First Project with Arduino Uno

Four colored LEDs controlled by joystick

# Parts

Arduino Uno

Breadboard

4 LEDs – one each red, blue, green, yellow

4 220 ohm resistors (red, red, black, black, brown color code)

1 long black jumper wire

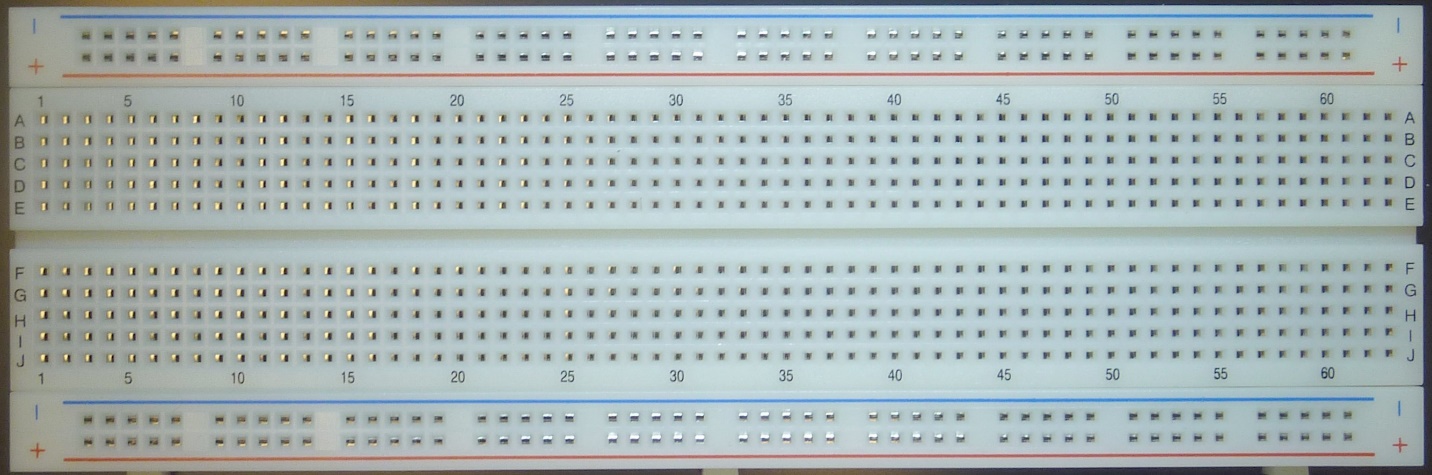
4 short black jumper wires

4 short colored jumper wires - one each red, blue, green, yellow

Joystick

4 male/female jumper wires

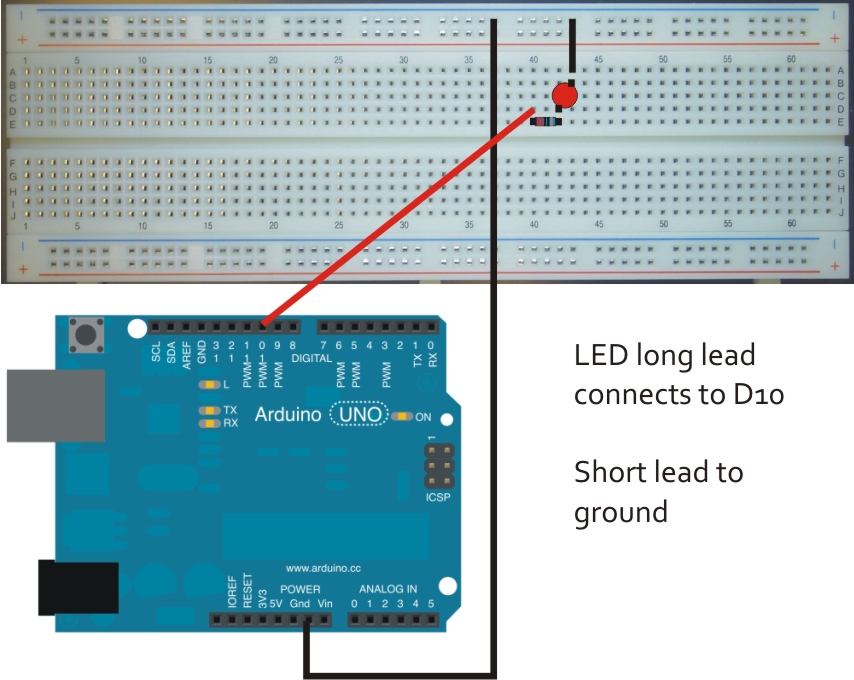
# Breadboard Connections

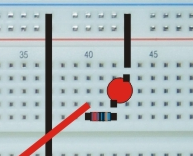


The outer edge red and blue (+ and - ) rows are power lines. On some breadboards, they connect the full length of the board. On others there may be a break in the middle, so you would have to put jumper wires to connect the left and right sides, if needed. The columns in the interior are connected vertically, but there is no connection between columns. So column 5, rows A-E all connect to each other, but to nothing else.

