Starter Kit for Arduino

catalogue

Arduino IDE(Integrated Development Environment)

Introduction

The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.

****

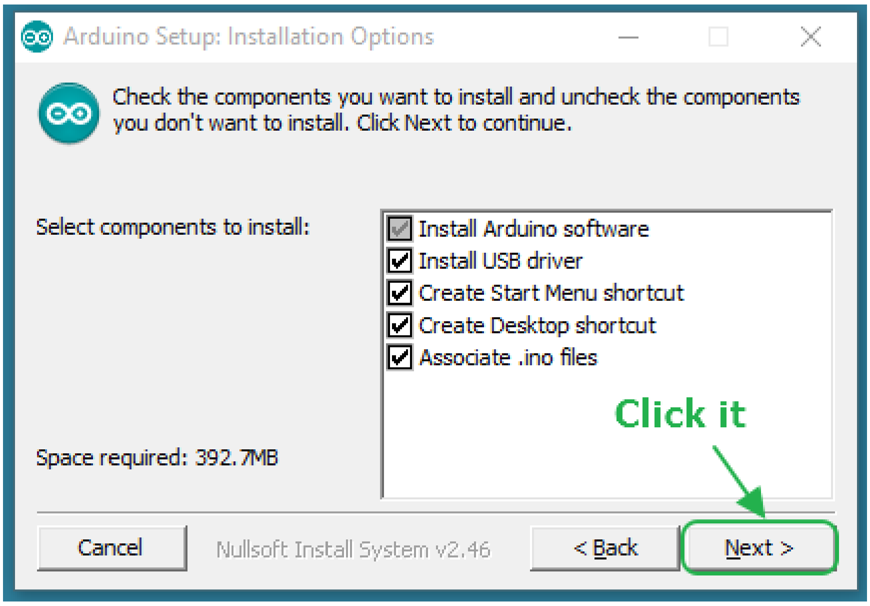
* About Elecrow:
* We are a leading manufacturer of electronic components for Arduino and Raspberry Pi.
* We have a professional engineering team dedicated to providing tutorials and support to help you get started.
* If you have any technical questions or suggestions, please feel free to contact our support staff via email at engle@elecrow.com
* We truly hope you enjoy the product, for more great products please visit
our company website: https://www.elecrow.com
or aliexpress store: https://www.aliexpress.com/store/1306340

Operation demo

Step 1: Install the Arduino Software (IDE)

Download the latest version from this page: http://arduino.cc/en/Main/Software

Next, proceed with the installation and please allow the driver installation process.



Arduino Setup: Installation Options

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

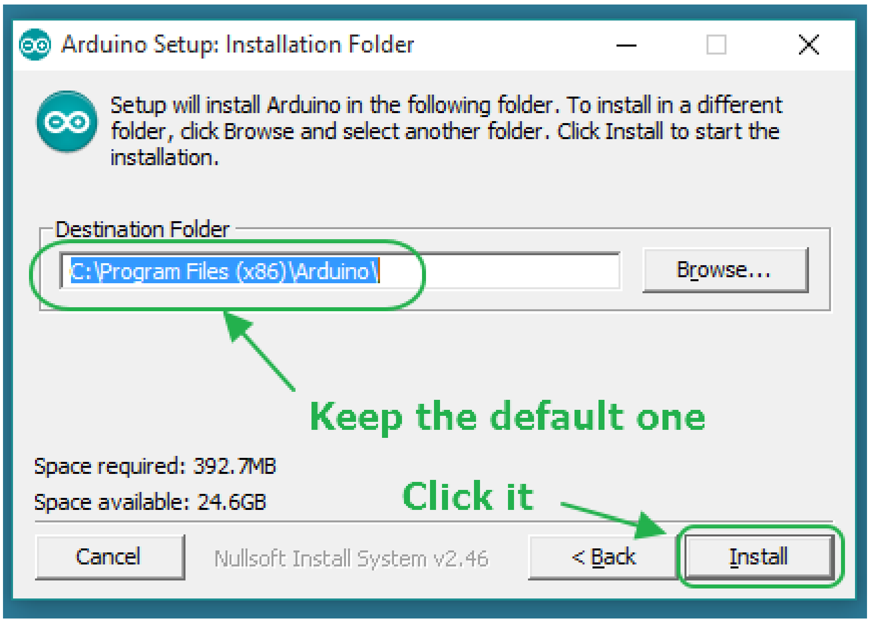
Select components to install:

- ☑ Install Arduino software
- ☑ Install USB driver
- ☑ Create Start Menu shortcut
- ☑ Create Desktop shortcut
- ☑ Associate .ino files

Space required: 392.7MB

Click it

Choose the components to install and click “next” button.



Arduino Setup: Installation Folder

Setup will install Arduino in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.

Destination Folder: C:\Program Files (x86)\Arduino\

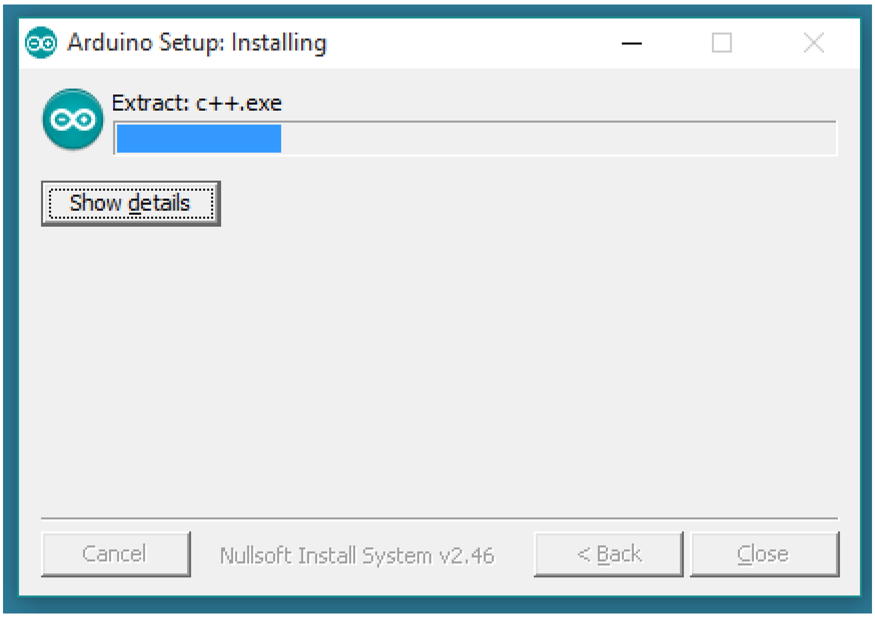
Keep the default one

Space required: 392.7MB

Space available: 24.6GB

Click it

Choose the installation directory.



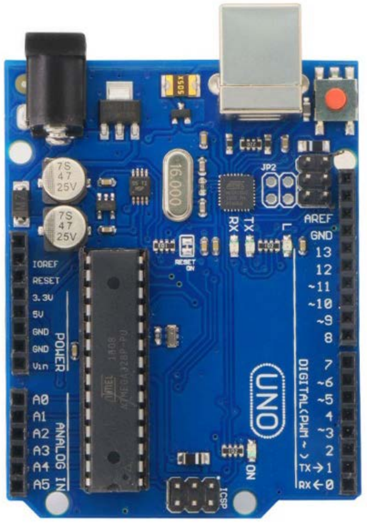
Arduino Setup: Installing

Extract: c++.exe

The process will extract and install all the required files to execute properly the Arduino Software (IDE)

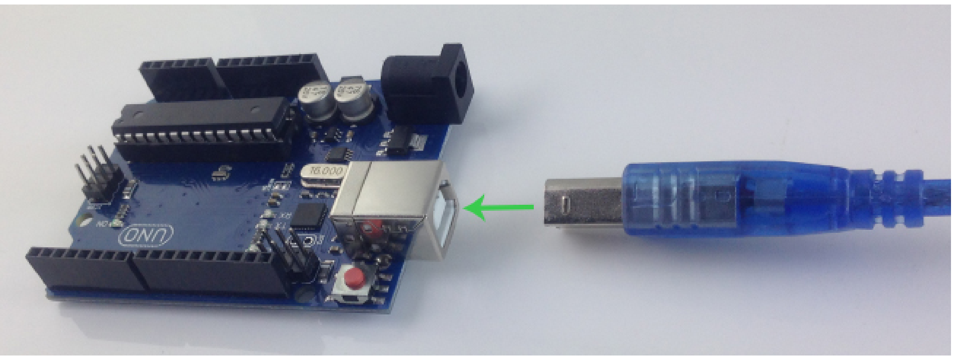
Step 2: Get an Uno R3 and USB cable

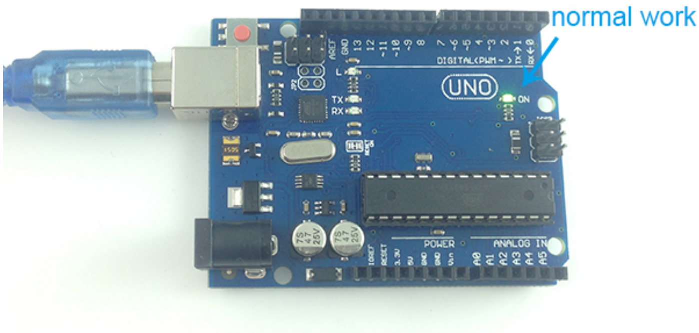
In this tutorial, you're using an Uno R3. You also need a standard USB cable (A plug to B plug): the kind you would connect to a USB printer, for example.



Step 3: Connect the board

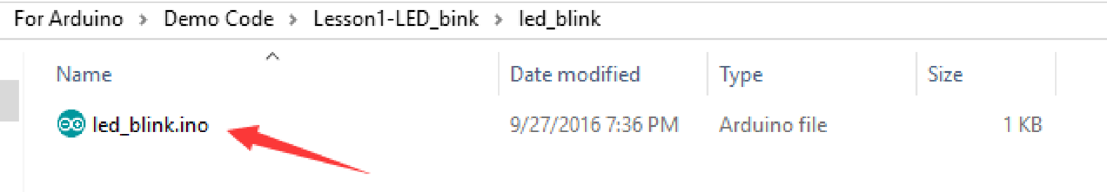
The USB connection with the PC is necessary to program the board and not just to power it up. The Uno and Mega automatically draw power from either the USB or an external power supply. Connect the board to your computer using the USB cable. **The green power LED (labelled PWR) should go on.**



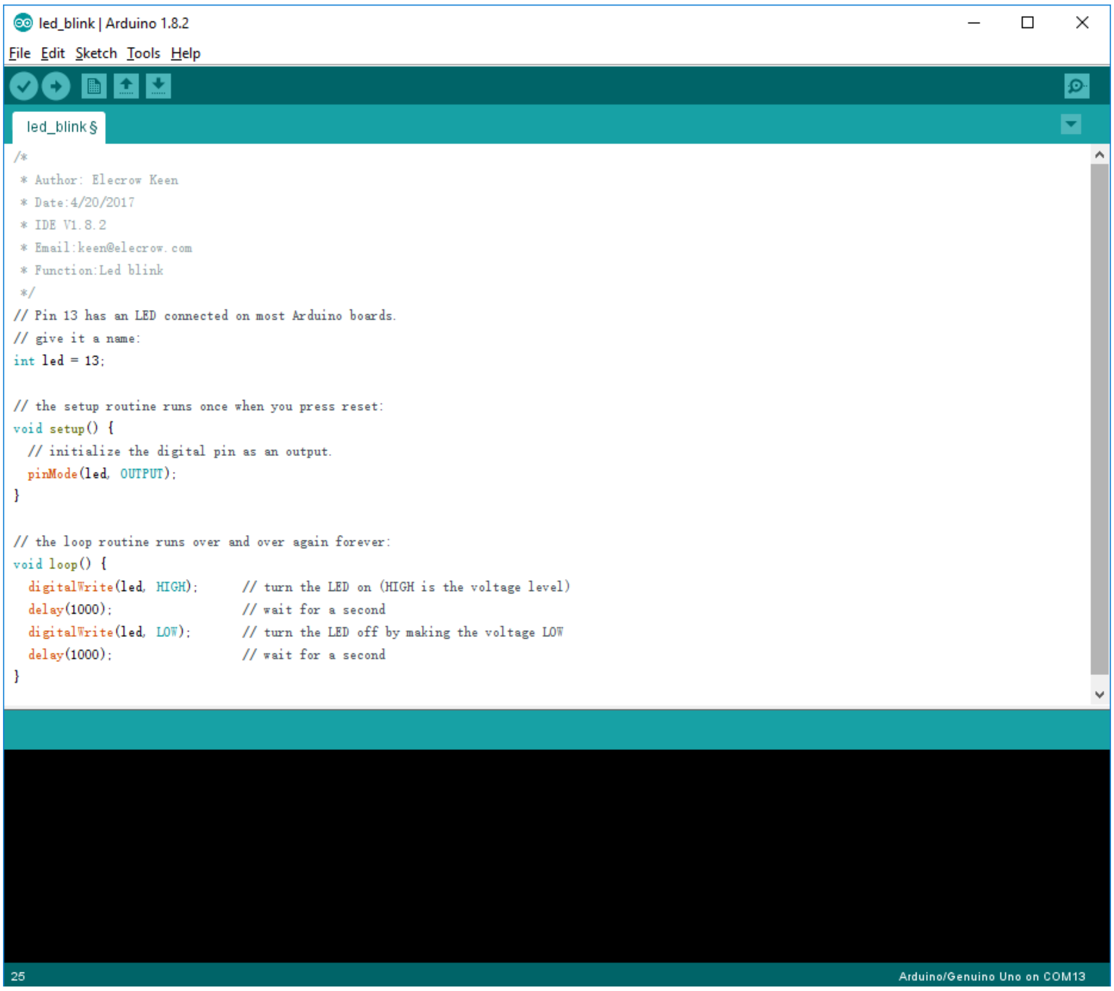


Step 4: Open Lesson 1: LED blink

Open the LED blink example sketch: CD>For Arduino>Demo Code>Lesson1-LED_bink>led_blink.



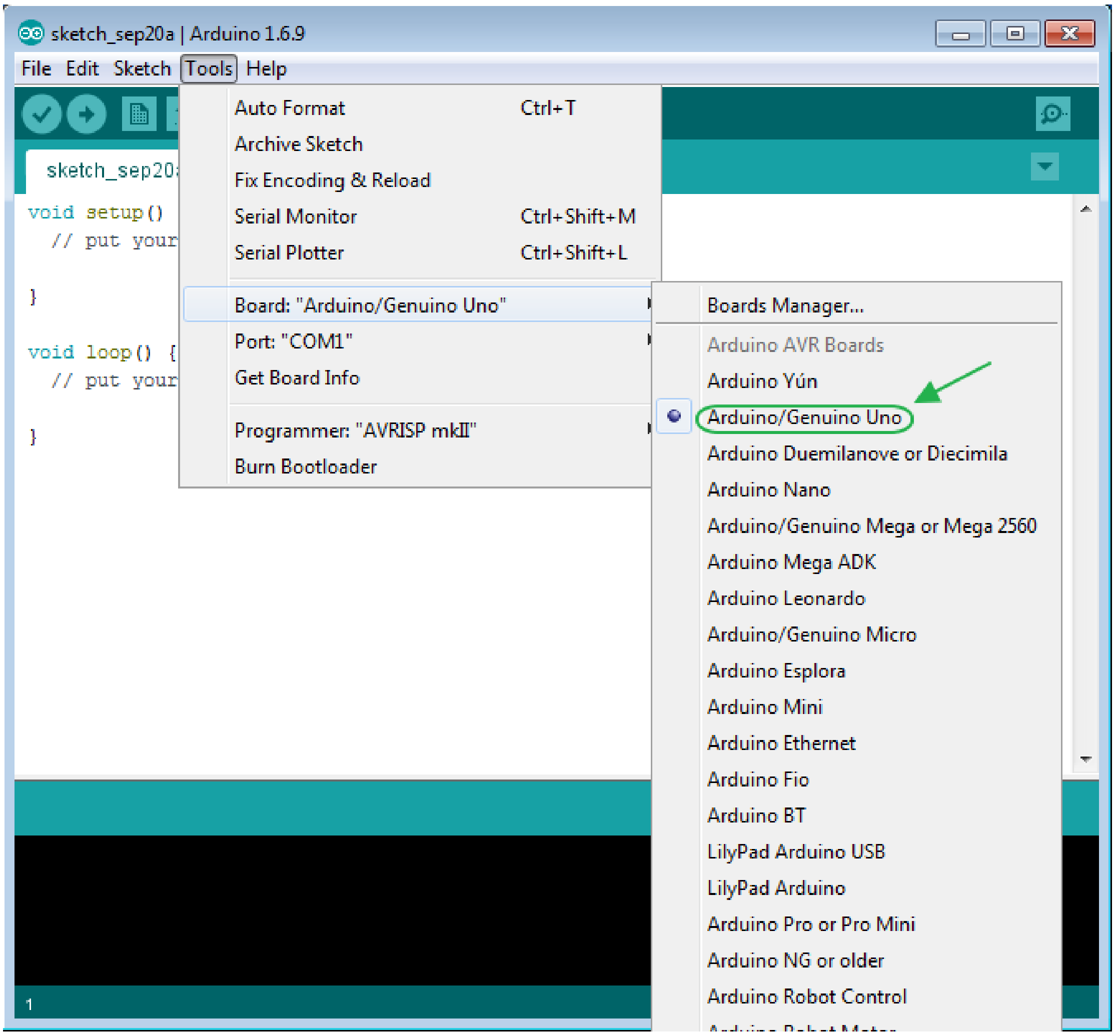
Name	Date modified	Type	Size
led_blink.ino	9/27/2016 7:36 PM	Arduino file	1 KB



```
/*
 * Author: Elecrow Keen
 * Date:4/20/2017
 * IDE V1.8.2
 * Email:keen@elecrow.com
 * Function:Led blink
 */
// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);      // turn the LED on (HIGH is the voltage level)
  delay(1000);                  // wait for a second
  digitalWrite(led, LOW);       // turn the LED off by making the voltage LOW
  delay(1000);                  // wait for a second
}
```

