and the Raspberry Pi expansion board.