Thomas Bock

EEE 174

Wednesday Lab Section

Lab 4 – Arduino

Professor Moyer

Lab Partners:

Anastasiya Herasimava

Slavik Chiley

Daniel Komac

**Introduction:**

This lab introduced the lab group to the Arduino programmable microcontroller. It is a very popular microcontroller meant for controlling I/O functions like LED’s, motors, and various sensors. It is programed in a language very close to C. The lab consisted of twelve projects, described in the Sparkfun Inventor’s kit. This is broken into 3 parts, consisting of 4 projects each. The programming code and instructions for assembling the circuits was given by the Sparkfun Inventor’s kit manual.

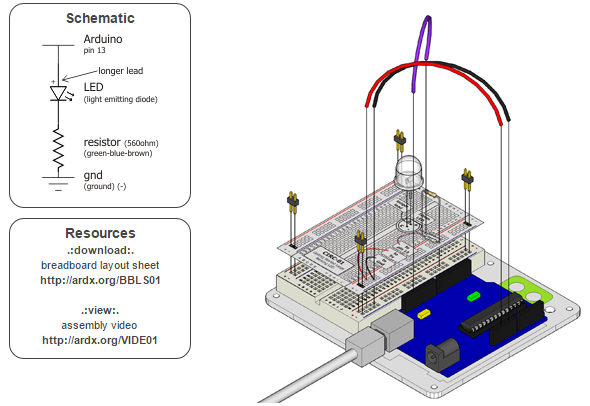
**Part 1: Circuits 1 – 4**

Purpose:

The first part of the Arduino lab consisted of some simple circuits to familiarize the lab group to the Arduino. These labs included some LED’s and motors, with code provided to power them.

Data:

Circuit 1 – Blinking LED



The first circuit is a single LED hooked up to the Arduino which is programmed to blink.

**Circuit 1 Code:**

int ledPin = 10;

void setup() {

pinMode(ledPin, OUTPUT); }

void loop() {

digitalWrite(ledPin, HIGH);

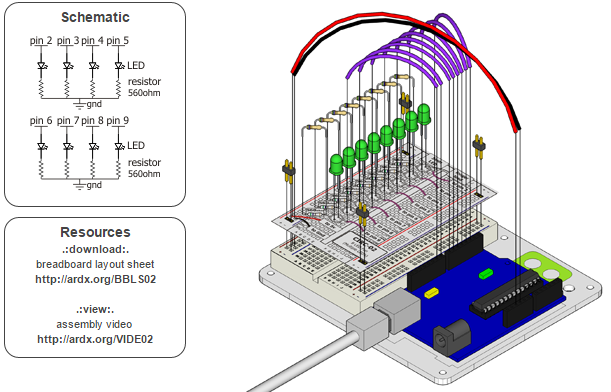
delay(1000);

digitalWrite(ledPin, LOW);

delay(1000);

}

**Circuit 2 – Multiple LEDs**



The second circuit built upon the first circuit by have the lab group attach 8 LED’s to the Arduino. The Arduino was then programmed to turn on the LED’s in a rotating matter.

**Circuit 2 Code:**

int ledPins[] = {2,3,4,5,6,7,8,9};

void setup()

{

for(int i = 0; i < 8; i++)

{

pinMode(ledPins[i],OUTPUT);

}

}

void loop()

{

oneAfterAnotherNoLoop();

}

void oneAfterAnotherNoLoop()

{

int delayTime = 100;

digitalWrite(ledPins[0], HIGH);

delay(delayTime);

digitalWrite(ledPins[1], HIGH);

delay(delayTime);

digitalWrite(ledPins[2], HIGH);

delay(delayTime);

digitalWrite(ledPins[3], HIGH);

delay(delayTime);

digitalWrite(ledPins[4], HIGH);

delay(delayTime);

digitalWrite(ledPins[5], HIGH);

delay(delayTime);

digitalWrite(ledPins[6], HIGH);

delay(delayTime);

digitalWrite(ledPins[7], HIGH);

delay(delayTime);

digitalWrite(ledPins[7], LOW);

delay(delayTime);

digitalWrite(ledPins[6], LOW);

delay(delayTime);

digitalWrite(ledPins[5], LOW);

delay(delayTime);

digitalWrite(ledPins[4], LOW);

delay(delayTime);

digitalWrite(ledPins[3], LOW);

delay(delayTime);

digitalWrite(ledPins[2], LOW);

delay(delayTime);

digitalWrite(ledPins[1], LOW);