Step 5: Select your board

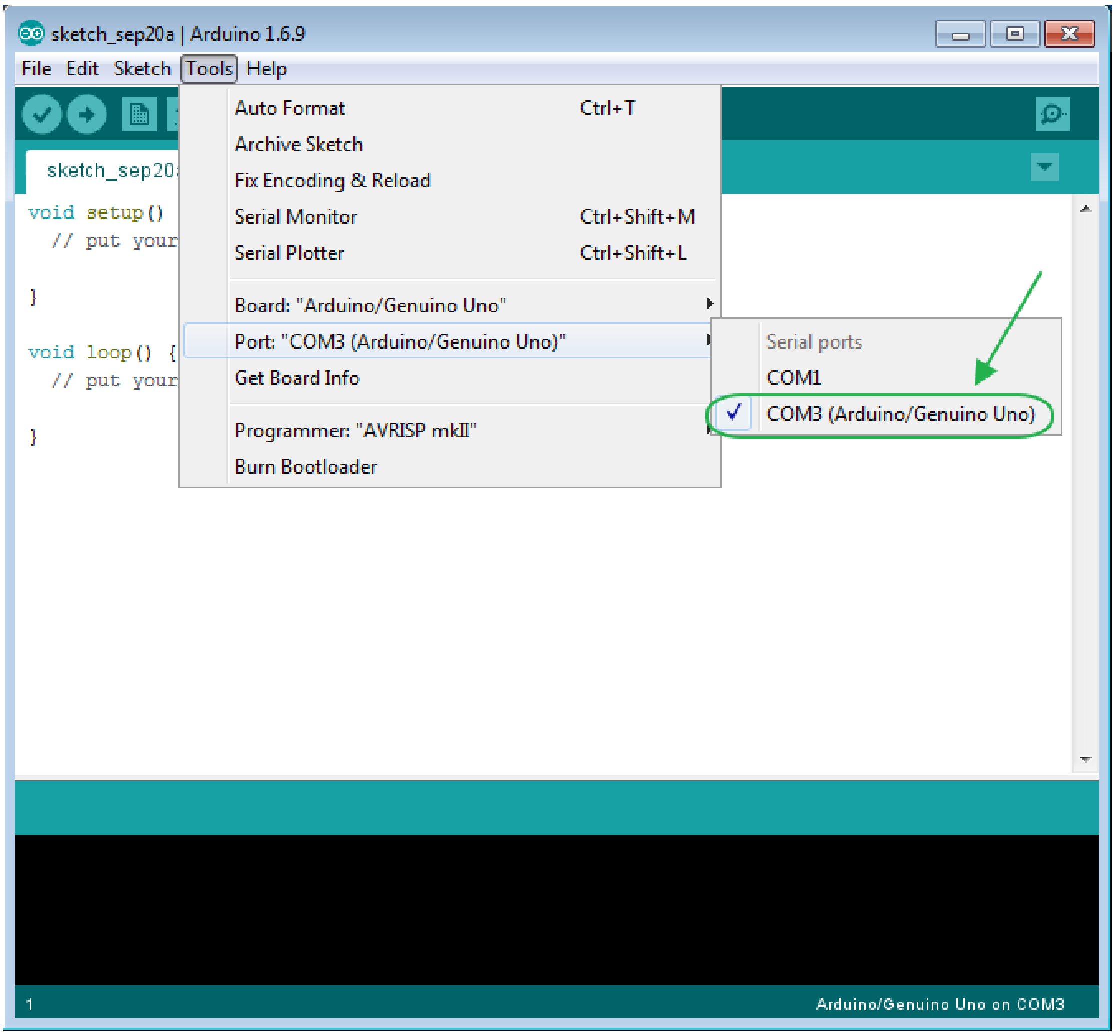
You'll need to select the entry in the Tools > Board menu that corresponds to your Arduino board.



```
void setup() {
  // put your
}

void loop() {
  // put your
}
```

Selecting an Arduino/Genuino Uno.

Step 6: Select your serial port


sketch_sep20a | Arduino 1.6.9
File
Edit
Sketch
Tools
Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
Board: "Arduino/Genuino Uno"
Port: "COM3 (Arduino/Genuino Uno)"
Get Board Info
Programmer: "AVRISP mkII"
Burn Bootloader
Serial ports
COM1
✓ COM3 (Arduino/Genuino Uno)
sketch_sep20

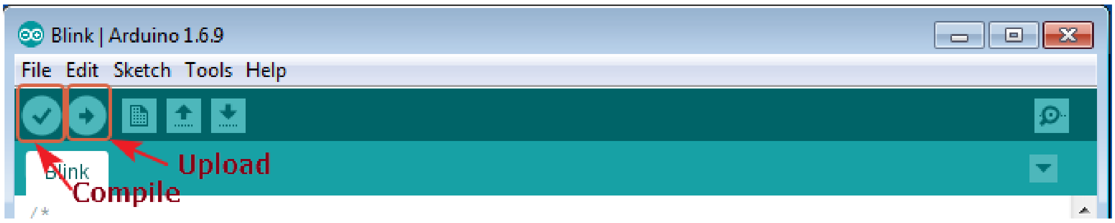
```
void setup()
  // put your

}

void loop() {
  // put your

}
```

1 Arduino/Genuino Uno on COM3

Select the serial device of the board from the Tools | Serial Port menu. This is likely to be COM3 or higher (COM1 andCOM2 are usually reserved for hardware serial ports). To find out, you can disconnect your board and re-open the menu; the entry that disappears should be the Arduinoboard. Reconnect the board and select that serial port.

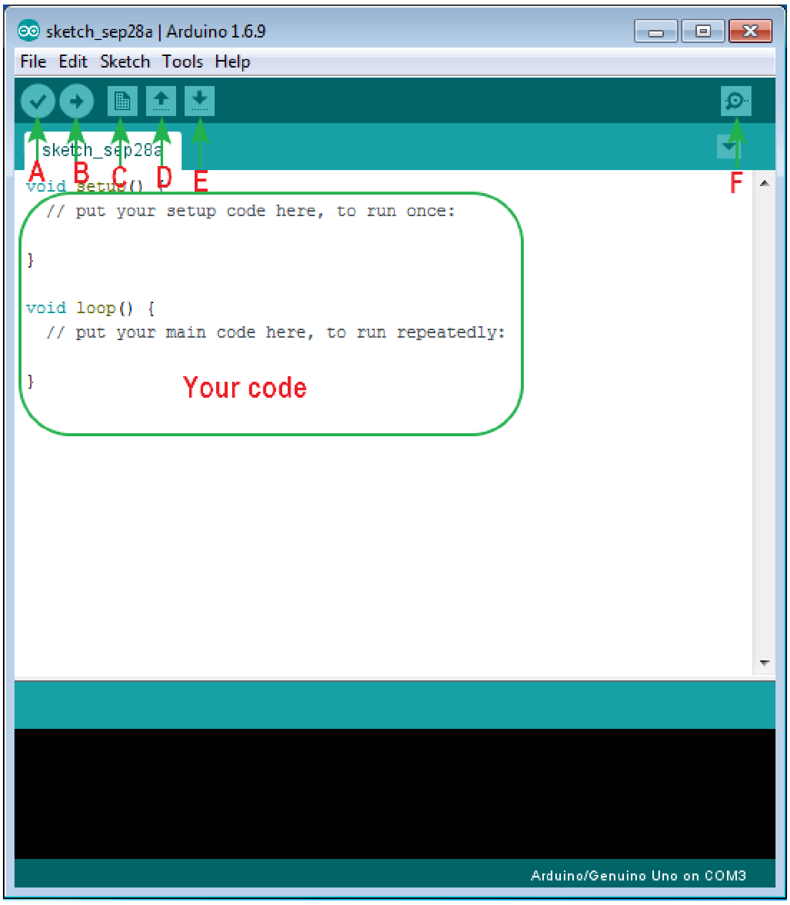
Step 7: Upload the program

Now, simply click the "Upload" button in the environment. Wait a few seconds - you should see the RX and TX leds on the board flashing. If the upload is successful, the message "Done uploading." will appear in the status bar.



Step 8: Result

A few seconds after the upload finishes, you should see the pin 13 (L) LED on the board start to blink (in orange). If it does, congratulations! You've gotten Arduino up-and-running.

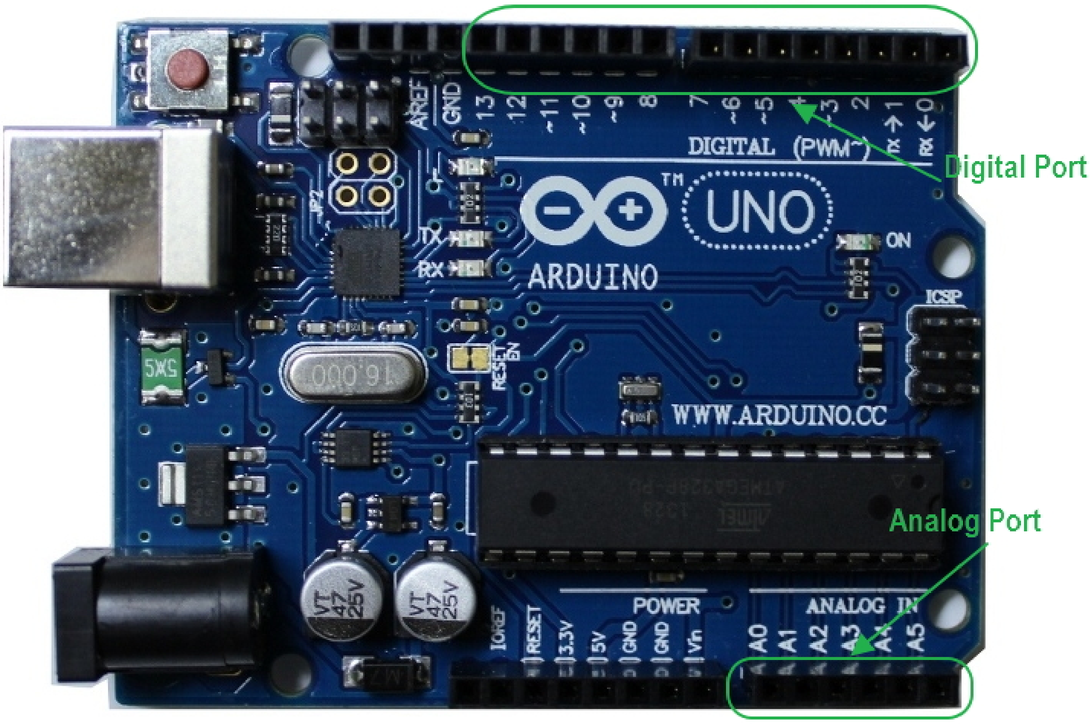

sketch_sep28a | Arduino 1.6.9
File Edit Sketch Tools Help
sketch_sep28a
A
B
C
D
E
F
Your code
Arduino/Genuino Uno on COM3

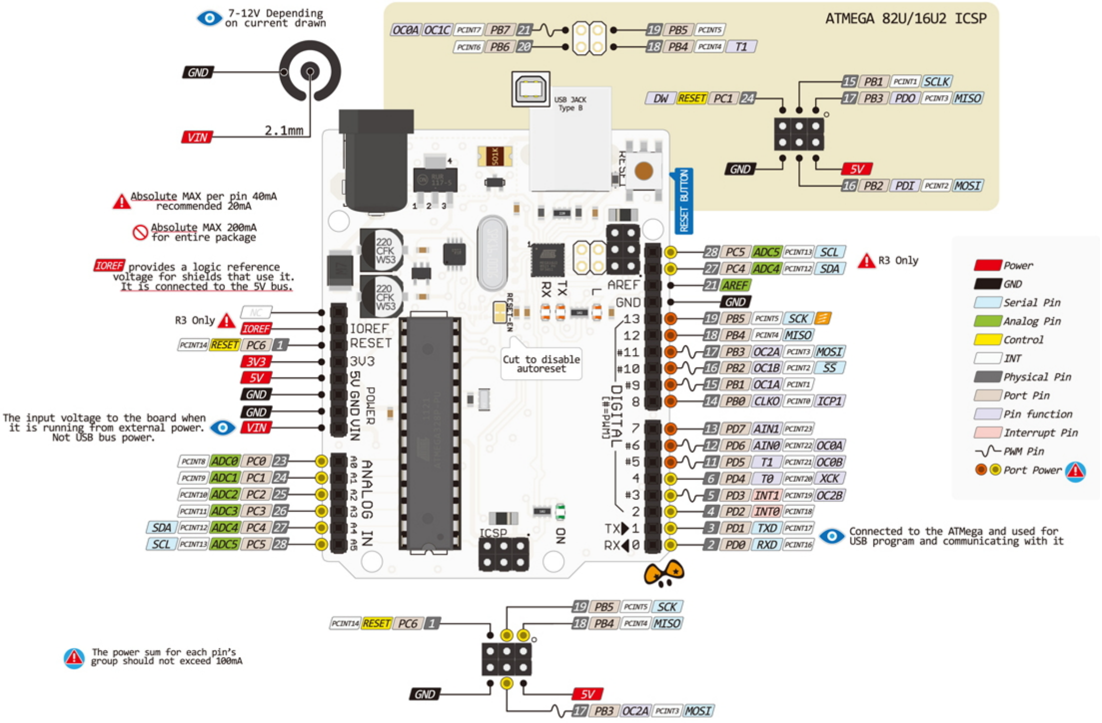
```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Arduino interface introduction

A	**->Compile**
B	**->Upload**
C	**->New**
D	**->Open**
E	**->Save**
F	**->Serial monitor**

Arduino UNO R3hardware introduction

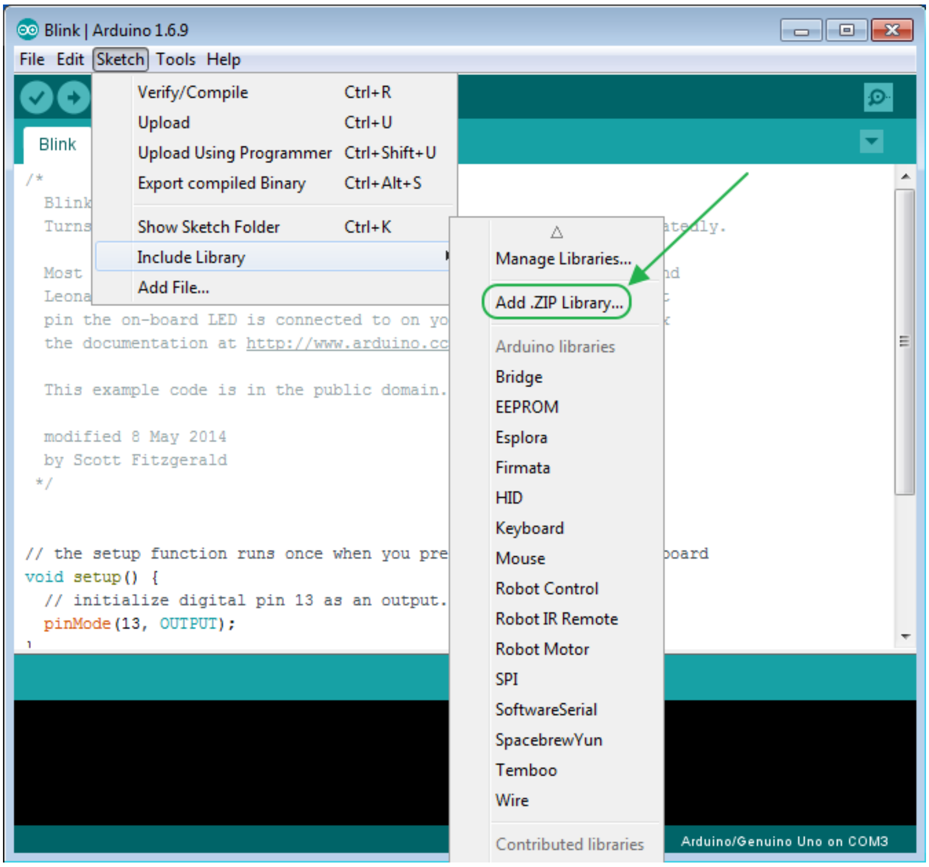




How to add library files

Step 1:

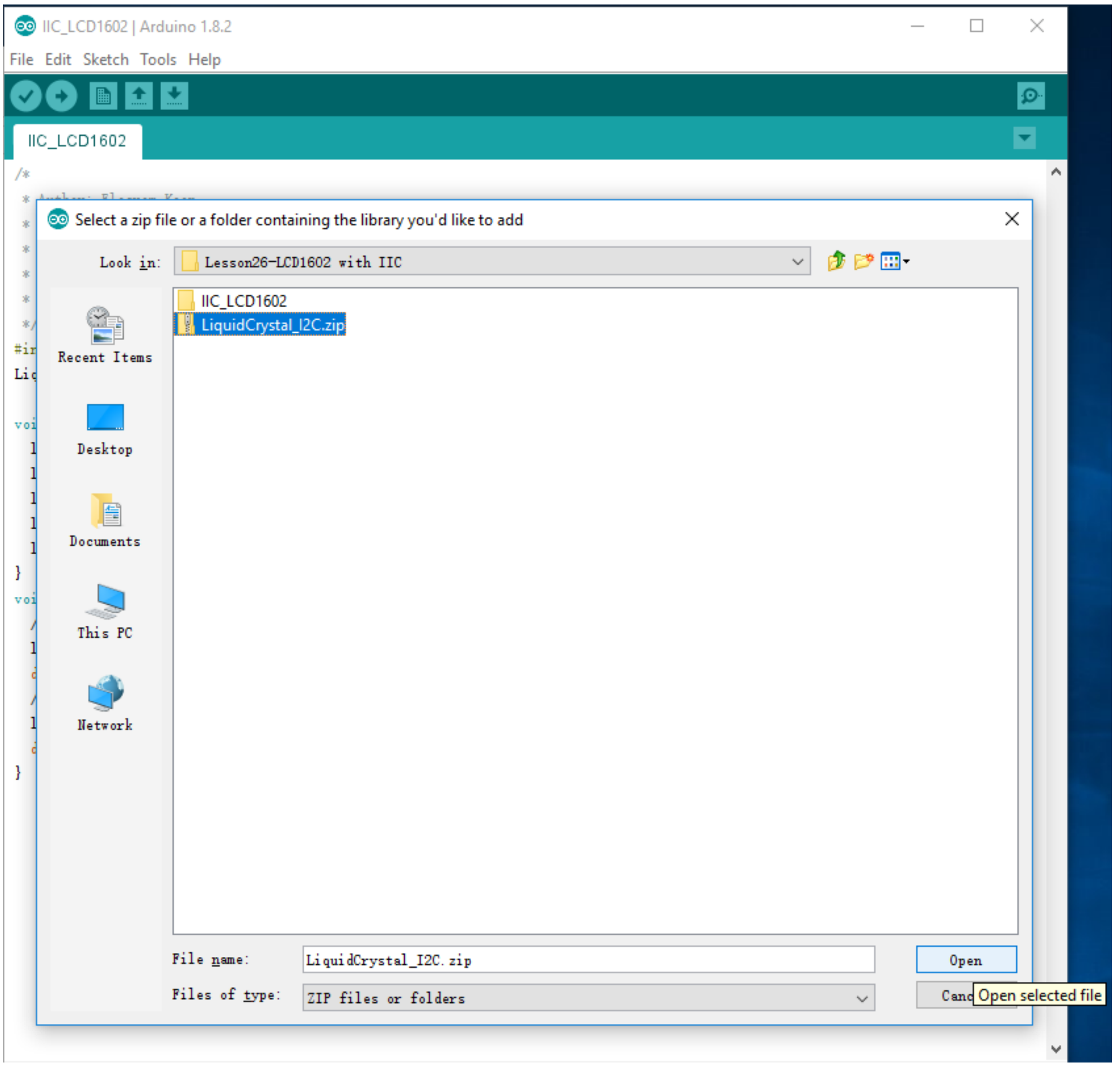
Add library file: Sketch>Include Library>Add.ZIP Library



```
/*
  Blink
  Turns                                           atedly.

