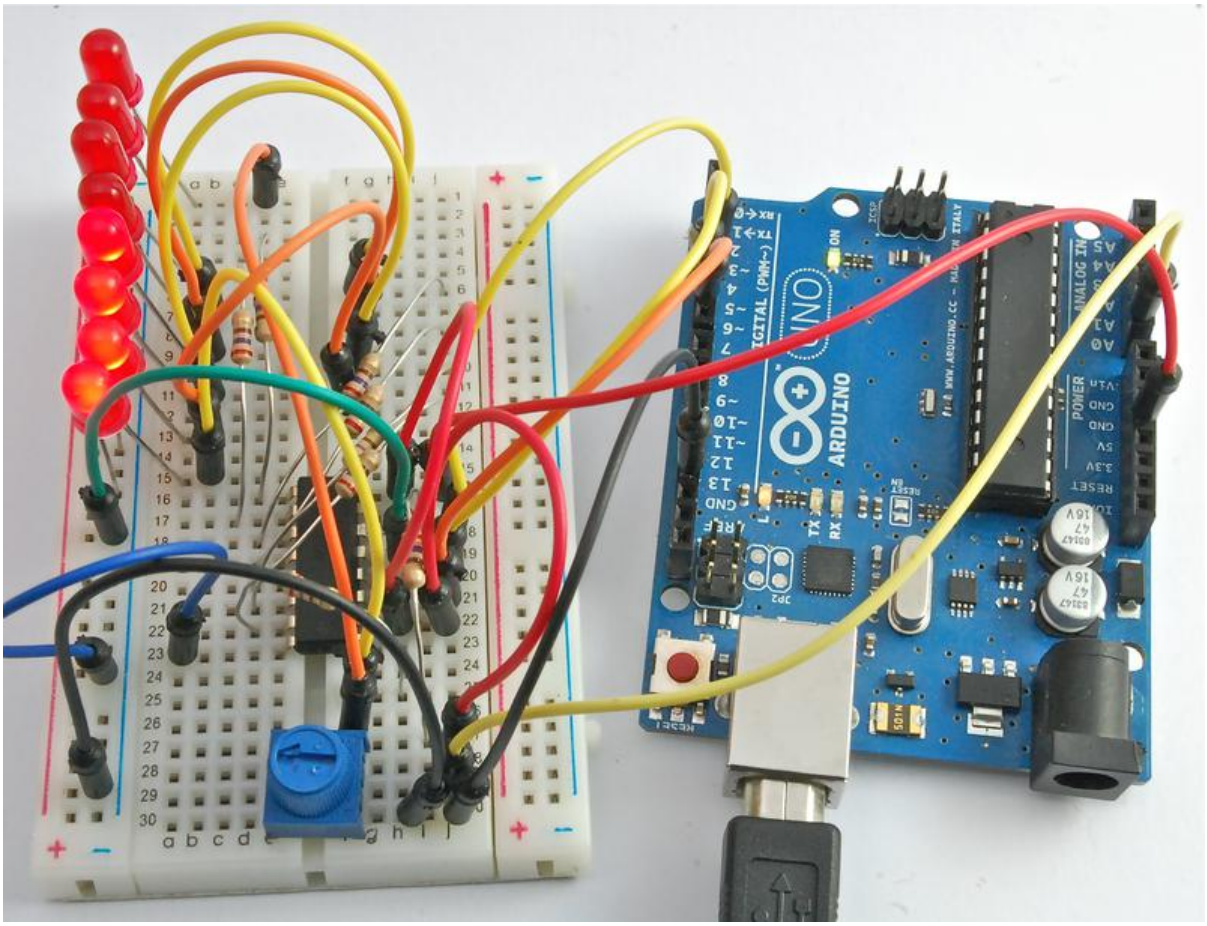
Arduino Lesson 8. Analog Inputs

Created by Simon Monk



https://learn.adafruit.com/adafruit-arduino-lesson-8-analog-inputs

Last updated on 2023-08-29 02:12:23 PM EDT

Table of Contents

Overview 3

Parts 3

- Part
- Qty

An Experiment 8

Variable Resistors (Pots) 10

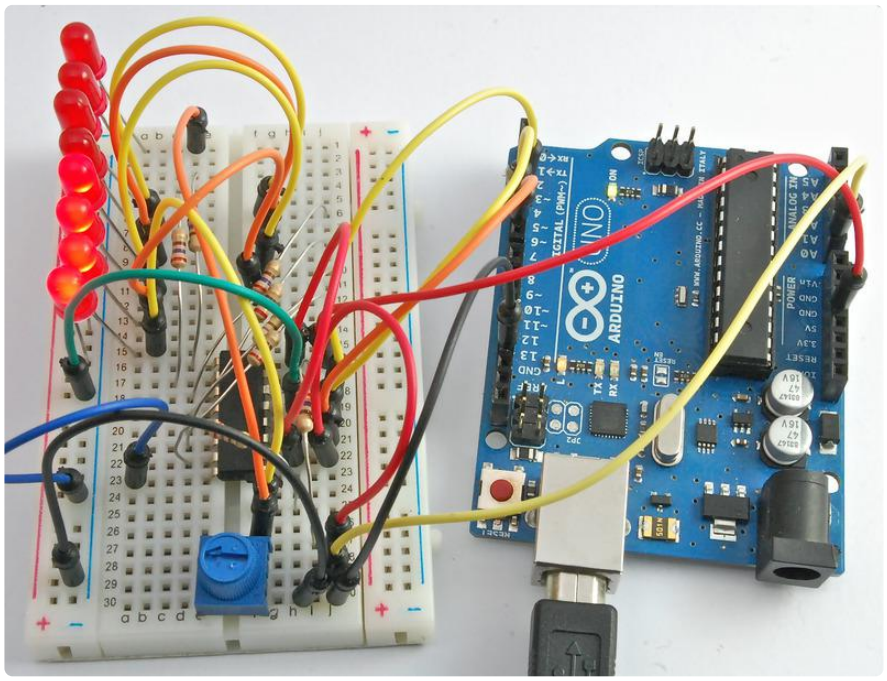
Breadboard Layout 11

Arduino Code 12

Other Things to Do 13

Overview

In this lesson, you will start by using the Serial Monitor to display analog readings, and then extend the project using eight LEDs from lesson 4, so that you can control the number of LEDs that are lit by turning the knob on a variable resistor.



Parts

To build the projects described in this lesson, you will need the following parts.

