# **User Manual**

# **Module Name:** MAX31865 PT100-PT1000 RTD Platinum Resistance Temperature Detector Module

**Theory:**

# Hi Friends! Today, we will see MAX31865IC based Temperature Detector Module. When we need precision temperature sensing, nothing beats a MAX31865 RTD Platinum Resistance Temperature Detector Module PT100-PT1000. Resistance temperature detectors (RTDs) are temperature sensors that contain a resistor that changes resistance value as its temperature changes; basically a kind of thermistor. In this sensor, the resistor is actually a small strip of Platinum with a resistance of 100 ohms at 0 ºC, thus the name PT100.

Compared to most NTC/PTC thermistors, the PT type of RTD is much most stable and precise (but also more expensive) PT100’s have been used for many years to measure temperature in laboratory and industrial processes, and have developed a reputation for accuracy (better than thermocouples), repeatability and stability.

# This Temperature Detector Module can sense the temperature using PT100 and PT1000 RTD probe. It gives the reading for both modules. All you need to do is do proper H/W connections, load the appropriate program and check the output by applying heat. For heat source you can use either soldering gun or hot gun. You need to hold it in front of the RTD(very close to each other). At that time , you will get appropriate readings.

OK!! Now, that’s enough for theory.Right? Let’s dive into Experiment given below.

**Experiment:**

Before uploading any program, make sure that you first do the H/W connections. The H/W connections are given below.

|  |  |
| --- | --- |
| **Arduino Board** | **MAX 31865 Board** |
| +5V | VIN |
| GND | GND |
| 10 | CS |
| 11 | D0 |
| 12 | D1 |
| 13 | CLK |

**Note:**

1. Short the **F+ and F-** terminals of connector.
2. Connect 2 pin RTD100 or RTD 1000 into **RTD+ and RTD-** terminals.

After H/W connection, its time for S/W.

The S/W code is given below:

**Program:**

/\* This is a library for the Adafruit PT100/P1000 RTD Sensor w/MAX31865\*/

#include <Adafruit\_MAX31865.h>

// Use software SPI: CS, DI, DO, CLK

Adafruit\_MAX31865 max = Adafruit\_MAX31865(10, 11, 12, 13); //Use this pin configuration

// use hardware SPI, just pass in the CS pin

//Adafruit\_MAX31865 max = Adafruit\_MAX31865(10);

// The value of the Rref resistor. Use 430.0 for PT100 and 4300.0 for PT1000

#define RREF 430.0

// The 'nominal' 0-degrees-C resistance of the sensor

// 100.0 for PT100, 1000.0 for PT1000

#define RNOMINAL 100.0

void setup()

{

Serial.begin(9600);

Serial.println("Adafruit MAX31865 PT100 Sensor Test!");

max.begin(MAX31865\_2WIRE); // set to 2WIRE or 4WIRE as necessary or keep it as it is.

// Do not change anything in it.

}

void loop()

{

uint16\_t rtd = max.readRTD();

Serial.print("RTD value: ");

Serial.println(rtd);

float ratio = rtd;

ratio /= 32768;

Serial.print("Ratio = ");

Serial.println(ratio,8);

Serial.print("Resistance = ");

Serial.println(RREF\*ratio,8);

Serial.print("Temperature = ");

Serial.println(max.temperature(RNOMINAL, RREF));

// Check and print any faults

uint8\_t fault = max.readFault();

if (fault)

{

Serial.print("Fault 0x");

Serial.println(fault, HEX);

if (fault & MAX31865\_FAULT\_HIGHTHRESH)

{

Serial.println("RTD High Threshold");

}

if (fault & MAX31865\_FAULT\_LOWTHRESH)

{

Serial.println("RTD Low Threshold");

}

if (fault & MAX31865\_FAULT\_REFINLOW)

{

Serial.println("REFIN- > 0.85 x Bias");

}

if (fault & MAX31865\_FAULT\_REFINHIGH)

{

Serial.println("REFIN- < 0.85 x Bias - FORCE- open");

}

if (fault & MAX31865\_FAULT\_RTDINLOW)

{

Serial.println("RTDIN- < 0.85 x Bias - FORCE- open");

}

if (fault & MAX31865\_FAULT\_OVUV)

{

Serial.println("Under/Over voltage");

}

max.clearFault();

}

Serial.println();

delay(1000);

}

**Note:**

1. This program is for PT-100 RTD but it also works for PT-1000 RTD.
2. For better results/accuracy, you can remove the Rref which is 430ohm(SMD resistor on board) and replace it with 4300ohm(SMD resistor). Change the “ values “ in above code(Rref and Rnominal) also. Change Rnominal to 1000.
3. Use 0.01uf or 0.1uf capacitor between RTD+ and RTD-.(optional)