  Most                                            nd
  Leona                                           c
  pin the on-board LED is connected to on yo      k
  the documentation at http://www.arduino.cc

  This example code is in the public domain.

  modified 8 May 2014
  by Scott Fitzgerald
 */


// the setup function runs once when you pre      board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
```

Step 2:

Select your library file compression package on the demo code file, as follows：



IIC_LCD1602 | Arduino 1.8.2

Select a zip file or a folder containing the library you'd like to add

Look in: Lesson26-LCD1602 with IIC

IIC_LCD1602

LiquidCrystal_I2C.zip

File name: LiquidCrystal_I2C.zip

Files of type: ZIP files or folders

Step3: Finish

Learning materials

Ebook

Introduction

The E-book about Arduino what we provided for you is carefully selected and comprehensive, it specially aims at solving the problems when you make projects such as syntax analysis, program optimization and so on.If you have any questions about the projects what we provided,you can also refer the content of e-books.

Path:\For Arduino\Ebook

Language Reference

https://www.elecrow.com/wiki/

http://wiring.org.co/reference/

https://www.arduino.cc/en/Reference/HomePage/

Lessons

Introduction

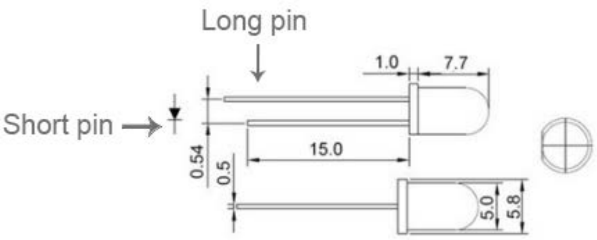
We will provide you not only the all involved courses about this kit but also to analyze each course. We sincerely hope that you can learn from the first course to the last course because it will lead you start with Arduino step by step, and it also let you jump from a newbieto a higher level for developing your own independent projects.

Lesson 1: LED Blink

Overview

The LED is designed for the beginners of Arduino. Blinking LED experiment is quite simple and it is the best choice to help you learn I/O pins. On this lesson, we are going to connect an LED to one of the digital pins.

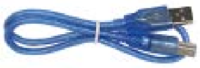
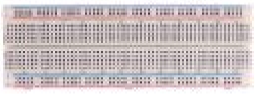
Specification



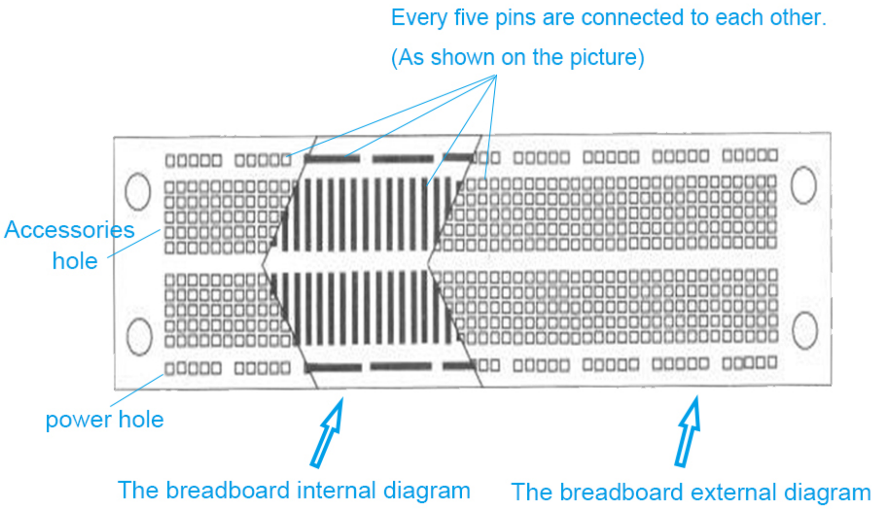
Pin definition

LED		UNO R3
Long pin	->	+5V
Short pin	->	GND

Hardware required

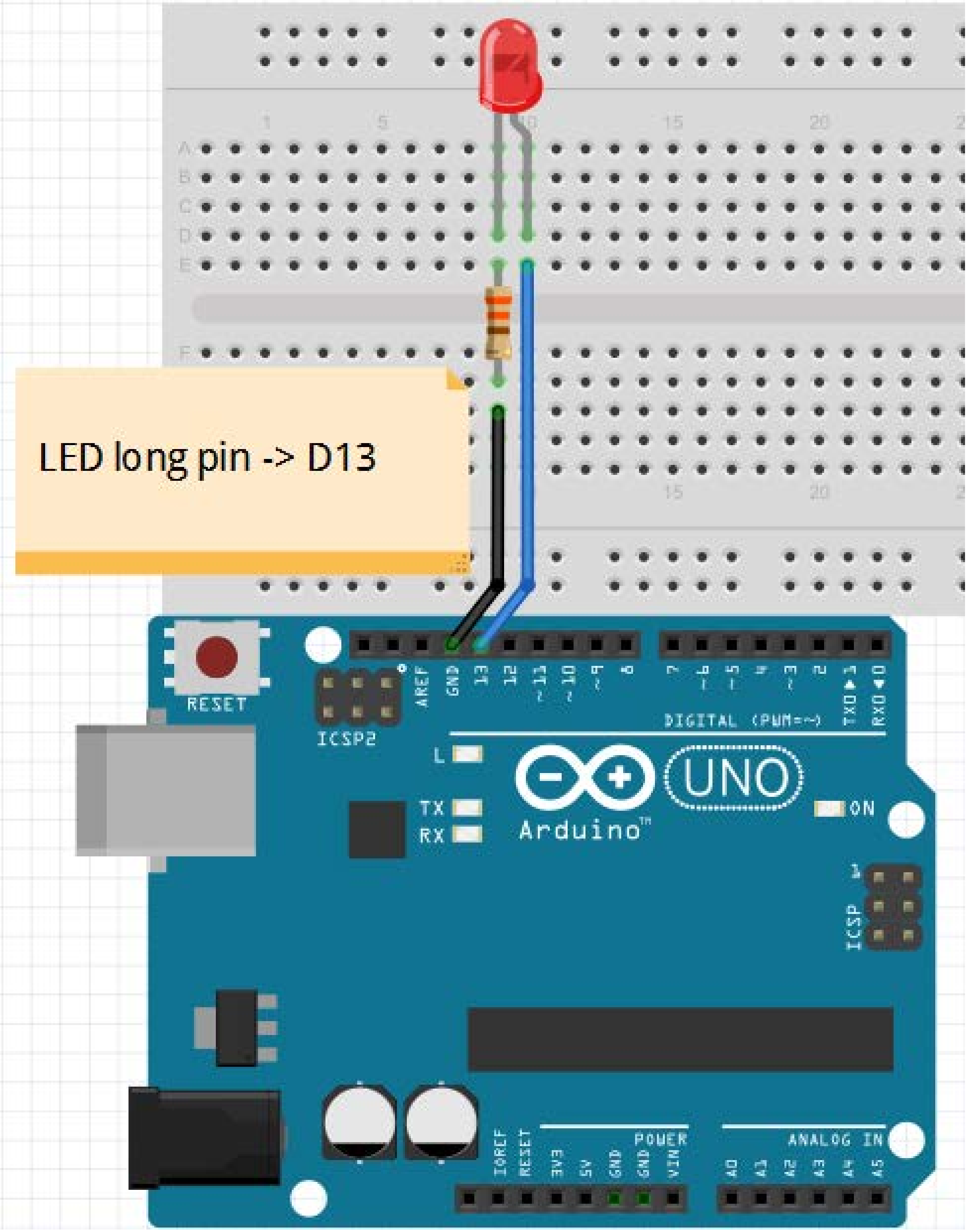
Material diagram	Material name	Number
	220/330Ω resistor	1
	LED	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Bread board schematic


Every five pins are connected to each other.
(As shown on the picture)
Accessories
hole
power hole
The breadboard internal diagram
The breadboard external diagram

All the tie points (indicated in the picture) of the different colors are connected together.

Connection diagram



LED long pin -> D13

Note：The longest LED of the pin is connected to the digital signal port 13(D13).

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips：click on the following name to jump to the web page.

If you fail to open, use the Adobe reader to open this document.

int

setup()

pinMode()

OUTPUT

loop()

HIGH

LOW

digitalWrite()

digitalRead()

delay()

; (semicolon)

{} (curly braces)

= (assign)

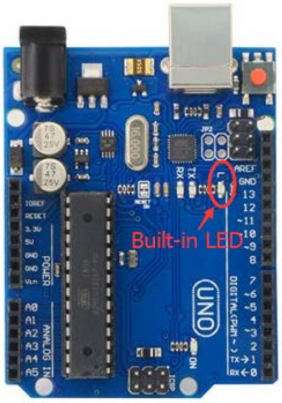
// (comment)

Application effect

Turns on an LED on for one second, then off for one second, repeatedly.

Lesson 2: Button

Overview



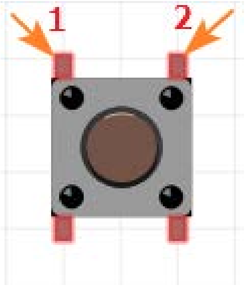
Button switches, familiar to most of us, are a switch value (digital value) component. When it's pressed, the circuit is in closed (conducting) state. This example turns on the built-in LED on pin 13 when you press the button.

Specification

Size: 6 x 6 x 5mm
Temperature: -30 ~ +70 Centigrade

Pin definition

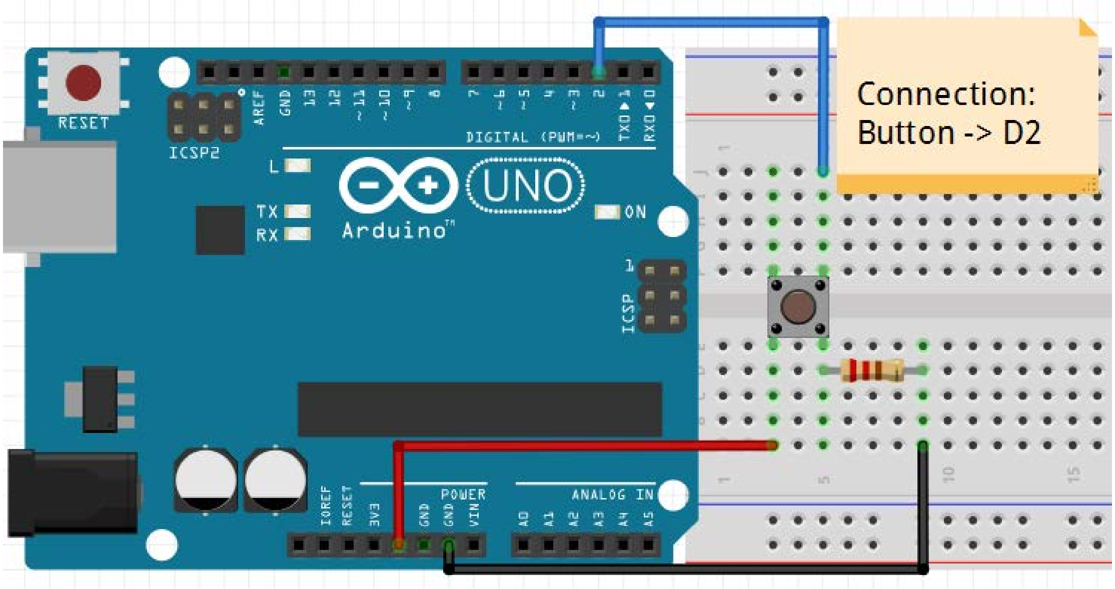
It is the definition of Button pin：



Hardware required

Material diagram	Material name	Number
	Button	1
	10KΩ resistor	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Connect three wires to the board. The first two, red and black, connect to the two long vertical rows on the side of the breadboard to provide access to the 5 volt supply and ground. The third wire goes from digital pin 2 to one leg of the pushbutton. That same leg of the button connects through a pull-down resistor (here 10K ohm) to ground. The other leg of the button connects to the 5 volt supply.

When the pushbutton is open (unpressed) there is no connection between the two legs of the pushbutton, so the pin is connected to ground (through the pull-down resistor) and we read a LOW. When the button is closed (pressed), it makes a connection between its two legs, connecting the pin to 5 volts, so that we read a HIGH.

