



37 SENSOR KIT TUTORIAL

FOR UNO AND MEGA V2.0

Preface

Our Company

Established in 2011, Elegoo Inc. is a professional manufacturer and exporter that is involved in the design, development, production and marketing of Arduino accessories, 3D printers, Raspberry Pi accessories and STM32. We are located in Shenzhen which is known as the Chinese Silicon Valley. All of our products comply with international quality standards and are greatly appreciated in a variety of different markets throughout the world.

Official website: <http://www.elegoo.com>

US Amazon store front: <http://www.amazon.com/shops/A2WWHQ25ENKVJ1>

CA Amazon store front: <http://www.amazon.ca/shops/A2WWHQ25ENKVJ1>

DE Amazon Store front: <http://www.amazon.de/shops/A1780XYQ9DFQM6>

Vitrine à l'Amazon FR: <http://www.amazon.fr/shops/ A1780XYQ9DFQM6>

Tienda de Amazon de ES.: <http://www.amazon.es/shops/A1780XYQ9DFQM6>

IT Amazon negozio: <http://www.amazon.it/shops/ A1780XYQ9DFQM6>

UK Amazon storefront: <http://www.amazon.co.uk/shops/A1780XYQ9DFQM6>

Our Tutorial

This tutorial is geared for beginners. You will learn all the basic information about how to use Arduino controller board, sensors and components. If you want to study Arduino in more depth, we recommend that you read the Arduino Cookbook written by Michael Margolis.

Our after-sales

If you have any question, comment or suggestion about our company, product or tutorial, please don't hesitate to send an email to service@elegoo.com (US and CA) or EUservice@elegoo.com (Europe). We are committed to continuous improvement so your feedback is invaluable.

Packing list

www.elegoo.com



JOYSTICK



RGB
LED



IR
RECEIVER



AVOIDANCE



18B20
TEMP



FLAME



TRACKING



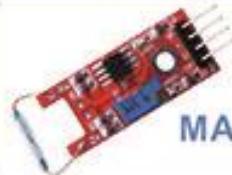
PHOTORESISTOR



IR
EMISSION

Contact us : service@elegoo.com

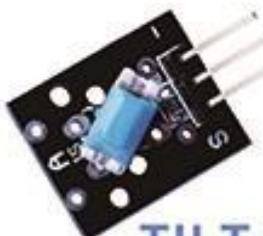
www.elegoo.com



MAGNETIC
SPRING



MENTAL
TOUCH



TIILT-SWITCH



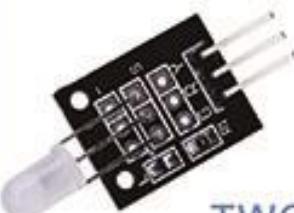
SMD
RGB



ROTARY
ENCODER



TEMP
AND
HUMIDITY



TWO-COLOR

Contact us : service@elegoo.com

www.elegoo.com



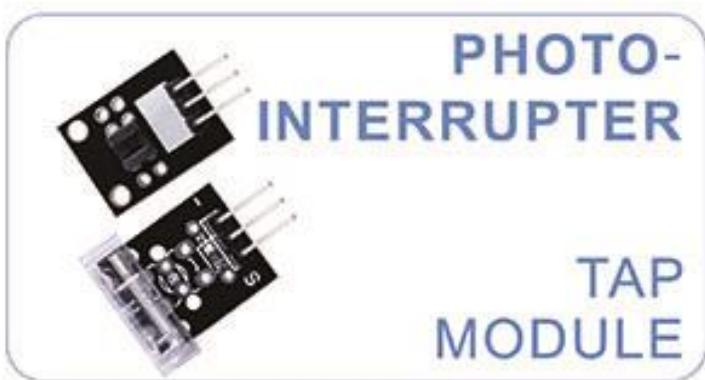
ULTRASONIC
SENSOR



DS-3231 RTC
MODULE



GY-521
MODULE



7
COLOR
FLASH



BUTTON



ACTIVE
BUZZER



LASER
EMIT



MEMBRANE
SWTCH



BIG
SOUND



WATER LEVER
SENSOR



DIGITAL
TEMPERATURE

Contact us : service@elegoo.com

www.elegoo.com



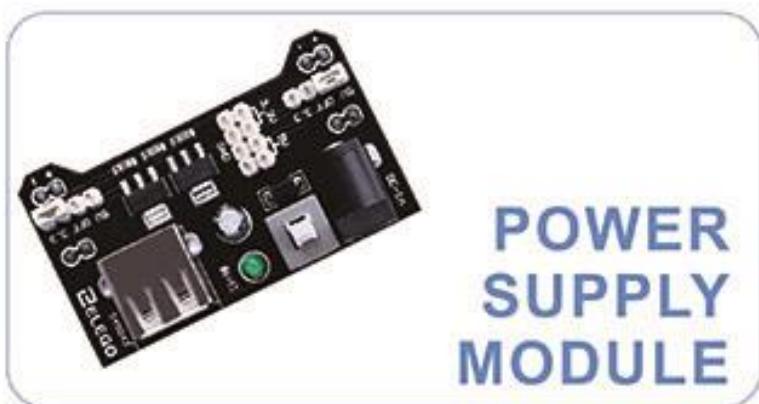
LCD 1602
MODULE
(WITH PIN HEADER)



RELAY



LINEAR
HALL



POWER
SUPPLY
MODULE



SMALL SOUND



SHOCK



PASSIVE
BUZZER



HC-SR501 PIR
MOTION SENSOR

Contact us : service@elegoo.com

Lesson Listing

Lesson 0 Installing IDE	8
Lesson 1 Add Libraries	16
Lesson 2 TEMP AND HUMIDITYMODULE.....	20
Lesson 3 DS18B20 DIGITAL TEMPERATURE SENSOR MODULE	26
Lesson 4 BUTTON SWITCH MODULE.....	32
Lesson 5 THREE TYPE OF SWITCH MODULE.....	36
Lesson 6 IR RECEIVER AND EMISSION.....	43
Lesson 7 ACTIVE AND PASSIVE BUZZER	51
Lesson 8 LASER MODULE.....	58
Lesson 9 SMD RGB MODULE AND RGB MODULE.....	62
Lesson 10 PHOTO-INTERRUPTER MODULE.....	69
Lesson 11 DUAL-COLOR COMMON-CATHODE LED	74
Lesson 12 PHOTORESISTOR MODULE	79
Lesson 13 BIG SOUND SENSOR MODULE AND SMALL SOUND SENSOR MODULE	85
Lesson 14 REED SWITCH MODULE	91
Lesson 15 DIGITAL TEMPERATURE MODULE.....	97
Lesson 16 LINEAR HALL SENSOR MODULE.....	102
Lesson 17 FLAME SENSOR MODULE	107
Lesson 18 MENTAL TOUCH MODULE	112
Lesson 19 7 COLOR FLASH LED MODULE	117
Lesson 20 JOYSTICK MODULE.....	121
Lesson 21 TRACKING MODULE.....	126
Lesson 22 Infrared 38KHz obstacle avoidance module	130
Lesson 23 ROTARY ENCODERS MODULE	136
Lesson 24 1 CHANNEL RELAY MODULE.....	142
Lesson 25 LCD Display	147
Lesson 26 Ultrasonic Sensor Module	153
Lesson 27 MPU6050 Module.....	157
Lesson 28 HC-SR501 PIR Sensor	161
Lesson 29 Water Level Detection Sensor Module.....	168
Lesson 30 Real Time Clock Module	173
Lesson 31 Keypad Module.....	177

Lesson 0 Installing IDE

Introduction

In this lesson, you will learn how to setup your computer to use Arduino and how to set about the lessons that follow.

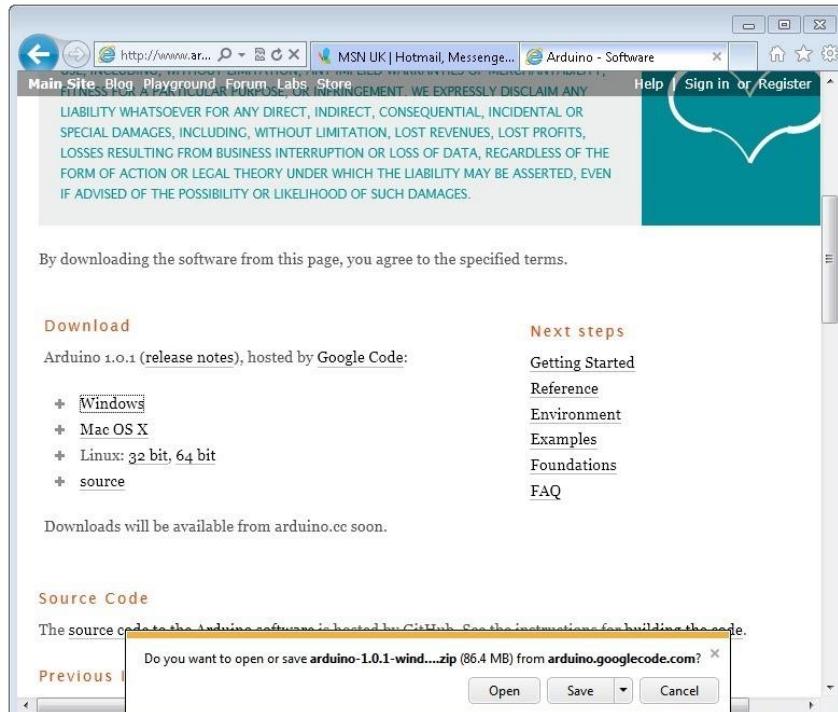
The Arduino software that you will use to program your Arduino is available for Windows, Mac and Linux. The installation process is different for all three platforms and unfortunately there is a certain amount of manual work to install the software. There is no installer program, but rather you have to unzip a folder which gives you an Arduino folder that contains the Arduino program and a few other items.

In a separate step, you must then install USB drivers,

Installing Arduino (Windows)

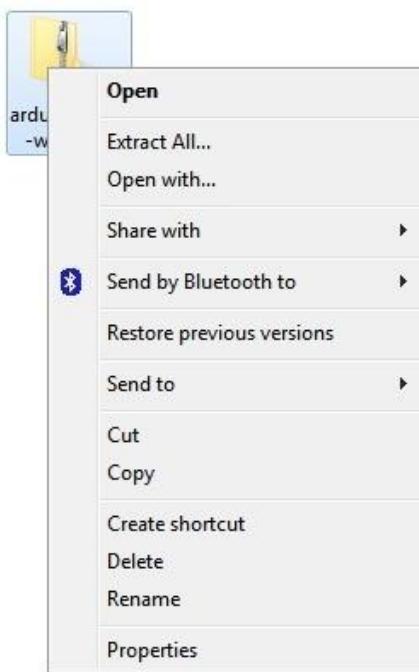
Get started by visiting the Arduino.cc website. As of April 2014, we suggest using v1.05 as 1.5 is still in beta. If 1.5 is no longer in beta by the time you read this, you can try it out!

Start by downloading the zip file for Windows. There is only one version of the software, whether you are using Windows XP through to Windows 7.



When the zip file has downloaded, extract the contents onto the Desktop, by right-clicking

on the file and selecting 'Extract All...' from the pop-up menu.



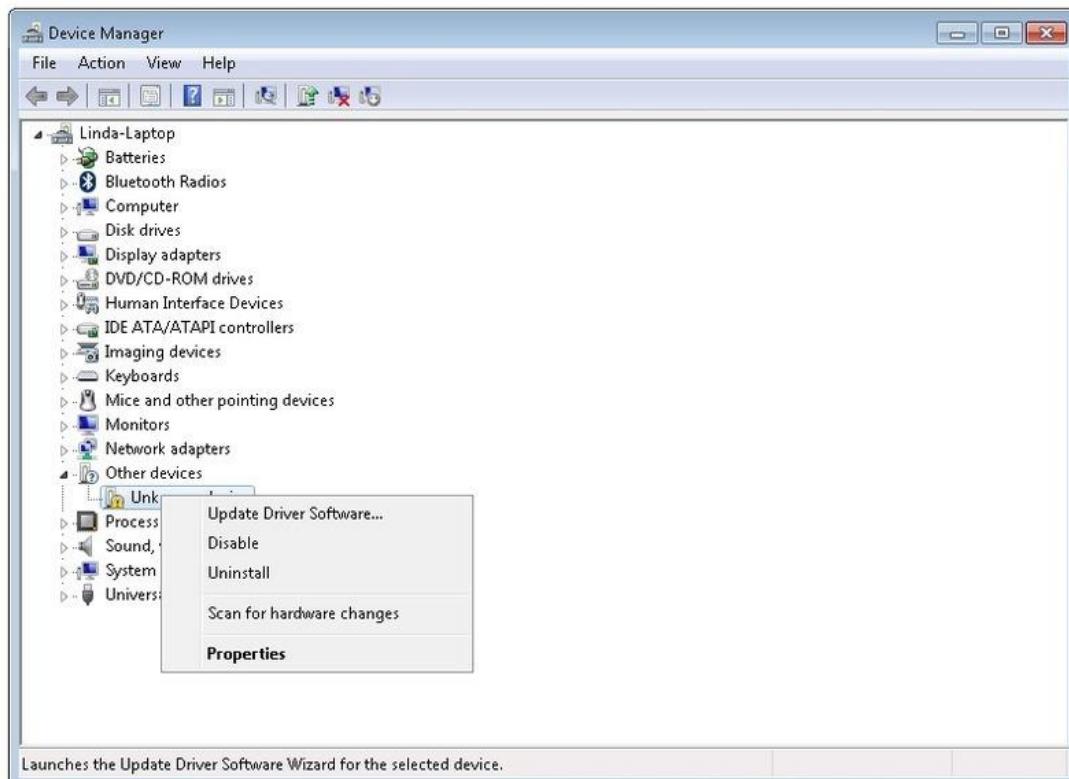
Next, select your Desktop and click 'Extract'. You can move it somewhere else onto your computer later, just by moving the folder, but for now, just keep it on the Desktop.

The Arduino folder contains both the Arduino program itself and the drivers that allow the Arduino to be connected to your computer by a USB cable. Before we launch the Arduino software, you are going to install the USB drivers.

Plug one end of your USB cable into the Arduino and the other into a USB socket on your computer. The power light on the LED will light up and you may get a 'Found New Hardware' message from Windows. Ignore this message and cancel any attempts that Windows makes to try and install drivers automatically for you.

The most reliable method of installing the USB drivers is to use the Device Manager. This is accessed in different ways depending on your version of Windows. In Windows 7, you first have to open the Control Panel, then select the option to view Icons, and you should find the Device Manager in the list.

Under 'Other Devices', you should see an icon for 'unknown device' with a little yellow warning triangle next to it. This is your Arduino.



Right-click on the device and select the top menu option (Update Driver Software...). You will then be prompted to either 'Search Automatically for updated driver software' or 'Browse my computer for driver software'. Select the option to browse and navigate to the arduino-1.0.2-windows\arduino1.0.2\drivers.



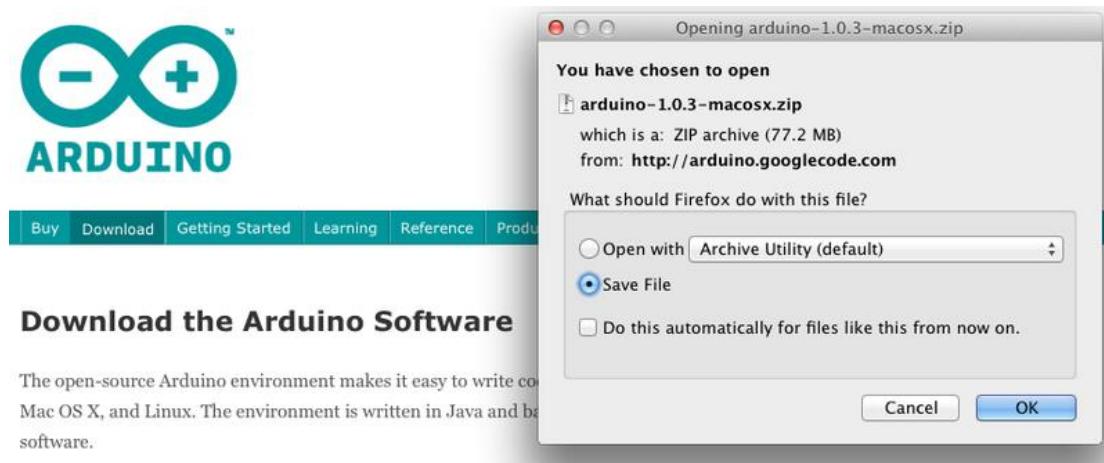
Click 'Next' and you may get a security warning, if so, allow the software to be installed. Once the software has been installed, you will get a confirmation message.



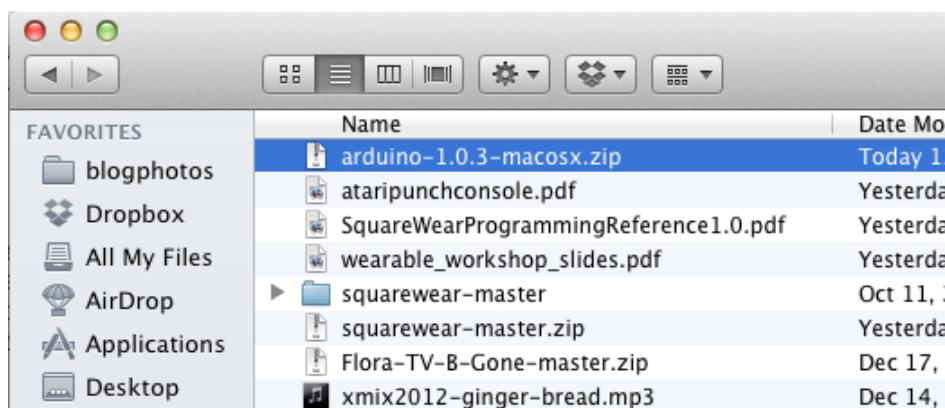
That's it, you are now ready for action, so Skip the next section on installation on Mac and Linux and move straight on to 'Boards and Ports'.

Installing Arduino (Mac and Linux)

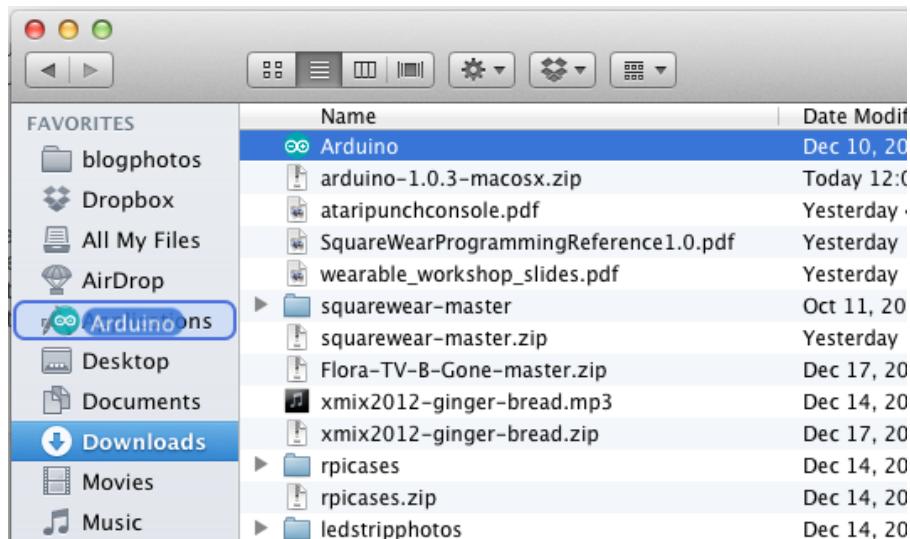
Get started by visiting the Arduino.cc website and downloading the matching IDE for your operating system. As of April 2014, we suggest using v1.0.5 as 1.5 is still in beta. If 1.5 is no longer in beta by the time you read this, you can try it out!

The image shows two side-by-side screenshots. On the left is the Arduino website homepage with the Arduino logo and navigation links for Buy, Download, Getting Started, Learning, Reference, and Products. A prominent "Download the Arduino Software" button is visible. On the right is a Mac OS X file download dialog box for "arduino-1.0.3-macosx.zip". The dialog shows the file path, size (77.2 MB), and source (http://arduino.googlecode.com). It asks "What should Firefox do with this file?", with "Save File" selected. There are checkboxes for "Open with Archive Utility (default)" and "Do this automatically for files like this from now on.". At the bottom are "Cancel" and "OK" buttons.

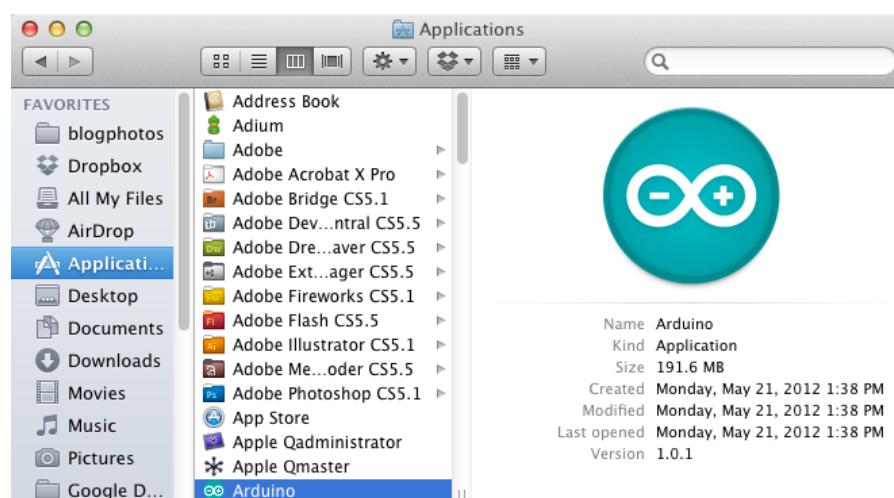
Save the install software to your desktop or wherever

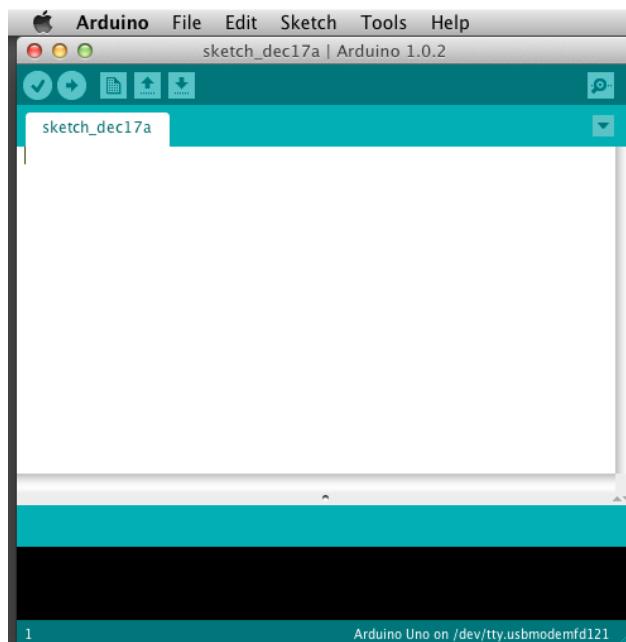


The process for installing the Arduino software on the Mac is a lot easier than on the PC. As before, the first step is to download the file. In the case of the Mac, it is a zip file.



Once downloaded, double-click on the zip file, which will extract a single file called 'Arduino.app'. This is the whole Arduino application, just drag it into your Applications Folder.



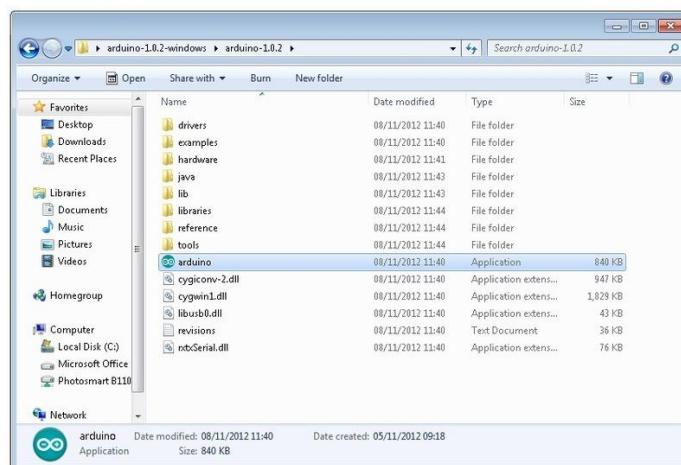


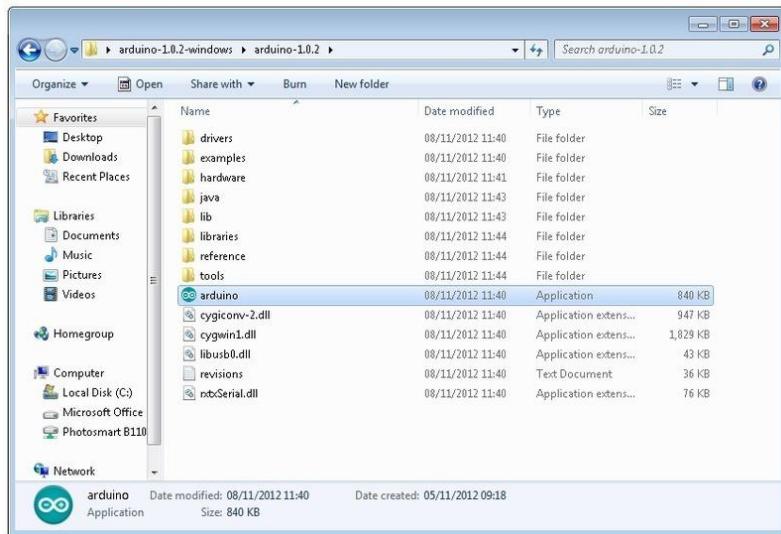
You can now find and launch the Arduino software in your Applications folder. As you are going to use it frequently, you may wish to right-click its icon in the dock and set it to 'Keep In Dock'.

There are many different LINUX distributions and the instructions for each distribution are a little different. The Arduino community has done a great job of putting together sets of instructions for each distribution.

Boards and Ports

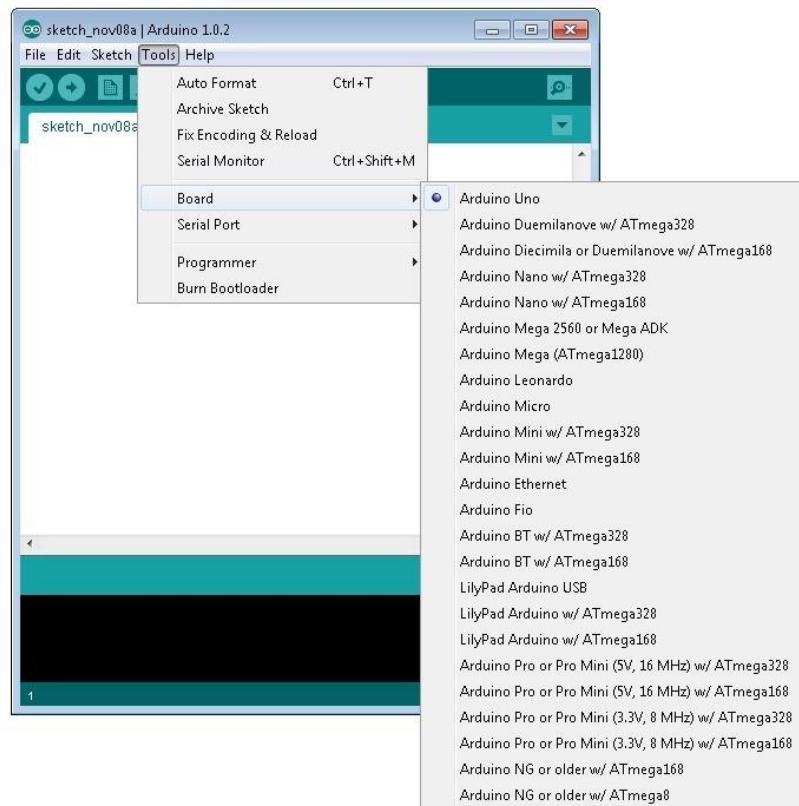
You are now ready to start the Arduino Software, so whatever platform you are using, open the Arduino folder and open the Arduino application contained within it.



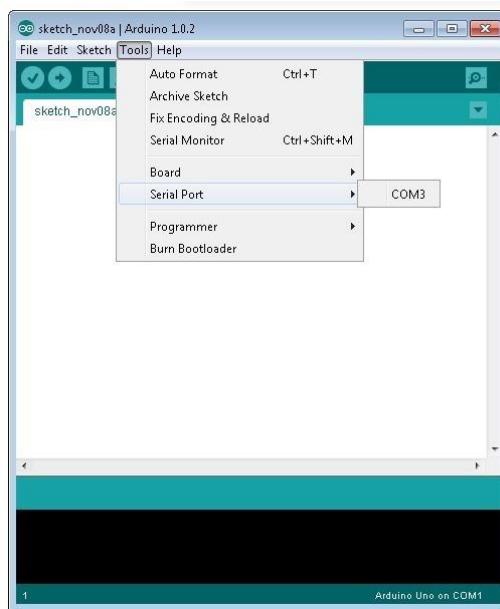


This will start the Arduino IDE, but before you can start programming, you have to tell the Arduino software which type of Arduino board you are using and also select the port to which it is connecting.

To tell the Arduino IDE which type of board you are using, from the 'Tools' menu, select Board and then 'Arduino Uno' or 'Leonardo' as appropriate.

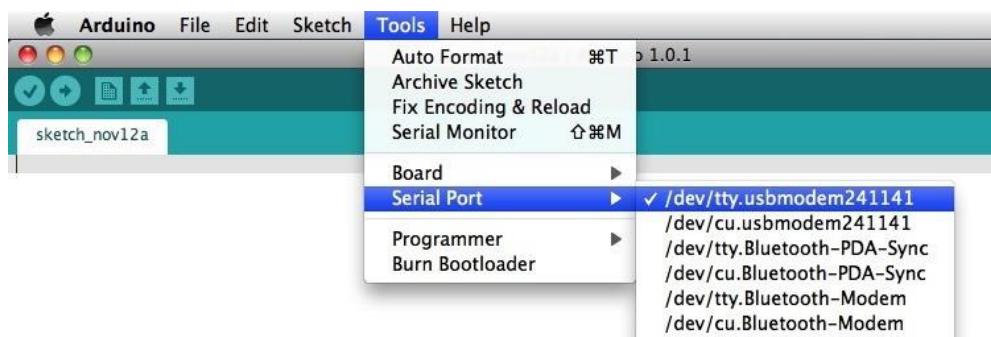


Also on the 'Tools' menu, you will find the 'Serial Port' option. Select this option.



If you are using Windows, there will probably only be one option here and it will either say COM3 or COM4. Even though there is only one option, you will still need to select it.

If you are using a Mac or Linux, there will be more options there, but it will usually be the top option in the list, as this will be the device most recently plugged in. This is useful, as the name of the port may not look like it has anything to do with Arduino. It will probably be called something like /dev/tty.usbmodemXXXX or /dev/ttyUSBn.



Lesson 1 Add Libraries

Once you are comfortable with the Arduino software and using the built-in functions, you may want to extend its functionality with additional libraries.

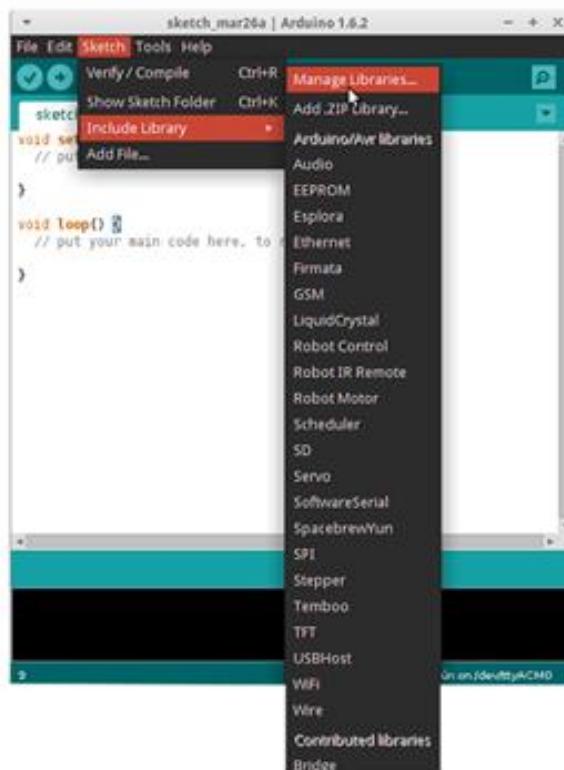
What are Libraries?

Libraries are a collection of code that makes it easy for you to connect to a sensor, display, module, etc. For example, the built-in LiquidCrystal library makes it easy to talk to character LCD displays. There are hundreds of additional libraries available on the Internet for download. The built-in libraries and some of these additional libraries are listed in the reference. To use these additional libraries, you will need to install them.

How to Install a Library

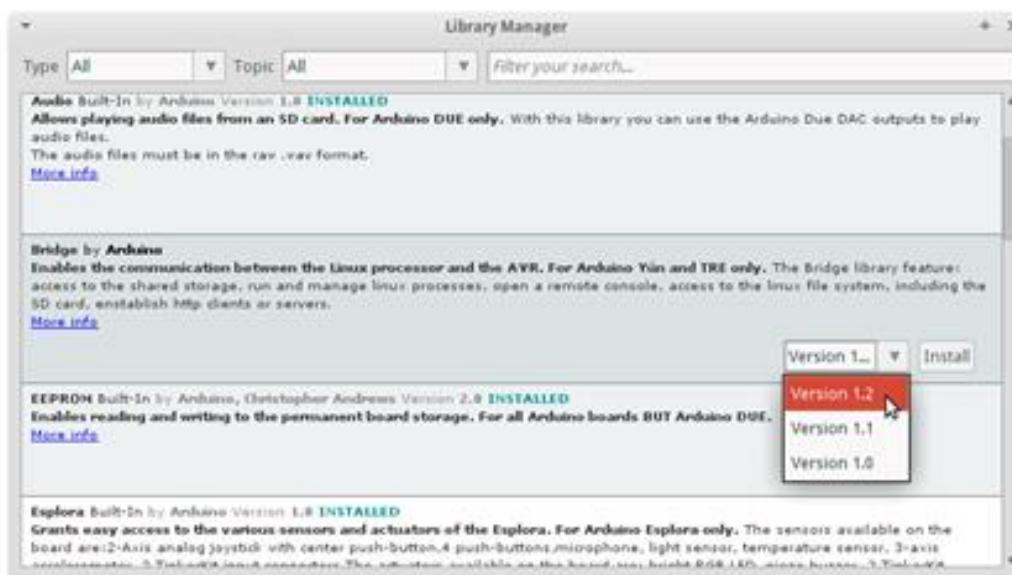
Using the Library Manager

To install a new library into your Arduino IDE, you can use the Library Manager (available from IDE version 1.6.2). Open the IDE and click Sketch > Include > Library > Manage Libraries.

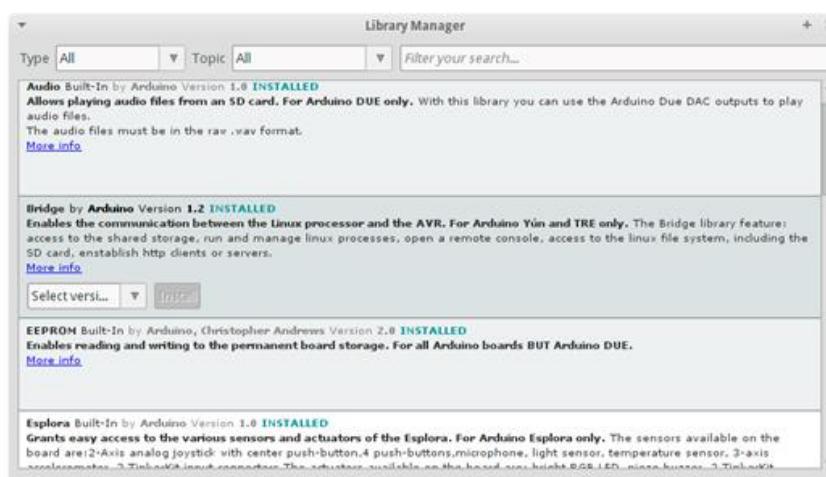


The library manager will open and you will find a list of libraries that are already installed or

ready for installation. In this example, we will install the Bridge library. Scroll down the list to find it, then select the version of the library you want to install. Sometimes, only one version of the library is available. If the version selection menu does not appear, don't worry; it is normal.



Finally click on install and wait for the IDE to install the new library. Downloading may take time depending on your connection speed. Once it has finished, an Installed tag should appear next to the Bridge library. You can close the library manager.

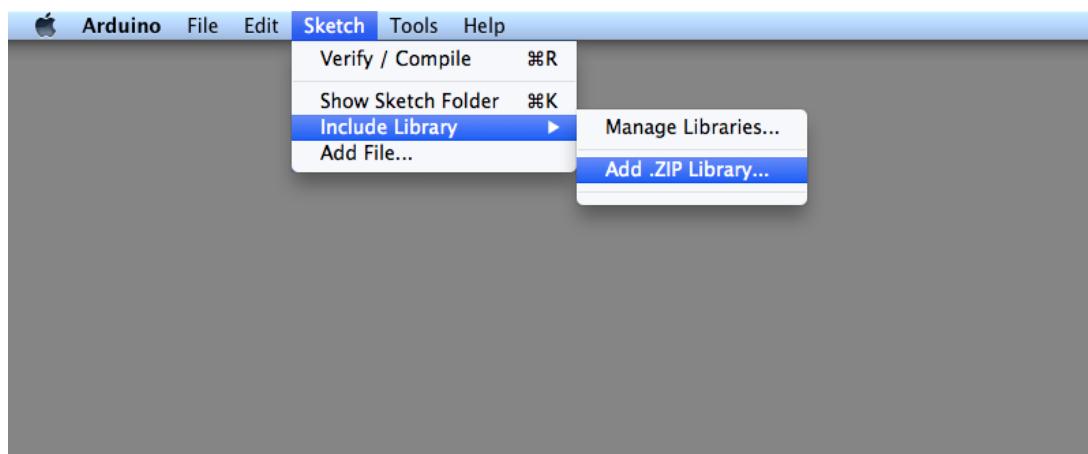


You can now find the new library available in the Include Library menu. If you want to add your own library open a new issue on github.

Importing a .zip Library

Libraries are often distributed as a ZIP file or folder. The name of the folder is the name of the library. Inside the folder will be the following: .cpp file, .h file, often a keywords.txt file, examples folder, and other files required by the library. Starting with version 1.0.5, you can install third-party libraries in the IDE. Do not unzip the downloaded library; leave it as-is.

In the Arduino IDE, navigate to Sketch > Include Library > Add .ZIP Library.



You will be prompted to select the library you would like to add. Navigate to the .zip file's location and open it.

Return to the Sketch > Import Library menu. You should now see the library at the bottom of the drop-down menu. It is ready to be used in your sketch. The zip file will have been expanded in the libraries folder in your Arduino sketches directory.

NB: The Library will be available to use in sketches, but examples for the library will not be shown in the File > Examples until after the IDE has restarted.

Manual installation

To install the library, first, quit the Arduino application. Then unzip the ZIP file containing the library. For example, if you're installing a library called "ArduinoParty", uncompress ArduinoParty.zip. It should contain a folder called ArduinoParty, with files like ArduinoParty.cpp and ArduinoParty.h inside. (If the .cpp and .h files aren't in a folder, you'll need to create one. In this case, you'd make a folder called "ArduinoParty" and move into it all the files that were in the ZIP file, like ArduinoParty.cpp and ArduinoParty.h.)

Drag the ArduinoParty folder into this folder (your libraries folder). Under Windows, it will likely

be called "My Documents\Arduino\libraries". For Mac users, it will likely be called "Documents/Arduino/libraries". On Linux, it will be the "libraries" folder in your sketchbook.

Your Arduino library folder should now look like this (on Windows):

My Documents\Arduino\libraries\ArduinoParty\ArduinoParty.cpp

My Documents\Arduino\libraries\ArduinoParty\ArduinoParty.h

My Documents\Arduino\libraries\ArduinoParty\examples

....

or like this (on Mac and Linux):

Documents/Arduino/libraries/ArduinoParty/ArduinoParty.cpp

Documents/Arduino/libraries/ArduinoParty/ArduinoParty.h

Documents/Arduino/libraries/ArduinoParty/examples

....

There may be more files than just the .cpp and .h files so make sure they're all there. (The library won't work if you put the .cpp and .h files directly into the libraries folder or if they're nested in an extra folder. For example: Documents\Arduino\libraries\ArduinoParty.cpp and Documents\Arduino\libraries\ArduinoParty\ArduinoParty.cpp won't work.)

Restart the Arduino application. Make sure the new library appears in the Sketch > Import Library menu. That's it. You've installed a library!

Summary

In this lesson, we will install all the libraries that we will use in the tutorial. Open the libraries folder and install the ZIP files one by one so we won't need to do this step in the following lessons. We just connect the component as the schematic and upload the code provided. Then the kit will work.

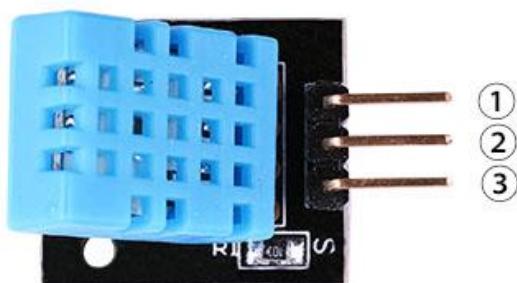
Lesson 2 TEMP AND HUMIDITYMODULE

Overview

In this tutorial we will learn how to use a DHT11 Temperature and Humidity Sensor.

It's accurate enough for most projects that need to keep track of humidity and temperature readings.

Again we will be using a Library specifically designed for these sensors that will make our code short and easy to write.



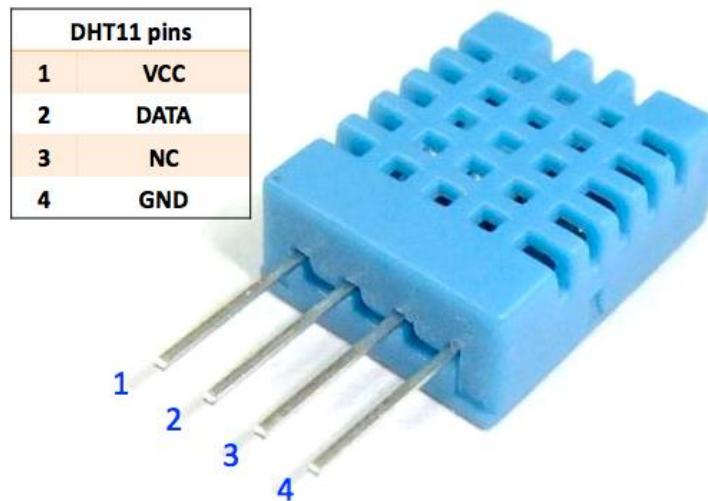
- 1.GND:ground
- 2.VCC: 3.3V-5V DC
- 3.OUTPUT

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x DHT11 module
- (x) x F-M wires

Component Introduction

Temp and humidity sensor:



DHT11 digital temperature and humidity sensor is a composite Sensor contains a calibrated digital signal output of the temperature and humidity. Application of a dedicated digital modules collection technology and the temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long-term stability. The sensor includes a resistive sense of wet components and an NTC temperature measurement devices, and connected with a high-performance 8-bit microcontroller.

Applications: HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, weather stations, home appliances, humidity regulator, medical and other humidity measurement and control.

Product parameters

Relative humidity:

Resolution: 16Bit

Repeatability: $\pm 1\%$ RH

Accuracy: At 25°C $\pm 5\%$ RH

Interchangeability: fully interchangeable

Response time: 1 / e (63%) of 25°C 6s

1m / s air 6s

Hysteresis: $<\pm 0.3\%$ RH

Long-term stability: $<\pm 0.5\%$ RH / yr in

Temperature:

Resolution: 16Bit

Repeatability: $\pm 0.2^\circ\text{C}$

Range: At 25°C $\pm 2^\circ\text{C}$

Response time: $1 / e$ (63%) 10s

Electrical Characteristics

Power supply: DC $3.5 \sim 5.5\text{V}$

Supply Current: measurement 0.3mA standby $60\mu\text{A}$

Sampling period: more than 2 seconds

Pin Description:

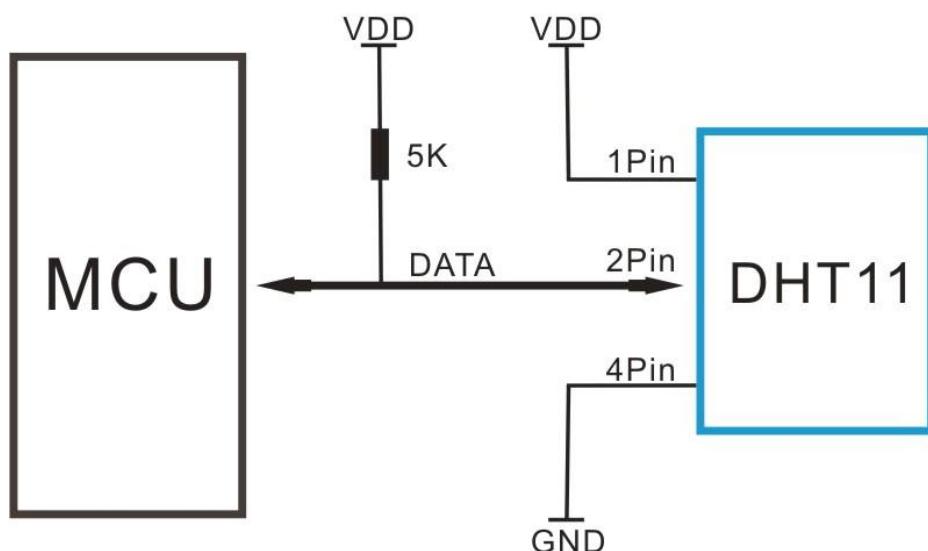
1, the VDD power supply $3.5 \sim 5.5\text{V}$ DC

2 DATA serial data, a single bus

3, NC, empty pin

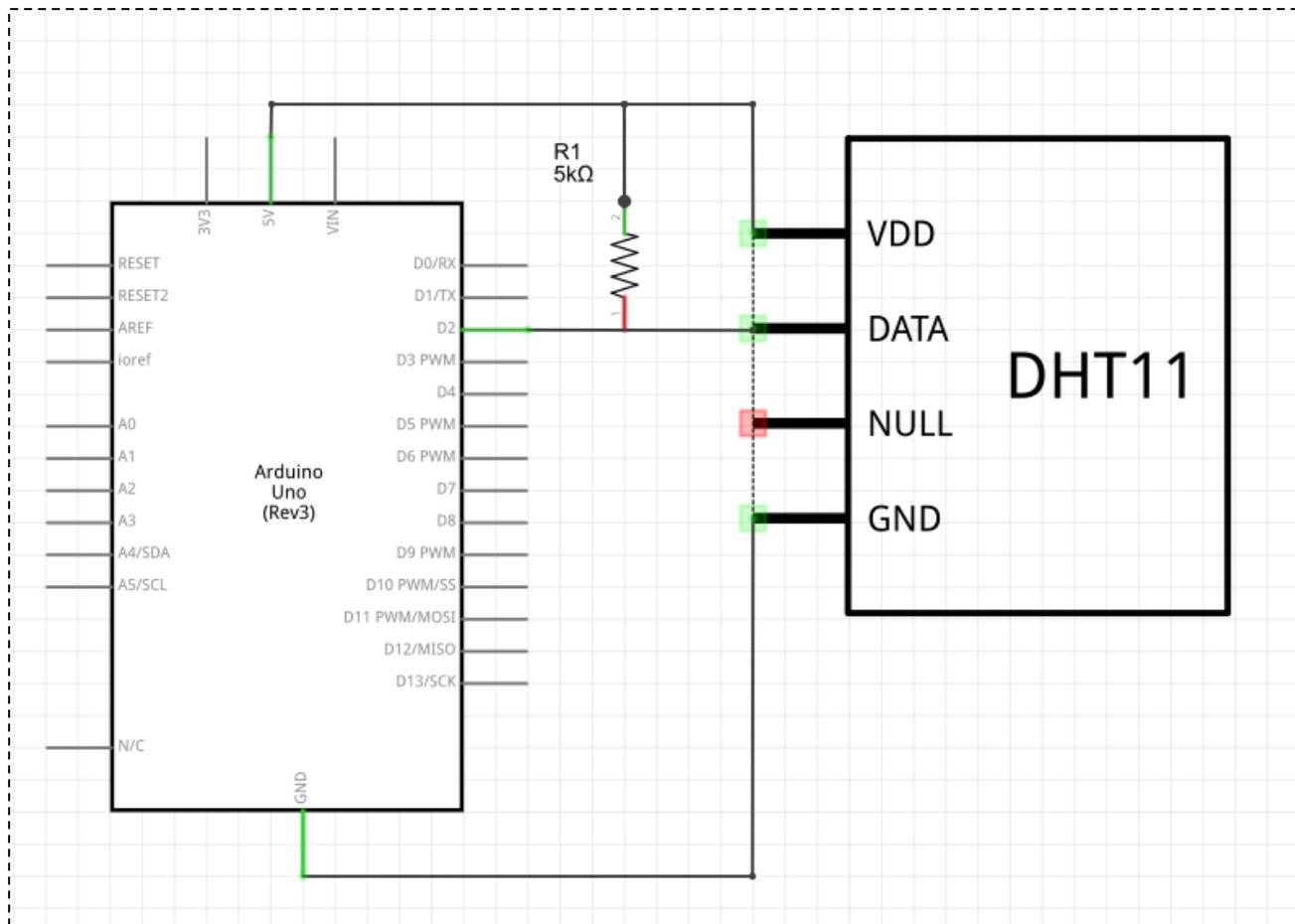
4, GND ground, the negative power

Typical Application

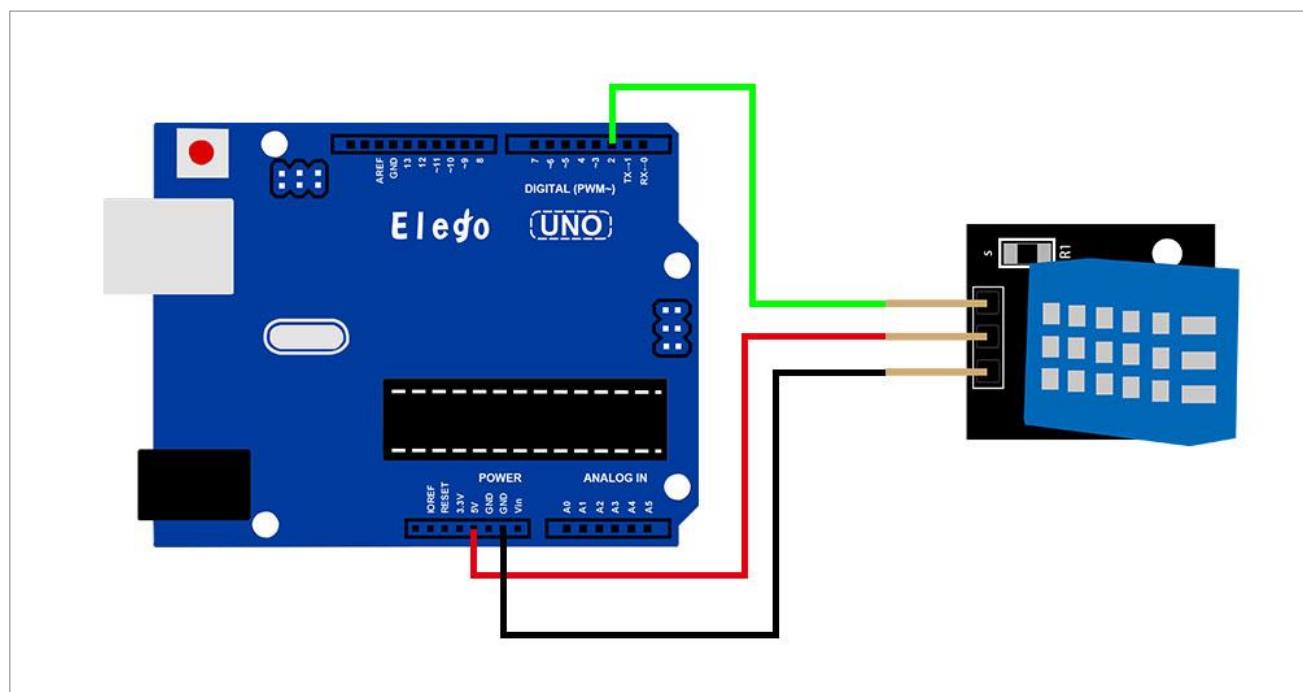


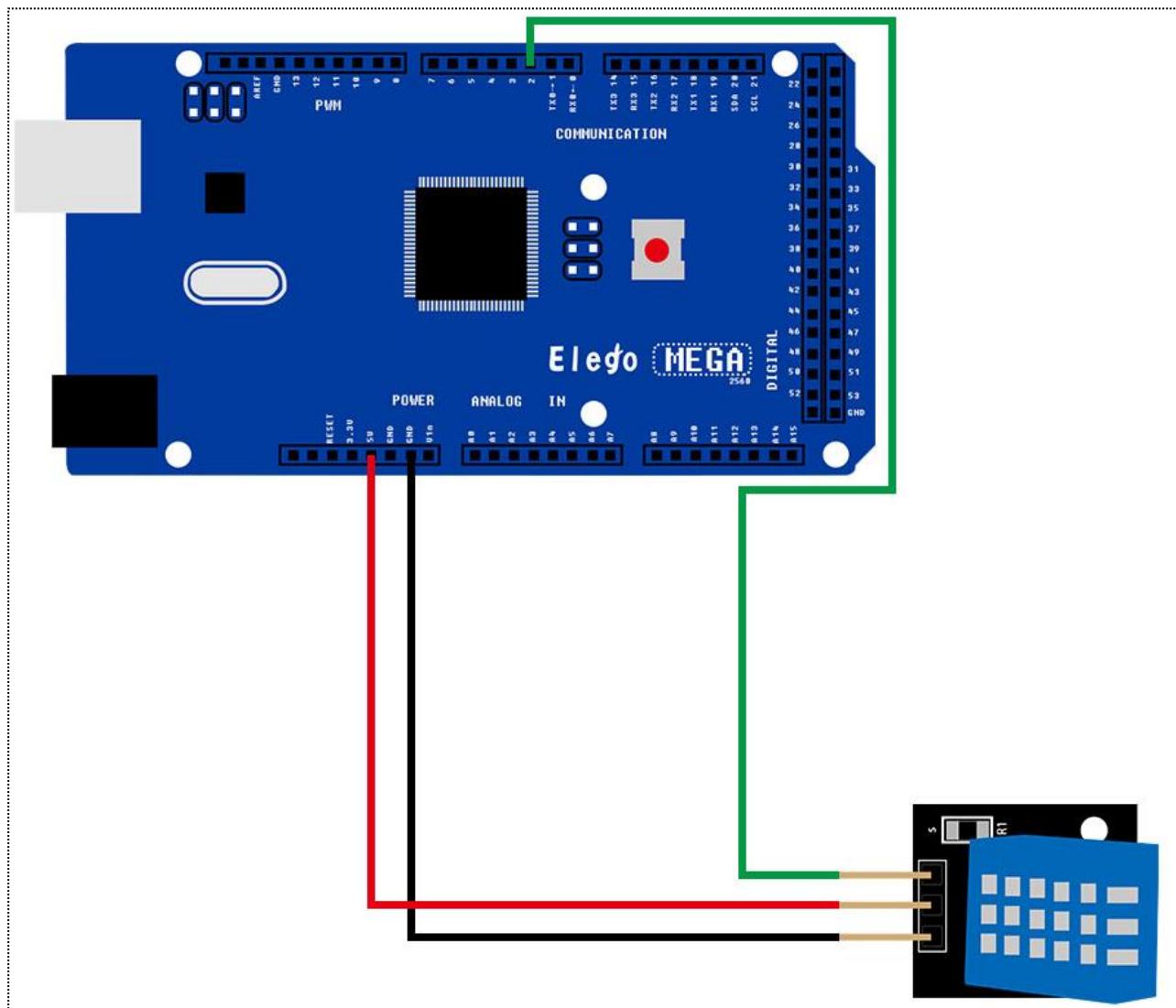
Connection

Schematic

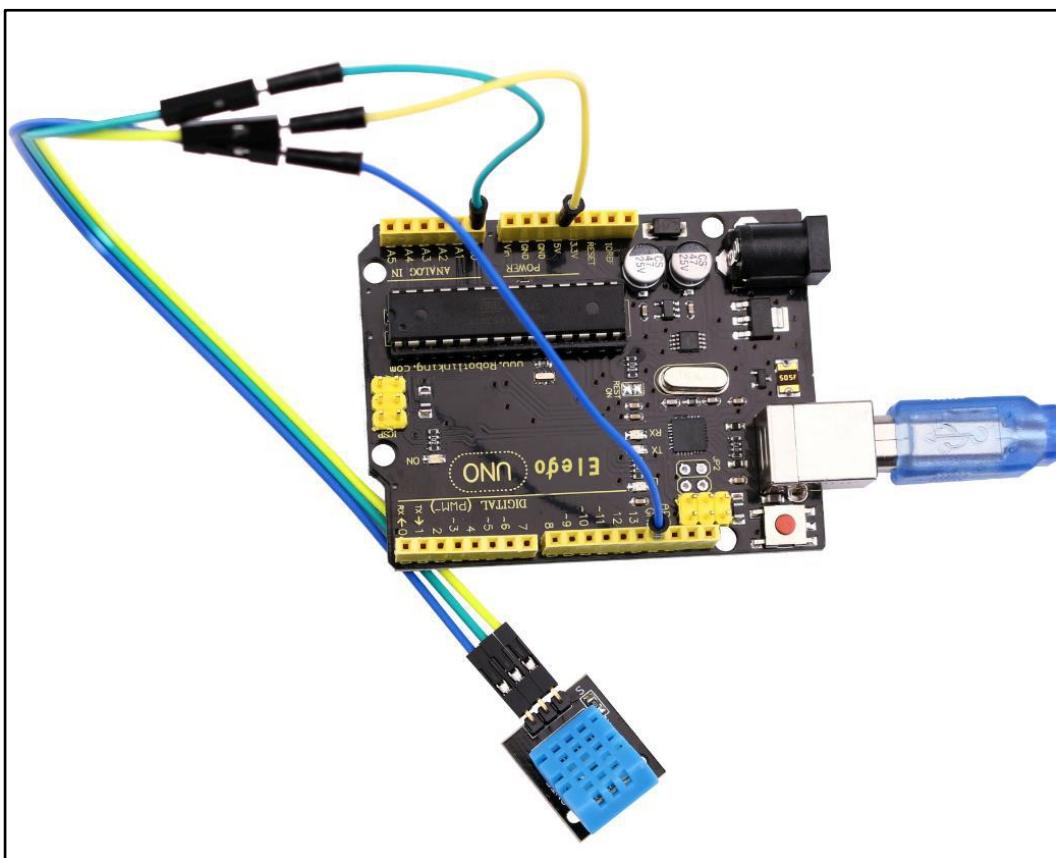


wiring diagram

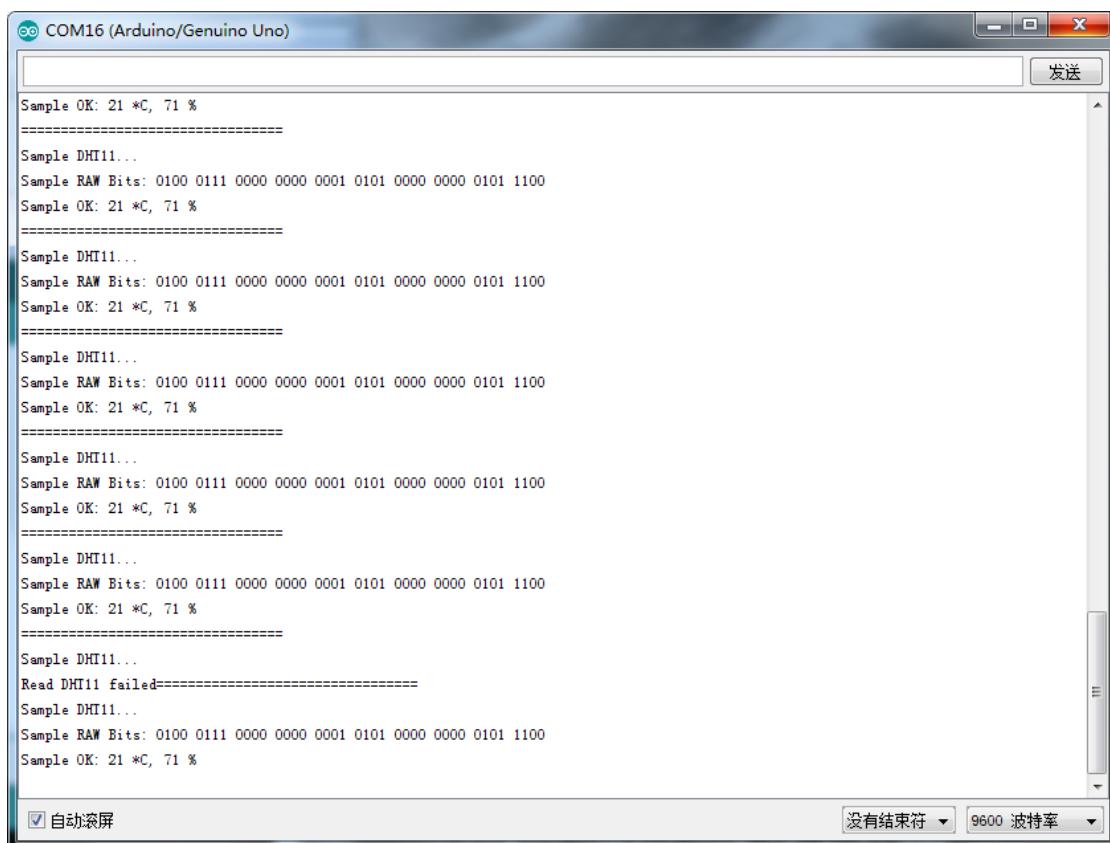




Result



Upload the program then open the monitor, we can see the data as below:(It shows the temperature of the environment, we can see it is 21 degree)



```
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Read DHT11 failed=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
```

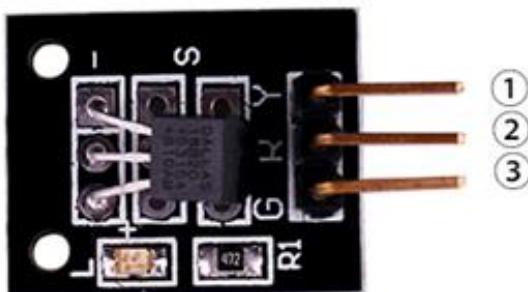
Lesson 3 DS18B20 DIGITAL TEMPERATURE SENSOR MODULE

Overview

In this experiment, we will learn how to use DS18B20 module test the environmental temperature and make a thermometer.

Since the previous temperature sensor output is analog. So we need to add additional A/D and D/A chip into the line transformation. More over, the Arduino external port is not rich resources and the utilization rate is not high. These cause a big challenge. So we are create the Ds18b20 module.

The new DS18B20 Temperature Sensor Module is very good solve the problem. It have the characteristic of the economy, unique 1-wire bus and it can fully apply the Arduino platform. Users can easily form a sensor network through using this module.



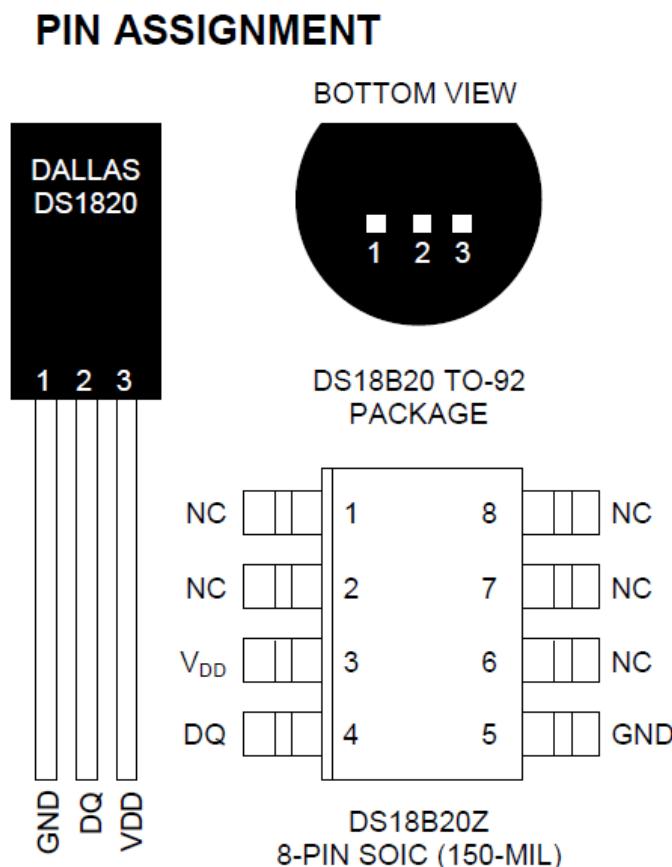
- 1.OUTPUT
- 2.VCC: 3.3V-5V DC
- 3.GND:ground

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x DS18B20 module
- (x) x F-M wires

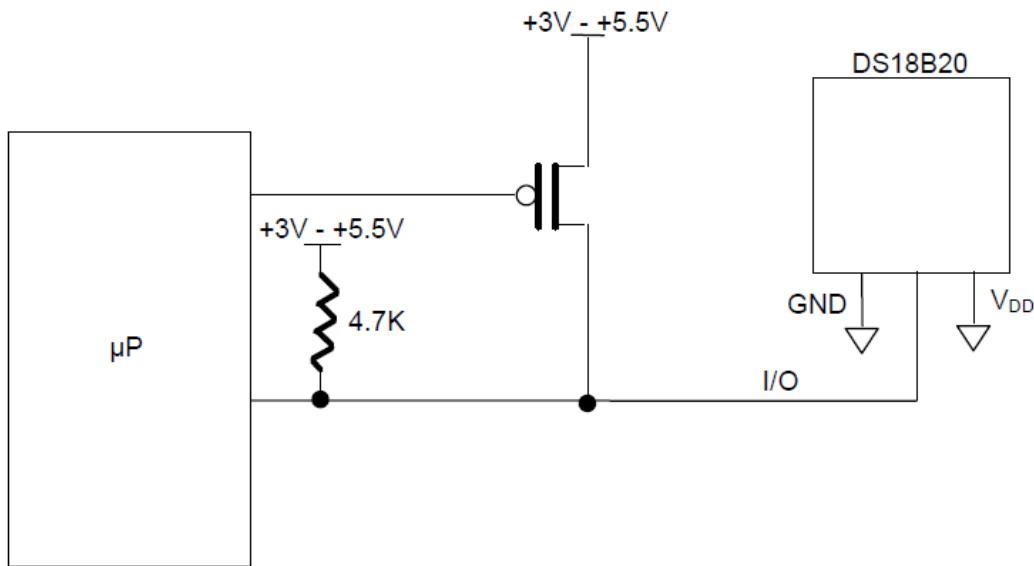
Component Introduction

DS18B20:



PIN DESCRIPTION

- GND - Ground
DQ - Data In/Out
V_{DD} - Power Supply Voltage
NC - No Connect



Principle

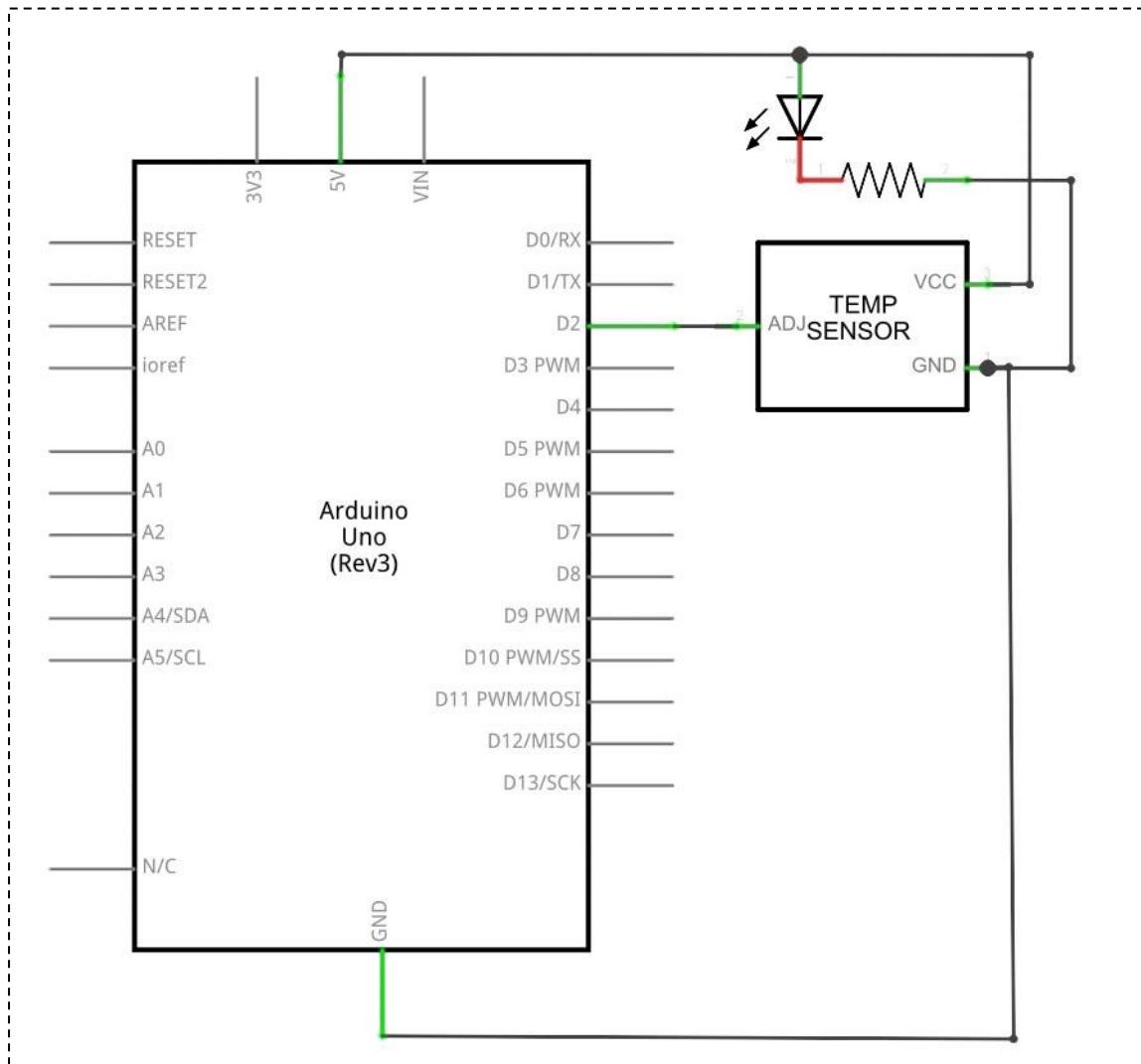
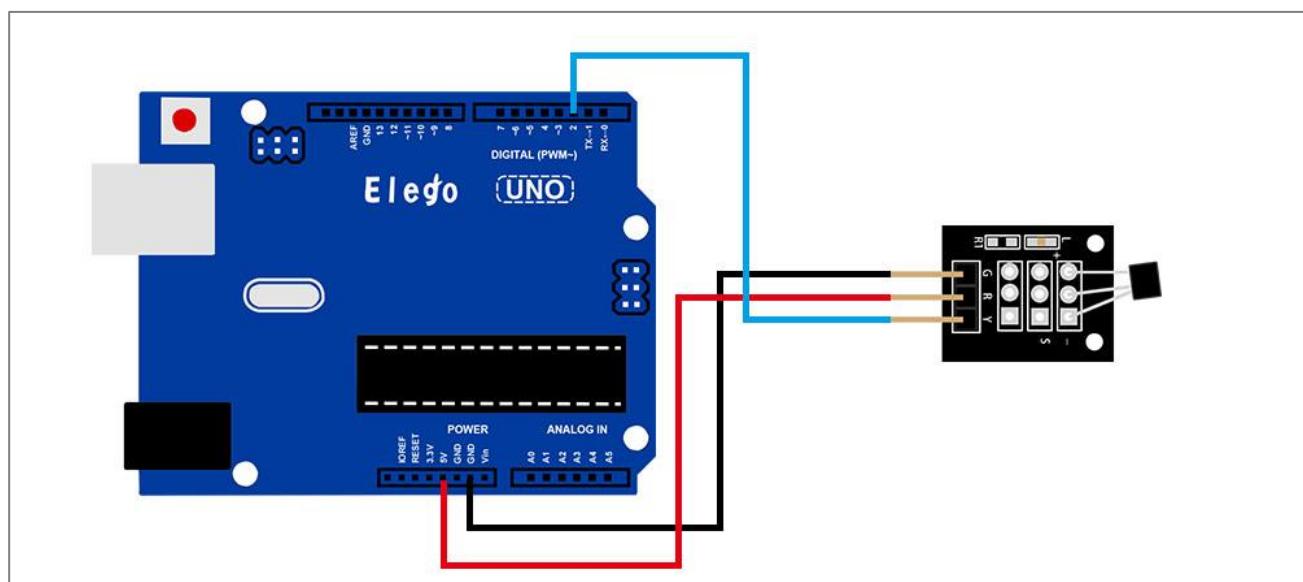
DS18B20 module is using a single bus. The power supply voltage range of 3.0 V to 5.5 V and no standby power supply. It can Measuretemperaturerangefor-55degree to +125degree with accuracy of +/-0.5°C.