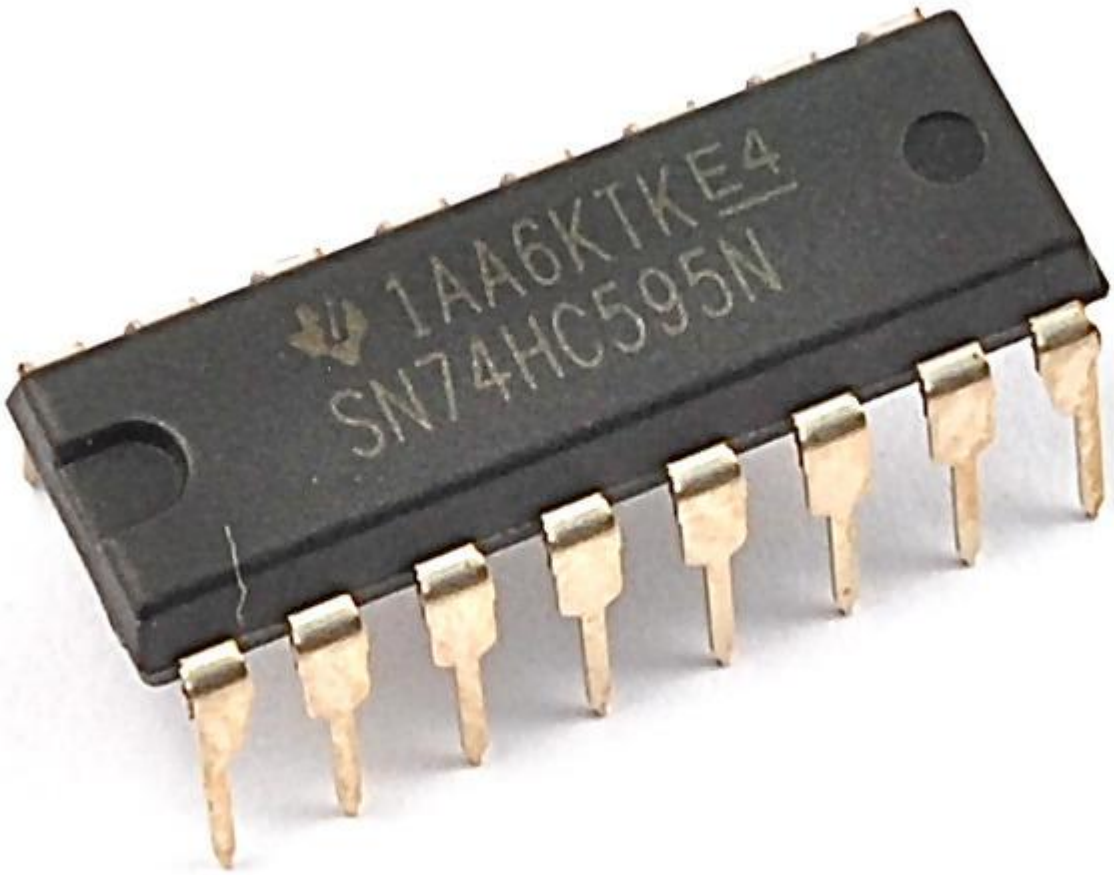
Part
Qty

5mm red LED

8

270 Ω Resistors (red, purple, brown stripes)