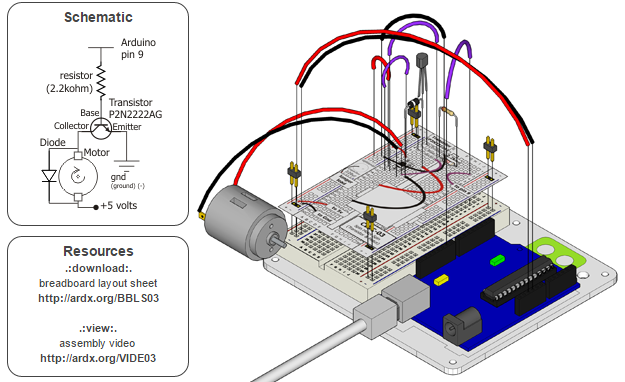
delay(delayTime);

digitalWrite(ledPins[0], LOW);

delay(delayTime);

}

**Circuit 3 – Transistor and Motor**



Circuit 3 consisted of building a circuit with a DC motor. Due to the amperage needs of the motor, a transistor was also used in the circuit. The Arduino was then programmed with instructions to turn the motor on and off.

**Circuit 3 Code:**

int motorPin = 1;

void setup()

{

pinMode(motorPin, OUTPUT);

}

void loop()

{

int onTime = 2500;

int offTime = 1000;

digitalWrite(motorPin, HIGH);

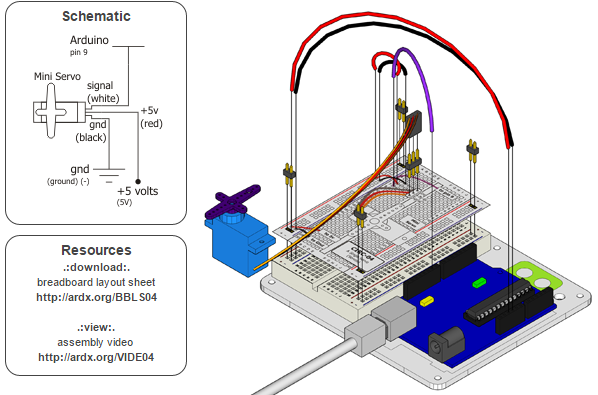
delay(onTime);

digitalWrite(motorPin, LOW);

delay(offTime);

}

**Circuit 4 – Single Servo**



The fourth circuit consisted of connecting a servo to the Arduino. To drive the servo, the Arduino will need to programed to include the servo library.

**Circuit 4 Code:**

#include <Servo.h>

Servo myservo;

int pos = 0;

void setup()

{

myservo.attach(12);

}

void loop()

{

for(pos = 0; pos < 180; pos += 1)

{

myservo.write(pos);

delay(15);

}

for(pos = 180; pos>=1; pos-=1)

{

myservo.write(pos);

delay(15);

}

}

**Results:**

The first part of the Arduino lab is a good introduction to the microcontroller’s design and programming language. There is very little to room to make mistakes due to the step by step nature of the instructions and the programming code being supplied.

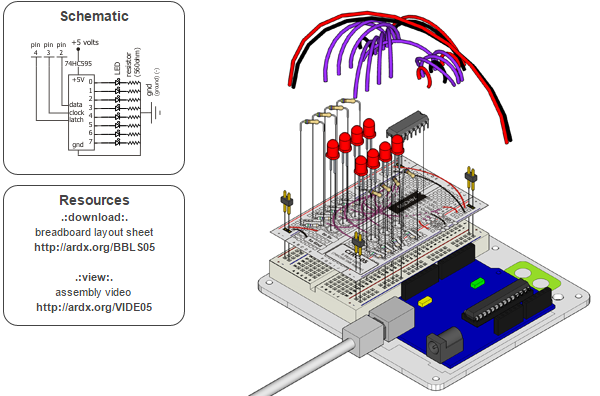
**Part 2: Circuits 5 – 8**

**Purpose:**

For part 2 of the lab, the lab group constructed more complex circuits involving the Arduino. These labs introduced input devices like buttons and potentiometers, integrated circuits such as registers, and buzzers to the lab group.

**Data:**

**Circuit 5 – 8 More LEDs**



Circuit 5 consisted of building a circuit with 8 LEDs, but this time also including a 16 bit shift register. The register is used to power the LEDs to count up in binary. The programming section includes specific code to work with the register.