# Set up one LED

Wire the red LED to Arduino pin D10 as shown on next page. Note that the short lead of the LED (cathode) goes to the negative or ground side, the long lead (anode) goes to positive, which is the D10 pin. The resistor can go on either side of the LED, just be consistent.





Close-up of the connections

# Test code for LED

int led = 10; //pin LED connected to

int delayTime = 500; //500 milliseconds = 1/2 second

// the setup function runs once when you press reset or power the board

void setup() {

// initialize digital pin 10 as an output.

pinMode(led, OUTPUT);

}

// the loop function runs over and over again forever

void loop() {

digitalWrite(led, HIGH); // turn the LED on (HIGH applies voltage)

delay(delayTime); // wait for a while

digitalWrite(led, LOW); // turn the LED off (drops the voltage)

delay(delayTime); // wait for while

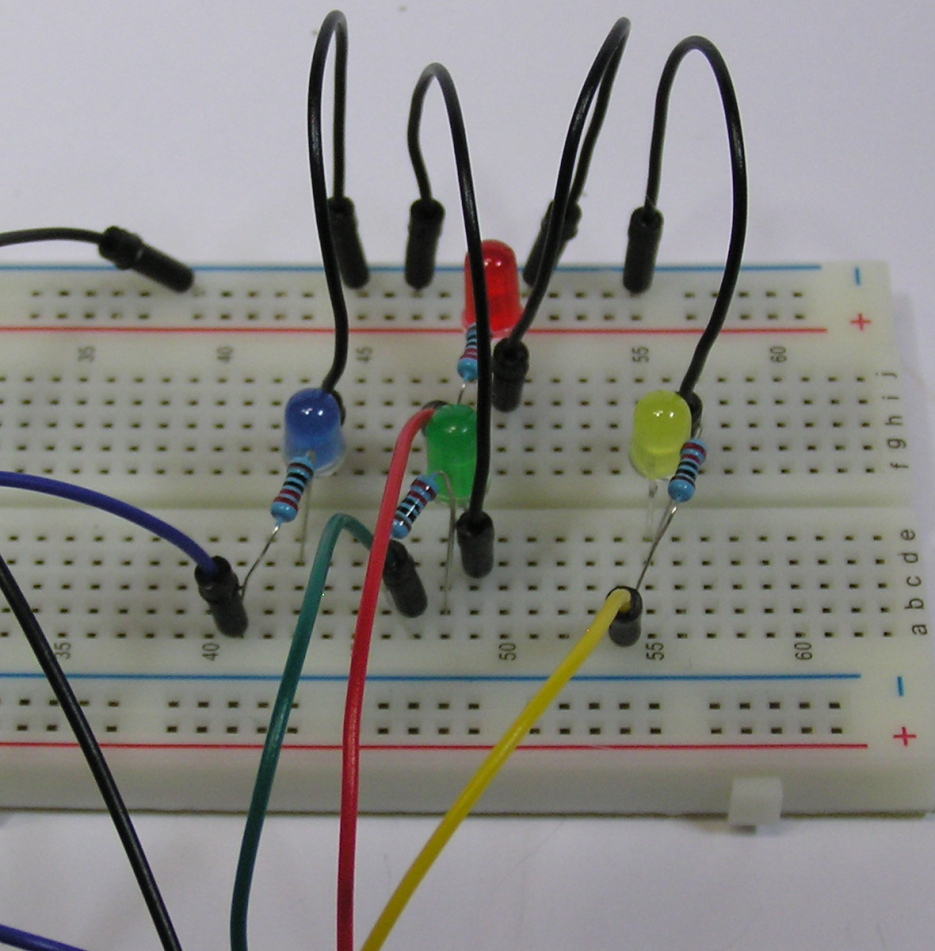
}

# Connect remaining LEDs

Now that you have one LED working, install the yellow, green and blue LEDS clockwise as shown below. Your exact positioning is not critical, as long as all the wiring is correct for each.

Arduino ground (black) 🡪 LED short lead.

LED long lead 🡪 resistor 🡪 Arduino digital pin



The colored wires should be connected to the Arduino as follows:

Red D10

Yellow D11

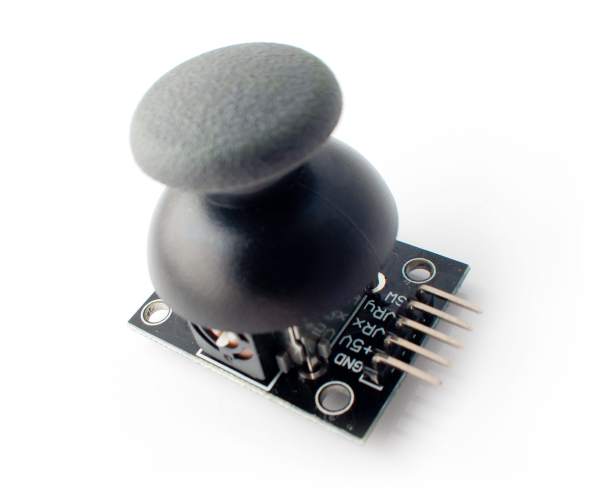
Green D12

Blue D13

All black leads should be connected to the row the ground wire from the Arduino is connected to.