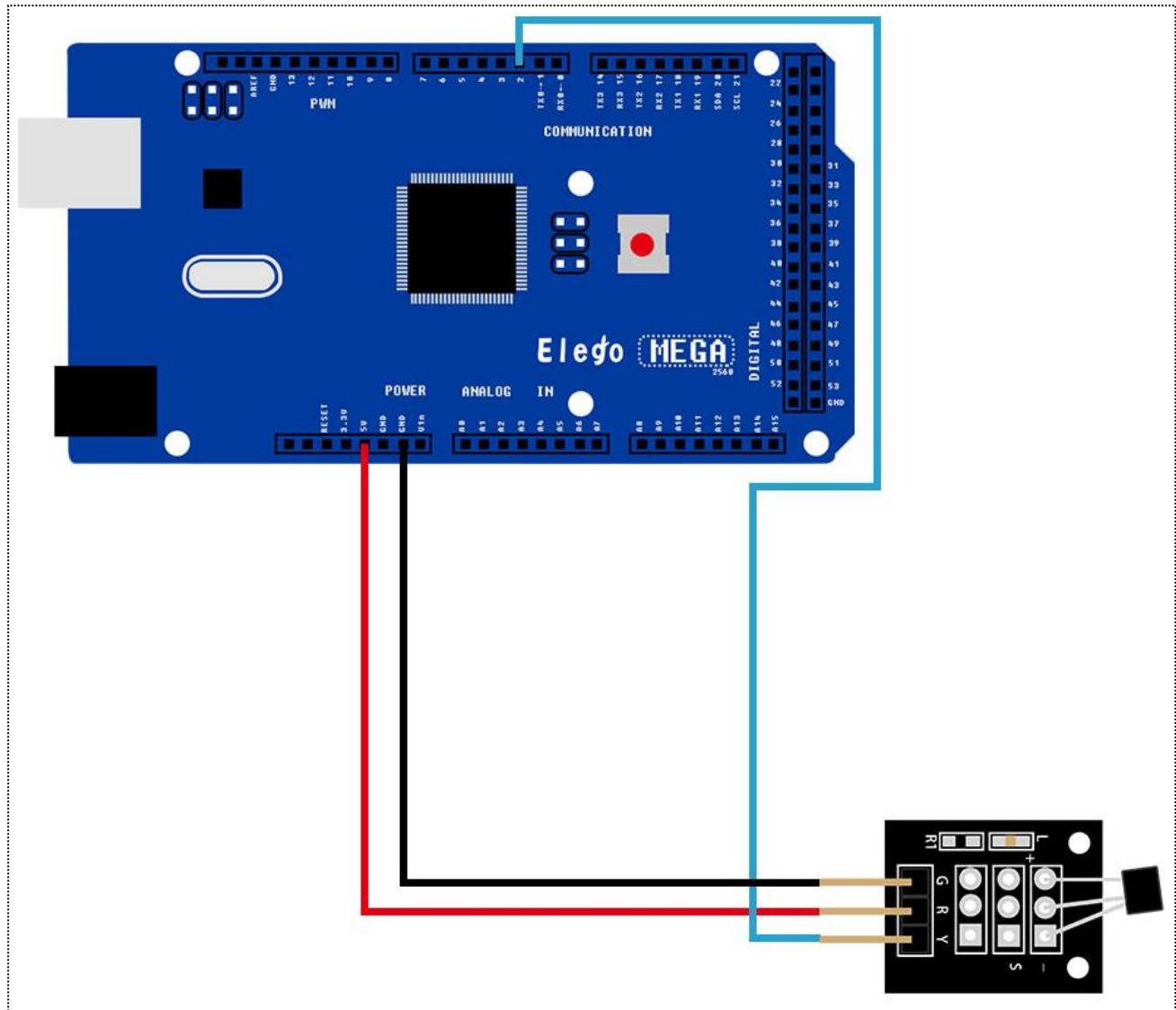
The programmable DPI of temperature sensor is From 9 to 12. temperature conversion is 12 digits lattice type. maximum is 750 milliseconds. Families can be defined non-volatile temperature alarm Settings.

Each DS18B20 contains a unique serial number so that multiple ds18b20 scan exist in a bus.

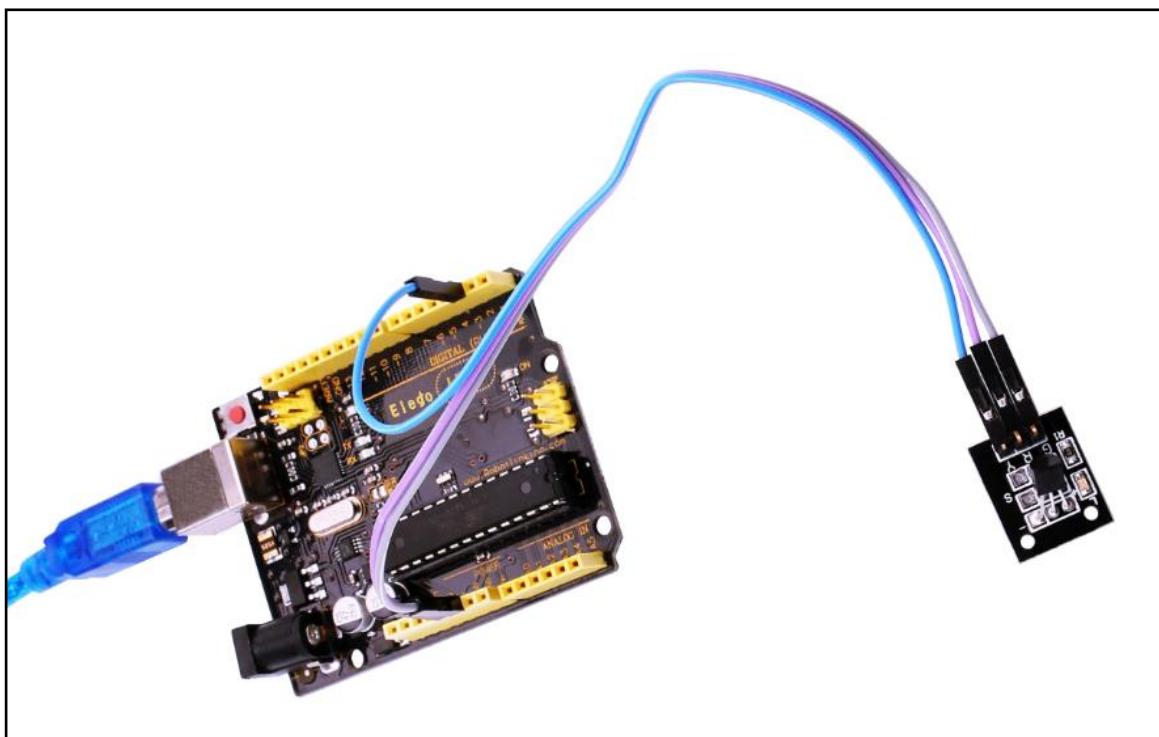
Connection

Schematic

**wiring diagram**

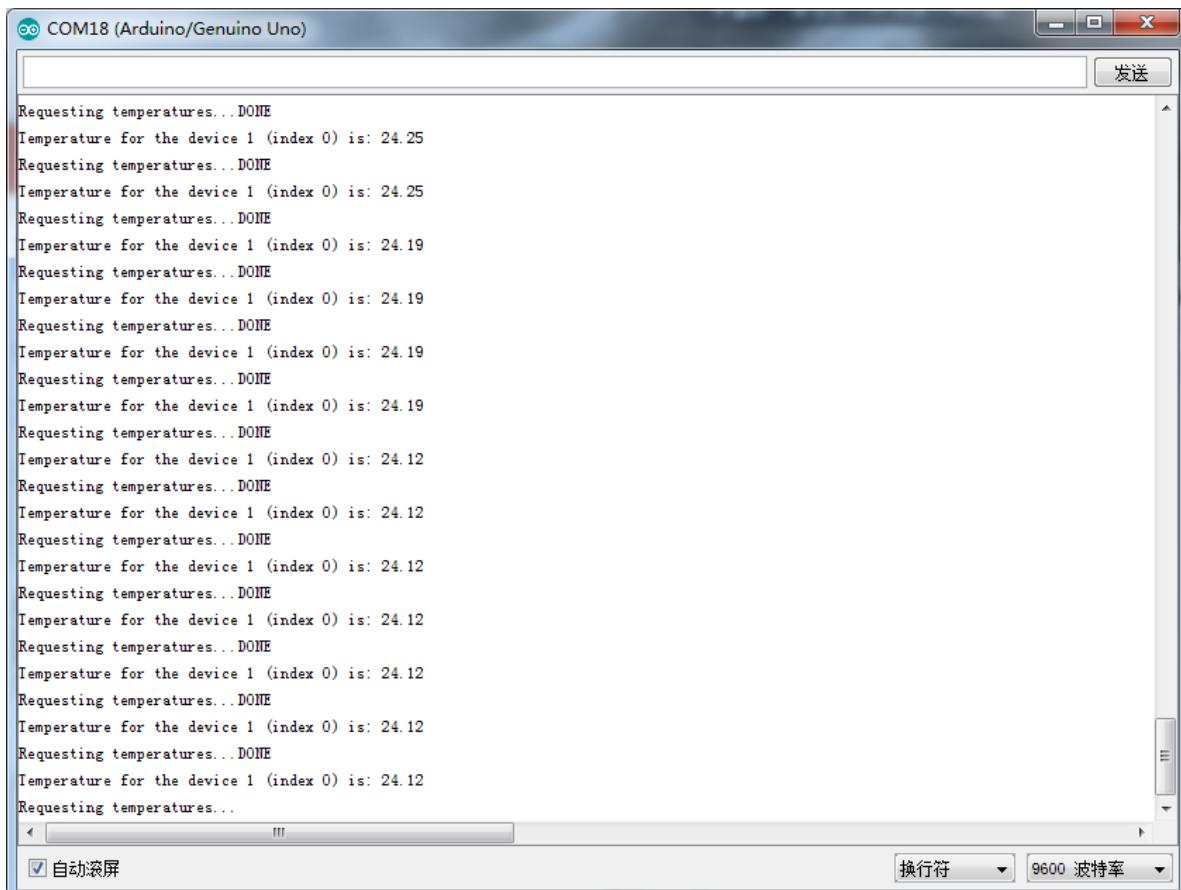


Result



Temperature sensor can detect temperature in numbers of different places at the same time.

Upload the program then open the monitor, we can see the data as below:

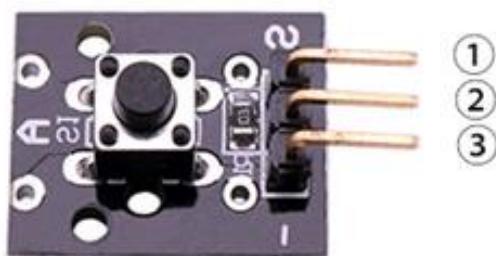


```
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.25
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.25
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.19
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.19
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.19
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.19
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.12
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.12
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.12
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.12
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.12
Requesting temperatures... DONE
Temperature for the device 1 (index 0) is: 24.12
Requesting temperatures...
```

Lesson 4 BUTTON SWITCH MODULE

Overview

In this experiment, we will learn how to use button switch.



- 1.OUTPUT
- 2.VCC: 3.3V-5V DC
- 3.GND:ground

Component Required:

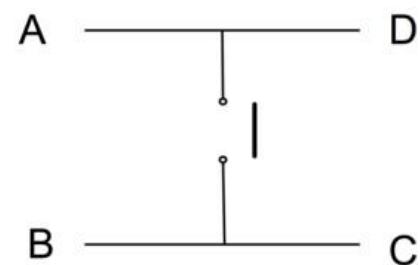
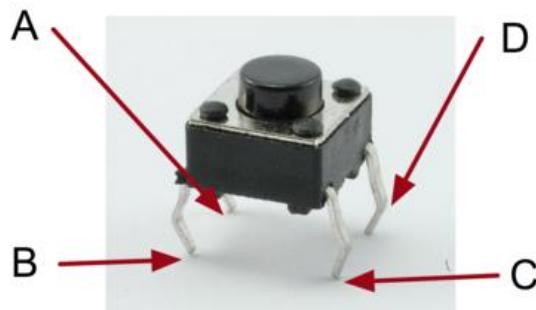
- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Button module
- (x) x F-M wires

Component Introduction

PUSH SWITCHES:

Switches are really simple components. When you press a button or flip a lever, they connect two contacts together so that electricity can flow through them.

The little tactile switches that are used in this lesson have four connections, which can be a little confusing.



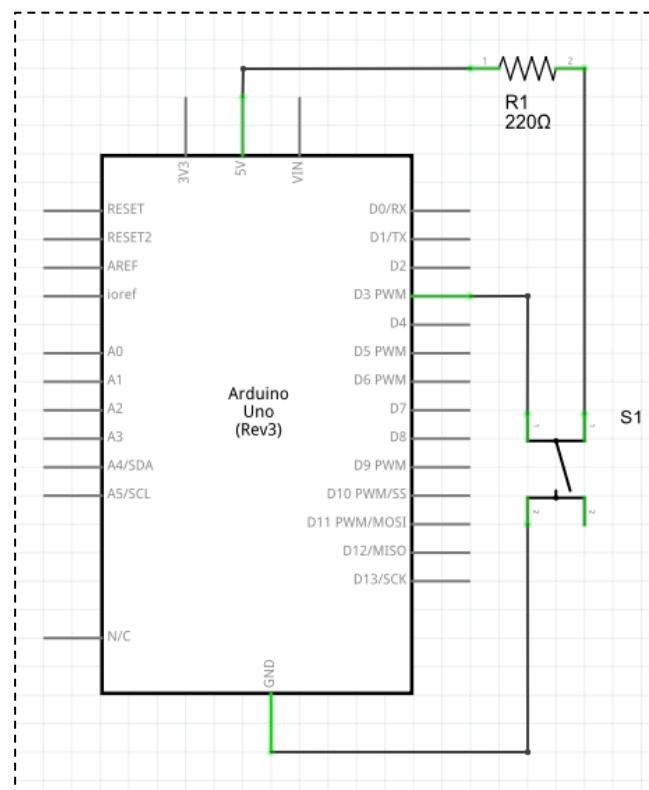
Actually, there are only really two electrical connections, as inside the switch package pins B and C are connected together, as are A and D.

Principle

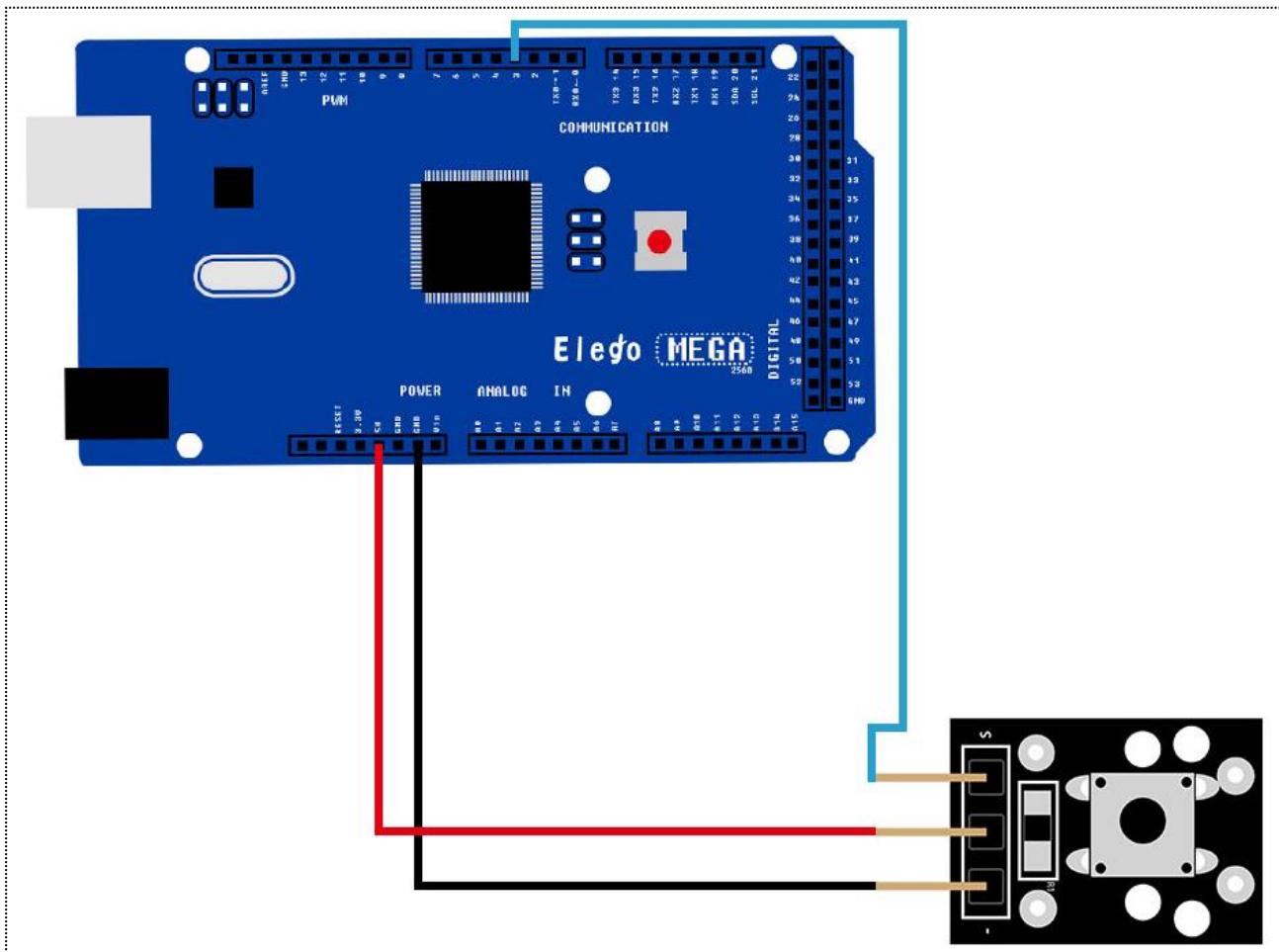
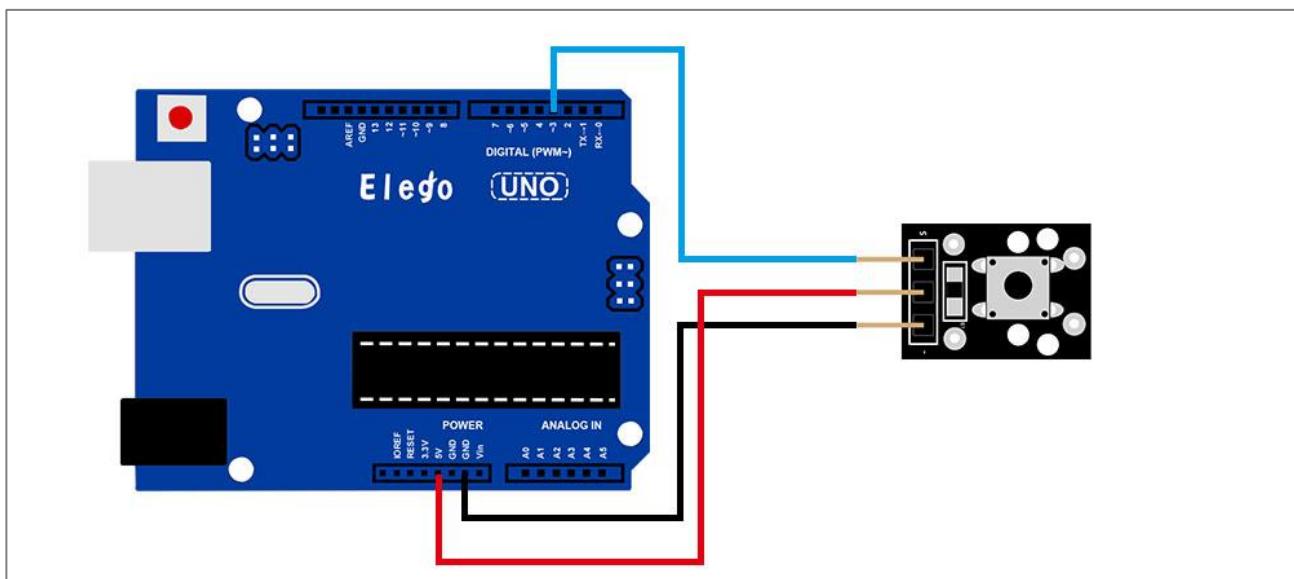
BUTTON SWITCH and number 13 port have the built-in LED simple circuit. To produce a switch flasher, we can use connect the digital port 13 to the built-in LED and connect the BUTTON SWITCHS port to number3 port of Elegoo Uno board. When the switch sensing, LED twinkle light to the switch signal.

Connection

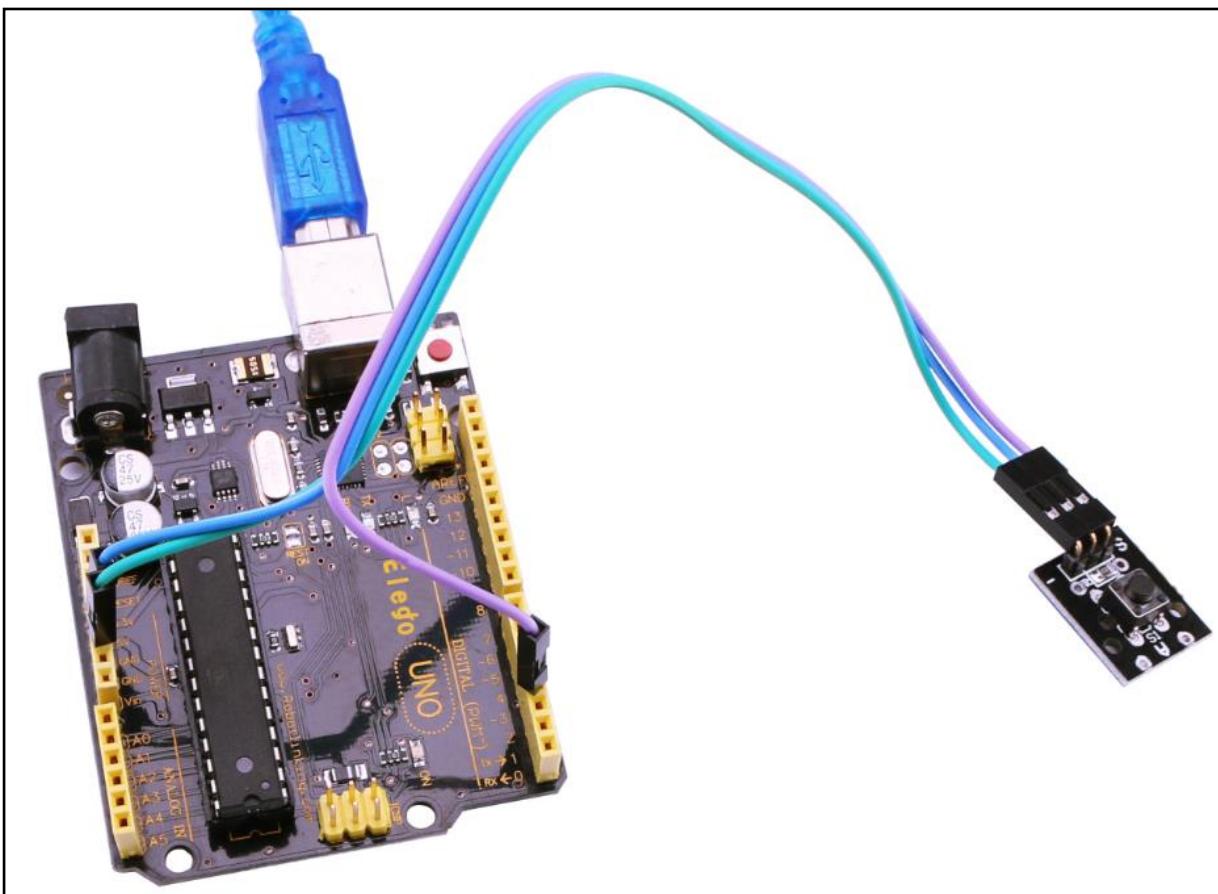
Schematic



wiring diagram



Result



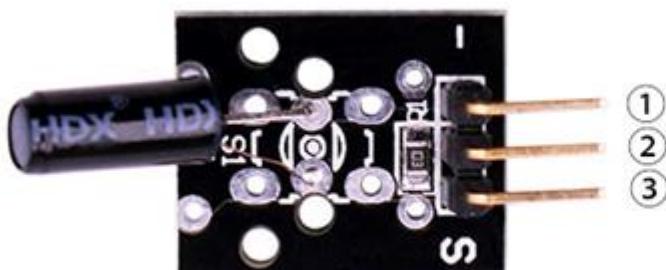
Connect the circuit as above and upload the program. Then push the button, you can see led on and off.

Lesson 5 THREE TYPE OF SWITCH MODULE

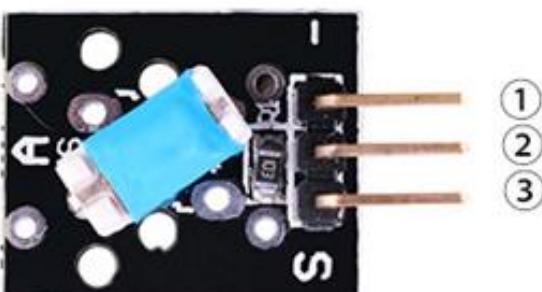
Overview

In this experiment, we will learn how to use switch modules.

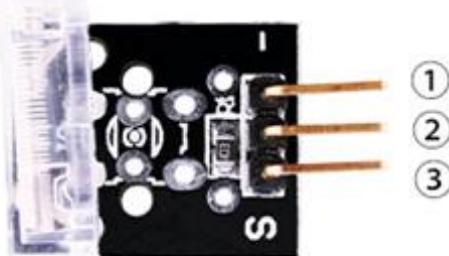
Including shock switch module, tilt switch module and knock switch module.



- 1.GND:ground
- 2.VCC: 3.3V-5V DC
- 3.OUTPUT



- 1.GND:ground
- 2.VCC:3.3V-5V DC
- 3.OUTPUT



- 1.GND:ground
- 2.VCC:3.3V-5V DC
- 3.OUTPUT

Component Required:

(1) x Elegoo Uno R3

(1) x USB cable

(1) x Shock switch module

(1) x tilt switch module

(1) x Knock switch module

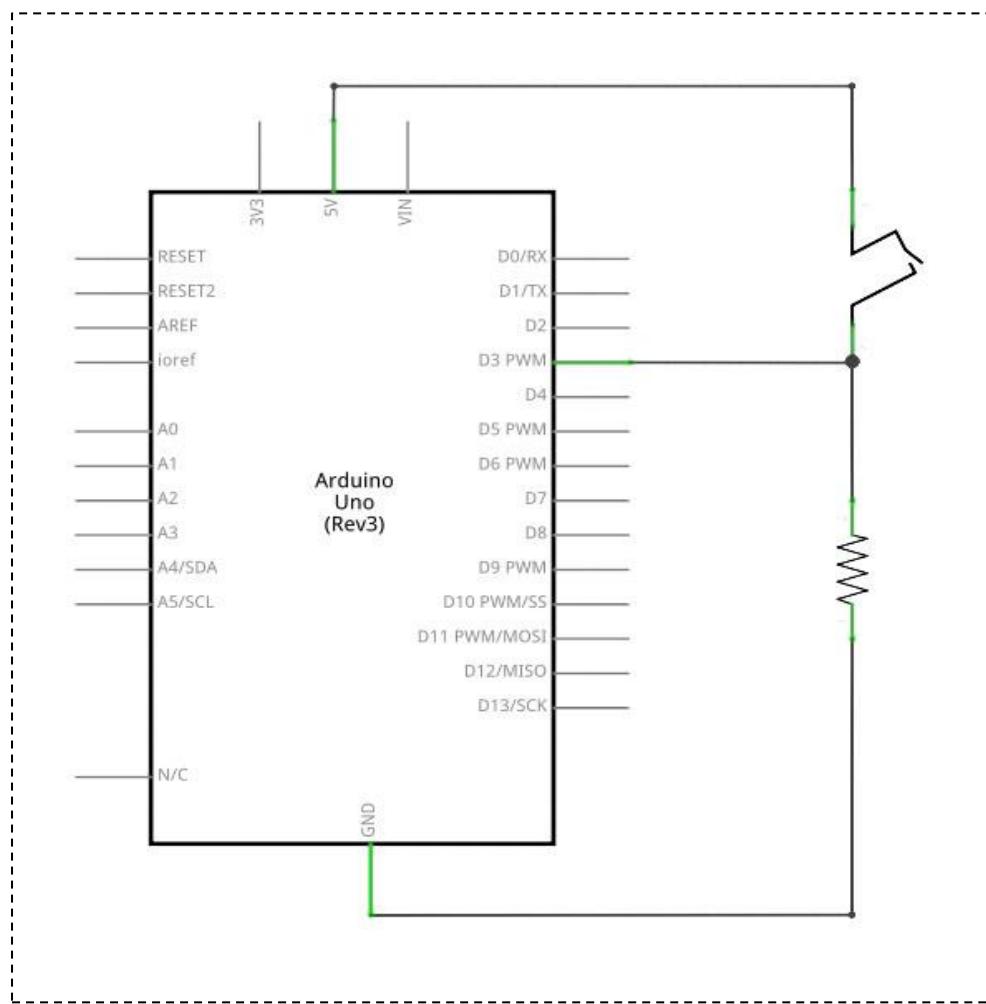
(x) x F-M wires

Principle

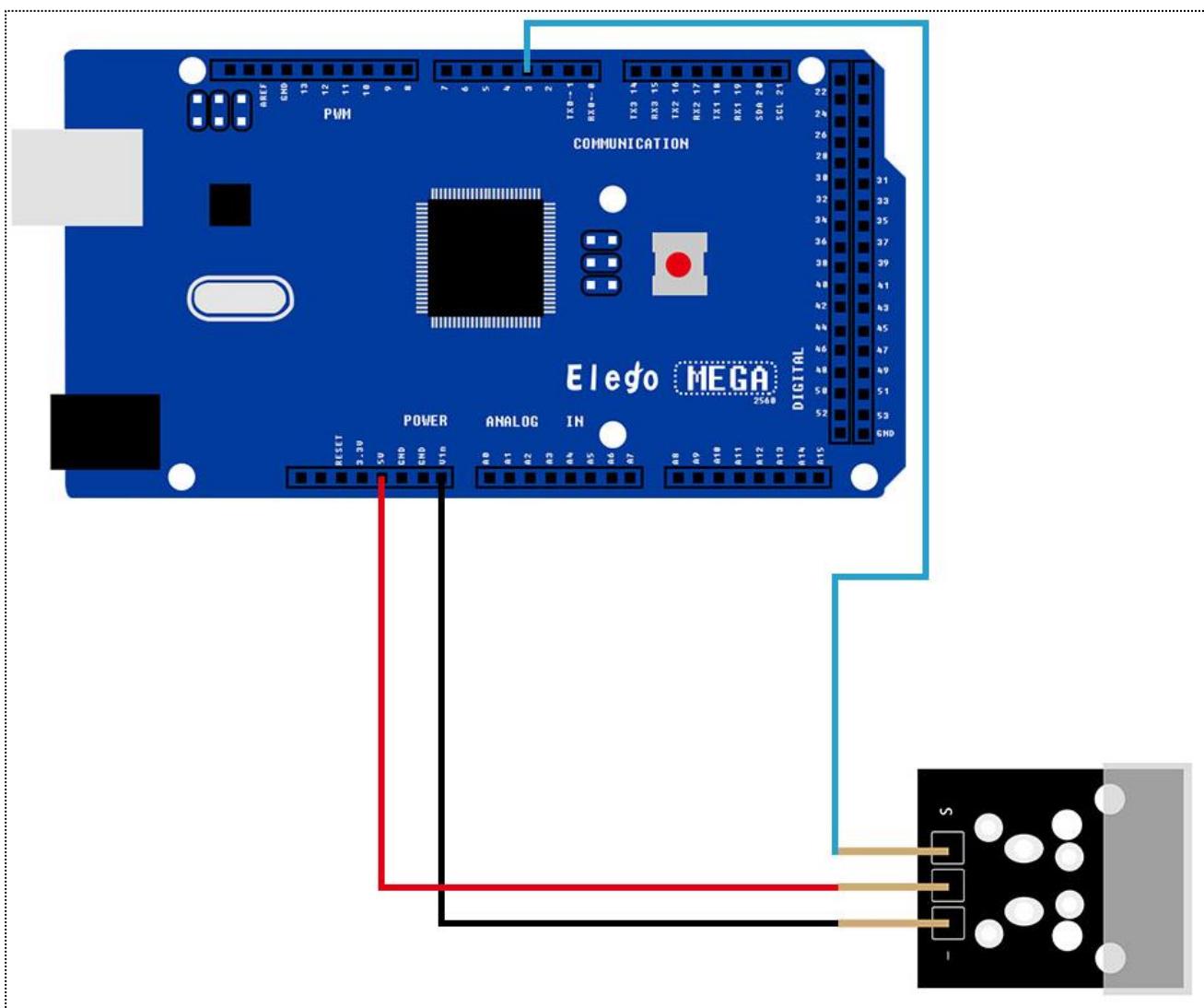
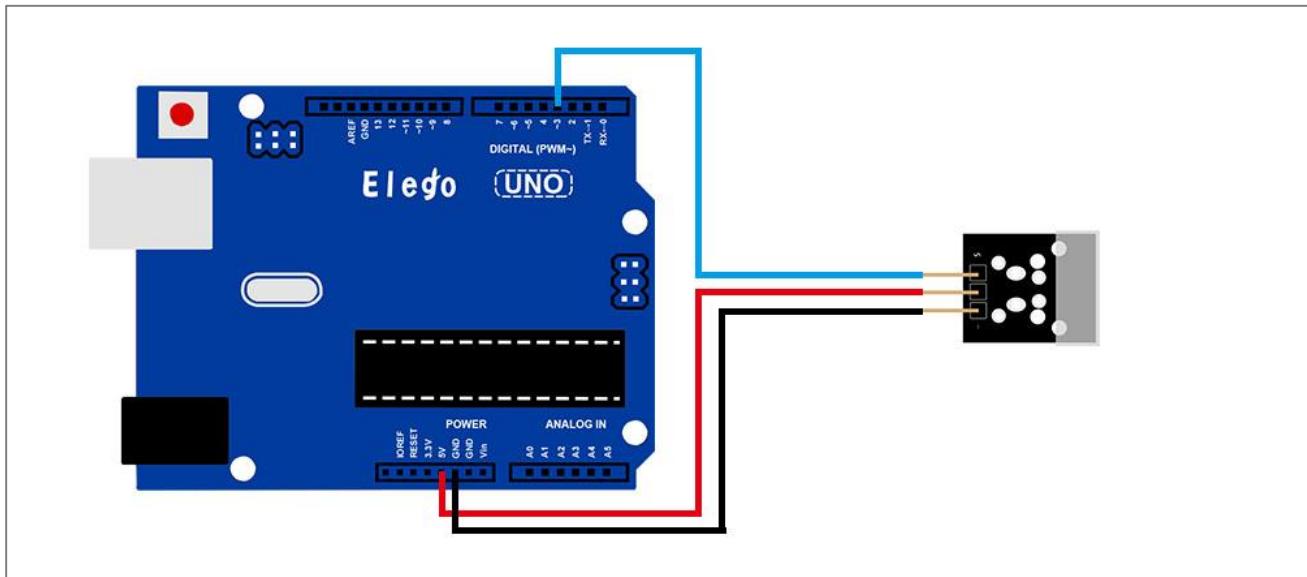
Switch and number 13 port have the built-in LED simple circuit. To produce a SWITCH flasher, we can use connect the digital port 13 to the built-in LED and connect the SWITCH S port to number 3 port of Elegoo Uno board. When the SWITCH sensing, LED twinkle light to the SWITCH signal.

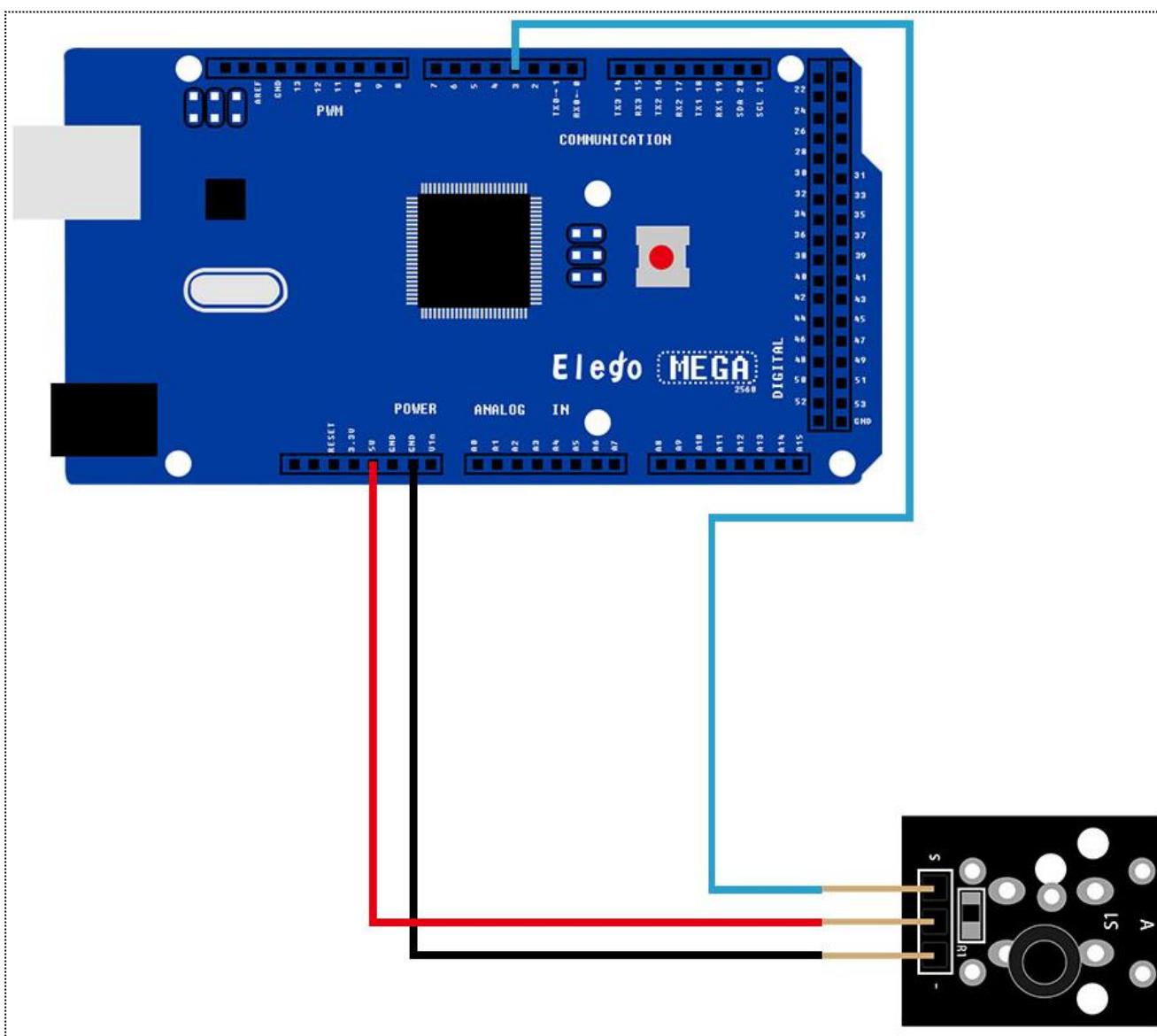
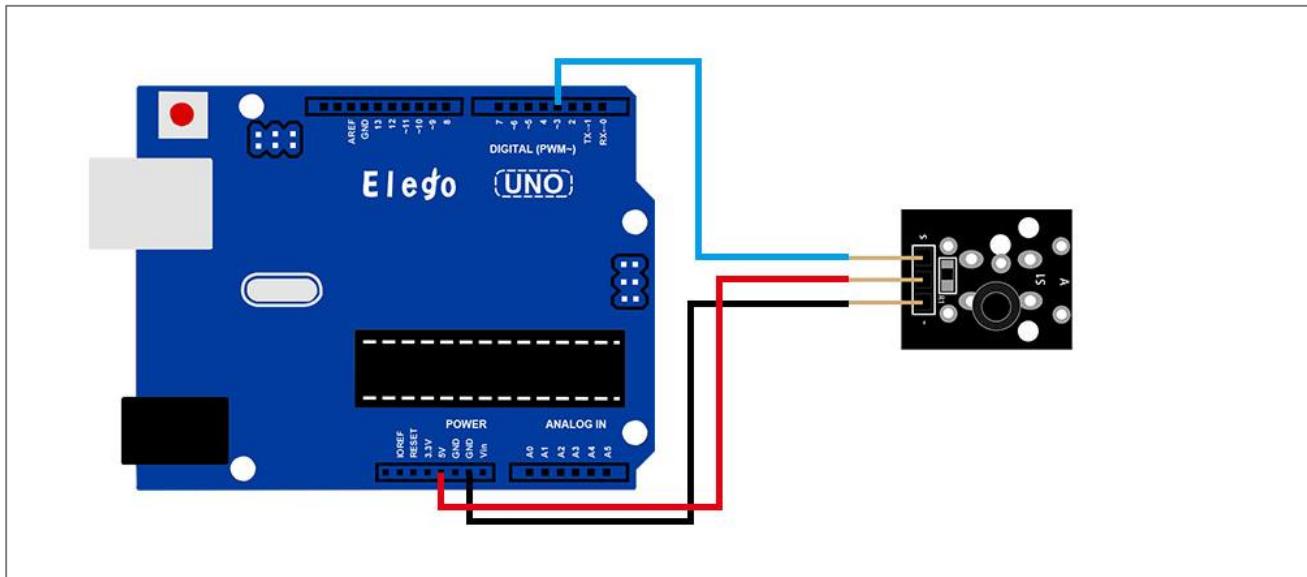
Connection

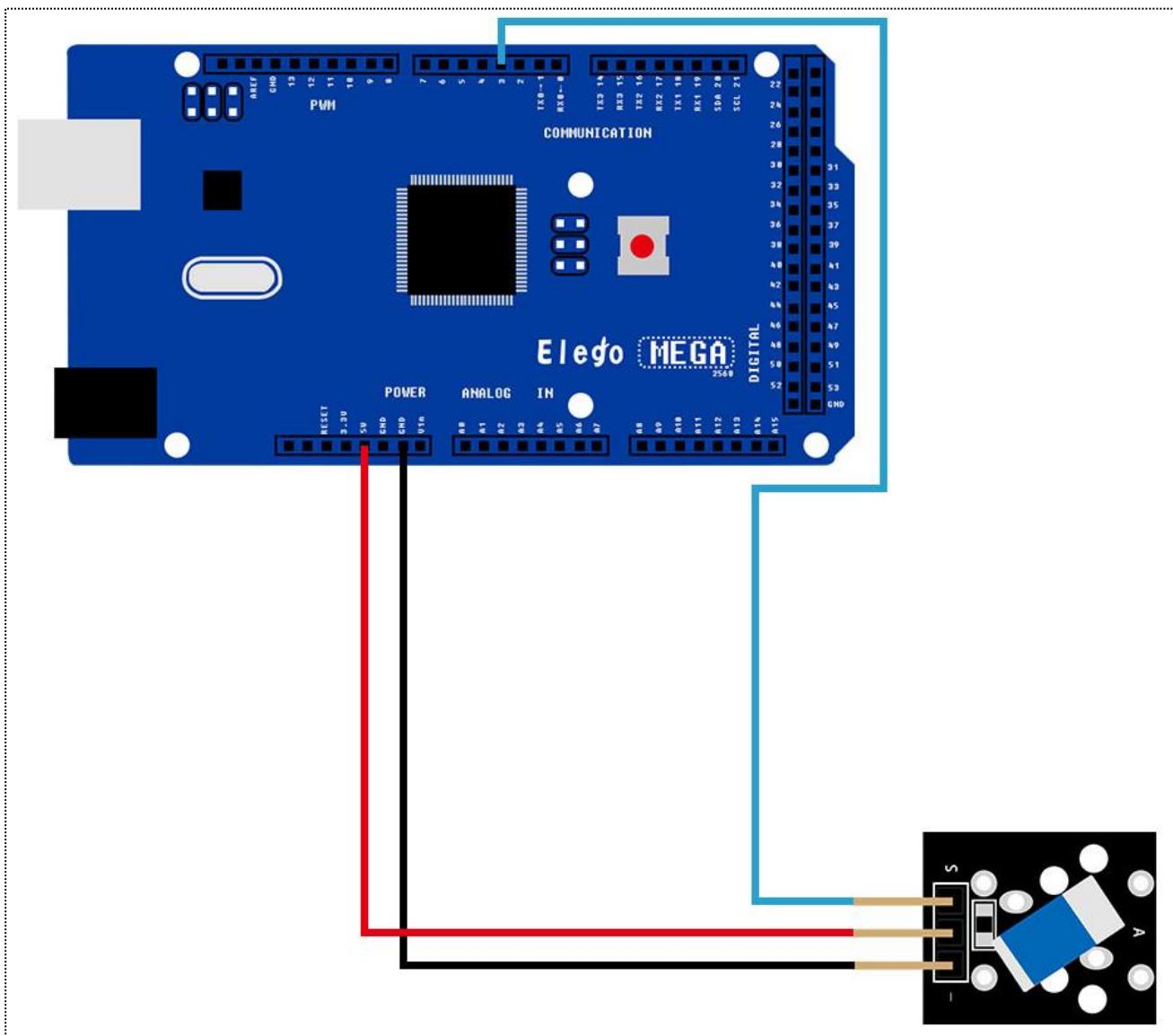
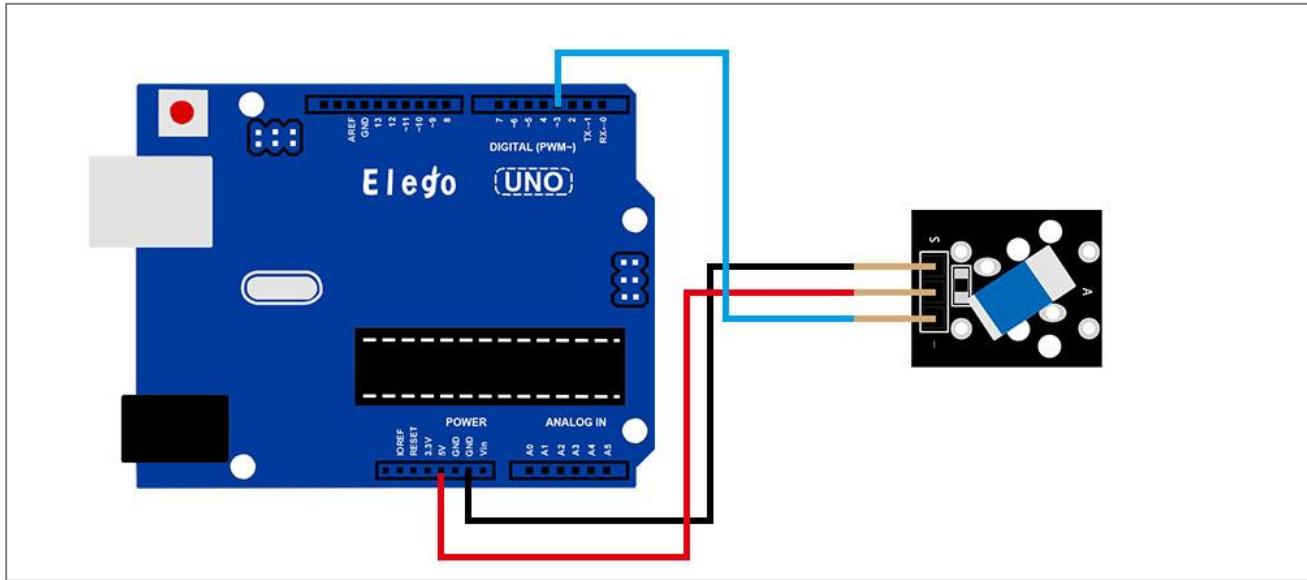
Schematic



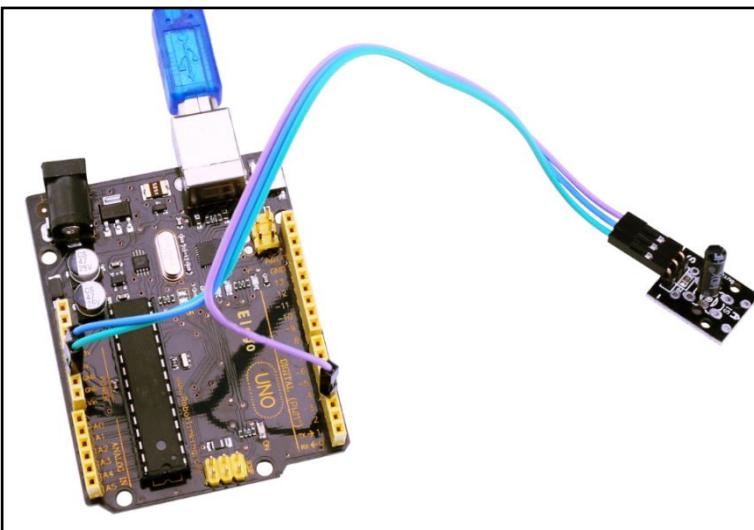
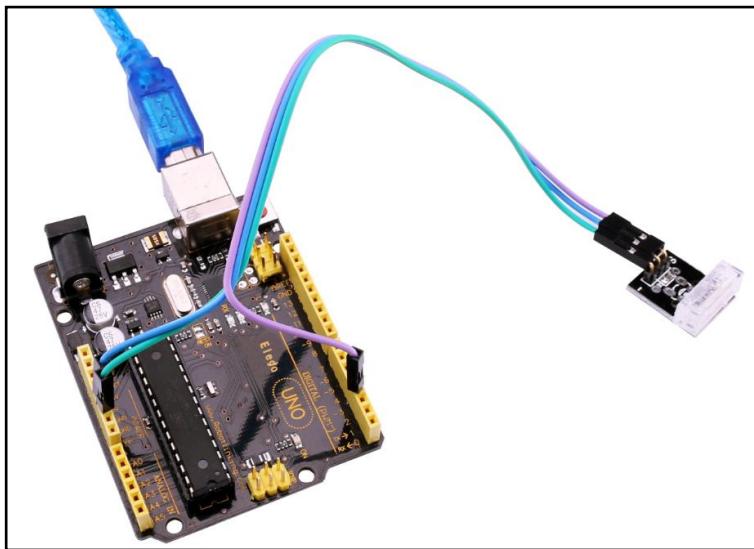
wiring diagram

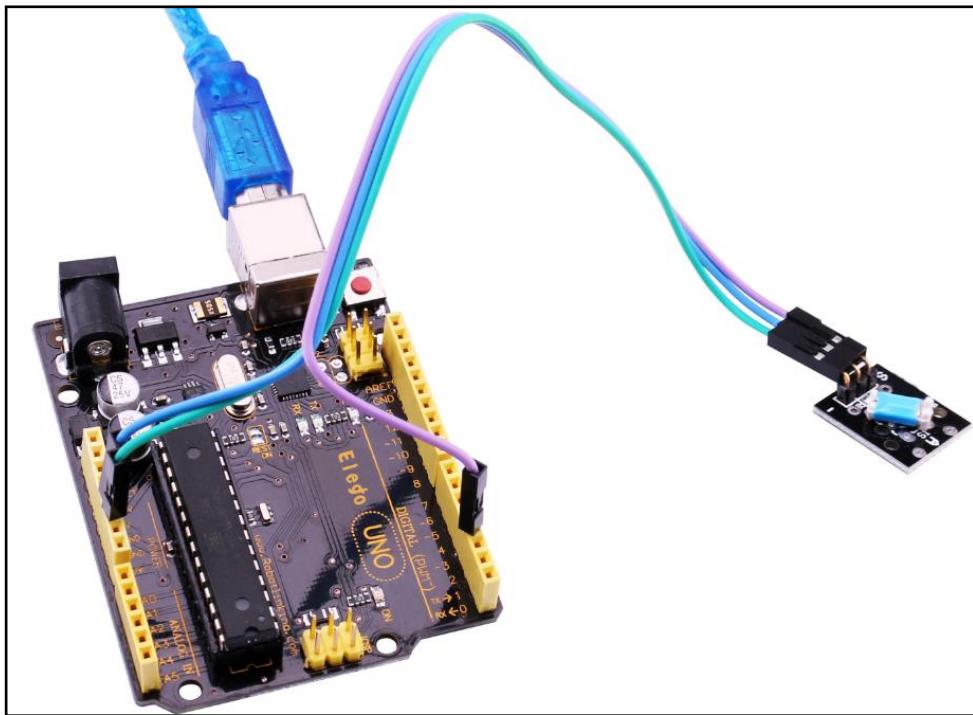






Result





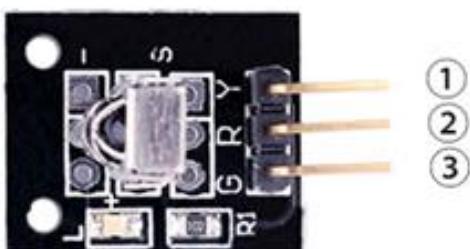
Connect the circuit as above and upload the program. Then lean or shock the sensor, you can see led on and off.

Lesson 6 IR RECEIVER AND EMISSION

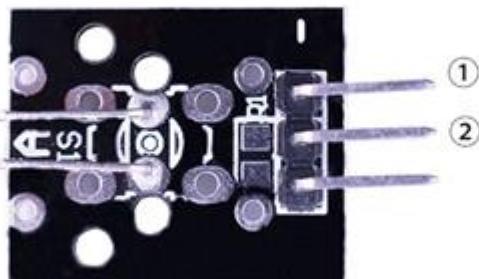
Overview

In this experiment, we will learn how to use Infrared Receiver and IR Emission module.

In fact now in our daily life they play an important role, a lot of household electrical appliances are used to this kind of device, such as air conditioning, TV, DVD, etc. Actually it is based on its wireless remote sensing and it is very convenient by using them.



- 1.OUTPUT
- 2.VCC:3.3V-5V DC
- 3.GND:ground



- 1.GND:ground
- 2.OUTPUT

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x IR Receiver module
- (1) x IR Emission module

(x) x F-M wires

Component Introduction

IR RECEIVER SENSOR:

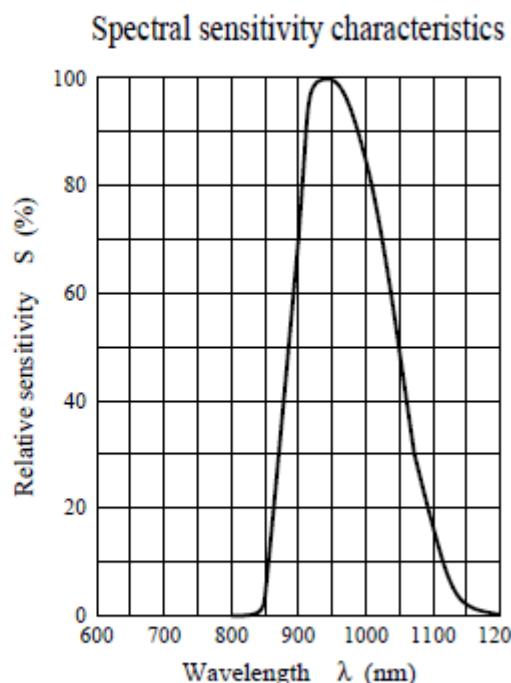
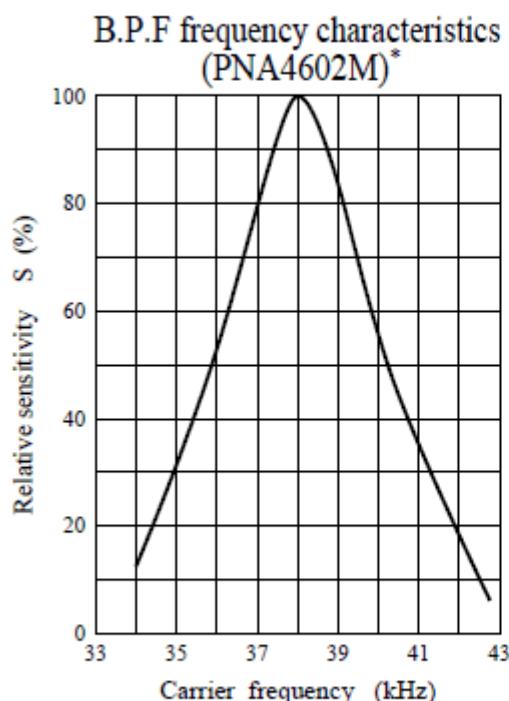
IR detectors are little microchips with a photocell that are tuned to listen to infrared light. They are almost always used for remote control detection - every TV and DVD player has one of these in the front to listen for the IR signal from the clicker. Inside the remote control is a matching IR LED, which emits IR pulses to tell the TV to turn on, off or change channels. IR light is not visible to the human eye, which means it takes a little more work to test a setup.

There are a few difference between these and say a CdS Photocells:

IR detectors are specially filtered for Infrared light, they are not good at detecting visible light. On the other hand, photocells are good at detecting yellow/green visible light, not good at IR light

- IR detectors have a demodulator inside that looks for modulated IR at 38 KHz. Just shining an IR LED won't be detected, it has to be PWM blinking at 38KHz. Photocells do not have any sort of demodulator and can detect any frequency (including DC) within the response speed of the photocell (which is about 1KHz)
- IR detectors are digital out - either they detect 38KHz IR signal and output low (0V) or they do not detect any and output high (5V). Photocells act like resistors, the resistance changes depending on how much light they are exposed to.

What You Can Measure



* The peaks for PNA4601M, PNA4608M, and PNA4610M are all f_0 .

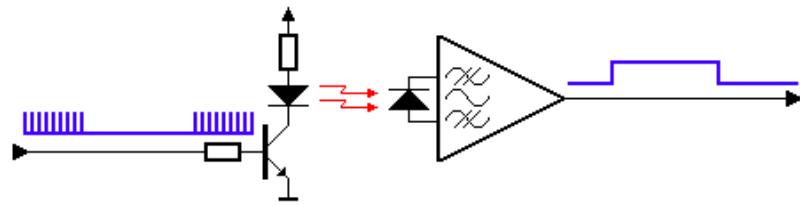
As you can see from these datasheet graphs, the peak frequency detection is at 38 KHz and the peak LED color is 940 nm. You can use from about 35 KHz to 41 KHz but the sensitivity will drop off so that it wont detect as well from afar. Likewise, you can use 850 to 1100 nm LEDs but they wont work as well as 900 to 1000nm so make sure to get matching LEDs! Check the datasheet for your IR LED to verify the wavelength.

Try to get a 940nm - remember that 940nm is not visible light (its Infra Red)!

Principle

Firstly, let's know the structure of the infrared receiving head: there are two important elements inside the infrared receiving head, IC and PD. IC is receiving head processing components, mainly composed of silicon and circuit. It is a highly integrated device. The main function is filter, plastic, decoding, amplification, etc. PD is a photosensitive diode. The main function is to receive the light signal.

Below is a brief working principle diagram:

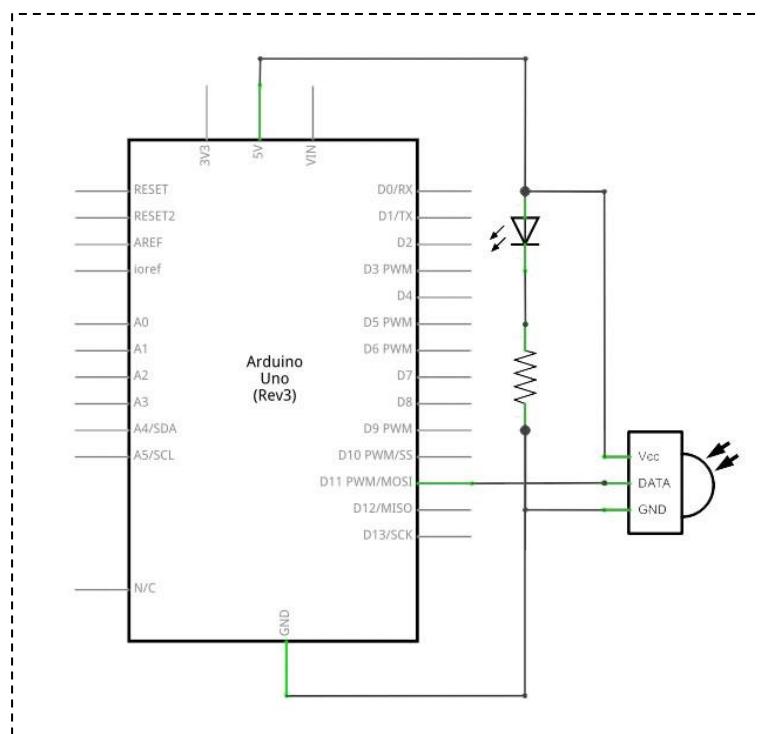


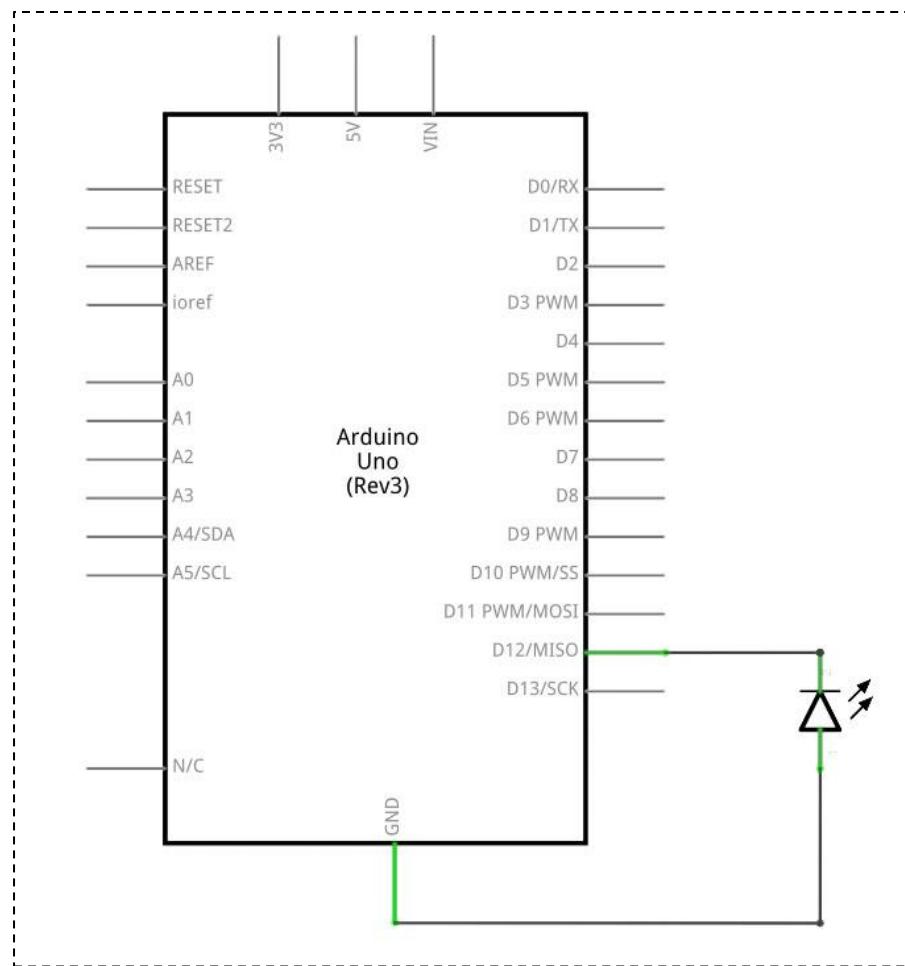
Infrared emitting diode launch out the modulation signal and infrared receiver head will receive, decode, filter and soon to regain the signal.

Infrared emitting diode: keep clean and in good condition. All the parameters in the process of working shall not exceed the limit value (positive To the current 30~60mA, positive pulse current 0.3~1A, reverse voltage 5V, dissipation power 90mW, working temperature range -25~+80 °C, and storage temperature range between 40~100 °C, the welding temperature 260 °C) infrared to be with a closed head should be matching use, otherwise it will influence the sensitivity.

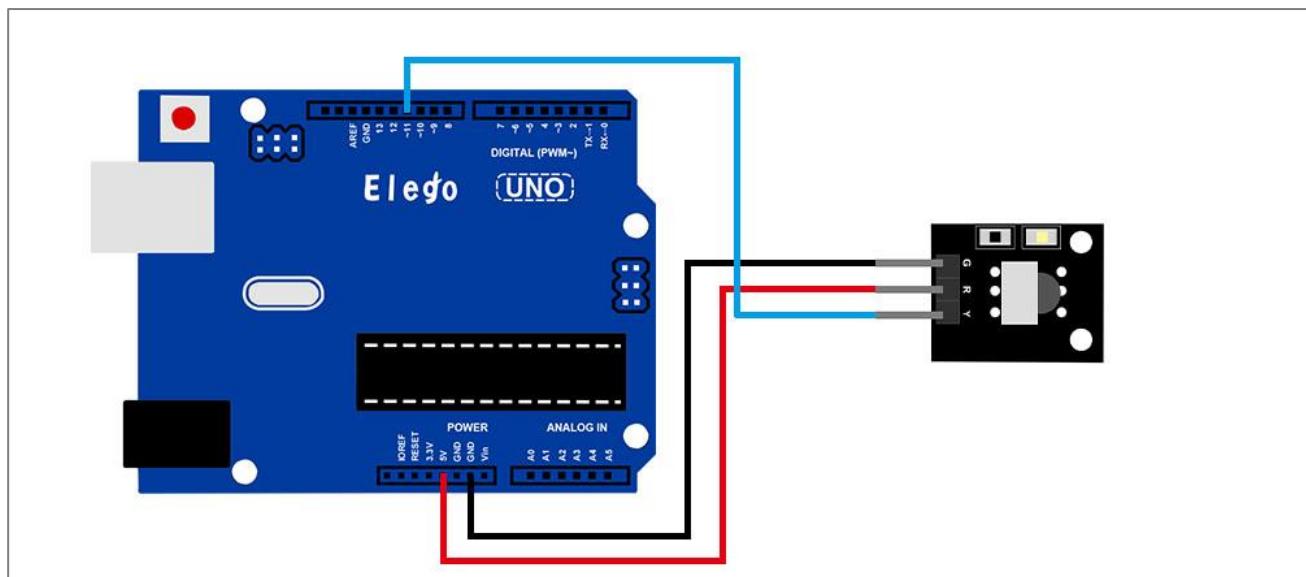
Connection

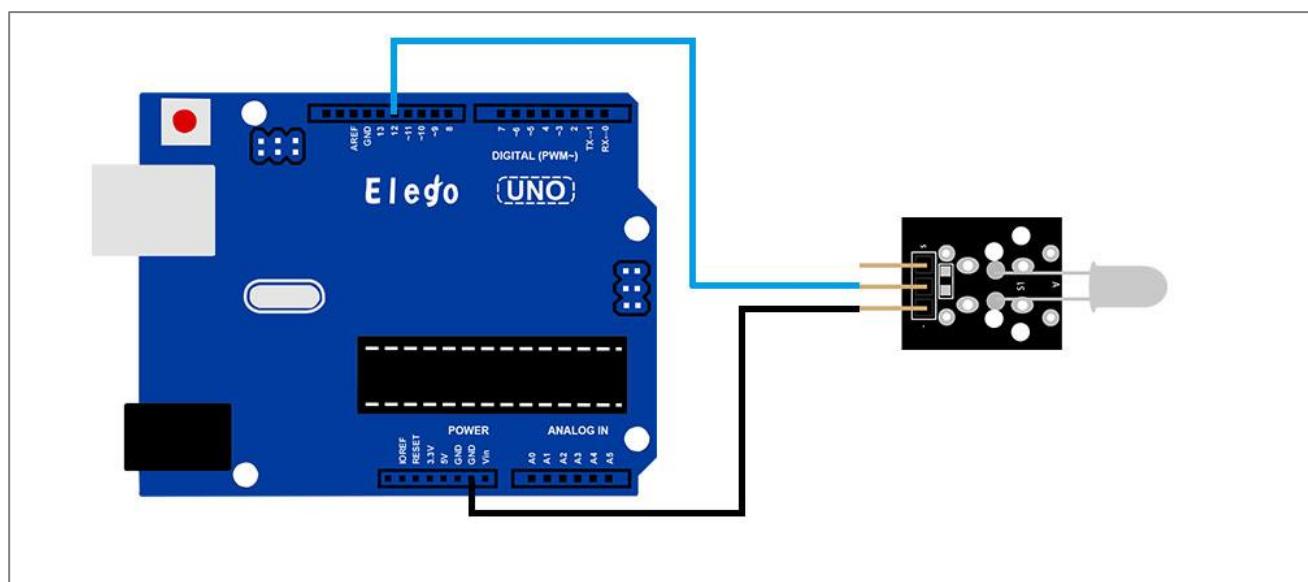
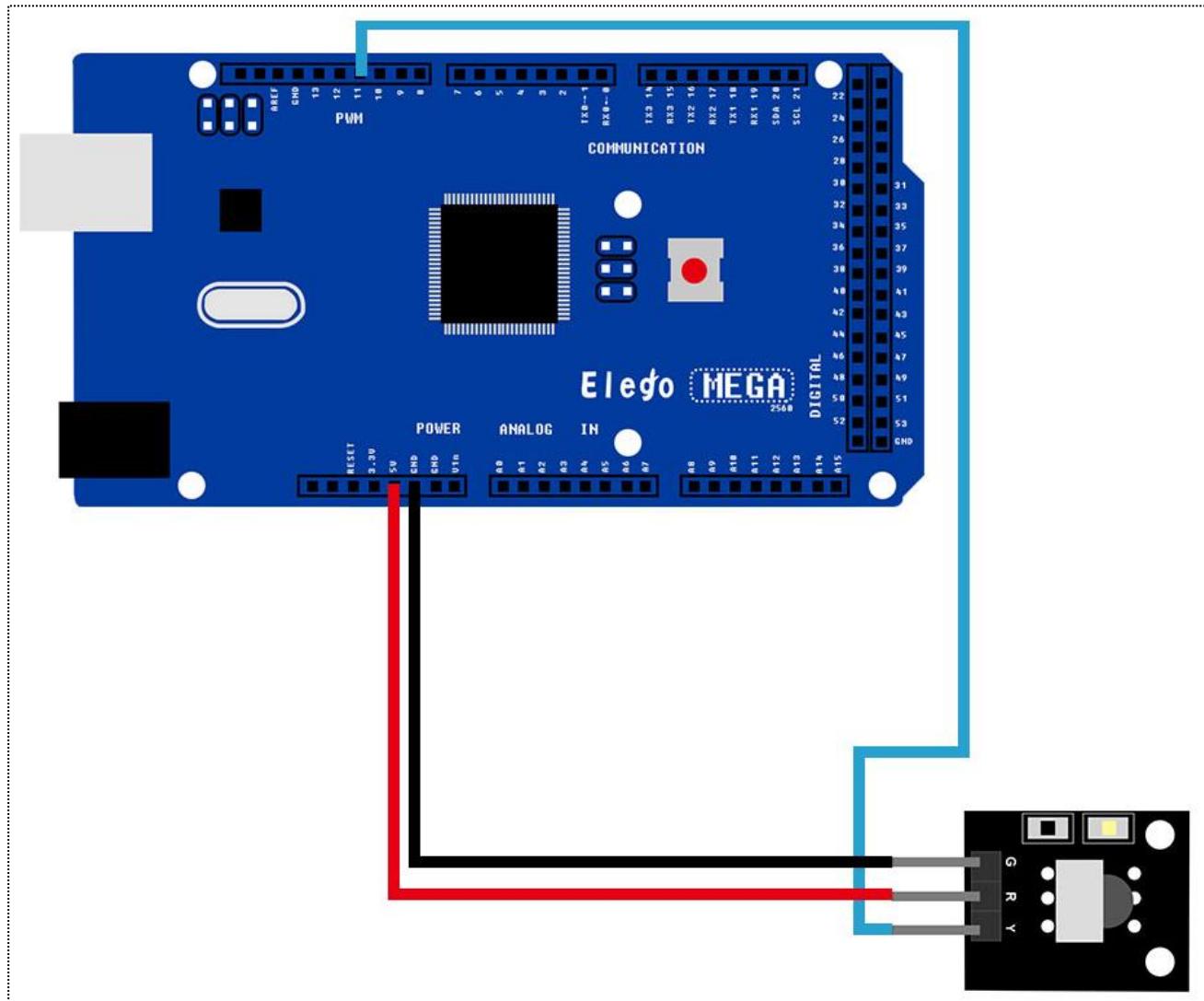
Schematic

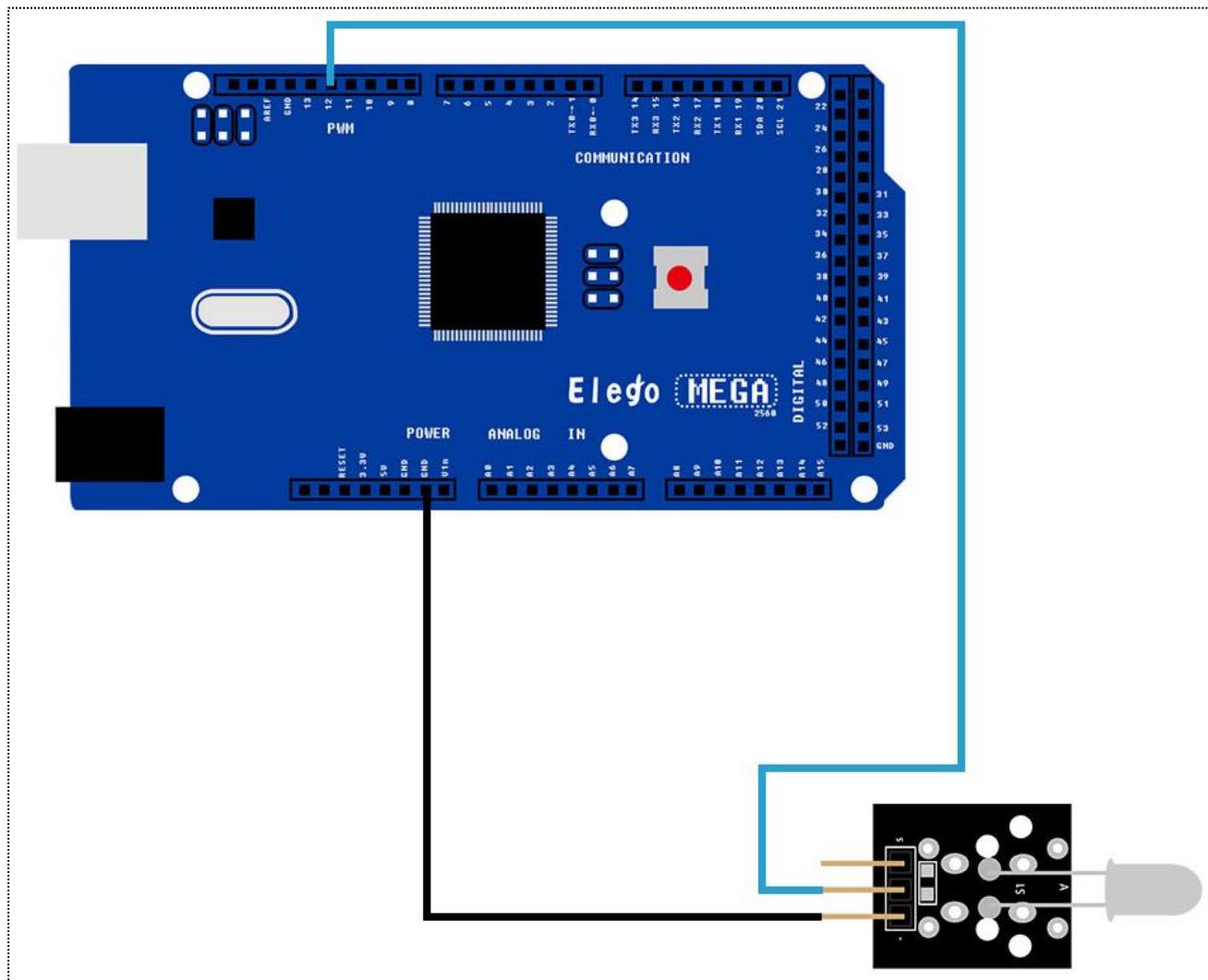




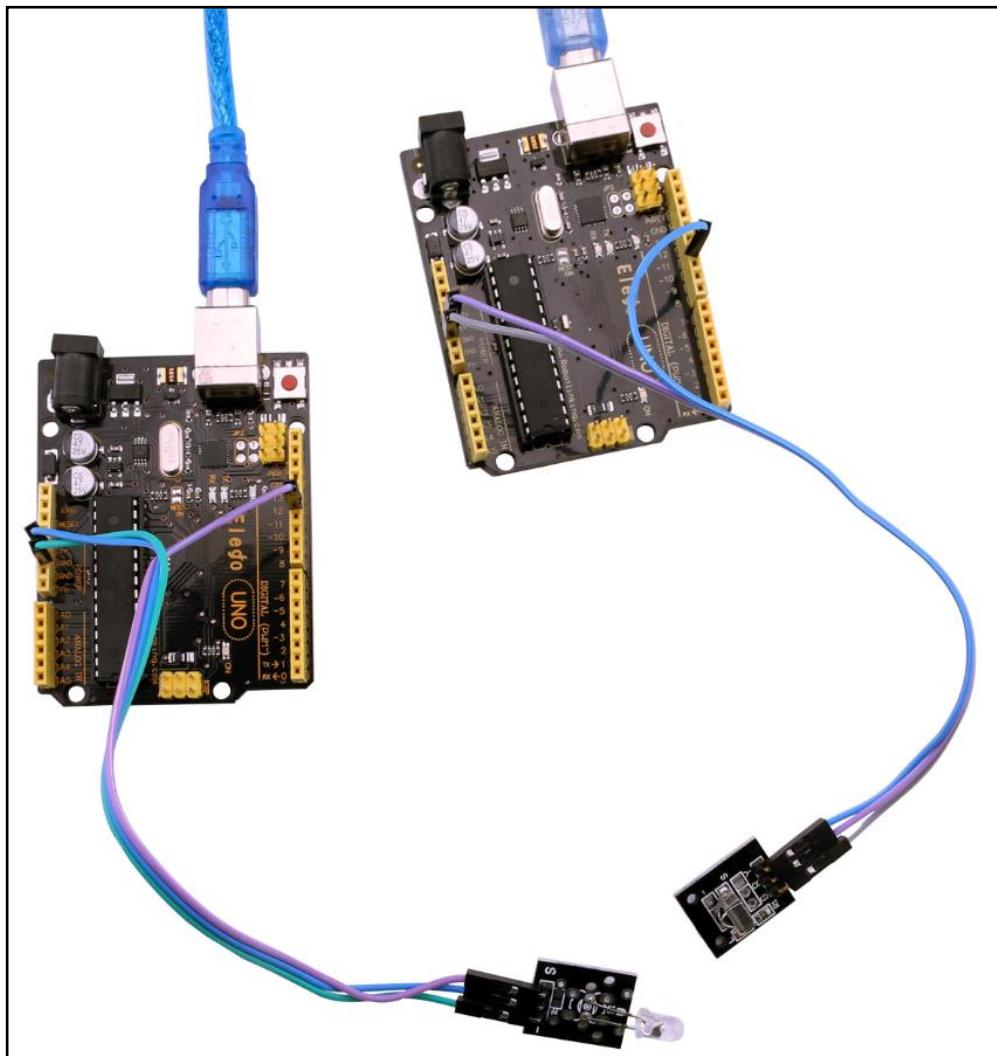
wiring diagram







Result

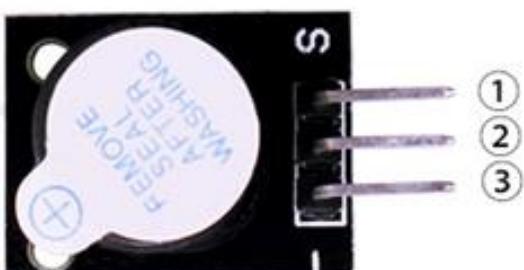


Lesson 7 ACTIVE AND PASSIVE BUZZER

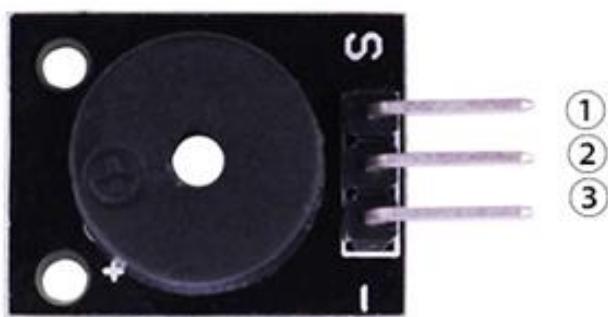
Overview

In this experiment, we will learn how to use buzzer module.