**Circuit 5 Code:**

int data = 38;

int clock = 37;

int latch = 36;

int ledState = 0;

const int ON = HIGH;

const int OFF = LOW;

void setup()

{

pinMode(data, OUTPUT);

pinMode(clock, OUTPUT);

pinMode(latch, OUTPUT);

}

void loop()

{

int delayTime = 100;

for(int i = 0; i < 256; i++){

updateLEDs(i);

delay(delayTime);

}

}

void updateLEDs(int value){

digitalWrite(latch, LOW);

shiftOut(data, clock, MSBFIRST, value);

digitalWrite(latch, HIGH);

}

void updateLEDsLong(int value){

digitalWrite(latch, LOW);

for(int i = 0; i < 8; i++){

int bit = value & B10000000;

value = value << 1;

if(bit == 128){digitalWrite(data, HIGH);}

else{digitalWrite(data, LOW);}

digitalWrite(clock, HIGH);

delay(1);

digitalWrite(clock, LOW);

}

digitalWrite(latch, HIGH);

}

int bits[] = {B00000001, B00000010, B00000100, B00001000, B00010000, B00100000, B01000000, B10000000};

int masks[] = {B11111110, B11111101, B11111011, B11110111, B11101111, B11011111, B10111111, B01111111};

void changeLED(int led, int state){

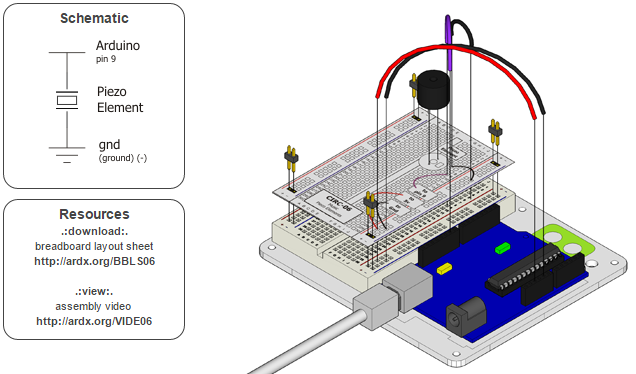
ledState = ledState & masks[led];

if(state == ON){ledState = ledState | bits[led];}

updateLEDs(ledState);

}

**Circuit 6 – Piezo Speaker**



Circuit 6 consisted of building a circuit with piezo speaker. Wiring a speaker is very simple, however programming the speaker to play a tune is somewhat complicated considering the use of timings, tones, and notes.

**Circuit 6 Code:**

int speakerPin = 22;

int length = 15;

char notes[] = "ccggaagffeeddc ";

int beats[] = { 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 4 };

int tempo = 300;

void playTone(int tone, int duration) {

for (long i = 0; i < duration \* 1000L; i += tone \* 2) {

digitalWrite(speakerPin, HIGH);

delayMicroseconds(tone);

digitalWrite(speakerPin, LOW);

delayMicroseconds(tone);

}

}

void playNote(char note, int duration) {

char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C' };

int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956 };

for (int i = 0; i < 8; i++) {

if (names[i] == note) {

playTone(tones[i], duration);

}

}

}

void setup() {

pinMode(speakerPin, OUTPUT);

}

void loop() {

for (int i = 0; i < length; i++) {

if (notes[i] == ' ') {

delay(beats[i] \* tempo); // rest

} else {

playNote(notes[i], beats[i] \* tempo);

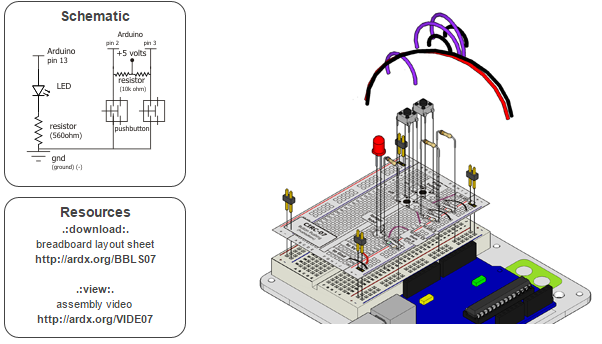
}

delay(tempo / 2);

}

}

**Circuit 7 – Pushbuttons**



Circuit 7 introduces the lab group to their first input device, the button. The circuit involves two buttons, used power on an LED. The programming code introduces the “digitalRead” command used to program actions based on the press of the button or other input device.