8

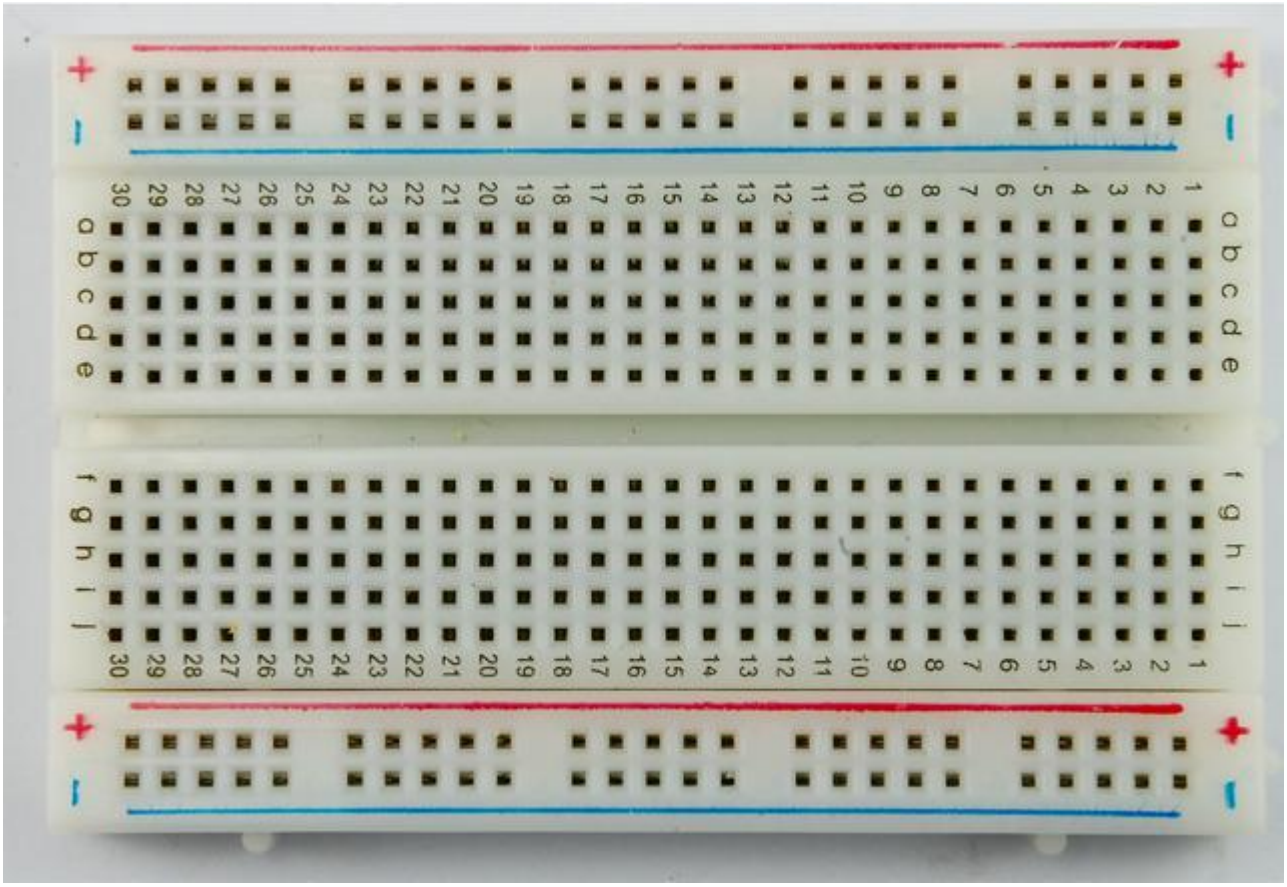


74HC595 Shift Register

1