With the Arduino we can complete a lot of interactive work, commonly what we used is the light shows. And we has been use the LED small lights in the experiment before. In this experiment we will make the circuit having noise. The common components that can make sound are buzzer and speakers. Compared to the speaker, buzzer is more simple and easy to use so in this experiment we adopts the buzzer.



- 1.OUTPUT
- 2.VCC: 3.3V-5V DC
- 3.GND:ground



- 1.OUTPUT
- 2.VCC: 3.3V-5V DC
- 3.GND:ground

Component Required:

(1) x Elegoo Uno R3

- (1) x USB cable
- (1) x Active buzzer module
- (1) x Passive buzzer module
- (x) x F-M wires

Component Introduction

Active Buzzer:

As a type of electronic buzzer with integrated structure, buzzers, which are supplied by DC power, are widely used in computers, printers, photocopiers, alarms, electronic toys, automotive electronic devices, telephones, timers and other electronic products for voice devices. Buzzers can be categorized as active and passive ones (see the following picture). Turn the pins of two buzzers face up, and the one with a green circuit board is a passive buzzer, while the other enclosed with a black tape is an active one.

The difference between an active buzzer and a passive buzzer is:

An active buzzer has a built-in oscillating source, so it will make sounds when electrified. But a passive buzzer does not have such source, so it will not tweet if DC signals are used; instead, you need to use square waves whose frequency is between 2K and 5K to drive it. The active buzzer is often more expensive than the passive one because of multiple built-in oscillating circuits.

Passive Buzzer:

Passive buzzer, in fact, just use PWM generating audio, drives the buzzer, allowing the air to vibrate, can sound. Appropriately changed as long as the vibration frequency, it can generate different sound scale. For example, sending a pulse wave can be generated 523Hz Alto Do, pulse 587Hz can produce midrange Re, 659Hz can produce midrange Mi. If you then with a different beat, you can play a song. Here be careful not to use the Arduino analog Write () function to generate a pulse wave, because the frequency analog Write () is fixed (500Hz), no way to scale the output of different sounds.

Distinguish between Active and Passive Buzzer

Teach you to distinguish between active and passive buzzer

Now a small buzzer on the market because of its small size (diameter of only 11 mm), light weight, low price, solid structure, and is widely used in a variety of needs audible electrical

equipment, electronic production and microcontroller circuits, etc. .

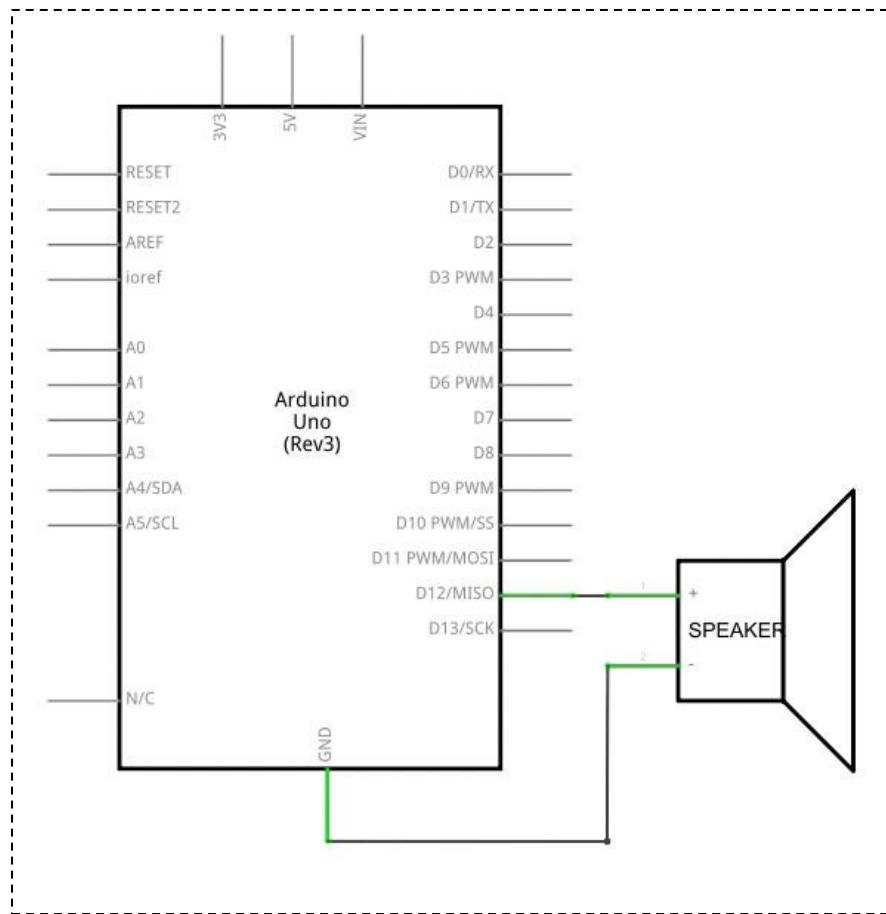
From the exterior, two kinds of buzzer seems the same, but a closer look, a slight difference between the height of the active buzzer a, height of 9mm, and passive buzzer b height of 8mm. When the buzzer as the two pins are facing up, you can see there is a green circuit board is passive buzzer, no closed circuit boards with a vinyl is active buzzer .

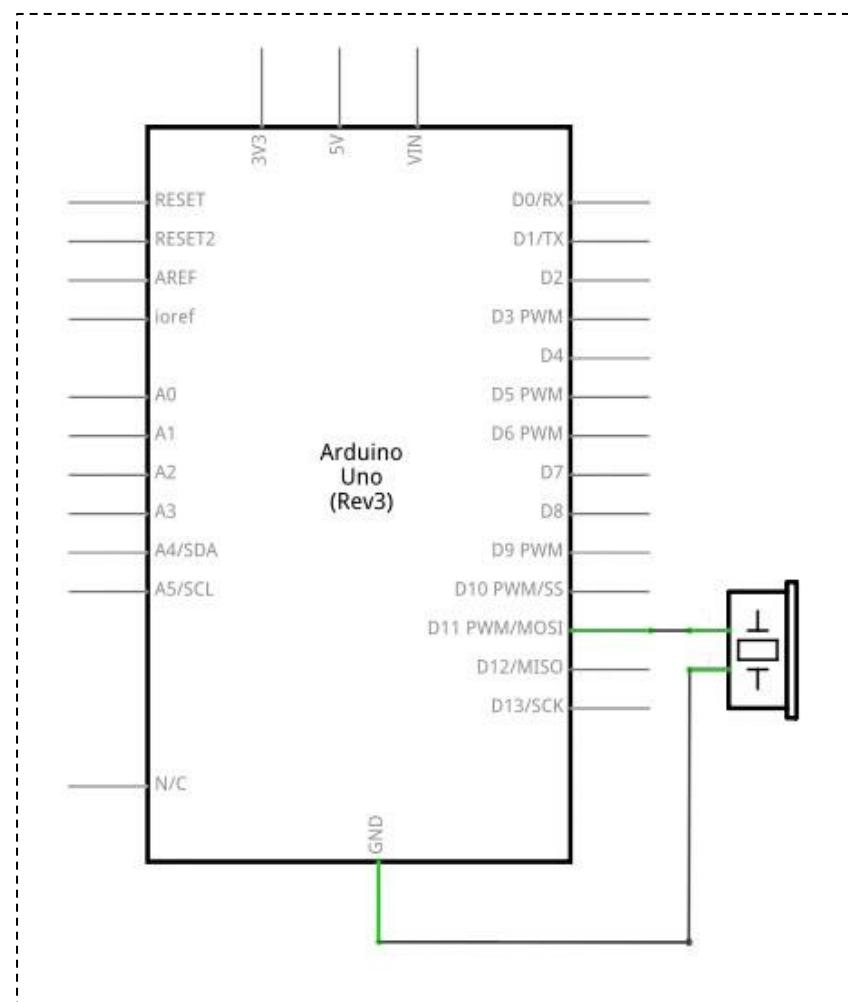
Further determine the active and passive buzzer, you can also use a multimeter to test the resistance profile Rx file: buzzer with black pen then "+" pin, the red pen to touch on another pin back and forth, if trigger a cracking, cracking sound and the resistance only 8Ω (or 16Ω) is passive buzzer; If you can emit continuous sound, and the resistance in Europe and more than a few hundred, and is an active buzzer.

Active buzzer directly connected to the rated power (new buzzer has stated on the label) can be a continuous sound; rather passive electromagnetic buzzer and speaker are the same, you need to take in order to sound the audio output circuit .

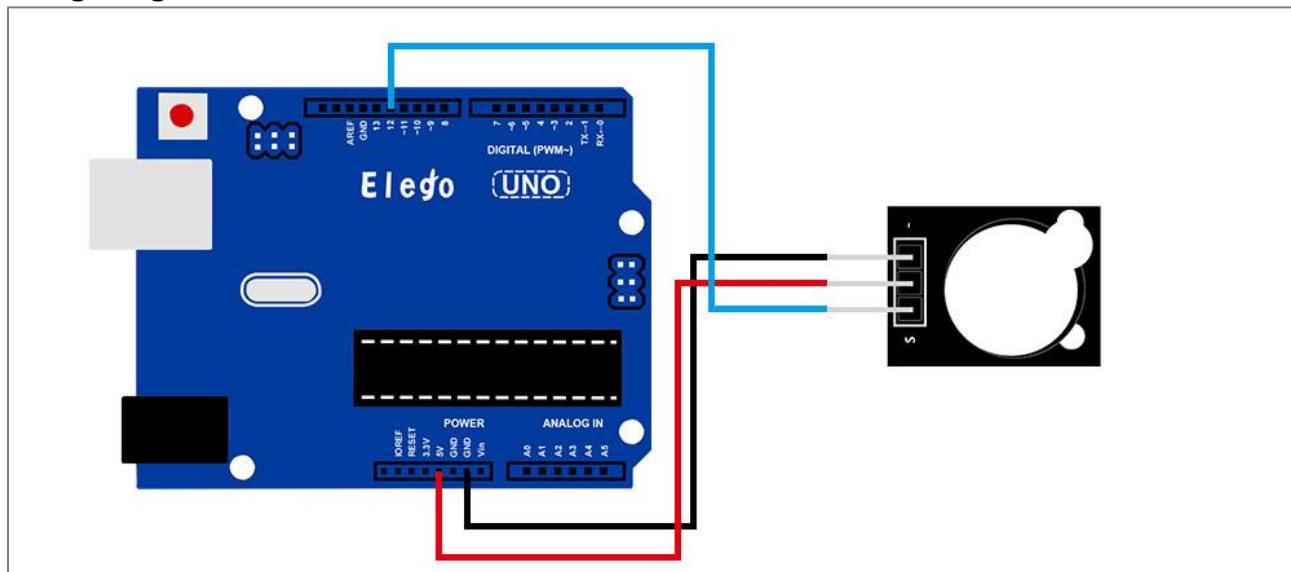
Connection

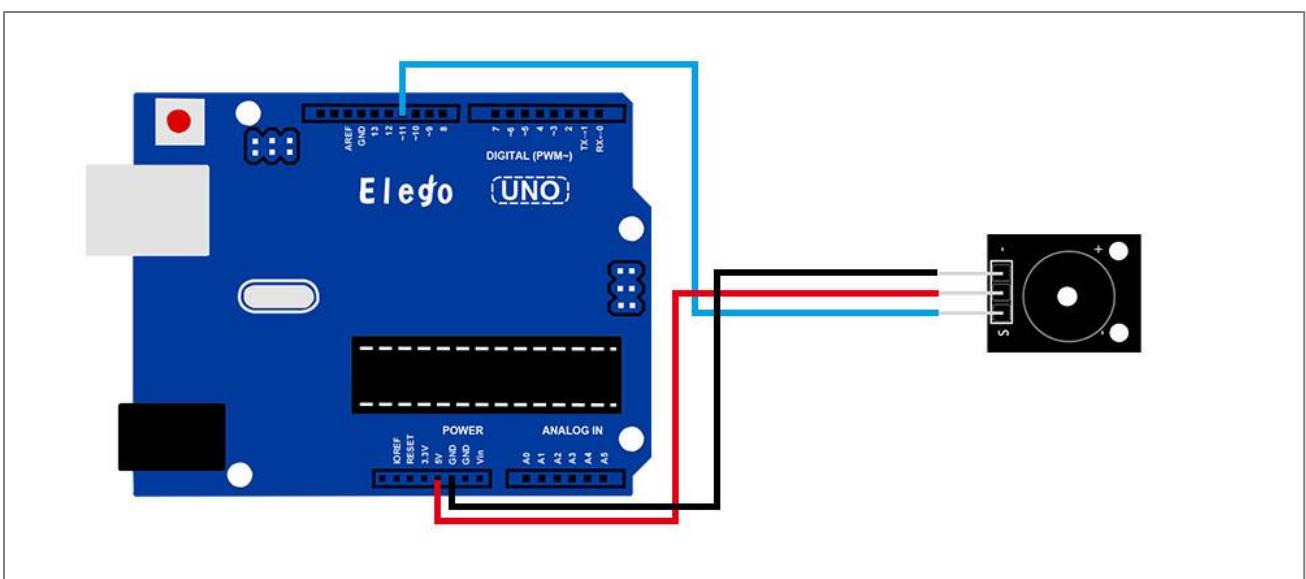
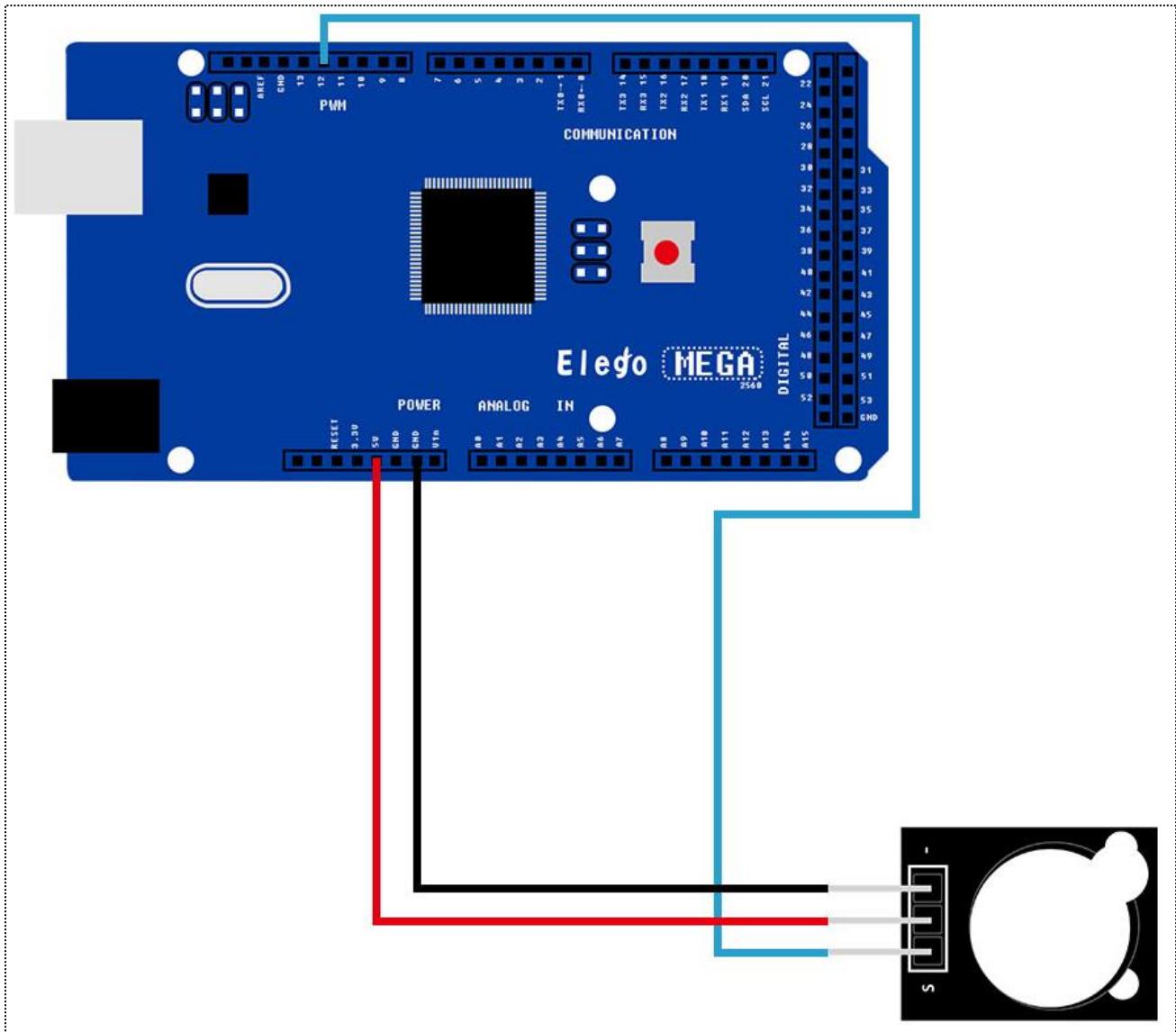
Schematic

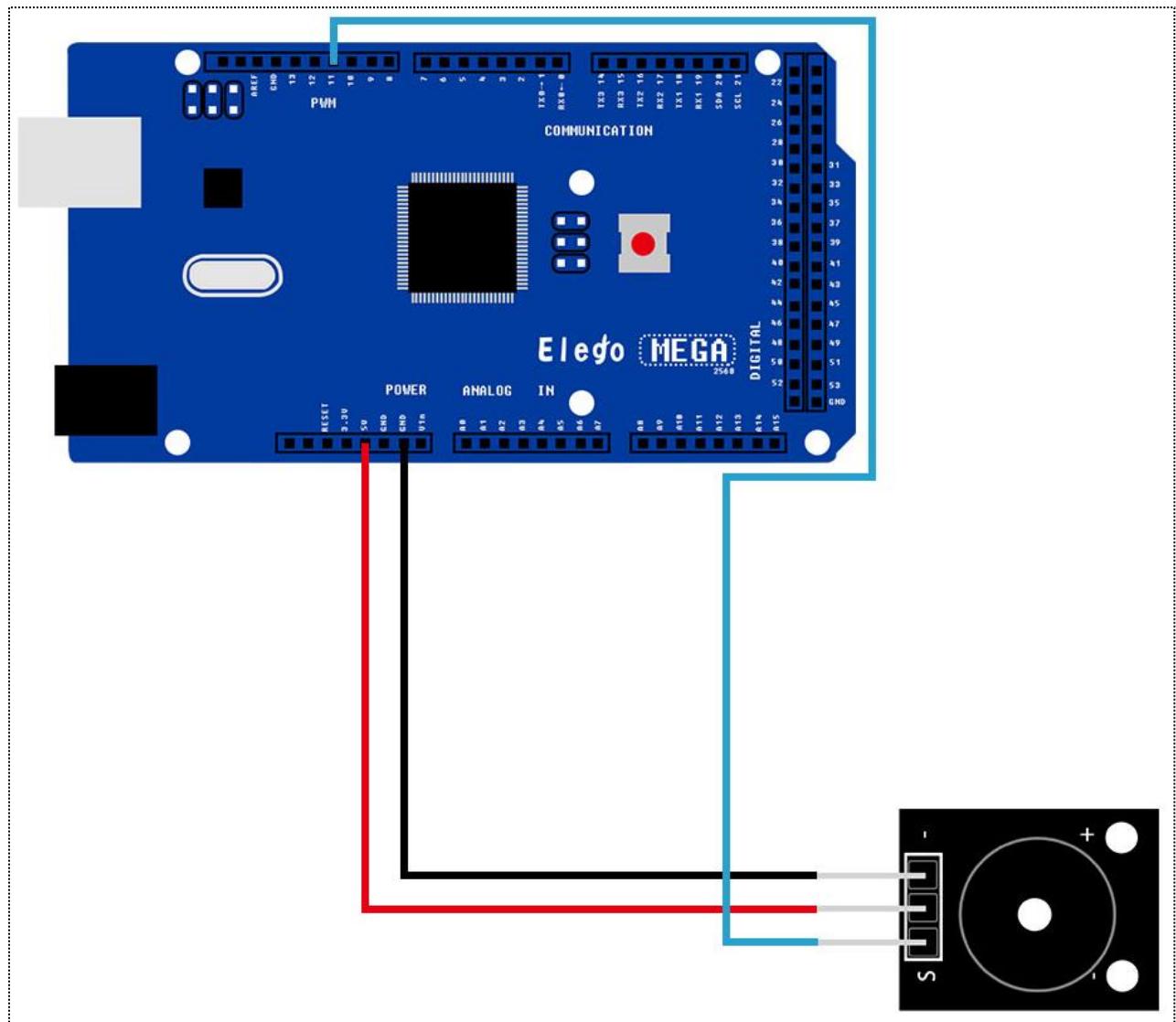




wiring diagram

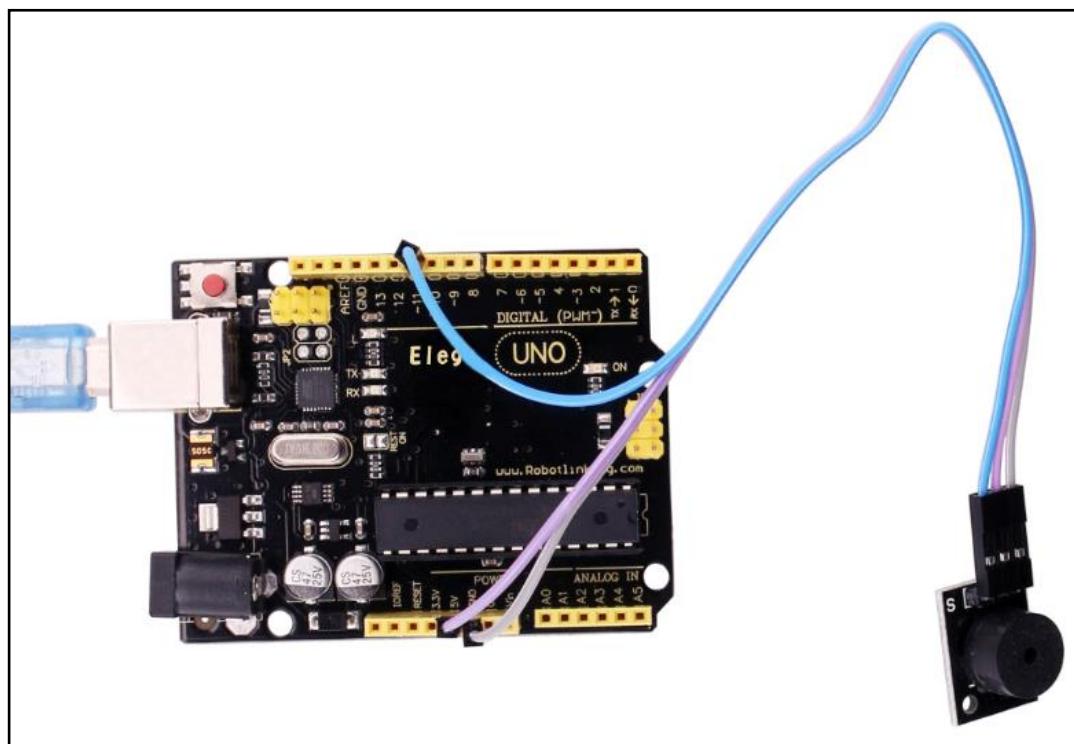
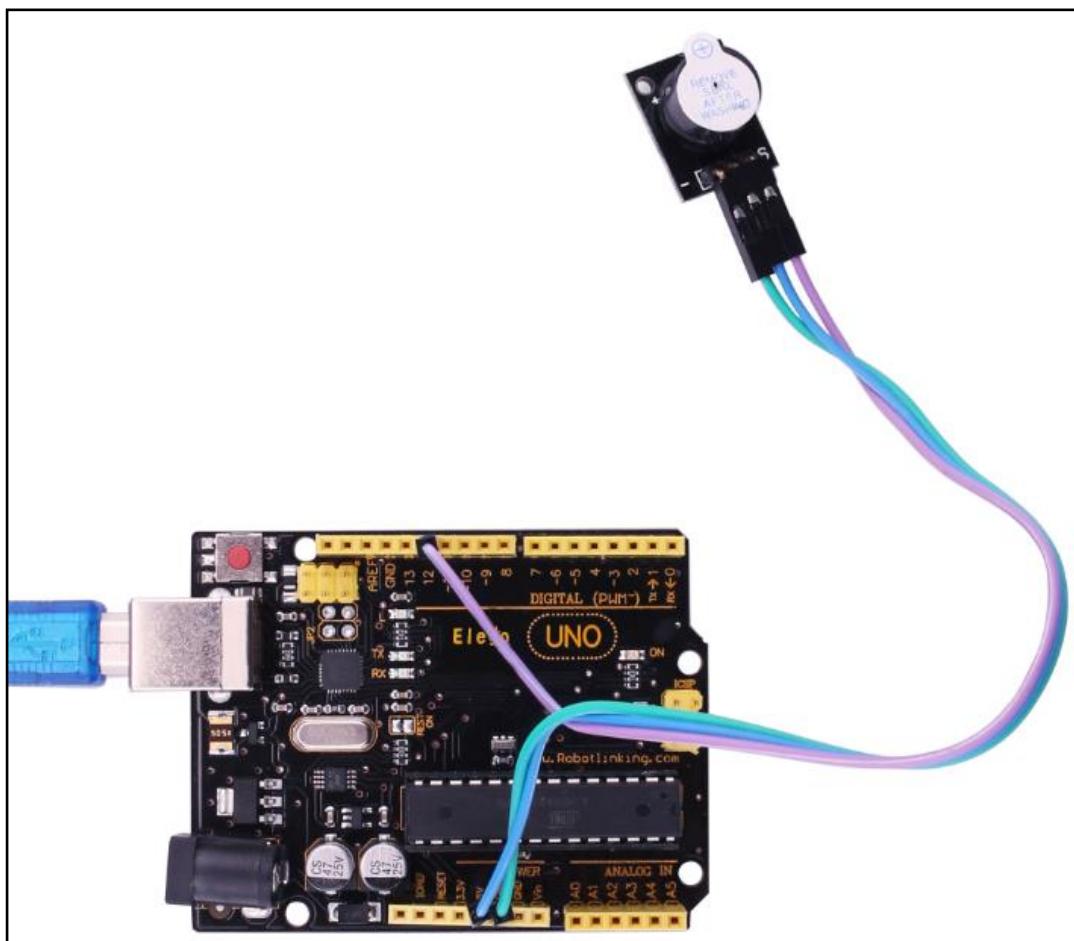






Result

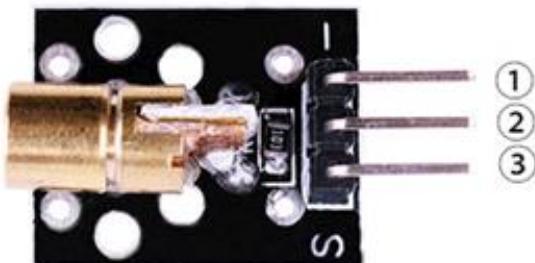
After we connect the circuit as the picture, we upload the program of each module. We can hear that the active buzzer can make sound in one voice. And the passive can sing a song.



Lesson 8 LASER MODULE

Overview

In this experiment, we will learn how to use laser module.



- 1.GND:ground
- 2.VCC:3.3V-5V DC
- 3.OUTPUT

Component Required:

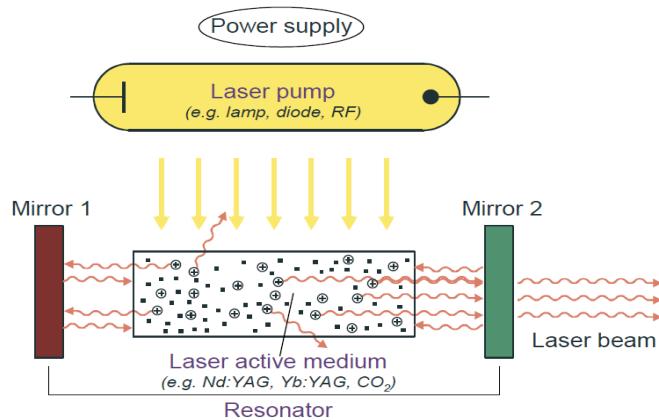
- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Laser module
- (x) x F-M wires

Component Introduction

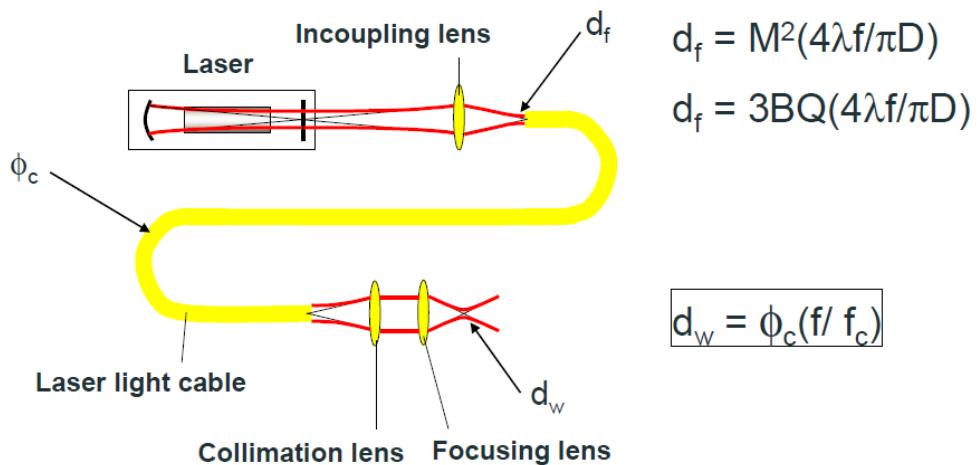
Laser sensor:



Laser basics

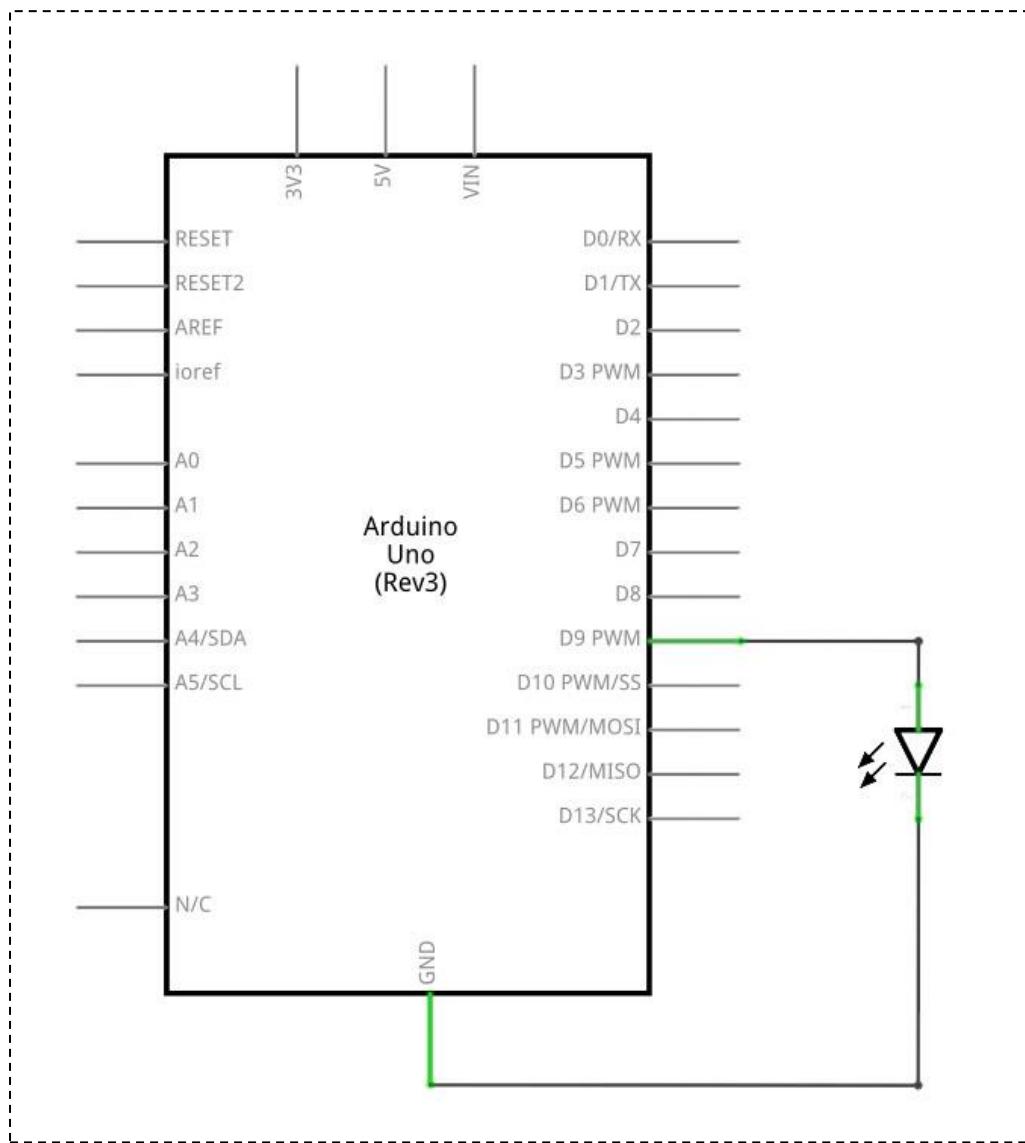


Spot size - YAG

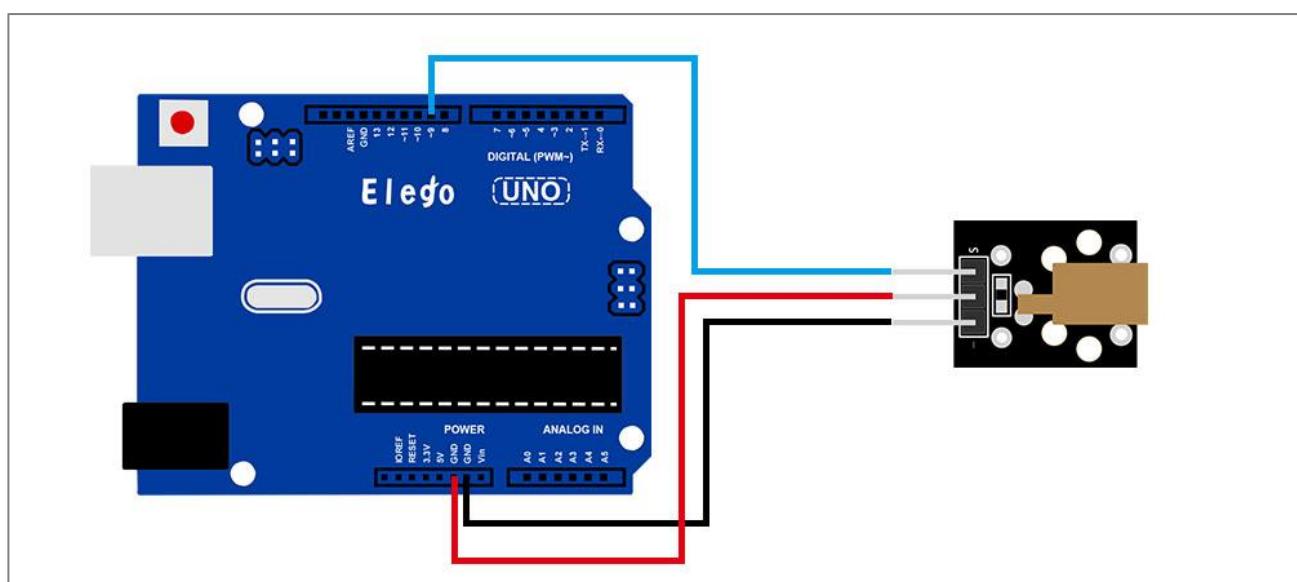


Connection

Schematic

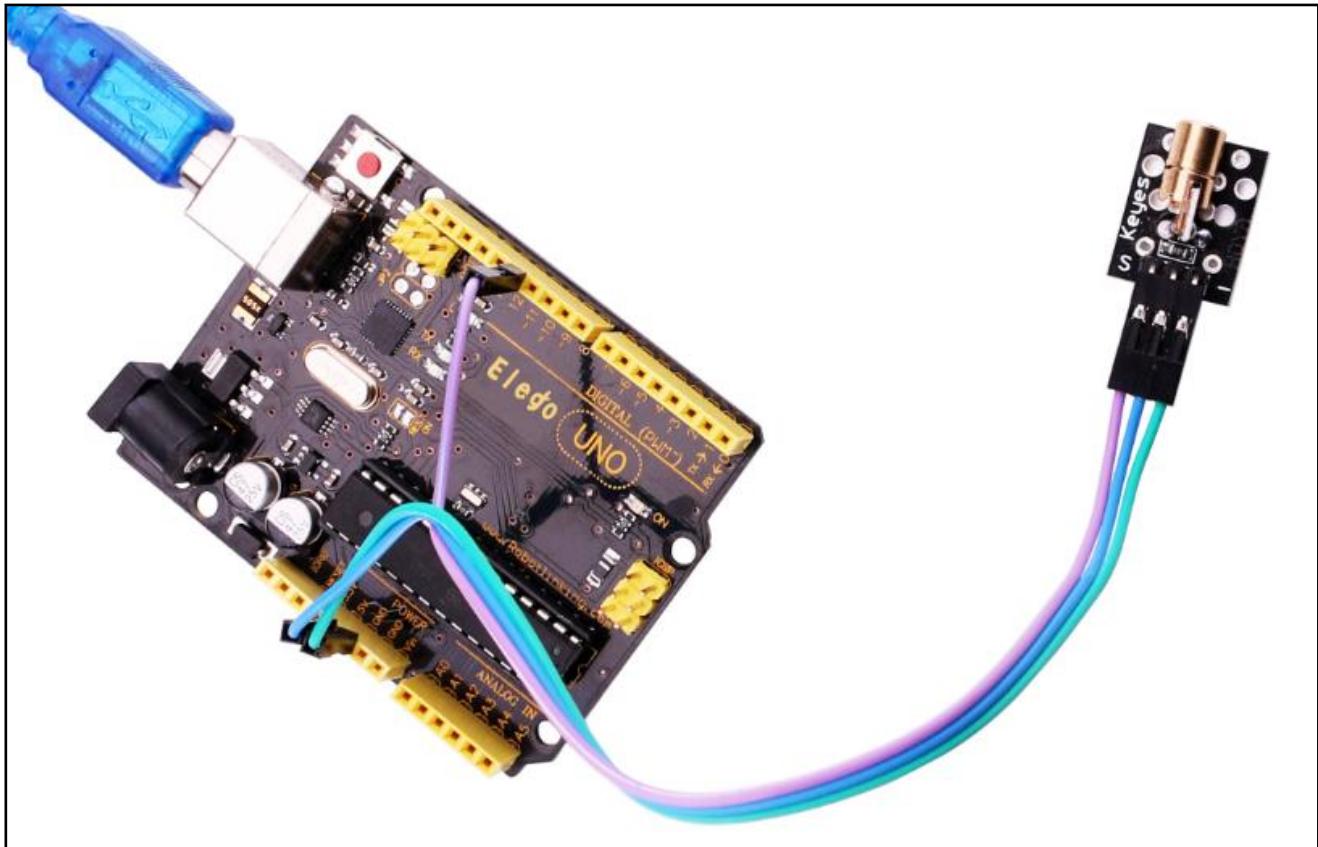


wiring diagram



Result

After we connect the circuit as the picture, we upload the program. We can see the module can emission lasers.



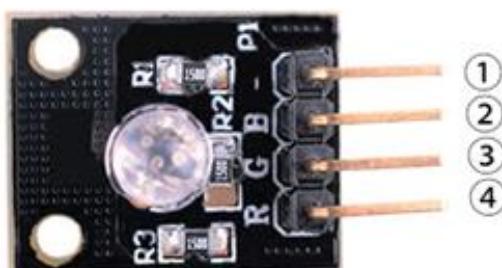
Lesson 9 SMD RGB MODULE AND RGB MODULE

Overview

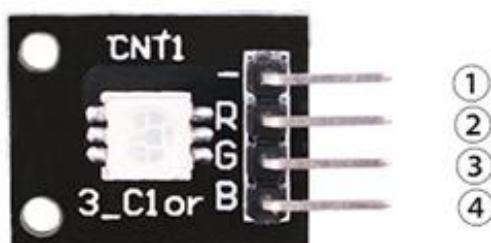
In this experiment, we will learn how to use SMD RGB module and RGB module.

Actually, the function of SMD RGB module and RGB module are almost the same. But we can choose the shape we like or we need.

SMD RGB LED module and RGB module are made from a patch of full-color LED. By adjusting the voltage input of R, G, B pins, we can adjust the strength of the three primary colors (red/blue/green) so as to implementation result of full color effect.



- 1.GND:ground
- 2.BLUE
- 3.GREED
- 4.RED



- 1.GND:ground
- 2.RED
- 3.GREED
- 4.BLUE

Component Required:

(1) x Elegoo Uno R3

(1) x USB cable

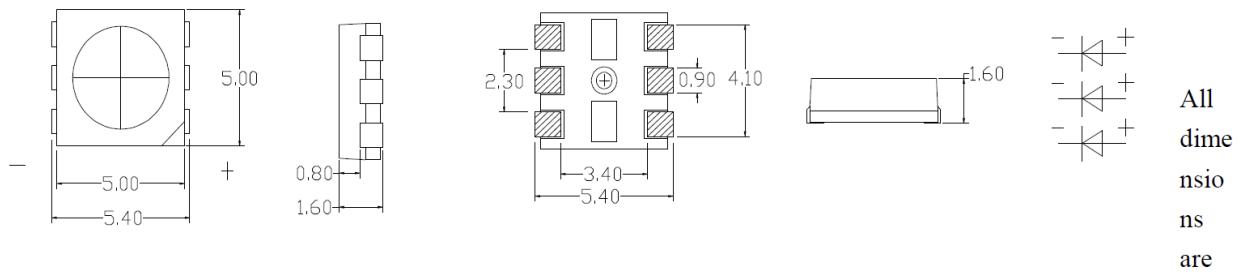
(1) x SMD RGB module

(1) x RGB module

(x) x F-M wires

Component Introduction

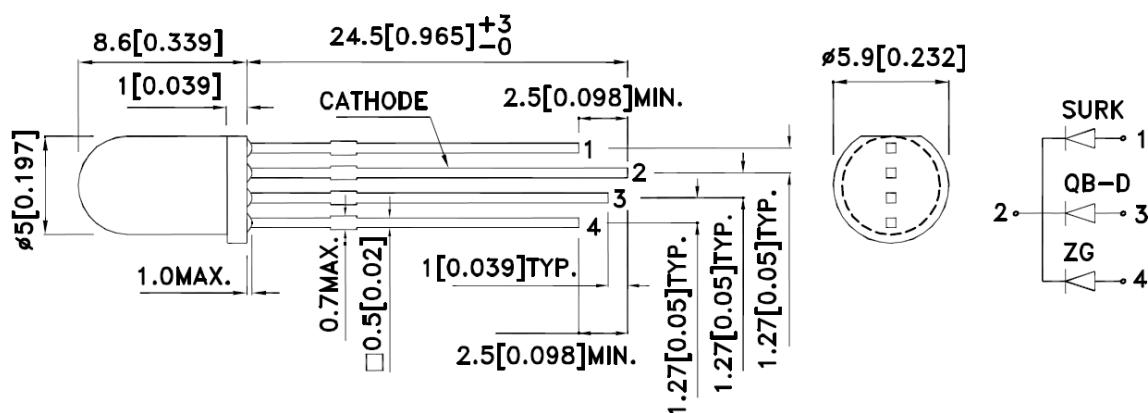
SMD RGB:



Tolerance is $\pm 0.25\text{mm}(0.10")$ unless otherwise noted

Parameter	Symbol	Value	Unit
Forward Current	I _f	20	mA
Reverse Voltage	V _r	5	V
Operating Temperature	T _{opr}	-25~+85	°C
Storage Temperature	T _{stg}	-35~+85	°C
Soldering temperature	T _{sol}	260±5°C (for 4sec)	°C
Power Dissipation	P _d	R=40 C/R=60	mW
Pulse Current	I _{FP}	100	mA

RGB:



Electrical / Optical Characteristics at TA=25°C

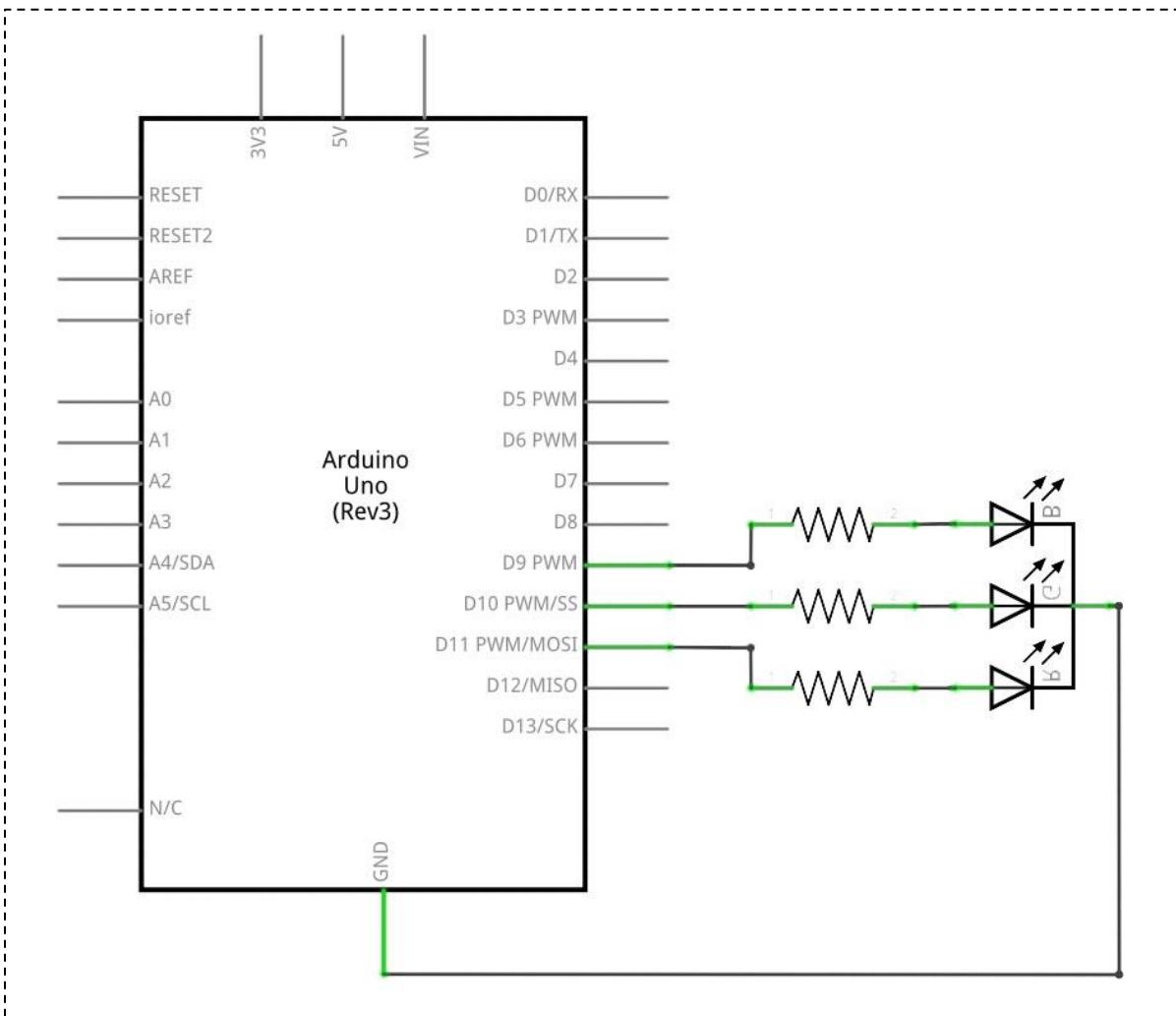
Symbol	Parameter	Device	Typ.	Max.	Units	Test Conditions
λ_{peak}	Peak Wavelength	Hyper Red Blue Green	650 468 515		nm	$I_F=20mA$
λ_D [1]	Dominant Wavelength	Hyper Red Blue Green	630 470 525		nm	$I_F=20mA$
$\Delta\lambda_{1/2}$	Spectral Line Half-width	Hyper Red Blue Green	28 25 30		nm	$I_F=20mA$
C	Capacitance	Hyper Red Blue Green	35 100 45		pF	$V_F=0V; f=1MHz$
V_F [2]	Forward Voltage	Hyper Red Blue Green	1.95 3.3 3.3	2.5 4 4.1	V	$I_F=20mA$
I_R	Reverse Current	Hyper Red Blue Green		10 50 50	uA	$V_R=5V$

Notes:

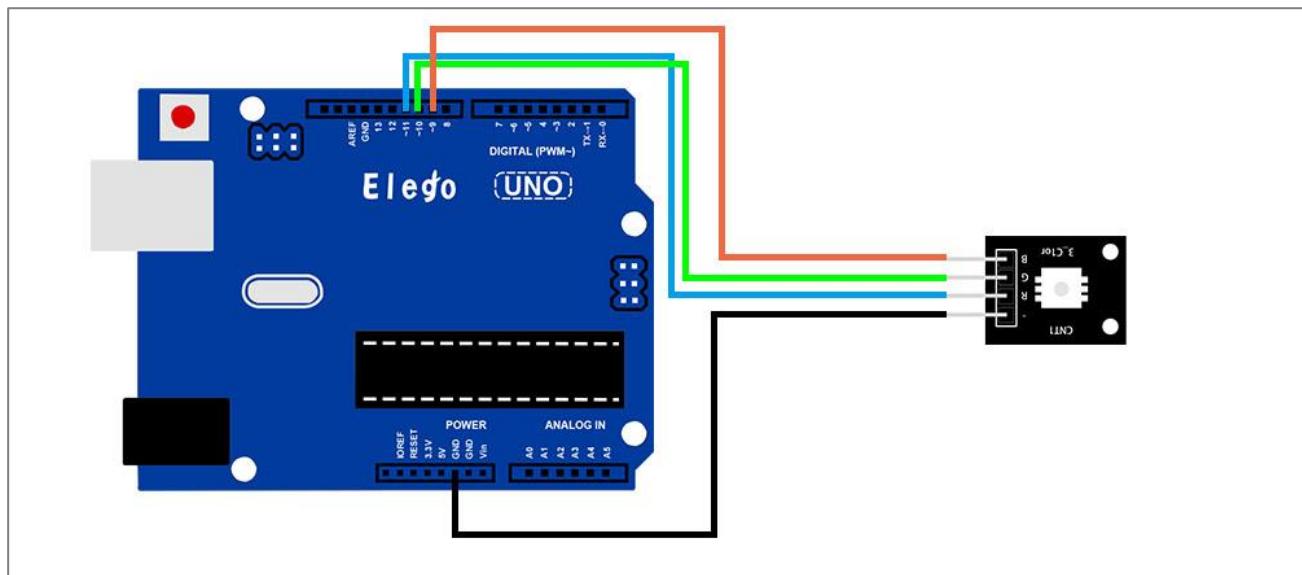
1. Wavelength: +/-1nm.
2. Forward Voltage: +/-0.1V.

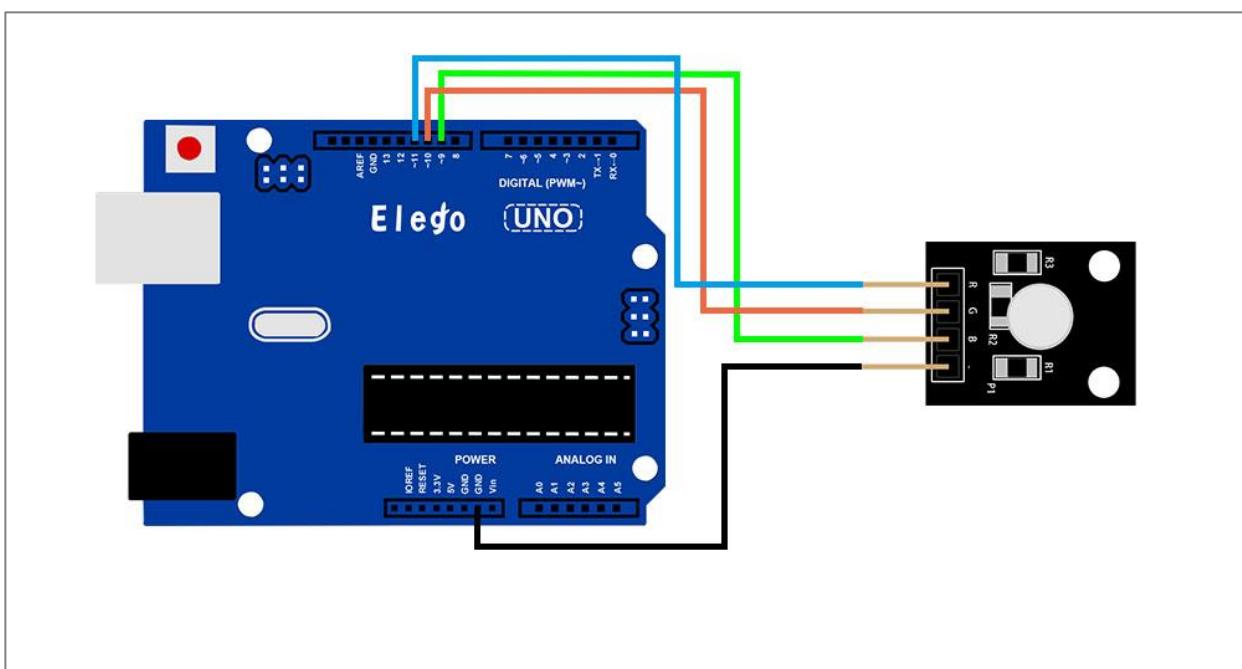
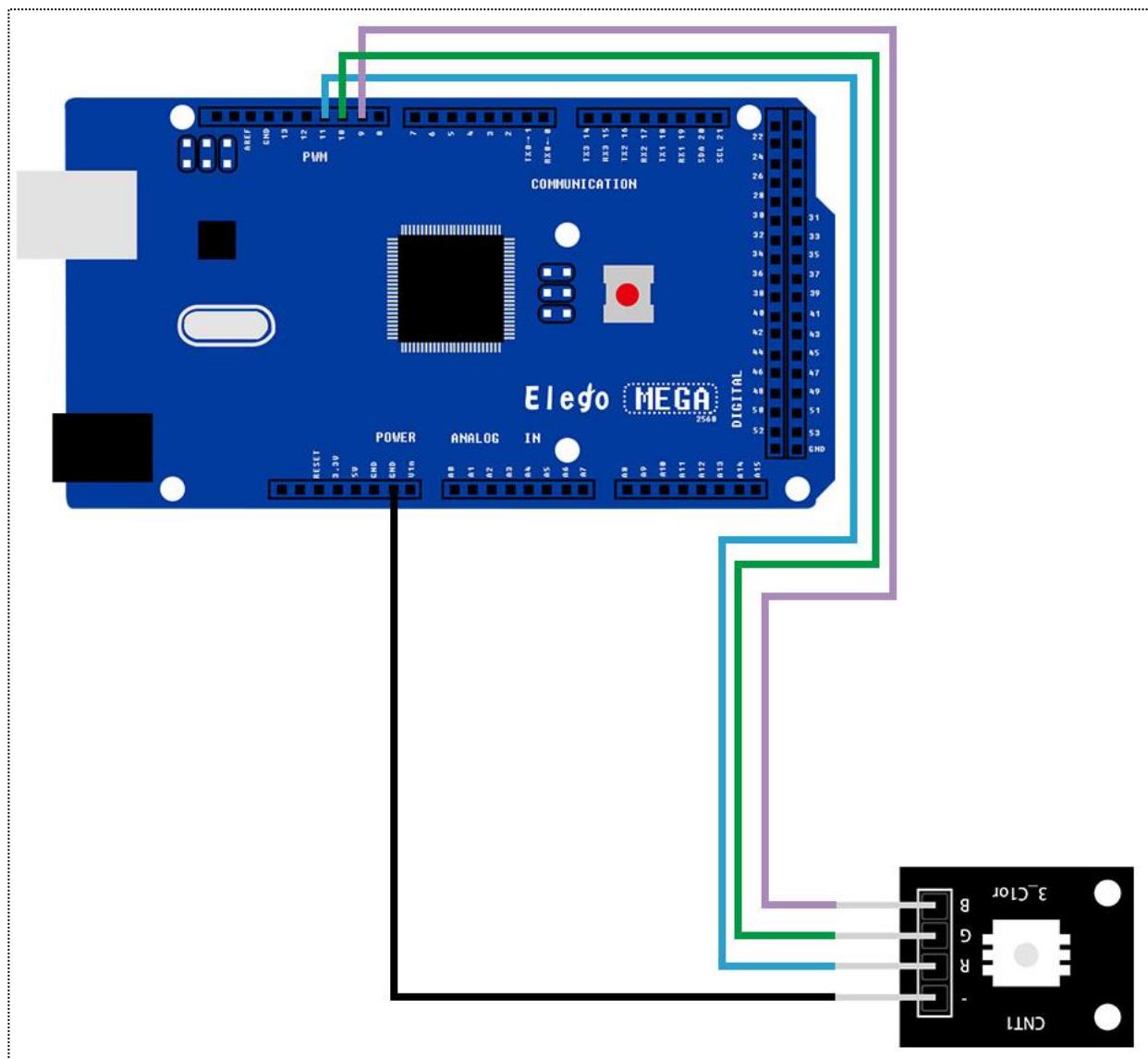
Connection

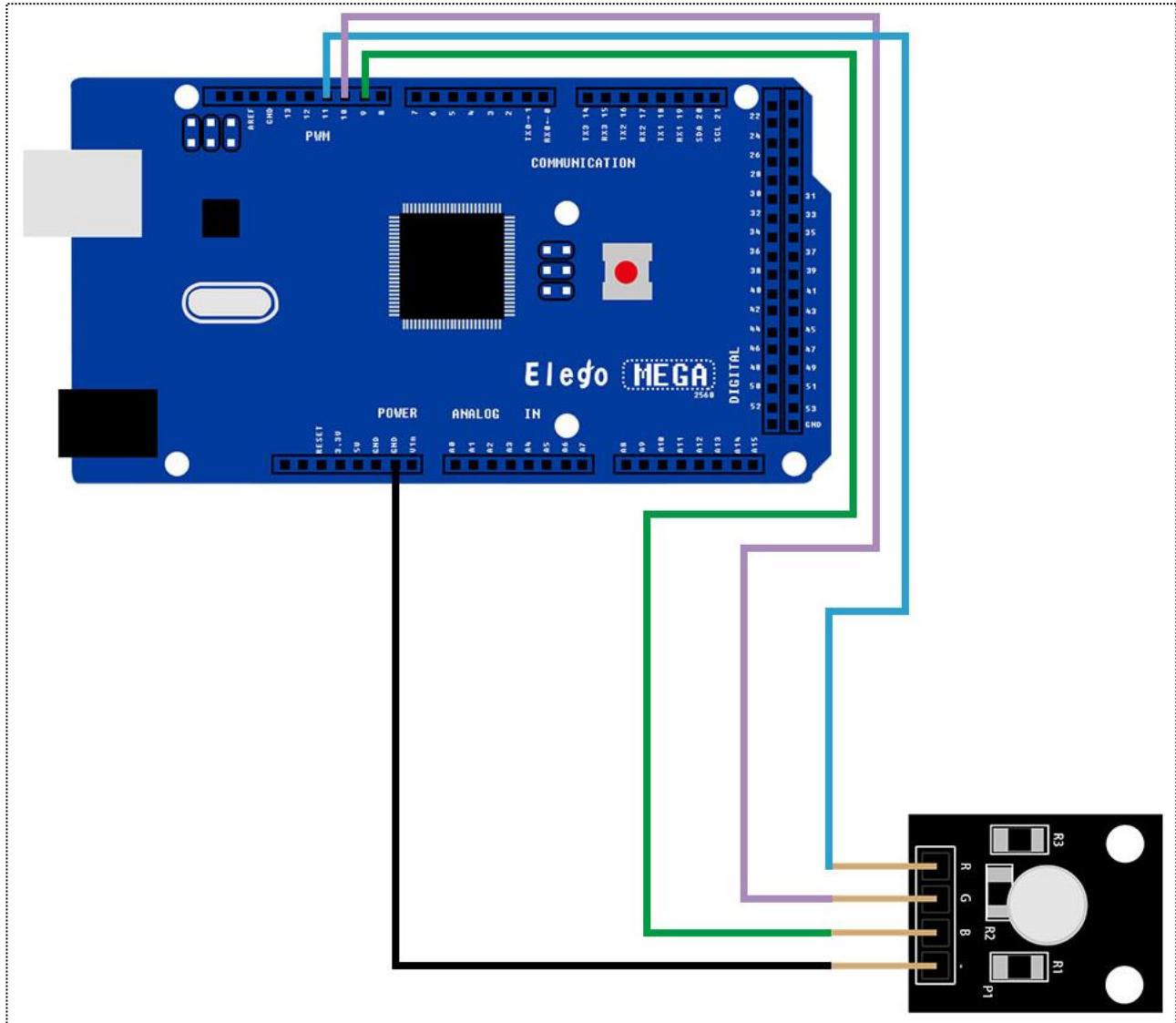
Schematic



wiring diagram

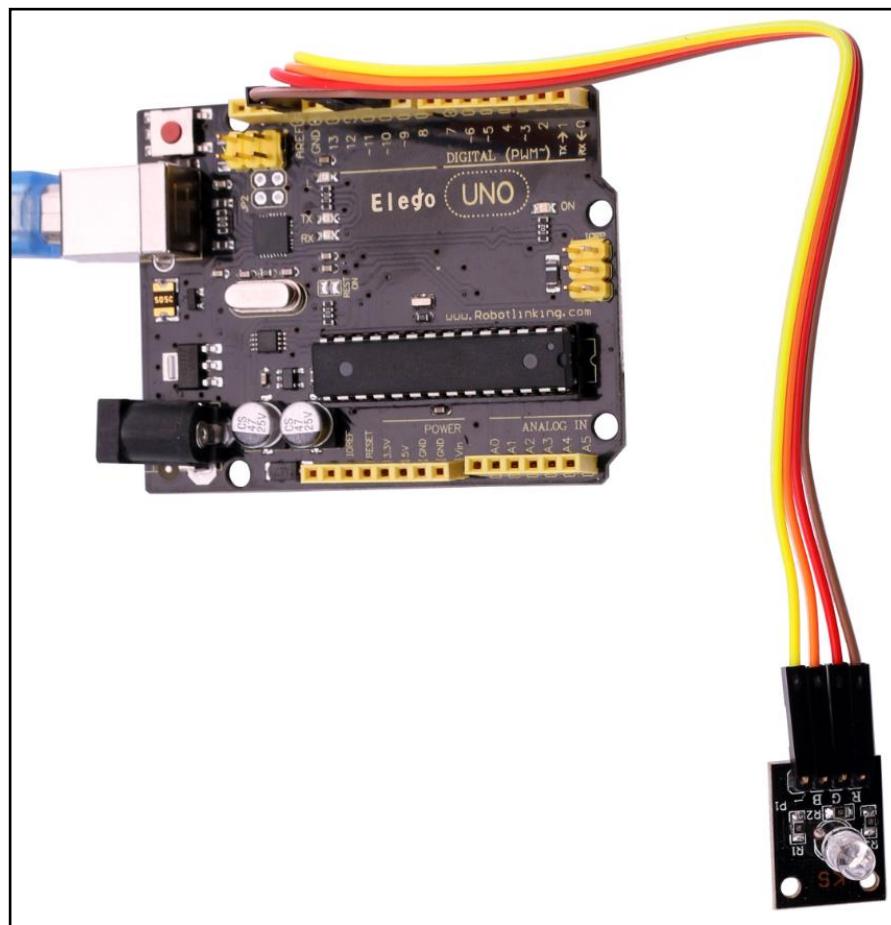
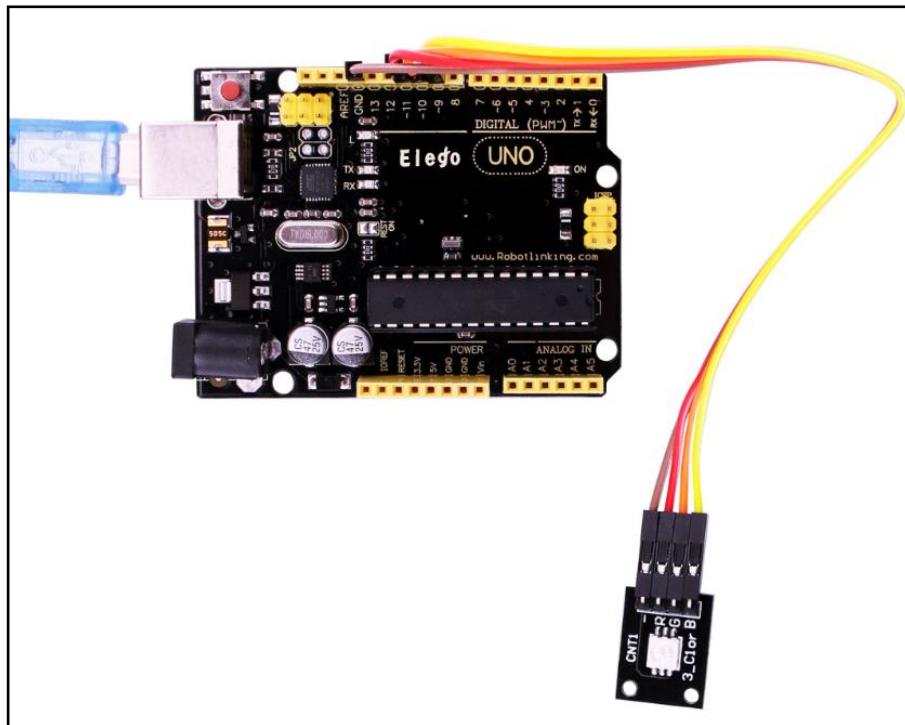






Result

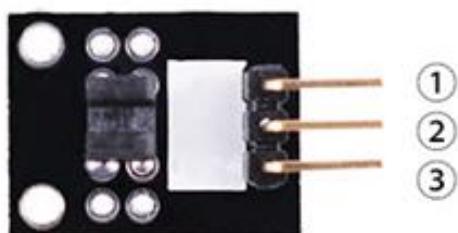
After we connect the circuit as the picture, we upload the program of each module. We can see the module changing their color as the code set. If you want to make it change the color in different way, you can revise the code.



Lesson 10 PHOTO-INTERRUPTER MODULE

Overview

In this experiment, we will learn how to use Photo-interrupter module.



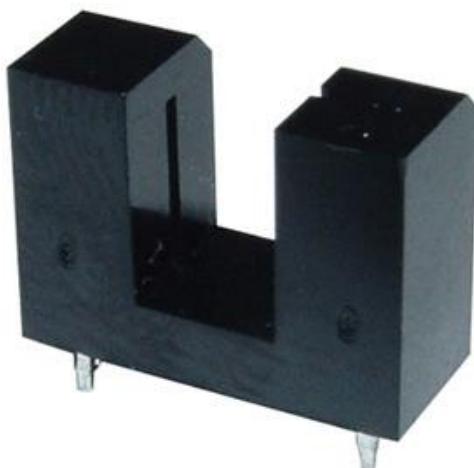
- 1.OUTPUT
- 2.VCC:3.3V-5V DC
- 3.GND:ground

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Photo-interrupter MODULE
- (x) x F-M wires

Component Introduction

Opto Interrupter Sensor:



Opto Interrupters are commonly used in many arcade games e.g. steering assembly in

older driving games, scoring switches in Whack A Crock etc. Uninterrupted light beam will turn the phototransistor "ON" connecting the ground to the game board input. When the light beam is interrupted the phototransistor turns "OFF", the ground is disconnected from the input and the pull-up resistor R1 forces the input to go "HIGH" (5V level).