**Circuit 7 Code:**

int buttonPin1 = 46;

int buttonPin2 = 47;

int ledPin = 42;

void setup() {

pinMode(ledPin, OUTPUT);

pinMode(buttonPin1, INPUT);

pinMode(buttonPin2, INPUT);

}

void loop(){

if (digitalRead(buttonPin1) == LOW){

digitalWrite(ledPin, HIGH);

}

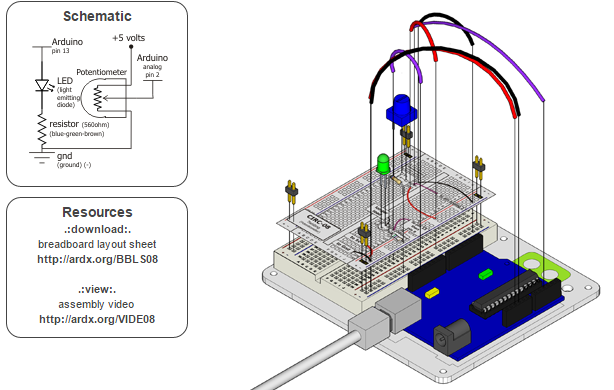
else if (digitalRead(buttonPin2) == LOW){

digitalWrite(ledPin, LOW);

}

}

**Circuit 8 - Potentiometer**



Circuit 8 consisted of building a circuit with a potentiometer. The potentiometer used in conjunction with one of the arduino’s analog inputs allows the user to program a blinking LED that can be sped up or slowed down with the potentiometer.

**Circuit 8 Code:**

int sensorPin = 0;

int ledPin = 51;

int sensorValue = 0;

void setup() {

pinMode(ledPin, OUTPUT);

}

void loop() {

sensorValue = analogRead(sensorPin);

digitalWrite(ledPin, HIGH);

delay(sensorValue);

digitalWrite(ledPin, LOW);

delay(sensorValue);

}

Results:

Part 2 of the lab introduced the lab group to some more complicated circuits involving the Arduino. These circuits took more time to set up than the first 4 circuits, however they gave a better understanding of the capabilities of the Arduino when utilizing both inputs and outputs connected to the Arduino.

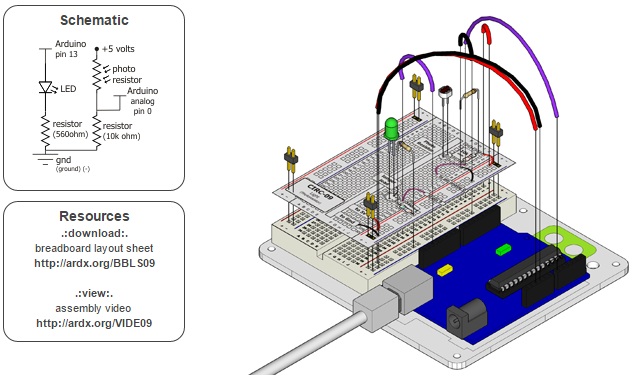
**Part 3: Circuits 9 – 12**

Purpose:

For part 3 of the lab, we needed to construct circuits 9 through 12 from the Inventors handbook. This included the usage of a photo resistor, a temperature sensor, a relay, and an RGB LED.

**Data:**

**Circuit 9 – Photo Resistor**



Circuit 9 consisted of building a circuit with a photo resistor. The photo resistor is used to control the brightness of a LED. With the circuit built and attached to the Arduino, the lab group used the code given in the instructions to program the Arduino.

**Circuit 9 Code:**

int lightPin = A0;

int ledPin = 9;

void setup()

{

pinMode(ledPin, OUTPUT); //sets the led pin to output

}

void loop()

{

int lightLevel = analogRead(lightPin);

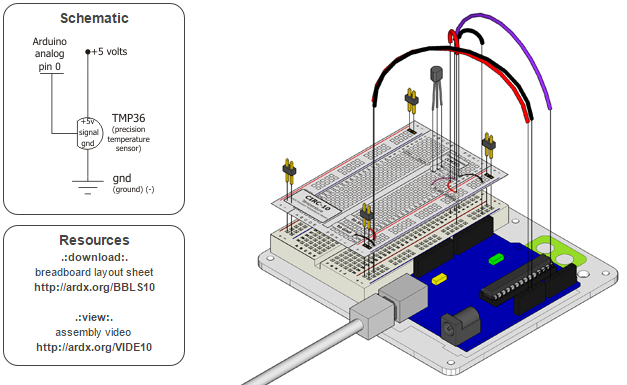
lightLevel = map(lightLevel, 0, 900, 0, 255);

lightLevel = constrain(lightLevel, 0, 255);

analogWrite(ledPin, lightLevel);

}

**Circuit 10 - Temperature**



Circuit 10 consisted of building a circuit with a temperature sensor. The temperature sensor is used to read and output the temperature of the circuit’s surrounding on the computer screen. This section utilized the serial monitor to show that the photo resistor was working. The serial monitor is useful for debugging Arduino circuits and code by outputting data to the serial monitor to show the user if the code is working properly.

