Circuit #1: Blinking an LED

Code:

```
int ledPin = 13;

 void setup() {
   pinMode(ledPin, OUTPUT);
}

 void loop() {
   digitalWrite(ledPin, HIGH); //LED on
   delay(1000); // wait second
   digitalWrite(ledPin, LOW); //LED off
   delay(1000); // wait second
}
```

or for PWM the output loop could read :

```
int ledPin = 11;

void setup() {
 pinMode(ledPin, OUTPUT);
}

 void loop() {
   analogWrite(ledPin, 255); // LED on
   delay(1000); // wait second
   analogWrite(ledPin, 0); // LED off
   delay(1000); // wait second
}
```

Circuit #2: Potentiometer

Code:

```
int sensorPin = 0;
int ledPin = 13;
int sensorValue = 0;

void setup() {
 pinMode(ledPin, OUTPUT);
}

void loop() {
//this line assigns whatever the analog Pin 0 reads to
sensorValue

 sensorValue = analogRead(sensorPin);

 digitalWrite(ledPin, HIGH);
 delay(sensorValue);
 digitalWrite(ledPin, LOW);
 delay(sensorValue);
}
```

Circuit #3: RGB LEDs

Code:

```
const int RED_LED_PIN = 9;
const int GREEN_LED_PIN = 10;
const int BLUE_LED_PIN = 11;
int redIntensity = 0;
int greenIntensity = 0;
int blueIntensity = 0;
const int DISPLAY_TIME = 100;

void setup() {
// No setup required but you still need it
}

void loop(){
   for (greenIntensity = 0; greenIntensity <= 255;
greenIntensity+=5) {
      redIntensity = 255-greenIntensity;
      analogWrite(GREEN_LED_PIN, greenIntensity);
      analogWrite(RED_LED_PIN, redIntensity);
      delay(DISPLAY_TIME);
  }
  for (blueIntensity = 0; blueIntensity <= 255; blueIntensity+=5)
{
      greenIntensity = 255-blueIntensity;
      analogWrite(BLUE_LED_PIN, blueIntensity);
      analogWrite(GREEN_LED_PIN, greenIntensity);
      delay(DISPLAY_TIME);
  }
  for (redIntensity = 0; redIntensity <= 255; redIntensity+=5) {
      blueIntensity = 255-redIntensity;
      analogWrite(RED_LED_PIN, redIntensity);
      analogWrite(BLUE_LED_PIN, blueIntensity);
      delay(DISPLAY_TIME);
  }
}
```

Circuit #4: Multiple LEDs

Code:

```
//this line below declares an array
int ledPins[ ] = {2,3,4,5,6,7,8,9};

void setup( ) {

//these two lines set Digital Pins # 0 – 8 to output
for(int i = 0; i < 8; i++){
pinMode(ledPins[i],OUTPUT);

}

void loop( ) {

//these lines turn the LEDs on and then off
for(int i = 0; i <= 7; i++){
  digitalWrite(ledPins[i], HIGH);
  delay(delayTime);
  digitalWrite(ledPins[i], LOW);
  }
}
```

Circuit #5: Push Buttons

Code:

```
const int buttonPin = 2;
const int ledPin =  13;

int buttonState = 0;

void setup() {
  pinMode(ledPin, OUTPUT);
  //this line below declares the button pin as input
  pinMode(buttonPin, INPUT);
}

void loop(){
  //this line assigns whatever the Digital Pin 2 reads to
  //buttonState
  buttonState = digitalRead(buttonPin);

  if (buttonState == HIGH) {
    digitalWrite(ledPin, HIGH);
   }
  else {
    digitalWrite(ledPin, LOW);
   }
}
```

Circuit #6: Photo Resistor

Code:

```
int lightPin = 0;
int ledPin = 13;

void setup() {
 pinMode(ledPin, OUTPUT);
}

void loop() {
  int lightLevel = analogRead(lightPin);
  lightLevel = map(lightLevel, 0, 900, 0, 255);
  lightLevel = constrain(lightLevel, 0, 255);
  analogWrite(ledPin, lightLevel);
}
```

