Đo nhiệt độ và hiển thị lên máy tính:

Sử dụng cảm biến nhiệt độ DS18B20

A black wire with a silver tube

Description automatically generated with medium confidence

+ Nguồn: 3 – 5.5V

+ Dải đo nhiệt độ: -55 đến 125 độ C ( -67 đến 257 độ F)

+ Sai số: +- 0.5 độ C khi đo ở dải -10 – 85 độ C

+ Độ phân giải: người dùng có thể chọn từ 9 – 12 bits

+ Chuẩn giao tiếp: 1-Wire ( 1 dây ).

+ Có cảnh báo nhiệt khi vượt ngưỡng cho phép và cấp nguồn từ chân data.

+ Thời gian chuyển đổi nhiệt độ tối đa : 750ms ( khi chọn độ phân giải 12bit ).

+ Mỗi IC có một mã riêng (lưu trên EEPROM của IC) nên có thể giao tiếp nhiều DS18B20 trên cùng 1 dây + Ống thép không gỉ (chống ẩm , nước) đường kính 6mm, dài 50mm + Đường kính đầu dò: 6mm + Chiều dài dây: 1m

-Sơ đồ nối mạch:

A circuit board with wires and wires

Description automatically generated

\*Sử dụng Arduino 1.8.19 viết bằng C++ để nạp code chạy mạch Arduino. Sau đó giao tiếp qua cổng COM vào máy tính, thông qua giao diện WinForm được lập trình bằng phần mềm Microsoft Visual Studio để điều khiển truyền nhận dữ liệu và hiển thị lên đồ thị.

\*Giải thích code Arduino 1.8.19

A screenshot of a computer

Description automatically generated

Vào Tools>Manager Libraries và thêm 4 thư viện

+DallasTemperature

+MAX31850 DallasTemp

+MAX31850 OneWire

+OneWire

Là các thư viện hỗ trợ để chạy cảm biến nhiệt độ DS18B20

Rồi ta sẽ khai báo vào Arduino

A screen shot of a computer

Description automatically generated

A white background with black text

Description automatically generated

#define led 13 //Khai báo chân 13(tức là con LED13 có sẵn trên mạch), đồng thời đặt tên chân 13 là led

#define ONE\_WIRE\_BUS 2 //Khai báo chân nhận dữ liệu analog là chân 2

OneWire oneWire(ONE\_WIRE\_BUS); //Cài đặt để sử dụng thư viện OneWire cho các đối tượng

DallasTemperature sensors(&oneWire); //Chuyển các đối tượng liên quan tới OneWire qua dùng cùng với thư viện Dallas (cụ thể ở đây là cảm biến DS18B20)

A screenshot of a computer code

Description automatically generated

Hàm setup, hàm này chỉ chạy 1 lần

Serial.begin(9600); //khởi tạo cổng kết nối với tần số Baud bằng 9600

pinMode(led, OUTPUT); //đặt tính hiệu ra ở chân 13 (vì led được khai báo ở trên là chân 13)

sensors.begin(); //Khởi chạy cảm biến DS18B20

A screenshot of a computer program

Description automatically generatedHàm loop() sẽ chạy lại code nhiều lần và không dừng lại, chú ý là mạch Arduino là không dừng được

String dulieu = ""; //Khai báo biến dulieu với kiểu biến String mang giá trị trống

nhay1: //tạo một cờ để dùng cho lệch goto

while(Serial.available()>0) //tạo vòng while đặt điều kiện là bất kể khi nào còn tín hiệu của cổng Serial được truyền vào thì sẽ lặp

{

char a = Serial.read(); //khai báo biến a bằng dữ liệu đọc được nhận từ Serial

dulieu += a; //đặt dulieu bằng dulieu được khai báo ở trên cộng với giá trị a là tín hiệu nhận được ở serial

delay(5); // dừng 5ms

}

dulieu.trim(); //vì dulieu ban đâu bằng “” cộng thêm giá trị từ serial nên bị thừa ra “” tính là 1 giá trị lạ nên mạch sẽ không hiểu được, nên ta phải cắt bớt các phần giá trị lạ đấy đi (đã thử dùng thẳng a để làm biến dữ liệu chính nhưng bị lỗi, và ảnh hưởng đến việc đặt lại dữ liệu về null phía trên)

while(dulieu == "TS") // vòng lặp while với điều kiện khi nhận được dữ liệu từ cổng Serial là “TS” thì sẽ lặp.