You can also wire this circuit the opposite way, with a pullup resistor keeping the input HIGH, and going LOW when the button is pressed. If so, the behavior of the sketch will be reversed, with the LED normally on and turning off when you press the button.

If you disconnect the digital I/O pin from everything, the LED may blink erratically. This is because the input is "floating" - that is, it will randomly return either HIGH or LOW. That's why you need a pull-up or pull-down resistor in the circuit.

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips：click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
const
INPUT

Application effect

When you press the button, the built-in LED will light up, release is extinguished.

Lesson 3: Active Buzzer

Overview

This is an active buzzer experiment. It has an inner vibration source and the direct power supply can make a sound.

Specification

Voltage: DC 5V
Min Sound Output at 10cm: 85dB;
Total Size (Pin Not Included): 12 x 9mm/0.47" x 0.35"(D*H)

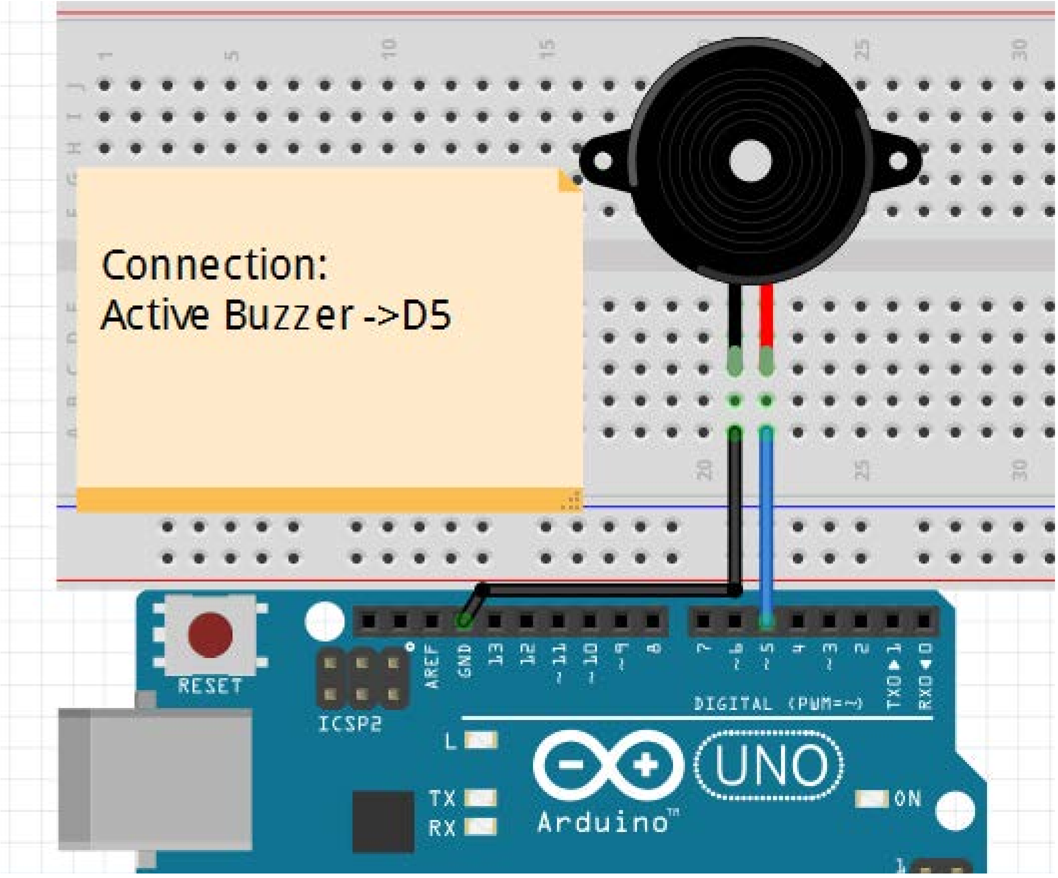
Pin definition

Active Buzzer		UNO R3
Long pin	->	D5
Short pin	->	GND

Hardware required

Material diagram	Material name	Number
	Active buzzer	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Note：The longest active buzzer of the pin is connected to the digital signal port 5 (D5).

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips：click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
digitalWrite()
pinMode()

Application effect

When the upload process is complete, the buzzer rings.

Lesson 4: Passive Buzzer

Overview

This is an Passive buzzer **experiment.** It cannot be actuated by itself, but external pulse frequencies. Different frequencies produce different sounds. We can use Arduino to code the melody of a song, which is actually quite fun and simple.

Specification

Working Voltage: 3V/5V
Resistance: 16Ohm
Resonance Frequency: 2KHZ

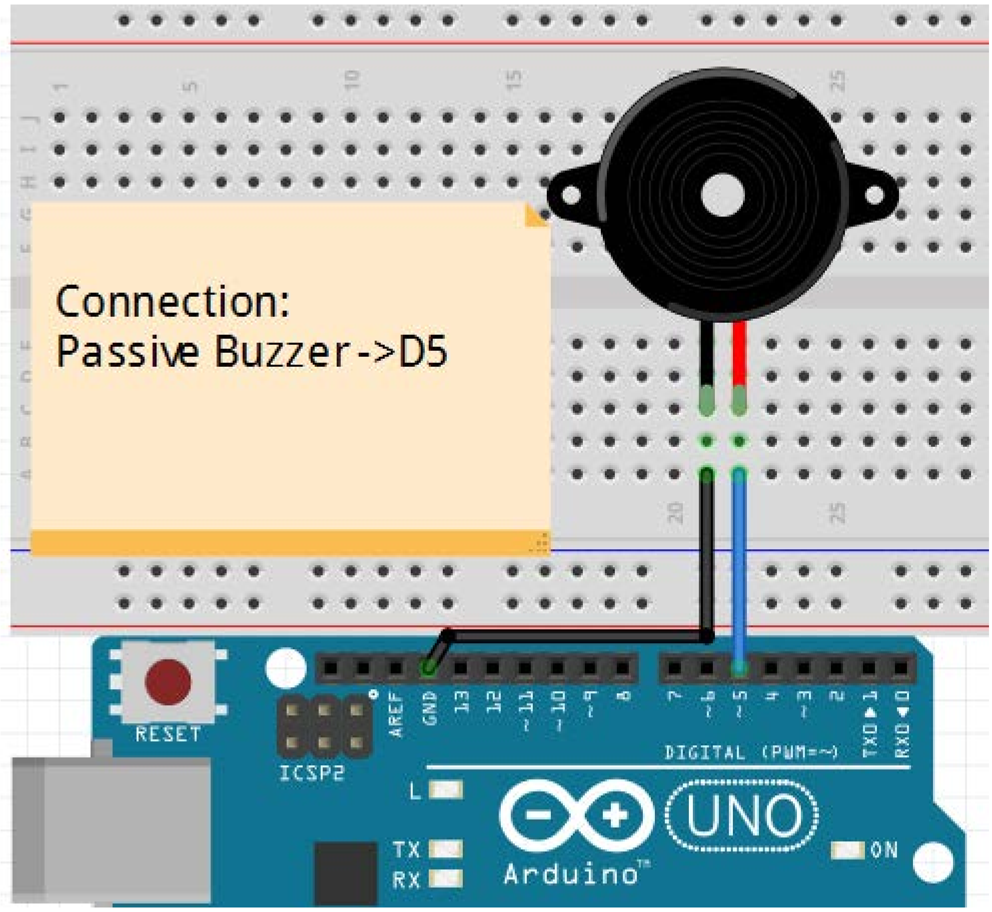
Pin definition

Passive Buzzer		UNO R3
Long pin	->	D5
Short pin	->	GND

Hardware required

Material diagram	Material name	Number
	Passive buzzer	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips： click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
#define
tone()

Application effect

When the upload process is complete, the buzzer sounds for 2 seconds.

Lesson 5: RGB LED

Overview

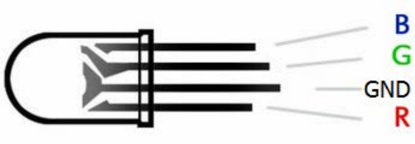
In this lesson, you will learn how to use a RGB (Red Green Blue) LED with an Arduino. You will use the analogWrite function of Arduino to control the color of the LED.

Specification

Emitting Light Color: Blue, Red, Green
Size(Approx): 5 x 35mm/ 0.2" x 1.37" (D * L)
Forward Voltage: 3.0-3.4V
Luminous Intensity: 12000-14000mcd

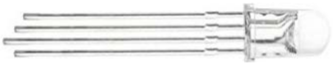
Pin definition

It is the definition of RGB LED pin：

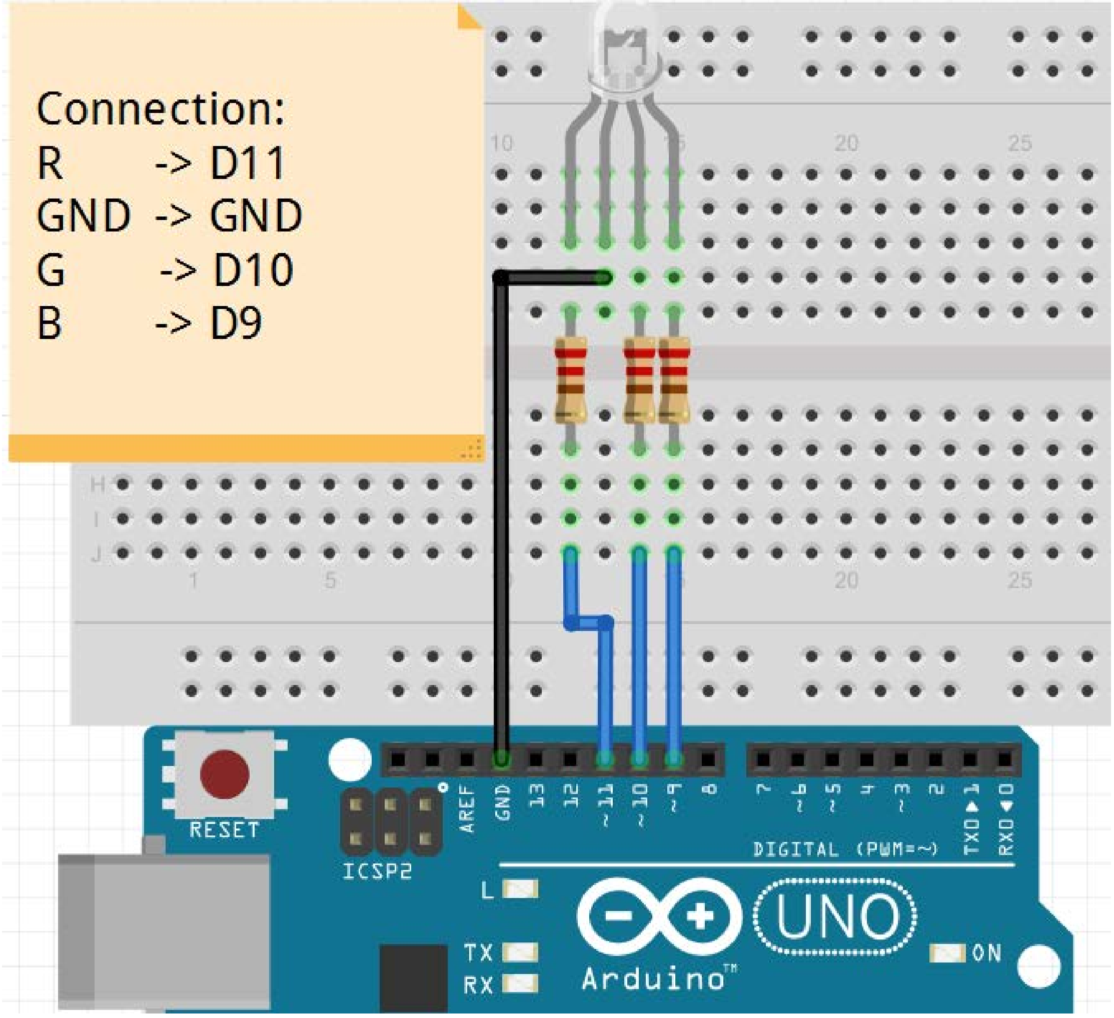


RGB LED		UNO R3
R	->	D11
GND	->	GND
G	->	D10
B	->	D9

Hardware required

Material diagram	Material name	Number
	RGB LED	1
	220Ω/330Ωresistor	3
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Note:The longest pin of the RGB LED is connected to the GND.

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

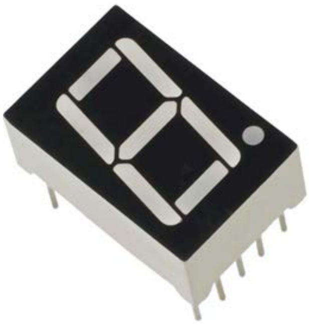
Tips：click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
analogWrite()
#define

Application effect

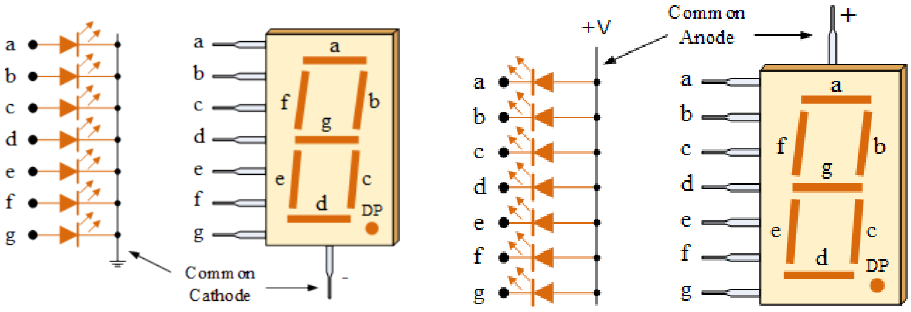
When the program is uploaded, you will see the LED loop emit 7 different colors of light.

Lesson 6: 1 Digit 7 Segment Displays

Overview



The 7-segment display, also written as “seven segment display”, consists of seven LEDs (hence its name) arranged in a rectangular fashion as shown. Each of the seven LEDs is called a segment because when illuminated the segment forms part of a numerical digit (both Decimal and Hex) to be displayed.



In general, common anode displays are more popular as many logic circuits can sink more current than they can source. Also note that a common cathode display is not a direct replacement in a circuit for a common anode display and vice versa, as it is the same as connecting the LEDs in reverse, and hence light emission will not take place.

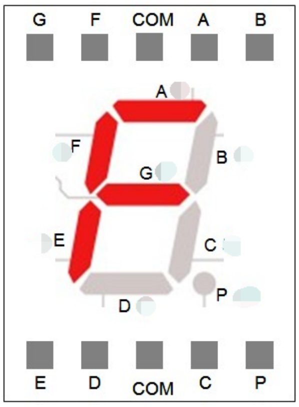
One this experiment, we use the 7-segment display (common cathode) to achieve time counting function.