**Circuit 10 Code:**

int temperaturePin = A0;

void setup()

{

Serial.begin(9600);

}

void loop()

{

float temperature = getVoltage(temperaturePin);

temperature = (temperature - .5) \* 100;

Serial.println(temperature);

delay(1000);

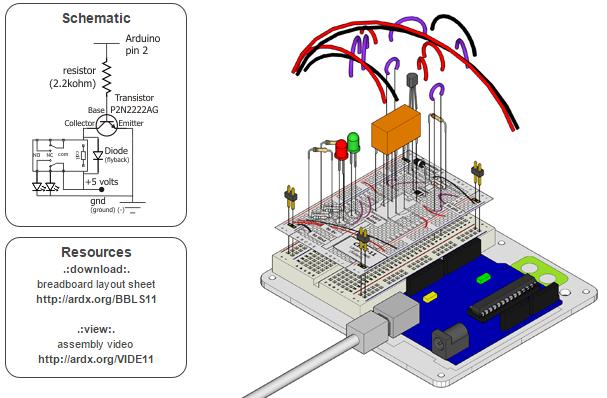
}

float getVoltage(int pin){

return (analogRead(pin) \* .004882814);

}

**Circuit 11 – Relay**



Circuit 11 consisted of building a circuit with a relay. The relay is used to switch between two different LEDs. This is useful in circuits that need a lot of amperage draw.

**Circuit 11 Code:**

int ledPin = 2;

void setup()

{

pinMode(ledPin, OUTPUT);

}

void loop()

{

digitalWrite(ledPin, HIGH);

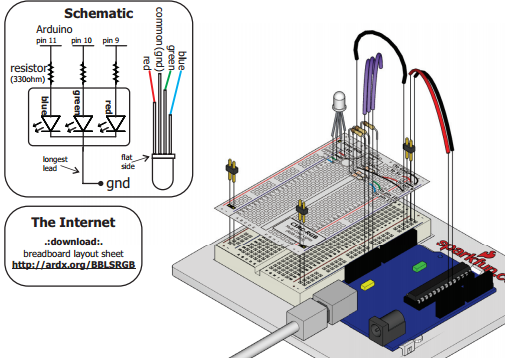
delay(1000);

digitalWrite(ledPin, LOW);

delay(1000);

}

**Circuit 12 – RGB LED**



Circuit 12 consisted of building a circuit with a RGB LED. The RGB LED is used to output different colors of light. With the circuit built and attached to the Arduino, we used the code seen below to control the circuit.

**Circuit 12 Code:**

int redLEDPin = 5;

int greenLEDPin = 9;

int blueLEDPin = 7;

int redSwitchPin = 7;

int greenSwitchPin = 6;

int blueSwitchPin = 5;

int red = 0;

int blue = 0;

int green = 0;

void setup()

{

pinMode(redLEDPin, OUTPUT);

pinMode(greenLEDPin, OUTPUT);

pinMode(blueLEDPin, OUTPUT);

pinMode(redSwitchPin, INPUT\_PULLUP);

pinMode(greenSwitchPin, INPUT\_PULLUP);

pinMode(blueSwitchPin, INPUT\_PULLUP);

}

void loop()

{

if (digitalRead(redSwitchPin) == LOW)

{

red ++;

if (red > 255) red = 0;

}

if (digitalRead(greenSwitchPin) == LOW)

{

green ++;

if (green > 255) green = 0;

}

if (digitalRead(blueSwitchPin) == LOW)

{

blue ++;

if (blue > 255) blue = 0;

}

analogWrite(redLEDPin, red);

analogWrite(greenLEDPin, green);

analogWrite(blueLEDPin, blue);

delay(10);

}

**Results:**

The third section of the lab made use of several sensors, the serial monitor, a relay and a multi-color LED. These devices are important for building more complicated circuits with several inputs and outputs. The serial monitor allows an easier method of debugging circuits and code, which is crucial when the circuits become more and more complex.

**Conclusion:**

The Arduino lab gave a basic understanding of the abilities of the Arduino microcontroller. The Arduino microcontroller has been easy to start using and shows that its usefulness is only limited by the devices you have connected to it, and it has the ability to design some rather complicated circuits. The ease of use of the programming language and the ability to monitor the microcontroller allow for the design of complicated circuits to be done rather easily. Even for members of the lab group that may have used an Arduino in the past, there was something new to learn.