sensors.requestTemperatures(); //gửi tín hiệu yêu cầu nhận nhiệt độ từ cảm biến

Serial.println(sensors.getTempCByIndex(0)); //ghi ra cổng Serial một dòng mang giá trị nhiệt độ

delay(1000); //dừng 1 giây

goto nhay1; //nhảy về vị trí cờ nhay1:

A computer screen shot

Description automatically generated

if(dulieu == "ON13") //Nếu dữ liệu đọc được bằng với ON13 thì sẽ các chạy dòng trong lệnh này

digitalWrite(led, HIGH); //gửi tính hiệu số ra chân 13 với tính hiệu high là bật

if(dulieu == "OFF13") //Nếu dữ liệu đọc được bằng với OFF13 thì sẽ các chạy dòng trong lệnh này

digitalWrite(led, HIGH); //gửi tính hiệu số ra chân 13 với tính hiệu low là tắt

\*Giải thích Code Microsoft Visual Studio WinForm viết bằng C#

A screenshot of a computer

Description automatically generated

Đầu tiên sử dụng kéo thả và đặt tên để có được giao diện như trên

Chuột phải vào và chọn View Code hoặc nhấn F7

A computer screen shot of a black screen

Description automatically generated

Khai báo các thư viện cần dùng

using System;

using System.Drawing;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO.Ports;

using ZedGraph;

A computer screen shot of a computer code

Description automatically generated

namespace WindowsFormsAppForArduino

public partial class Form1 : Form

là phần tạo lên tự có, để tạo các lớp của Solution chứa code

public Form1() //khởi tạo hàm công khai Form1 là toàn bộ cửa sổ của ứng dụng

InitializeComponent(); //có sẵn khi khởi tạo

string[] Baud = { "1200", "2400", "4800", "9600", "19200" }; //Tạo mẳng Baud mang kiểu dữ liệu string dùng để chọn tần số Baud

cRate.Items.AddRange(Baud); //Cho tần số Baud có thể chọn được vào ComboBox BaudRate

A screen shot of a computer program

Description automatically generated

private void Form1\_Load(object sender, EventArgs e) //Tạo hàm riêng tư cho sự kiện khi mà Form1 được tải thì sẽ chạy các dòng lệnh ở trong

cPort.DataSource = SerialPort.GetPortNames(); //Lấy tên của các cổng COM kết nối với máy tính ghi vào trong ComboBox để có thể được chọn

cRate.SelectedIndex = 3; //Chọn tốc độ Baud mặc định trong ComboBox là số ở vị trí thứ 3 trong mảng Baud được khởi tạo trước đó (chú ý đếm từ 0 đến 3 thay cho 1 đến 3)

GraphPane myPanne = zedGraphControl1.GraphPane; //lệnh cài đặt cho biểu đồ hiển thị nhiệt độ

zedGraphControl1.GraphPane.Fill = new Fill(Color.FromArgb(255, 255, 255)); //chỉnh màu nền ngoài cho biểu đồ theo màu RGB, mang giá trị 255, 255, 255 tức là màu trắng. Số thứ nhất đại diện màu đỏ, thứ 2 đại diện mà xanh lá, thứ 3 đại diện màu xanh dương.

zedGraphControl1.GraphPane.Chart.Fill = new Fill(Color.FromArgb(255, 200, 100)); //chỉnh màu cho phần nền phía trong của biểu đồ

A screenshot of a computer

Description automatically generated

myPanne.Title.Text = "Temperature Sensor";

myPanne.YAxis.Title.Text = "Degrees Celsius";

myPanne.XAxis.Title.Text = "Time";

//chỉnh tên cho các trục

A screenshot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated

RollingPointPairList list = new RollingPointPairList(500000); //lệnh khởi tạo cho cái đường kẻ do nhiệt độ trên biểu đồ