Principle

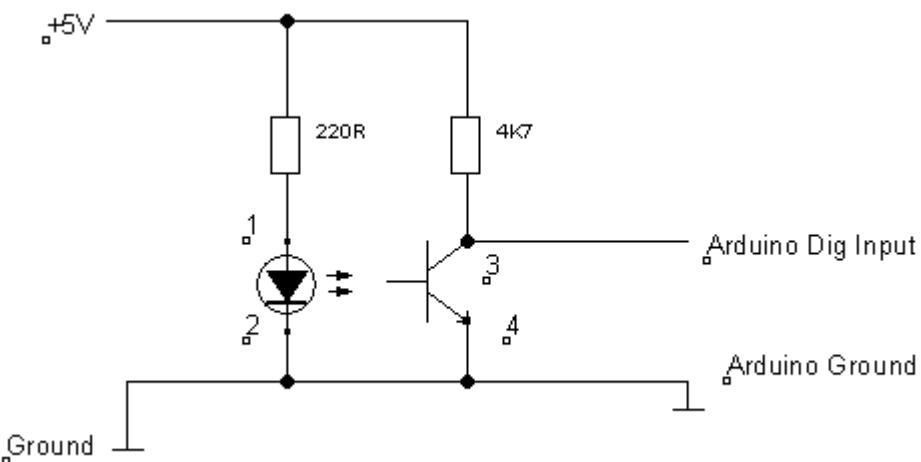
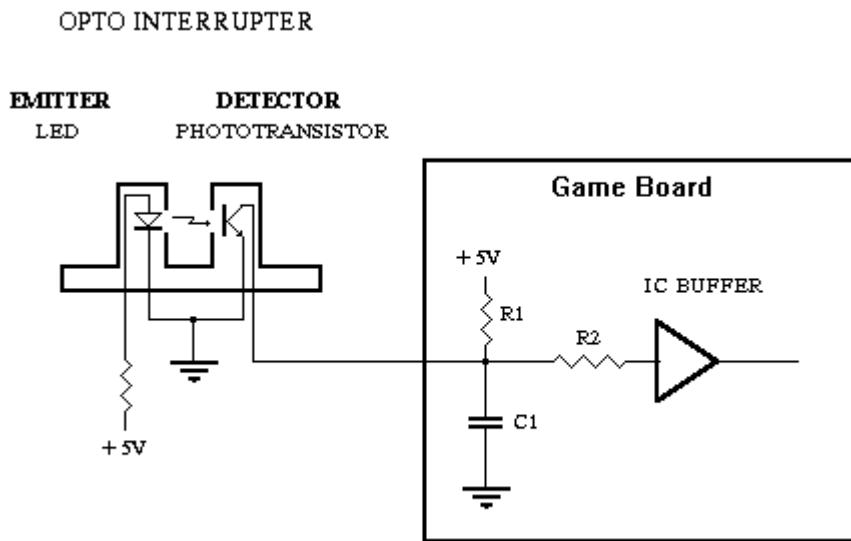
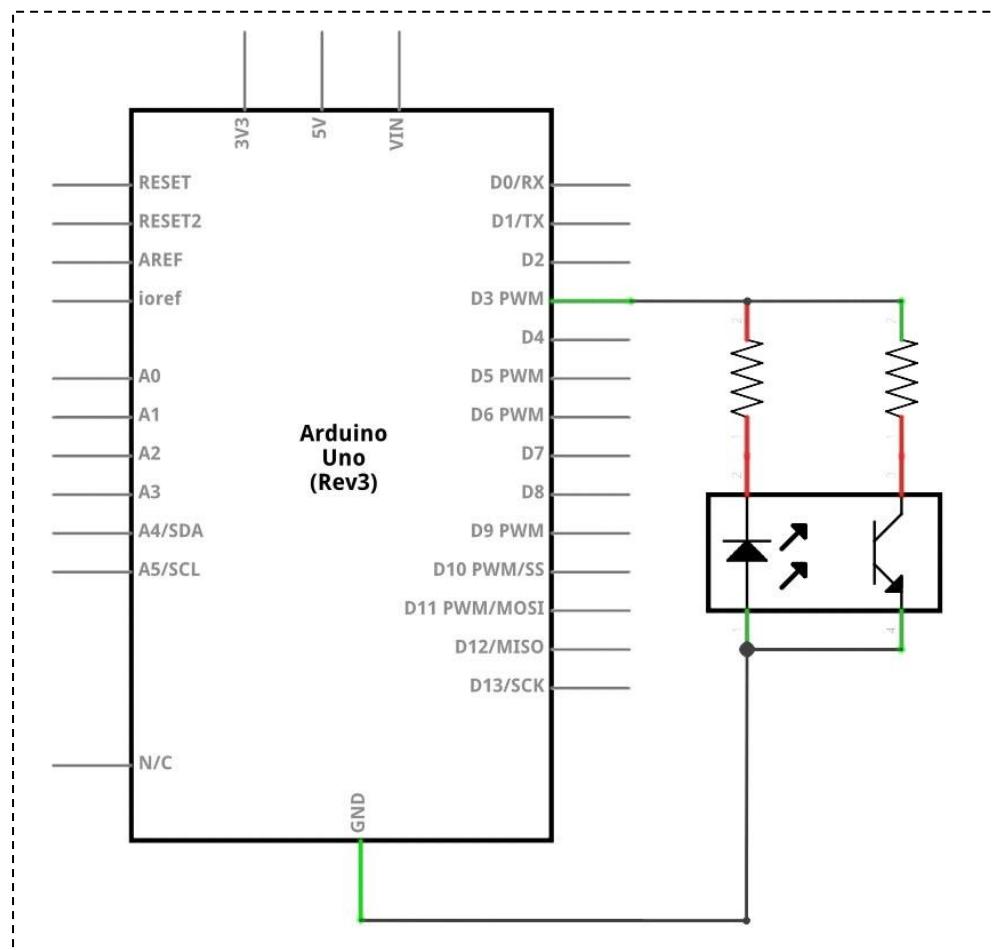


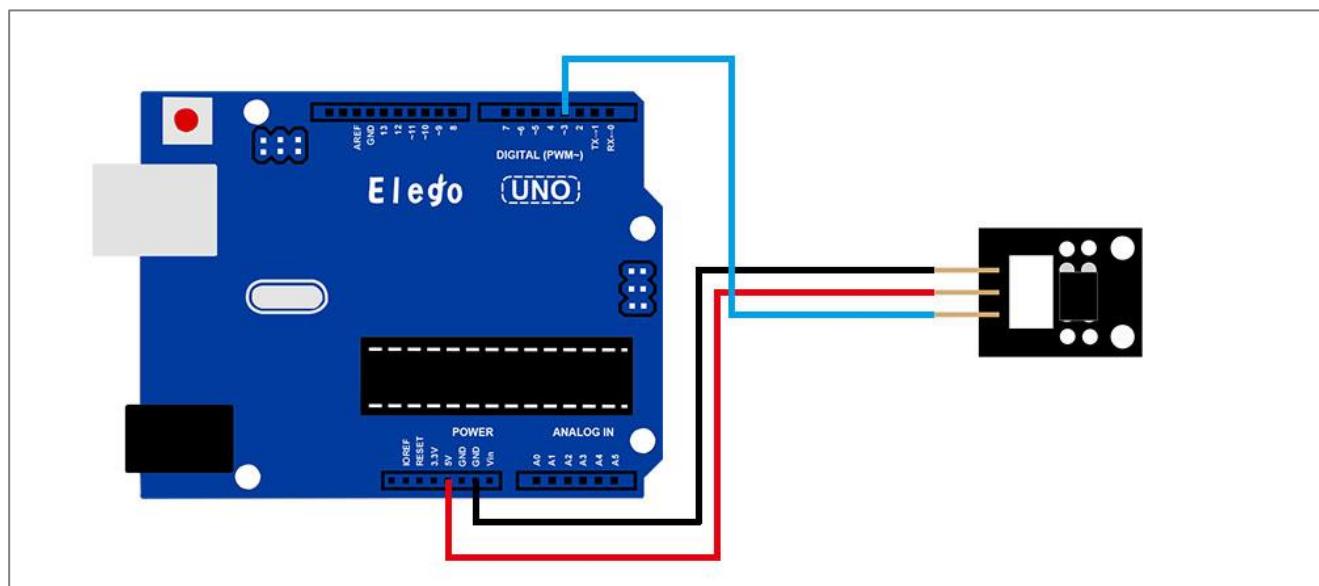
Photo-interrupter module and number 13 port have the built-in LED simple circuit. To produce a switch flasher, we can use connect the digital port 13 to the built-in LED and connect the Photo-interrupter MODULE S port to number 3 port of Elegoo Uno board. When the switch sensing, LED twinkle light to the switch signal.

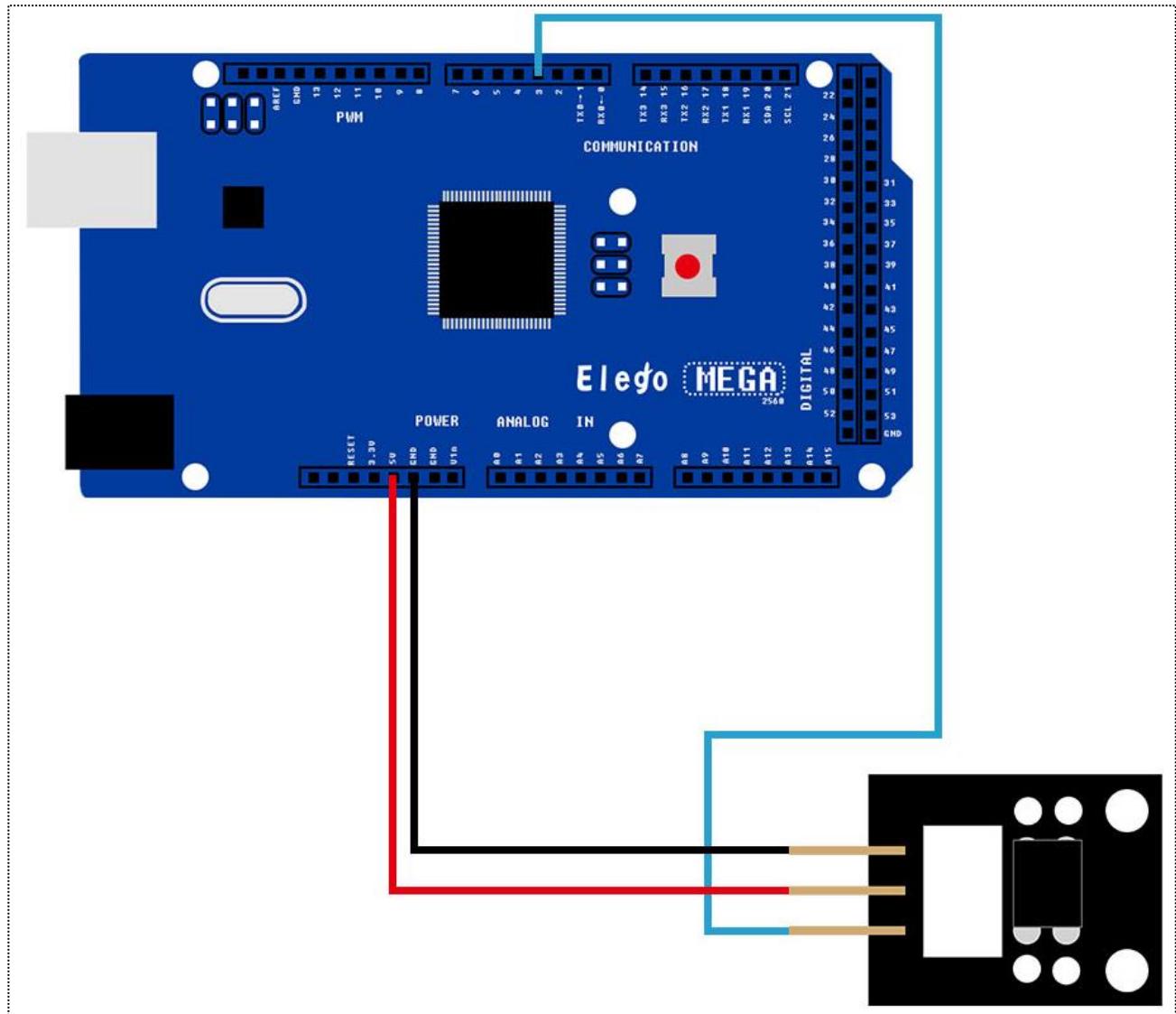
Connection

Schematic



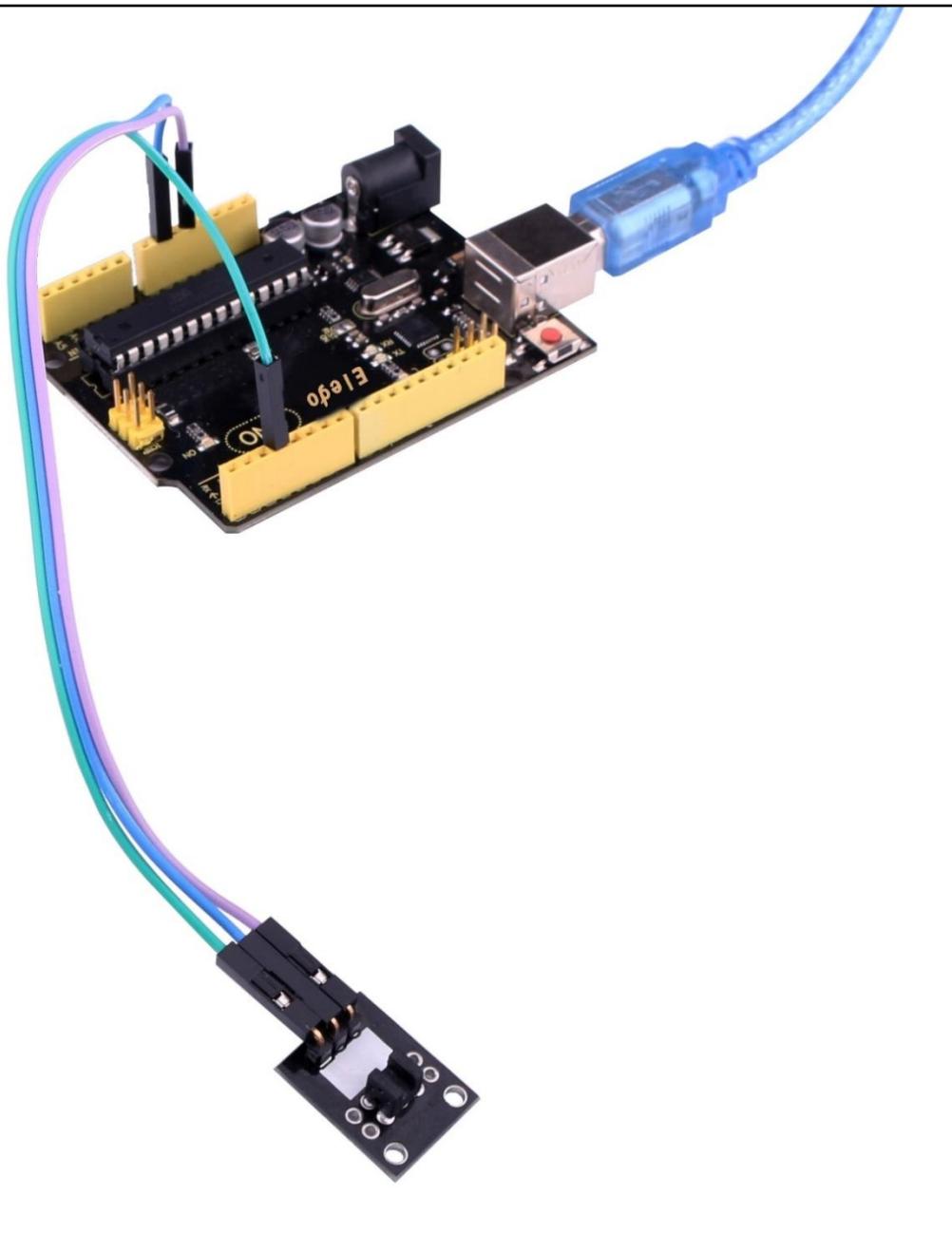
wiring diagram





Result

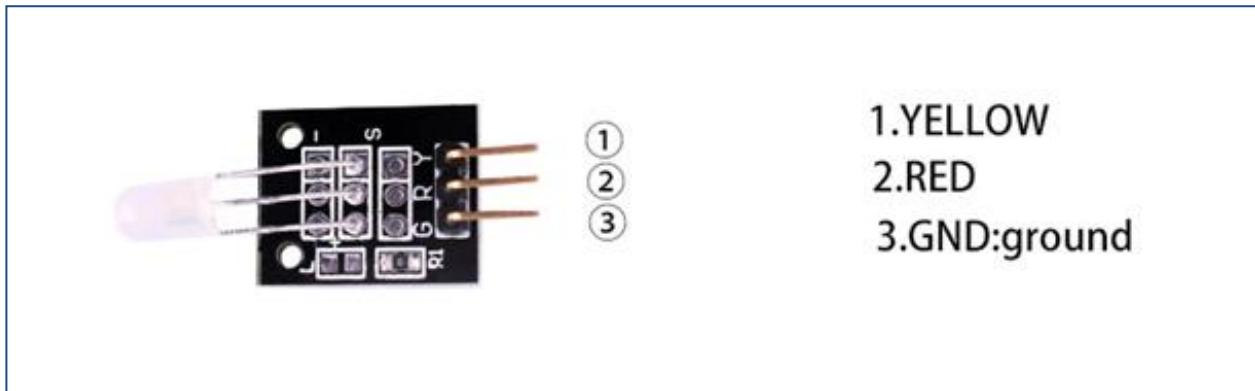
After we connect the circuit as the picture, we upload the program, we sensing the opto interrupter, then we can see the LED13 light up and light off.



Lesson 11 DUAL-COLOR COMMON-CATHODE LED

Overview

In this experiment, we will learn how to use Dual-color Common-Cathode LED.



Component Required:

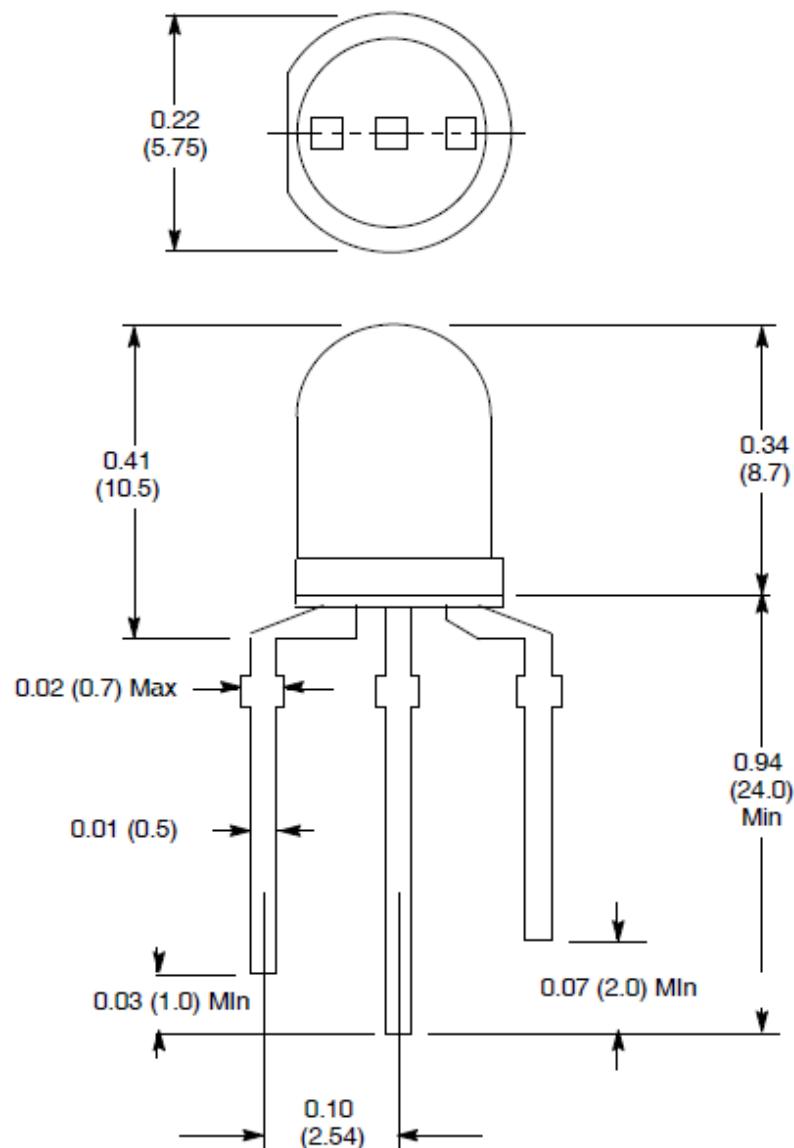
- (1) x Elegoo Uno R3
- (1) x USB cable
- (2) x Dual-color Common-Cathode LED
- (x) x F-M wires

Component Introduction

Dual-color Common-Cathode led:

Electro-Optical Characteristics: ($T_A = +25^\circ\text{C}$ unless otherwise specified)

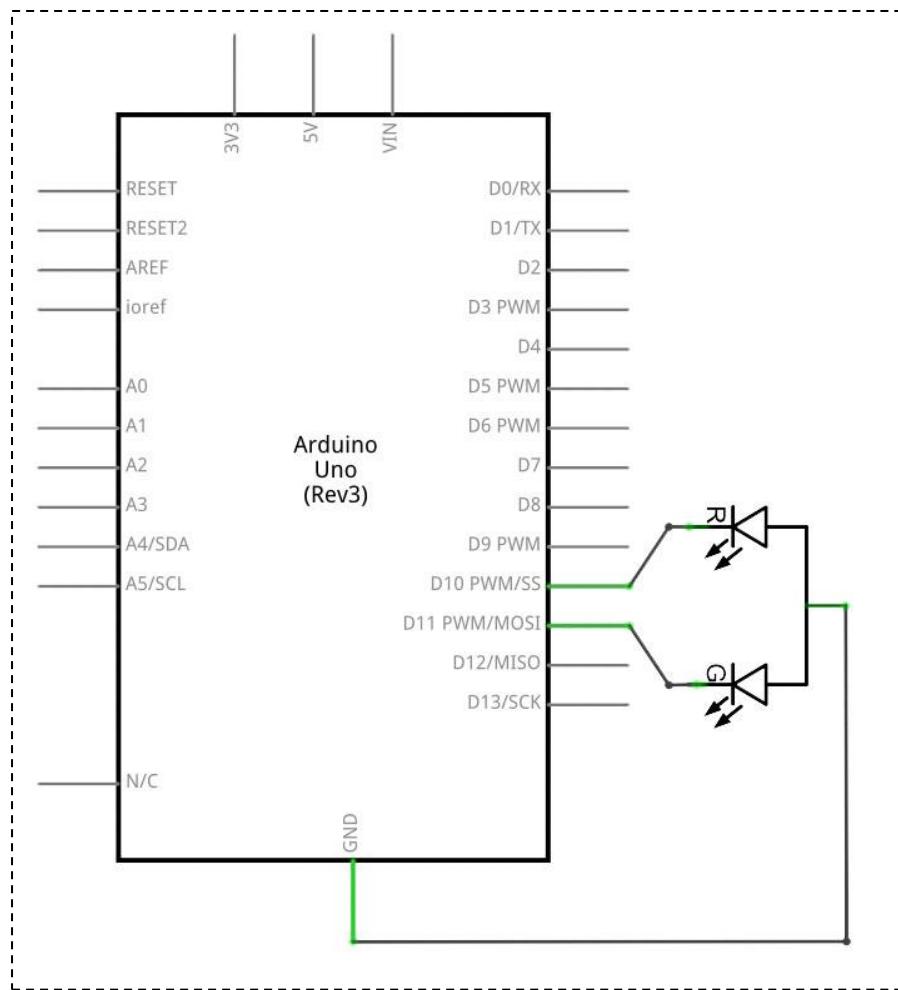
Parameter	Symbol	Test Conditions	Min	Typ	Max	Unit
View Angle of Half Power	$2\theta_{1/2}$	IF = 20mA	-	40	-	deg
Forward Voltage High Efficiency Red	VF	IF = 20mA	-	2.05	2.80	V
Yellow-Green			-	2.15	2.80	V
Luminous Intensity (Note 1)	IV	IF = 20mA	35	60	-	mcd
Peak Emission Wavelength High Efficiency Red	λ_p	IF = 20mA	-	625	-	nm
Yellow-Green			-	570	-	nm
Dominate Wave Length (Note 2) High Efficiency Red	$\lambda_d(\text{HUE})$	IF = 20mA	-	618	-	nm
Yellow-Green			-	567	-	nm



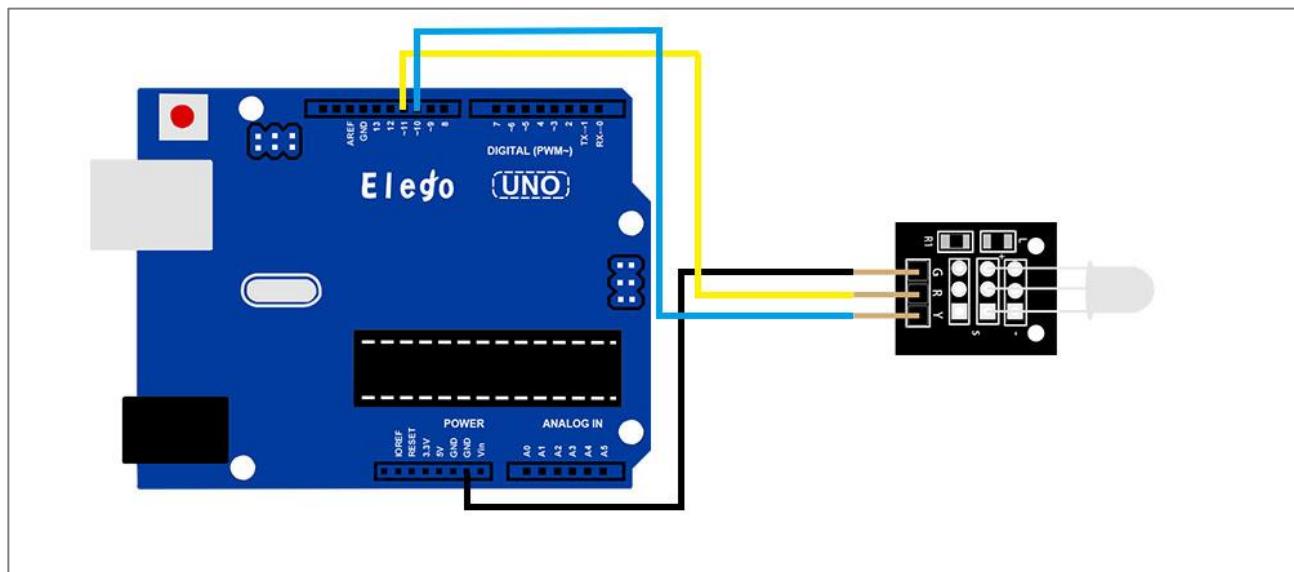
1. Red +
2. Common Lead -
3. Green +

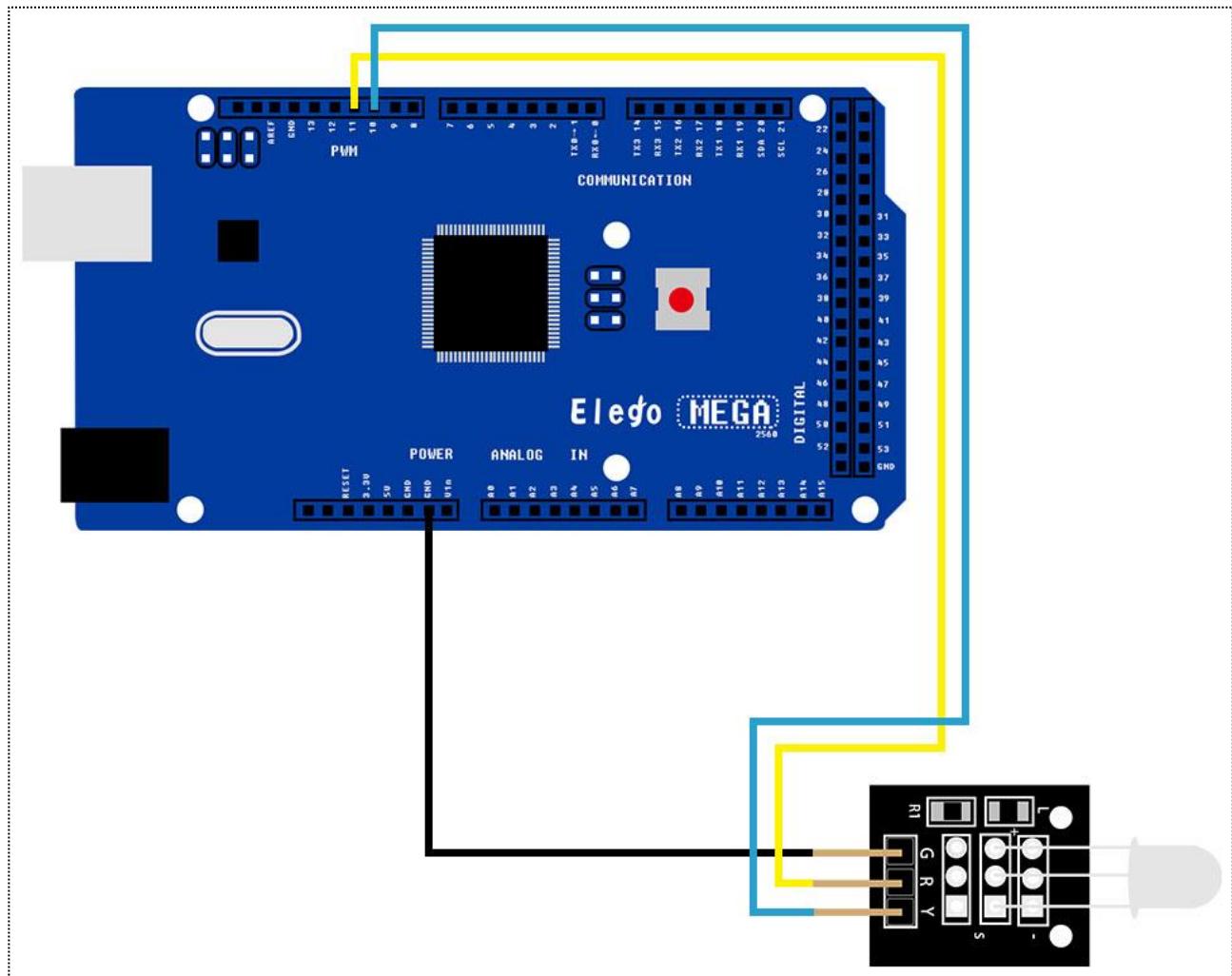
Connection

Schematic



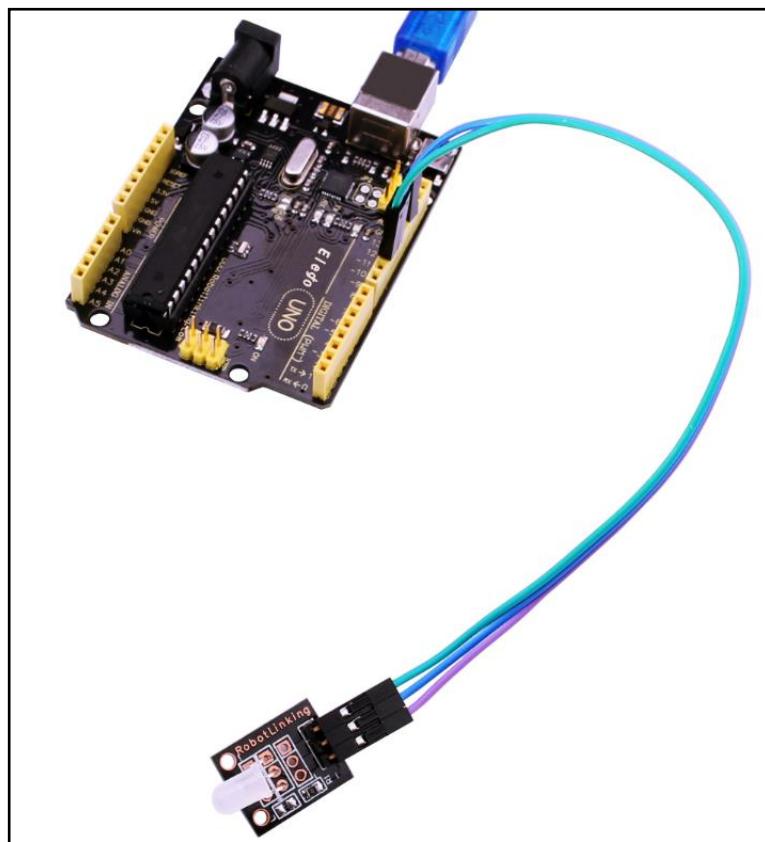
wiring diagram





Result

After we connect the circuit as the picture, we upload the program of each module. We can see the module changing their color as the code set. If you want to make it change the color in different way, you can revise the code.

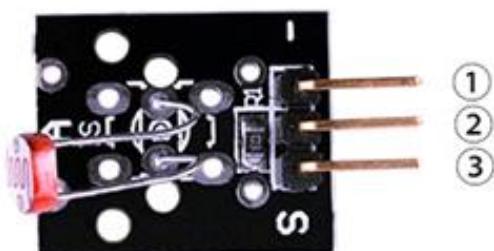


Lesson 12 PHOTORESISTOR MODULE

Overview

In this experiment, we will learn how to use the photo-resistor module.

Photo-resistor is very common in our daily life. it is mainly used in intelligent switch so as to bring convenience to our life. At the same time, in our daily life, we also use it in electronic design. So in order to use it in a better, we provide the corresponding modules to help us to use it more conveniently and efficiently.



- 1.GND:ground
- 2.VCC:3.3V-5V DC
- 3.OUTPUT

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Photoresistor module
- (x) x F-M wires

Component Introduction

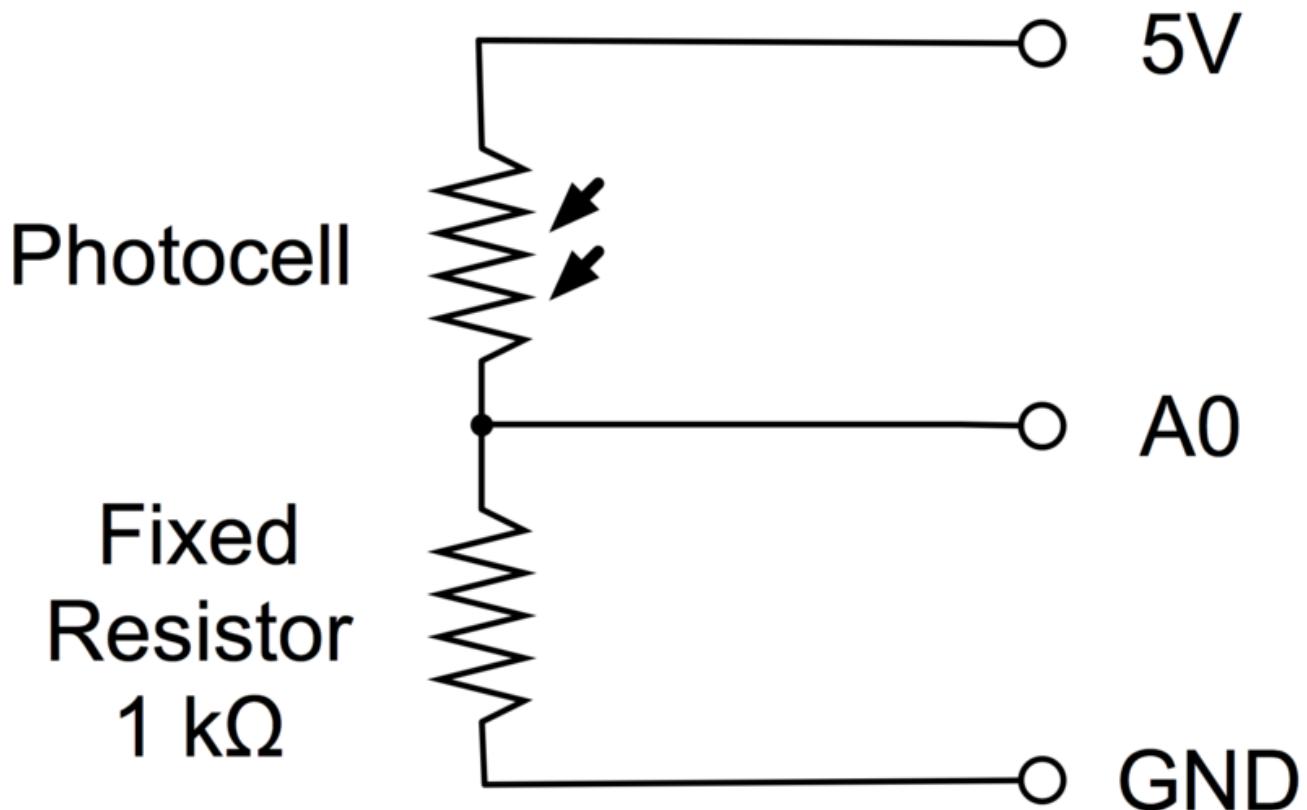
PHOTOCELL:

The photocell used is of a type called a light dependent resistor, sometimes called an LDR. As the name suggests, these components act just like a resistor, except that the resistance changes in response to how much light is falling on them.

This one has a resistance of about $50\text{ k}\Omega$ in near darkness and $500\text{ }\Omega$ in bright light. To convert this varying value of resistance into something we can measure on an Arduino's analog input,

it need to be converted into a voltage.

The simplest way to do that is to combine it with a fixed resistor.



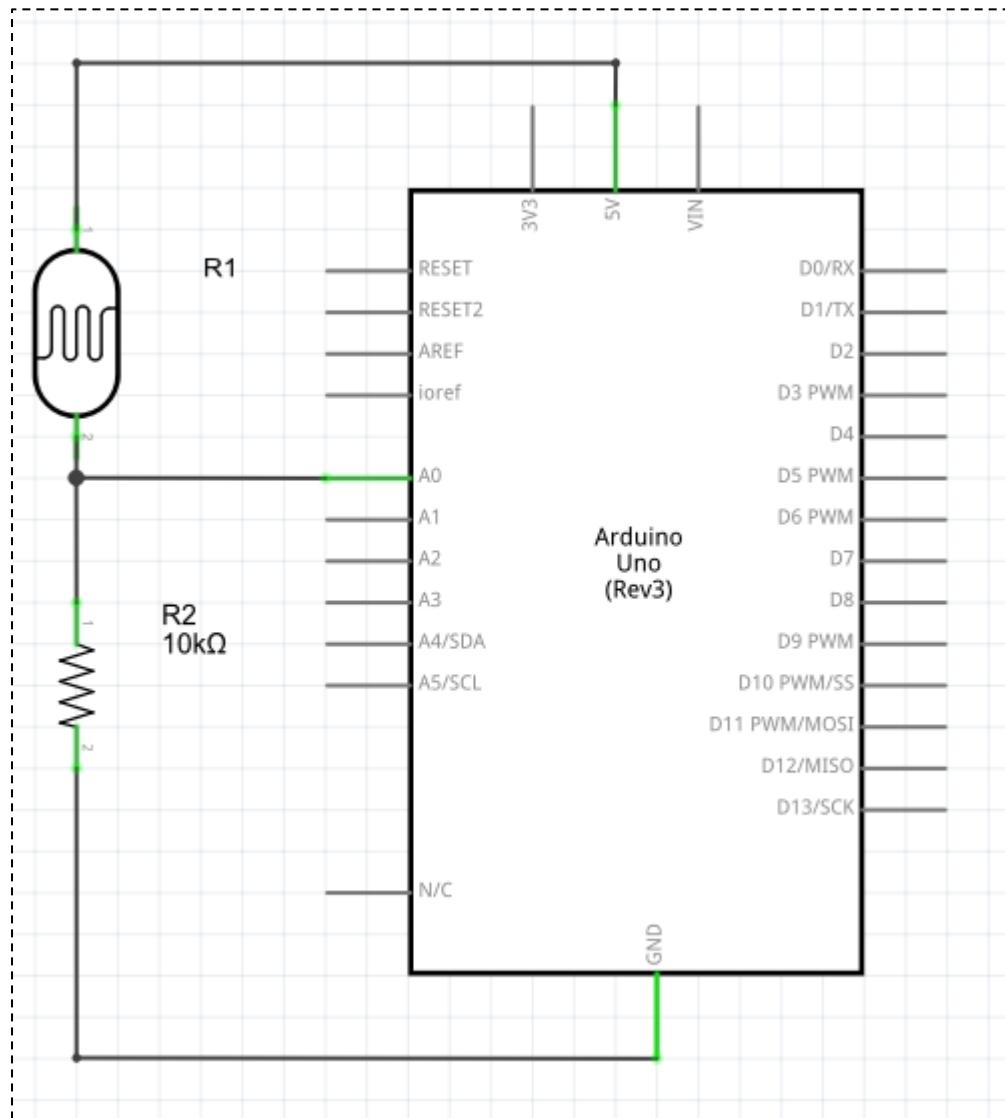
The resistor and photocell together behave rather like a pot. When the light is very bright, then the resistance of the photocell is very low compared with the fixed value resistor, and so it is as if the pot were turned to maximum.

When the photocell is in dull light the resistance becomes greater than the fixed 1kΩ resistor and it is as if the pot were being turned towards GND.

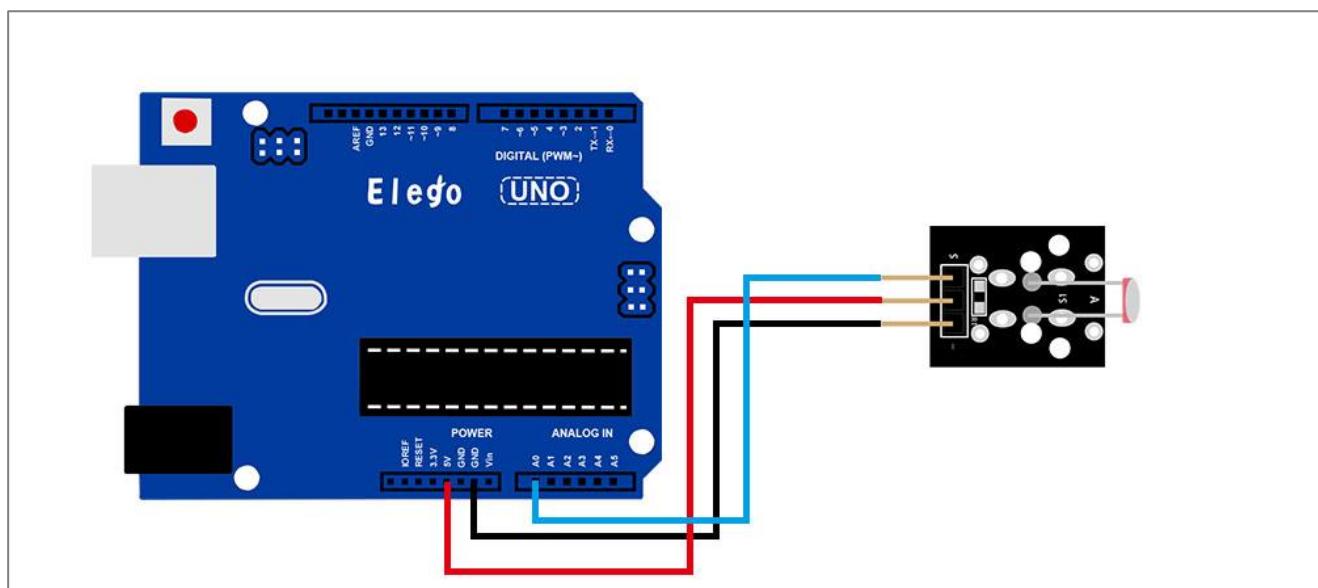
Load up the sketch given in the next section and try covering the photocell with your finger, and holding it near a light source.

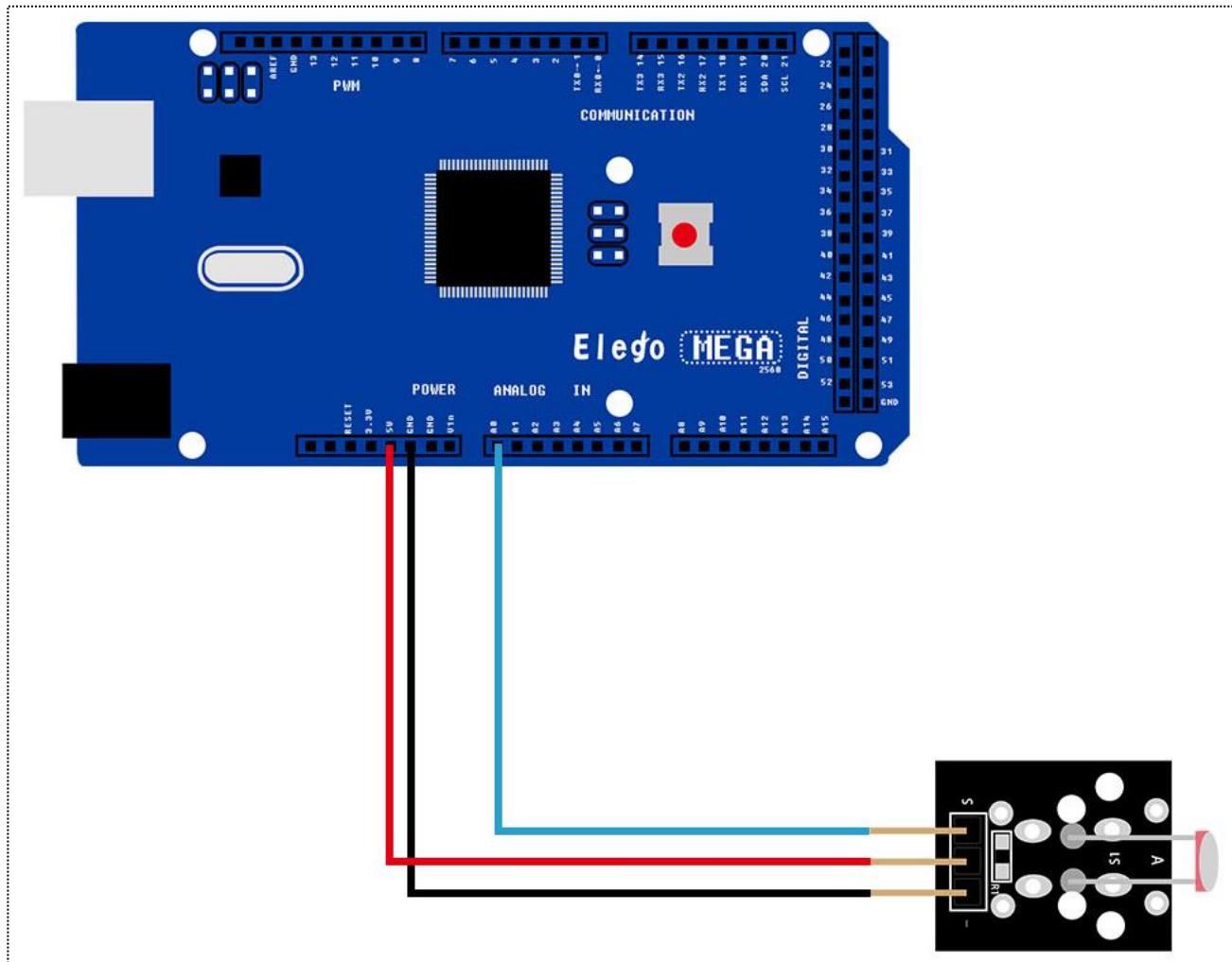
Connection

Schematic

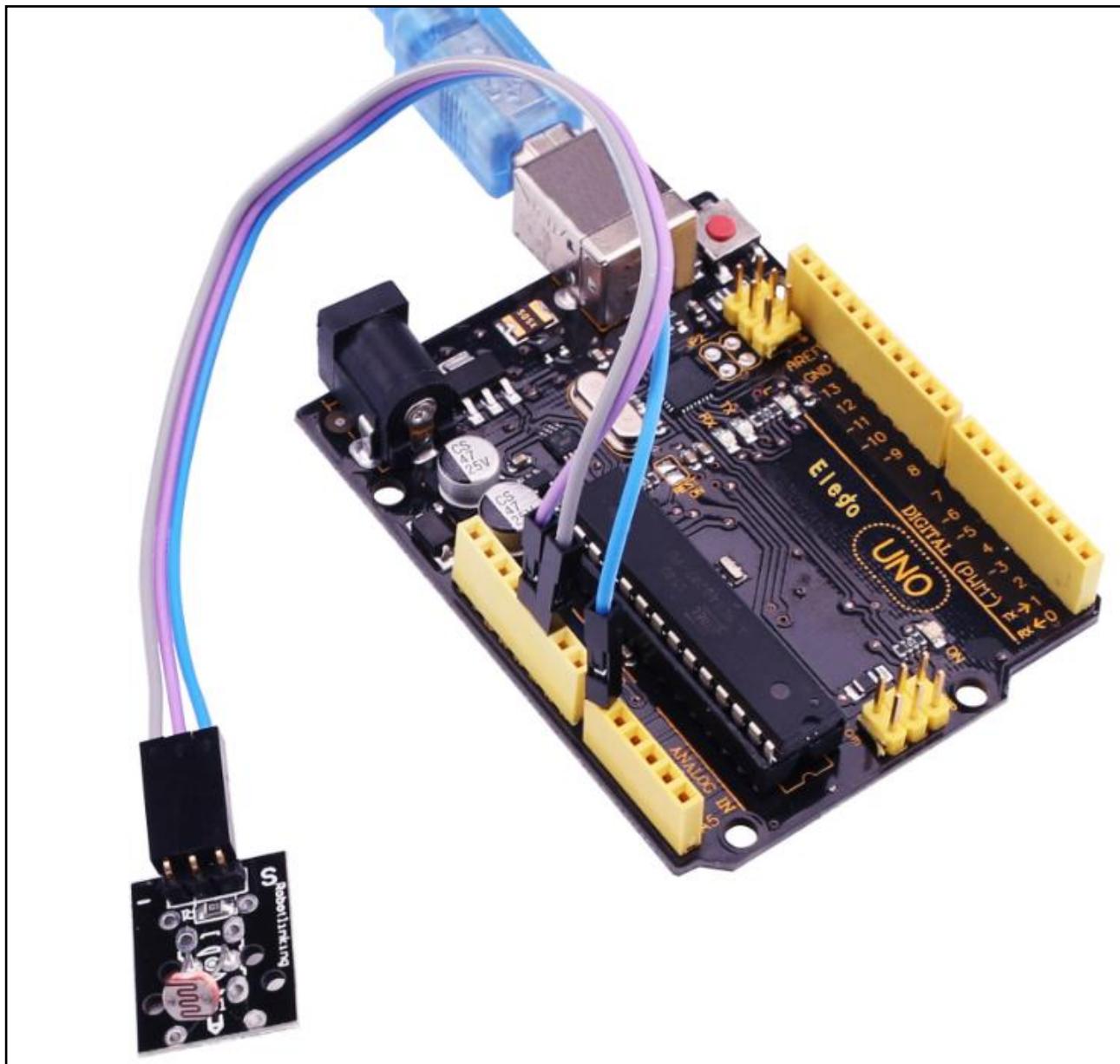


wiring diagram

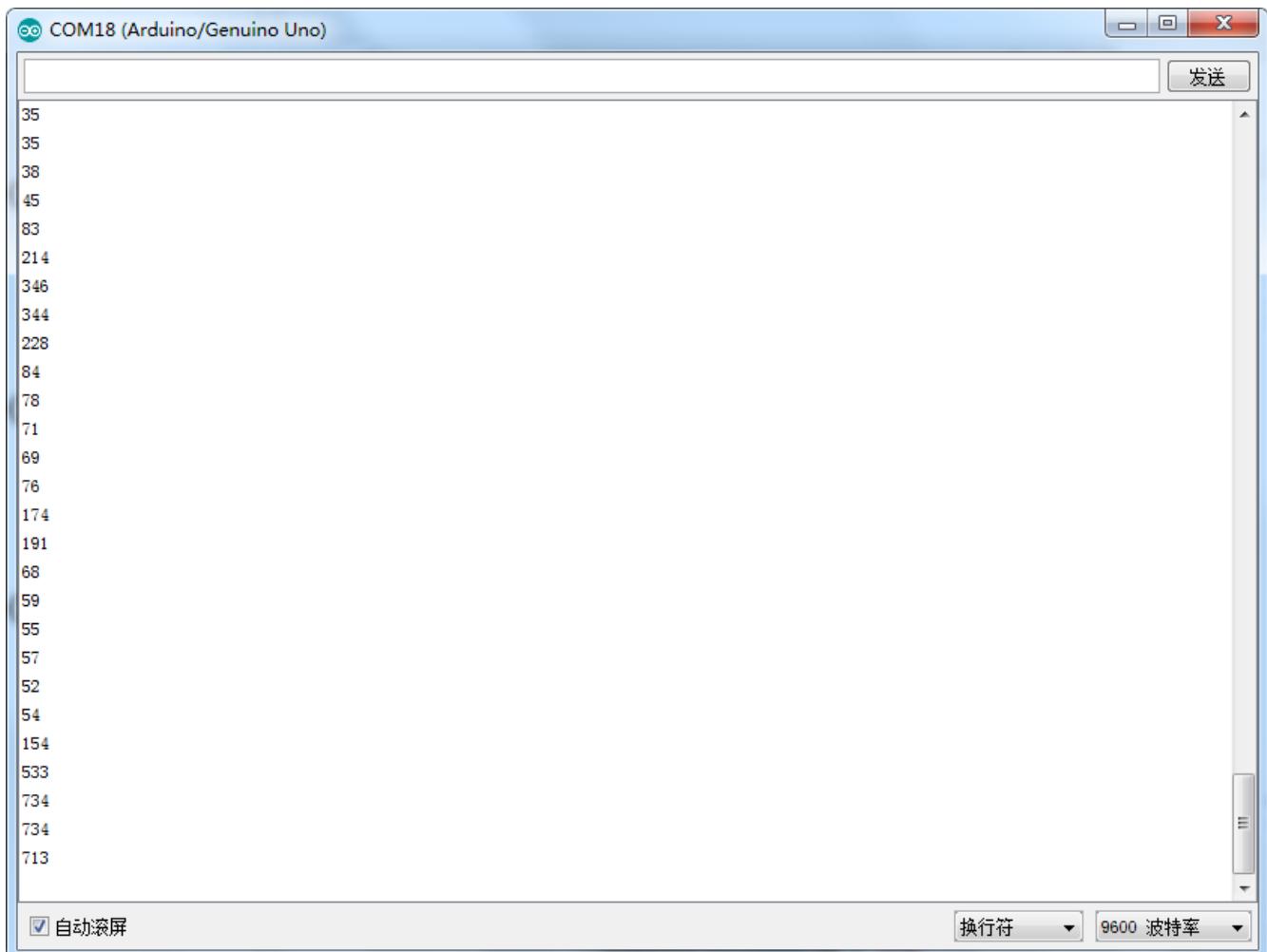




Result



Upload the program then open the monitor, we can see the data as below:

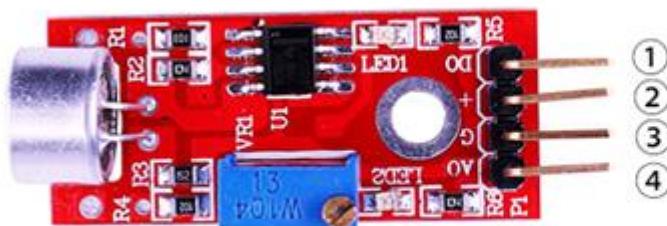


In the test, we only read the output analog voltage value of photo-resistor module. In the test results, we will find that when there is lighting, high voltage output equivalently of switch on, when there is no light, low voltage equivalently of switch off. This is what we can use this in practice.

Lesson 13 BIG SOUND SENSOR MODULE AND SMALL SOUND SENSOR MODULE

Overview

In this experiment, we will learn how to use the High-sensitive Voice Sensor.



- 1.D0:digital output
- 2.VCC: 3.3V-5V DC
- 3.GND:ground
- 4.AO:analog output



- 1.D0:digital output
- 2.VCC: 3.3V-5V DC
- 3.GND:ground
- 4.AO:analog output

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Big sound sensor module
- (1) x Small sound sensor module
- (x) x F-M wires

Component Introduction

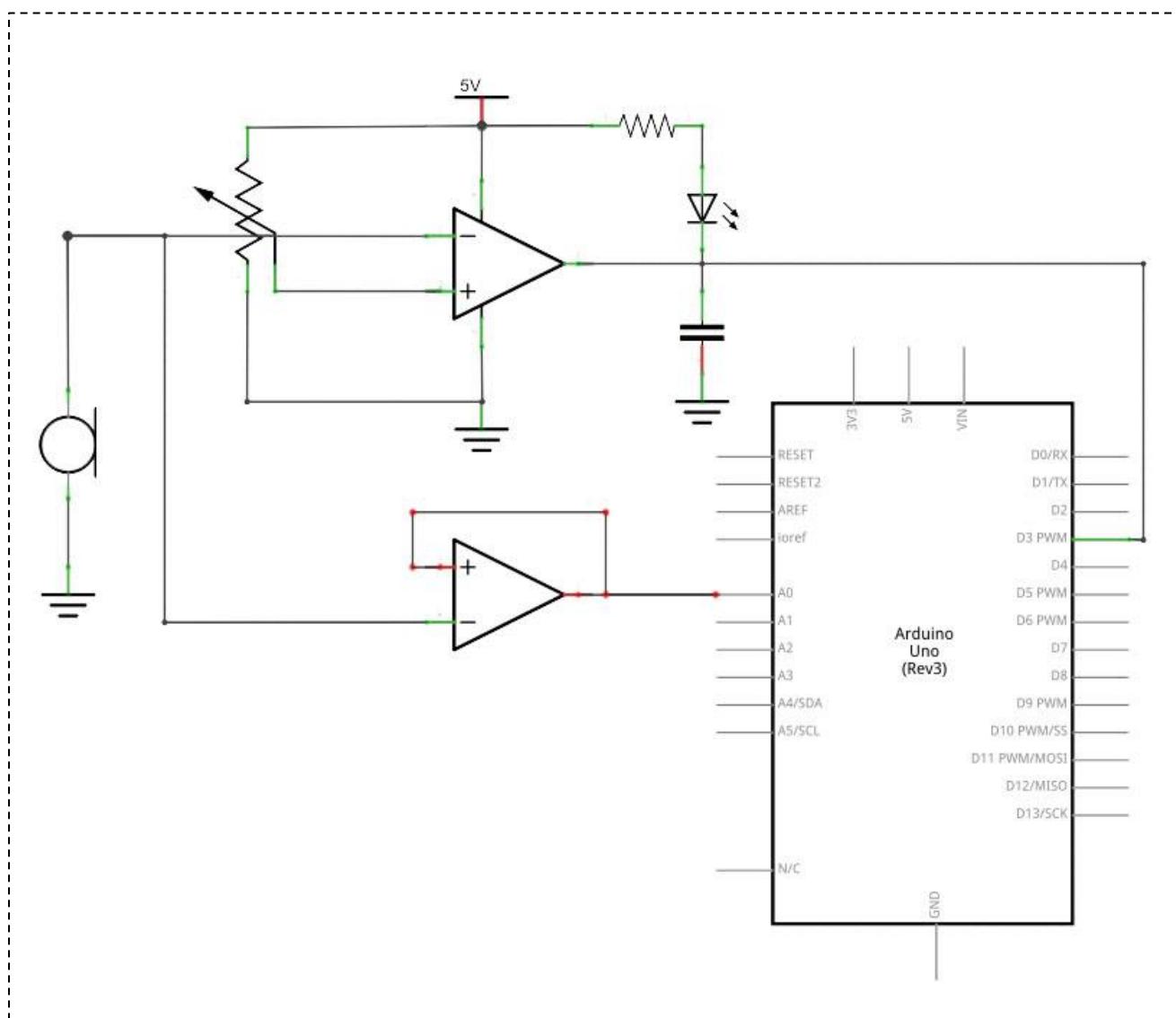
Sound sensor:

The sound sensor module provides an easy way to detect sound and is generally used for detecting sound intensity. This module can be used for security, switch, and monitoring applications. Its accuracy can be easily adjusted for the convenience of usage.

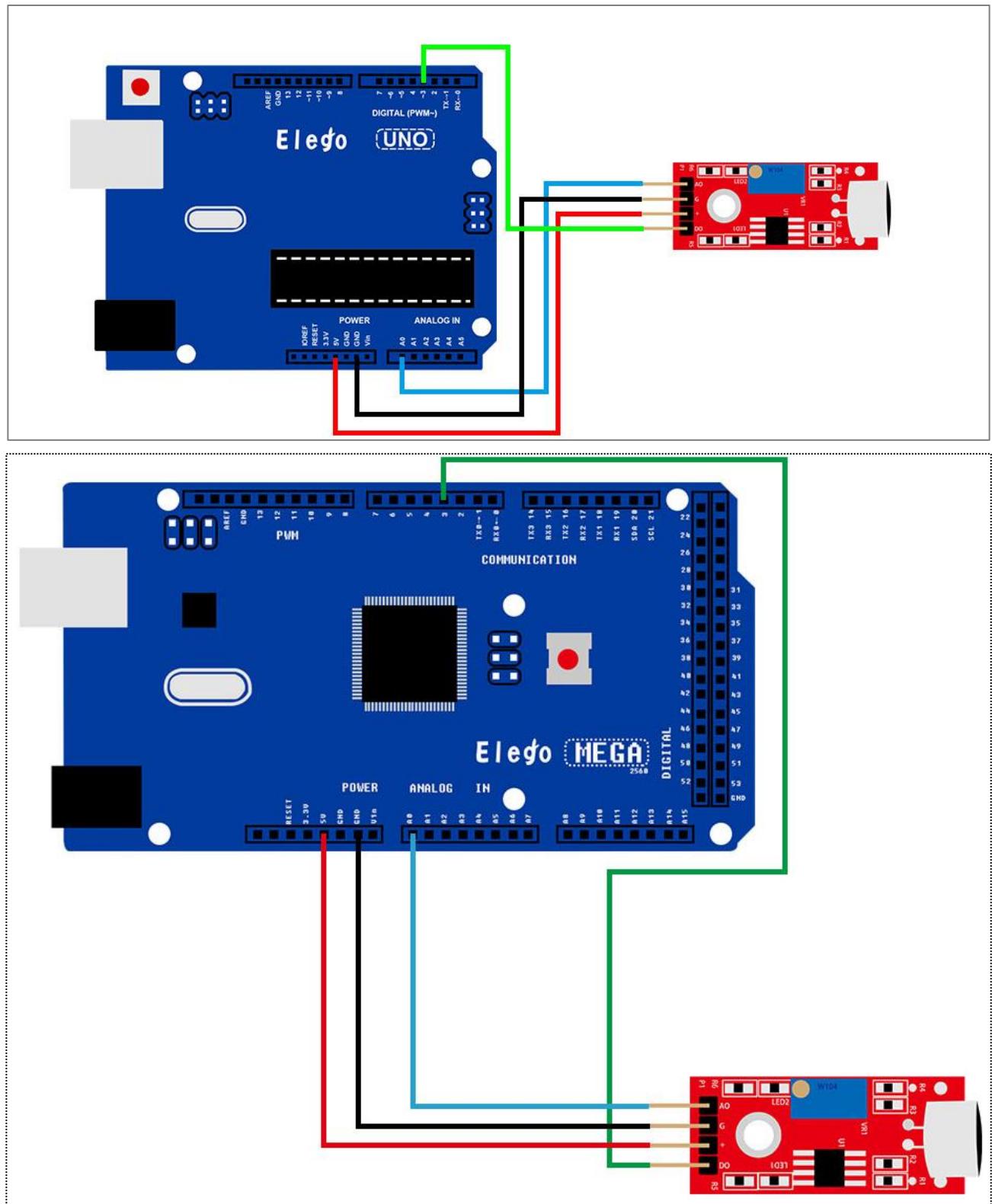
It uses a microphone which supplies the input to an amplifier, peak detector and buffer. When the sensor detects a sound, it processes an output signal voltage which is sent to a microcontroller then performs necessary processing.

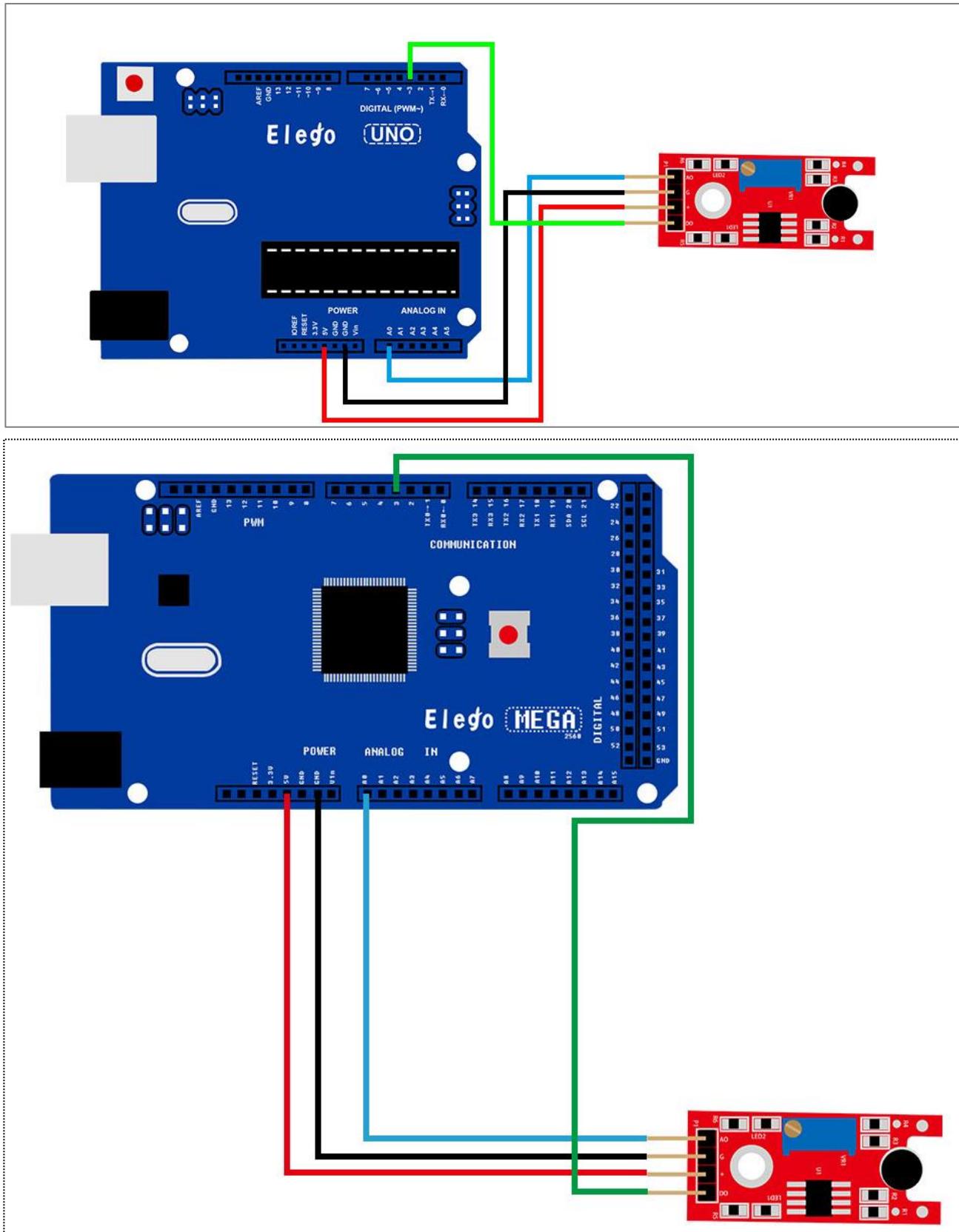
Connection

Schematic



wiring diagram





Result

High-sensitive Voice Sensor has two output:

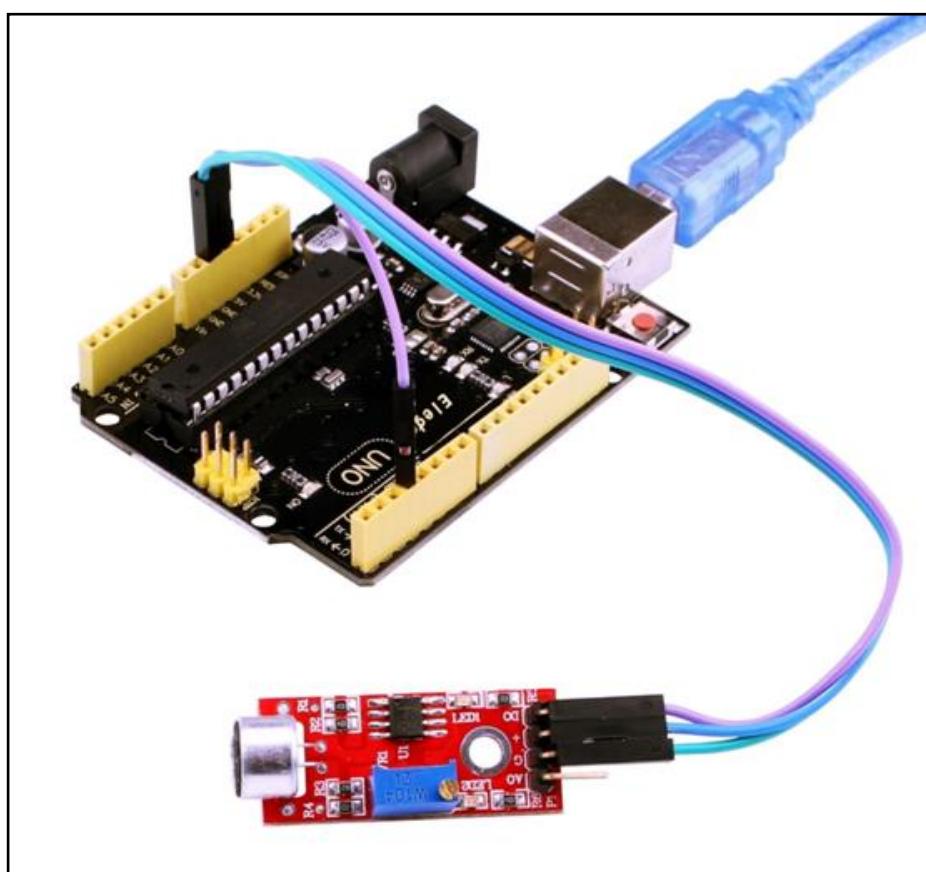
AO, analog output, real-time output voltage signal of microphone

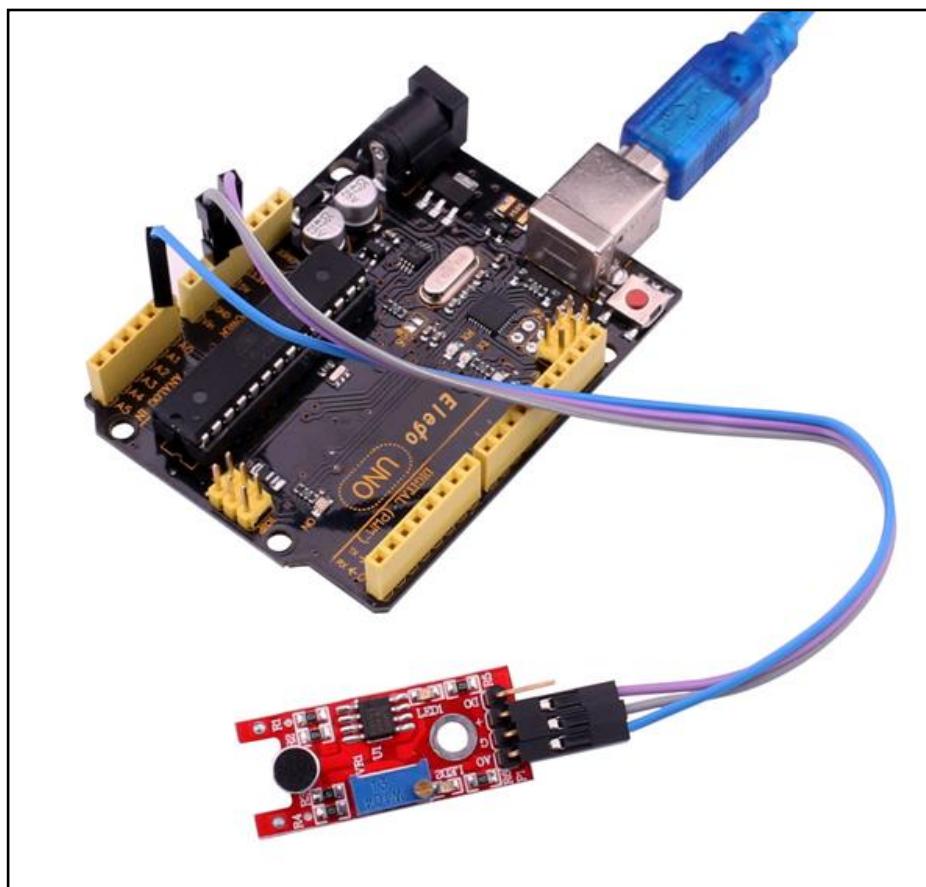
DO, when the intensity of the sound to reach a certain threshold, the output high and low level signal, the threshold-sensitivity can be achieved by potentiometer adjustment period.

The module features:

The installation of 3mm screw holes use 5 v dc power supply have analog output

turn a threshold level of output microphone GaoGan degree, high sensitivity. has a power light the comparator output indicator light. Upload the program then open the monitor, we can see the data as below:





```
COM16 (Arduino/Genuino Uno)
发送
40
47
42
34
24
43
43
47
42
50
36
31
28
37
53
26
58
34
45
35
40
35
42
56
36
36
36
自动滚屏 换行符 9600 波特率
```

Lesson 14 REED SWITCH MODULE

Overview

In this experiment, we will learn how to use reed switch and mini reed switch module.



- 1.D0:digital output
- 2.VCC: 3.3V-5V DC
- 3.GND:ground
- 4.AO:analog output

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Reed switch module
- (x) x F-M wires

Component Introduction

Reed Switch and Reed Sensor Activation:

Although a reed switch can be activated by placing it inside an electrical coil, many reed switches and reed sensors are used for proximity sensing and are activated by a magnet. As the magnet is brought into the proximity of the reed sensor/switch, the device activates. As the magnet is removed from the proximity of the reed sensor/switch, the device deactivates. However, the magnetic interaction involved in activating the reed switch contacts is not necessarily obvious. One way of thinking about the interaction is that the magnet induces magnetic poles into the metal parts of the reed switch and the resulting attraction between the electrical contacts causes the reed switch to activate.

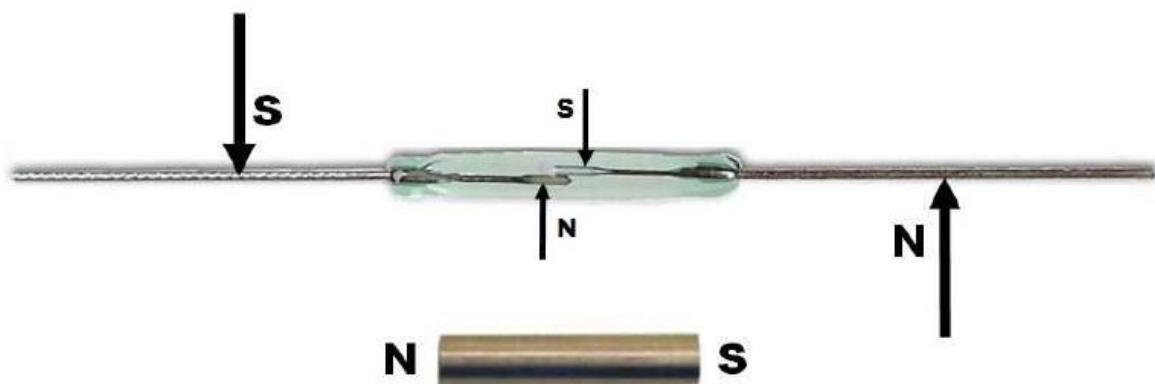


Figure 1 – Magnetic Induction

Another equally valid way of thinking about the interaction between a magnet and a reed switch is that the magnet induces magnetic flux through the electrical contacts. When the magnetic flux is high enough, the magnetic attraction between the contacts causes the reed switch to close.