Circuit #7: Temperature Sensor

Code:

```
int temperaturePin = 0;

void setup() {

//Serial comm. at a Baud Rate of 9600
  Serial.begin(9600);
}

void loop() {

//Calls the function to read the sensor pin
  float temp = getVoltage(temperaturePin);


//Below is a line that compensates for an offset
//(see datasheet)
  temp = (temp - .5) * 100;

   //This line displays the variable temperature after all
  //the math
  Serial.println(temp);
  delay(1000);
}

//function that reads the Arduino pin and starts to convert //
it to degrees

float getVoltage(int pin) {
  return (analogRead(pin) * .004882814);
}
```

Circuit #8: A Single Servo

Code:

```
//include the servo library for use
#include <Servo.h>
Servo myservo;  //create servo object

int pos = 0;

void setup() {
  myservo.attach(9);
}
void loop() {
//moves servo from 0° to 180°
  for(pos = 0; pos < 180; pos += 1) {
    myservo.write(pos);
    delay(15);
    }
  // moves servo from 180° to 0°
  for(pos = 180; pos>=1; pos-=1)  {
    myservo.write(pos);
    delay(15);
    }
}
```

Circuit #9: Flex Sensor

Code:

```
#include <Servo.h> //include the servo library
Servo myservo;

int potpin = 0; //sets pin 0 to read the flex sensor
int val;

void setup() {
  Serial.begin(9600);
  myservo.attach(9);
}

void loop() {
  val = analogRead(potpin); //get a reading from the flex
sensor
  Serial.println(val);
  val = map(val, 50, 300, 0, 179);
  myservo.write(val);
  delay(15);
}
```

Circuit #10: Soft Potentiometer

Code:

```
const int RED_LED_PIN = 9;
const int GREEN_LED_PIN = 10;
const int BLUE_LED_PIN = 11;

void setup() {
//No setup necessary but you still need it
}

void loop() {
  int sensorValue = analogRead(0);
  int redValue = constrain(map(sensorValue, 0, 512, 255,
0),0,255);
  int greenValue = constrain(map(sensorValue, 0, 512, 0,
255),0,255)-constrain(map(sensorValue, 512, 1023, 0,
255),0,255);
  int blueValue = constrain(map(sensorValue, 512, 1023, 0,
255),0,255);

  analogWrite(RED_LED_PIN, redValue);
  analogWrite(GREEN_LED_PIN, greenValue);
  analogWrite(BLUE_LED_PIN, blueValue);
}
```

Circuit #11: Piezo Elements

Note:

This section contains only the two functions needed to make the piezo play a note of a given duration. These functions are called in the loop ( ) function.

Code:

```
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);
      }
   }
}
```

Circuit #12: Spinning a Motor

Code:

```
int motorPin = 9;

void setup() {
  pinMode(motorPin, OUTPUT);
}

void loop() {
  for (int i = 0; i < 256; i++){
    analogWrite(motorPin, i);
    delay(50);
   }
}
```

Circuit #13: Relays

Code:

```
int ledPin =  2;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  //set the transistor on
  digitalWrite(ledPin, HIGH);
  // wait for a second
  delay(1000);
  // set the transistor off
  digitalWrite(ledPin, LOW);
 // wait for a second
 delay(1000);
}
```

Circuit #14: Shift Register

Code:

```
int data = 2;
int clock = 3;
int latch = 4;

int ledState = 0;
const int ON = HIGH;
const int OFF = LOW;

void setup() {
  pinMode(data, OUTPUT);
  pinMode(clock, OUTPUT);
  pinMode(latch, OUTPUT);
}

void loop(){
  for(int i = 0; i < 256; i++) {
    updateLEDs(i);
    delay(25);
    }
}

void updateLEDs(int value) {
  digitalWrite(latch, LOW);
  shiftOut(data, clock, MSBFIRST, value);
  digitalWrite(latch, HIGH);
}
```