LineItem line = myPanne.AddCurve("Tem", list, Color.Blue, SymbolType.Diamond); //lệnh cài đặt, hiệu chỉnh cho đường kẻ, đặt tên cho đường kẻ là “Tem”, mang dữ liệu list của lệnh khởi tạo phía trên, chỉnh màu cho đường kẻ màu xanh dương, để mỗi điểm đo được là một hình kim cương(thực ra nó là cái hình thoi)

myPanne.XAxis.Scale.Min = 0; //đặt giá trị ban đầu nhỏ nhất của trục x của biểu đồ bằng 0

myPanne.XAxis.Scale.Max = 50; //đặt giá trị ban đầu lớn nhất của trục x của biểu đồ bằng 50

myPanne.XAxis.Scale.MinorStep = 1; //đặt giải phân cách bằng 1 tức cách 1 mốc là 1 giá trị

myPanne.XAxis.Scale.MajorStep = 2; //đặt số lần phân cách trước khi có 1 giải phân cách mang giá trị thấy được trên biểu đồ

Tương tự với trục Y

myPanne.YAxis.Scale.Min = 0;

myPanne.YAxis.Scale.Max = 50;

myPanne.YAxis.Scale.MinorStep = 1;

myPanne.YAxis.Scale.MajorStep = 2;

zedGraphControl1.AxisChange(); //lệnh cho phép thay đổi giá trị các trục

A screen shot of a computer program

Description automatically generated

Hàm thực hiện việc vẽ đường giá trị nhiệt độ lên đồ thị

A computer screen shot of a program code

Description automatically generated

Hàm thực hiện nhận tín hiệu của cổng COM rồi truyền ngược lại dữ liệu qua hàm vẽ để thực hiện lệnh vẽ

A computer screen shot of a program

Description automatically generated

private void bConnect\_Click(object sender, EventArgs e) //tạo hàm riêng tư chạy khi có sự kiện nhấn chuột vào nút Connect

if (!serCOM.IsOpen && cPort.Items.Count != 0) //kiểm tra cổng COM nếu tắt và có phát hiện được cổng COM đang kết nối với máy thì sẽ chạy dòng lệnh ở dưới

try //dùng lệnh try catch để chống ngoại lệ, phần mềm sẽ chạy phần lệnh ở catch nếu phần lệnh của try rơi vào trường hợp ngoại lệ, chống được việc phần mềm crash hoặc window nhảy lên cửa sổ thông báo lỗi

{

bConnect.Text = "Disconnect"; //nút Connect chuyển thành Disconnect

Note.Text = "Connected"; //dòng note phía dưới chuyển thành Connected

serCOM.PortName = cPort.Text; //ghi tên port bằng với tên port mà mình chọn ở ComboBox

serCOM.BaudRate = Convert.ToInt32(cRate.Text); //chuyển dữ liệu đã chọn từ ComboBox sang int32 rồi ghi vào để chỉnh tần số Baud(lưu ý tốc độ Baud phải trùng với tốc độ được cài đặt trong mạch Arduino(9600) thì mới chạy được)

serCOM.Open(); //Mở cổng COM

}

catch

{

Note.Text = "Detect COM port change during connect"; //dòng note phía dưới chuyển thành Detect COM port change during connect

bConnect.Text = "Connect"; // dòng note phía dưới chuyển thành Detect COM port change during connect

