Project Skynet

Software Code : Project_Skynet/FormMain.cs

분당중학교 3학년 김태욱

(2015.07~09)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO.Ports;

namespace Project_Skynet
{
    public partial class FormMain : Form
    {
        private int CK = 0;
        private FormLoading frmLoad = null;
        private FormControl frmControl = null;
        public int z;

        public FormMain()
        {
            InitializeComponent();
            this.hideForm();
        }

        private void FormMain_Load(object sender, EventArgs e)
        {
            this.startLoad(); //start loading
            panel3.Visible = false;
            timer2.Enabled = true;
            btn_Disconnect.Enabled = false;
            string[] Ports = SerialPort.GetPortNames();
            foreach(string port in Ports) {
                combo_PortName.Items.Add(port);
            }

            txt_CMD.AppendText(Environment.NewLine + "S K Y N E T - V1.0");
            txt_CMD.AppendText(Environment.NewLine + "Net-Based Device Controlling   ↩
              System");
            txt_CMD.AppendText(Environment.NewLine + "This program was coded by Tyler   ↩
              Kim");
            txt_Serial.AppendText(Environment.NewLine + "S K Y N E T - V1.0");
            txt_Serial.AppendText(Environment.NewLine + "Net-Based Device Controlling   ↩
              System");
            txt_Serial.AppendText(Environment.NewLine + "This program was coded by   ↩
              Tyler Kim");
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            blackscreen();
            if (CK == 0) //the very first
            {
                this.startLoad();
                timer1.Enabled = false;
            }
            else if (CK == 1) //after loading has completed
            {
                this.endLoad();
                //this.Visible = true;
                timer1.Enabled = false;
            }
            else if (CK == 2)
            {
                Thread.Sleep(1000);
                this.frmControl = new FormControl(this); //send "this" form to control
                  form
                frmControl.WindowState = FormWindowState.Maximized; //controls form
                  control
                frmControl.Show();
                panel3.Visible = false;
                timer1.Enabled = false;
            }
        }
        private void timer2_Tick(object sender, EventArgs e)
        {
            txt_Date.Text = DateTime.Now.ToString("yyyy-MM-dd");
            txt_Time.Text = DateTime.Now.ToString("tt hh:mm:ss");
        }


        private void frmLoad_frmLoad_FormLoadProgressCompleted(object sender,
          FormLoadProgressCompletedEventArgs e)
        {
            CK = 1;
            timer1.Interval = 1000;
            timer1.Enabled = true; //causes the form to close
            this.WindowState = FormWindowState.Maximized;
            this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None;
            //this.TopMost = true;
        }

        private void btn_Connect_Click(object sender, EventArgs e)
        {
            try
            {
                if (combo_PrgMode.Text != "" || combo_PortName.Text != "")
                {
                    blackscreen();

                    btn_Connect.Enabled = false;
                    panel3.Visible = true;
                    pb_loading.Image = Image.FromFile("aperture.gif");
                    CK = 2;
                    timer1.Interval = 3000;
                    timer1.Enabled = true;
                    btn_Disconnect.Enabled = true;
                    if (combo_PrgMode.Text == "ON")
                    {
                        z = 1;
                        lb_PrgMode.Text = "ON";
                    }
                    else if (combo_PrgMode.Text == "OFF")
                    {
                        z = 0;
                        lb_PrgMode.Text = "OFF";
                    }
                }
                else
                {
                    MessageBox.Show("Please Set Your Programming Mode.");
                }
            }
            catch (Exception ex)
            {
                redscreen();
                txt_CMD.AppendText(Environment.NewLine + "ERROR >>" + ex.Message);
            }
        }

        private void btn_Disconnect_Click(object sender, EventArgs e)
        {
            blackscreen();
            if (this.frmControl != null)
            {
                this.frmControl.Disconn();
            }
        }

        private void combo_Device_SelectedIndexChanged(object sender, EventArgs e)
        {
            if (combo_Device.Text == "T-33")
            {
                combo_PortName.Text = "COM5";
                txt_BaudRate.Text = "57600";
            }
            else
            {
                combo_PortName.Text = "Error";
                txt_BaudRate.Text = "Error";
            }
        }
        private void FormMain_MouseClick(object sender, MouseEventArgs e)
        {
            blackscreen();
        }

        private void btn_Close_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        #region Methods
        private void startLoad()
        {
            if (this.frmLoad == null)
            {
                frmLoad = new FormLoading();
                frmLoad.FormLoadProgressCompleted +=                                            ↩
                  frmLoad_frmLoad_FormLoadProgressCompleted; //make it available to get ↩
                   event
                frmLoad.WindowState = FormWindowState.Normal;
                frmLoad.Show(); //start to load ->go to Loading form
            }
        }
        private void endLoad()
        {
            if (this.frmLoad != null)
            {
                frmLoad.Close();
                frmLoad = null; //close the load form
            }
        }
        private void hideForm()
        {
            this.WindowState = FormWindowState.Minimized;
        }

        public void blackscreen()
        {
            txt_CMD.BackColor = Color.Black;
            txt_Serial.BackColor = Color.Black;
        }

        public void redscreen()
        {
            txt_CMD.BackColor = Color.Red;
            txt_Serial.BackColor = Color.Red;
        }

        #endregion

        public void WriteCMD(string str)
        {
            txt_CMD.AppendText(str + Environment.NewLine);
        }

        public void ConnectWithPortName(string str)
        {
            str = combo_PortName.Text;

        }
        public void WriteSerial(string str)
        {
            txt_Serial.AppendText(DateTime.Now + ">>" + str + Environment.NewLine);

        }

        public int CheckPrgMode()
        {
            if (combo_PrgMode.Text == "ON")
            {
                z = 1;
                return z;
            }
            else if (combo_PrgMode.Text == "OFF")
            {
                z = 0;
                return z;
            }
            else
            {
                return z;
            }
        }



    }
}
```