Each segment of the display consists of an LED.   So when you use it, you also need use a current-limiting resistor.

Specification

Null

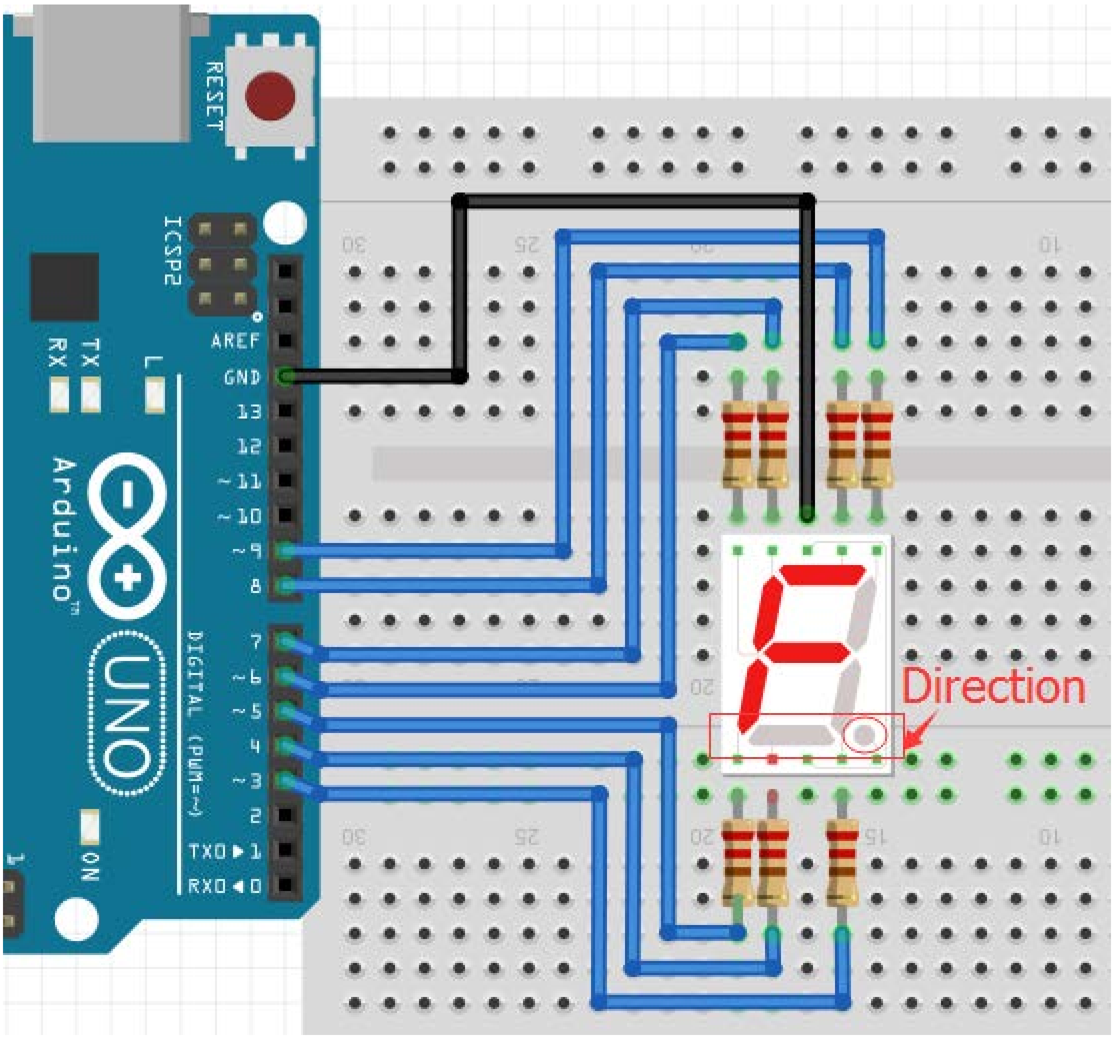
Pin definition



Hardware required

Material diagram	Material name	Number
	1 digit LED Segment Displays(common cathode)	1
	220/330Ω resistor	
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Note：Pay attention to the direction of digital tube.

Connection:

UNO R3		SEG
D3	->	C
D4	->	D
D5	->	E
D6	->	G
D7	->	F
D8	->	A
D9	->	B
GND	->	COM

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

array

Application effect

You will see the number on the digital tube increased from 0 to 9.

Lesson 7: 4 Digit 7 Segment Displays

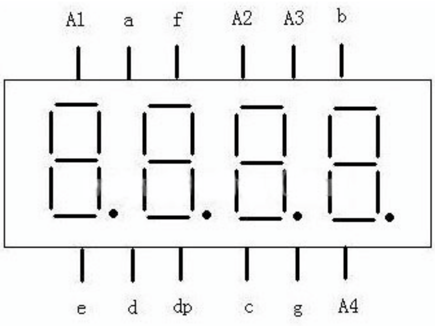
Overview

In this tutorial I will be showing you how to use a 4 digit 7 segment display to achieve time counting function.

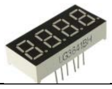
Specification

Null

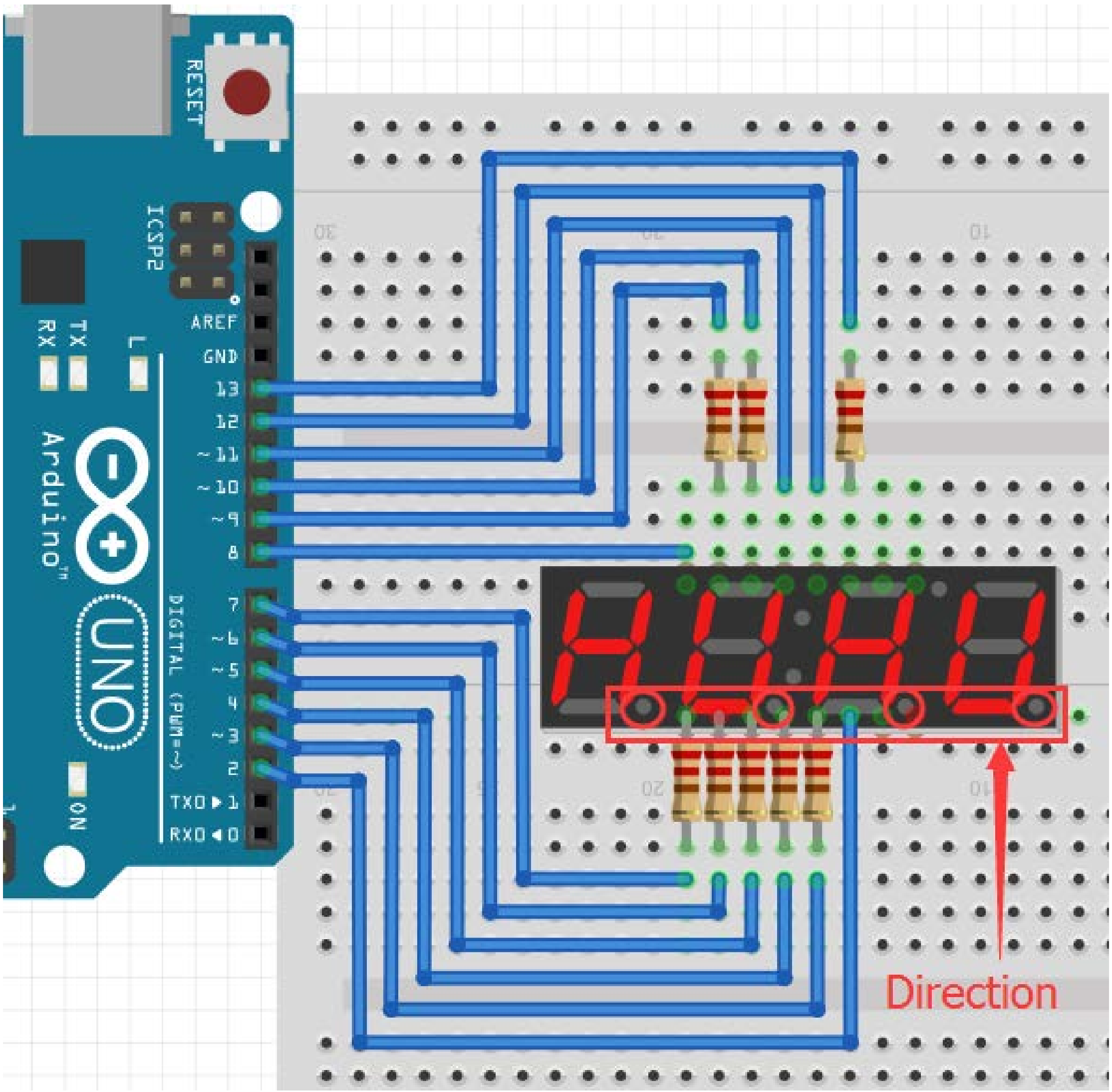
Pin definition



Hardware required

Material diagram	Material name	Number
	4 digit LED Segment Displays(common anode)	1
	220/330Ω resistor	8
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Note： Pay attention to the direction of digital tube.

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Long
switch()
case

Application effect

The time counting function, you will see the number of digital tube display increasingly.

Lesson 8: 74HC595 and Flow Led Experiment

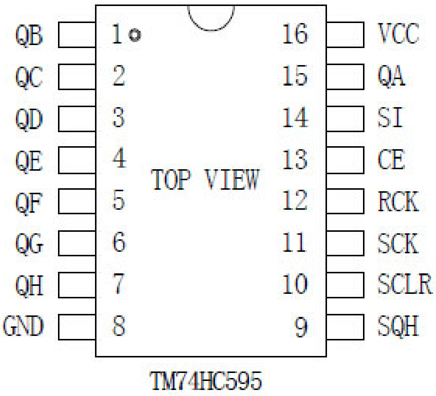
Overview

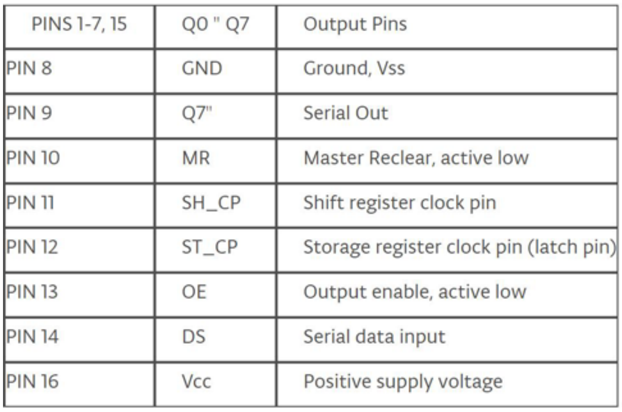
The 74HC595 consists of an 8−bit shift register and a storage register with three−state parallel outputs. It converts serial input into parallel output so you can save IO ports of an MCU. You can read and understand the pin diagram of 74HC595 to learn more.
In this leeson, you can just use 3 I/O ports to control 8 LED works.

Specification

Please view 74HC595-datasheet.pdf
Path: \Public_materials\Datasheet\74HC595-datasheet.pdf

Pin definition



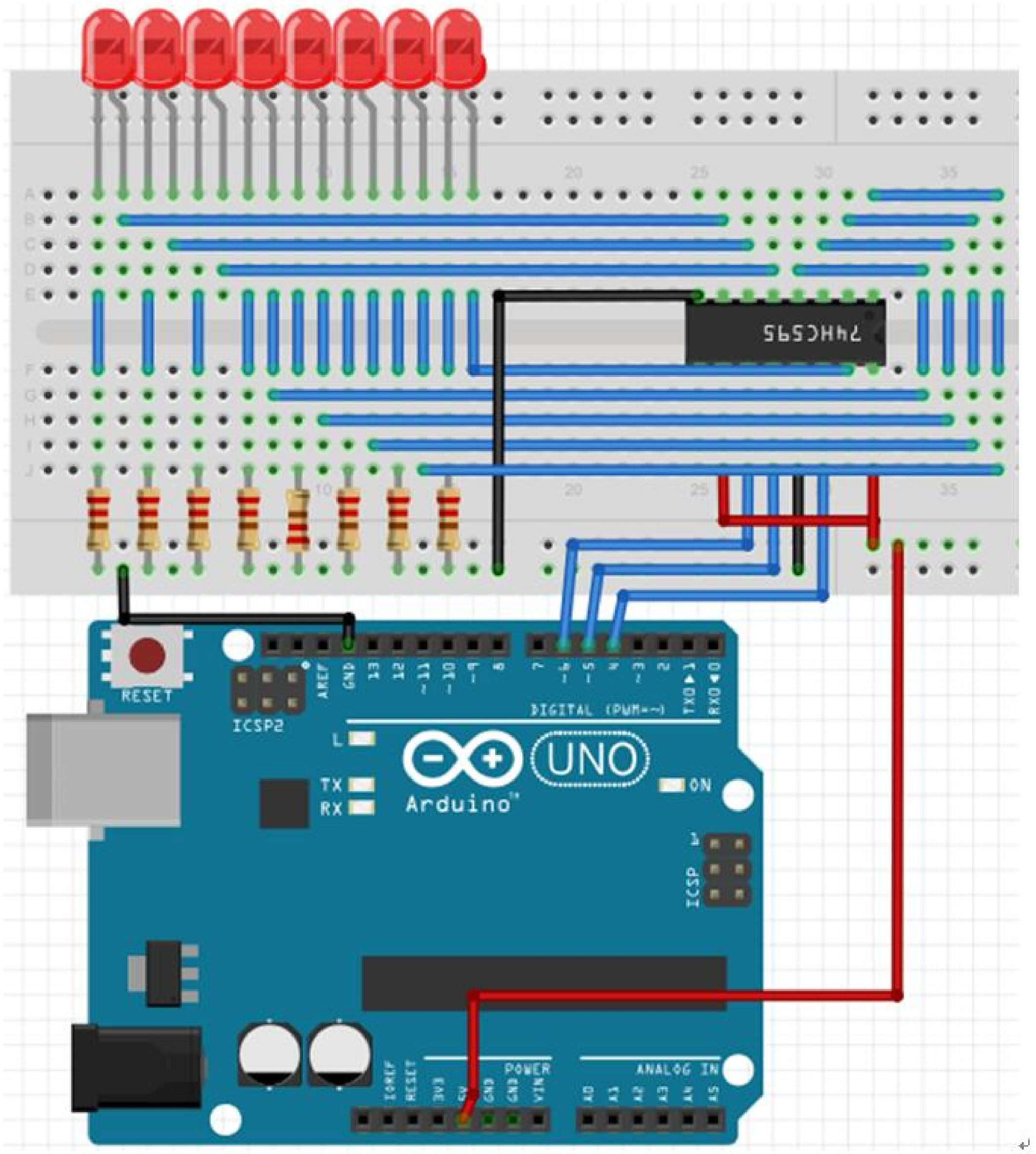


PINS 1-7, 15	Q0 " Q7	Output Pins
PIN 8	GND	Ground, Vss
PIN 9	Q7"	Serial Out
PIN 10	MR	Master Reclear, active low
PIN 11	SH_CP	Shift register clock pin
PIN 12	ST_CP	Storage register clock pin (latch pin)
PIN 13	OE	Output enable, active low
PIN 14	DS	Serial data input
PIN 16	Vcc	Positive supply voltage

Hardware required

Material diagram	Material name	Number
	74HC595	1
	LED	8
	220/330Ω resistor	8
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

bitset();
shiftOut();

Application effect

Only need 3 I/O ports can be used to control the eight LED and you can see 8 LED will flashing like flow water.

Lesson 9: LCD1602 with IIC

Overview

This lesson will teach you how to use LCD1602 with IIC.

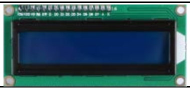
Specification

Please view LCD1602-datasheet.pdf and PCF8574.pdf.
Path:\Datasheet\

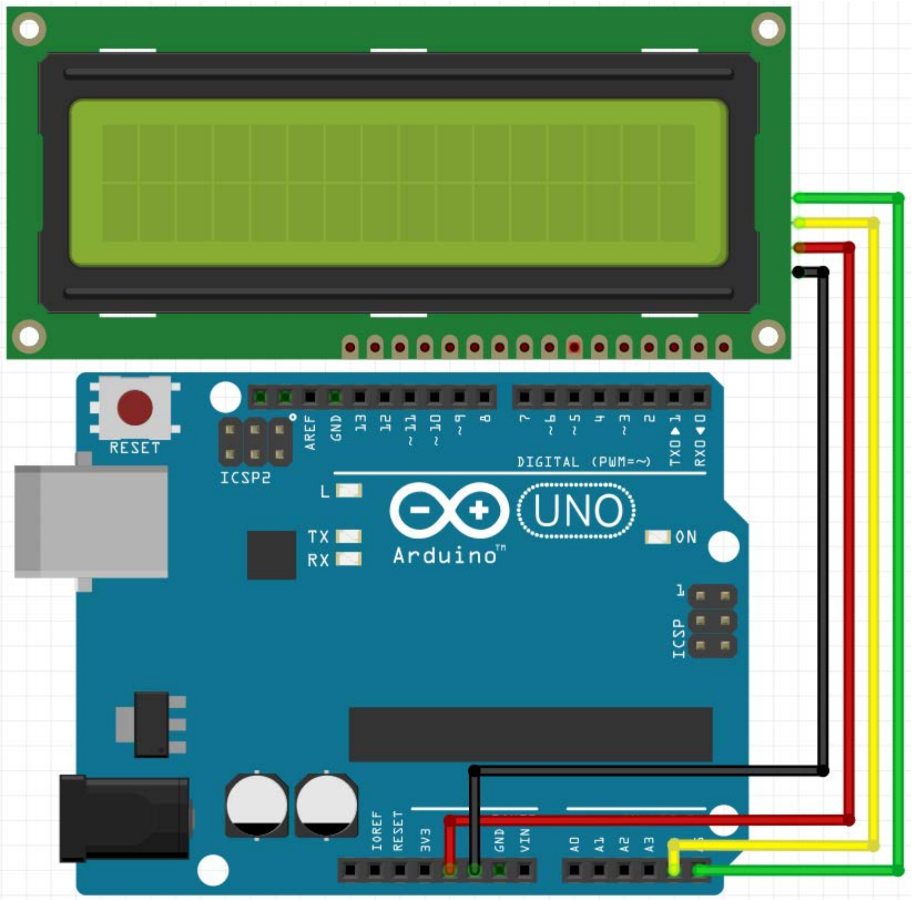
Pin definition

Null.

Hardware required

Material diagram	Material name	Number
	LCD1602_IIC	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



UNO R3		LCD1602_IIC
GND	->	GND
+5V	->	VCC
SDA	->	A4
SCL	->	A5

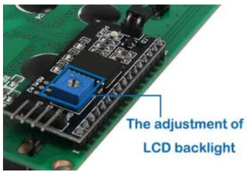
Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

If you have added the library, skip it.

Otherwise, you need to add the **LiquidCrystal_I2C** to the Arduino library file directory, otherwise the compiler does not pass. **Please refer to 'How to add library files'.**

If the LCD does not display or brightness is not enough, please adjusted the potentiometer.



Language reference

lcd.begin()
lcd.print()
lcd.setCursor()

Application effect

You will see the LCD display string, while the LCD backlight every 500ms lit once.

Lesson 10: Relay Module Experiment

Overview

This lesson will teach you how to use a button to control a relay experiment.The Delay() function is not used to eliminate jitter and improve the running efficiency of the program.

Specification

Null

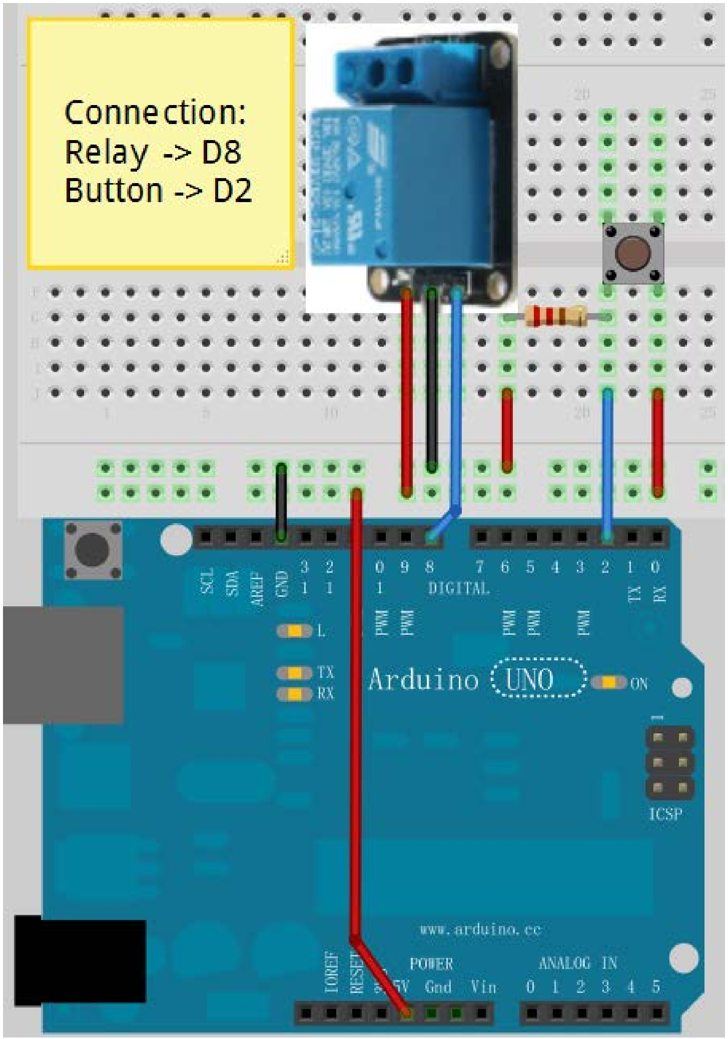
Pin definition

Relay module		UNO R3
S	->	D8
+	->	VCC
-	->	GND

Hardware required

Material diagram	Material name	Number
SONGLE 10A 250VAC 10A 125VAC 10A 30VDC 10A 28VDC SRD-05VDC-SL-C IN GND VCC	Relay module	1
	Button	1
	10KΩ resistor	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Connection:
Relay -> D8
Button -> D2

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips： click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