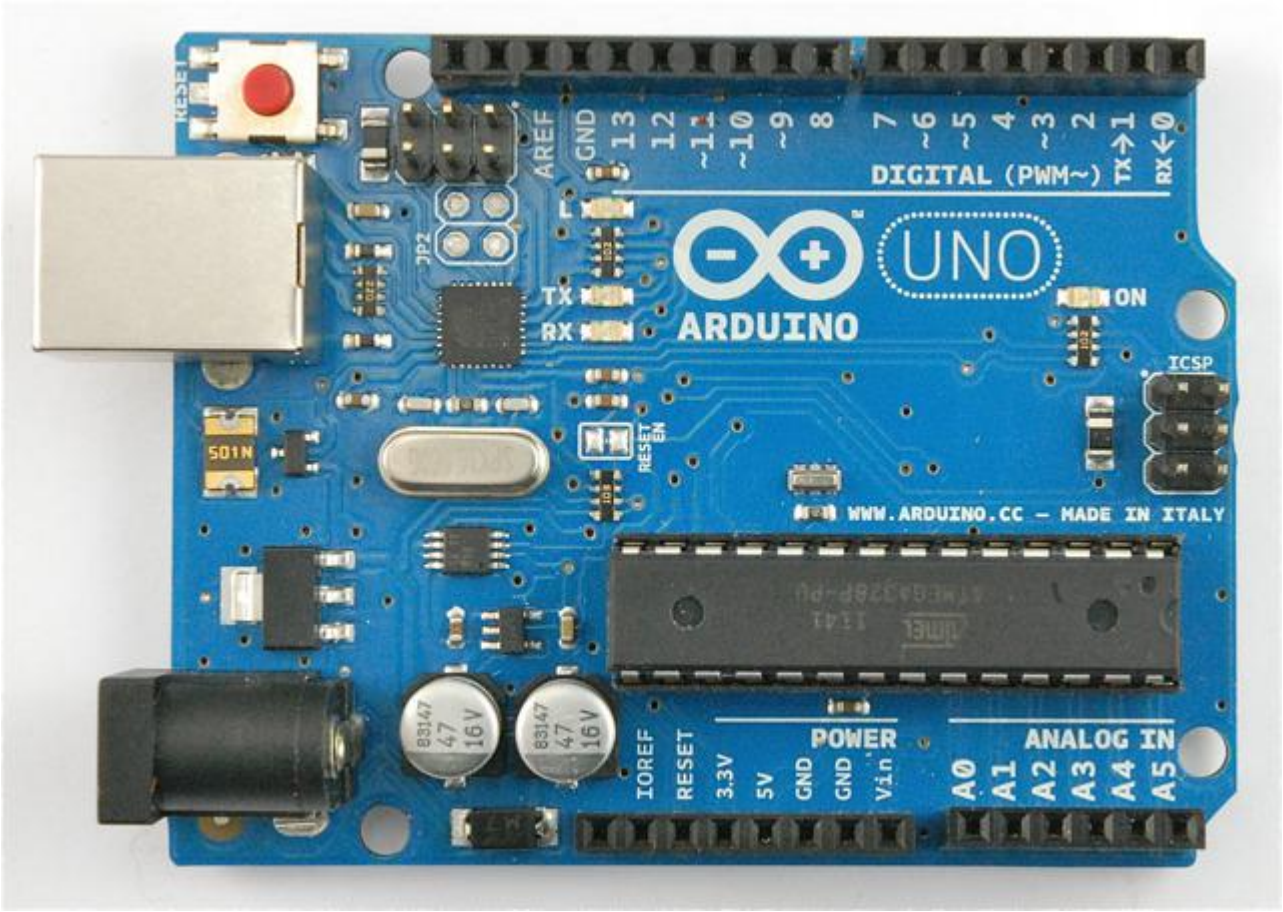
10 kΩ variable resistor (pot)

