// June 2017 -------------- Written in C# Windows Form Application with Visual Studio 2013

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO.Ports;

namespace Serial

{

public partial class Form1 : Form

{

bool flag = false; // A global variable defined for coloring the textbox

bool flag2 = false;

public SerialPort serialport1; // This is a must to make a copy of our class

// Will be used in Button Section

public Form1()

{

InitializeComponent();

getAvailablePorts();

}

private void label2\_Click(object sender, EventArgs e)

{

}

private void Form1\_Load(object sender, EventArgs e) // Open the port

{

}

void getAvailablePorts()

{

string[] ports = SerialPort.GetPortNames();

comboBox1.Items.AddRange(ports);

}

private void comboBox2\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void button3\_Click(object sender, EventArgs e)

{

button3.Text = "Close"; // change the button3 name from Open port to close!

serialport1 = new SerialPort(); // We need to define a new function for serialport1

serialport1.PortName = comboBox1.Text;

serialport1.BaudRate = Convert.ToInt32(comboBox2.Text);

if (flag2 == true)

{

this.Close(); // Close the form

timer1.Stop();

Environment.Exit(0); // completely exit from the program

}

serialport1.Open();

flag2 = true;

progressBar1.Value = 100;

System.Threading.Thread.Sleep(200);

textBox1.Enabled = true;

button3.Enabled = true;

if (comboBox1.Text == "" || comboBox2.Text == "" )

{

textBox2.Text = "Please check your settings!"; // You have not chosen anything!

}

}

private void button4\_Click(object sender, EventArgs e)

{

progressBar1.Value = 0;

button3.Enabled = false;

button3.Enabled = true;

textBox1.Enabled = false;

}

private void progressBar1\_Click(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void button1\_Click\_3(object sender , EventArgs e)

{

timer1.Start();

}

private void timer1\_Tick(object sender, EventArgs e)

{

int Henri1;

int Henri3;

int Henri7;

// int Henri8;

string Henri2;

string Henri4;

string Henri6;

progressBar1.Value = 50;

byte[] bytesToSend = new byte[14] { }; // Here is the code you need to add written in your electronics firmware!

serialport1.Write(bytesToSend, 0, 14);

System.Threading.Thread.Sleep(200); // wait for .5 sec

byte [] buffer = new byte[serialport1.BytesToRead];

serialport1.Read(buffer, 0, buffer.Length); // in here buffer.Length = 14

StringBuilder strHEX = new StringBuilder();

if (buffer.Length > 0)

{

Henri1 = buffer[0];

if (Henri1 == 0xFE)

{

Henri3=buffer[17];

Henri7 = buffer[16];

Henri2 = Henri3.ToString("X"); // Appending

Henri4 = Henri7.ToString("X");

Henri6 = Henri2 + Henri4;

int decValue = Convert.ToInt32(Henri6, 16);

if (flag == false)

{

textBox2.BackColor = Color.Yellow;

flag = true;

}

else

{

textBox2.BackColor = Color.LightBlue;

flag = false;

}

textBox2.Text = (decValue / 100).ToString() + "." + (decValue % 100).ToString() + " v";

progressBar1.Value = 85;

}

}

else

{

textBox2.BackColor = Color.Red;

}

}

private void groupBox1\_Enter(object sender, EventArgs e)

{

}

private void button2\_Click\_1(object sender, EventArgs e)

{

timer1.Stop();

}

private void Close(object sender, EventArgs e)

{

}

}

}