const
millis()

Application effect

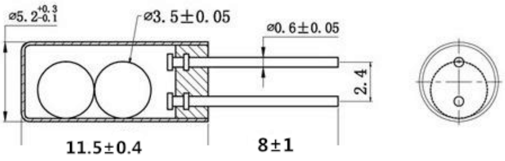
When the button is pressed, the state of the relay will be changed.

Lesson 11: Tilt Switch

Overview

This tilt switch can easily be used to detect orientation. Inside the can is a ball that make contact with the pins when the case is upright. Tilt the case over and the balls don’t touch, thus not making a connection.

Specification



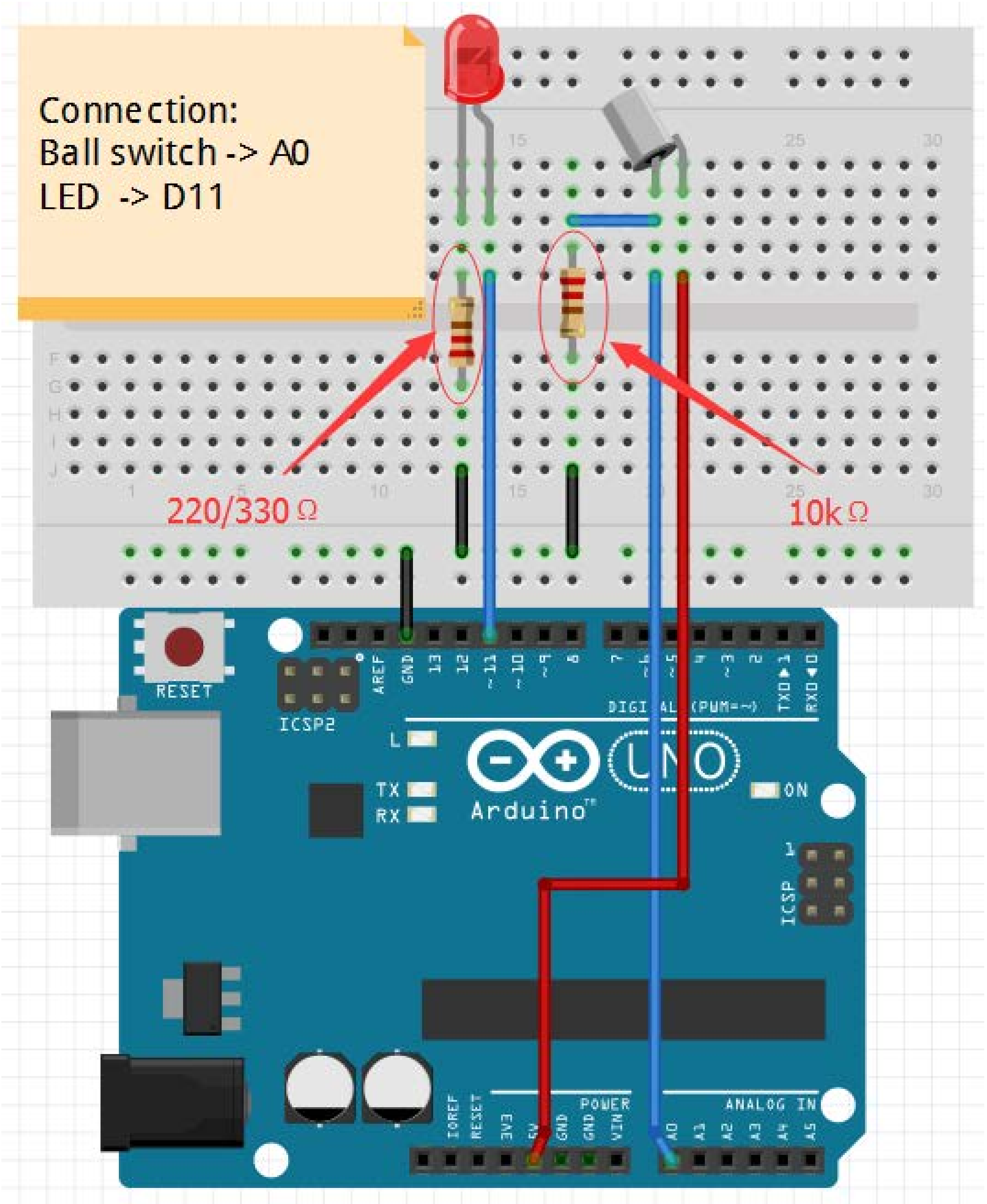
sPin definition

Nopolarity.

Hardware required

Material diagram	Material name	Number
	Ballswitch	1
	LED	1
	220/330Ω resistor	1
	10KΩ resistor	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Note：The longest LED of the pin is connected to the digital signal port 11 (D11).
Ball switch`s pin is not divided into positive and negative polarity.

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips：click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
If()
else

Application effect

LED light up when you lean or knock on ball switch.

Lesson 12: Photoresistor

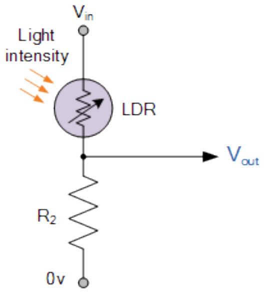
Overview

A Photoconductive light sensor does not produce electricity but simply changes its physical properties when subjected to light energy. The most common type of photoconductive device is the Photoresistor which changes its electrical resistance in response to changes in the light intensity.

Specification

Null

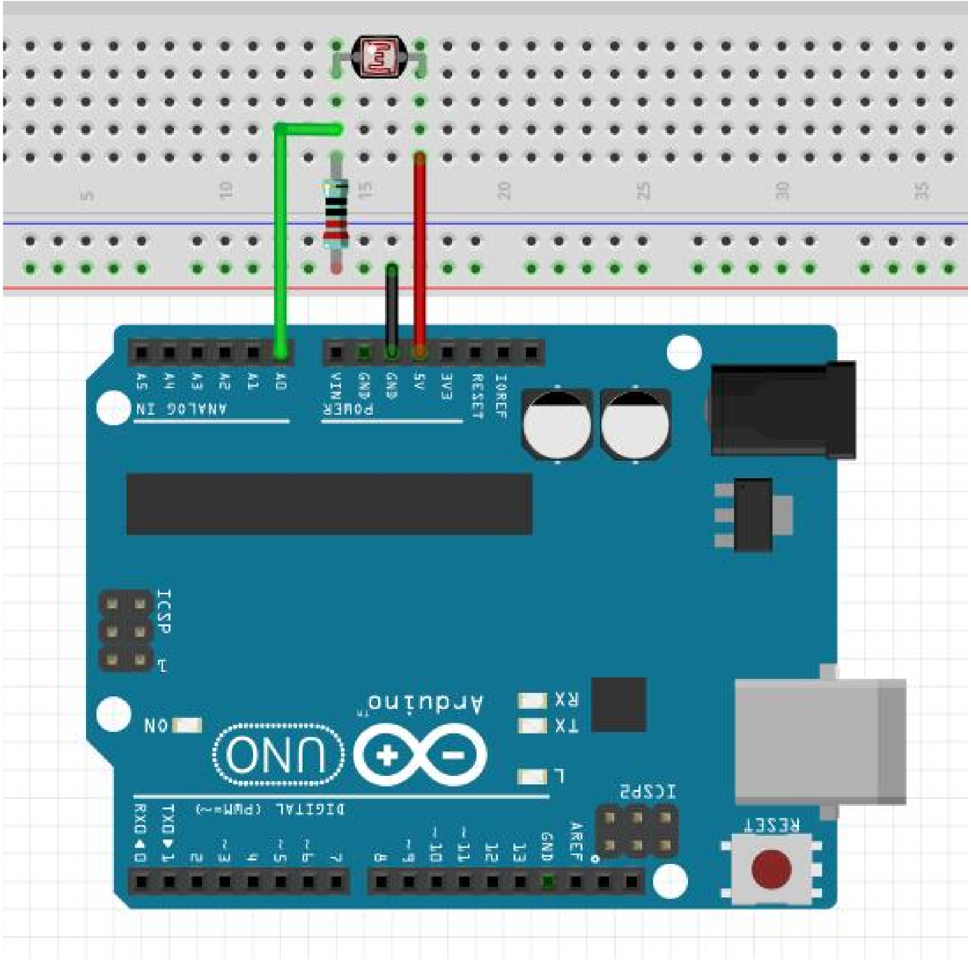
Pin definition



Hardware required

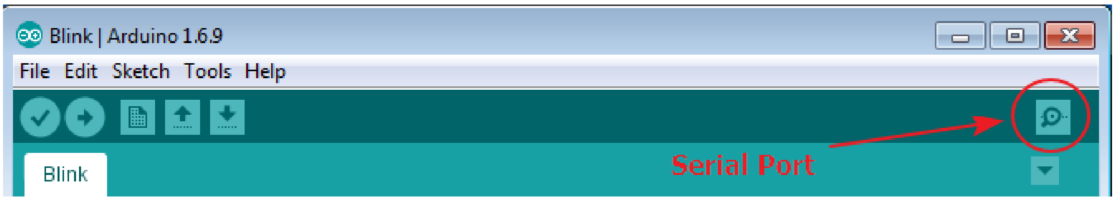
Material diagram	Material name	Number
	Photoresistor	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Compile and upload

Tips: Refer to the operation demo (Step4 to Step8). And open the serial port.



Language reference

Null.

Application effect

You can see the real-time illumination value in the monitor.

Lesson 13: Flame Alarm System

Overview

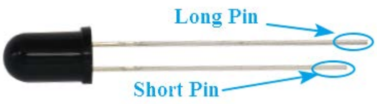
The Flame Sensor can be used to detect fire source or other light sources of the wavelength in the range of 760nm - 1100 nm. It is a high speed and high sensitive NPN silicon phototransistor.

This lesson will teach you how to make a Flame alarm system.

Specification

Null

Pin definition

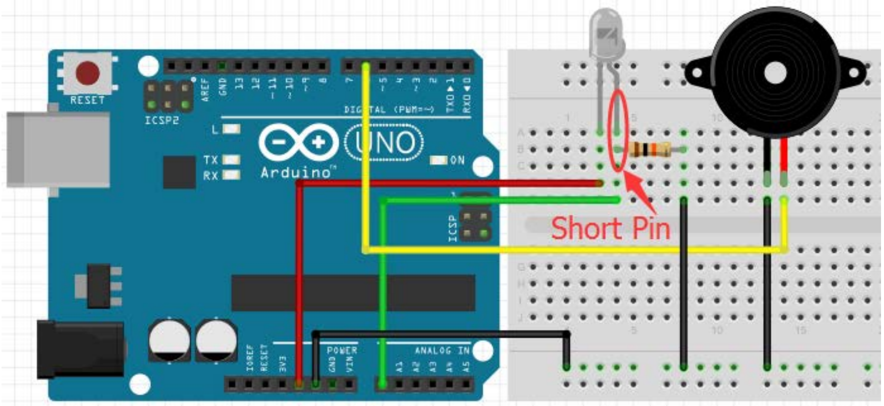


Hardware required

Material diagram	Material name	Number
	Active buzzer	1

	Flame Sensor	1
	10KΩ resistor	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Flame sensor		UNO R3
Short Pin	->	+5V
Long Pin	->	A0
PassiveBuzzer	->	D6

Note：The short pin of the Flame sensor is connected to +5v.

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Null

Application effect

We can simulate a flame environment. Turn on the lighter and then near the flame sensor, you will hear the buzzer sound.

Lesson 14: Analog Temperature

Overview

Thermistors are thermally sensitive resistors whose prime function is to exhibit a large, predictable and precise change in electrical resistance when subjected to a corresponding change in body temperature.
This lesson we will teach you how to read the value of the thermistor**.**

Specification

Model: MF52-103
Insulation Material: Ceramic
Color: Black
Rated Power: 0.05W
Resistance Value: 10k
Resistance Tolerance: H (Â±3%)
B Value: 3950K
Pin Pitch: 1.5mm / 0.059"

Pin definition

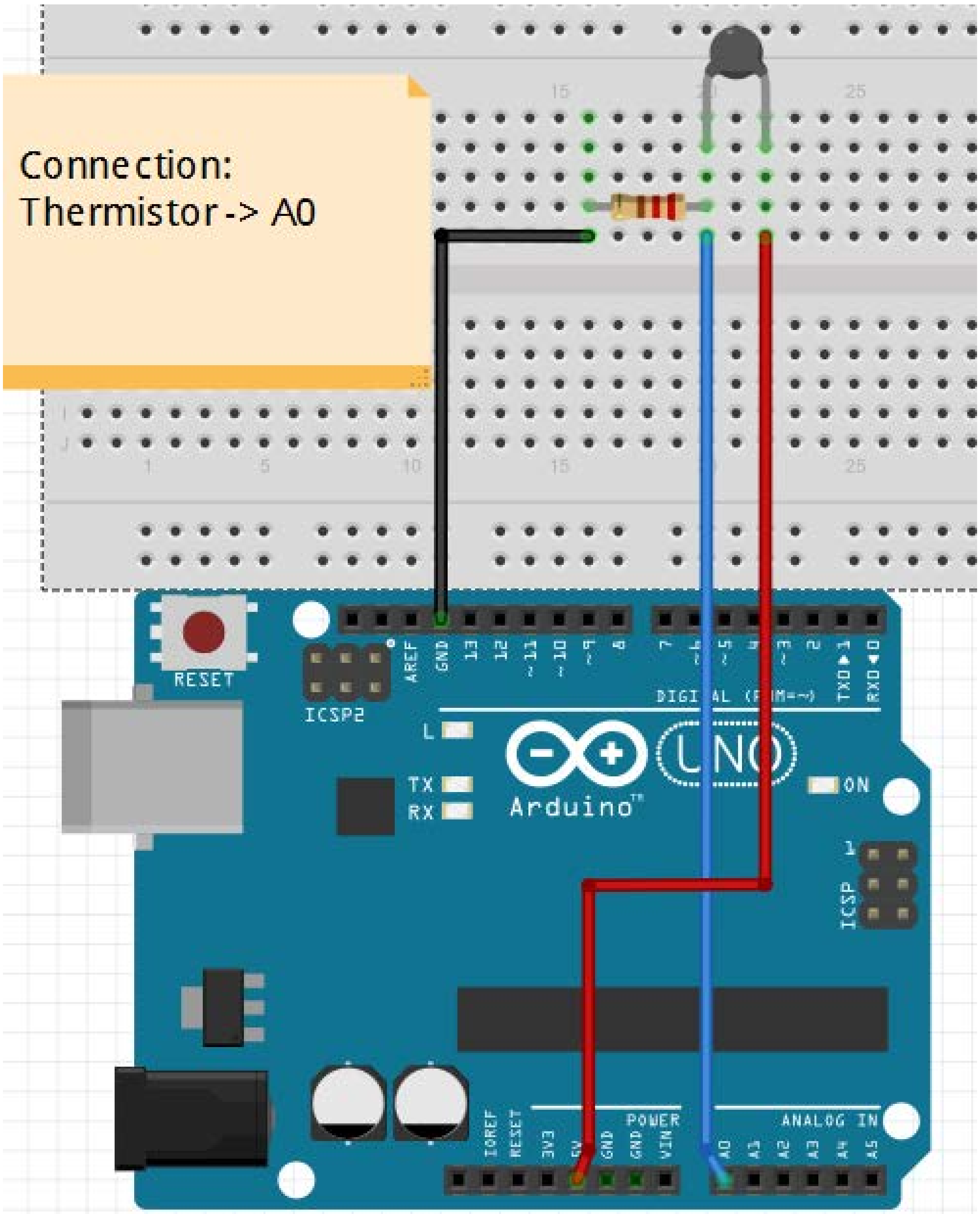
Nonpolar

Hardware required

Material diagram	Material name	Number

	Thermistor	1
	10KΩ resistor	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

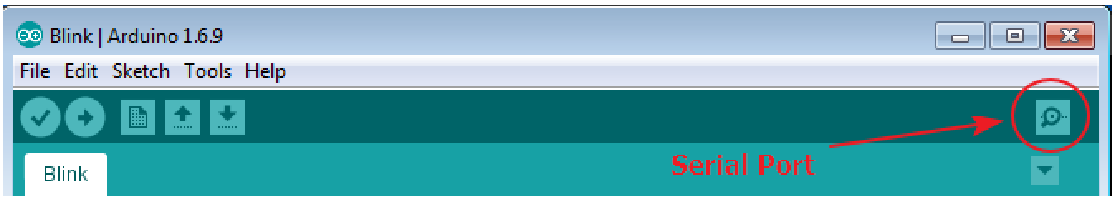
Connection diagram



Note: Thermistor `s pin does not distinguish between positive and negative poles.

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8). And open the serial port.



Language reference

serial
DEC

Application effect

After uploading the program, open the serial port monitor, you will see a series of temperature values.

Lesson 15: Soil Moisture Sensor

Overview

The soil moisture sensor can read the moisture around the soil. So it can be used to monitor your garden soil moisture and remind you to water the flowers.

Specification

Supply voltage : 3.3V or 5V
Input signal: 0~4.2V
Rated Current : 35mA
Output range and soil moisture :
0~300 : Dry
300~700 : Damp
700~950 : Enough moisture

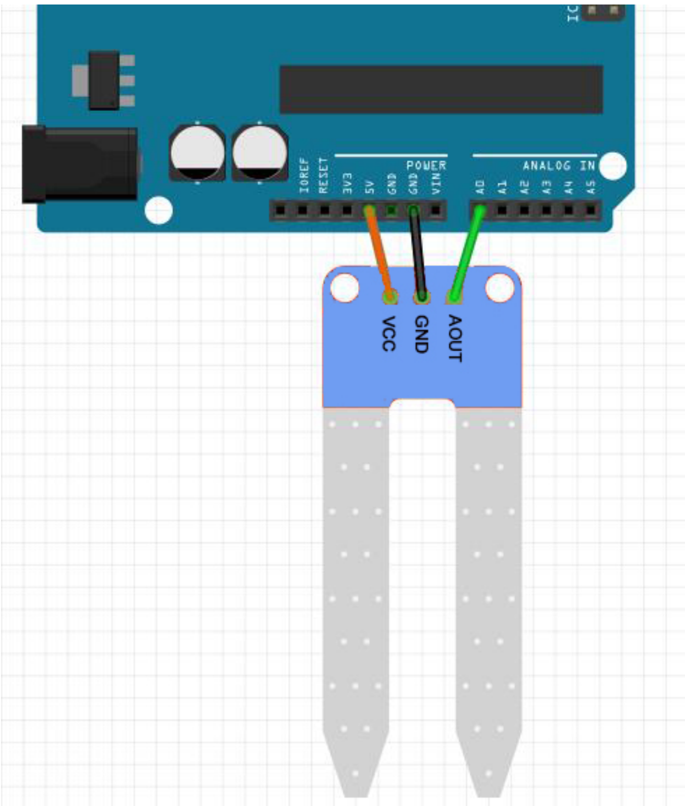
Pin definition

Null

Hardware required

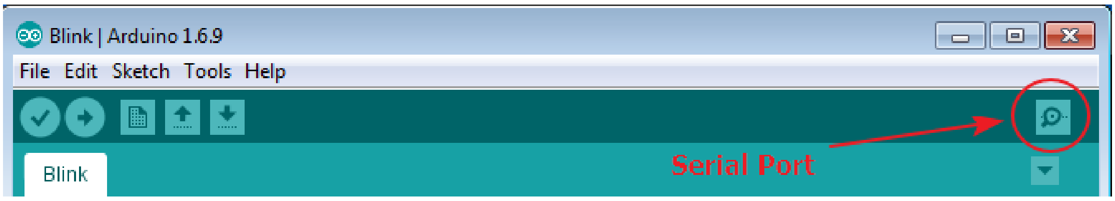
Material diagram	Material name	Number