1



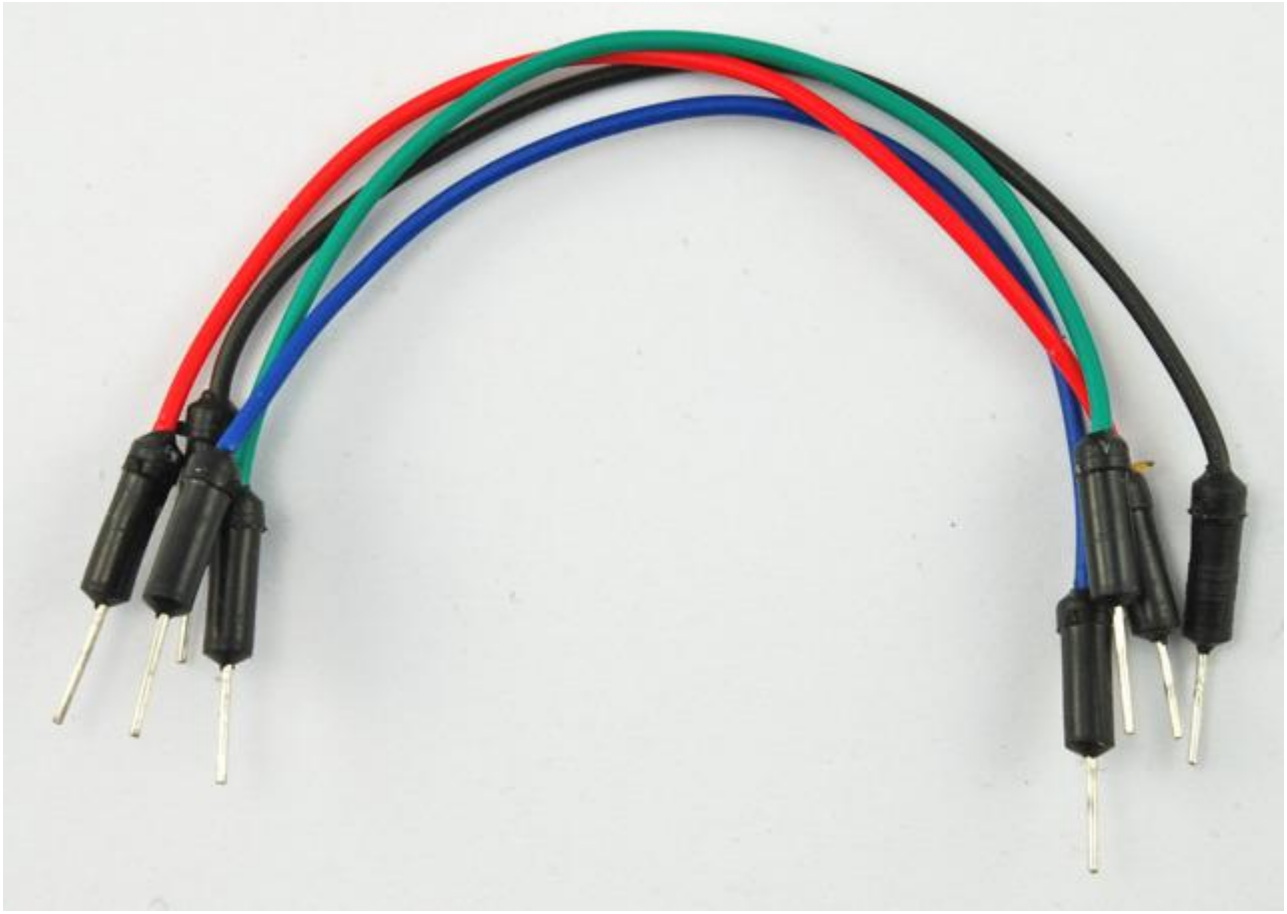
Half-size Breadboard

1



Arduino Uno R3

1



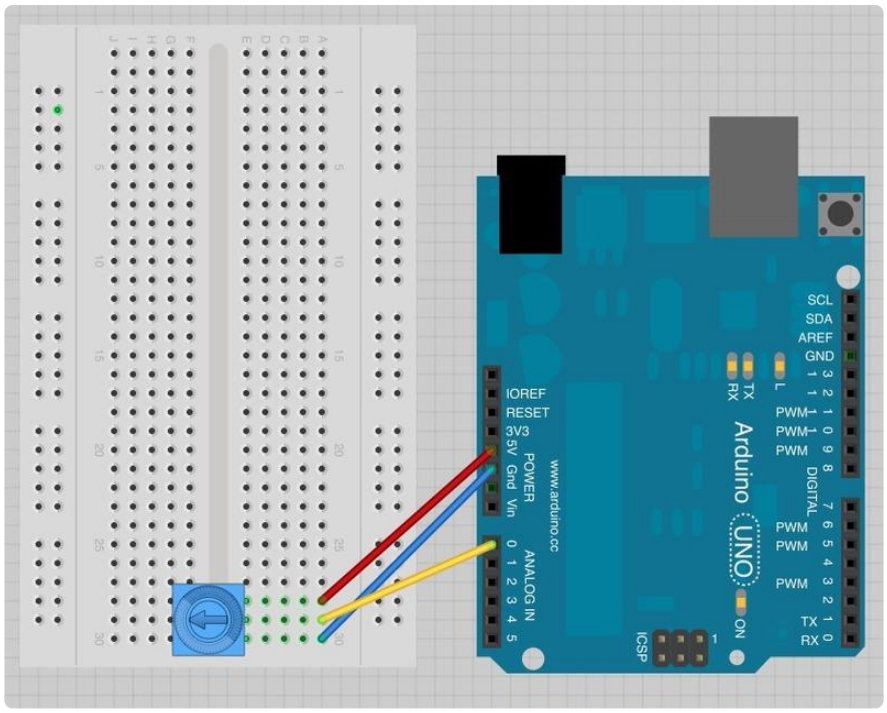
Jumper wire pack

1

An Experiment

Before we go ahead and use the LEDs, you can try a little experiment using just the variable resistor also known as a potentiometer (often called a 'pot' for short) and the Arduino Serial Monitor.