//serCOM.Close(); //đóng cổng COM

}

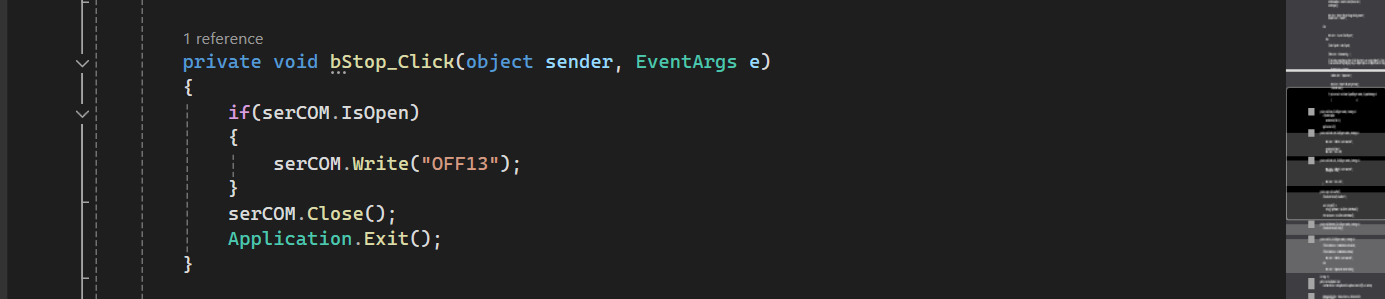
A screenshot of a computer program

Description automatically generated

Else //trong trường hợp không đúng điều kiện với lệnh if phía trên thì sẽ chạy các dòng trong này

if (cPort.Items.Count == 0) //nếu không có cổng COM để chọn trong ComboBox thì sẽ chạy các dòng trong đó

Note.Text = "Can not find COM port"; //dòng note phía dưới chuyển thành Can not find COM port



private void bStop\_Click(object sender, EventArgs e) //tạo hàm hoạt động khi nhấn nút Stop

if(serCOM.IsOpen) //nếu cổng COM mở thì chạy dòng phía trong

{

serCOM.Write("OFF13"); //gửi lệnh “OFF13” ra mạch Arduino

}

serCOM.Close(); //đóng cổng COM

Application.Exit(); //thoát khỏi ứng dụng

A screenshot of a computer

Description automatically generated

private void bLED13\_ON\_Click(object sender, EventArgs e) // tạo hàm hoạt động khi nhấn nút ON13

if (!serCOM.IsOpen) //khi cổng COM đóng thì chạy lệnh ở trong

Note.Text = "COMPort is not connected"; // dòng note phía dưới chuyển thành COMPort is not connected

Else //trong trường hợp ngược lại

serCOM.Write("ON13"); //gửi tín hiệu ON13 qua mạnh Arduino

Note.Text = "LED 13 ON"; // dòng note phía dưới chuyển thành LED 13 ON

A screen shot of a computer

Description automatically generated

Tương tự với nút OFF13

A screenshot of a computer program

Description automatically generated

Và nút Temperature Sensor

A computer screen shot of a program code

Description automatically generated

private async void LoadPort() //khởi tạo hàm đồng bộ LoadPort

cPort.DataSource = null; //Xóa dữ liệu trong ComboBox chọn port

cPort.Items.Clear(); //Xóa mọi ô chọn trong ComboBox chọn port

await Task.Run(() => //buộc các dòng lệnh trước phải chạy rồi mới chạy các dòng trong này

string[] portNames = SerialPort.GetPortNames(); //tạo biến mảng portNames được gắn giá trị các port được kết nối tới PC

return portNames; //trả về giá trị portNames

cPort.DataSource = SerialPort.GetPortNames(); //truyền lại dữ liệu vào ComboBox

cRate.SelectedIndex = 3; //đặt lại tần số Baud về mặc định

private void bRefresh\_Click\_1(object sender, EventArgs e) //khởi tạo hàm hoạt động khi nhấn nút Refresh

LoadPort(); //chạy hàm LoadPort đã khởi tạo ở trên

Khi hoàn thành ta có thể Build ra một bản Release cho ứng dụng của chúng ta. Ta có thể gửi ứng dụng qua các thiết bị khác để chạy mạch Arduino đã được nạp code từ phần mềm Arduino 1.8.19 trước đó.

Ưu:

-Chạy đúng yêu cầu

-Đo và hiển thị lên biều đồ cùng nhiệt độ thực

-Có thể gửi ứng dụng qua thiết bị khác không có Visual Studio để điều khiển mạch

-Có khả năng kiểm tra kết nối của cổng COM để tránh crash ứng dụng

-Có thể tùy ý di chuyển biều đồ to nhỏ gần xa để đánh giá giá trị tốt nhất

Nhược:

-Còn chưa hoàn toàn tối ưu

-Chưa đa dạng chức năng

-Một số lỗi nhỏ vẫn chưa giải quyết được