	DHT11	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Compile and upload

Tips: Refer to the operation demo (Step4 to Step8). And open the serial port.



Language reference

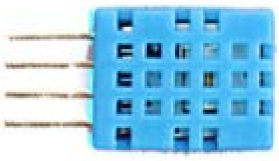
Null.

Application effect

When you insert the sensor into the soil, you can see the real-time humidity value of the soil in the monitor.

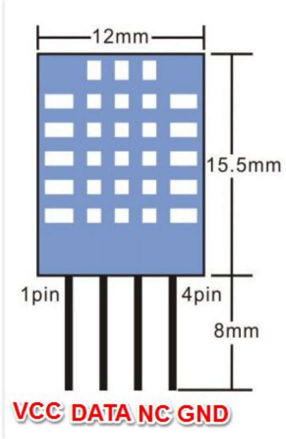
Lesson 16: DHT11 Experiment

Overview



This is an experiment on temperature and humidity, you will learn the external library files to simplify the process.

Specification

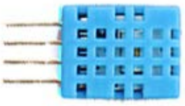


Please view DHT11-datasheet.pdf.
Path: \Datasheet\ DHT11-datasheet.pdf

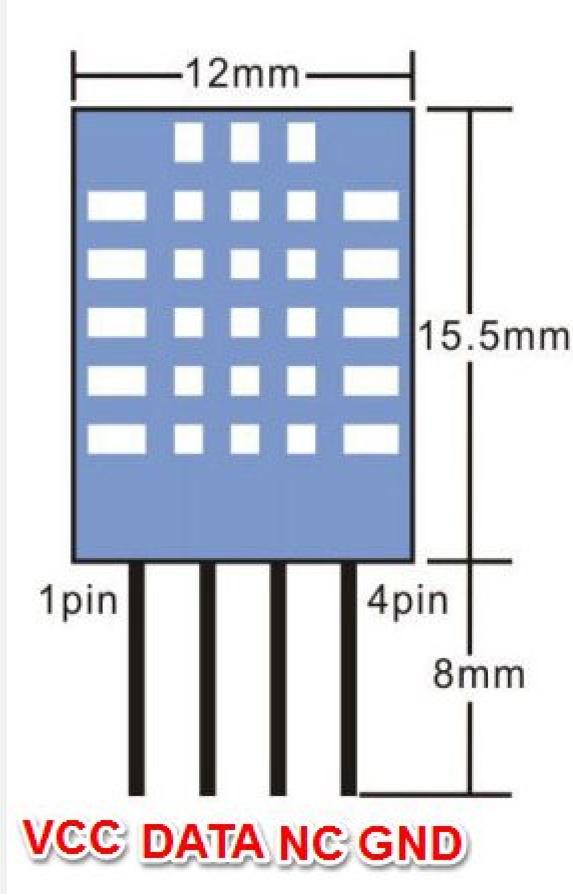
Pin definition

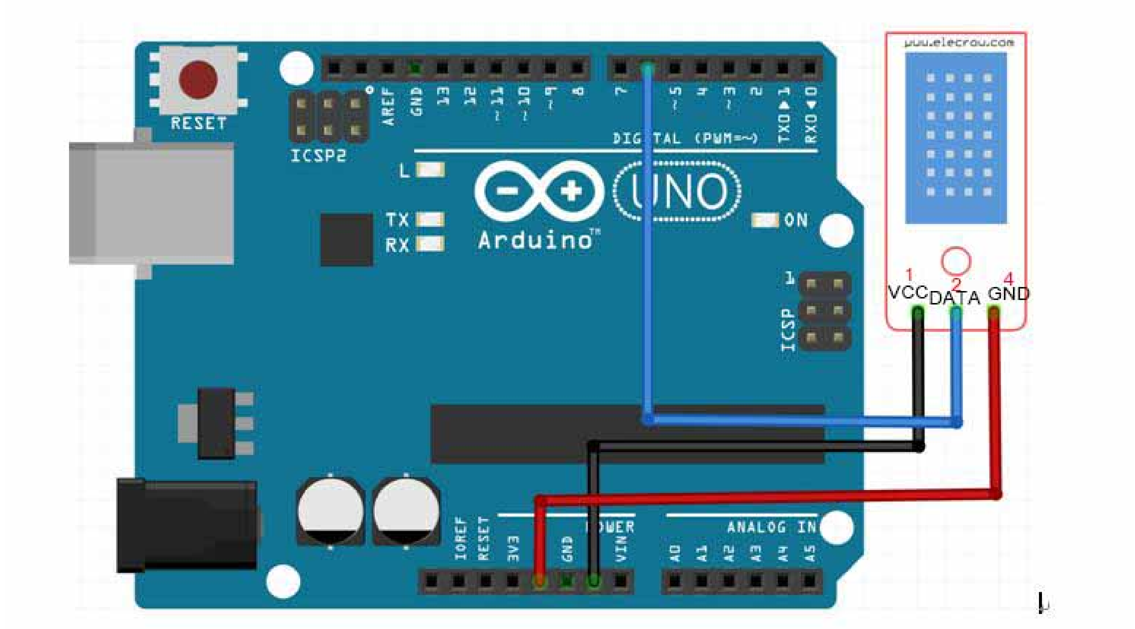
UNO R3		DHT11
GND	->	GND/'-'
D5	->	DATA/'out'
+5V	->	VCC/'+'

Hardware required

Material diagram	Material name	Number
	DHT11	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram





Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).
If you have added the library, skip it.
Otherwise, you need to add the **DHT.h"** to the Arduino library file directory, otherwise the compiler does not pass. **Please refer to 'How to add library files.docx'.**

Language reference

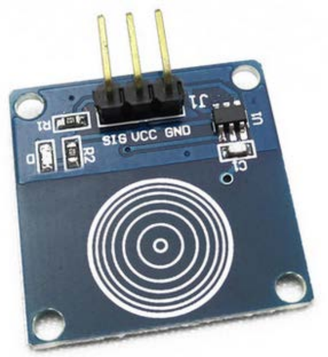
Tips：click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
serial

Application effect

Open the serial port monitor, you will see the value returned by DHT11.

Lesson 17: Touch Lamp

Overview



This is a touch sensor to control the LED lamp experiment, it can control each LED light, but also can achieve the effect of breathing light.

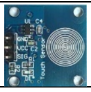
Specification

Null

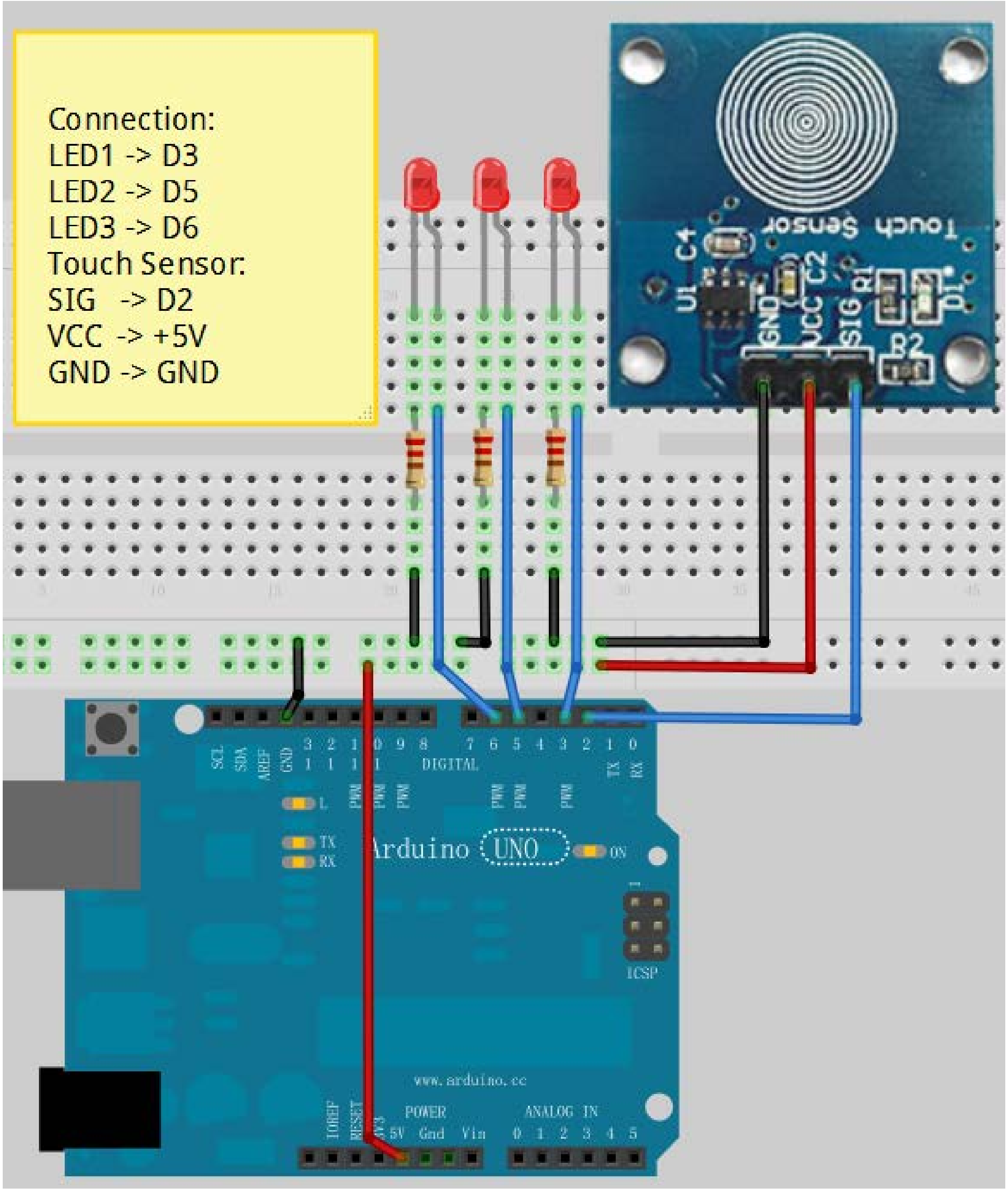
Pin definition

Touch Sensor		UNO R3
GND	->	GND
VCC	->	+5V
SIG	->	D2

Hardware required

Material diagram	Material name	Number
	Touch Sensor	1
	LED	3
	220/330Ω resistor	3
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Note：The longest LED of the pin is connected to the digital signal port.

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips： click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
attachInterrupt
switch(case)

Application effect

Through the touch panel, you can control the LED light.

Lesson 18: Ultrasonic Ranging

Overview

The HC-SR04 ultrasonic sensor uses sonar to determine distance to an object like bats or dolphins do. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package.
This is a experimental use of ultrasonic module (HCSR04) test distance. Ultrasonic module is generally used in the robot.

Specification

Please view "HCSR04.pdf"
Path: \Datasheet\HCSR04.pdf

Pin definition

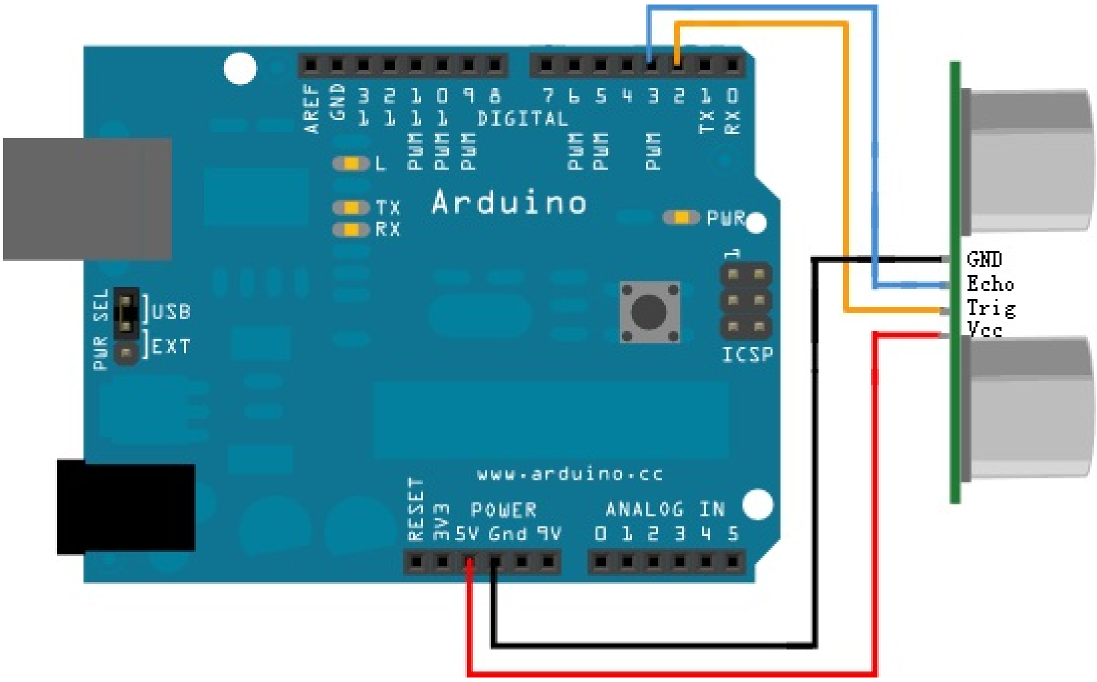
HC SR04		UNO R3
Vcc	->	VCC
Trig	->	D2
Echo	->	D3

Gnd	->	GND

Hardware required

Material diagram	Material name	Number
	HCSR04	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Connection:

HC SR04		UNO R3
Vcc	->	VCC
Trig	->	D2
Echo	->	D3
Gnd	->	GND

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Tips： click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
delayMicroseconds()

Application effect

Open the serial port monitor, and you will see the data returned by the ultrasonic module.

button will have the corresponding coding.

Lesson 19: Sweep

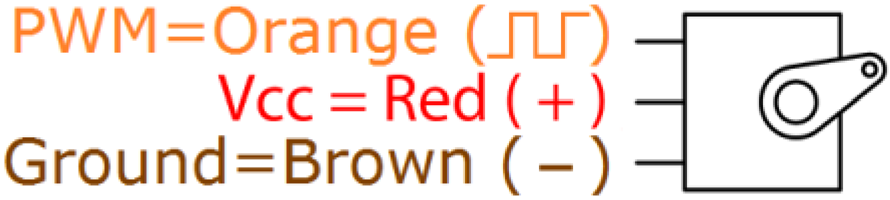
Overview

Sweeps the shaft of a RC servo motor back and forth across 180 degrees.
This example makes use of the Arduino servo library.

Specification

Please view SG90Servo-datasheet.pdf.
Path: \Datasheet\ SG90Servo-datasheet.pdf

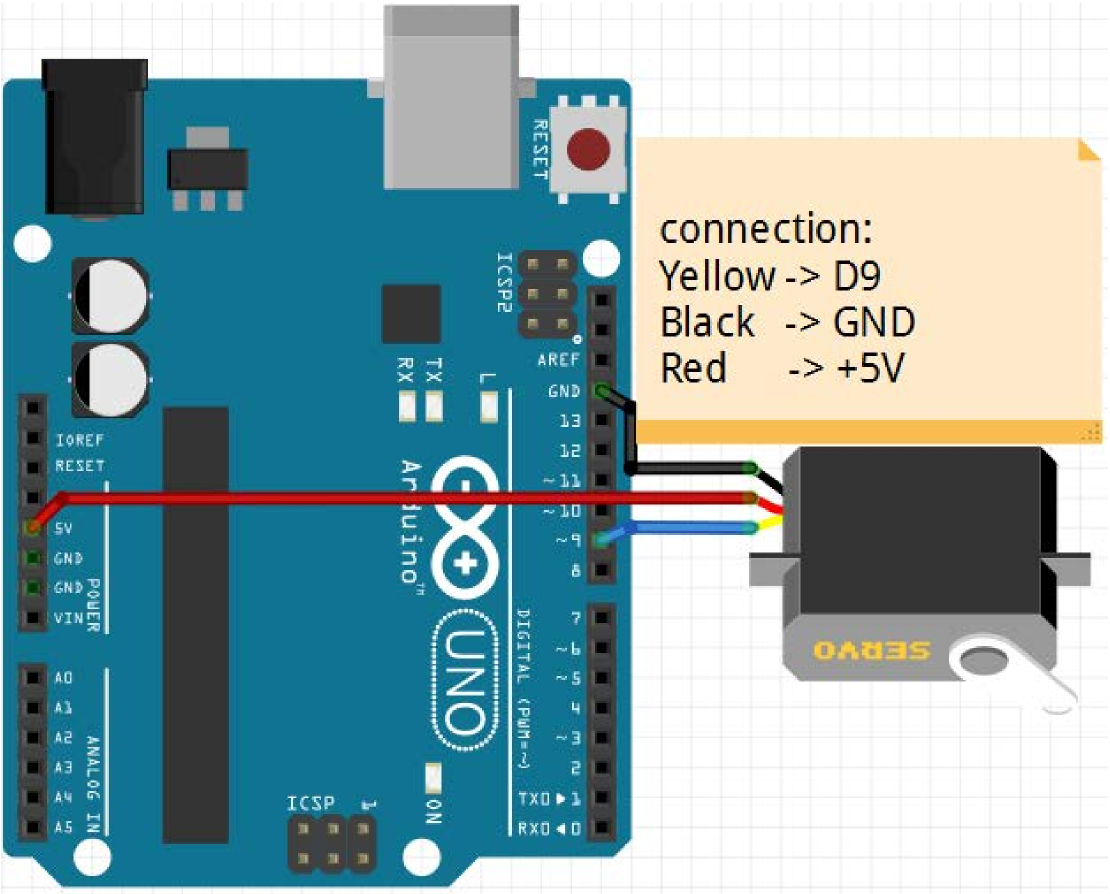
Pin definition



Hardware required

Material diagram	Material name	Number
	9g Servo	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Null

Application effect

You will see the servo motor turning 180 degrees back and forth.

Lesson 20: DC motor

Overview

In this lesson, we will learn how to control the direction and speed of a small-sized DC motor. The basic problem in using Arduino's digital pins to control the motor directly is that it is very difficult to reverse the voltage. So we need to use a L293D chip.

This 5V DC motor give the two terminals of the copper sheet one high and one low level, and the motor will rotate. Before proceeding further it is recommended that you read and understand the pin diagram of L293D.

Now, lets control an DC Motor with a L293D chip and make it so it will go clockwise or counter clockwise.