Connect up your breadboard as shown below:



Load the following sketch onto your Arduino.

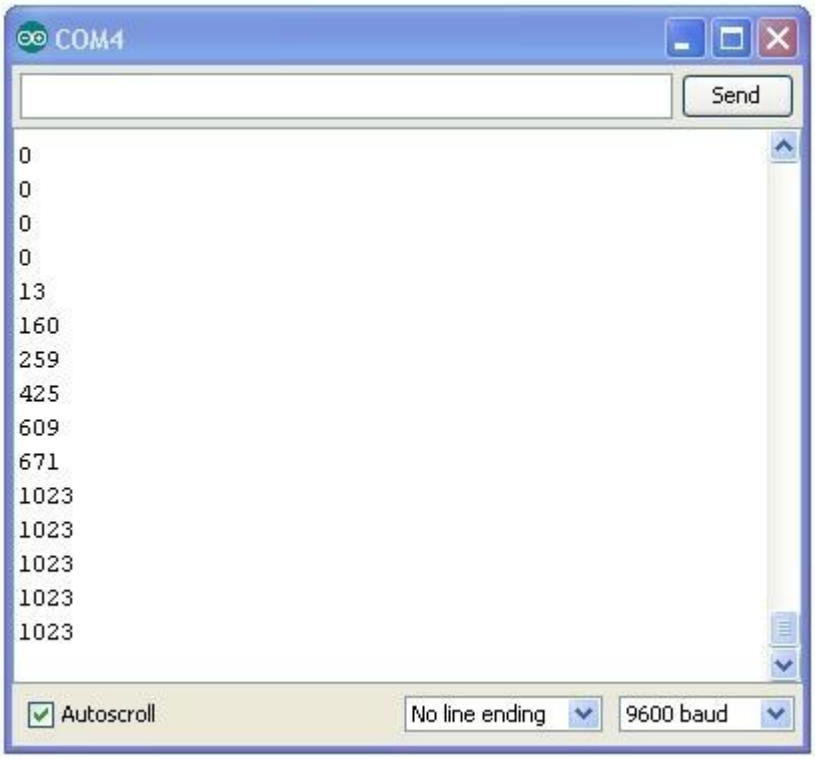
```
/*
Adafruit Arduino - Lesson 8. Analog Inputs
*/

int potPin = 0;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int reading  = analogRead(potPin);
  Serial.println(reading);
  delay(500);
}
```