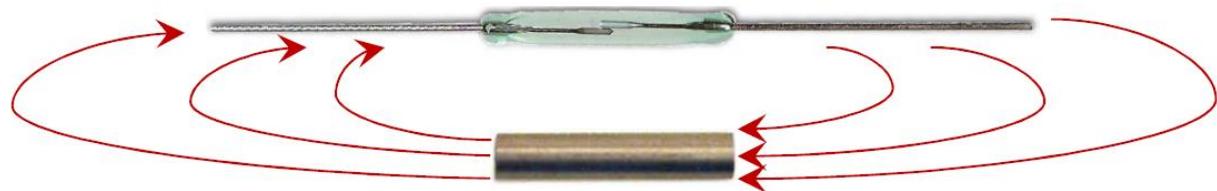


Figure 2 – Magnetic Flux

The following are examples of typical reed switch and reed sensor activate distances.

Difference between the reed switch module and mini reed switch module

As we can see, the reed switch module is bigger than the mini reed switch module. So the bigger one may have more function than the mini one. The reed switch can output in two ways: digital and analog. The mini reed can only output in digital.

In the 37 sensor kit, there have 7 red pcb modules. The difference between the red and small pcb is same as above.

Principle

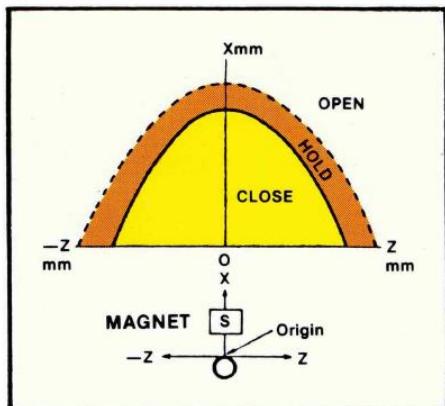


Figure 3 – Magnet Parallel to Reed Sw.

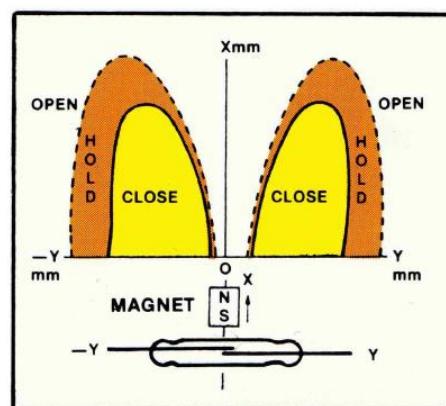


Figure 4 – Magnet Perpendicular to Reed Sw.

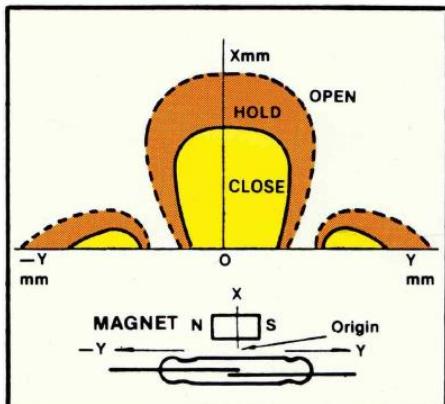
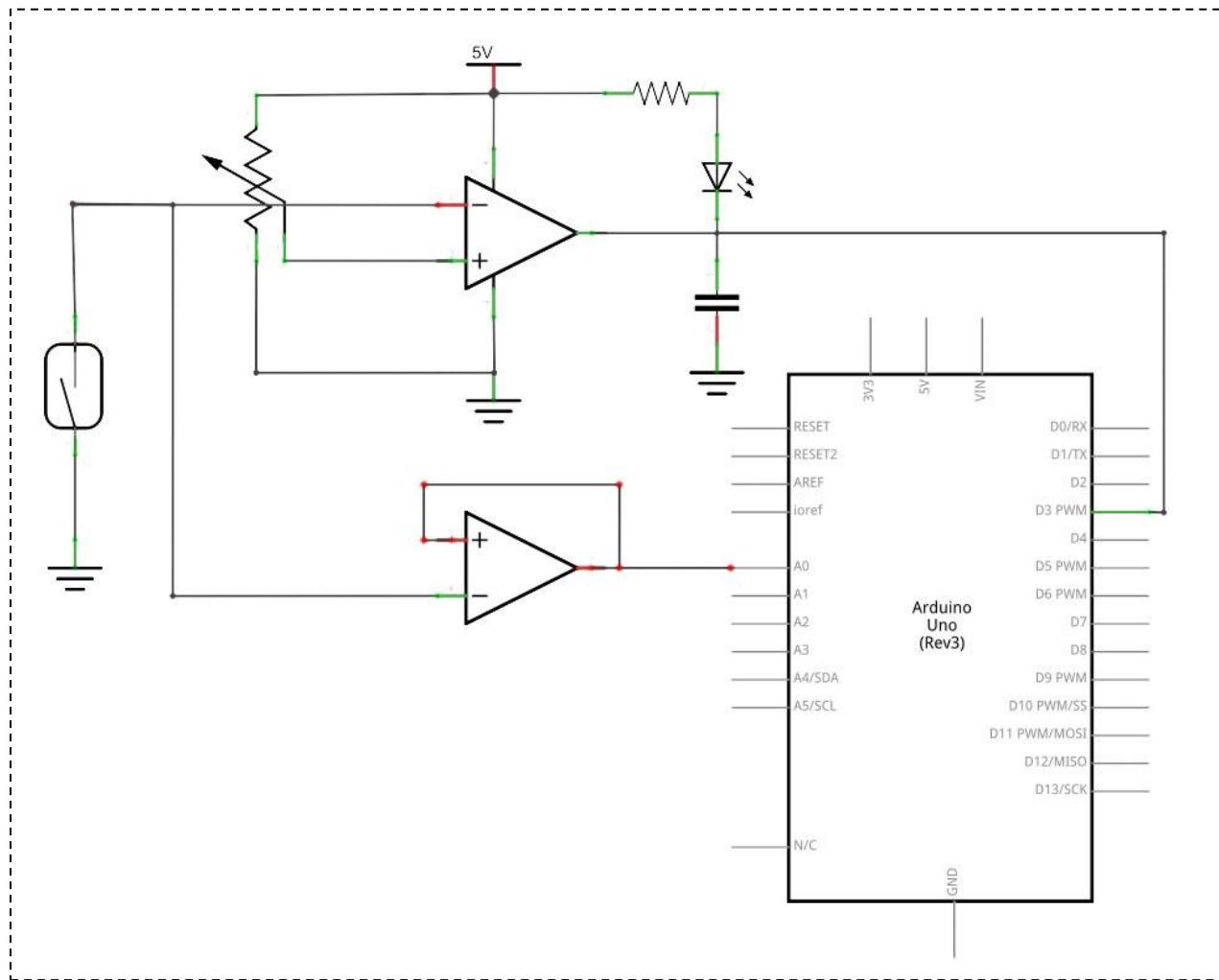


Figure 5 – Magnet Parallel to Reed Sw.

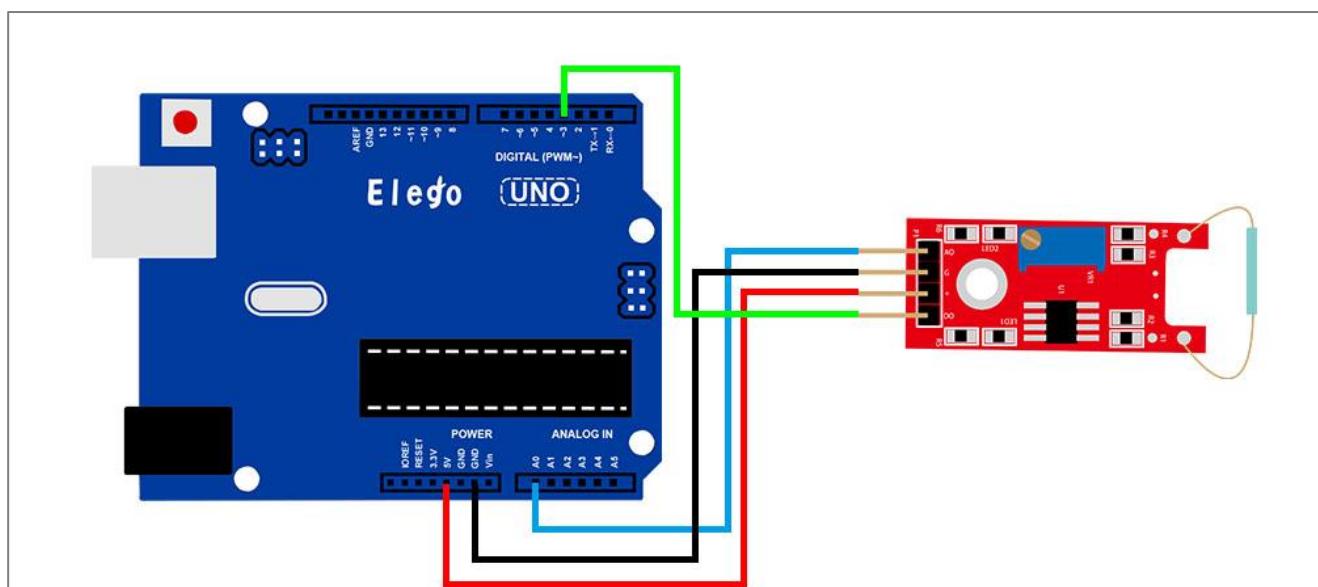
As can be seen, the magnetic orientation and location relative to the reed switch play important roles in the activation distances. In addition, the size of the activate regions(lobes) will vary depending on the strength of the magnet and the sensitivity of the reed switch. Proper orientation of the magnet with respect to the reed sensor/switch is an important consideration in meeting the application's requirements across the tolerance range for mechanical systems, magnetic strength and reed sensor or reed switch sensitivity.

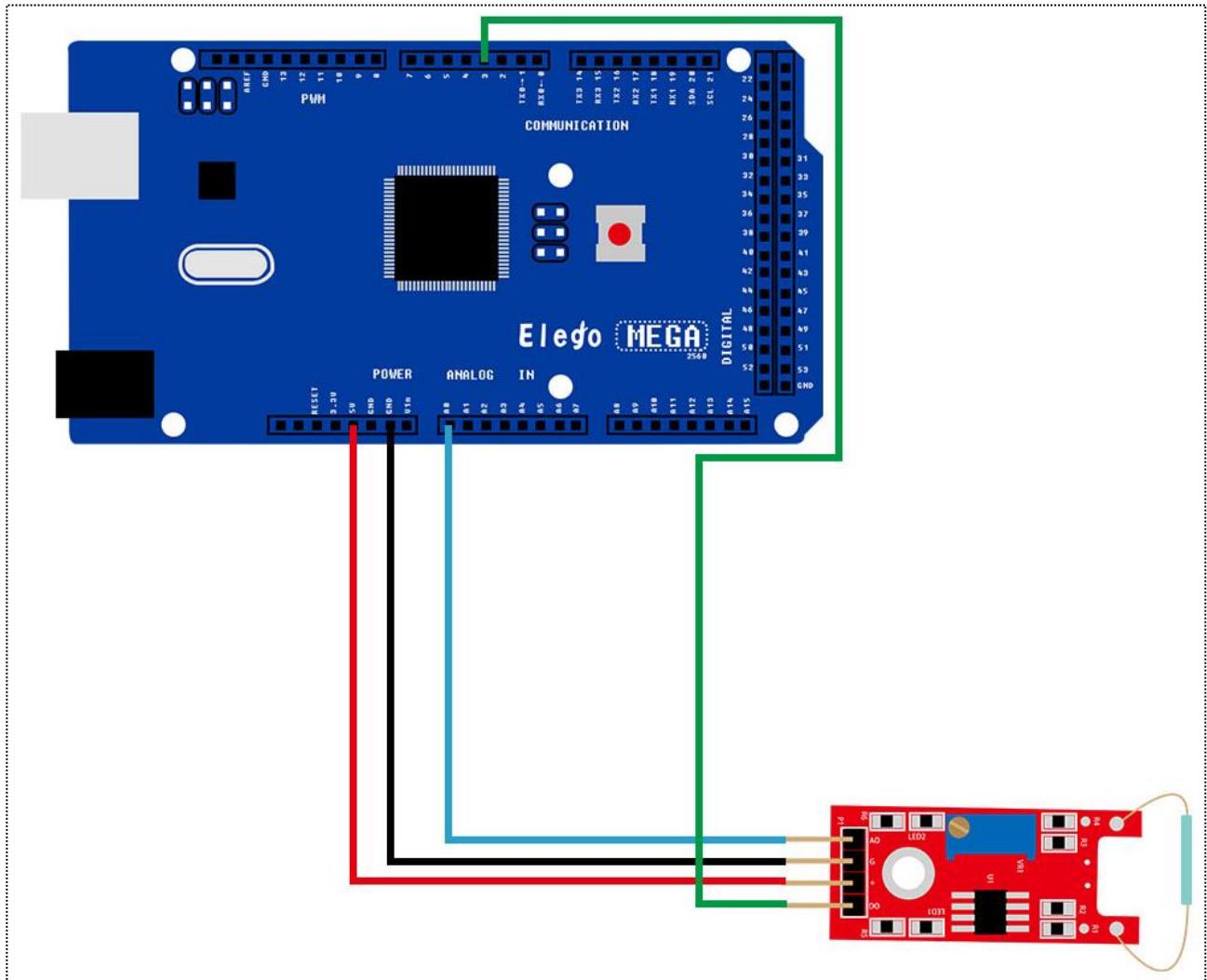
Connection of reed switch module

Schematic

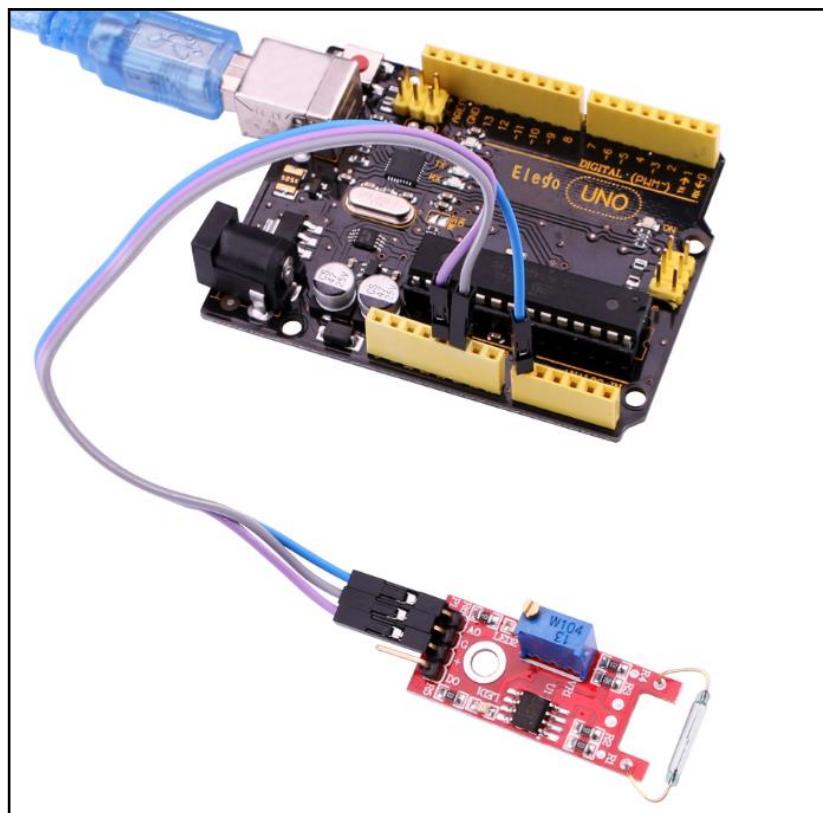


wiring diagram



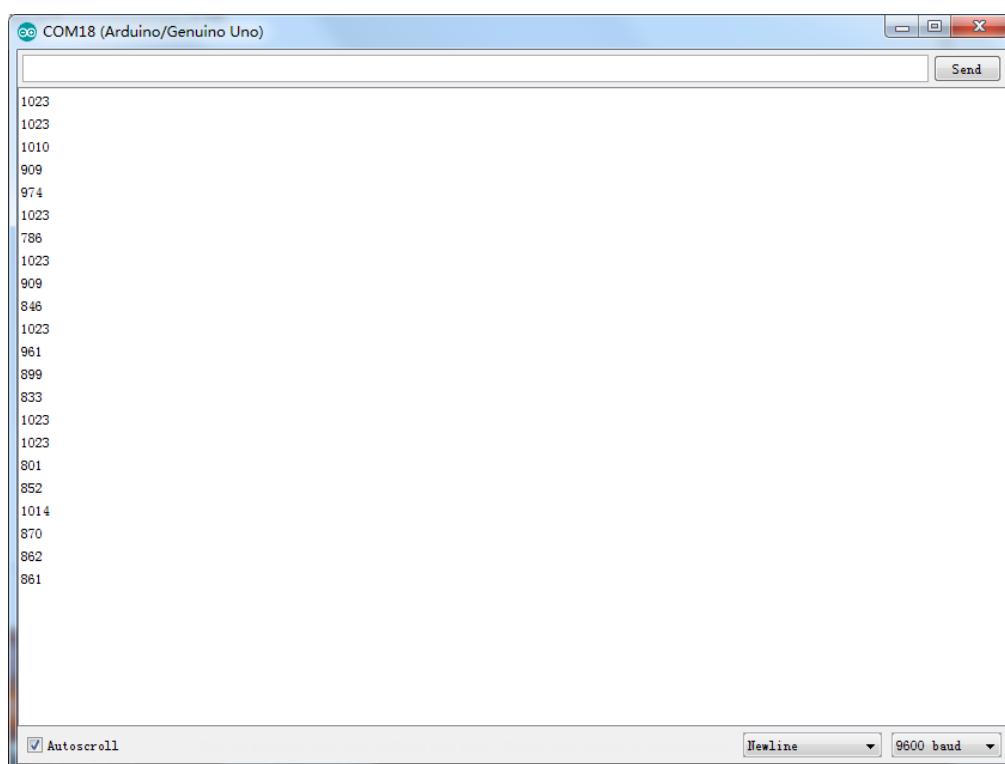


Result



In this experiment, we use the AO pin of reed switch module. When the sensor sensing magnetism, the module will output a data which reflect the strength of the magnetism. The number is from 0 to 1023.

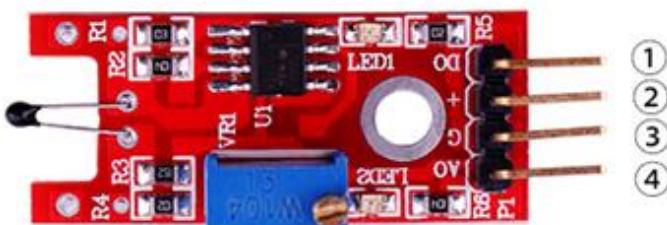
Upload the program then open the monitor, we can see the data as below:



Lesson 15 DIGITAL TEMPERATURE MODULE

Overview

In this experiment, we will learn how to use digital temperature module and analog temp module.



- 1.D0:digital output
- 2.VCC: 3.3V-5V DC
- 3.GND:ground
- 4.AO:analog output

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x digital temperature module
- (x) x F-M wires

Component Introduction

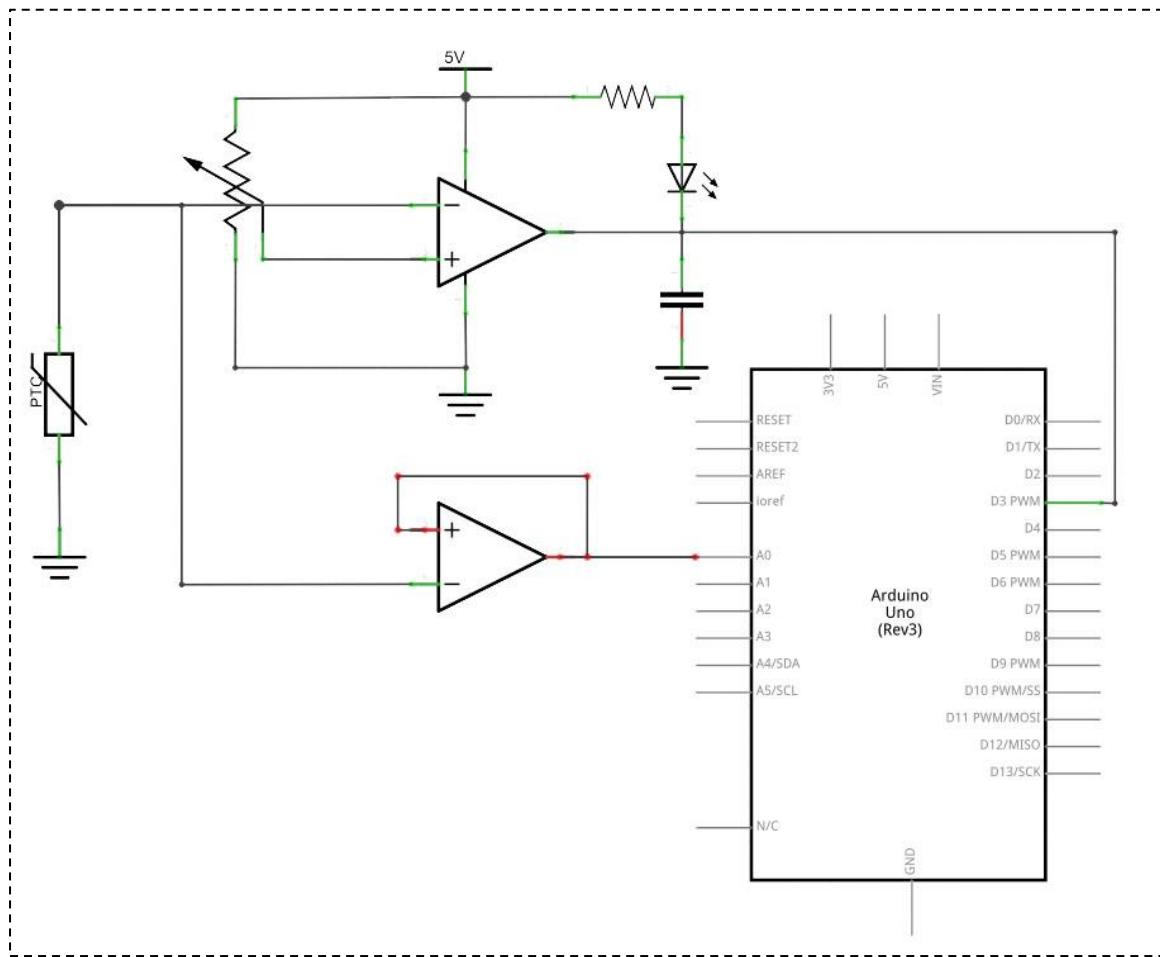
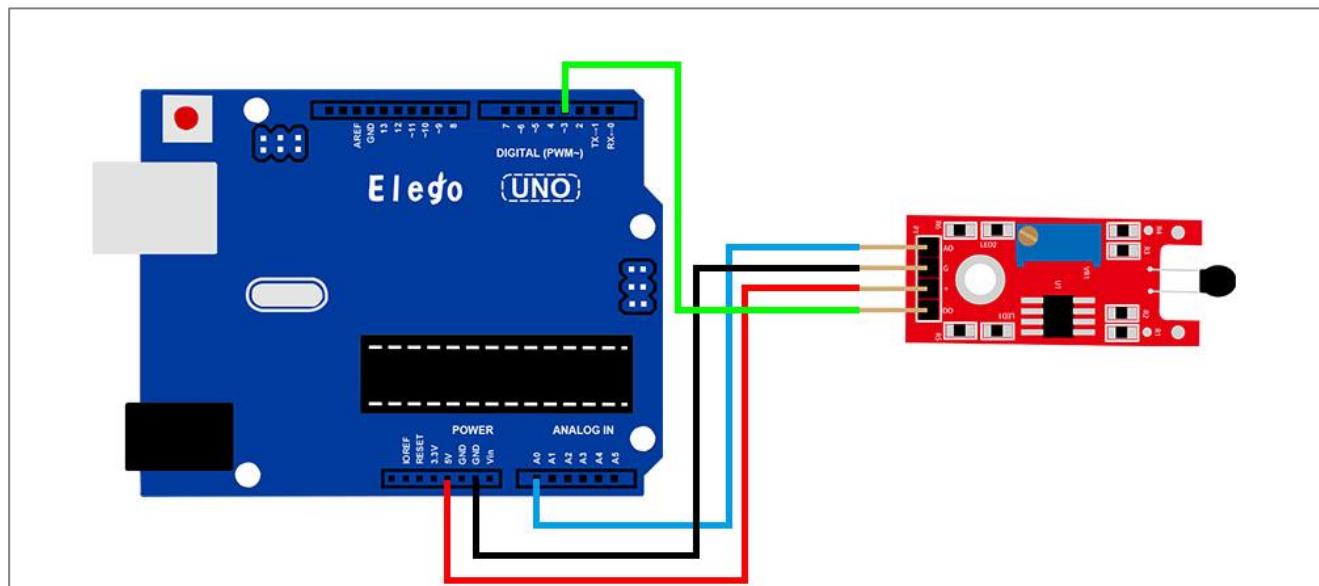
Thermistor:

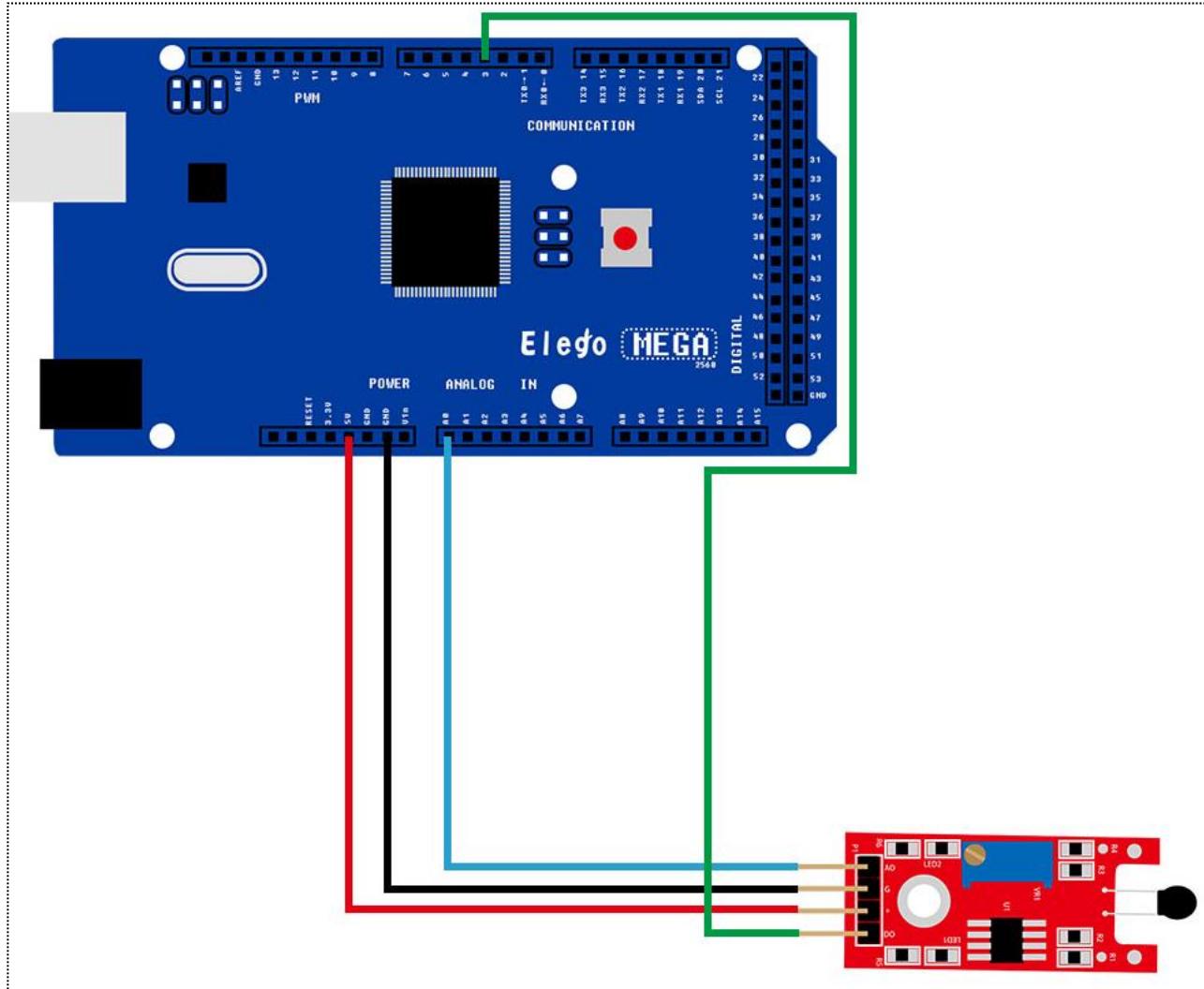


These thermistors have a narrow tolerance on the B-value, the result of which provides a very small tolerance on the nominal resistance value over a wide temperature range. For this reason the usual graphs of $R = f(T)$ are replaced by Resistance Values at Intermediate Temperatures Tables, together with a formula to calculate the characteristics with a high precision.

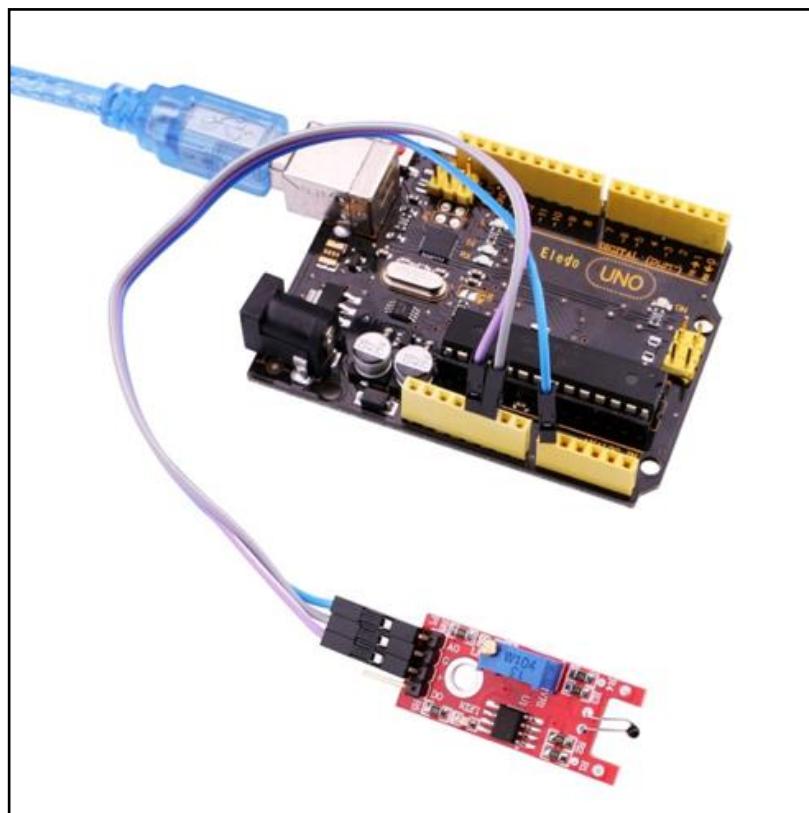
RS stock no.	151-215	151-221	151-237	151-243
Resistance at +25°C	3kΩ	5kΩ	10kΩ	100kΩ
Temperature range	-55°C to +150°C			
Tolerance (0 to +70°C)	±0.2°C			
Dissipation constant	1mW			
Time constant	10s			

Connection of temperature module

Schematic**wiring diagram**

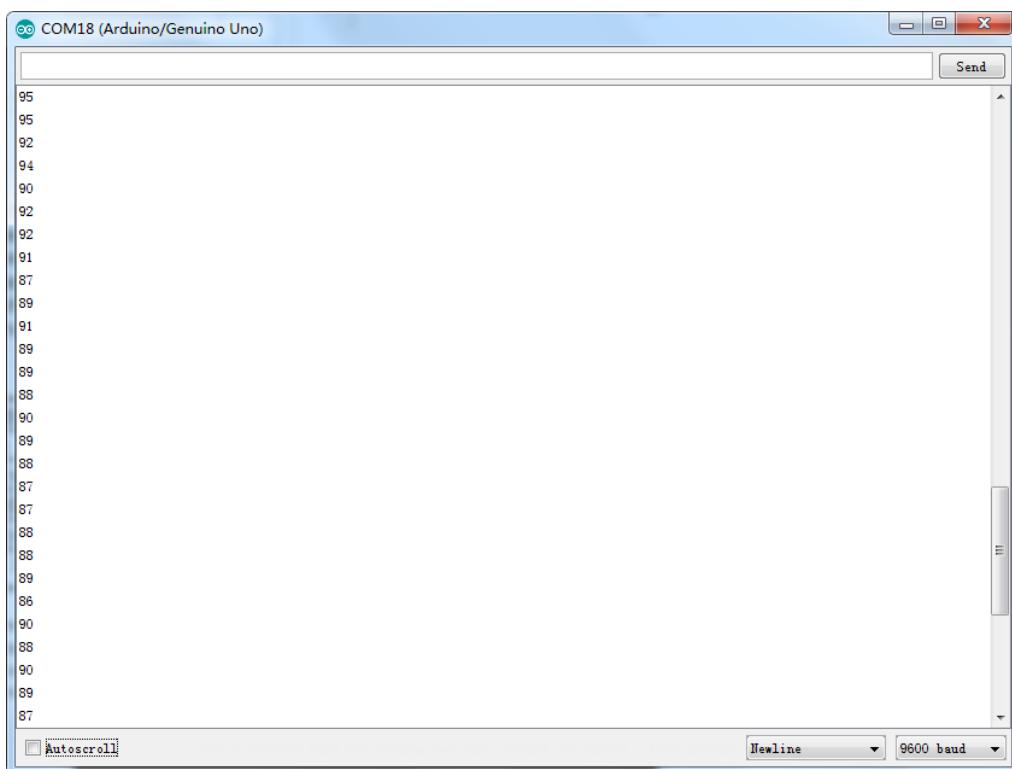


Result



In this experiment, we use the AO pin of reed switch module. When the sensor sensing temperature, the module will output a data which reflect the temperature. The number is from 0 to 1023.

Upload the program then open the monitor, we can see the data as below:

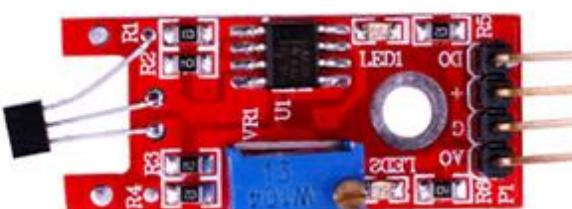
A screenshot of the Arduino Serial Monitor window titled "COM18 (Arduino/Genuino Uno)". The window shows a list of numerical values ranging from 86 to 95, each on a new line. The bottom status bar indicates "Autoscroll" is checked, and the baud rate is set to "9600 baud".

```
95
95
92
94
90
92
92
91
87
89
91
89
89
88
90
89
88
88
87
87
88
88
89
89
86
90
88
90
89
87
```

Lesson 16 LINEAR HALL SENSOR MODULE

Overview

In this experiment, we will learn how to use the linear hall sensor module.



- 1
- 2
- 3
- 4

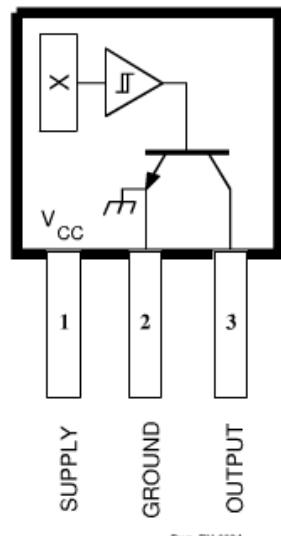
- 1.D0:digital output
- 2.VCC: 3.3V-5V DC
- 3.GND:ground
- 4.AO:analog output

Component Required:

- (1) x Elegoo Uno R3
- (1) x Linear hall sensor module
- (1) x USB cable
- (x) x F-M wires

Component Introduction

Hall Sensor:

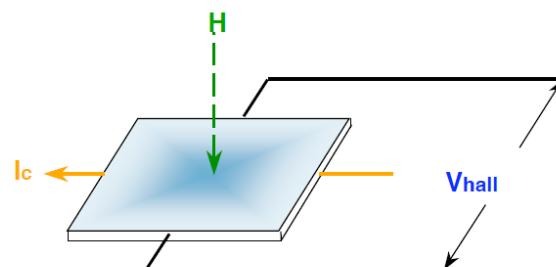


Pinning is shown viewed from branded side.

ABSOLUTE MAXIMUM RATINGS at $T_A = +25^\circ\text{C}$

Supply Voltage, V_{CC}	28 V
Reverse Battery Voltage, V_{RCC}	-35 V
Magnetic Flux Density, B	Unlimited
Output OFF Voltage, V_{OUT}	28 V
Reverse Output Voltage, V_{OUT}	-0.5 V
Continuous Output Current, I_{OUT}	25 mA
Operating Temperature Range, T_A	
Suffix 'E-'	-40^\circ\text{C} \text{ to } +85^\circ\text{C}
Suffix 'L-'	-40^\circ\text{C} \text{ to } +150^\circ\text{C}
Storage Temperature Range,	
T_S	-65^\circ\text{C} \text{ to } +170^\circ\text{C}

$$V_{Hall} \propto I_c \times H$$



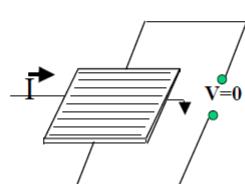
V_{Hall} = Output Hall-effect voltage

H = Magnetic Flux created by magnet or current-carrying conductor

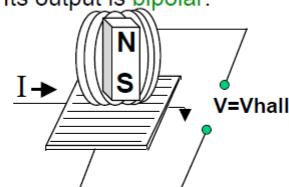
I_c = Constant supply current

Hall-effect Sensing Mechanism

- The current source is applied through a thin sheet of semiconductor material.

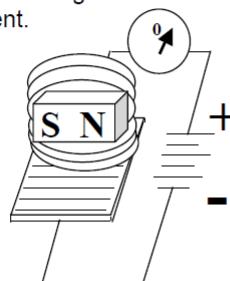


- A magnetic field applied perpendicular to the element creates a voltage change = V_{Hall} . Its output is bipolar.

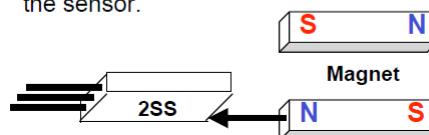


Magnetoresistive Sensing Mechanism

- A magnetic field applied parallel to the element changes its resistance and creates a current.



- MR is omnipolar—either pole will operate the sensor.



Design Factors – Magnetic Types

Unipolar: Only a south pole will operate the sensor. The sensor turns on with the south pole(+) and off when the south pole is removed.

- Bipolar: Sensor output is pole-dependent. A south pole (+) is designed to activate the sensor; a north pole(-) is designed to deactivate. It's possible that the sensor could turn off and still be within a positive Gauss level.

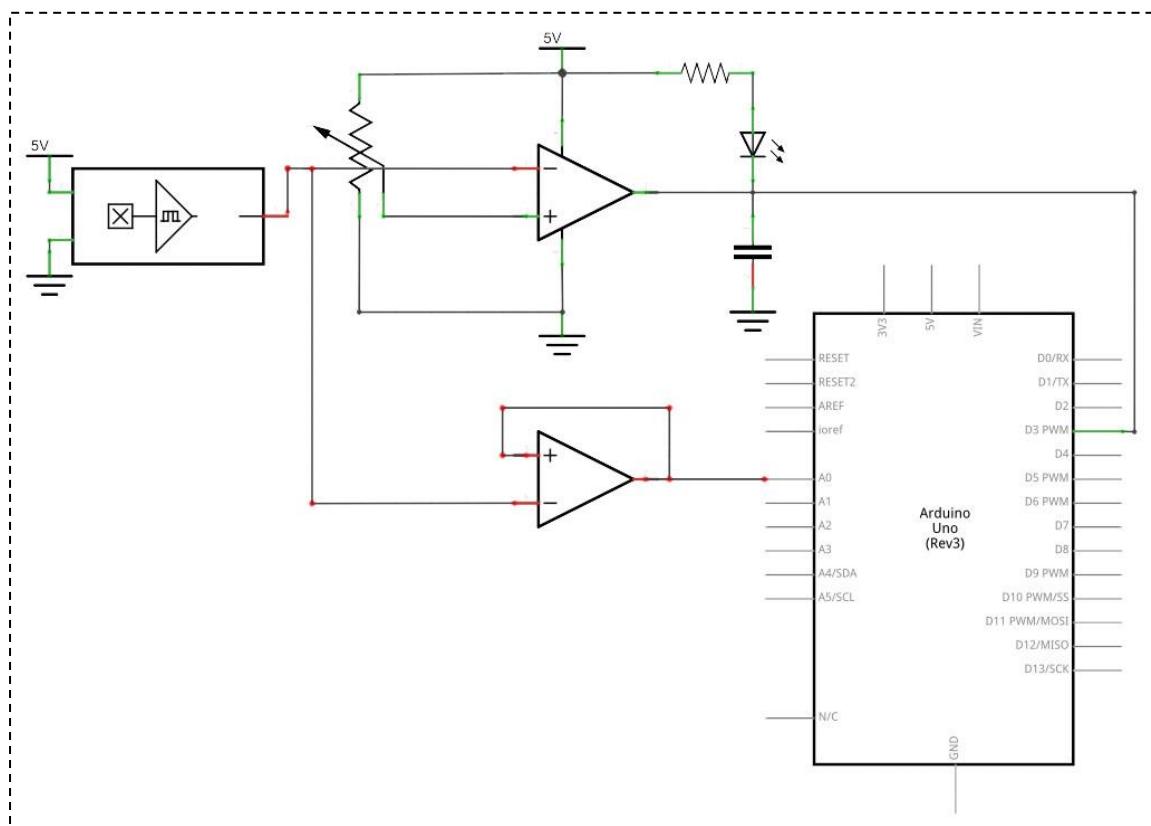
- Latching: Specifications are tighter on latching. Sometimes it is designed to make certain that when the south pole(+) is removed from the sensor, it will stay on until it sees the opposite pole(-).

- Omnipolar: The sensor is designed to operate with either magnetic pole(+) or (-).

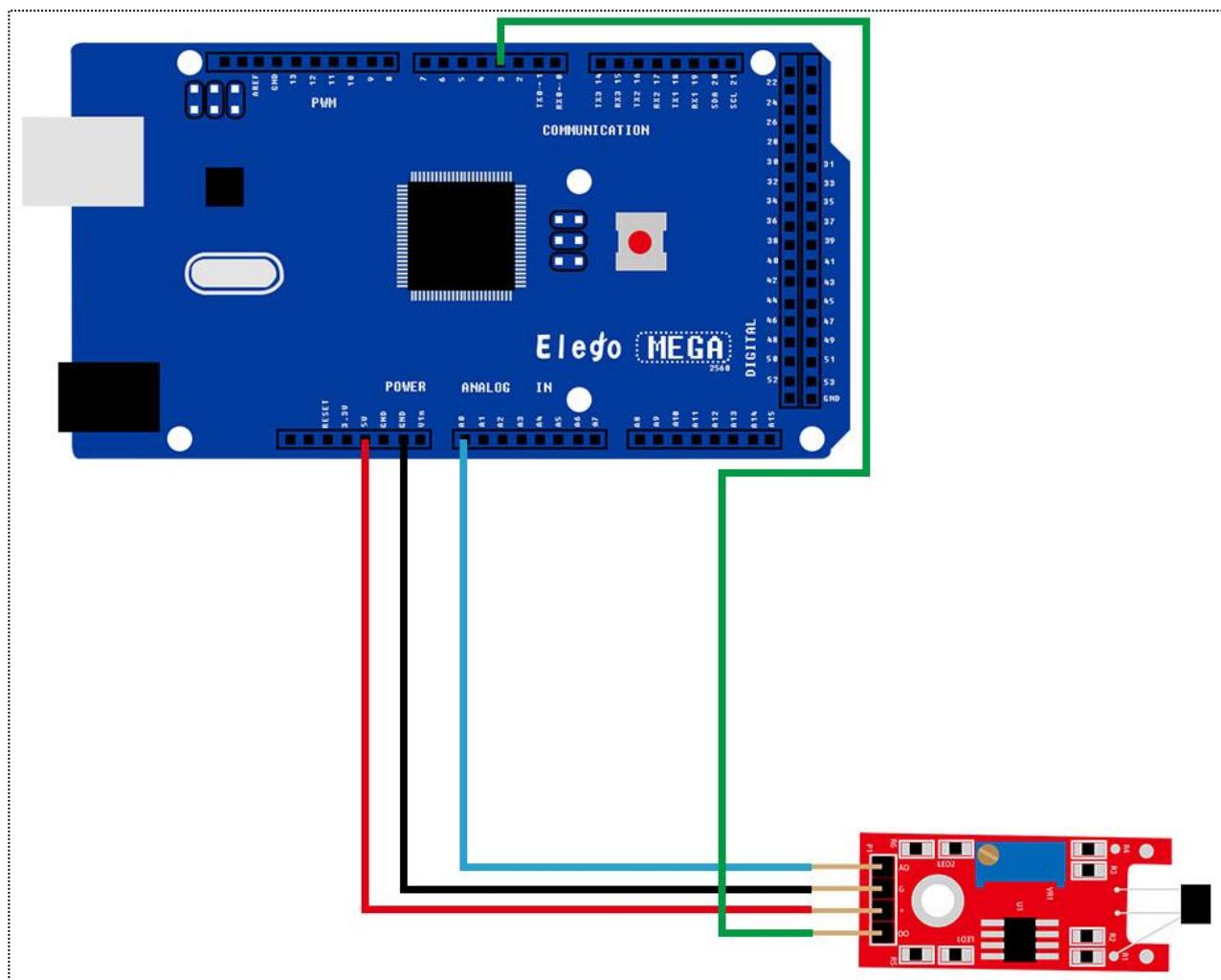
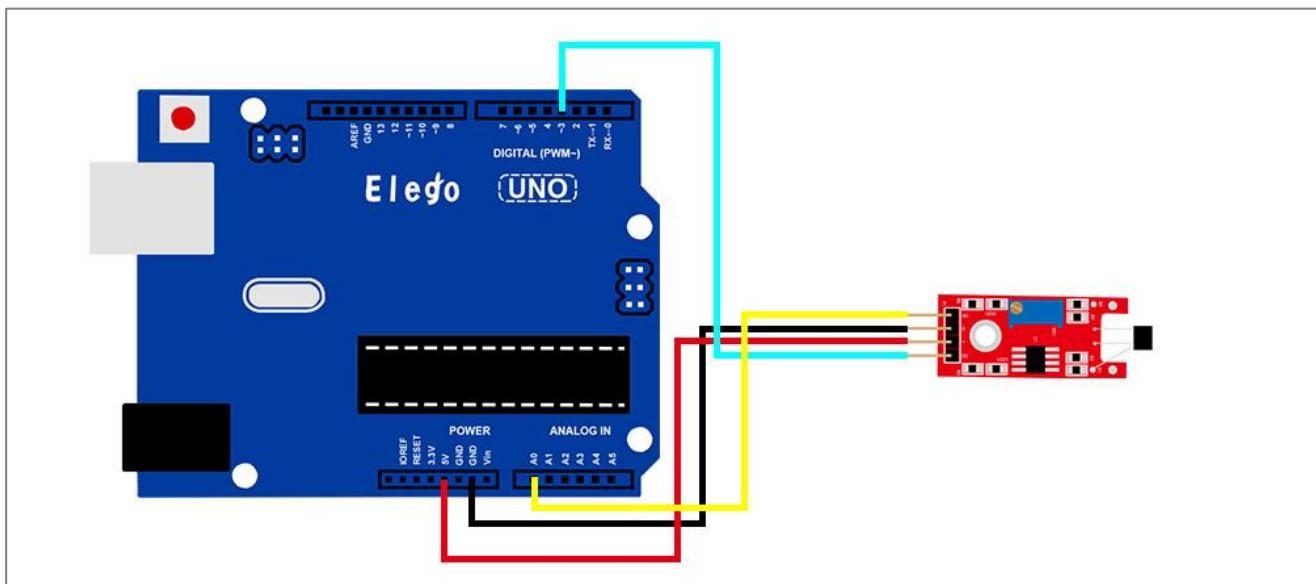
- Ratiometric linear: Output is proportional to magnetic field strength. Output sensitivity range is 2.5 – 3.75 mV per unit of Gauss.

Connection

Schematic

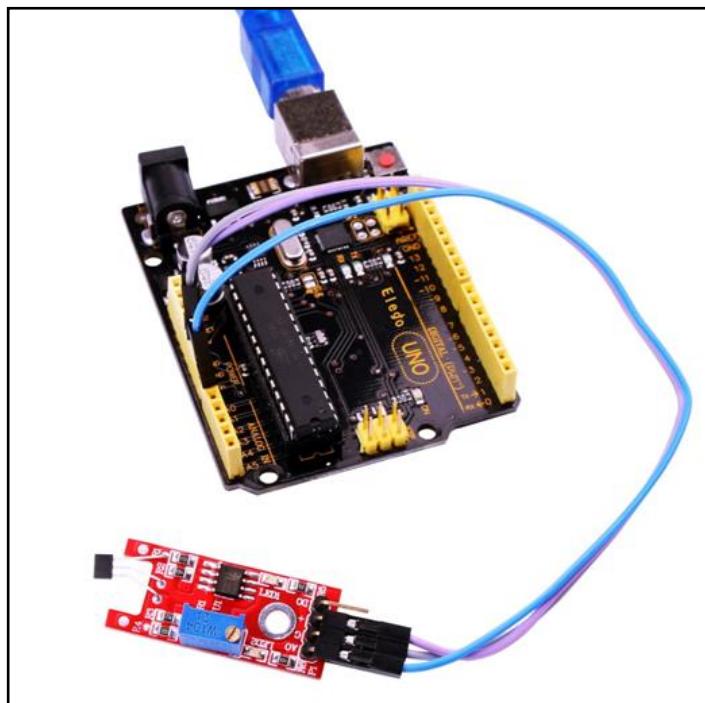


wiring diagram

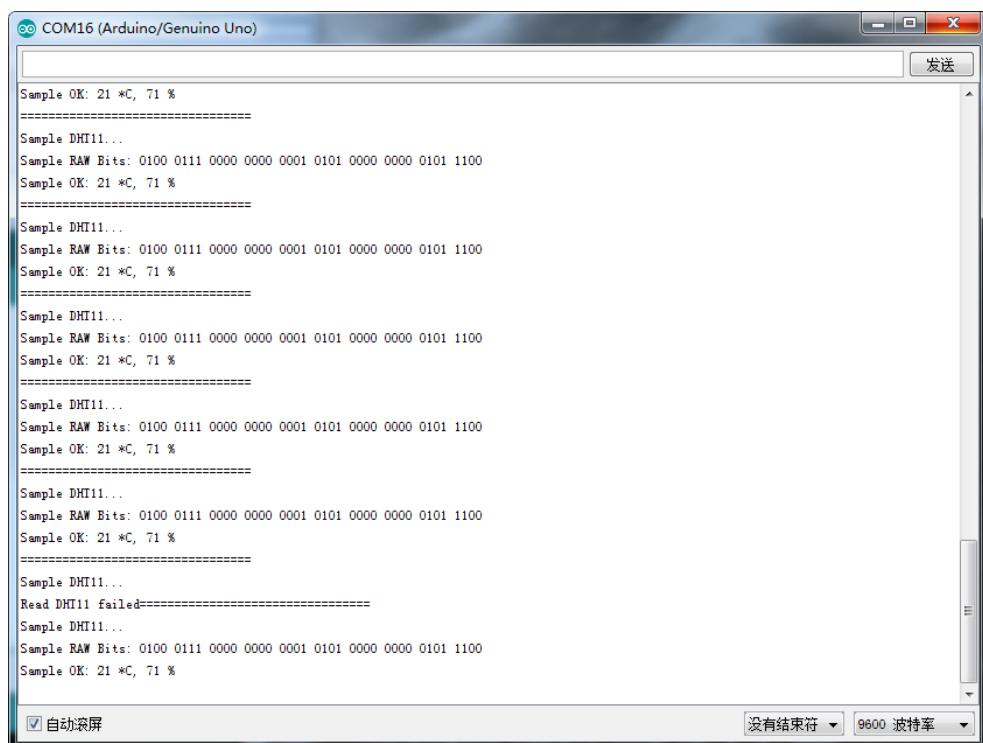


Result

For the hall sensor module, we can choose the output: digital output or analog output. In the following picture, we use the DO port to output. so we can see that if the hall sensor sensing the magnetic force, the light will turn on.



Upload the program then open the monitor, we can see the data as below:



```
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %
=====
Sample DHT11...
Read DHT11 failed=====
Sample DHT11...
Sample RAW Bits: 0100 0111 0000 0000 0001 0101 0000 0000 0101 1100
Sample OK: 21 *C, 71 %

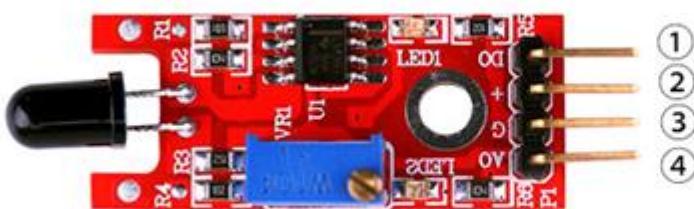
[自动滚屏] [没有结束符] [9600 波特率]
```

Lesson 17 FLAME SENSOR MODULE

Overview

In this experiment, we will learn how to use the flame sensor module.

This module is sensitive to the flame and radiation. It also can detect ordinary light source in the range of a wavelength 760nm-1100 nm. The detection distance is up to 100 cm. The Flame sensor can output digital or analog signal. It can be used as a flame alarm or in fire fighting robots.



- 1.D0:digital output
- 2.VCC: 3.3V-5V DC
- 3.GND:ground
- 4.AO:analog output

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Flame sensor module
- (x) x F-M wires

Component Introduction

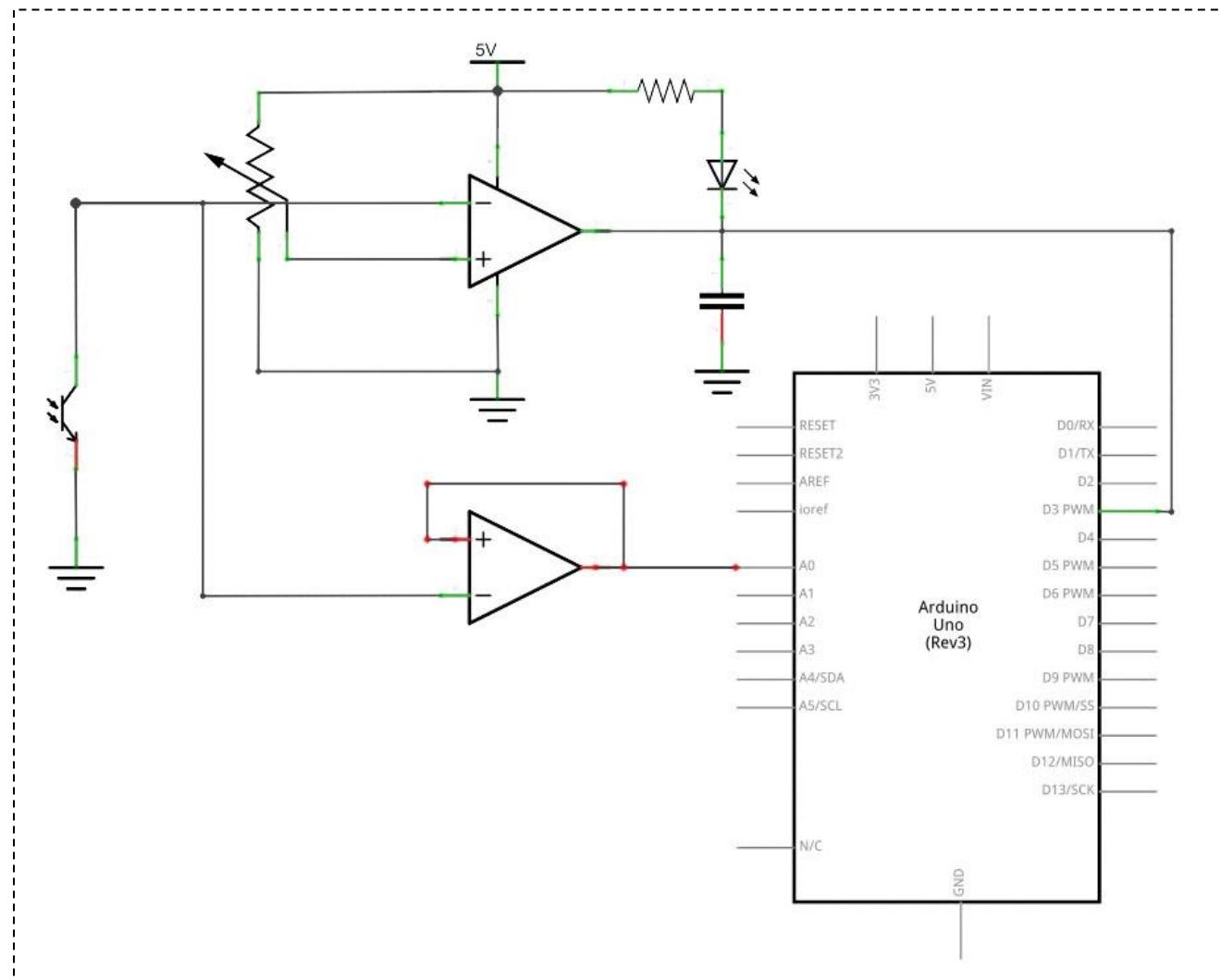
Flame sensor:

- Detects a flame or a light source of a wavelength in the range of 760nm-1100 nm
- Detection distance: 20cm (4.8V) ~ 100cm (1V)
- Detection angle about 60 degrees, it is sensitive to the flame spectrum.
- Comparator chip LM393 makes module readings stable.
- Adjustable detection range.

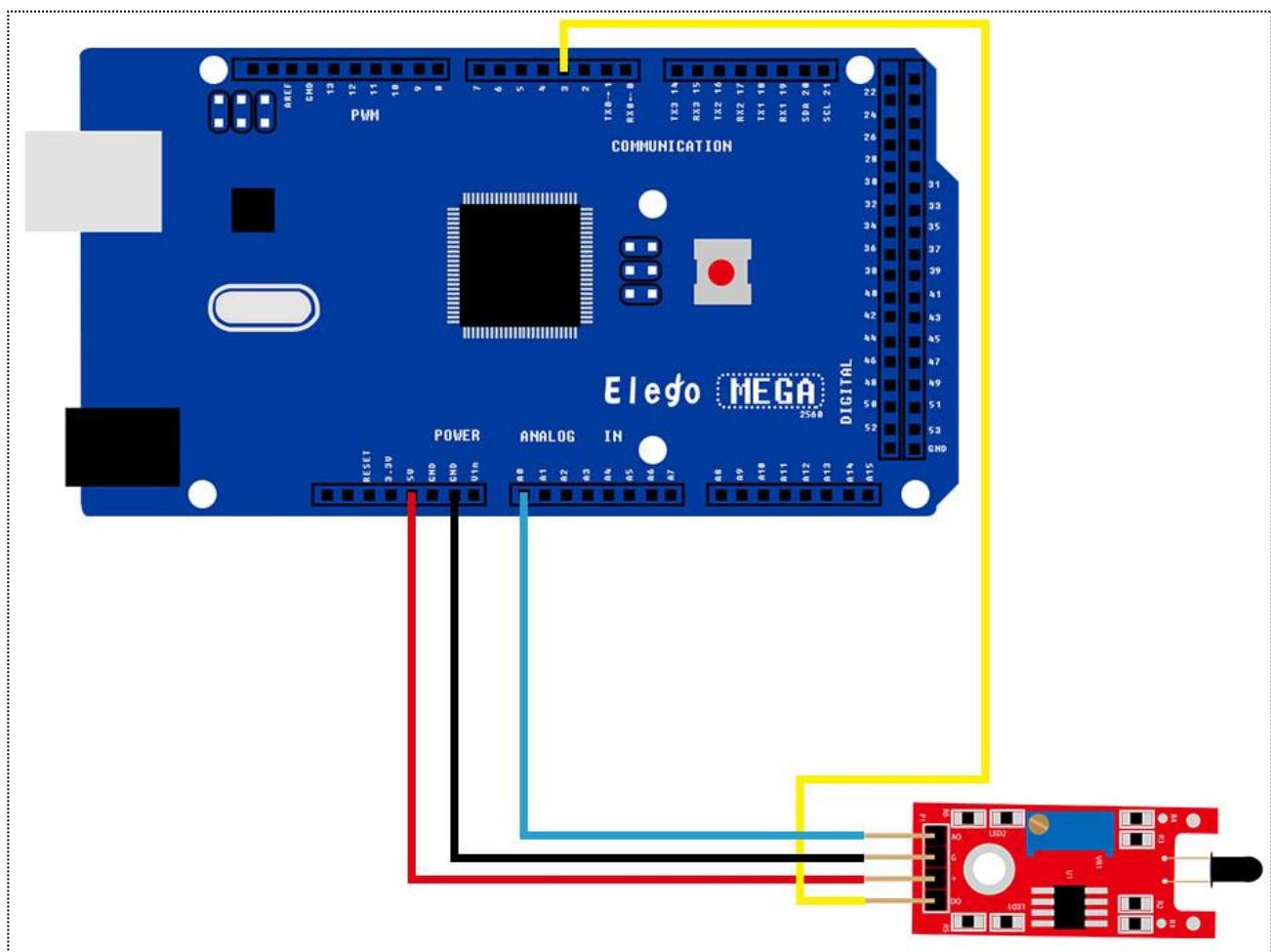
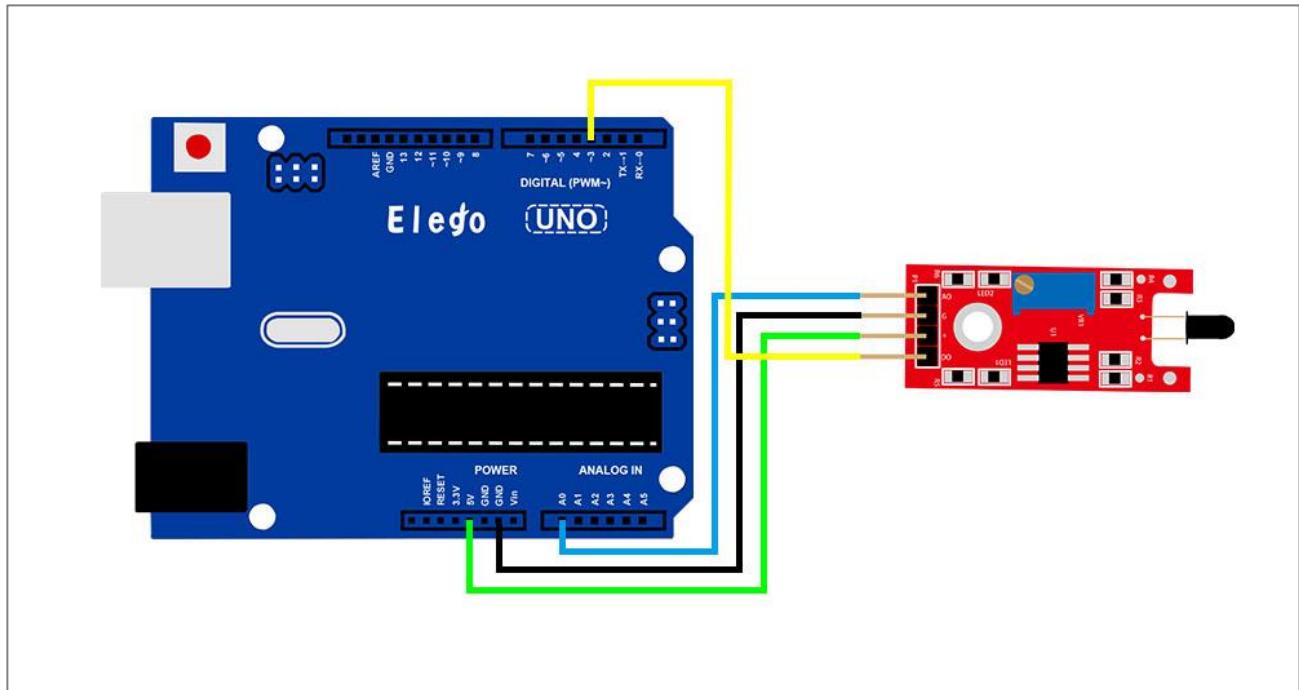
- Operating voltage 3.3V-5V
- Digital and Analog Output
 - " DO digital switch outputs (0 and 1)
 - " AO analog voltage output
- Power indicator and digital switch output indicator

Connection

Schematic

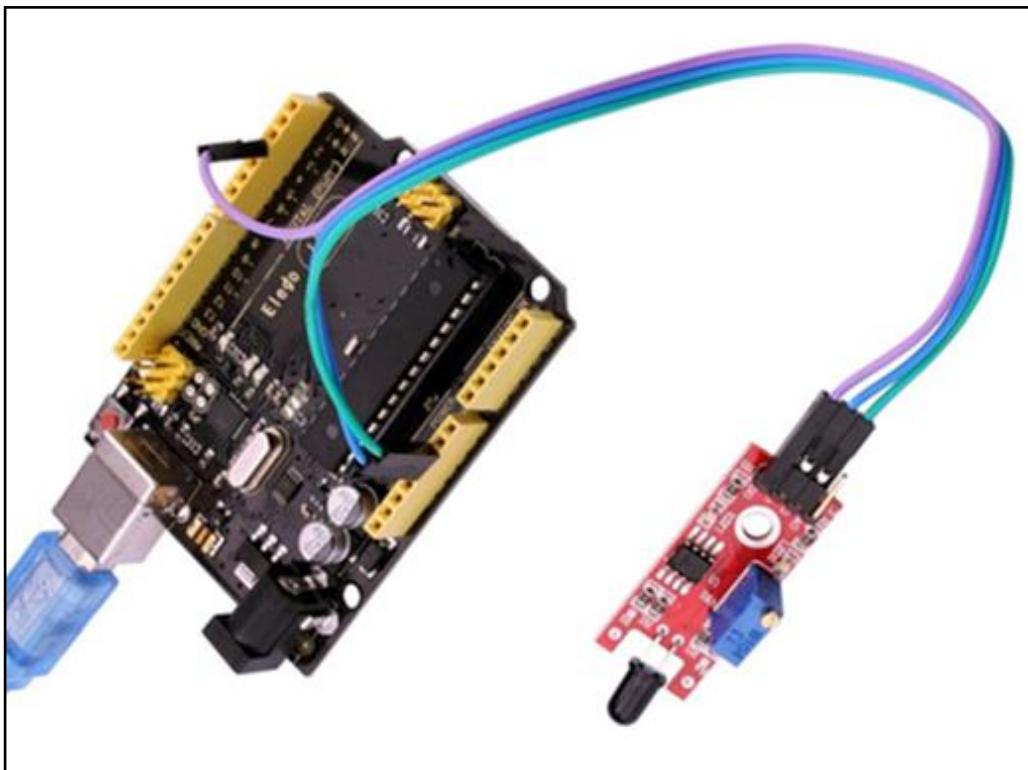


wiring diagram

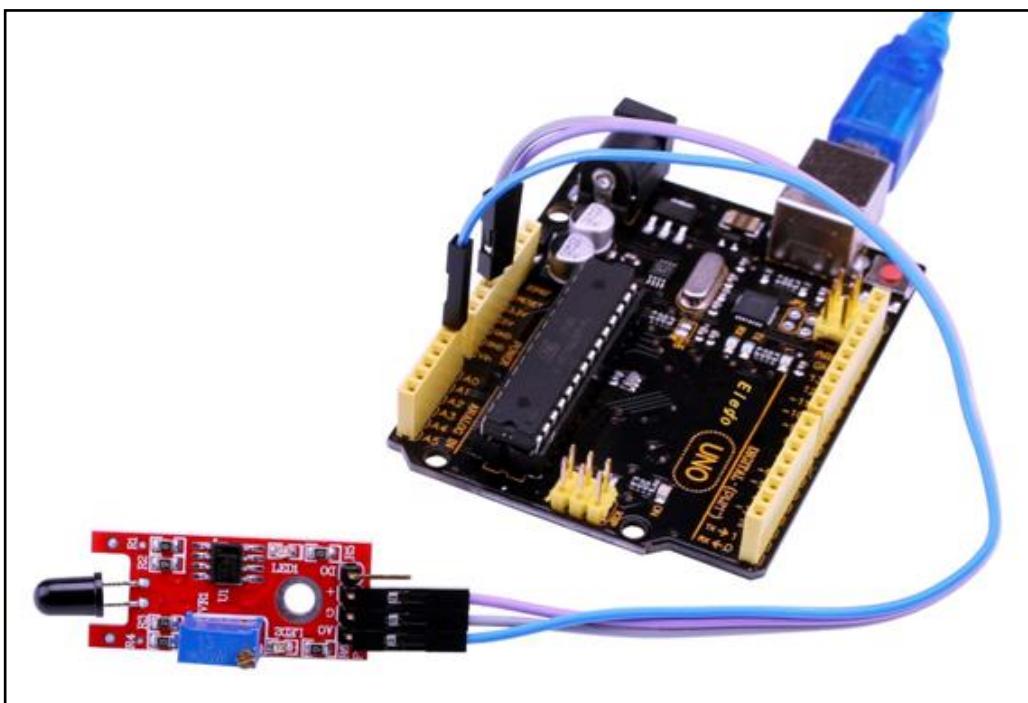


Result

For the flame sensor module, we can choose the output: digital output or analog output. In the following picture, we use the DO port to output. so we can see that if the flame sensor sensing the flame, the light will turn on.



In the following picture, we use the AO port to output. so we can see that if the flame sensor sensing the flame, the module will output a data which reflect the strength of the flame. The number is from 0 to 1023.



Upload the program then open the monitor, we can see the data as below:

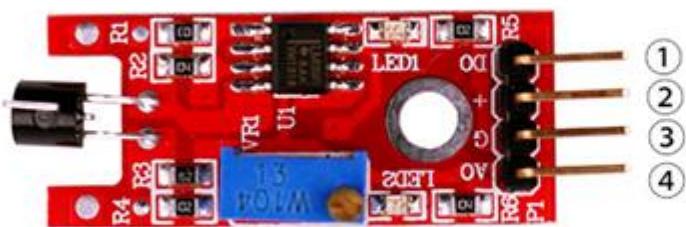
```
856  
804  
807  
965  
974  
989  
987  
817  
850  
867  
835  
942  
944  
811  
64  
513  
333  
342  
307  
356  
293  
302  
284  
323  
332  
896  
957
```

自动滚屏 换行符 9600 波特率

Lesson 18 MENTAL TOUCH MODULE

Overview

In this experiment, we will learn how to use the mental touch module.