Specification

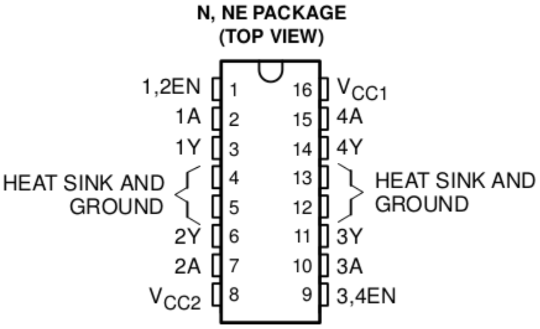
Path: \Datasheet\ L293D-Datasheet

Pin definition

DC motor:

Null

L293D:

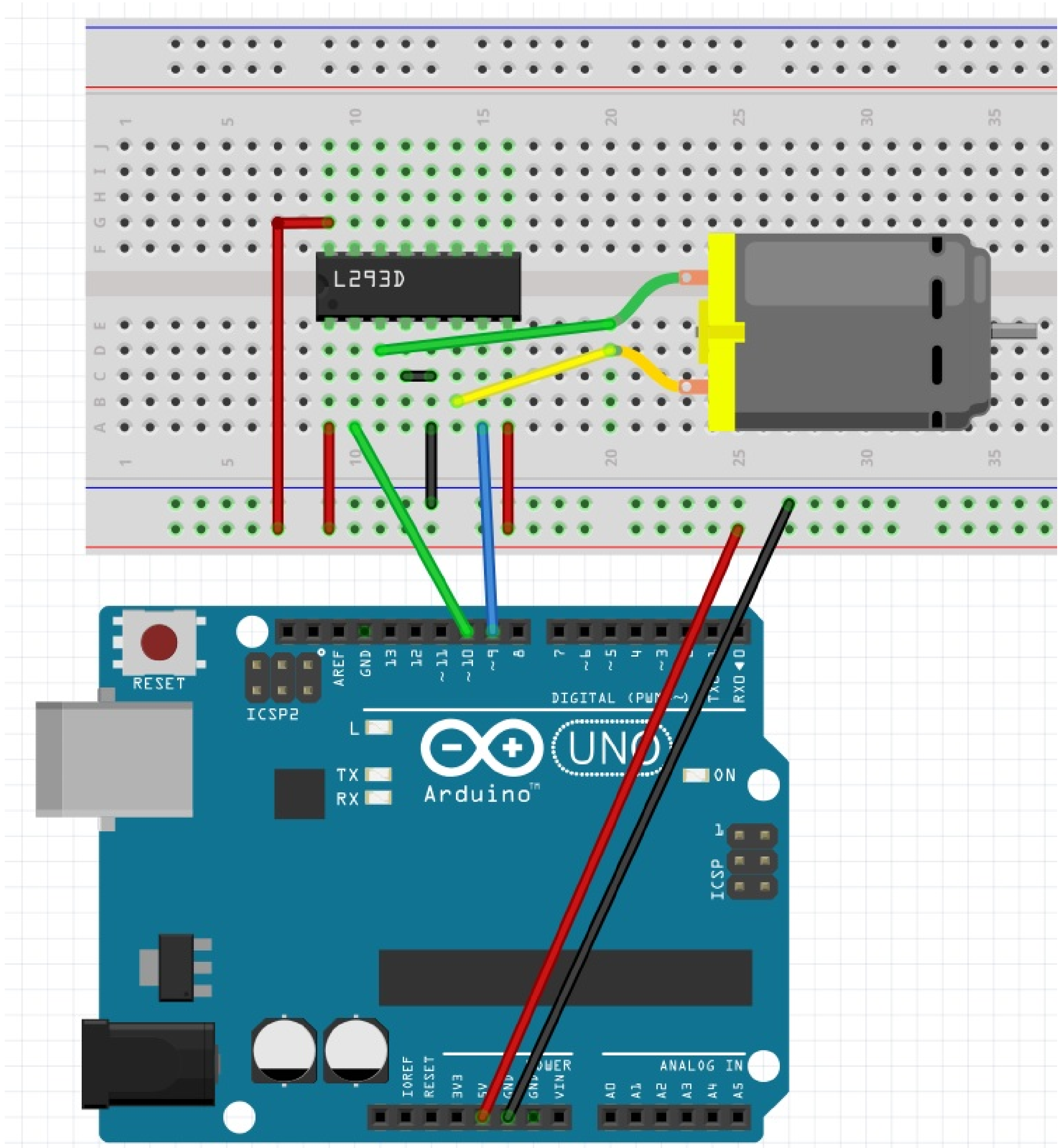


Hardware required

Material diagram	Material name	Number
	Small-sized DC motor	1
	L293D	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Pin EN is an enabling pin and works with High level. A stands for input and Y for output. when pin EN is High level, if A is High, Y outputs High level; if A is Low, Y outputs Low level. When pin EN is Low level, the L293D does not work.Because we just needs to drive one motor in this lesson, so use one side of the L293D.

Connection diagram



Connection

L293D		Uno R3
Pin1	->	5V
Pin2	->	D10
Pin4	->	GND
Pin5	->	GND
Pin7	->	D9
Pin8	->	5V
Pin16	->	5V

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Null

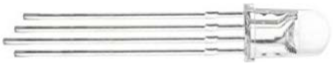
Application effect

You can see the DC motor will begin rotating left and right.

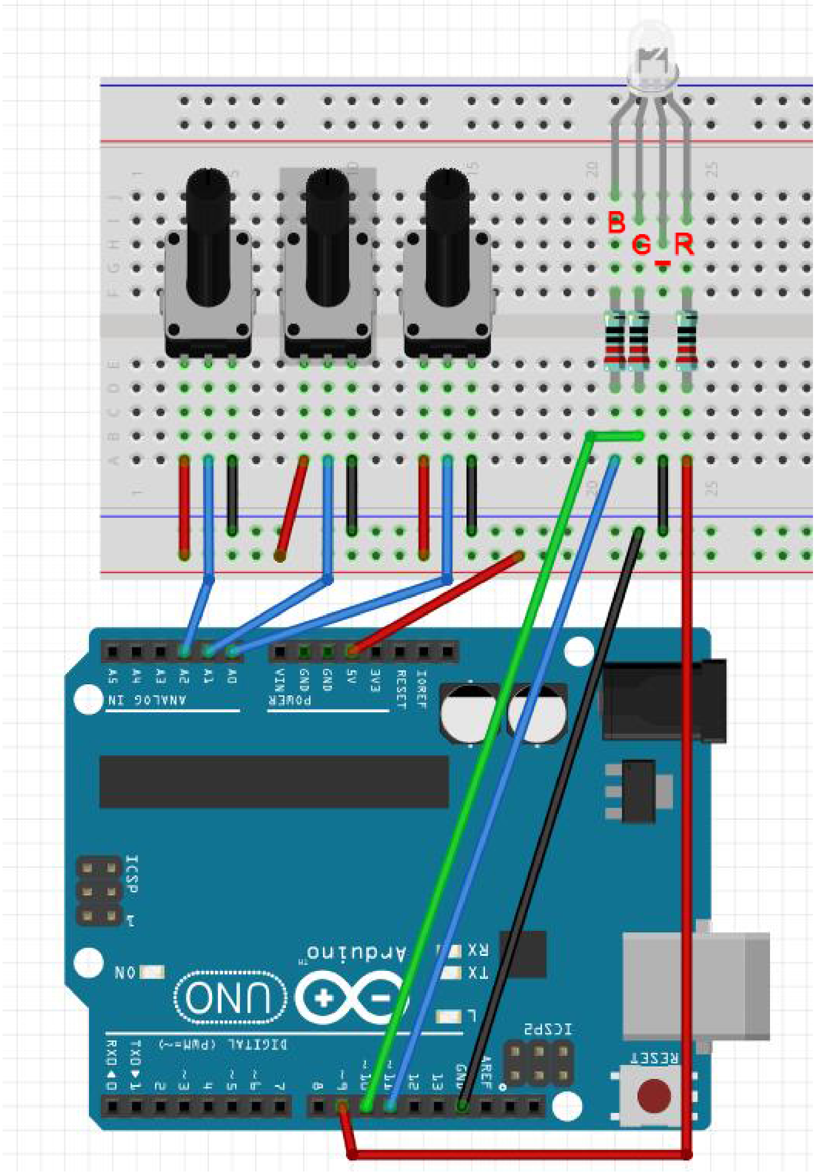
Lesson 21: Fun Experiment--Color Dimmer

We have learned the RGB LED, it can achieve discoloration. In this lesson, let's do a mood light and make it switch with your heart.

Hardware required

Material diagram	Material name	Number
	RGB LED	1
	10KΩpotentiometer	3
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Connection

RGB LED		UNO
R	->	9
G	->	10
B	->	11

Compile and upload

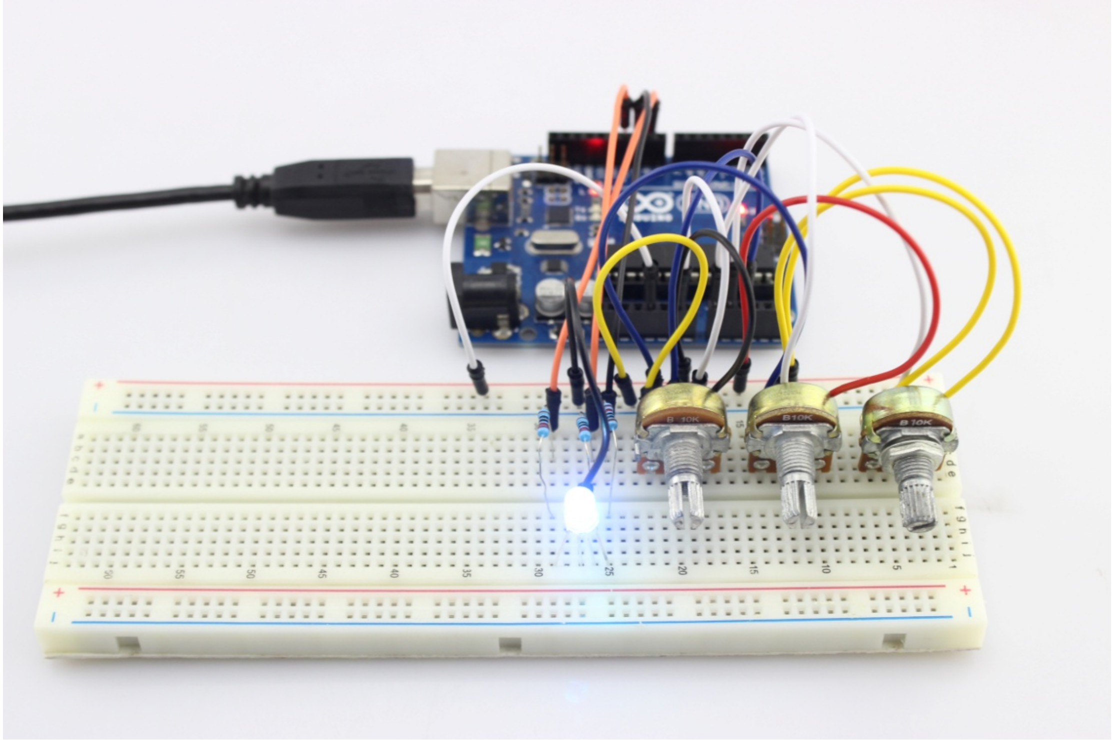
Tips: Refer to the operation demo (Step4 to Step8).

Language reference

map()

Application effect

You can rotate the potentiometer to switch the color of the RGB LED, just like color dimmer is cool.

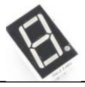


Lesson 22: Fun Experiment--Traffic Light

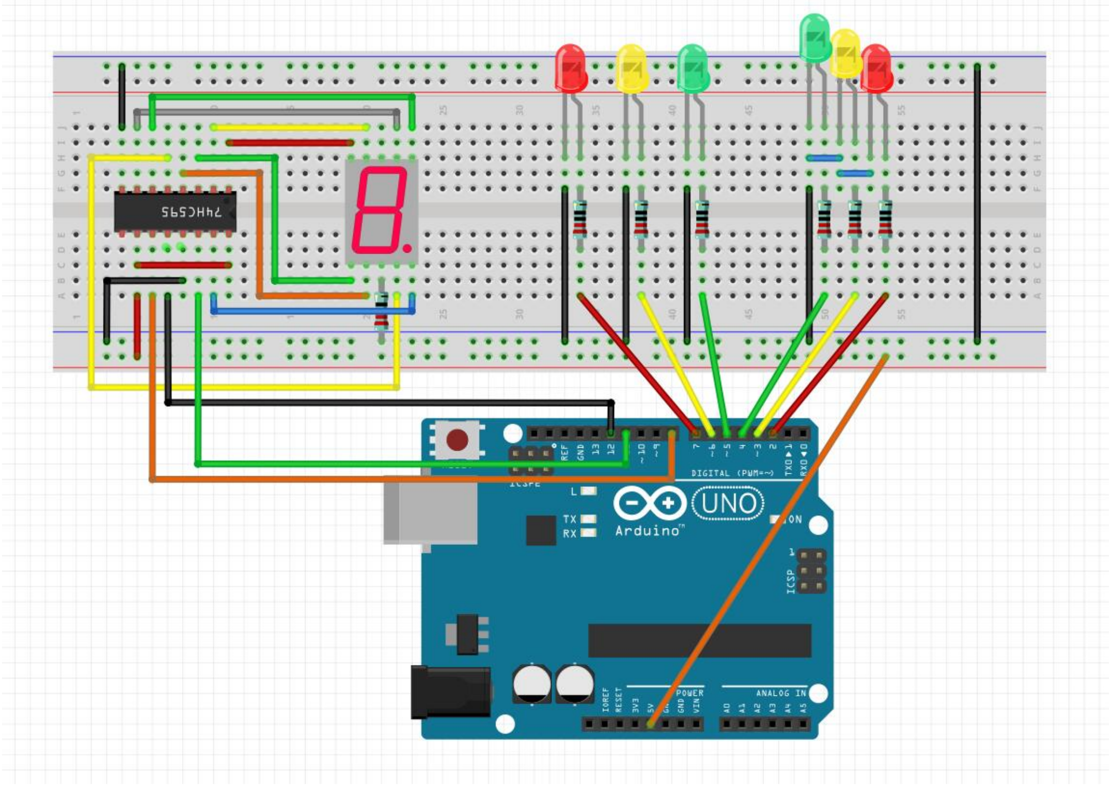
In this lesson, we will use the Arduino to imitate the traffic lights. The lights turn from green to yellow, then turn red and then start all over lights. The time interval follows the traffic signal standard interval. This item can be used in traffic lights simulation or toys.

Hardware required

Material diagram	Material name	Number
	LED(red、yellow、green)	3f
	1 digit LED Segment	1

	Displays(common cathode)	
	74HC595	1
	220/330Ω resistor	7
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Compile and upload

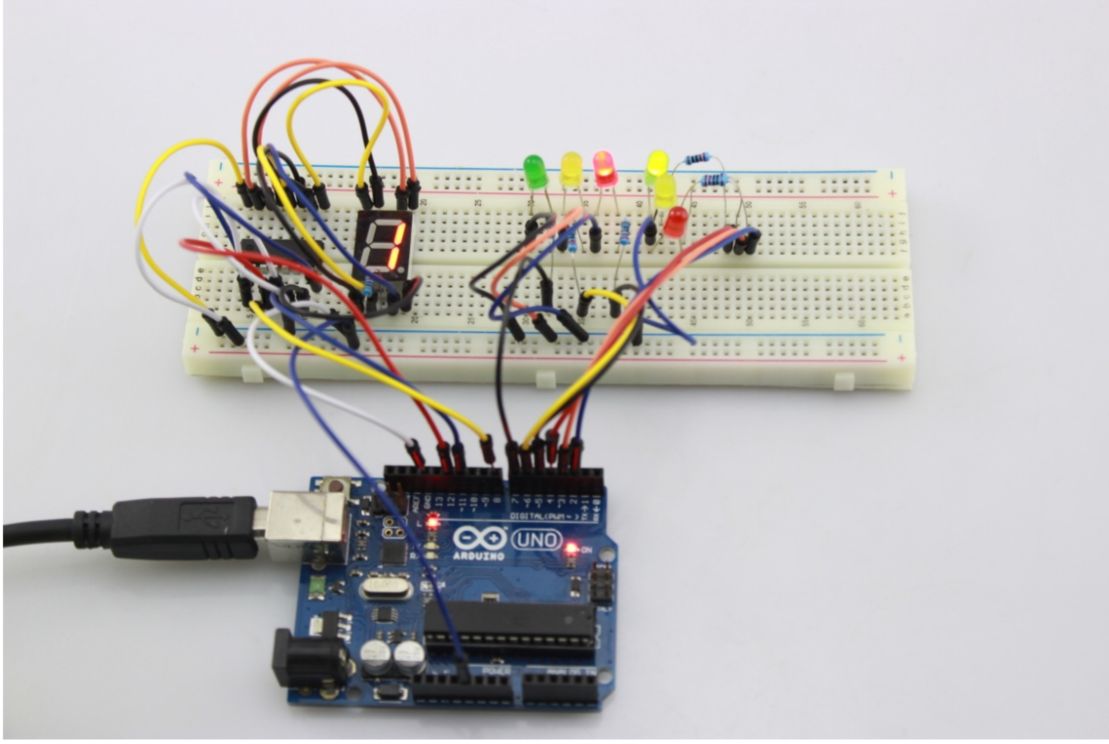
Tips: Refer to the operation demo (Step4 to Step8).

Language reference

shiftOut();

Application effect

First, the LED display counts down from 9s, and the red light in the NS and the green one in the EW light up. Then it counts down from 3, and the green LED in the EW goes out when the yellow lights up, with the NS red light still on. After 3s, the 7-segment counts down from 9s again. At the same time, the red light in the EW and the green in the NS light up. 9s later, it counts down from 3s, when the yellow light in the NS lights up and the red in the EW keeps on.



Lesson 23: Fun Experiment--Intelligent Fire

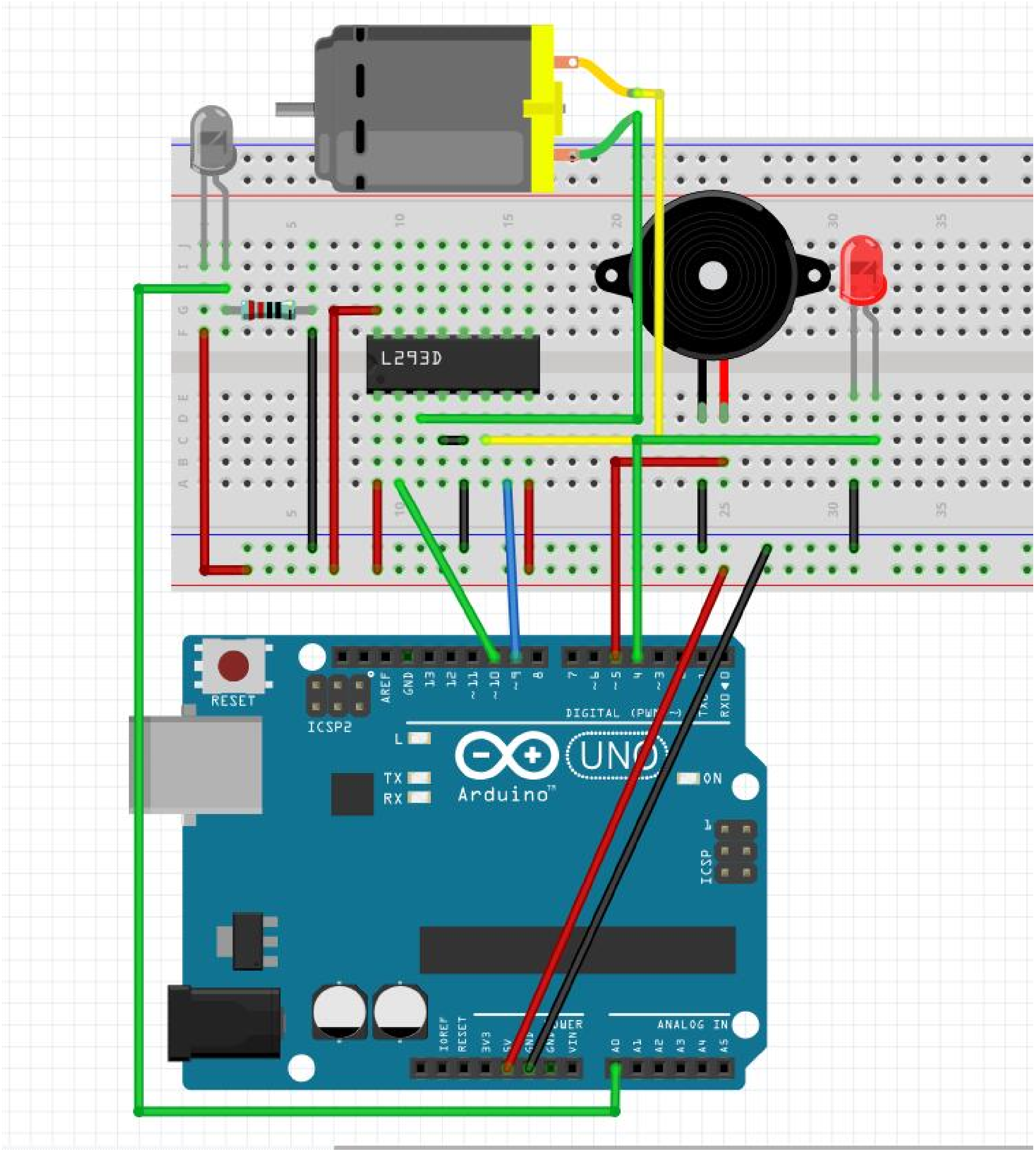
In this lesson, we will simulate a smart fire extinguishing device.

Hardware required

Material diagram	Material name	Number
	LED	1

	Active buzzer	1
	L293D	1
	Flame Sensor	1
	Small-sized DC motor	1
	220/330Ω resistor	Several
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Compile and upload

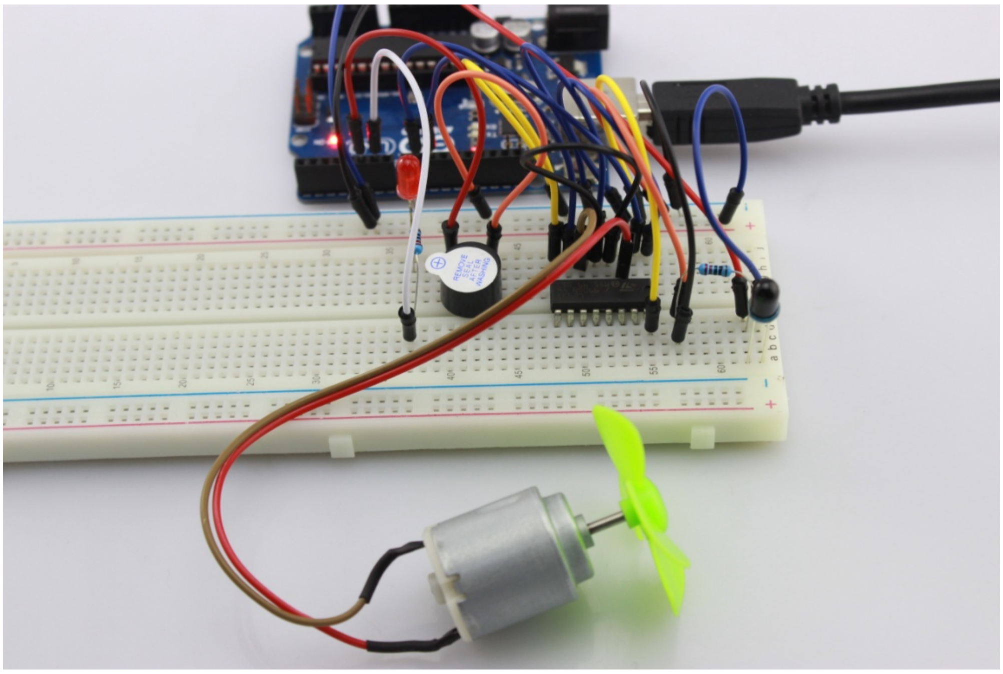
Tips: Refer to the operation demo (Step4 to Step8).

Language reference

Null.

Application effect

Once the sensor test fire, the buzzer will alarm and the red LED will light flashes, then open the fan to put out a fire.



Lesson 24: Fun Experiment--Theremin

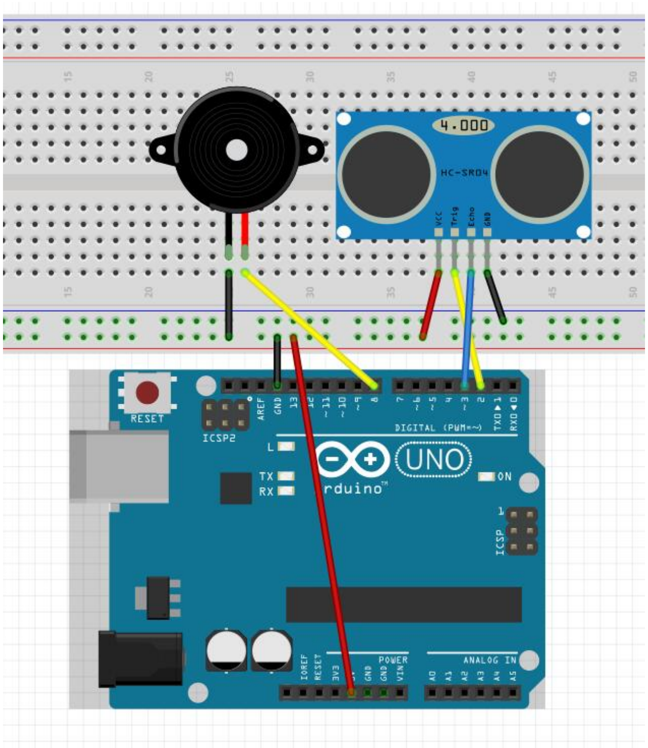
The Theremin is a kind of electronic musical instrument which don't need to body contact. It's so cool and interesting. In this lesson, let's do a simple Theremin.

Hardware required

Material diagram	Material name	Number
	HCSR04	1
	Passive buzzer	1
	220/330Ω resistor	Several
	USB Cable	1

	UNO R3	1
	Breadboard	1
	Jumper wires	Several

Connection diagram



Connection

HCSR04		arduino
TrigPin	->	D2
EchoPin	->	D3
VCC	->	5V
GND	->	GND

Buzzer		arduino
Pin1	->	D8
Pin2	->	GND

Compile and upload

Tips: Refer to the operation demo (Step4 to Step8).

Language reference

DelayMicroseconds();

PulseIn();
map();
tone();

Application effect

Now, use your hand to get close to the ultrasonic sensor slowly, it will making a changing tone of voice. You can even try to play simple melodies.