Now open the Serial Monitor, and you will see a stream of numbers appearing.



Turn the knob on the variable resistor and you will see the number change between 0 and 1023.

The Serial Monitor is displaying the analog reading value from A0 using the line:

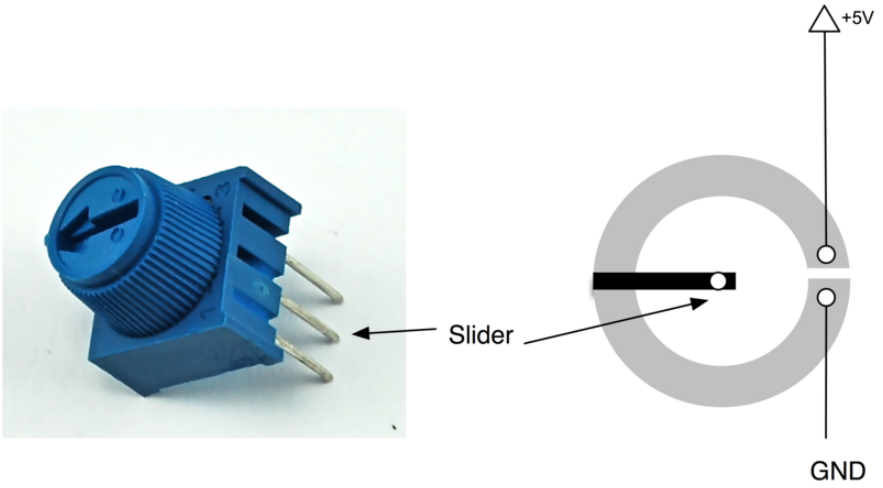
```
int reading  = analogRead(potPin);
```

The voltage at A0 is being transformed into a number between 0 and 1023.

Variable Resistors (Pots)

For historical reasons, variable resistors are often called 'pots' which is short for 'potentiometers'.

In our experiment with the Serial Monitor, the pot is somehow varying the voltage at A0 and the little test sketch is converting this voltage into a number between 0 and 1023.

