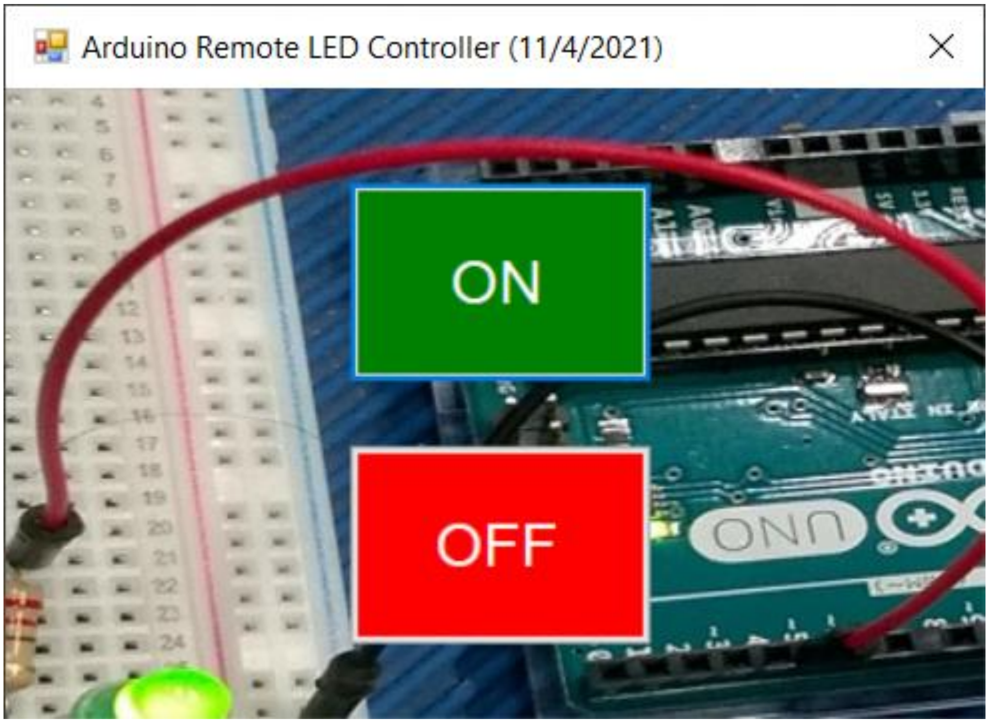
UCI STUDY GROUP 2021
Arduino and C# Programming



This is called “Blinky” and it uses a Windows form C# app to control an LED on a breadboard that is managed by an Arduino UNO.

The circuit uses an LED, & 220ohm resistor connected to pin D5 on the UNO (or equivalent pin). The buttons control the LED.

Be sure to check and see which port# the Arduino is using and hard code that into the C# app where the SerialPort is instantiated.

Arduino UNO Code

Breadboard LED Blinky using Serial Port
Created 11/5/2021
Ron Kessler

Works with Visual Studio 2019 C# Desktop App

```
const int myLEDPin = 5;
char myCmd;      //holds value transmitted


void setup()
{
 Serial.begin(9600);    //Start serial port @9600 baud rate

 pinMode(myLEDPin, OUTPUT);    //Define D5 as output pin to power LED

 digitalWrite(myLEDPin, LOW);  //make sure it is off to start with

 }
void loop()
{
  //---if com port is open then go ahead
  if(Serial.available())
  {
    myCmd = Serial.read();    //read any incoming data
    switch(myCmd)     //switch can only handle int, char, boolean
    {
      case '0':
        digitalWrite(myLEDPin, LOW);  //turn off
        break;
      case '1':
        digitalWrite(myLEDPin, HIGH); //turn on
        break;

    }
  }
}
```

ARDUINO ONBOARD LED TEST

Ron Kessler for UCI Study Group

Created 11/5/2021

```
using System;
using System.IO.Ports;
using System.Windows.Forms;

namespace Blinky_Remote_Control
{
    public partial class frmMain : Form
    {

        public frmMain()
        {
            InitializeComponent();
        }
        private void btnON_Click(object sender, EventArgs e)
        {
            if (!serialPort1.IsOpen)
            {
                try
                {
                    serialPort1.Open();
                    serialPort1.Write("1");
                    serialPort1.Close();

                }
                catch (Exception ex)
                {
                     MessageBox.Show("There was an error. Please make sure that the correct port was
                     selected," + " and the device, plugged in." + ex.Message.ToString());
                }
            }
        }

        private void btnOFF_Click(object sender, EventArgs e)
        {
            if (!serialPort1.IsOpen)
            {
                try
                {
                    serialPort1.Open();
                    serialPort1.Write("0");
                    serialPort1.Close();
                }
                catch (Exception ex)
                {
                    MessageBox.Show("There was an error. Please make sure that the correct port was selected," +
                        " and the device, plugged in." + ex.Message.ToString());
                }
            }
        }


    }
}
```