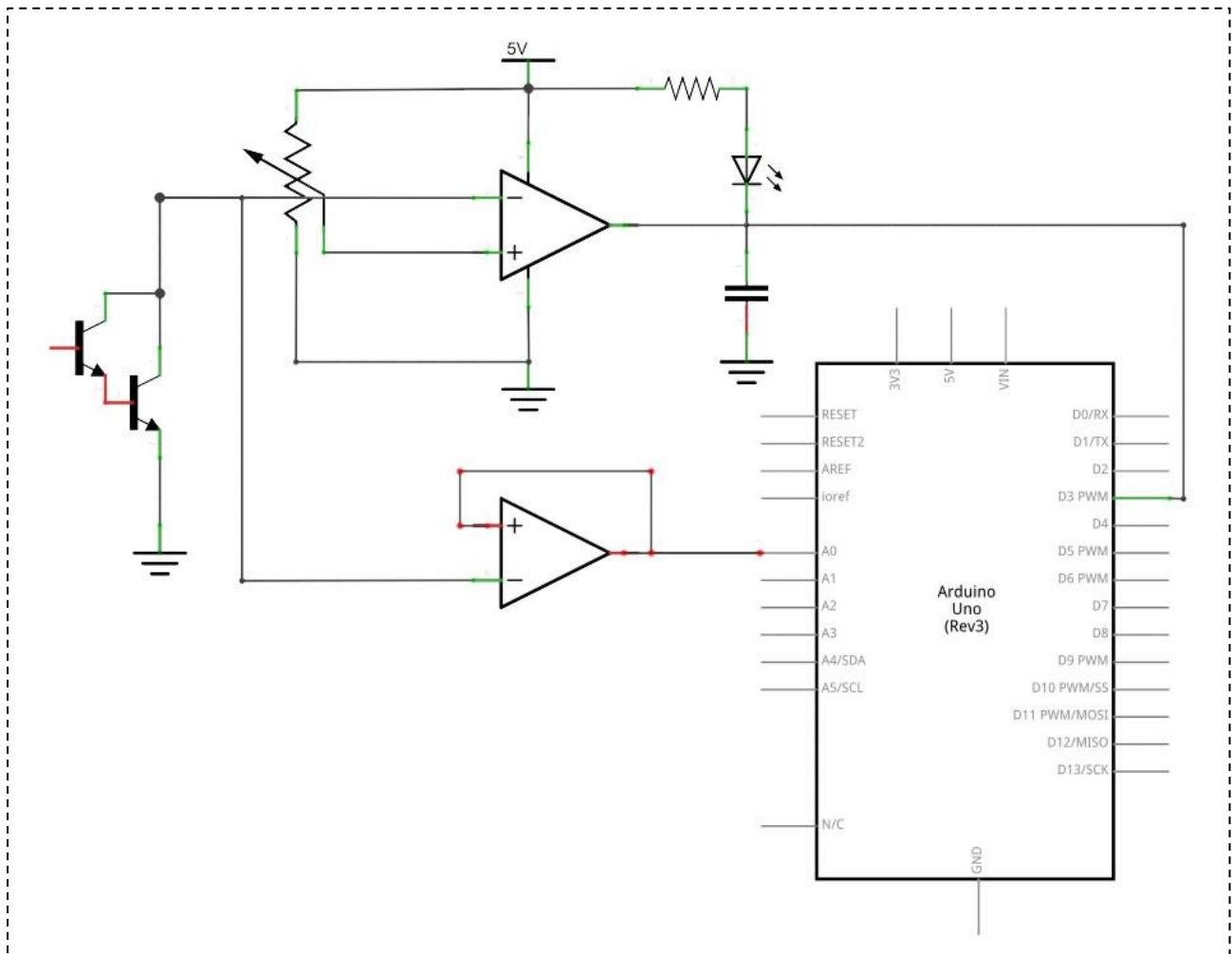
- 1.D0:digital output
- 2.VCC: 3.3V-5V DC
- 3.GND:ground
- 4.AO:analog output

Component Required:

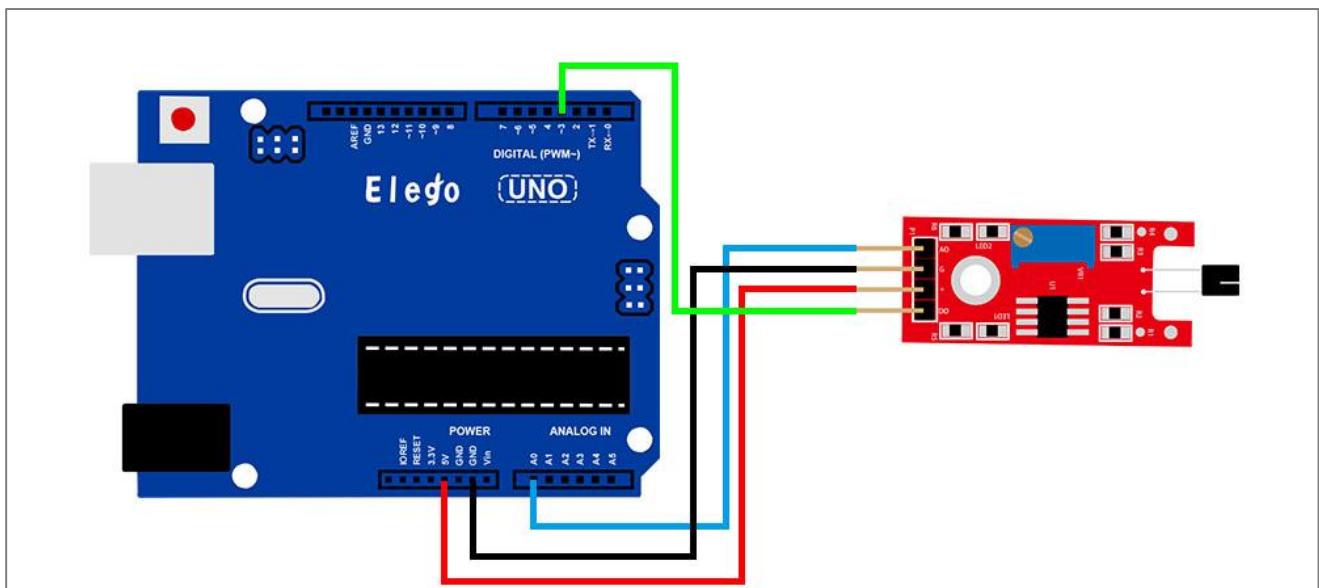
- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Mental touch module
- (x) x F-M wires

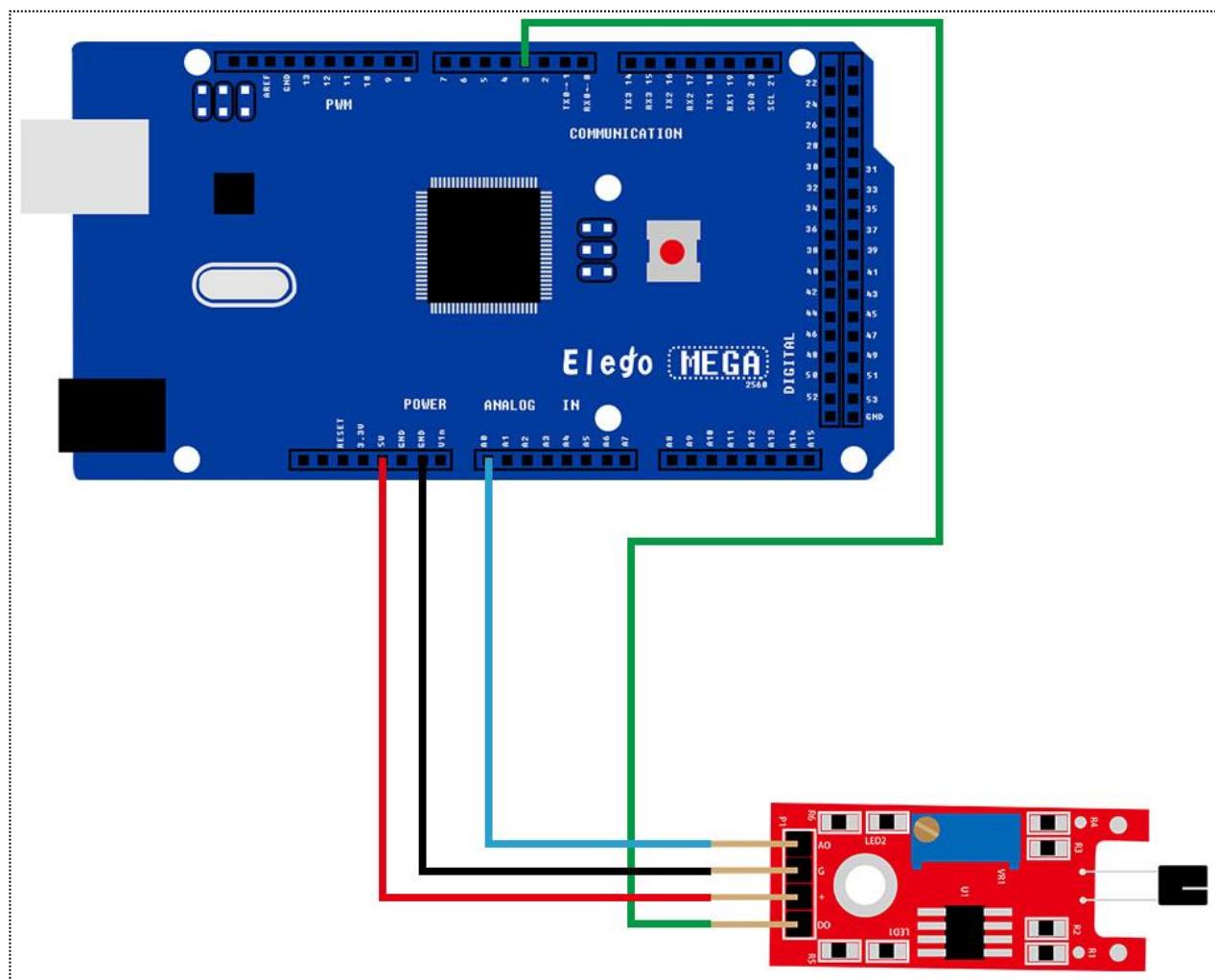
Connection

Schematic



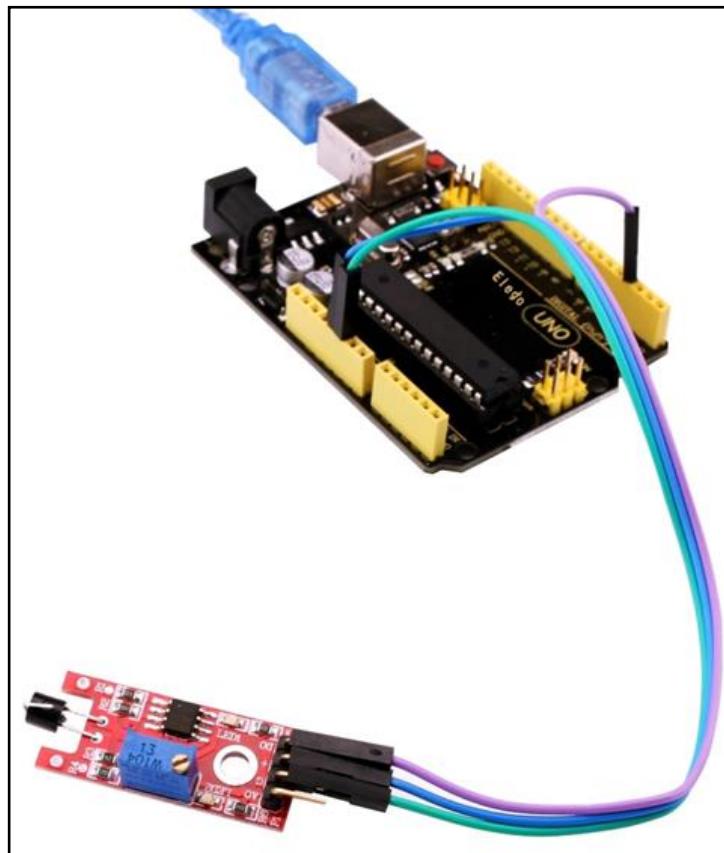
wiring diagram



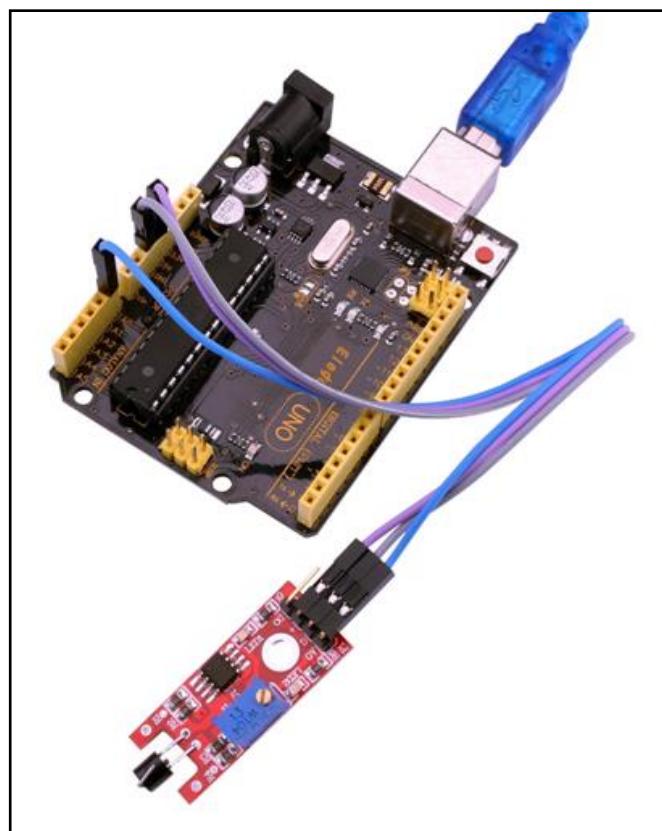


Result

For the mental touch module, we can choose the output: digital output or analog output. In the following picture, we use the DO port to output. so we can see that if the sensor sensing, the light will turn on.



In the following picture, we use the AO port to output. so we can see that if the sensor sensing, the module will output a data. The number is from 0 to 1023.



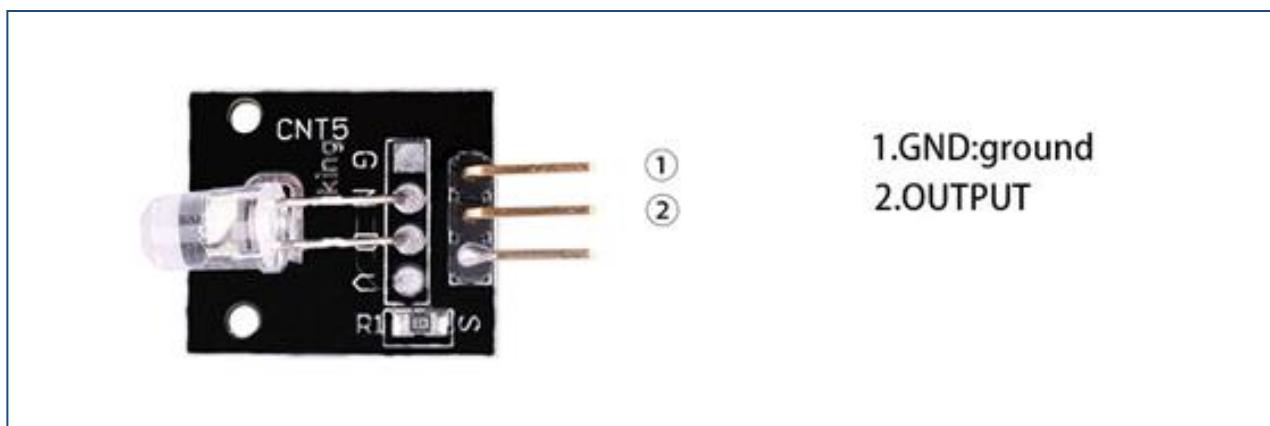
Upload the program then open the monitor, we can see the data as below:

```
1020
1023
147
154
1023
1023
146
1018
1020
1023
1023
146
1023
146
1016
1023
1023
145
1023
146
1016
1019
1021
145
1023
144
1020
```

Lesson 19 7 COLOR FLASH LED MODULE

Overview

In this experiment, we will learn how to use the 7 color flash LED module.



Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x 7 color flash led module
- (x) x F-M wires

Component Introduction

7 color flash led:

7 color flashing LED module automatically uses 5mm round high-brightness light-emitting diode which has the following characteristics:

Product Type: LED

Product Model: YB-3120B4PnYG-PM

Shape: Round LED 5mm DIP type

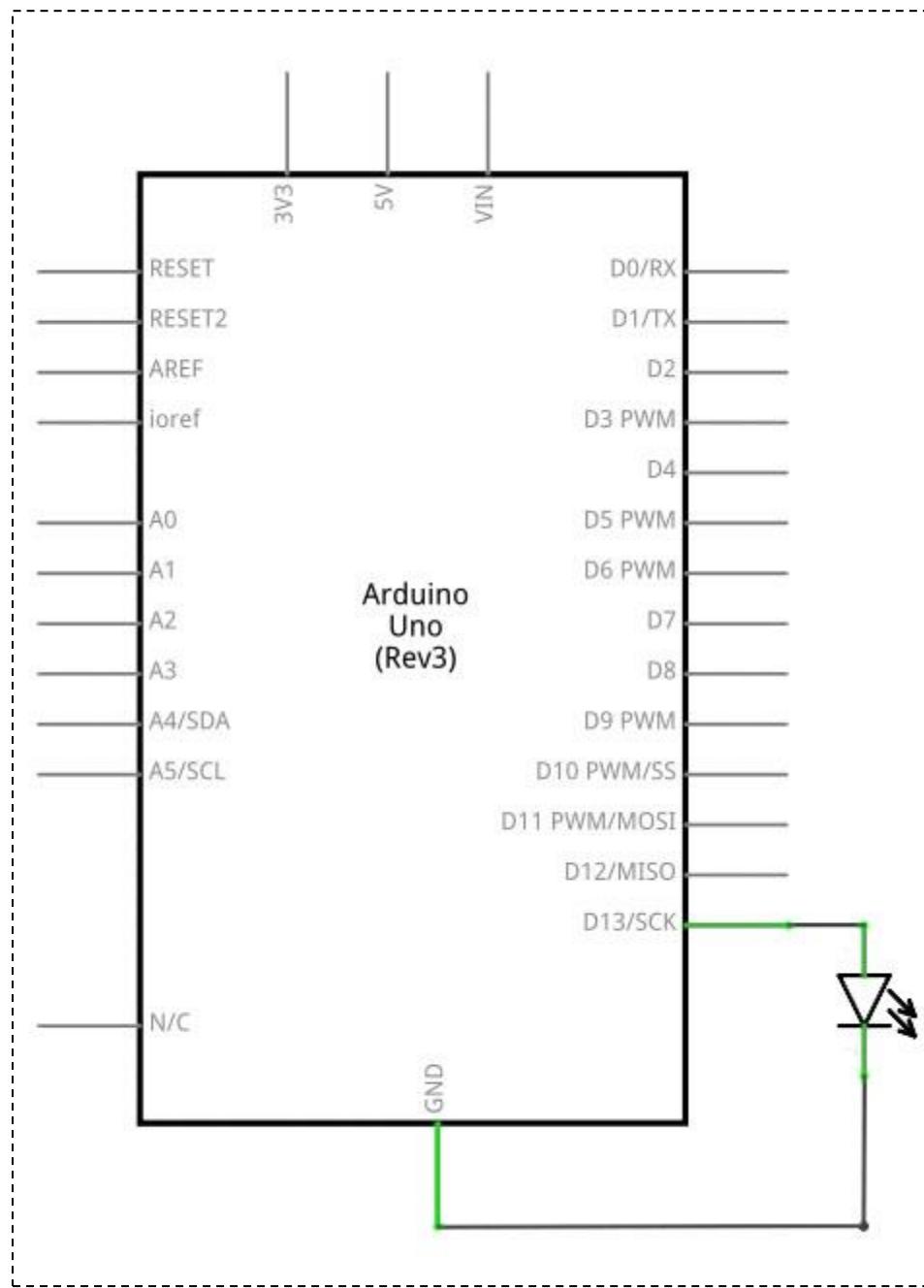
Color: pink yellow green (high brightness)

Lens type: white mist

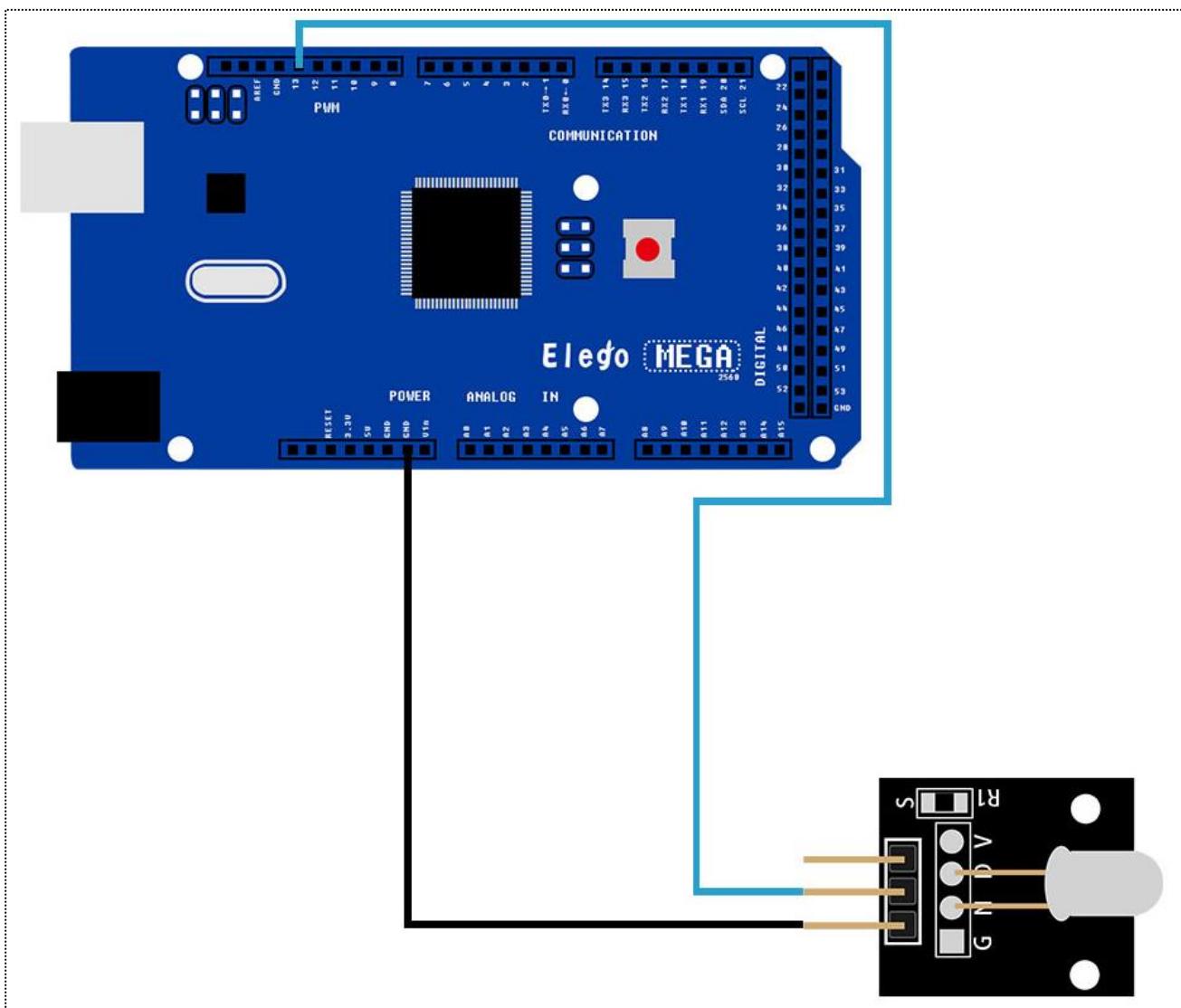
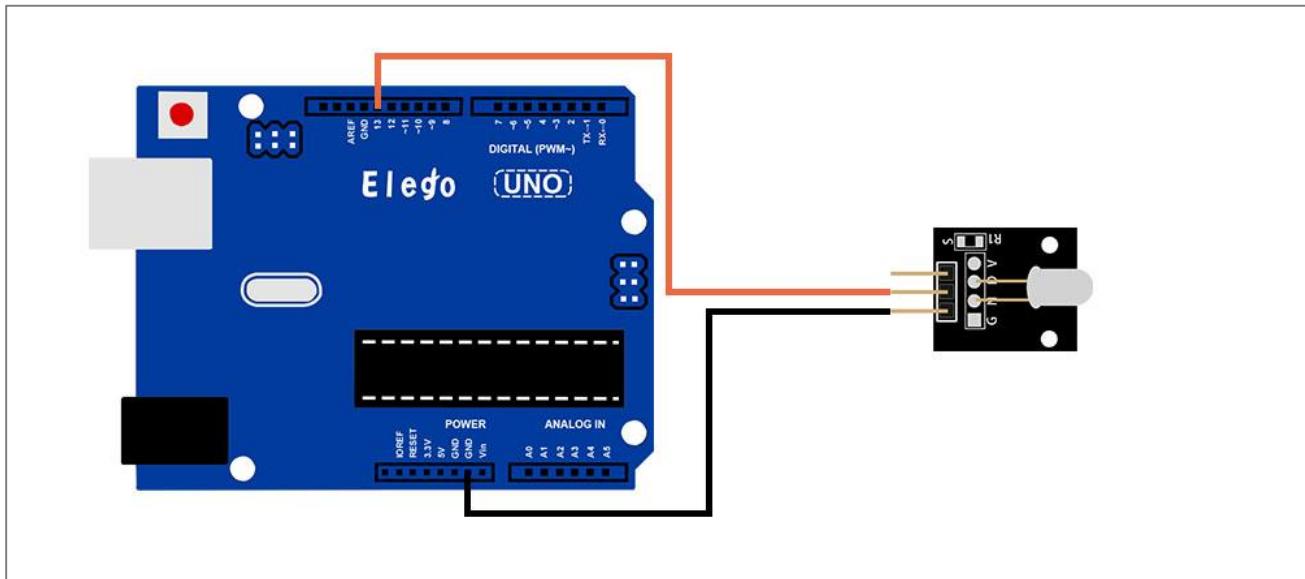
Standard Forward Voltage :3.0-4 .5 V

Connection

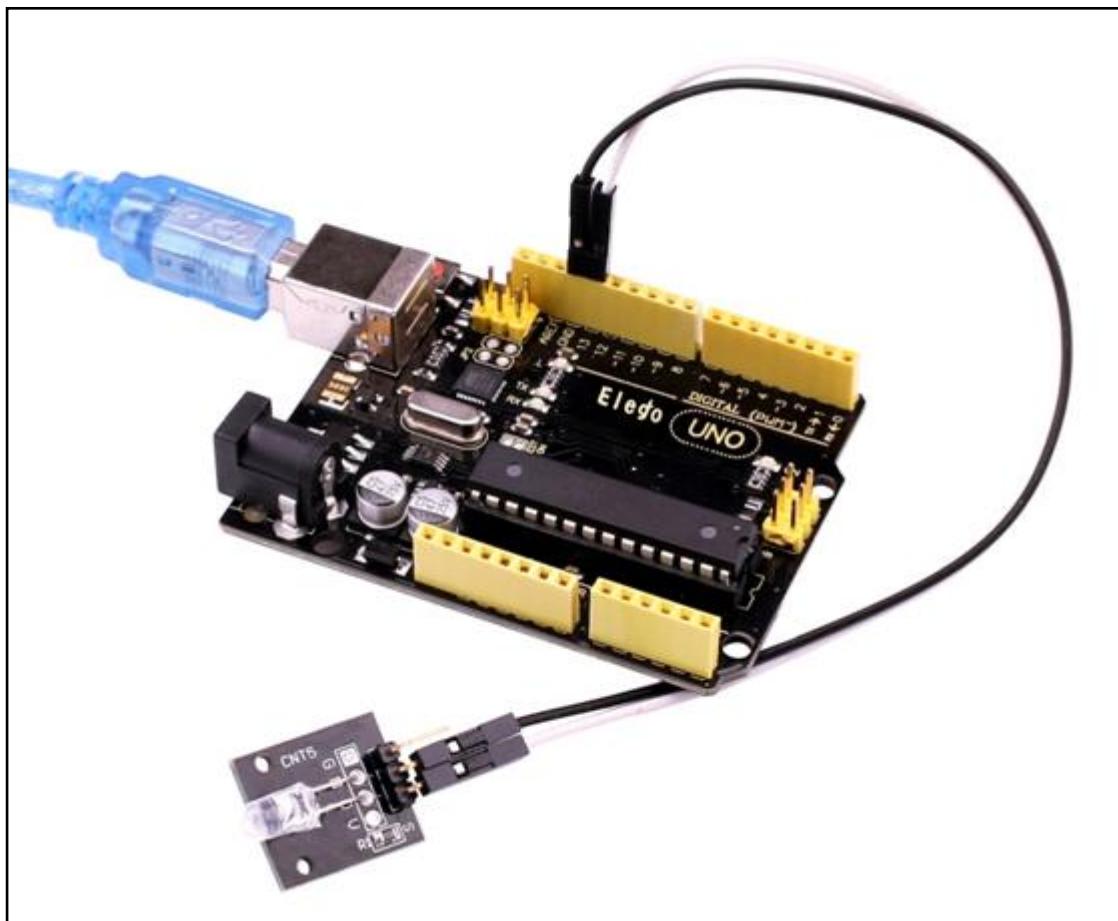
Schematic



wiring diagram



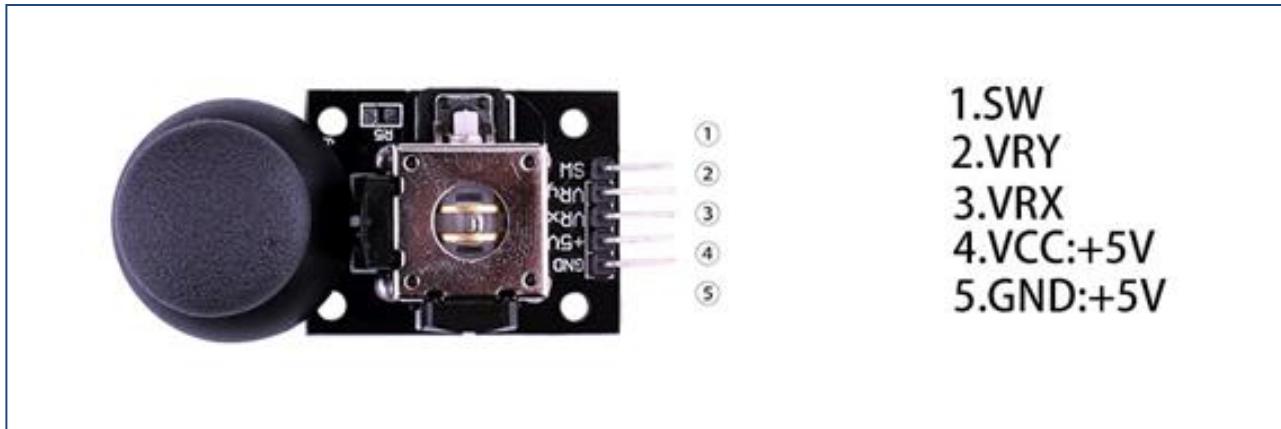
Result



Lesson 20 JOYSTICK MODULE

Overview

Just like a joystick on game console, you can control x, y and z dimensions input by this joystick module. It can be considered as combination of potentiometers and one button. Data type of the x, y dimensions are analog input signals and z dimension is digital input signal, thus the x and y ports connect to analog pins of Sensor Shield, while z port connects to digital pin.



Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Joystick module
- (x) x F-M wires

Component Introduction

Joystick sensor:

Lots of robot projects need joystick. This module provides an affordable solution. By simply connecting to two analog inputs, the robot is at your commands with X, Y control. It also has a switch that is connected to a digital pin. This joystick module can be easily connected to Arduino by IO Shield. This module is for Arduino (V5) with cables supplied.

Specification

Supply Voltage: 3.3V to 5V

Interface: Analog x2, Digital x1

Size: 40*28mm

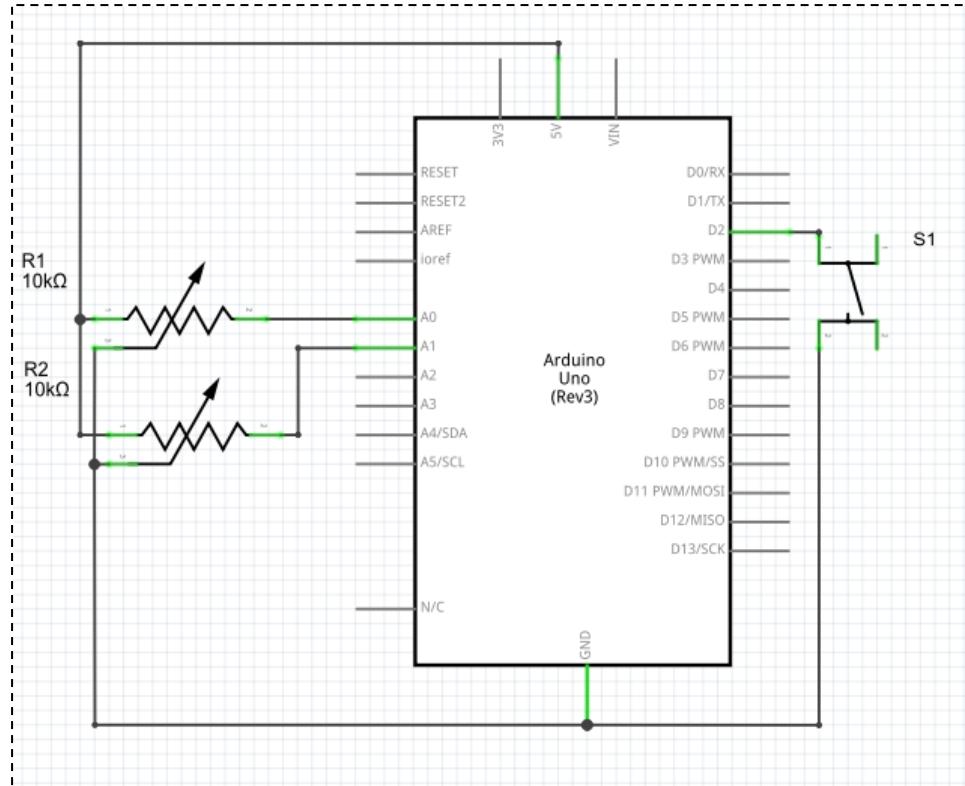
Weight: 12g

The module has 5 pins: Vcc, Ground, X, Y, Key. Note that the labels on yours may be slightly different, depending on where you got the module from. The thumb stick is analog and should provide more accurate readings than simple 'directional' joysticks tact use some forms of buttons, or mechanical switches. Additionally, you can press the joystick down (rather hard on mine) to activate a 'press to select' push-button.

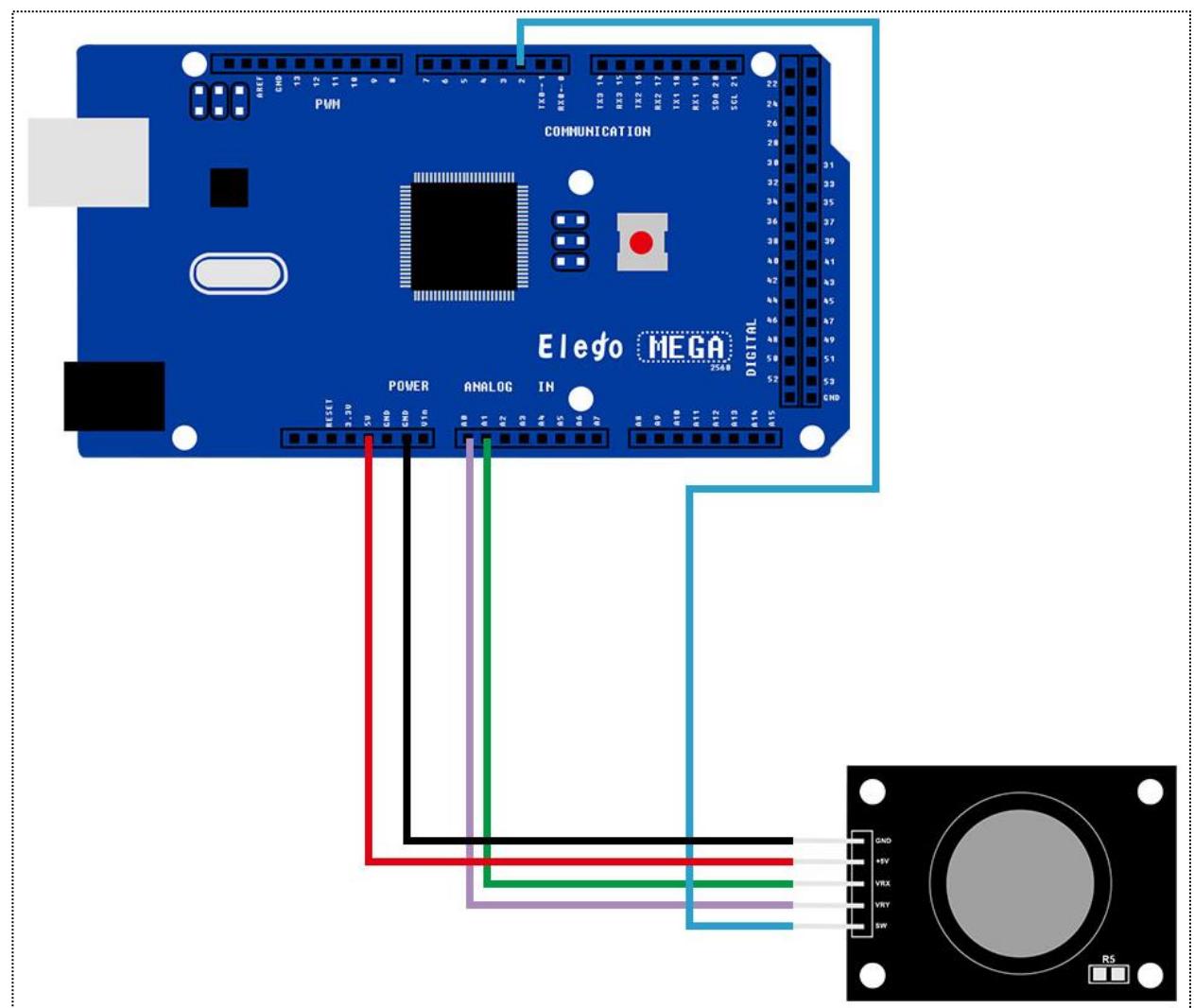
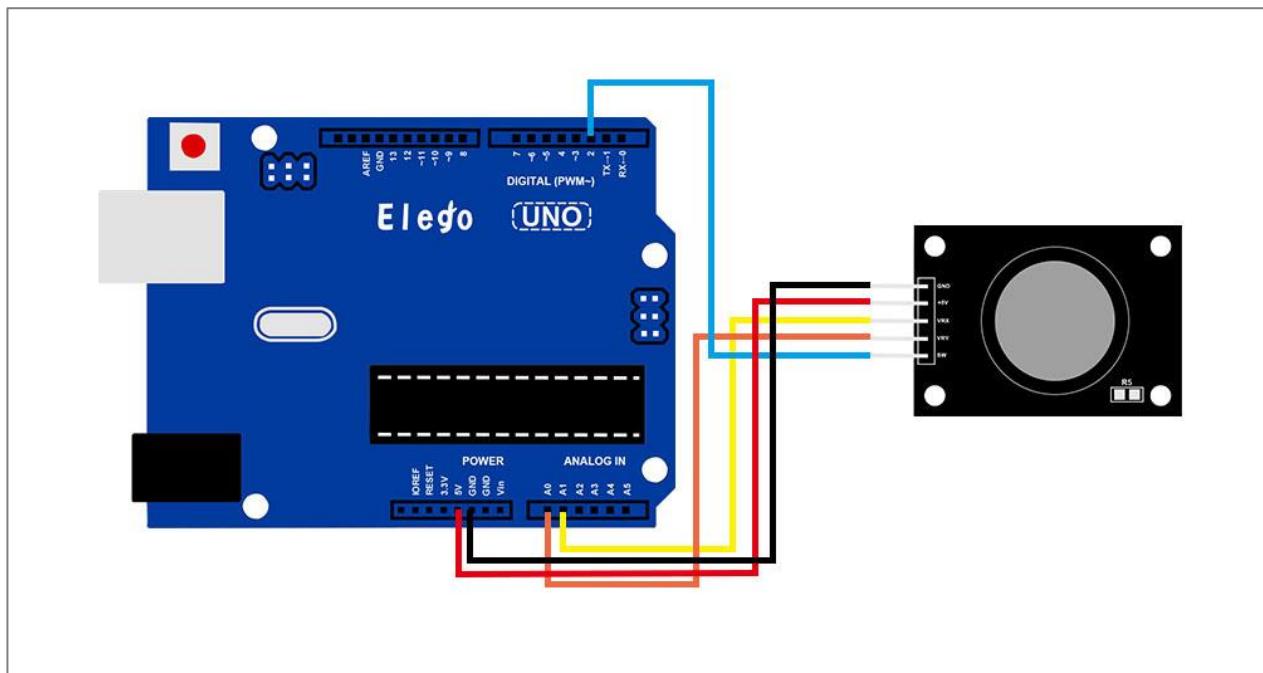
We have to use analog Arduino pins to read the data from the X/Y pins, and a digital pin to read the button. The Key pin is connected to ground, when the joystick is pressed down, and is floating otherwise. To get stable readings from the Key /Select pin, it needs to be connected to Vcc via a pull-up resistor. The built in resistors on the Arduino digital pins can be used. For a tutorial on how to activate the pull-up resistors for Arduino pins, configured as inputs

Connection

Schematic



wiring diagram

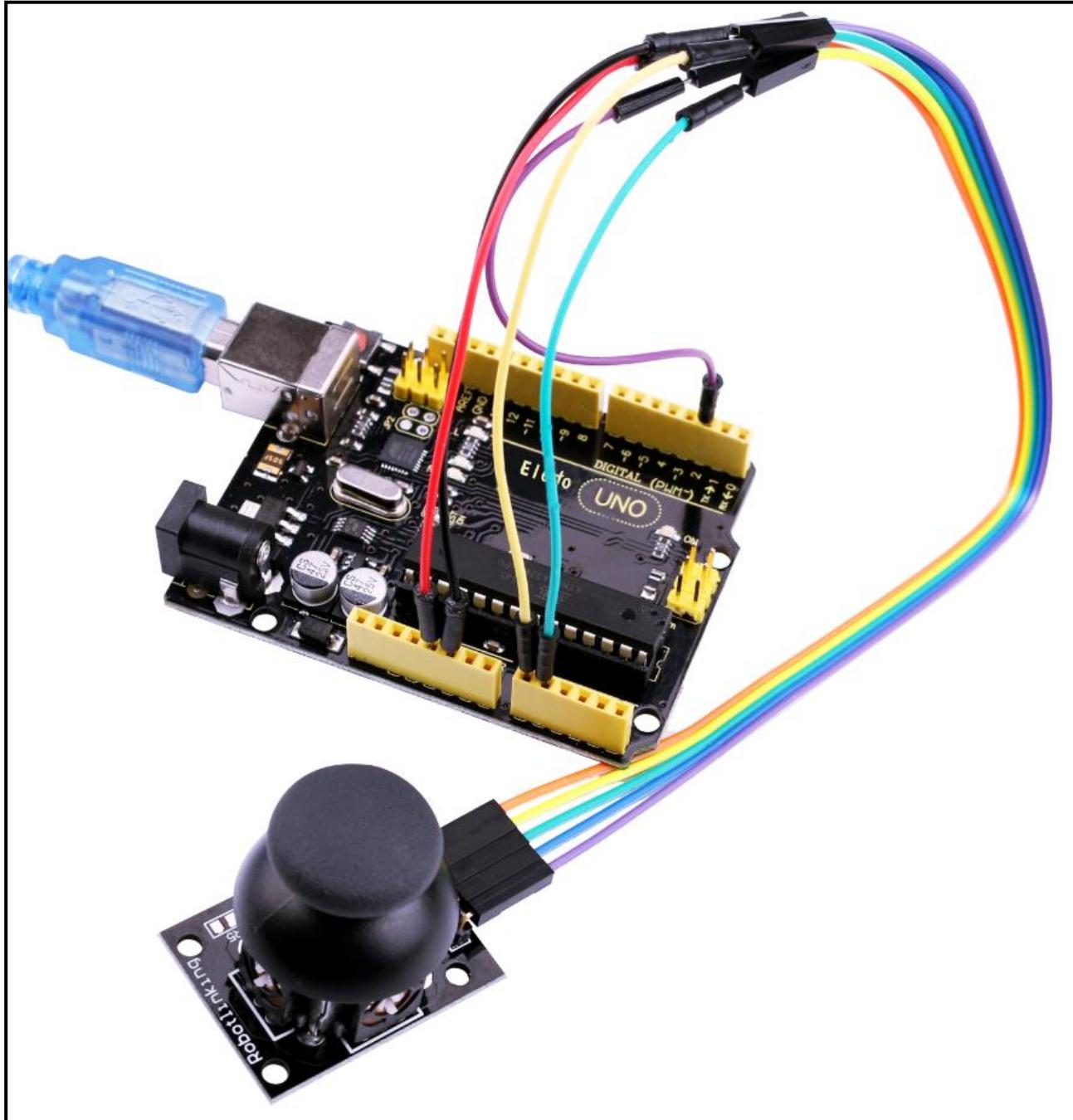


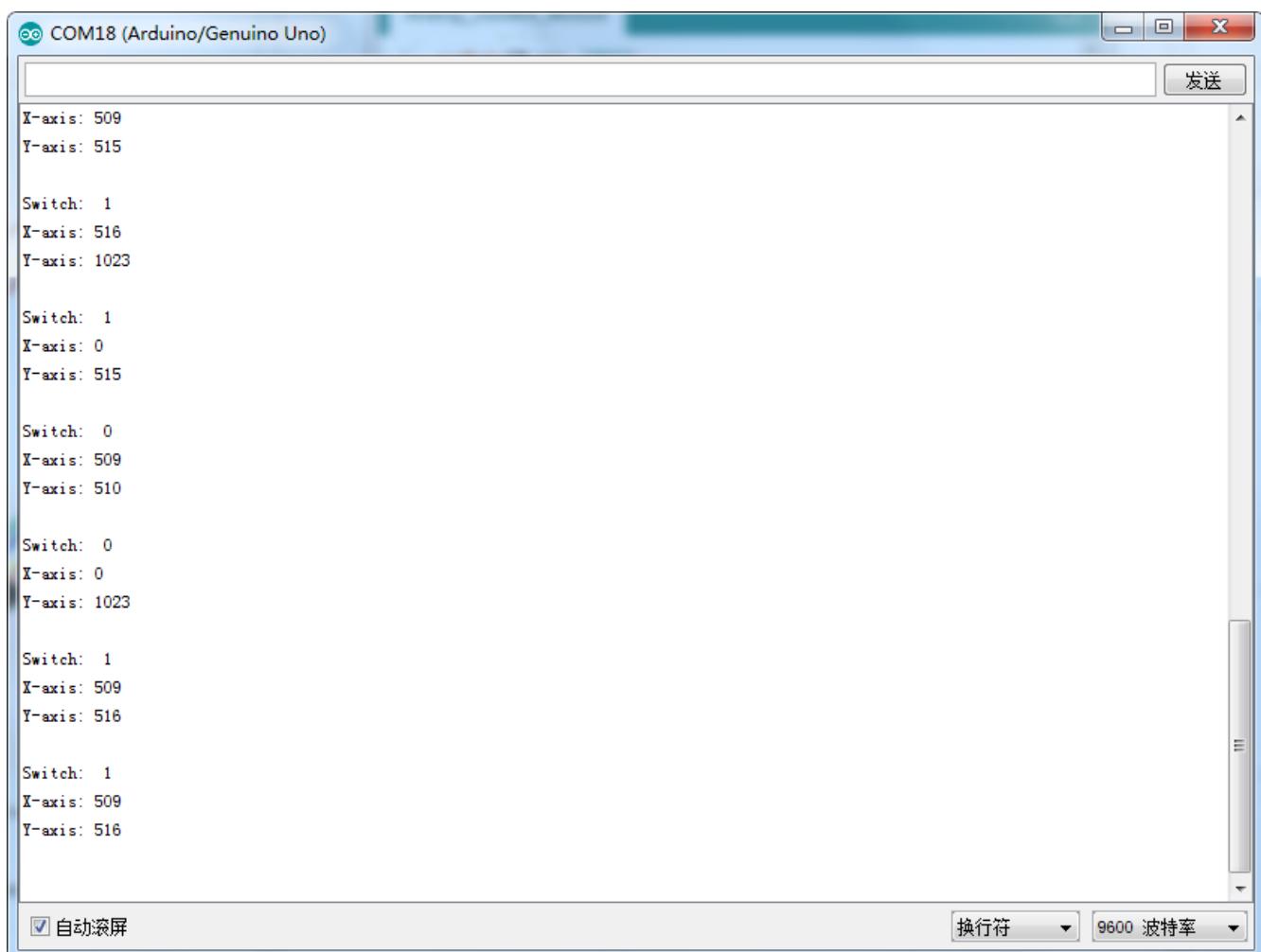
Result

Analog joysticks are basically potentiometers so they return analog values.

When the joystick is in the resting position or middle, it should return a value of about 512.

The range of values go from 0 to 1024.





```
X-axis: 509  
Y-axis: 515  
  
Switch: 1  
X-axis: 516  
Y-axis: 1023  
  
Switch: 1  
X-axis: 0  
Y-axis: 515  
  
Switch: 0  
X-axis: 509  
Y-axis: 510  
  
Switch: 0  
X-axis: 0  
Y-axis: 1023  
  
Switch: 1  
X-axis: 509  
Y-axis: 516  
  
Switch: 1  
X-axis: 509  
Y-axis: 516
```

Lesson 21 TRACKING MODULE

Overview

In this experiment, we will learn how to use the tracking module and avoidance module. Infrared obstacle avoidance sensor is designed for the design of a wheeled robot obstacle avoidance sensor distance adjustable. This ambient light sensor Adaptable, high precision, having a pair of infrared transmitter and receiver, transmitter tubes emit a certain frequency of infrared, When detecting the direction of an obstacle (reflector), the infrared receiver tube receiver is reflected back, when the indicator is lit, Through the circuit, the signal output interface output digital signal that can be detected by means of potentiometer knob to adjust the distance, the effective distance From 2 ~ 40cm, working voltage of 3.3V-5V, operating voltage range as broad, relatively large fluctuations in the power supply voltage of the situation Stable condition and still work for a variety of microcontrollers, Arduino controller, BS2 controller, attached to the robot that can sense changes in the ir surroundings



- 1.OUTPUT
- 2.VCC: 3.3V-5V DC
- 3.GND:ground

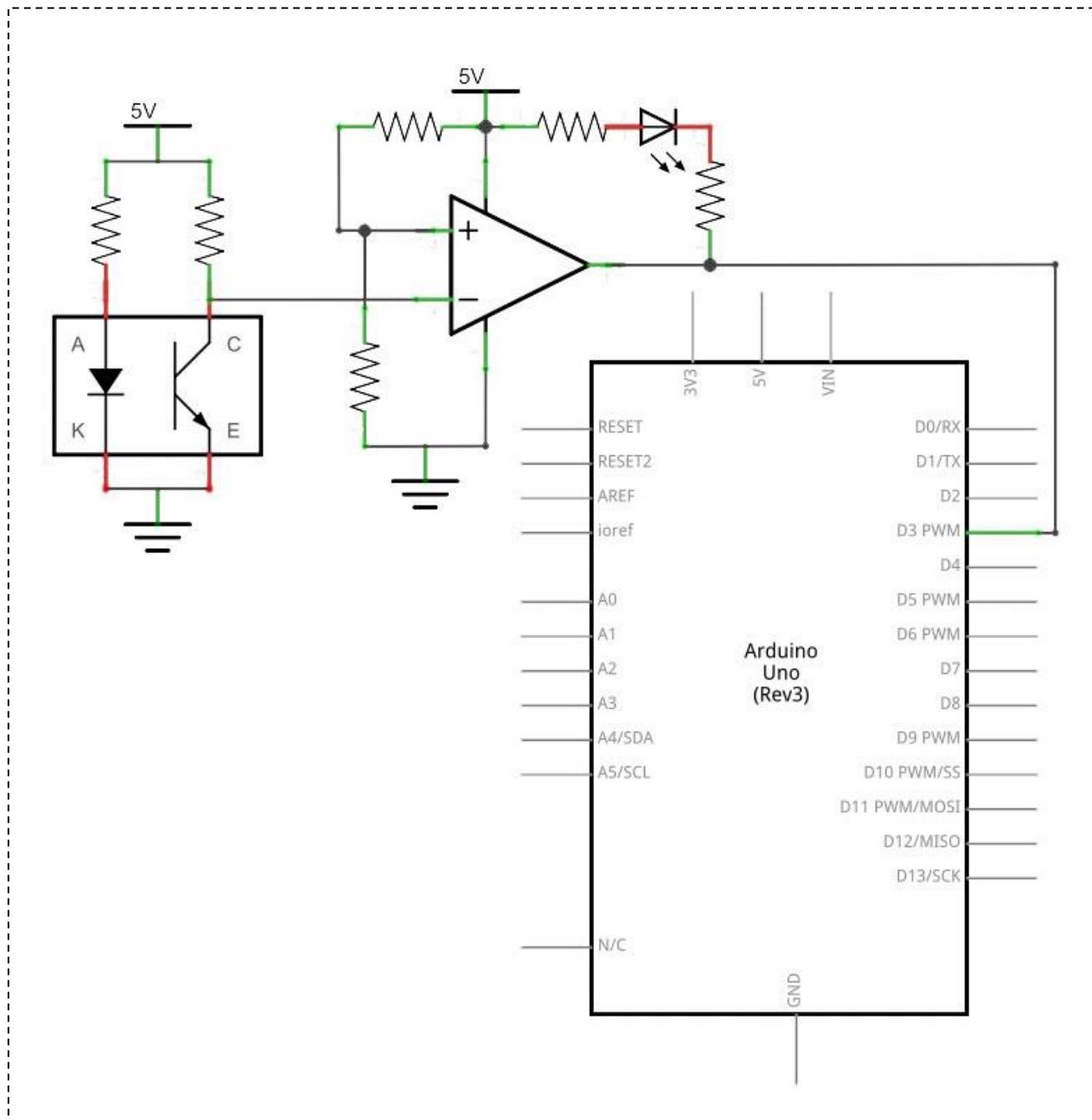
Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Tracking module
- (x) x F-M wires

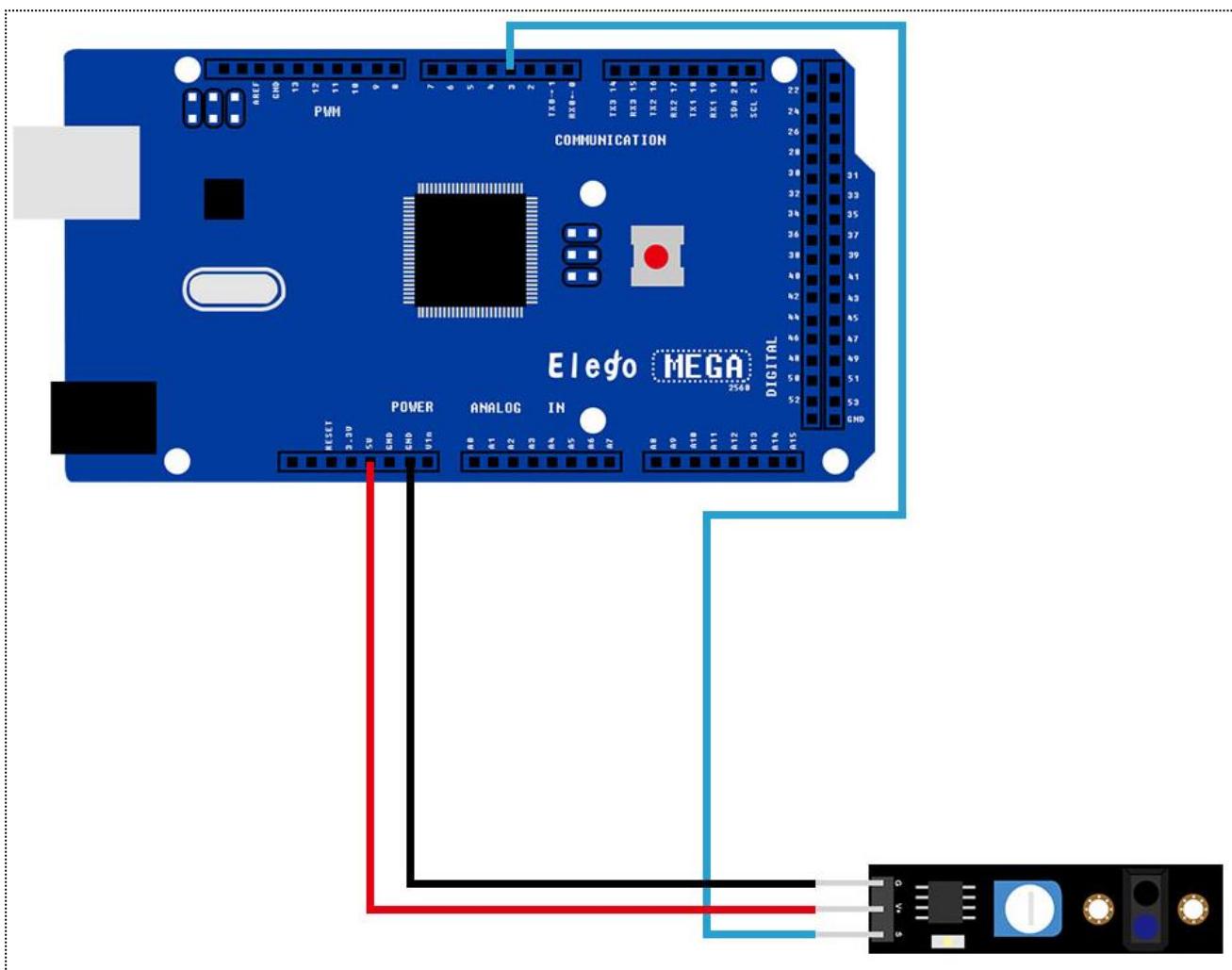
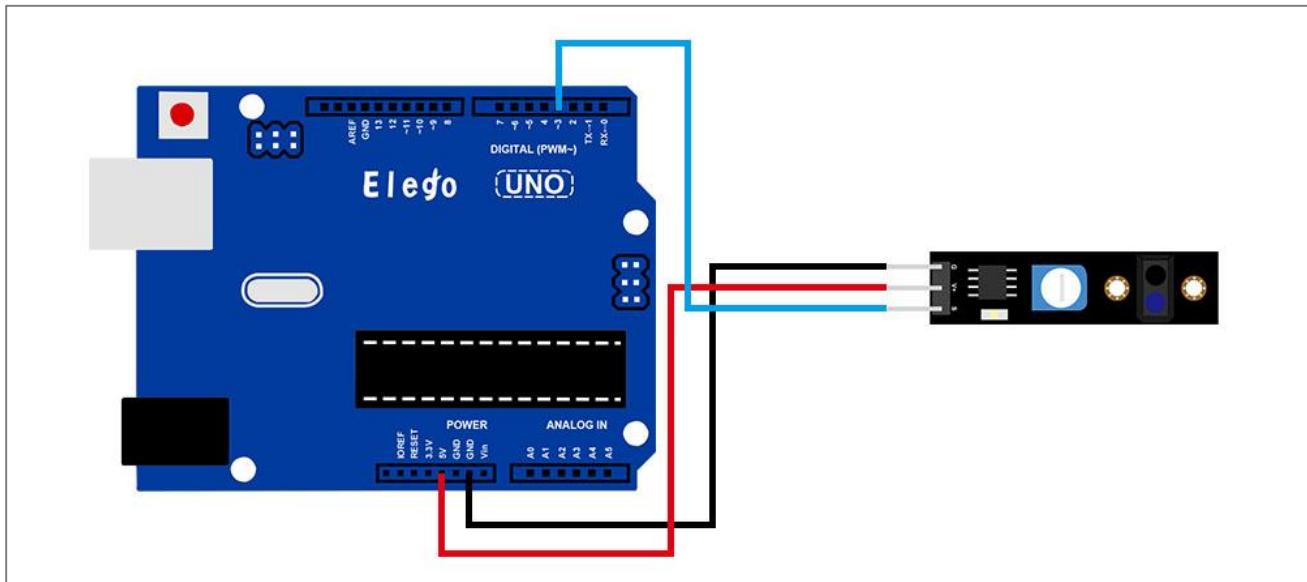
Component Introduction

Connection

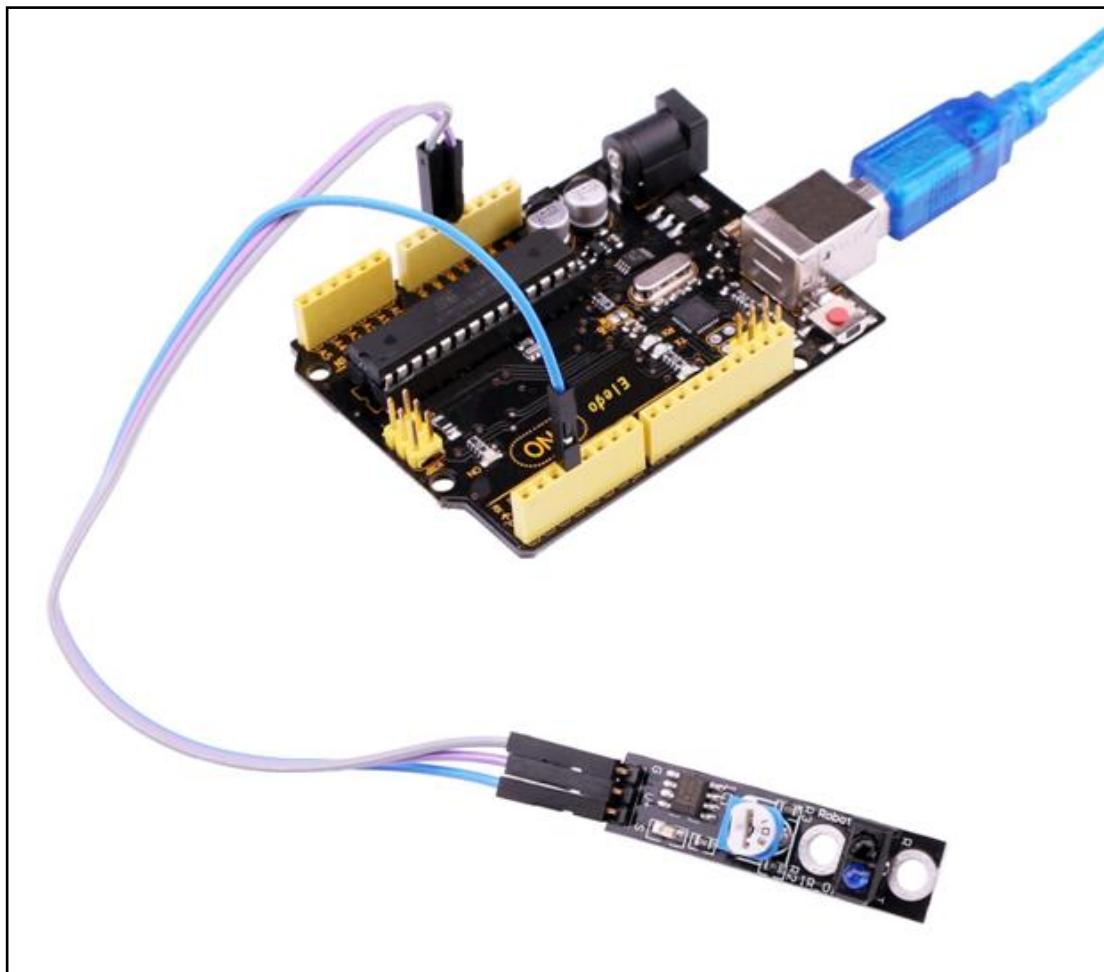
Schematic



wiring diagram



Result

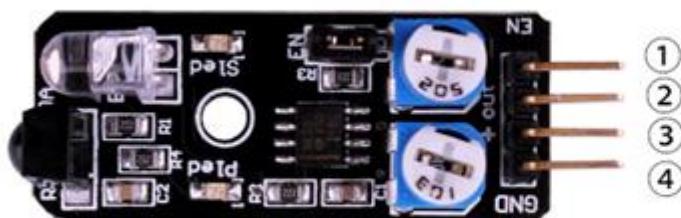


Upload the program then open the monitor, we can see the data.

Lesson 22 Infrared 38KHz obstacle avoidance module

Overview

Non logic chip oscillation frequency regulation 38KHz detection circuit. This infrared 38KHZ obstacle avoidance module can completely block the traditional photoelectric obstacle detection distance and a single normally open or normally closed signal output function.



- 1.EN
- 2.OUTPUT
- 3.VCC: 3.3V-5V DC
- 4.GND:ground

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Obstacle avoidance module
- (x) x F-M wires

Component Introduction

Obstacle avoidance module:

Specifications

Working voltage: DC 3.3V-5V

Working current: $\geq 20\text{mA}$

Operating temperature: $-10^{\circ}\text{C} - +50^{\circ}\text{C}$

detection distance :2-40cm

IO Interface: 4-wire interfaces (- / + / S / EN)

Output signal: TTL level (low level there is an obstacle, no obstacle high)

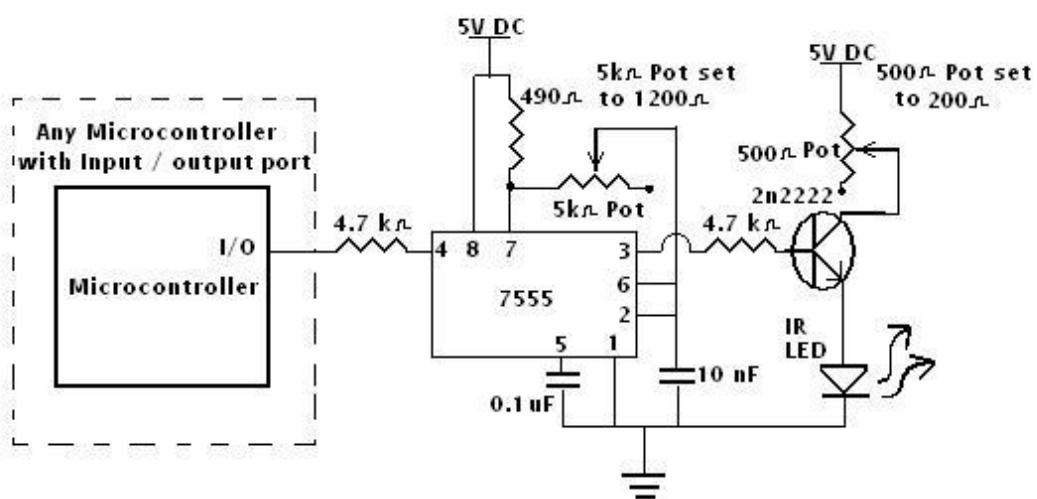
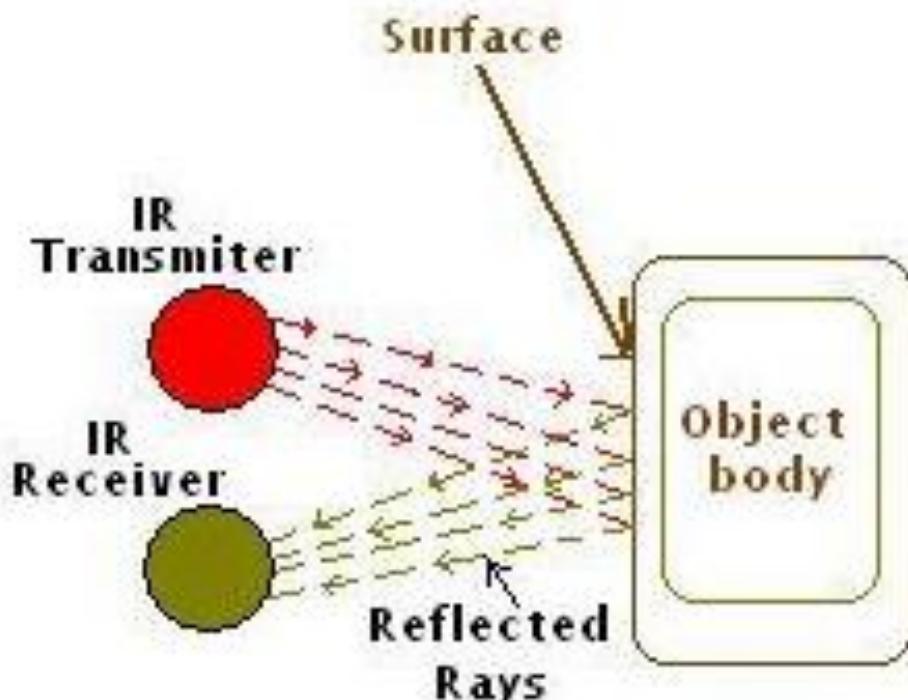
Adjustment: adjust multi-turn resistance

Effective angle: 35 °

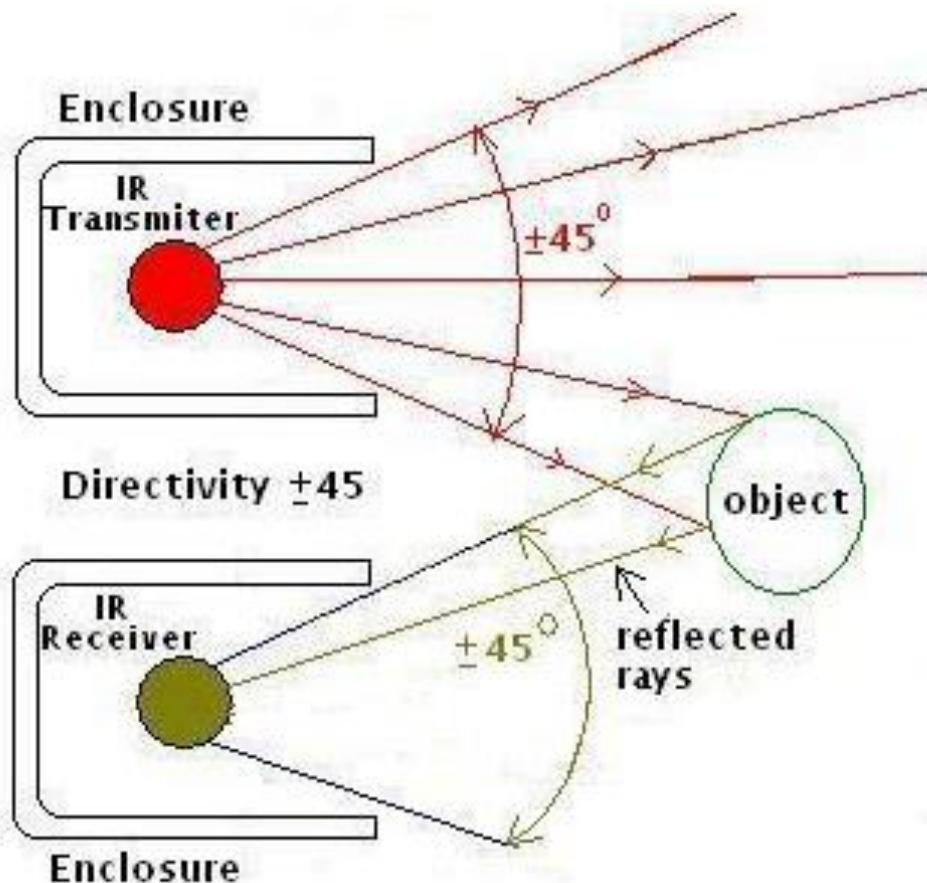
Size: 28mm × 23mm

Weight Size: 9g

The basic concept of IR(infrared) obstacle detection is to transmit the IR signal(radiation) in a direction and a signal is received at the IR receiver when the IR radiation bounces back from a surface of the object.



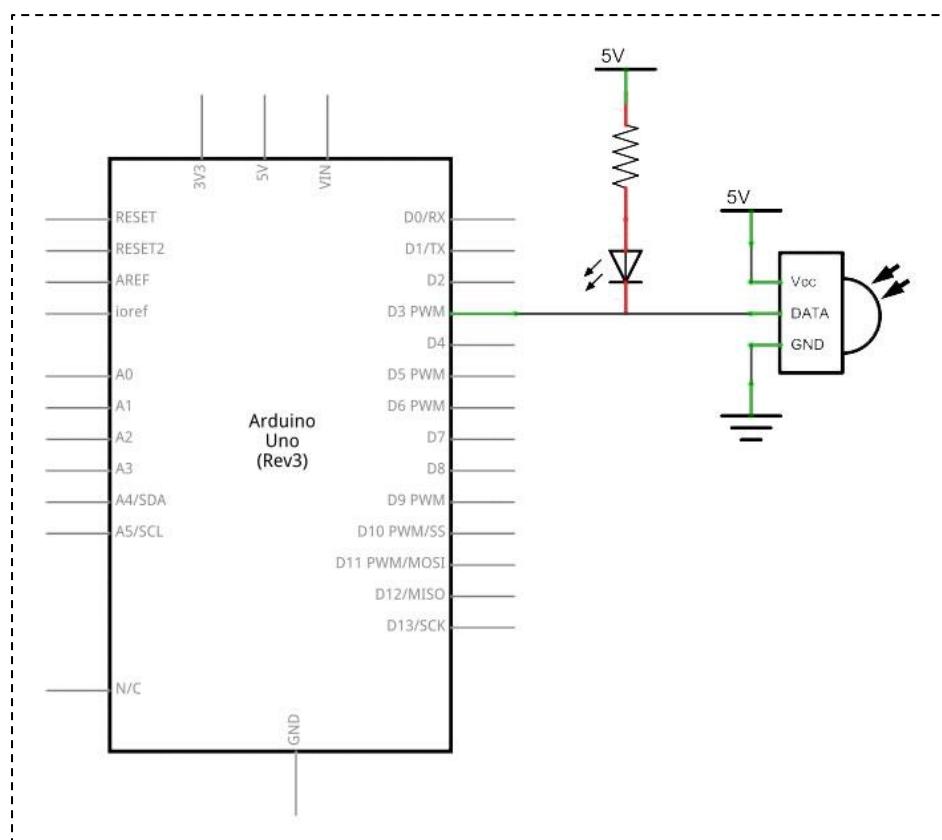
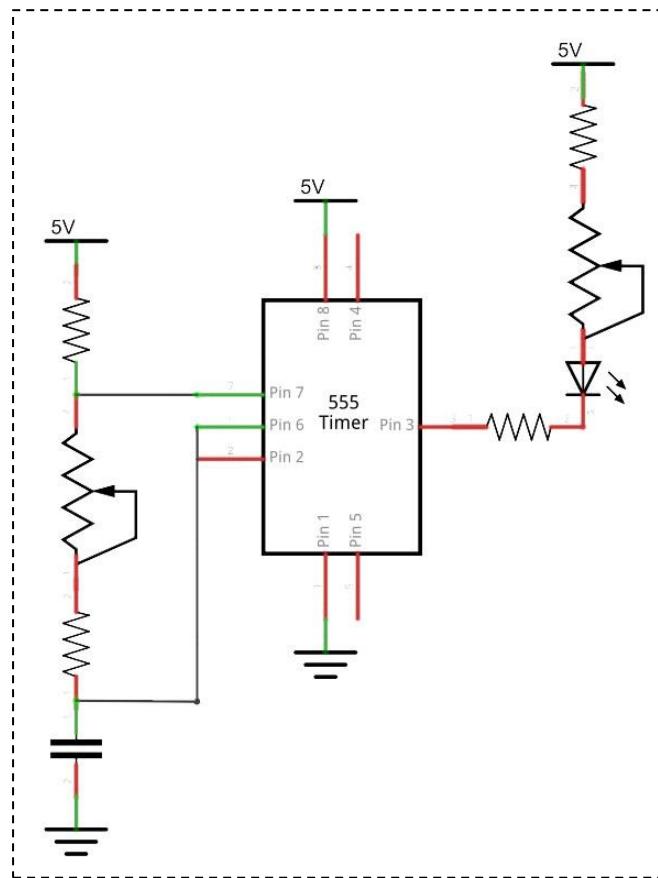
There are two potentiometers on the module one controlling operating frequency (centered at 38 kHz) the other controlling intensity. The detector was designed for 38 kHz and the onboard oscillator circuit is based on a 555 timer. Tweaking gives a little better range but I'd suggest leaving it alone because the useful range is narrow.