# Connect the joystick

The joystick uses four male/female wires (any color, but black & white make sense for ground and 5v.) If you want to connect the switch as well, you’ll need a fifth wire.



X

Y

1023

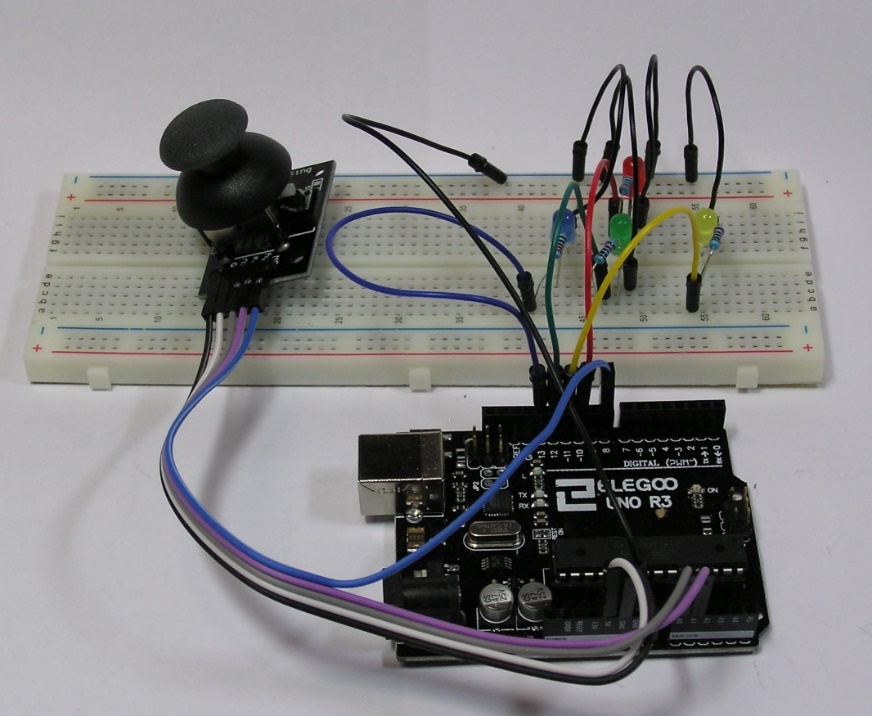
0

1023

0

Make connection between the ground (GND) on the joystick and the Arduino. Connect the joystick +5V to the 5V connection on Arduino. VRx goes to Analog 0 and VRy goes to Analog 1. If used, the SW (switch) goes to whatever Digital pin you choose.

Enter and run the sketch code, you should be able to control the LEDs with the joystick!



The joystick returns values ranging 0 to 1023, where approximately 512 is the neutral position. The code below doesn’t register motion till the joystick is giving a value at least 112 greater or smaller than the neutral value, in order to allow for a bit of errant motion. You can experiment with tighter or looser tolerances.

# Joystick code

int joyPinX = 0; //x input

int joyPinY = 1; //y input

int red = 10;

int yellow = 11;

int green = 12;

int blue = 13;

void setup() {

// put your setup code here, to run once:

pinMode( red, OUTPUT );

pinMode( yellow, OUTPUT );

pinMode( green, OUTPUT );

pinMode( blue, OUTPUT );

for( int i = red; i <= blue; i++ )

digitalWrite( i, LOW ); //turn all LEDs off at start

}

void loop() {

// put your main code here, to run repeatedly:

int joyX = 0;

int joyY = 0;

joyX = analogRead( joyPinX );

delay( 20 );

joyY = analogRead( joyPinY );

//up/down

if( joyY > 625 )

digitalWrite( red, HIGH );

else if( joyY <400 )

digitalWrite( green, HIGH );

//right/left

if( joyX < 400 )

digitalWrite( yellow, HIGH );

else if( joyX > 625 )

digitalWrite( blue, HIGH );

delay( 1000 ); //leave on for 1 second

//turn off all LEDs

for( int i = red; i <= blue; i++ )

digitalWrite( i, LOW ); //turn all LEDs off

delay( 1000 ); //pause before next input processing

}

# What You’ve Learned

## Arduino Hardware

5V and Gnd power connections

Digital vs Analog pins

LED and resistor use

Joystick

## Arduino Code Statements

pinMode( )

analogRead( )

digitalWrite( )

delay( )

## C Programming Statements

Variable declaration

for loop

if branching

Numeric comparison

Assignment

Function calls

# Resistors in Arduino Kit

Ohms Color Pattern

10 Brown Black Black Gold Brown

100 Brown Black Black Black Brown

220 Red Red Black Black Brown

330 Orange Orange Black Black Brown

1000 Brown Black Black Brown Brown

2000 Red Black Black Brown Brown

5100 Green Brown Black Brown Brown

10K Brown Black Black Red Brown

100k Brown Black Black Orange Brown

1000K Brown Black Black Yellow Brown