**Output:**

Image 1

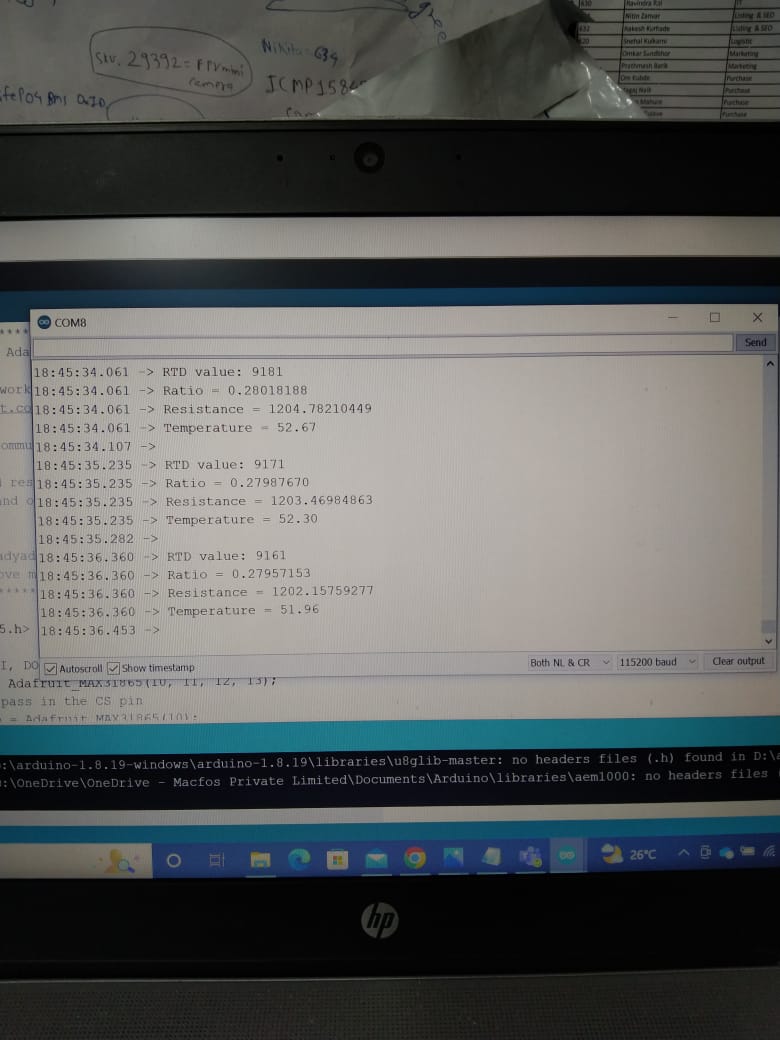
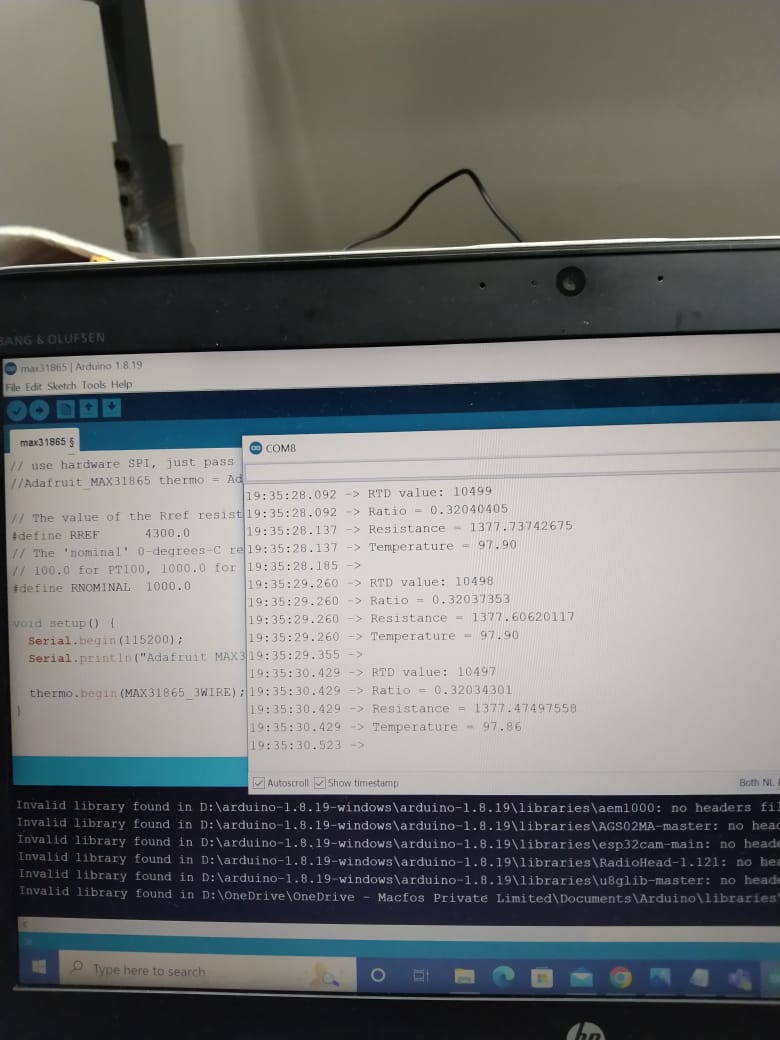


Image 2



-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------