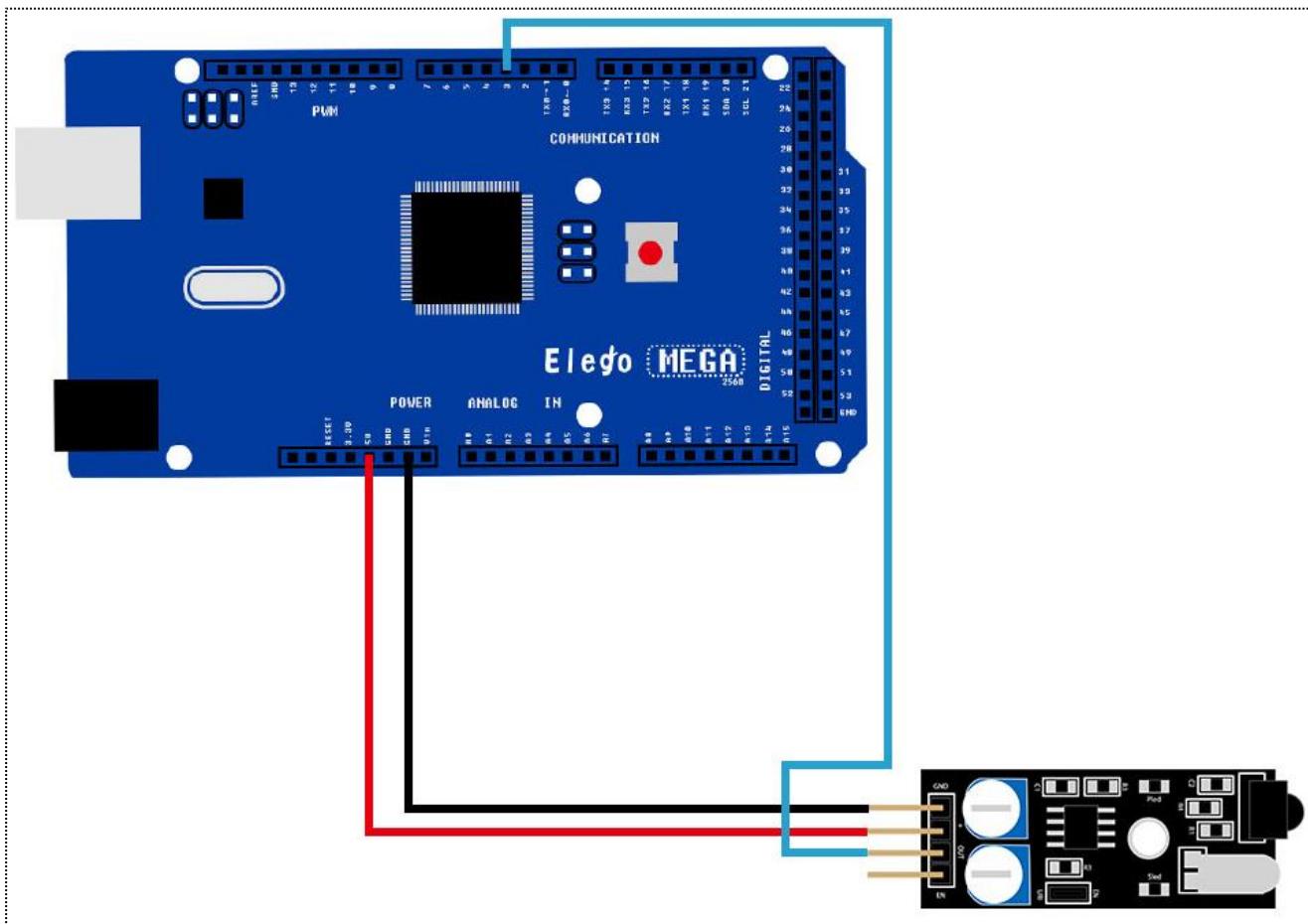
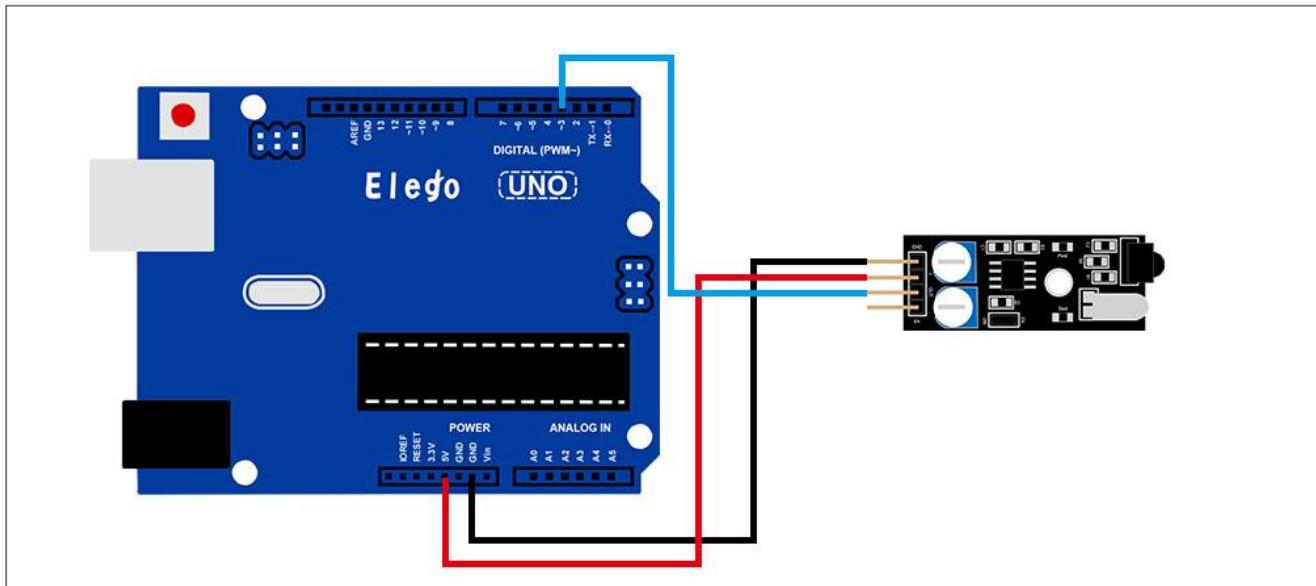
Infrared obstacle avoidance sensor is designed for the design of a wheeled robot obstacle avoidance sensor distance adjustable. This ambient light sensor Adaptable, high precision, having a pair of infrared transmitter and receiver, transmitter tubes emit a certain frequency of infrared, When detecting the direction of an obstacle (reflector), the infrared receiver tube receiver is reflected back, when the indicator is lit, Through the circuit, the signal output interface output digital signal that can be detected by means of potentiometer knob to adjust the distance, the effective distance From 2 ~ 40cm, working voltage of 3.3V-5V, operating voltage range as broad, relatively large fluctuations in the power supply voltage of the situation Stable condition and still work for a variety of microcontrollers, Arduino controller, BS2 controller, attached to the robot that Can sense changes in their surroundings.

Connection

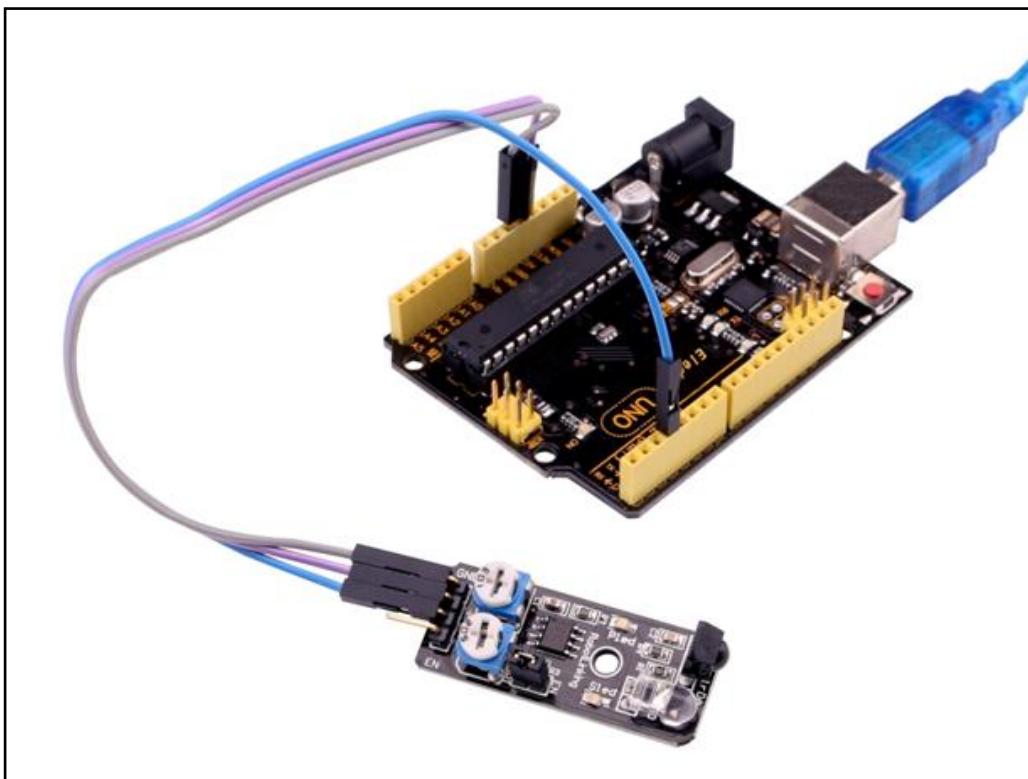
Schematic



wiring diagram



Result

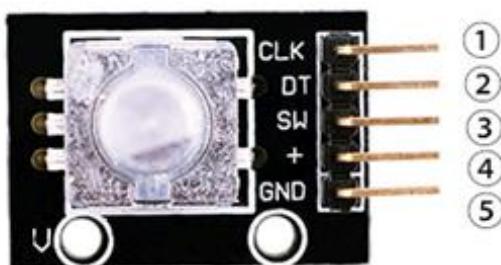


Here we use the obstacle avoidance module and a digital interface, built-in 13 LED build a simple circuit, making avoidance warning lamp, the obstacle avoidance Sensor Access Digital 3 interface, when obstacle avoidance sensor senses a signal, LED light, and vice versa off.

Lesson 23 ROTARY ENCODERS MODULE

Overview

In this experiment, we will learn how to use the rotary encoders module.



- 1.CLK
- 2.DT
- 3.SW
- 4.VCC:3.3V-5V DC
- 5.GND:ground

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x Rotary Encoders module
- (x) x F-M wires

Component Introduction

Rotary Encoders:

Mechanical Specifications:

- Operating Temp: -10°C to 70°C
- Storage Temp: -40°C to 85°C
- Rotational Torque: 50gf.cm max.
- No. and Pos. of detents: 12 detents (Step angle 30°±3°)
- Terminal Strength: A static load for 300gf.cm shall be applied to the tip of the terminals for 10 sec. in any direction
- Shaft push-pull strength: 5.1kgf
- Rotational life: 30,000 cycles

Note:

- RoHS Compliant

Electrical Specifications:

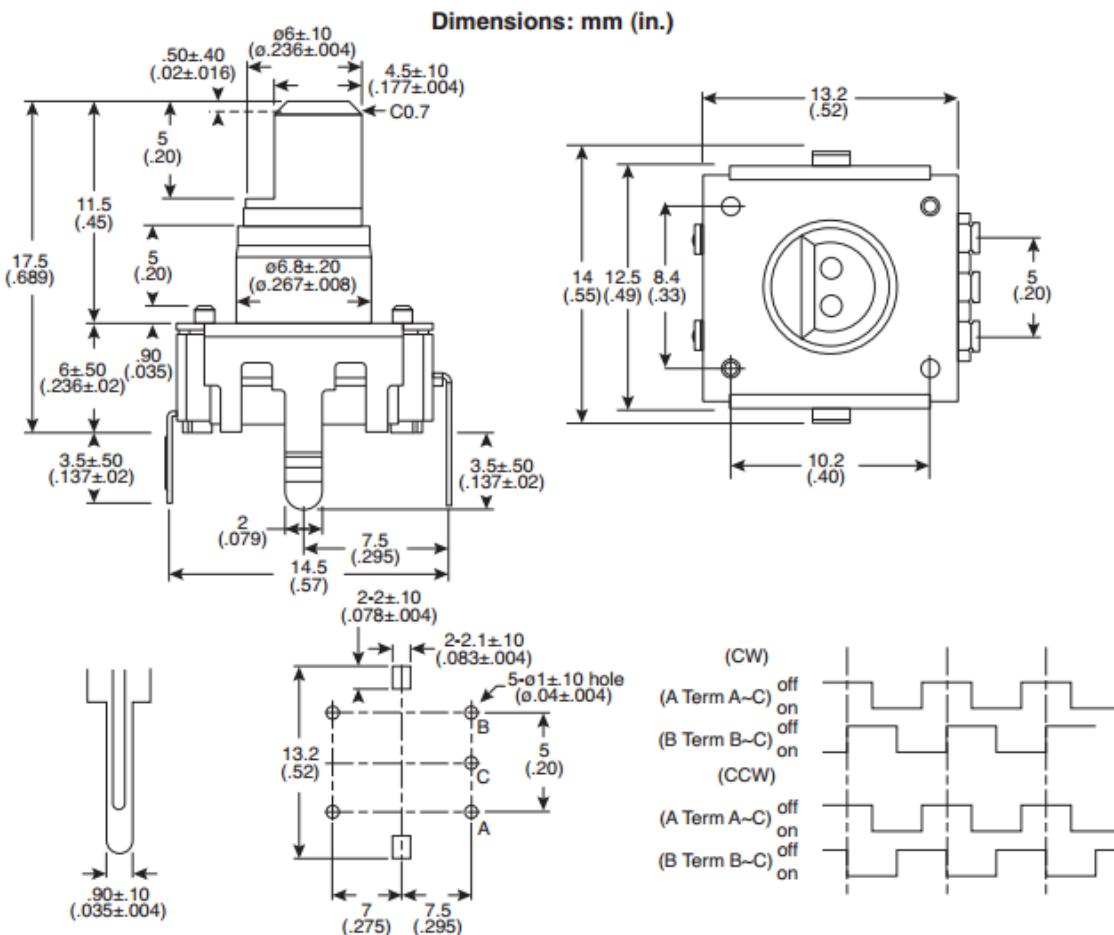
- Rating: 1mA/10VDC
- Insulation Resistance: 50VDC 10MΩ Min.
- Dielectric Strength: 50VAC for 1 min.
- Resolution: 12 pulses/360° for each phase

Soldering Specifications:

- Soldering: To be performed in 5 seconds within 260±5°C
- Manual Soldering: To be performed in 3 seconds within 350±5°C
- Preheating: The entire flow duration should not exceed 2 min., and soldering surface temperature (undersurface of PCB) shall be settled within 100°C

Push-on Switch Specifications:

- Type: Single Pole Single Throw (Push on)
- Rating: 10mA/5VDC
- Switch Travel (mm): 0.5±0.4
- Operating Force: 200~460gf
- Operating Life: 20,000 times



Principle

Incremental encoder

Incremental encoders give two-phase square wave, the phase difference between them 90 °, often referred to as A and B channels. One of the channels is given and speed-related information, at the same time, by sequentially comparing two channel signals, the direction of rotation of the information obtained. There is also a special signal called Z or Zero channel, which gives the absolute zero position encoder, the signal is a square wave with the center line of channel A square wave coincide.

Clockwise counterclockwise

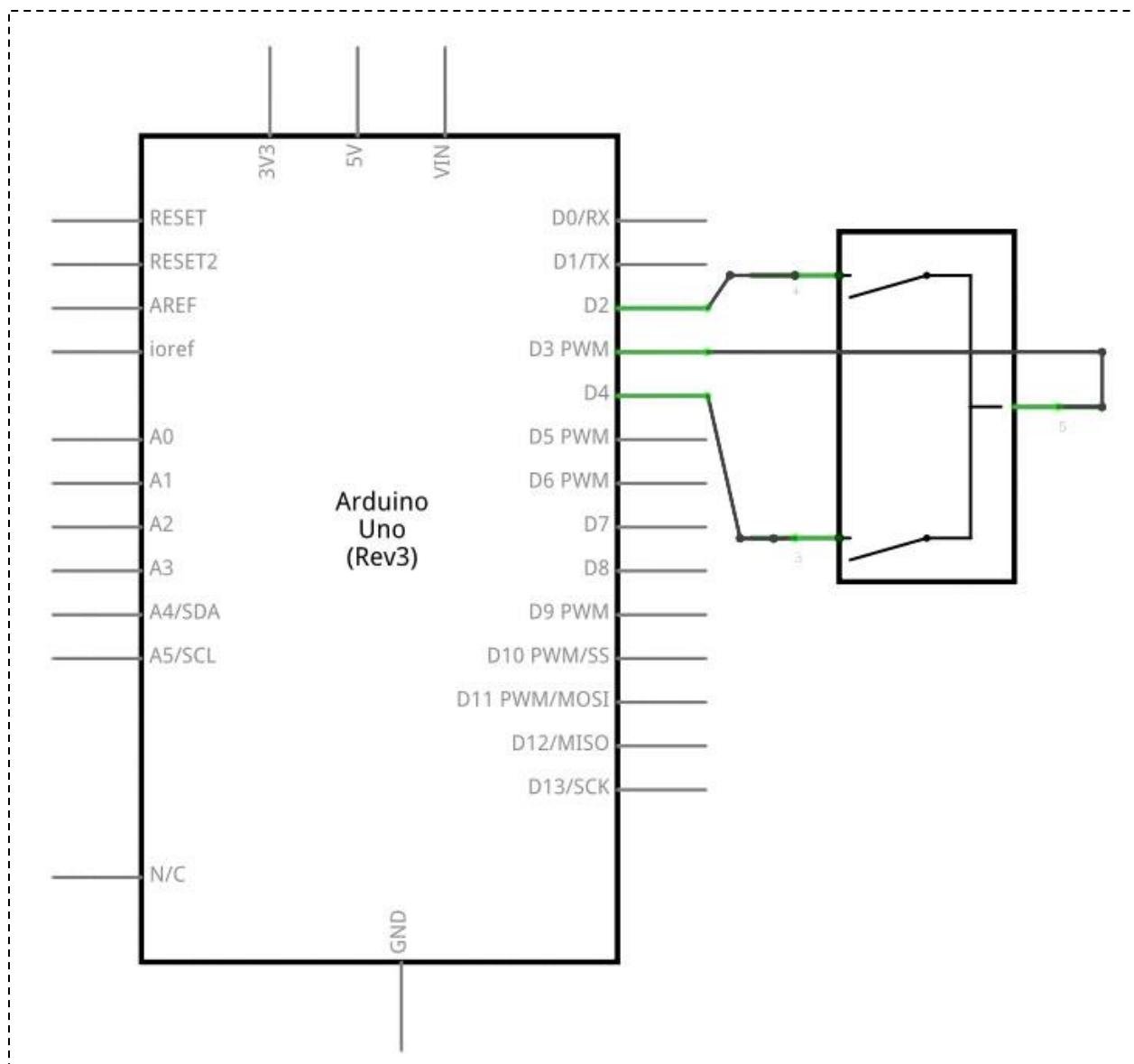
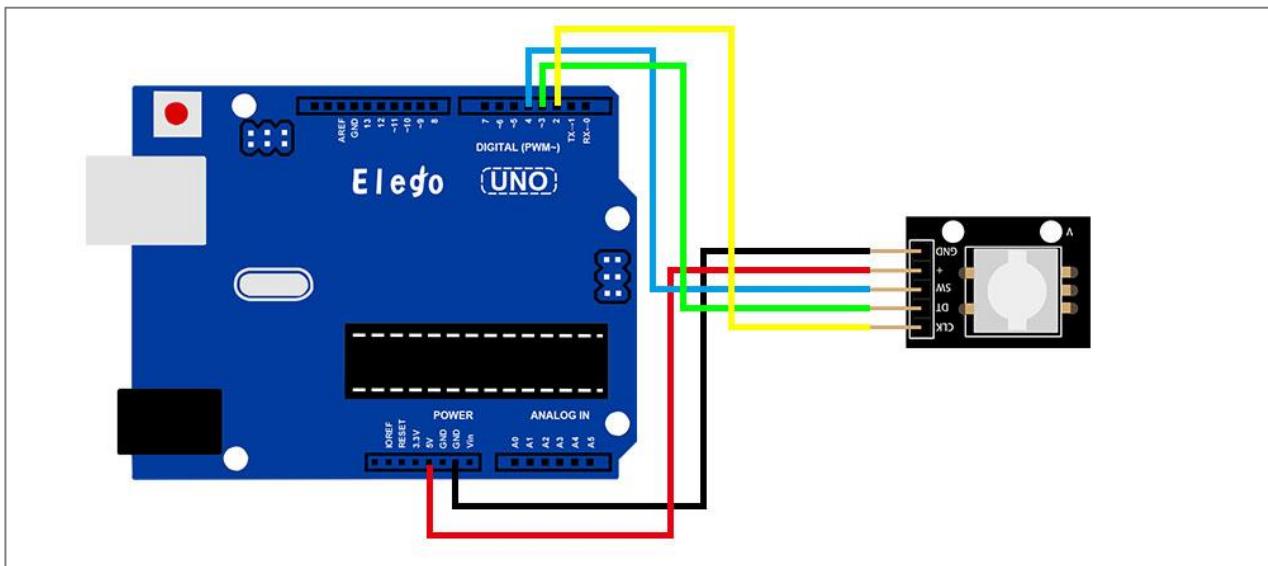
A	B
1	1
0	1
0	0
1	0
1	1
1	0
0	0
0	1

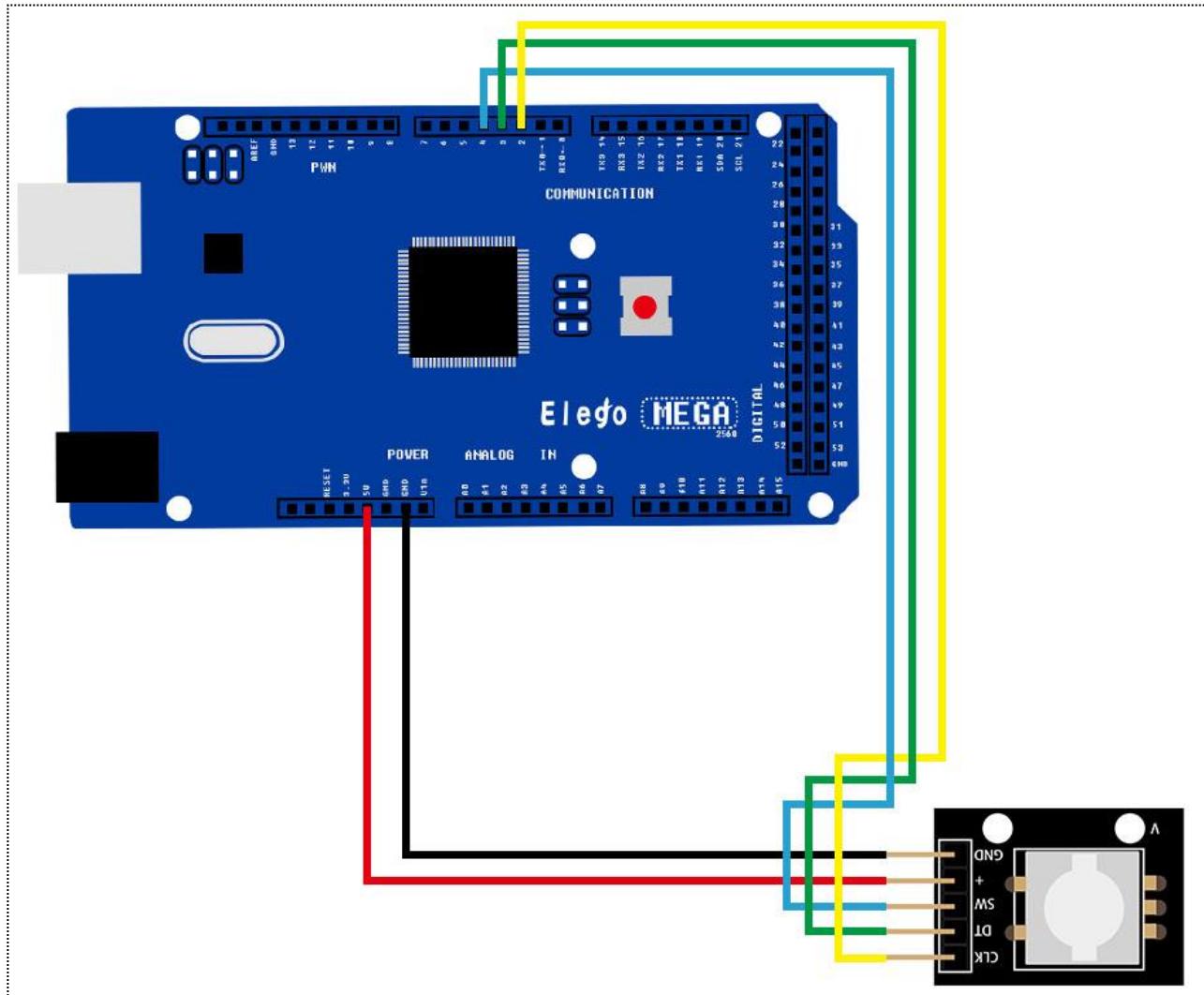
Incremental encoder accuracy depends on the mechanical and electrical two factors, these factors are: Raster indexing error, disc eccentricity, bearing eccentricity, e-reading Several means into the optical portion of the errors and inaccuracies. Determine the encoder resolution is measured in electrical degrees, the encoder accuracy depends Set the pulse encoder generates indexing. The following electrical degrees with a 360 ° rotation of the shaft to said machine, and rotation of the shaft must be a full week of Period. To know how much electrical equivalent of the mechanical angle of 360 degrees can be calculated with the following formula: Electrical 360 = Machine 360 ° / n ° pulses / revolution

Encoder indexing error is the electrical angle of the unit two successive pulse maximum offset to represent. Error exists in any encoder, which is caused by the aforementioned factors. Eltra encoder maximum error is ± 25 electrical degrees (declared in any condition), equivalent to the rated Offset values ± 7%, as the phase difference 90 ° (electrical) of the two channels of the maximum deviation ± 35 electrical degrees is equal to ± 10% deviation left Ratings Right.

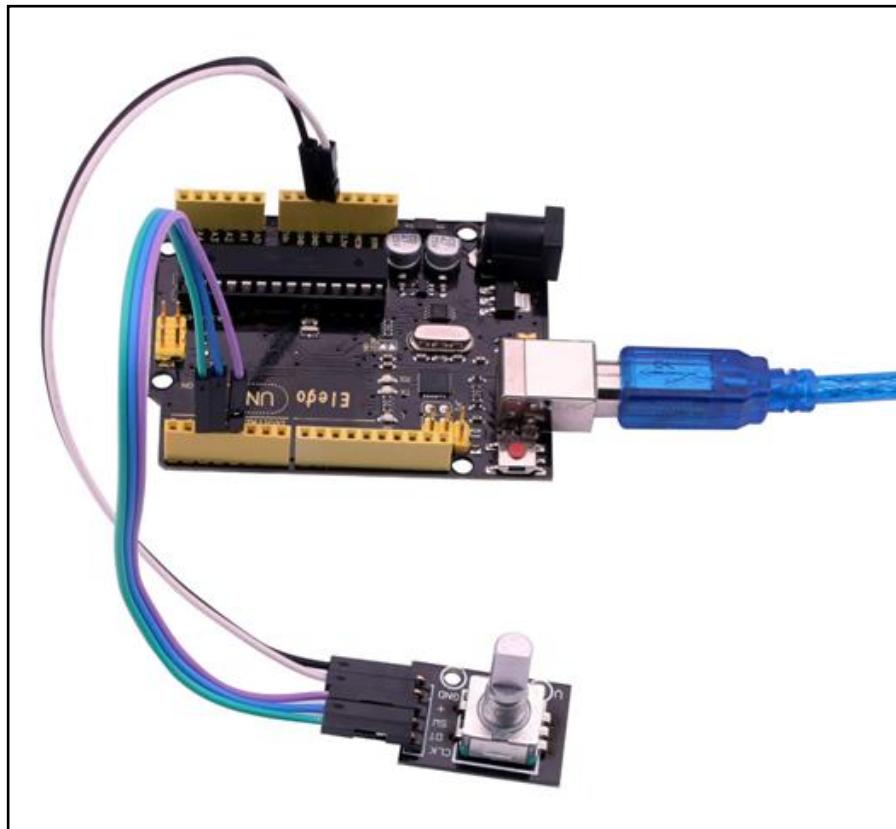
Connection

Schematic

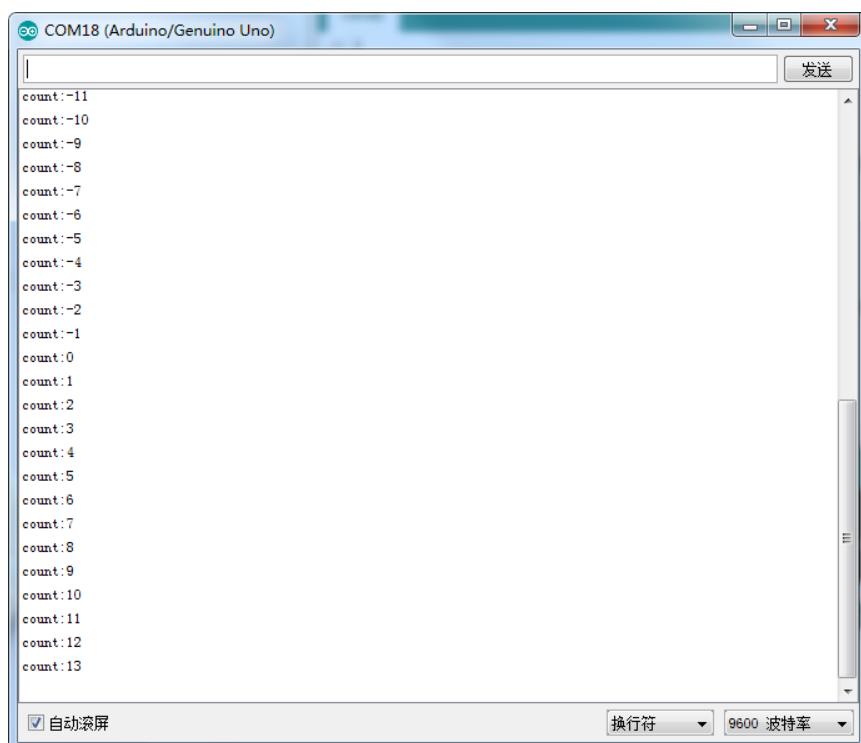
**wiring diagram**



Result



Upload the program, rotate the encodes then open the monitor, we can see the data as below:



```
count:-11
count:-10
count:-9
count:-8
count:-7
count:-6
count:-5
count:-4
count:-3
count:-2
count:-1
count:0
count:1
count:2
count:3
count:4
count:5
count:6
count:7
count:8
count:9
count:10
count:11
count:12
count:13
```

Lesson 24 1 CHANNEL RELAY MODULE

Overview

In this experiment, we will learn how to use the 1 channel relay module.

Relay is a kind of component when the change of the input variables (incentive) to specified requirements, the output electric circuits of the charged amount occurs due to the step change of a kind of electrical appliances. This company produces the relay module can meet in 28 v to 240 v ac or dc power to control all kinds of other electric parts. MCU can be used to achieve the goal of timing control switch. Can be applied to guard against theft and alarm, toys, construction and other fields. Relay is an electrical control device. It has a control system (also called input circuit) and control system (also called the output circuit), the interaction between. Usually used in automatic control circuit, it is actually with a small current to control large current operation of a kind of "automatics"



- 1.GND:ground
- 2.VCC:3.3V-5V DC
- 3.OUTPUT

Component Required:

- (1) x Elegoo Uno R3
- (1) x USB cable
- (1) x 1 channel relay module
- (x) x F-M wires

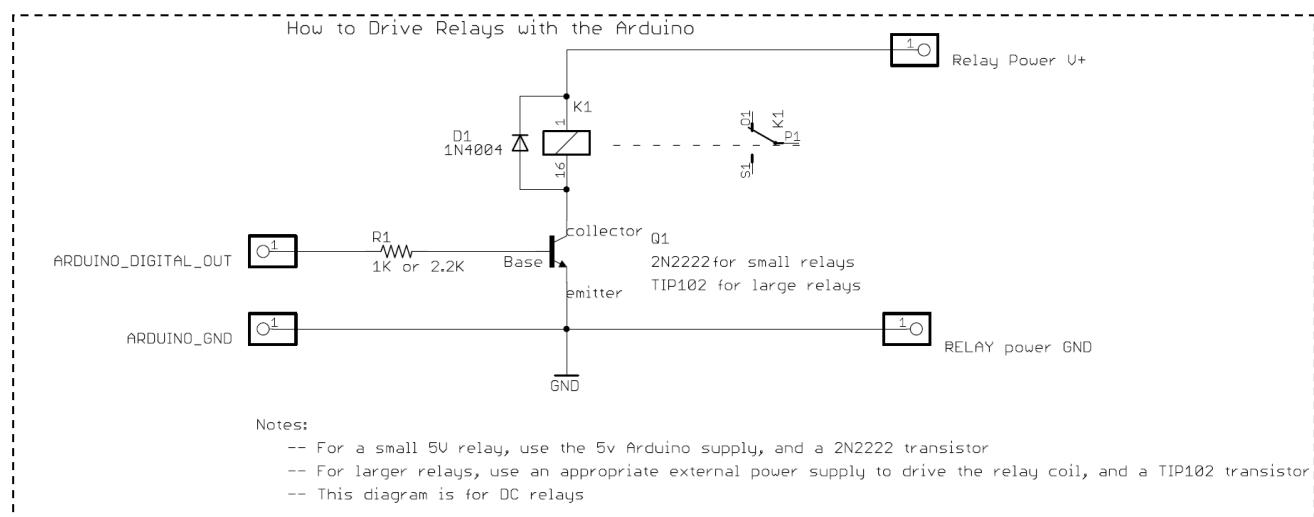
Component Introduction

Relay:

A relay is an electrically operated switch. Many relays use an electromagnet to mechanically operate a switch, but other operating principles are also used, such as solid-state relays. Relays are used where it is necessary to control a circuit by a low-power signal (with complete electrical isolation between control and controlled circuits), or where several circuits must be controlled by one signal. The first relays were used in long distance telegraph circuits as amplifiers: they repeated the signal coming in from one circuit and re-transmitted it on another circuit. Relays were used extensively in telephone exchanges and early computers to perform logical operations.

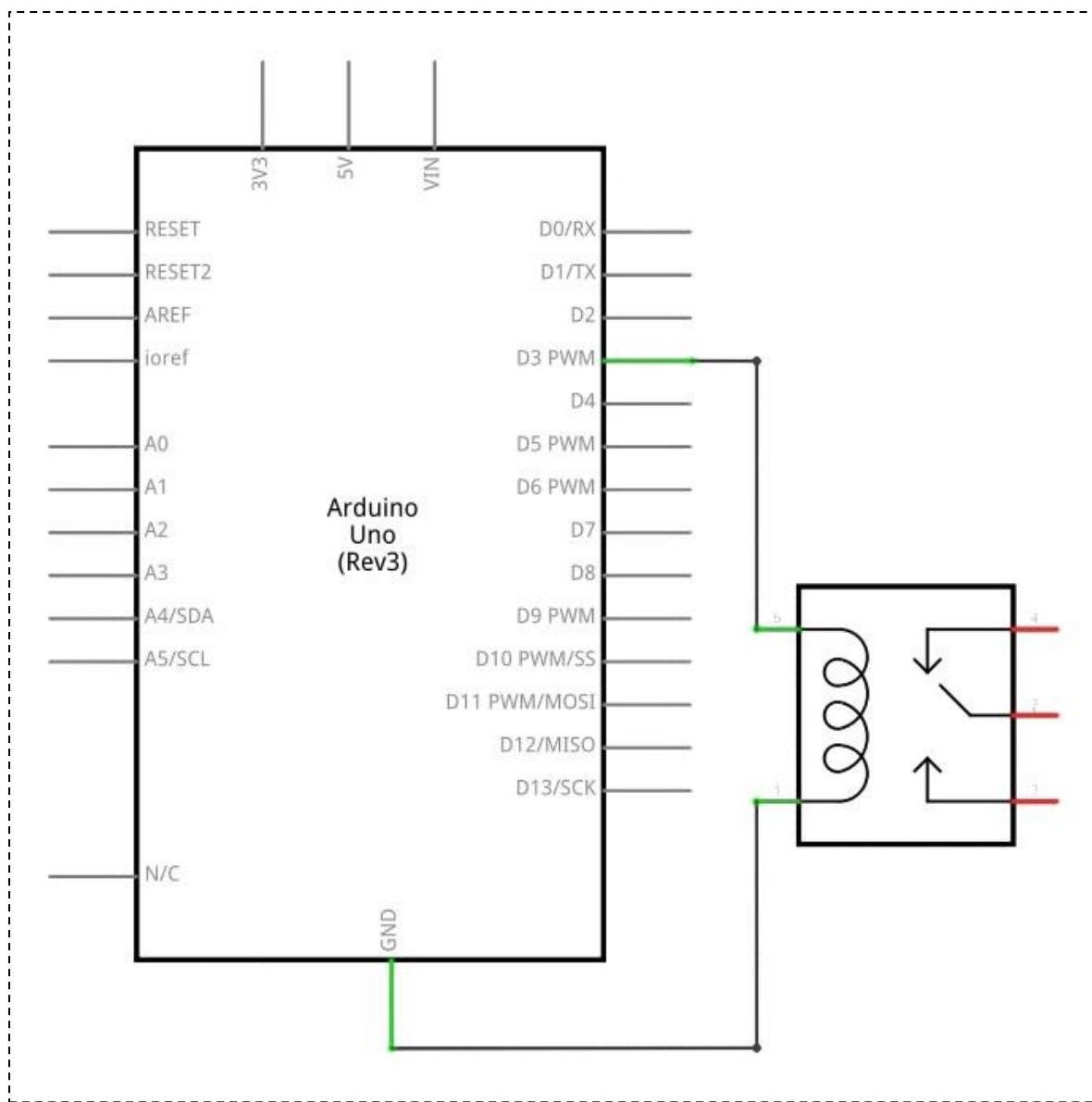
A type of relay that can handle the high power required to directly control an electric motor or other loads is called a contactor. Solid-state relays control power circuits with no moving parts, instead using a semiconductor device to perform switching. Relays with calibrated operating characteristics and sometimes multiple operating coils are used to protect electrical circuits from overload or faults; in modern electric power systems these functions are performed by digital instruments still called "protective relays".

Bellow is the schematic of how to drive relay with arduino (down load from the arduino.cc)

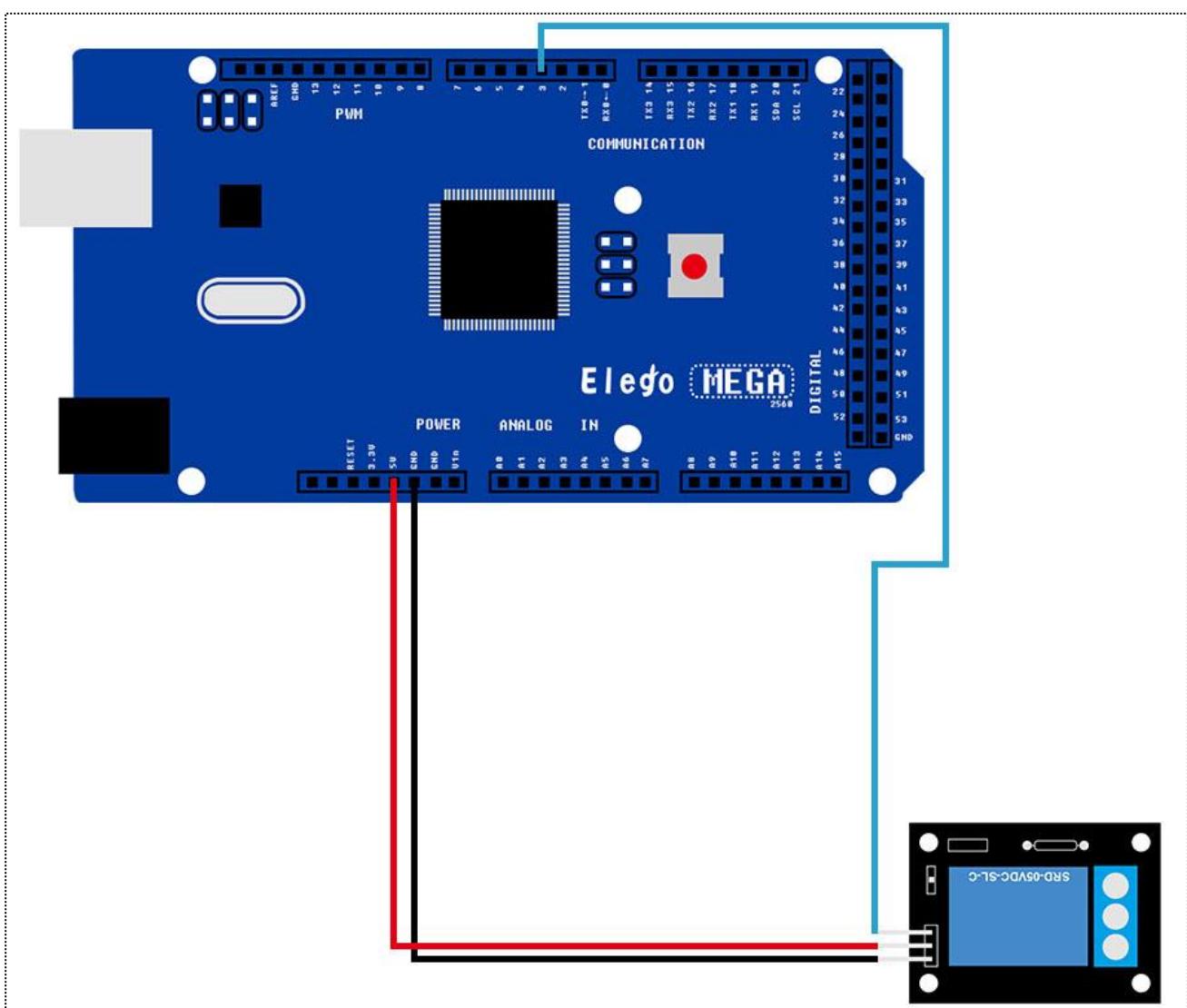
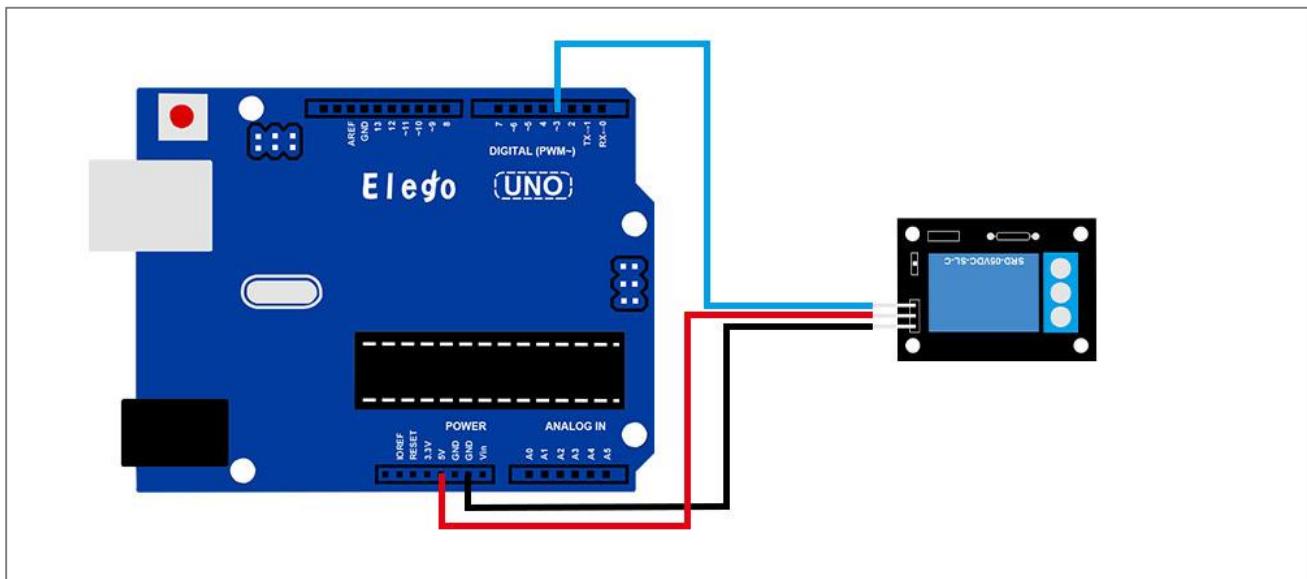


Connection

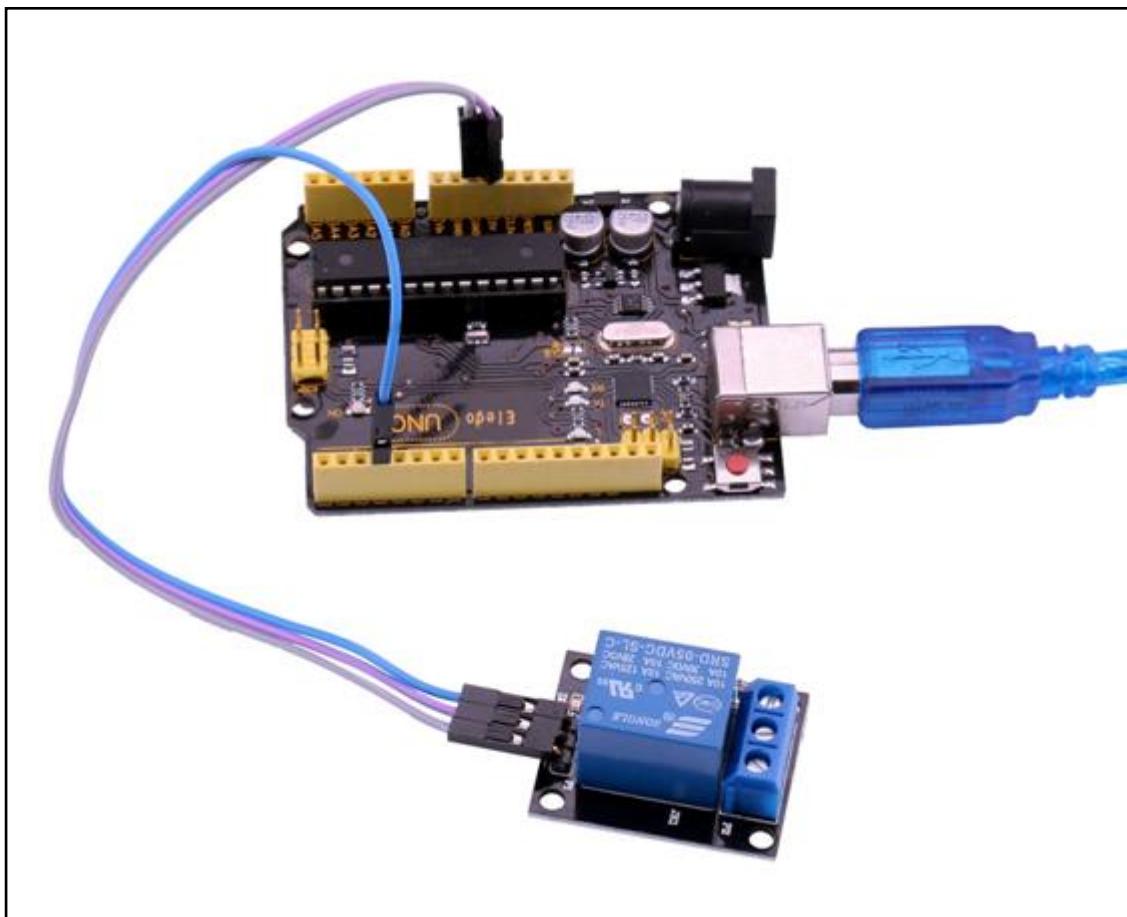
Schematic



wiring diagram



Result



Lesson 25 LCD Display

Overview

In this lesson, you will learn how to wire up and use an alphanumeric LCD display.

The display has an LED backlight and can display two rows with up to 16 characters on each row. You can see the rectangles for each character on the display and the pixels that make up each character. The display is just white on blue and is intended for showing text.

In this lesson, we will run the Arduino example program for the LCD library, but in the next lesson, we will get our display to show the temperature and light level, using sensors.

Component Required:

(1) x Elegoo Uno R3

(1) x LCD1602 display

(1) x Potentiometer

(1) x Breadboard

(16) x F-M wires

Component Introduction

LCD1602

Introduction to the pins of LCD1602:

VSS: A pin that connects to ground

VDD: A pin that connects to a +5V power supply

VO: A pin that adjust the contrast of LCD1602

RS: A register select pin that controls where in the LCD's memory you are writing data to. You can select either the data register, which holds what goes on the screen, or an instruction register, which is where the LCD's controller looks for instructions on what to do next.

R/W: A Read/Write pin that selects reading mode or writing mode

E: An enabling pin that, when supplied with low-level energy, causes the LDC module to execute

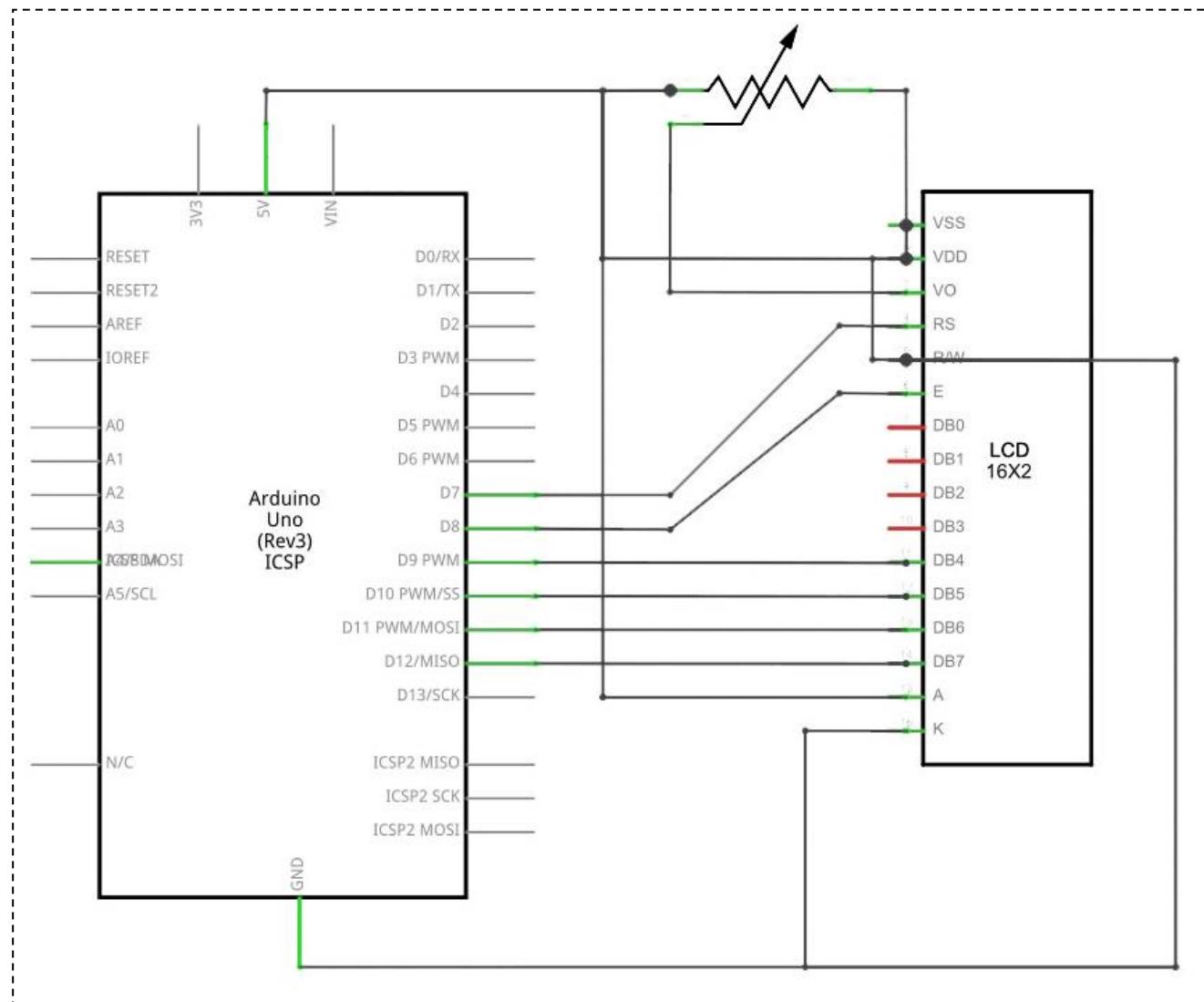
relevant instructions.

D0-D7: Pins that read and write data

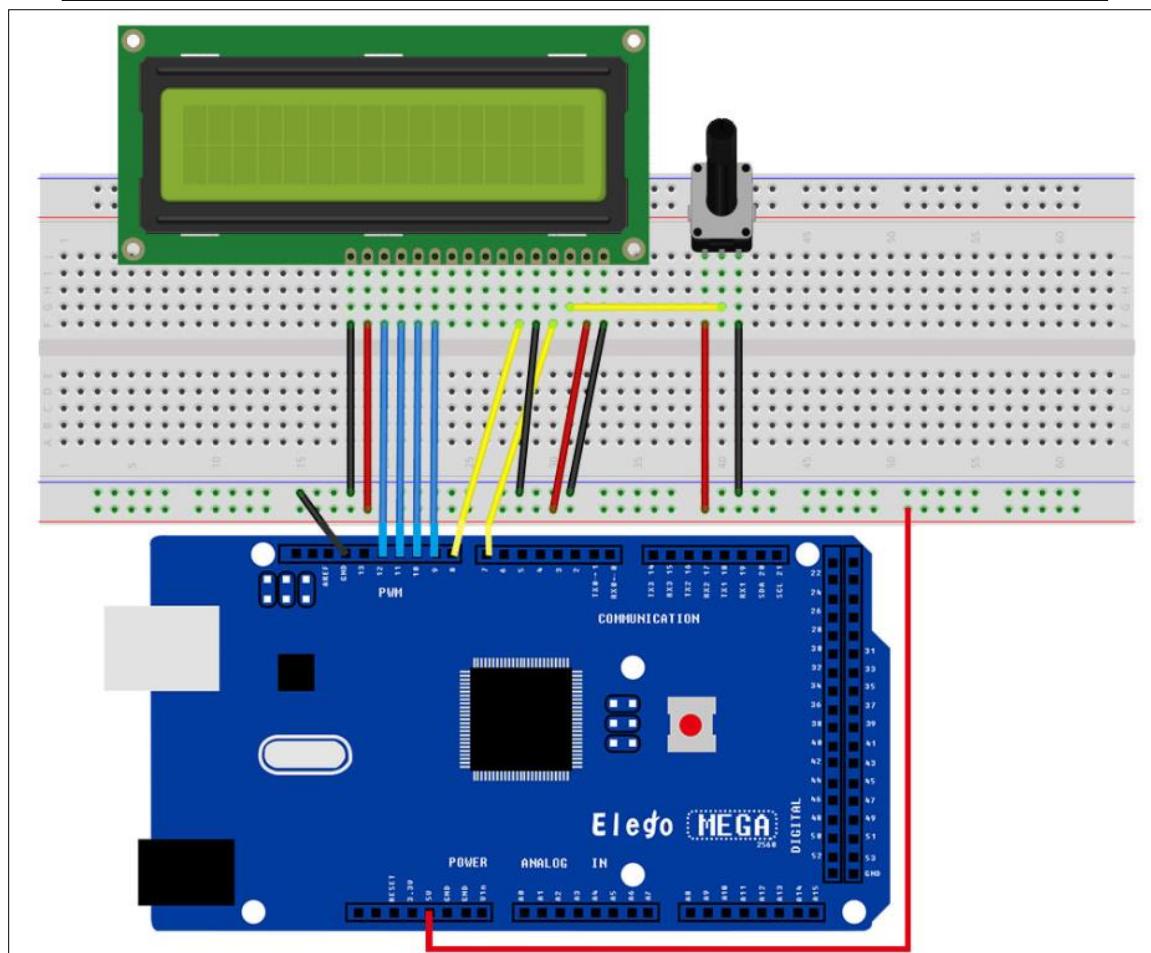
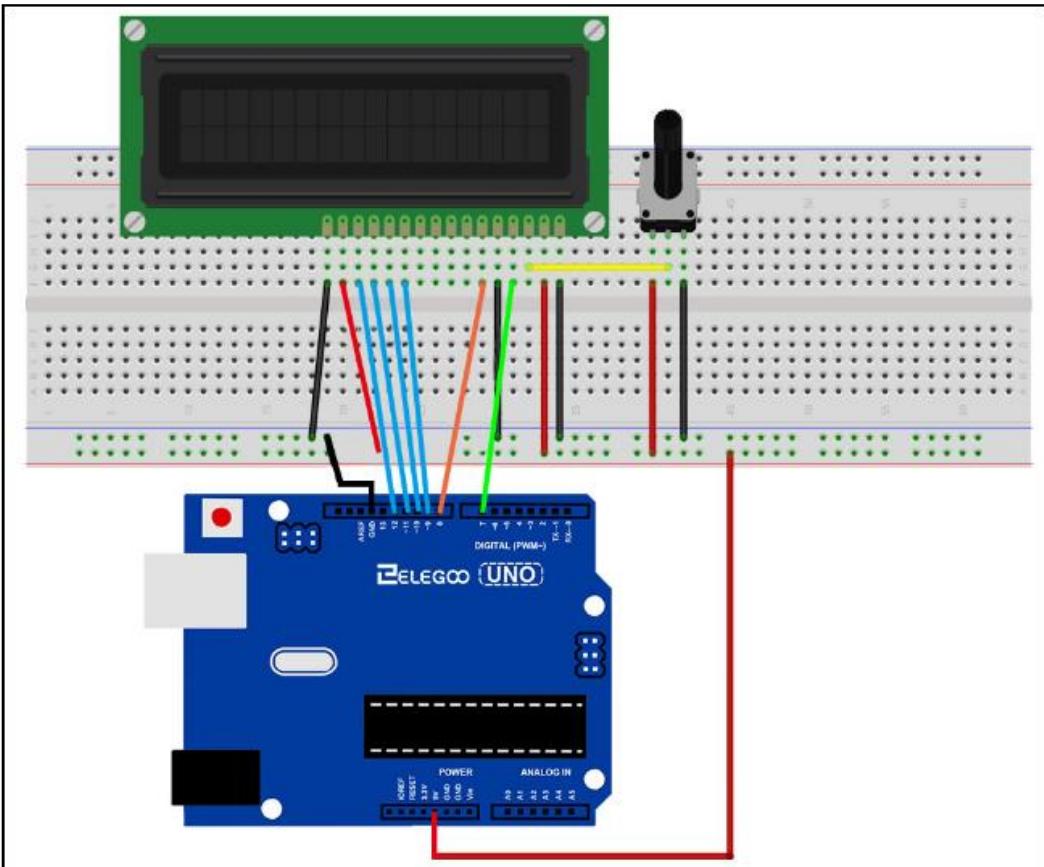
A and K: Pins that control the LED backlight

Connection

Schematic



wiring diagram



The LCD display needs six Arduino pins, all set to be digital outputs. It also needs 5V and GND connections.

There are a number of connections to be made. Lining up the display with the top of the breadboard helps to identify its pins without too much counting, especially if the breadboard has its rows numbered with row 1 as the top row of the board. Do not forget, the long yellow lead that links the slider of the pot to pin 3 of the display. The 'pot' is used to control the contrast of the display.

You may find that your display is supplied without header pins attached to it. If so, follow the instructions in the next section.

Code

Now that we have the physical setup, all we need now is the code.

Before you can run this, you have to make sure that you have install the <LiquidCrystal> library. Or you need to install again. If you do not do this, your code won't work.

This example uses different pins to the ones we use, so find the line of code below:

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

and change it to be:

LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

Upload the code to your Arduino board and you should see the message 'hello, world' displayed, followed by a number that counts up from zero.

The first thing of note in the sketch is the line:

#include <LiquidCrystal.h>

This tells Arduino that we wish to use the Liquid Crystal library.

Next we have the line that we had to modify. This defines which pins of the Arduino are to be connected to which pins of the display.

LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

The arguments to this are as follows:

Display Pin Name Display Pin Number Arduino Pin (in this example) RS 4 7 E 6 8 D4 11 9 D5 12
10 D6 13 11 D7 14 12

After uploading this code, make sure the backlight is lit up, and adjust the potentiometer all

the way around until you see the text message

In the 'setup' function, we have two commands:

```
lcd.begin(16, 2);  
lcd.print("hello, world!");
```

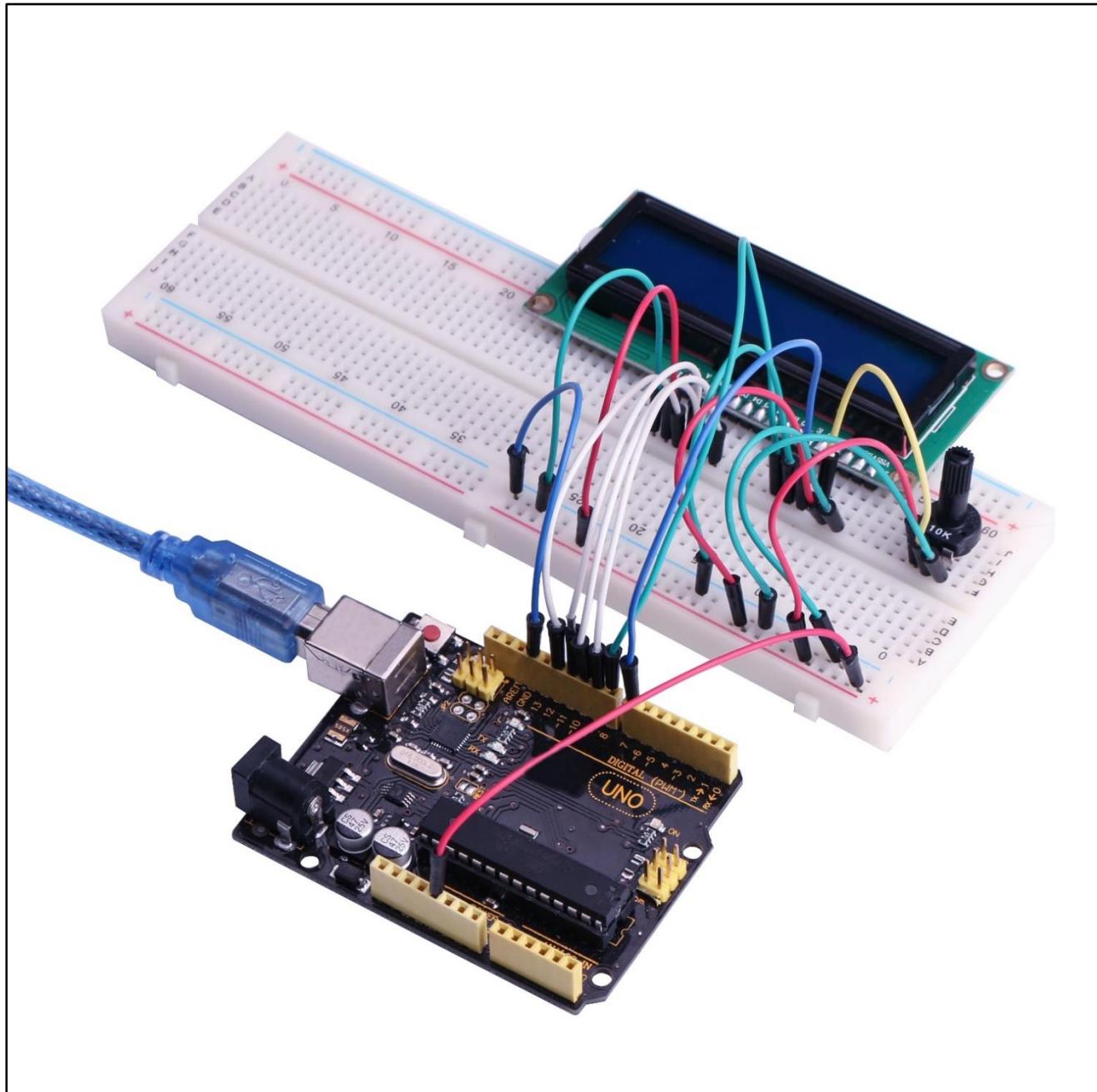
The first tells the Liquid Crystal library how many columns and rows the display has. The second line displays the message that we see on the first line of the screen.

In the 'loop' function, we also have two commands:

```
lcd.setCursor(0, 1);  
lcd.print(millis()/1000);
```

The first sets the cursor position (where the next text will appear) to column 0 & row 1. Both column and row numbers start at 0 rather than 1.

The second line displays the number of milliseconds since the Arduino was reset.



Lesson 26 Ultrasonic Sensor Module

Overview

Ultrasonic sensor are great for all kind of projects that need distance measurements, avoiding obstacles as examples.

The HC-SR04 are inexpensive and easy to use since we will be using a Library specifically designed for these sensor.

Component Required:

- (1) x Elegoo Uno R3
- (1) x Ultrasonic sensor module
- (4) x F-M wires

Component Introduction

Ultrasonic sensor

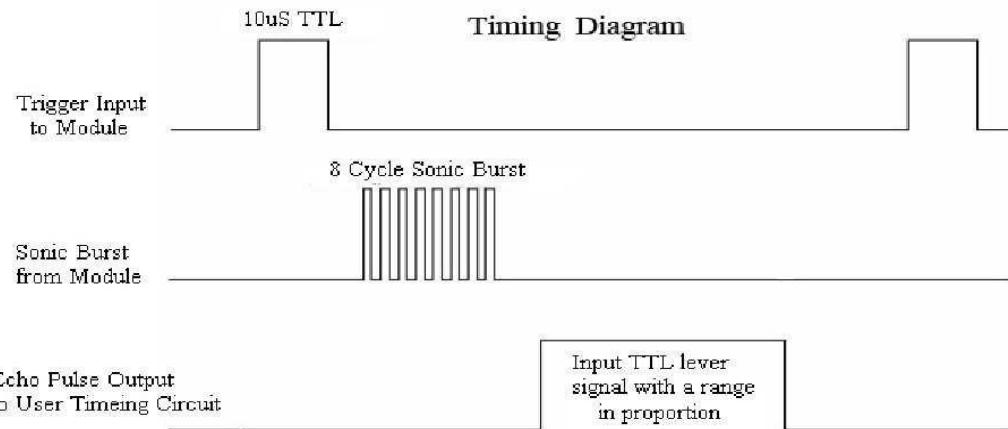
Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The module includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- (1) Using IO trigger for at least 10us high level signal,
- (2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
- (3) If the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to receiving.

Test distance = (high level time × velocity of sound (340M/S) / 2

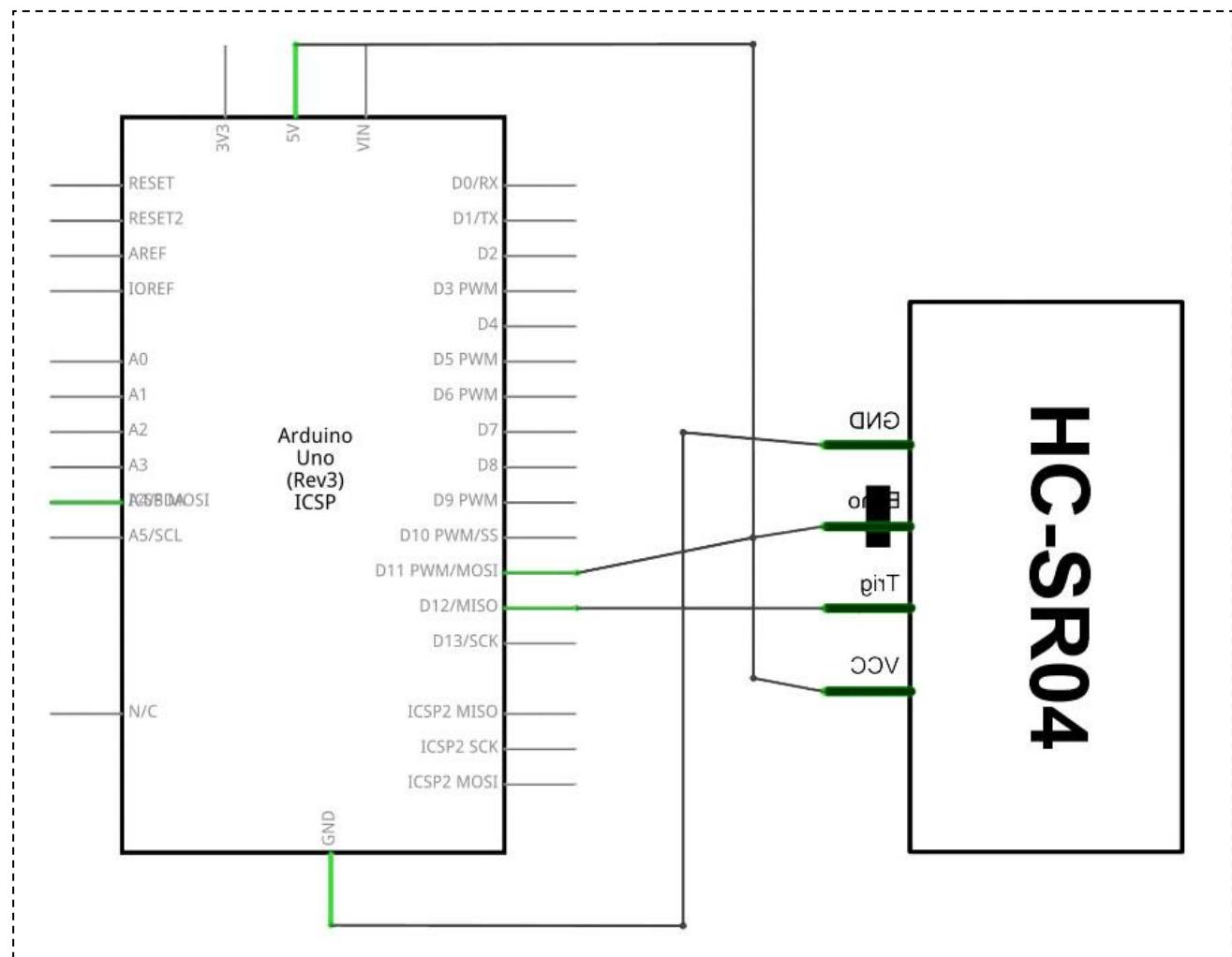
The Timing diagram is shown below. You only need to supply a short 10μs pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion .You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula: $μS / 58 = \text{centimeters}$ or $μS / 148 = \text{inch}$; or: the

range = high level time * velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.

