```
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.Net;

namespace Test_9_5_24
{
    public partial class Form1 : Form
    {
        int startflag = 0;
        int flag_sensor;
        string RxString;
        string temp = "30";

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            serialPort1.PortName = "COM4";
            serialPort1.BaudRate = 115200;
    serialPort1.Open();
    if (serialPort1.IsOpen)
    {
        // startSerial.Enabled = false;
        // serialStop.Enabled = true;
        textBox1.ReadOnly = false;
    }

}

private void Stop_Click(object sender, EventArgs e)
{
    serialPort1.Close();
    //  startSerial.Enabled = true;
    //  serialStop.Enabled = false;
    textBox1.ReadOnly = true;

}

private void Form1_Load(object sender, EventArgs e)
{
    if (serialPort1.IsOpen)
        serialPort1.Close();

    serialPort1.PortName = "COM4";
    serialPort1.BaudRate = 115200;
}
```

```
        private void textBox1_TextChanged(object sender, EventArgs e)
        {

        }

        private void label1_Click(object sender, EventArgs e)
        {

        }

        private void Current_data_Click(object sender, EventArgs e)
        {
            textBox1.AppendText(RxString);
        }


        private void SerialPort1_DataReceived(object sender,
System.IO.Ports.SerialDataReceivedEventArgs e)
        {

            RxString = serialPort1.ReadExisting();
            this.Invoke(new EventHandler(Current_data_Click));



        }

        private void timer1_Tick_Tick(object sender, EventArgs e)
        {
```

```
if (!string.Equals(textBox1.Text, ""))
{
    if (serialPort1.IsOpen) serialPort1.Close();
    try
    {
         if (RxString[0] == 'B')
          {
              flag_sensor = 10;


          }

        const string WRITEKEY = "S0G72DJK4L2KVBY8";
        string strUpdateBase = "http://api.thingspeak.com/update";



        string strUpdateURI = strUpdateBase + "?api_key=" + WRITEKEY;
        string strField1 = textBox1.Text;

        HttpWebRequest ThingsSpeakReq;
        HttpWebResponse ThingsSpeakResp;

        if (flag_sensor == 10)
        {

            strUpdateURI += "&field4=" + strField1;
            flag_sensor = 10;
        }
}



                ThingsSpeakReq = (HttpWebRequest)WebRequest.Create(strUpdateURI);
                ThingsSpeakResp = (HttpWebResponse)ThingsSpeakReq.GetResponse();
                ThingsSpeakResp.Close();

                if (!(string.Equals(ThingsSpeakResp.StatusDescription, "OK")))
                {
                    Exception exData = new Exception(ThingsSpeakResp.StatusDescription);
                    throw exData;
                }

            }
            catch (Exception ex)
            {

            }
            textBox1.Text = "";


            serialPort1.Open();
        }







    }
}
```

```
}
```