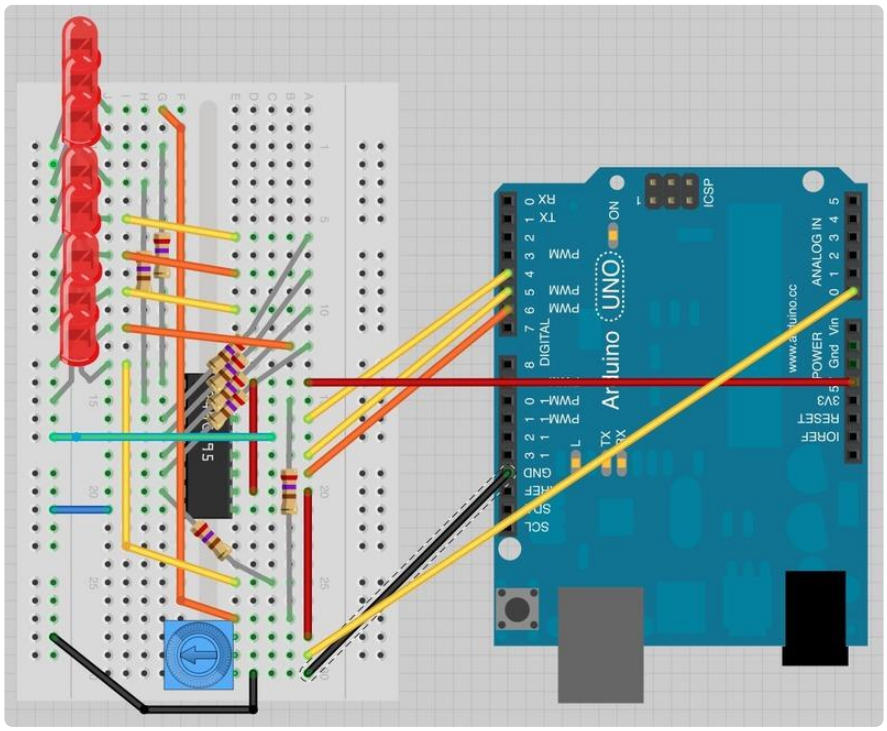


Your pot has a circular 'track' that acts as a resistor, in our case it's a 10 kΩ resistor. However, the difference with a pot, is that there is also a middle connection called the 'slider'. This connection is rotated when you turn the pot. So if you connect one end of the pot to 5V and the other to GND, then the voltage at the slider will vary between 0 and 5V as you turn it.

Breadboard Layout

Let's do something more interesting with the pot. We can use it to control the number of LEDs lit.

This breadboard layout is based on that of lesson 4, there are a few jumpers moved, and the pot and it's connections to the Arduino have been added.



Arduino Code

Load the following sketch onto your Arduino board.

```
/*
Adafruit Arduino - Lesson 8. Analog Inputs - LEDs
*/

int potPin = 0;
int latchPin = 5;
int clockPin = 6;
int dataPin = 4;

int leds = 0;

void setup()
{
  pinMode(latchPin, OUTPUT);
  pinMode(dataPin, OUTPUT);
  pinMode(clockPin, OUTPUT);
}

void loop()
{
  int reading  = analogRead(potPin);
  int numLEDSLit = reading / 114;  //1023 / 9
  leds = 0;
  for (int i = 0; i &lt; numLEDSLit; i++)
  {
    bitSet(leds, i);
  }
  updateShiftRegister();
}

void updateShiftRegister()
{
   digitalWrite(latchPin, LOW);
   shiftOut(dataPin, clockPin, LSBFIRST, leds);
   digitalWrite(latchPin, HIGH);
}
```

You should recognize much of this code from lesson 4. So refer back to that lesson for more information about how the LEDs are controlled.

The key parts of the sketch as far as analog inputs are concerned are the line where we define the analog pin that we are going to connect to the slider of the pot:

```
int potPin = 0;
```

Note that we do not need to put anything in 'setup' to set the pin mode for an analog input.

In the main loop, we read the analog value like this:

```
  int reading  = analogRead(potPin);
```

But then this reading between 0 and 1023 needs converting into a number of LEDs to light, between 0 and 8. The range of numbers, between 0 and 8 is actually 9 values. So we need to scale the reading by 1023 divided by 9 or 114.

```
int numLEDSLit = reading / 114;
```

To light the right number of LEDs, we use the 'for' loop to count from 0 up to 'numLEDSLit' setting the bit at that position.

```
  leds = 0;
  for (int i = 0; i &lt; numLEDSLit; i++)
  {
    bitSet(leds, i);
  }
```

Finally we update the shift register with a call to:

```
updateShiftRegister();
```

Other Things to Do

It is actually simpler to light a single LED to indicate the position of the knob. Try modifying your sketch to do this.

Click Here for the Next Lesson

About the Author

Simon Monk is author of a number of books relating to Open Source Hardware. The following books written by Simon are available from Adafruit: Programming Arduino (http://adafru.it/1019), 30 Arduino Projects for the Evil Genius (http://adafru.it/868) and Programming the Raspberry Pi ().