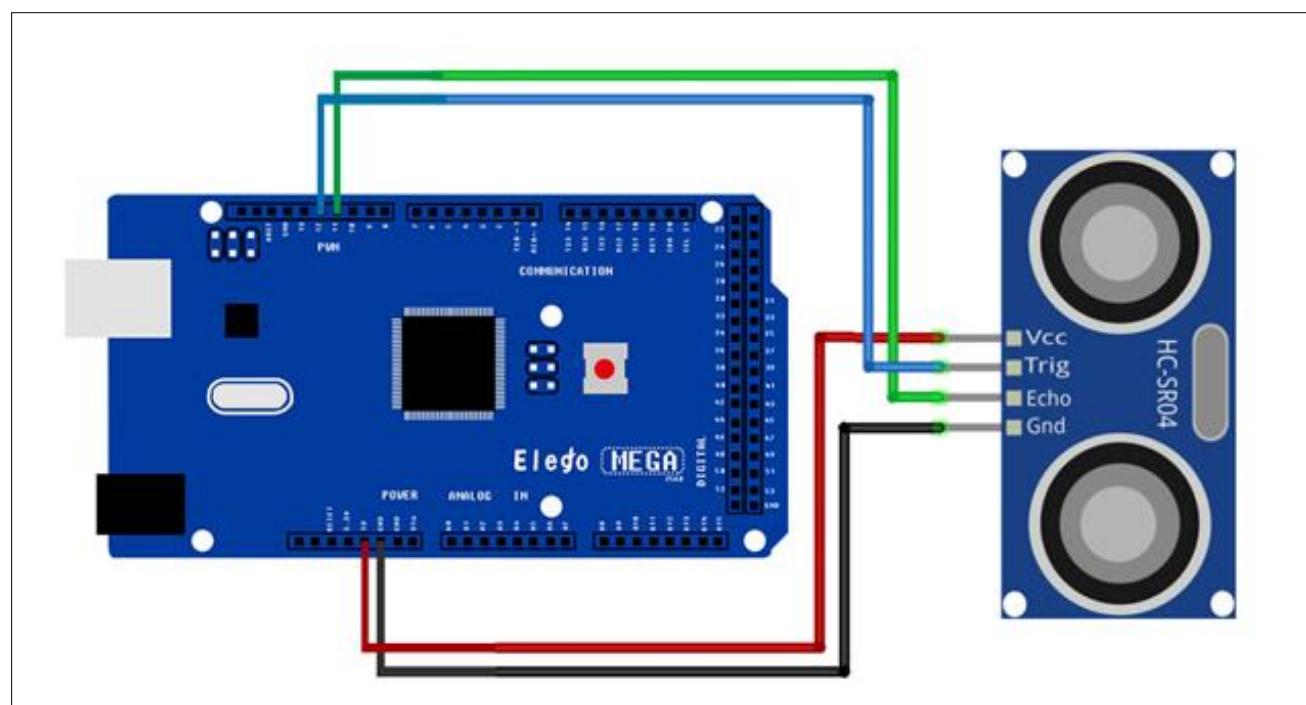
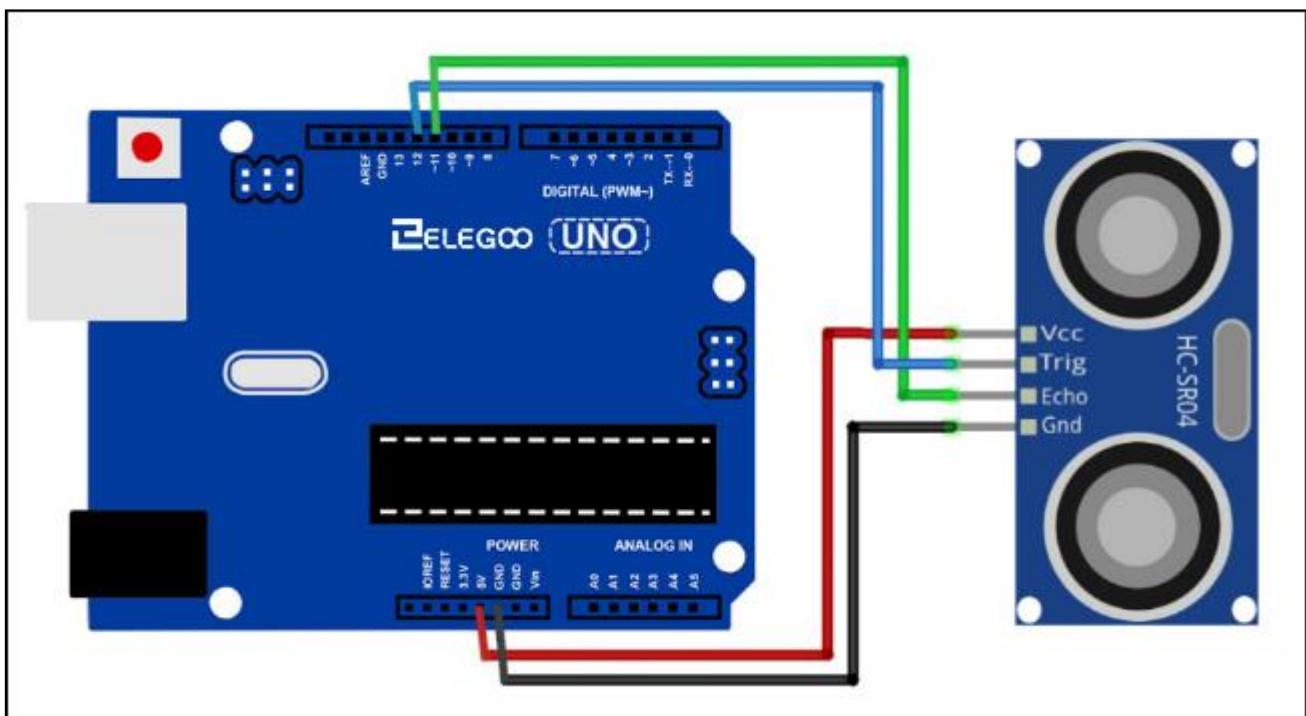


Connection

Schematic



Wiring diagram



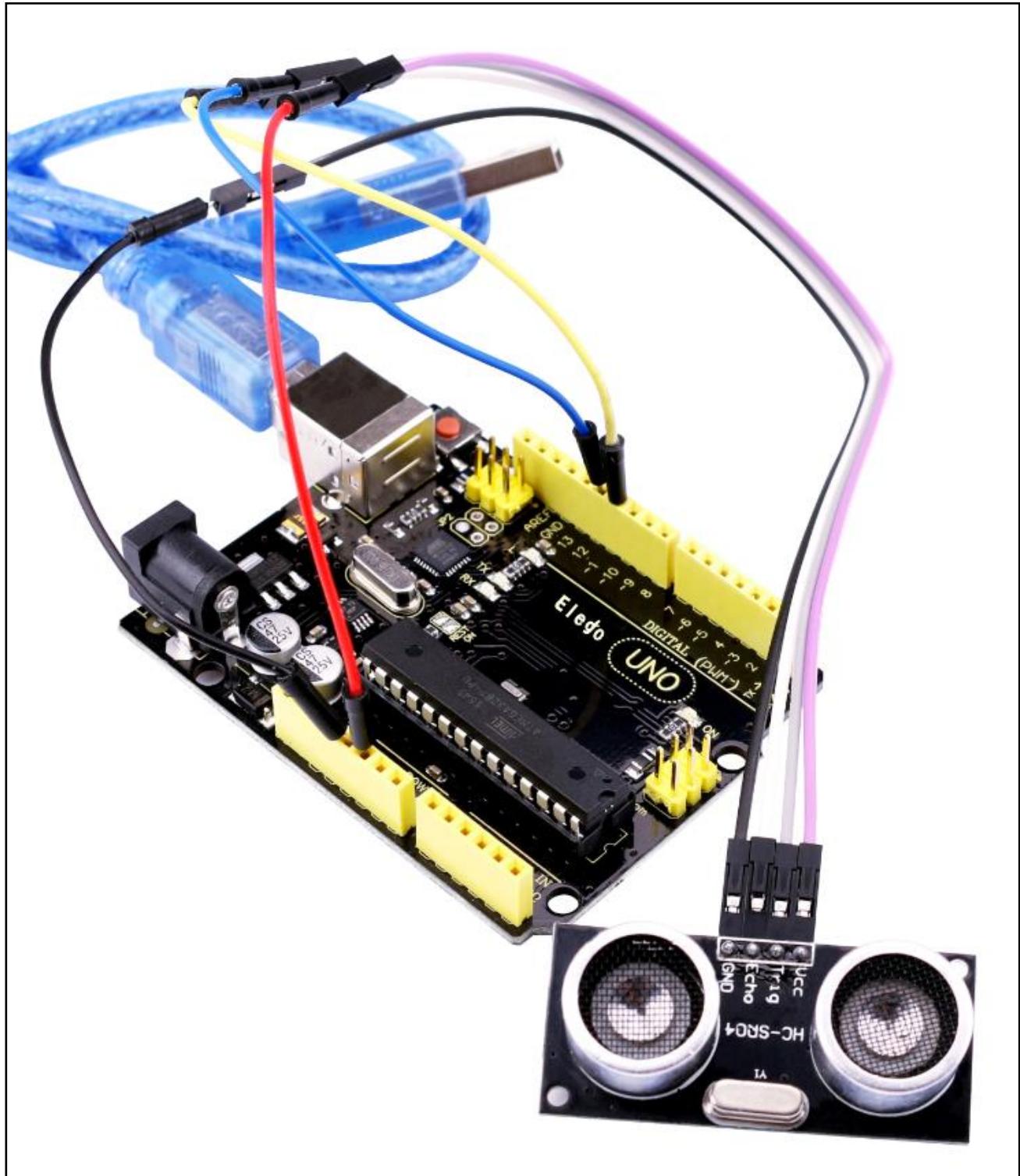
Code

Using a Library designed for these sensors will make our code short and simple.

We include the library at the beginning of our code, and then by using simple commands we can control the behavior of the sensor.

Now that we have the physical setup, all we need now is the code.

Before you can run this, you have to make sure that you have install the < HC-SR04_Library > library. Or you need to install again. If you do not do this, your code won't work.



Lesson 27 MPU6050 Module

Overview

In this lesson, we will learn how to use MPU6050 module which is one of the best IMU (Inertia Measurement Unit) sensors, compatible with arduino. IMU sensors like the MPU 6050 are used in self balancing robots, UAVs, smart phones, etc.

Component Required:

- (1) x Elegoo Uno R3
- (1) x MPU6050 module
- (8) x F-M wires

Component Introduction

MPU6050 SENSOR

The InvenSense MPU-6050 sensor contains a MEMS accelerometer and a MEMS gyro in a single chip. It is very accurate, as it contains 16-bits analog to digital conversion hardware for each channel. Therefore it captures the x, y, and z channel at the same time. The sensor uses the I2C-bus to interface with the Arduino.

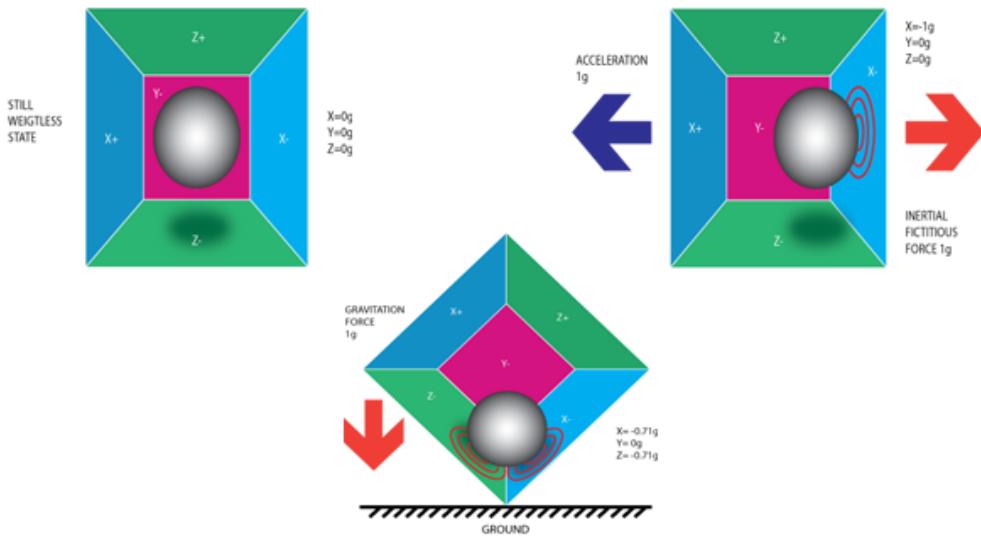
IMU sensors are one of the most inevitable type of sensors used today in all kinds of electronic gadgets. They are seen in smart phones, wearables, game controllers, etc. IMU sensors help us in getting the attitude of an object, attached to the sensor in three dimensional space. These values usually in angles, thus help us to determine its attitude. Thus, they are used in smart phones to detect its orientation. And also in wearable gadgets like the nike fuel band or fit bit, which use IMU sensors to track movement.

How does it work?

IMU sensors usually consists of two or more parts. Listing them by priority, they are : accelerometer, gyroscope, magnetometer and altimeter. The MPU 6050 is a 6 DOF (Degrees of Freedom) or a six axis IMU sensor, which means that it gives six values as output. Three values from the accelerometer and three from the gyroscope. The MPU 6050 is a sensor

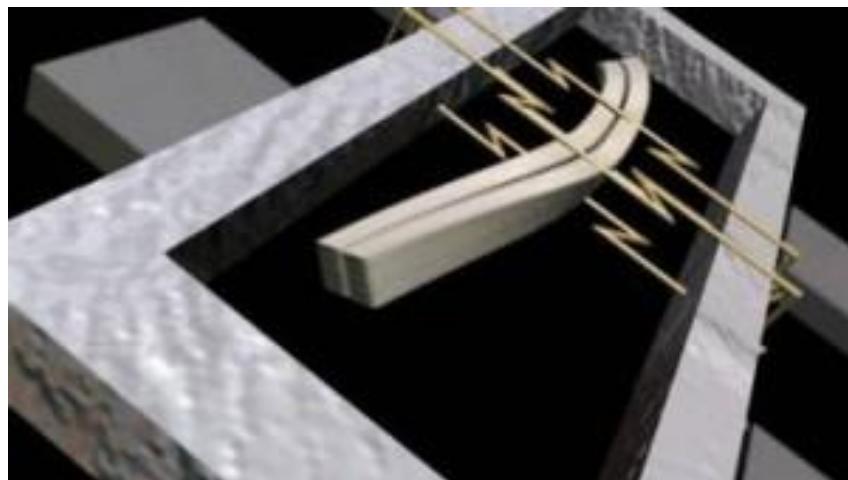
based on MEMS (Micro Electro Mechanical Systems) technology. Both the accelerometer and the gyroscope is embedded inside a single chip. This chip uses I2C (Inter Integrated Circuit) protocol for communication.

How does an accelerometer work?



An accelerometer works on the principle of piezo electric effect. Here, imagine a cuboidal box, having a small ball inside it, like in the picture above. The walls of this box are made with piezo electric crystals. Whenever you tilt the box, the ball is forced to move in the direction of the inclination, due to gravity. The wall with which the ball collides, creates tiny piezo electric currents. There are totally, three pairs of opposite walls in a cuboid. Each pair corresponds to an axis in 3D space: X, Y and Z axes. Depending on the current produced from the piezo electric walls, we can determine the direction of inclination and its magnitude. For more information check this.

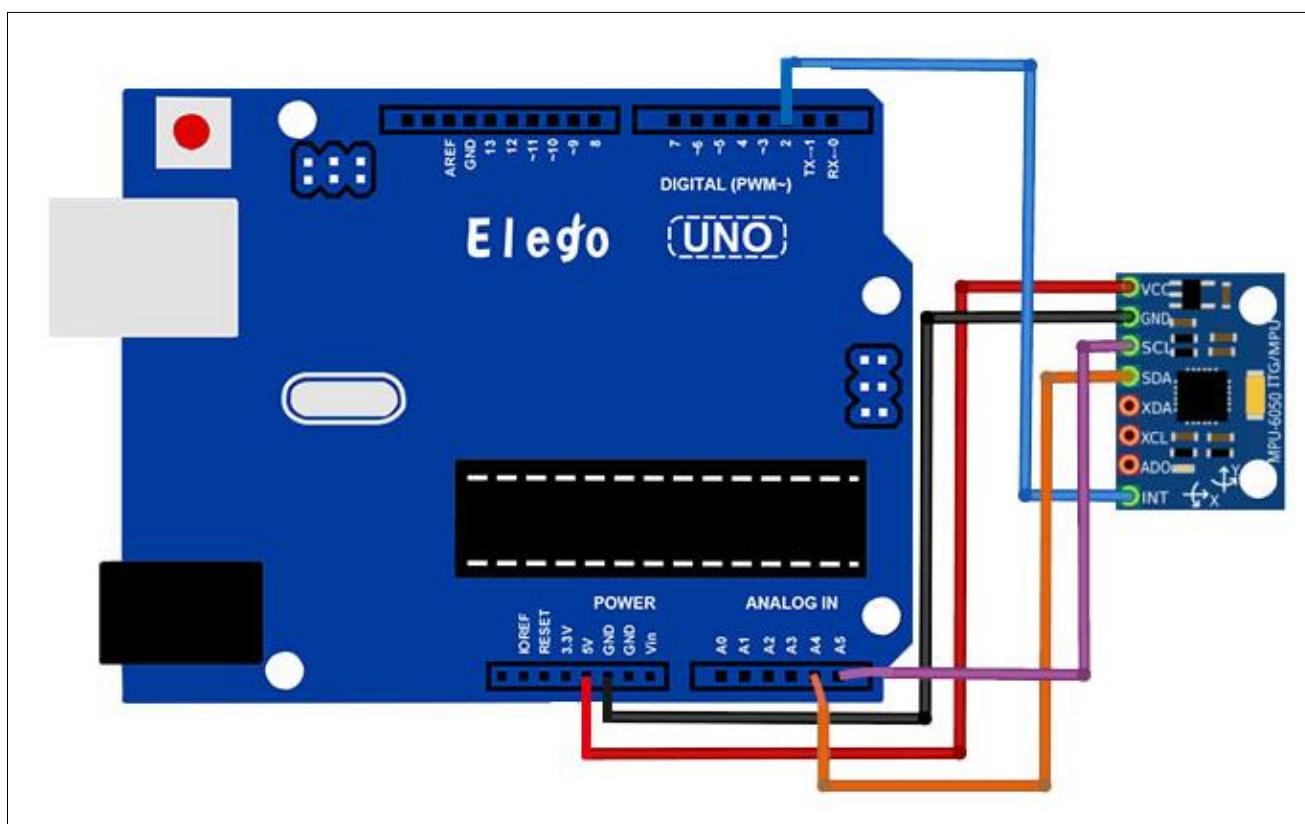
How does a gyroscope work?



Gyroscopes work on the principle of Coriolis acceleration. Imagine that there is a fork like structure, that is in constant back and forth motion. It is held in place using piezo electric crystals. Whenever, you try to tilt this arrangement, the crystals experience a force in the direction of inclination. This is caused as a result of the inertia of the moving fork. The crystals thus produce a current in consensus with the piezo electric effect, and this current is amplified. The values are then refined by the host microcontroller.

Connection

wiring diagram



The MPU 6050 communicates with the Arduino through the I2C protocol. The MPU 6050 is connected to Arduino as shown in the following diagram. Here, if your MPU 6050 module has a 5V pin, then you can connect it to your arduino's 5V pin. Else, you will have to connect it to the 3.3V pin. Next, the GND of the arduino is connected to the GND of the MPU 6050.

The program we will be running here, also takes advantage of the arduino's interrupt pin. Therefore, connect arduino's digital pin 2 (interrupt pin 0) to the pin labelled as INT on the MPU 6050. Next, we need to set up the I2C lines. For this connect the pin labelled as SDA on the

MPU 6050 to the arduino's analog pin 4 (SDA). And the pin labelled as SCL on the MPU 6050 to the arduino's analog pin 5 (SCL). And that's it, you have finished wiring up the Arduino MPU 6050.

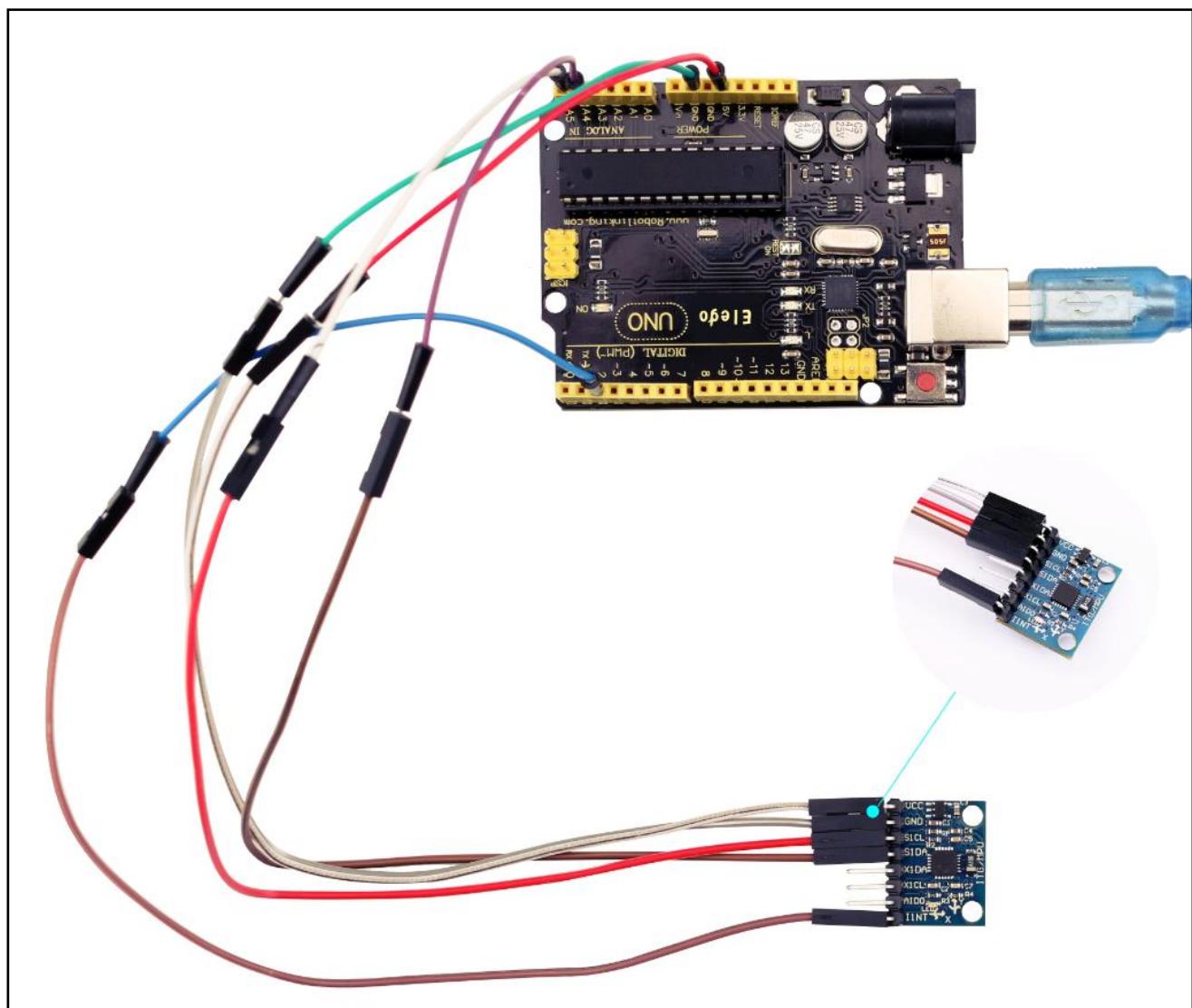
Libraries needed

I2Cdev

MPU6050

The Code

See the code file.



Lesson 28 HC-SR501 PIR Sensor

Overview

In this lesson you will learn how to use a PIR movement detector with an UNO.

The UNO is the heart of this project. It 'listens' to the PIR sensor and when motion is detect, instructs the LED to light on or shut off.

Component Required:

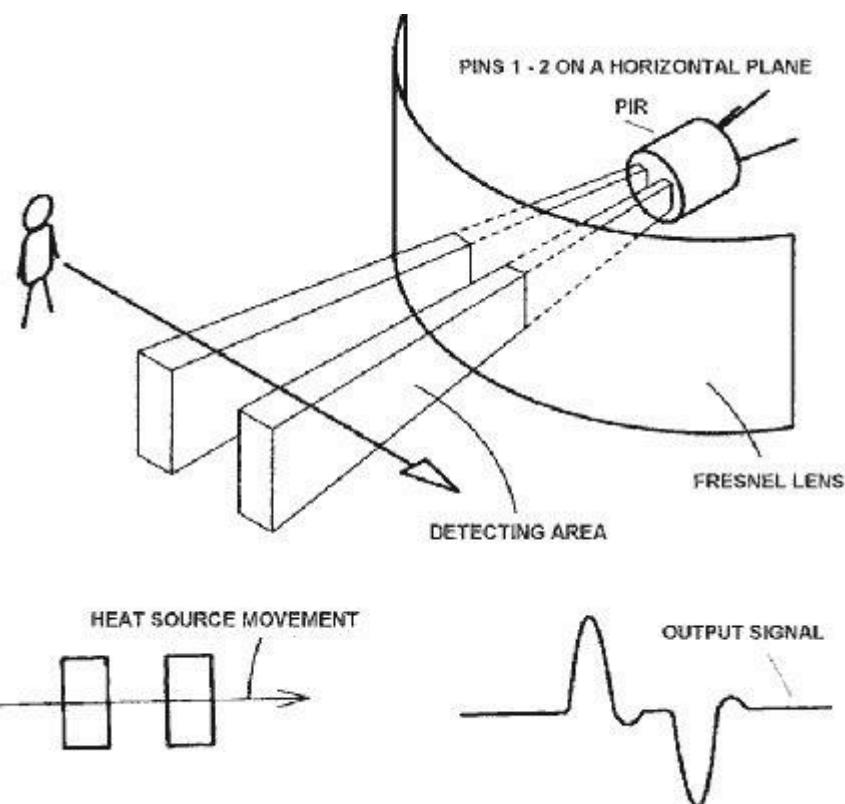
- (1) x Elegoo Uno R3
- (1) x HC-SR501
- (3) x F-M wires

Component Introduction

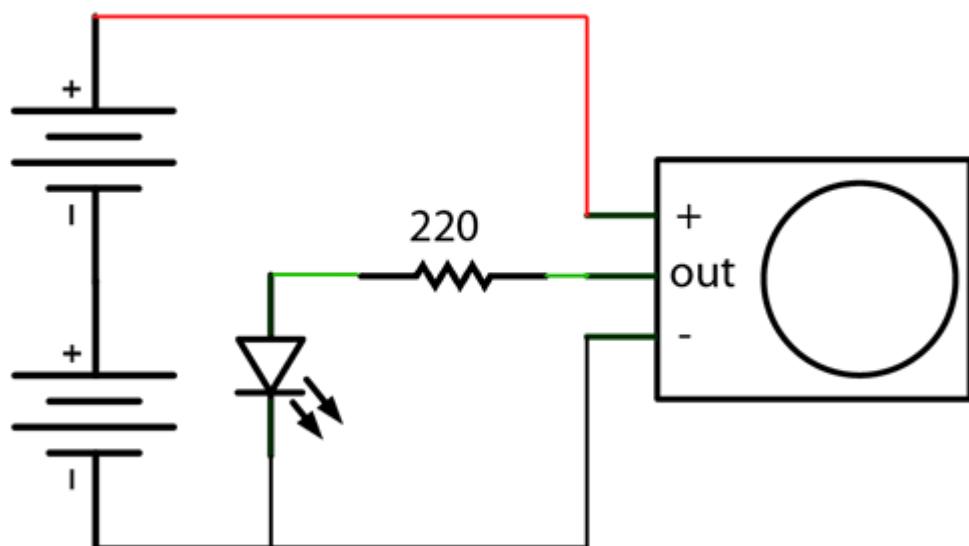
PIR SENSOR:

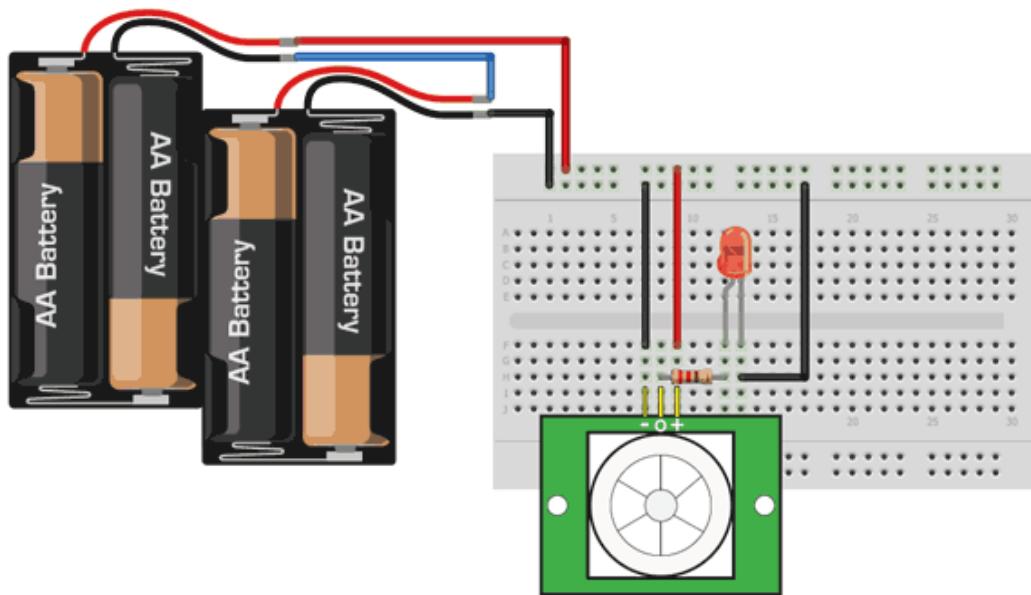
PIR sensors are more complicated than many of the other sensors explained in this tutorial (like photocells, FSRs and tilt switches) because there are multiple variables that affect the sensors input and output. .

The PIR sensor itself has two slots. Each slot is made of a special material that is sensitive to IR. The lens used here is not really doing much and so we see that the two slots can 'see' out past some distance (basically the sensitivity of the sensor). When the sensor is idle, both slots detect the same amount of IR, the ambient amount radiated from the room or walls or outdoors. When a warm body like a human or an animal passes by, it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves. When the warm body leaves the sensing area, the reverse happens, whereby the sensor generates a negative differential change. These change pulses are what is detected.



TESTING PIR SENSOR MODULE:





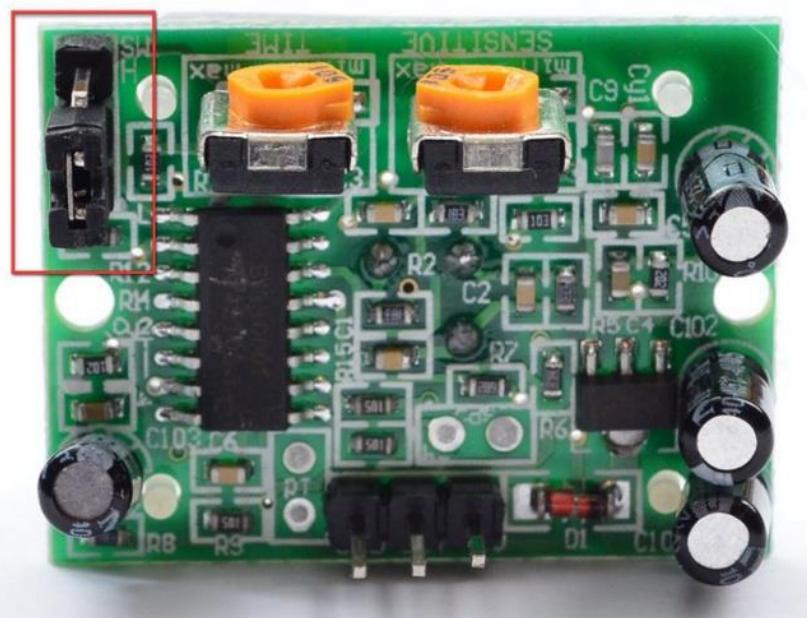
Now when the PIR detects motion, the output pin will go "high" to 3.3V and light up the LED!

Once you have the breadboard wired up, insert the batteries and wait 30-60 seconds for the PIR to 'stabilize'. During that time the LED may blink a little. Wait until the LED is off and then move around in front of it, waving a hand, etc, to see the LED light up!

Retriggering

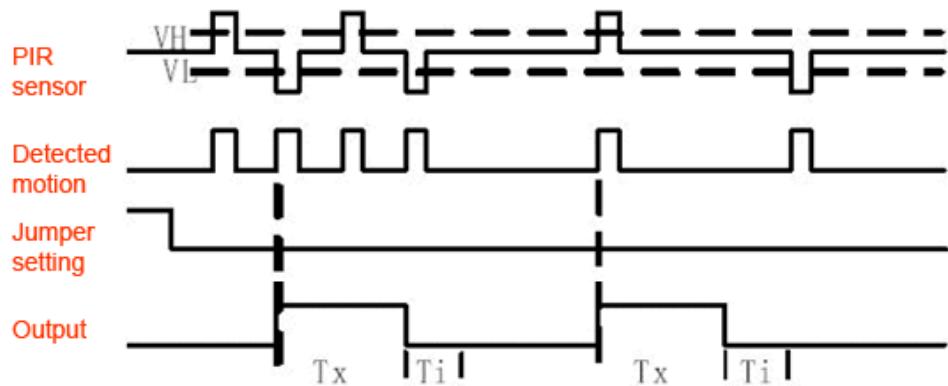
There are a couple options you may have with your PIR. The first is the 'retriggering' option.

Once you have the LED blinking, look on the back of the PIR sensor and make sure that the jumper is placed in the L position as shown below.

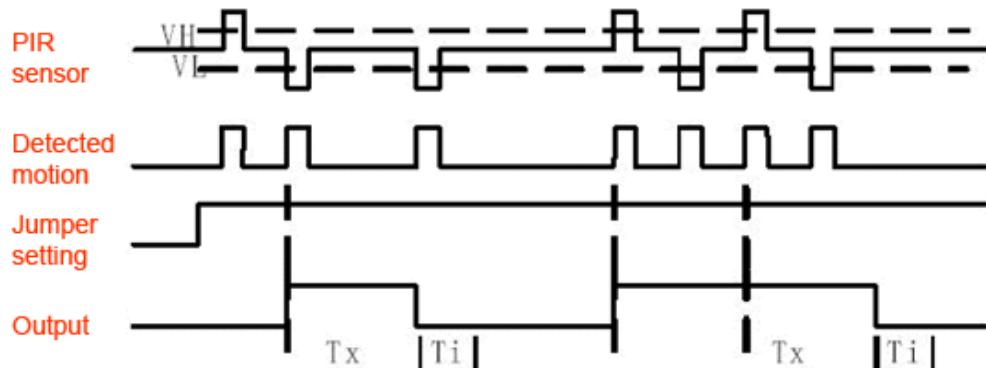


Now set up the testing board again. You may notice that when connecting up the PIR sensor

as above, the LED does not stay on when moving in front of it but actually turns on and off every second or so. That is called "non-retriggering".

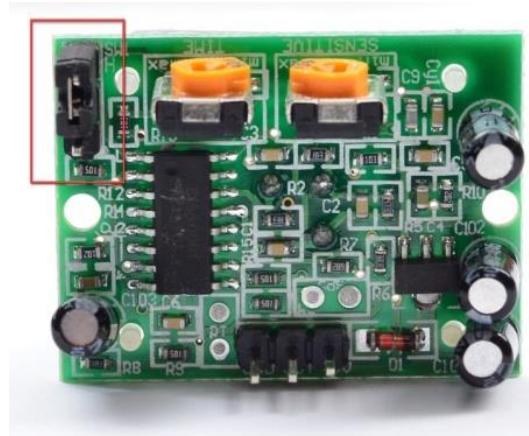


Change the jumper so that it is in the H position. If you set up the test, you will notice that now the LED does stay on the entire time that something is moving. That is called "retriggering".



(The graphs above are from the BISS0001 datasheet.)

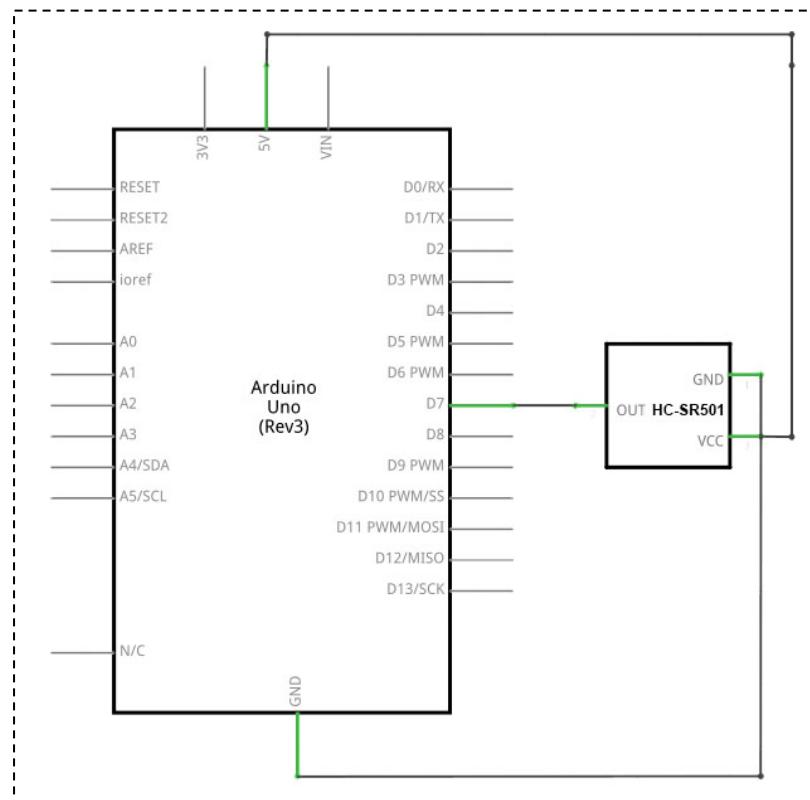
For most applications, "retriggering" (jumper in H position as shown below) mode is a little nicer.



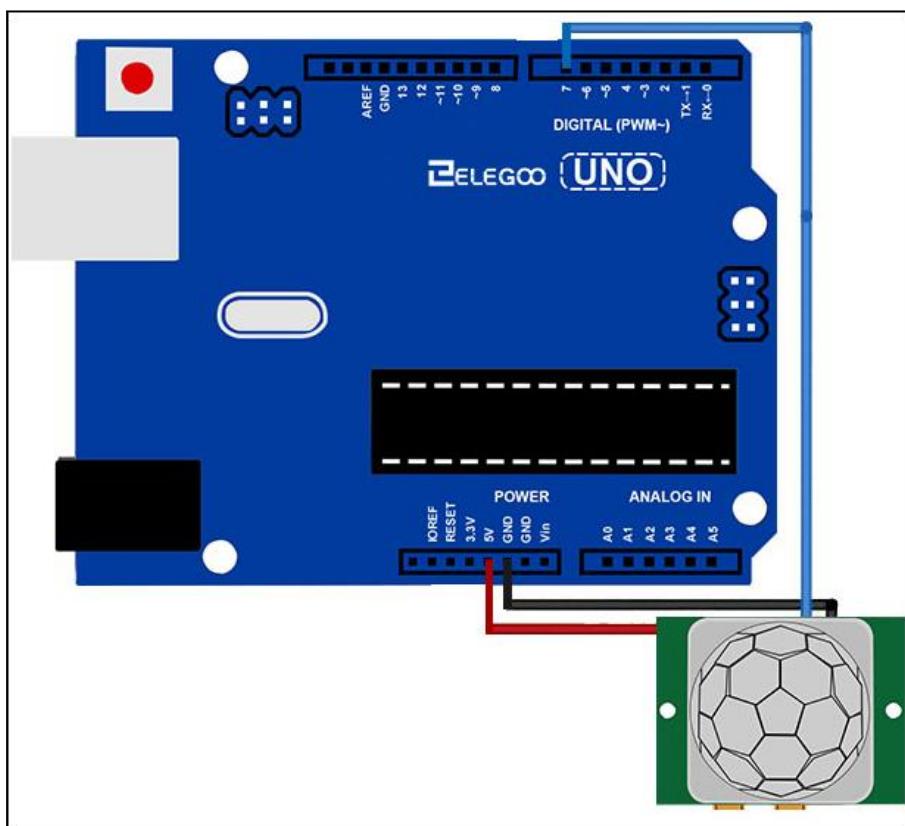
If you need to connect the sensor to something edge-triggered, you'll want to set it to "non-retriggering" (jumper in L position).

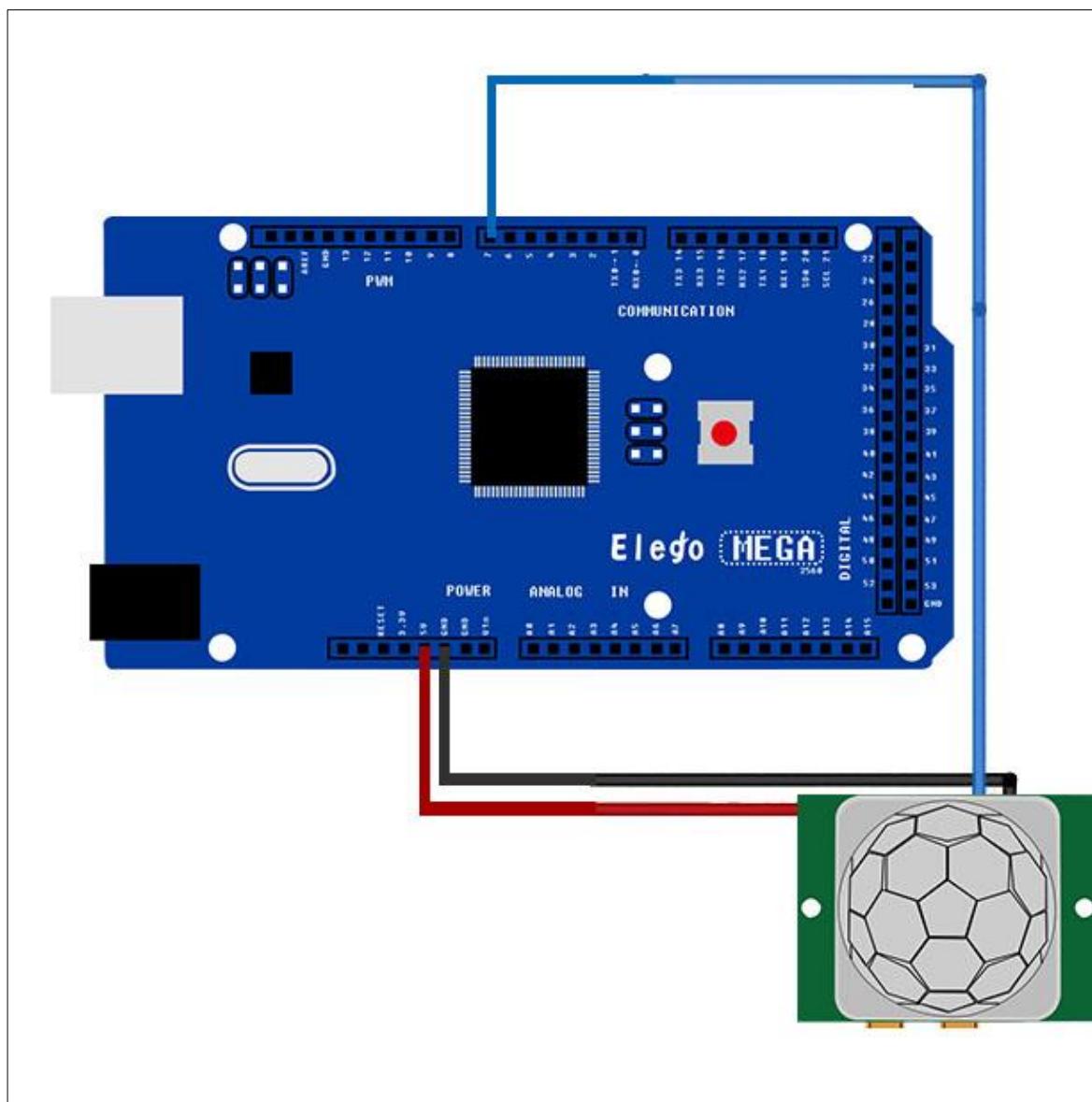
Connection

Schematic



wiring diagram





Connecting PIR sensors to a microcontroller is really simple. The PIR acts as a digital output so all you need to do is listen for the pin to flip high (detected) or low (not detected).

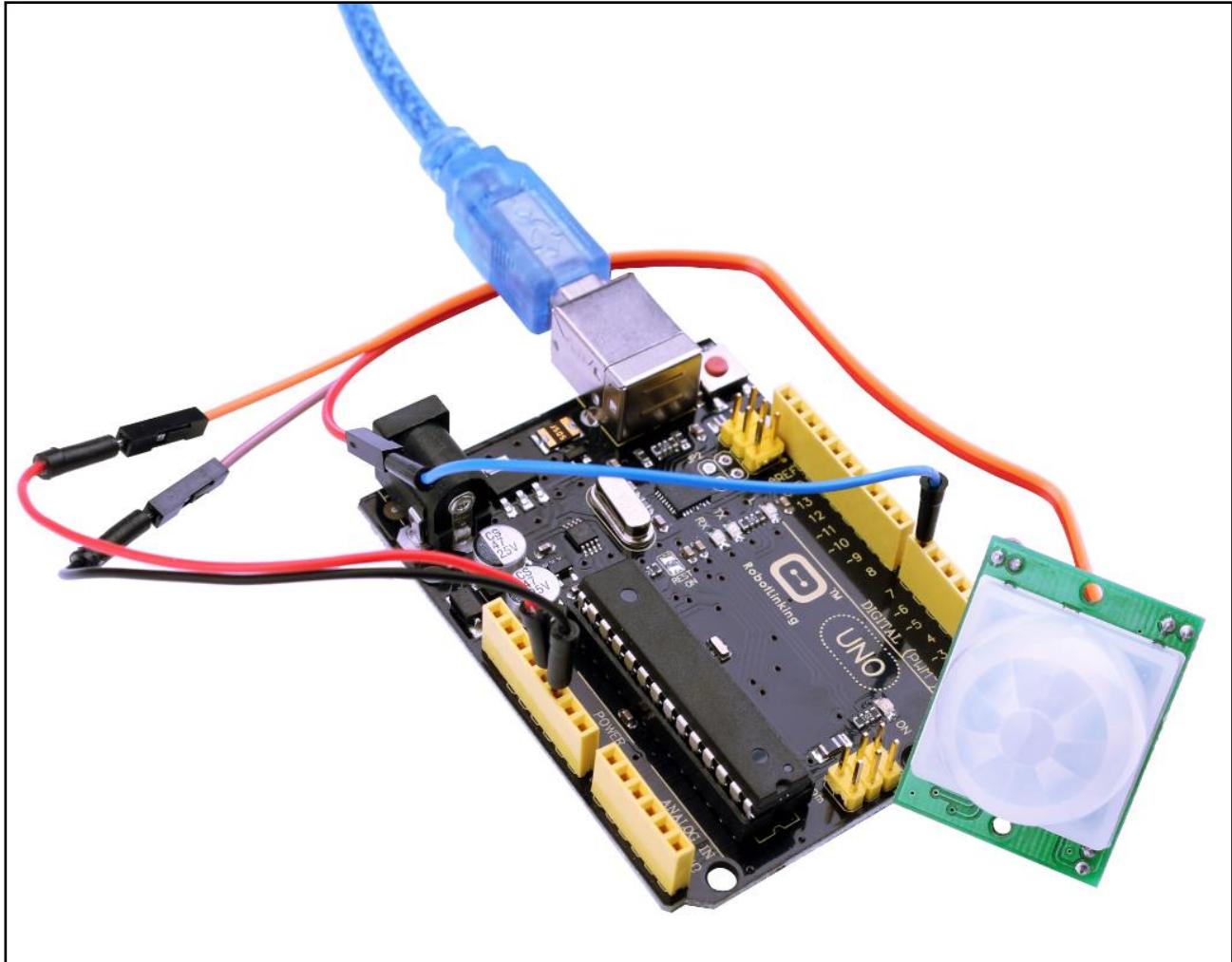
It's likely that you'll want rerigging, so be sure to put the jumper in the H position!

Power the PIR with 5V and connect ground to ground. Then connect the output to a digital pin. In this example, we'll use pin 2.

Code

The code is very simple, and it basically just keeps track of whether the input to pin 2 is high or low. It also tracks the state of the pin, so that it prints out a message when motion has started and stopped.

Don't forget that there are times when you don't need a microcontroller. A PIR sensor can be connected to a relay (perhaps with a transistor buffer) without a micro!



Lesson 29 Water Level Detection Sensor Module

Overview

In this lesson, you will learn how to use a water level detection sensor module.

This module can perceive the depth of water and the core component is an amplifying circuit which is made up of a transistor and several pectinate PCB routings. When put into the water, these routings will present a resistor that can change along with the change of the water's depth. Then, the signal of water's depth is converted into the electrical signal, and we can know the change of water's depth through the ADC function of UNO R3.

Component Required:

- (1) x Elegoo Uno R3
- (1) x Water lever detection sensor module
- (3) x F-M wires

Component Introduction

Water sensor:

A water sensor brick is designed for water detection, which can be widely used in sensing the rainfall, water level, even the liqueate leakage. The brick is mainly composed of three parts: an electronic brick connector, a $1\text{ M}\Omega$ resistor, and several lines of bare conducting wires.

This sensor works by having a series of exposed traces connected to ground. Interlaced between the grounded traces are the sense traces.

The sensor traces have a weak pull-up resistor of $1\text{ M}\Omega$. The resistor will pull the sensor trace value high until a drop of water shorts the sensor trace to the grounded trace. Believe it or not this circuit will work with the digital I/O pins of your UNO R3 board or you can use it with the analog pins to detect the amount of water induced contact between the grounded and sensor traces.

This item can judge the water level through with a series of exposed parallel wires stitch to

measure the water droplet/water size. It can easily change the water size to analog signal, and output analog value can directly be used in the program function, then to achieve the function of water level alarm.

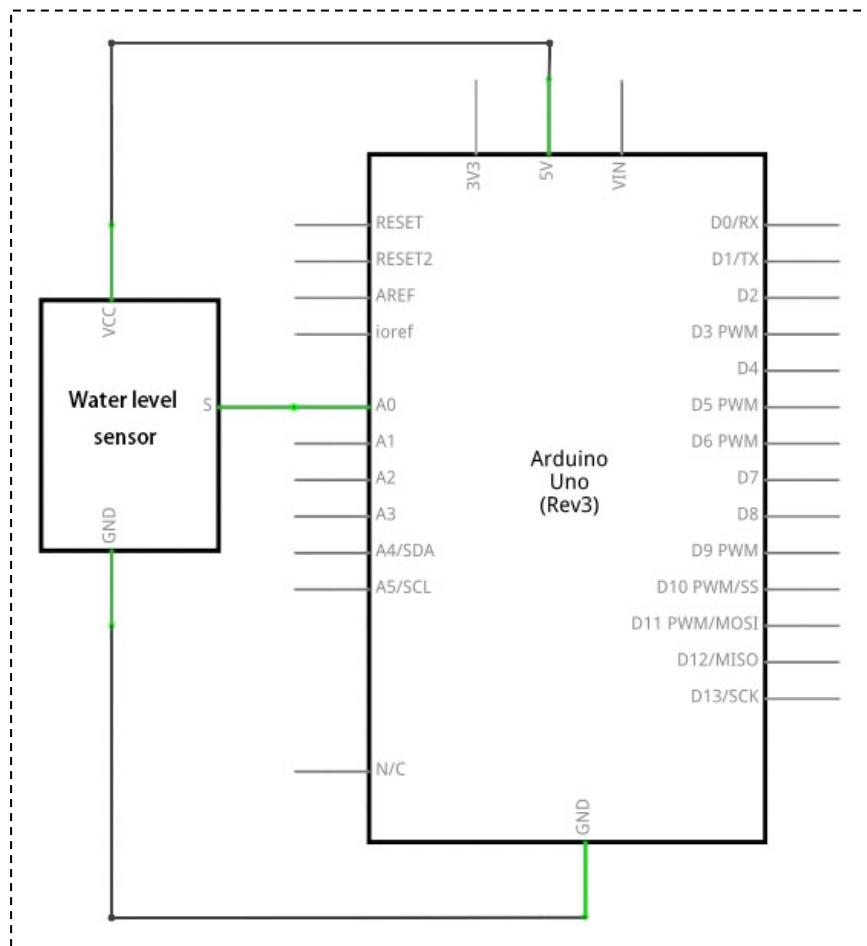
It has low power consumption, and high sensitivity.

Features:

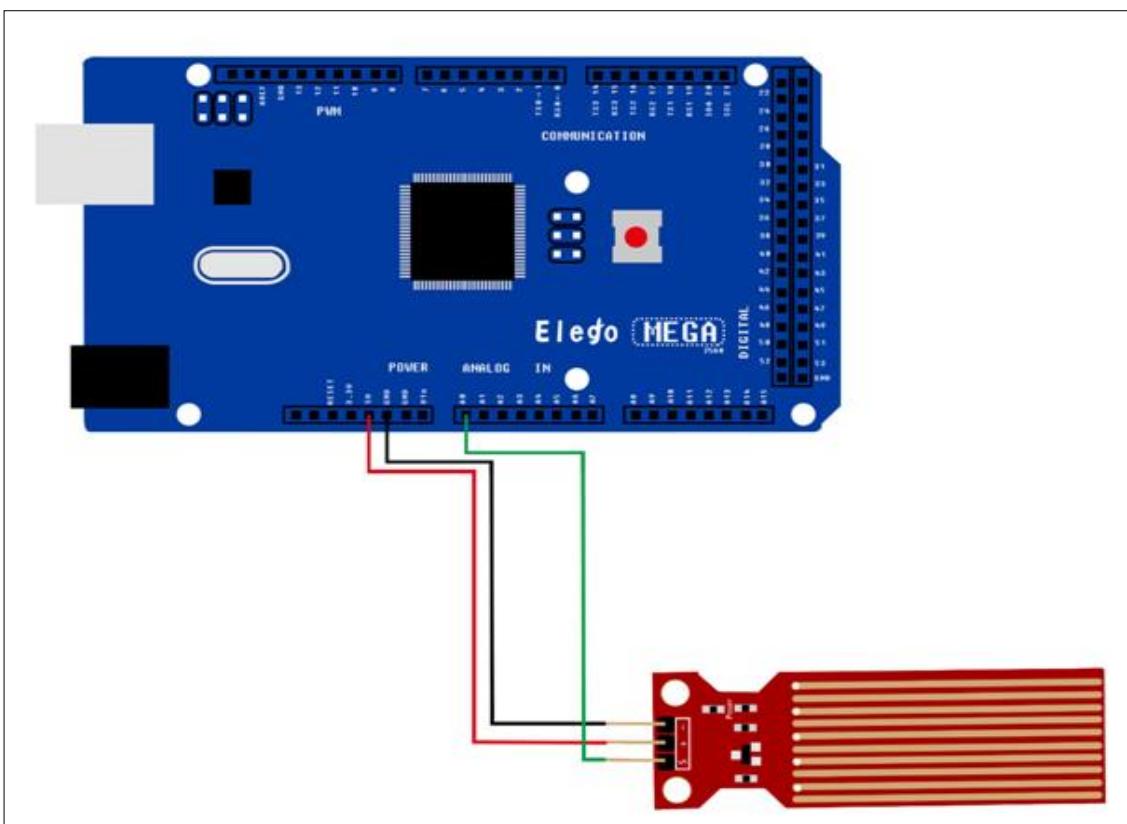
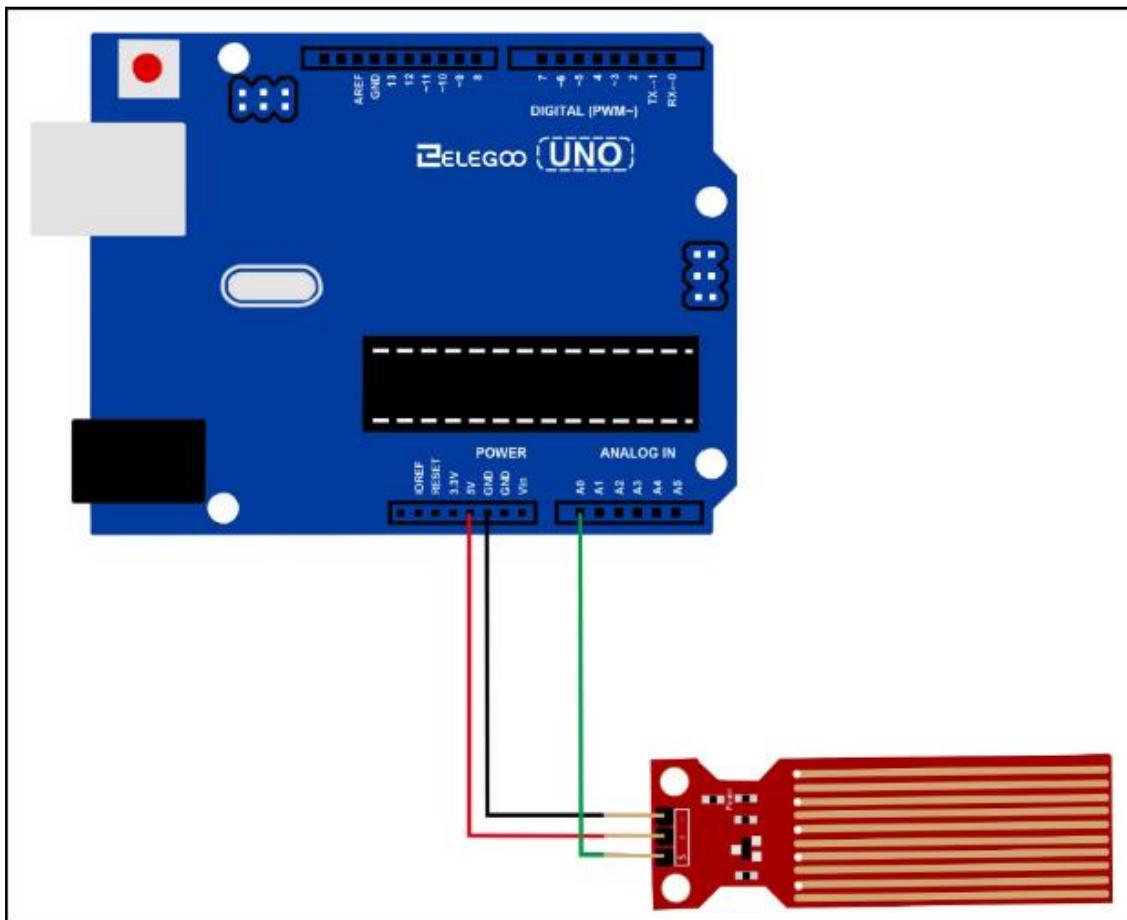
- 1、Working voltage: 5V
- 2、Working Current: <20ma
- 3、Interface: Analog
- 4、Width of detection: 40mm×16mm
- 5、Working Temperature: 10°C~30°C
- 6、Output voltage signal: 0~4.2V

Connection

Schematic



wiring diagram



Wiring tips: Power supply (+) is connected to 5V of UNO R3 board, ground electrode (-) is connected to GND. Signal output (S) is connected to the ports (A0-A5) which have function of inputting analog signal in UNO R3 board, random one is OK, but it should define the same demo code as the routine.

Code

See the code file.



open the monitor you can see the data as below:

ADC14 level is 83
ADC14 level is 94
ADC14 level is 105
ADC14 level is 116
ADC14 level is 127
ADC14 level is 116
ADC14 level is 105
ADC14 level is 94
ADC14 level is 83
ADC14 level is 72
ADC14 level is 61
ADC14 level is 50
ADC14 level is 39
ADC14 level is 28
ADC14 level is 17
ADC14 level is 6
ADC14 level is 17
ADC14 level is 28
ADC14 level is 39
ADC14 level is 50
ADC14 level is 61
ADC14 level is 72
ADC14 level is 83
ADC14 level is 94
ADC14 level is 105
ADC14 level is 116
ADC14 level is 127

自动滚屏 换行符 9600 波特率

Lesson 30 Real Time Clock Module

Overview

In this lesson, you will learn how to use the DS3231, clock module that displays the year, month, day, hour, minute, second and week. Support is via a backup battery trickle charger, which can be used unless being connected to UNO with only three data cables.

Component Required:

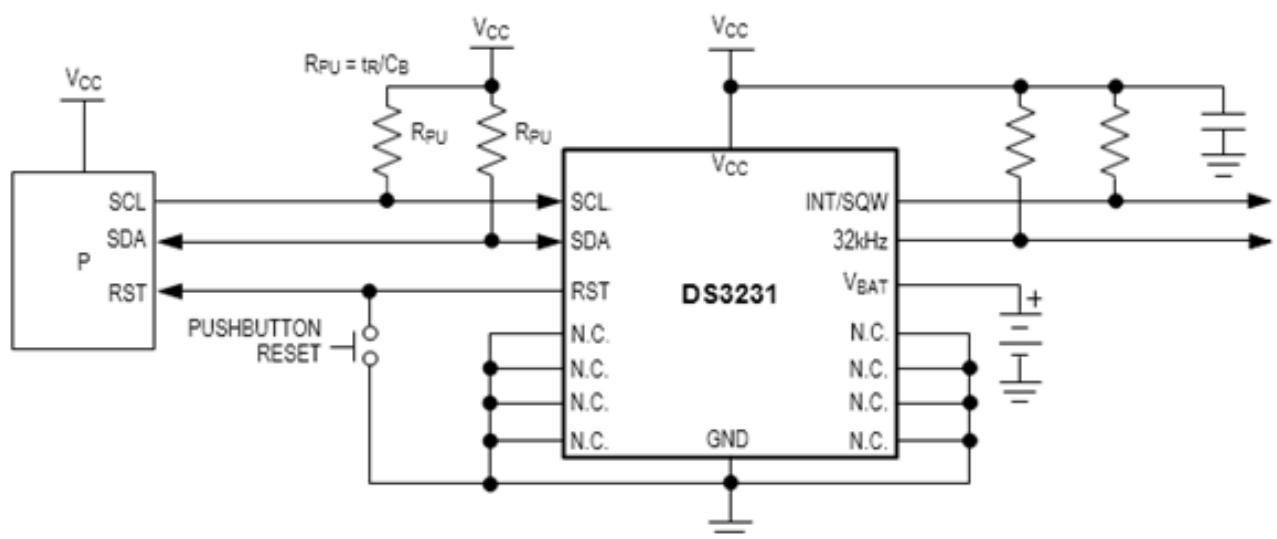
- (1) x Elegoo Uno R3
- (1) x DS3231 real time clock module
- (4) x F-M wires

Component Introduction

DS1302

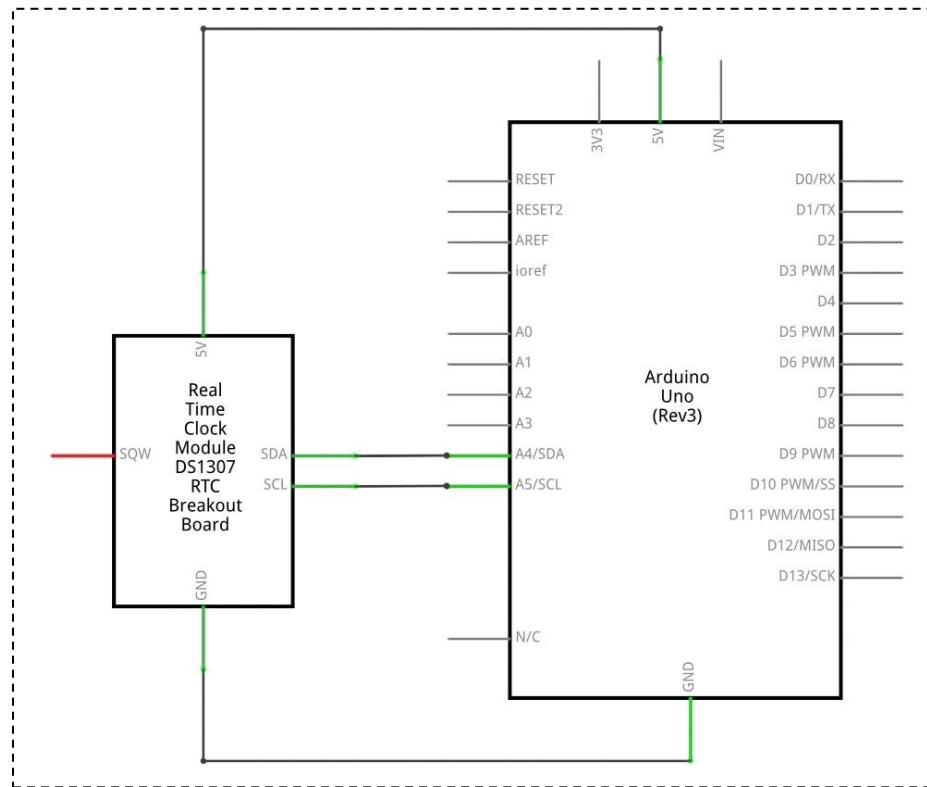
DS3231

The DS3231 is a simple time-keeping chip. It has an integrated battery, so the clock can continue keeping time even when unplugged.

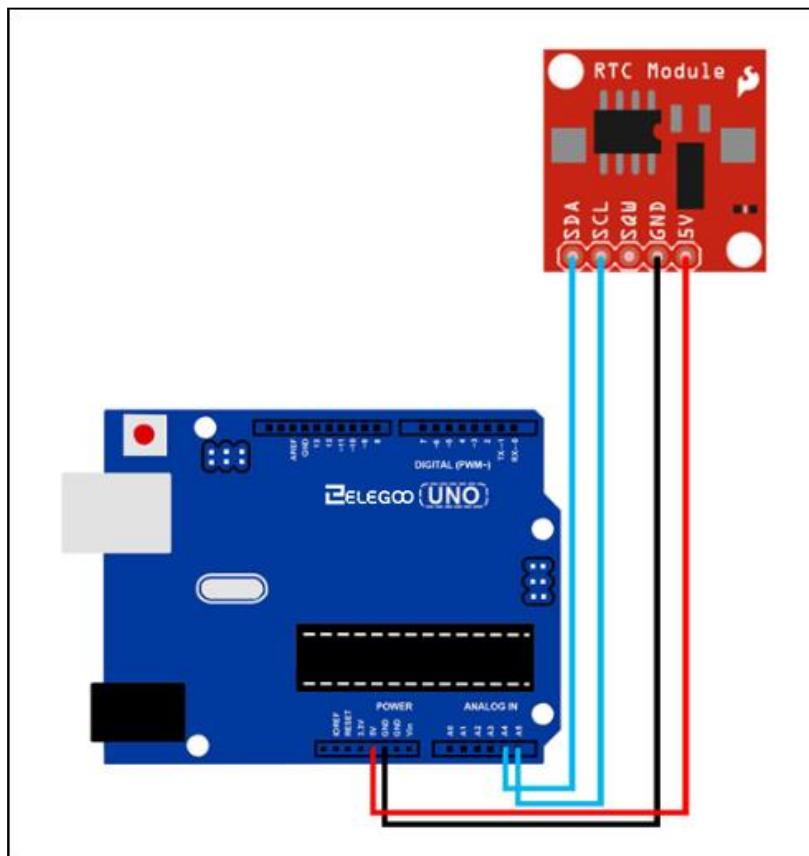


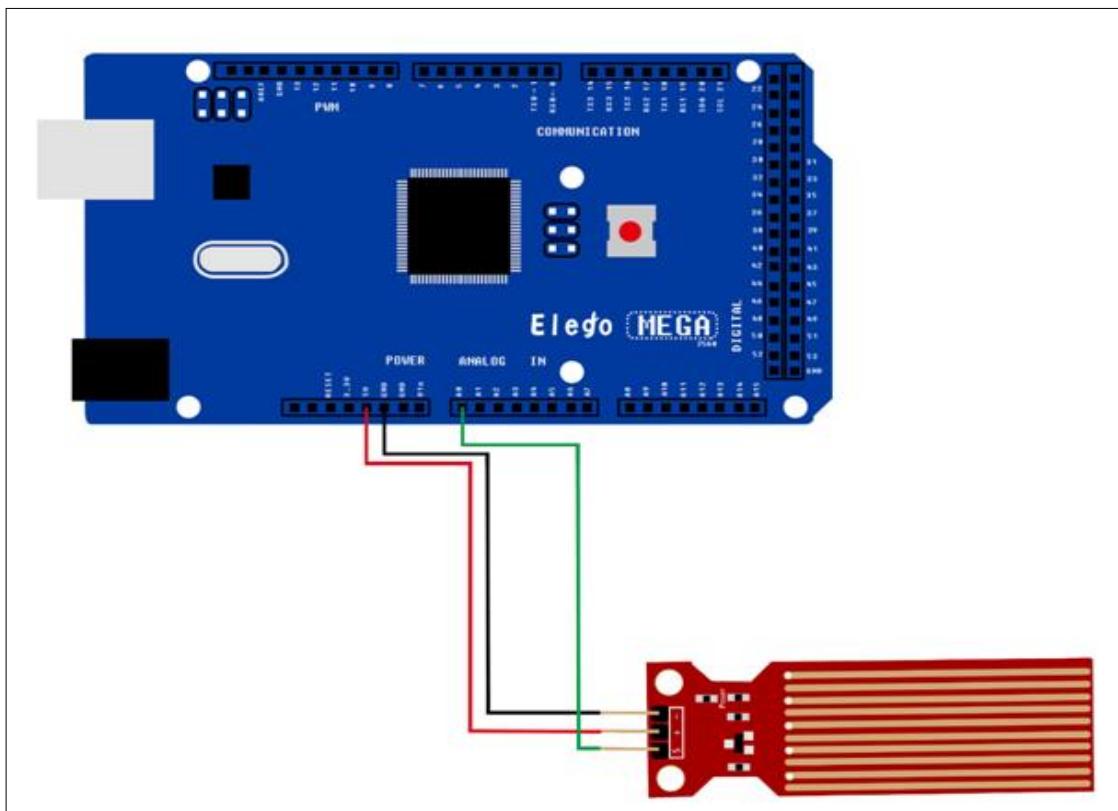
Connection

Schematic



wiring diagram





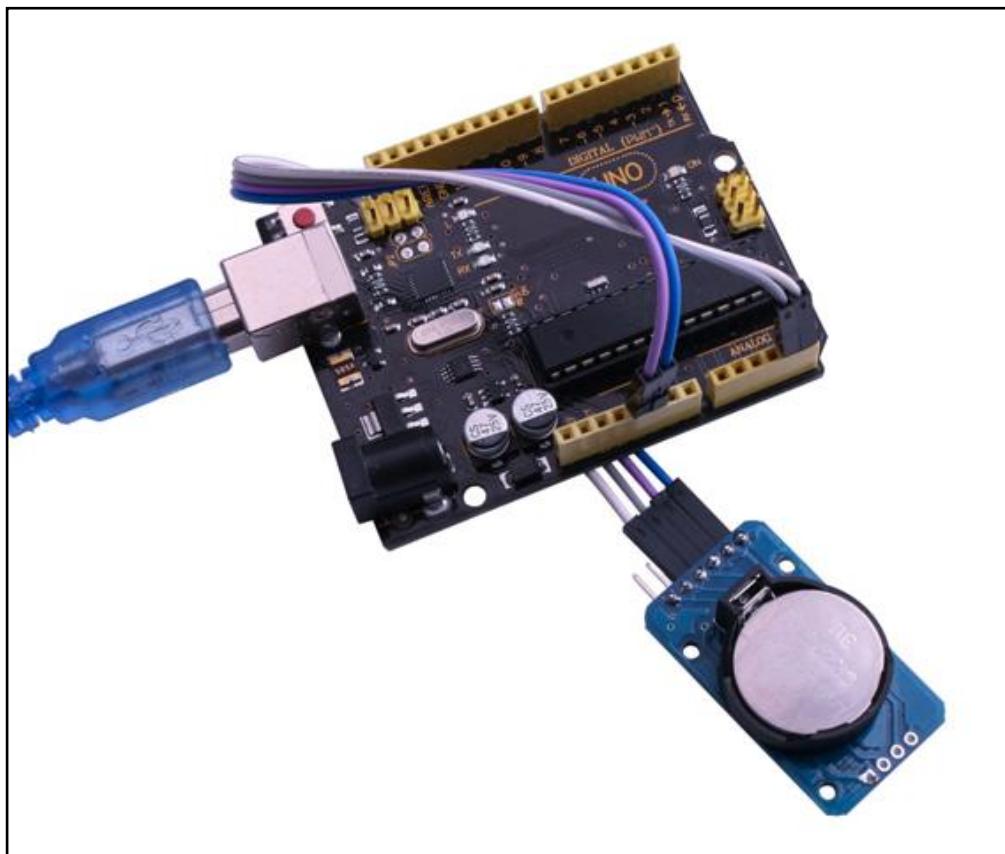
Set up according to the following image.

Ignore the 32K and SQW pins; you will not need them. Plug the SCL pin into your UNO R3 board A5 port, and the SDA pin into the A4 port. The VCC pin plugs into the 5V port, and the GND plugs into the GND port.

Code

Now that we have the physical setup, all we need now is the code.

Before you can run this, make sure that you have installed the <DS3231> <OneWireKeypad> library or re-install it if needed. Otherwise, your code won't work.



open the monitor you can see the module can read the time as below:

```
Raw data: 2016-4-29 14:10:36
Raw data: 2016-4-29 14:10:37
Raw data: 2016-4-29 14:10:38
Raw data: 2016-4-29 14:10:39
Raw data: 2016-4-29 14:10:40
Raw data: 2016-4-29 14:10:41
Raw data: 2016-4-29 14:10:42
Raw data: 2016-4-29 14:10:43
Raw data: 2016-4-29 14:10:44
Raw data: 2016-4-29 14:10:45
Raw data: 2016-4-29 14:10:46
Raw data: 2016-4-29 14:10:47
Raw data: 2016-4-29 14:10:48
Raw data: 2016-4-29 14:10:49
Raw data: 2016-4-29 14:10:50
Raw data: 2016-4-29 14:10:51
Raw data: 2016-4-29 14:10:52
Raw data: 2016-4-29 14:10:53
Raw data: 2016-4-29 14:10:54
Raw data: 2016-4-29 14:10:55
```

Lesson 31 Keypad Module

Overview

In this project, we will go over how to integrate a keyboard with an UNO R3 board so that the UNO R3 can read the keys being pressed by a user.

Keypads are used in all types of devices, including cell phones, fax machines, microwaves, ovens, door locks, etc. They're practically everywhere. Tons of electronic devices use them for user input.

So knowing how to connect a keypad to a microcontroller such as an UNO R3 board is very valuable for building many different types of commercial products.

At the end when all is connected properly and programmed, when a key is pressed, it shows up at the Serial Monitor on your computer. Whenever you press a key, it shows up on the Serial Monitor. Later, in another project, we will connect the keypad circuit, so that it will get displayed on an LCD. But for now, for simplicity purposes, we start at simply showing the key pressed on the computer.

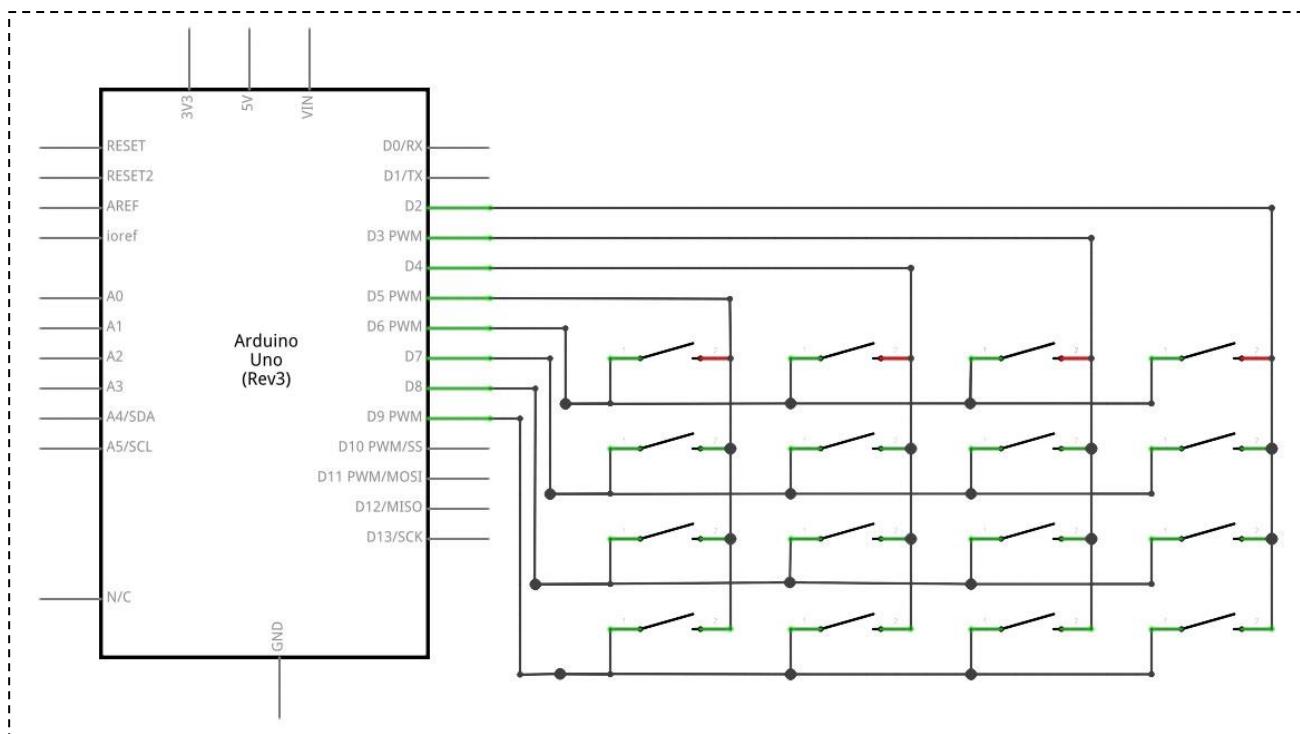
For this project, the type of keypad we will use is a matrix keypad. This is a keypad that follows an encoding scheme that allows it to have much less output pins than there are keys. For example, the matrix keypad we are using has 16 keys (0-9, A-D, *, #), yet only 8 output pins. With a linear keypad, there would have to be 17 output pins (one for each key and a ground pin) in order to work. The matrix encoding scheme allows for less output pins and thus much less connections that have to be made for the keypad to work. In this way, they are more efficient than linear keypads, being that they have less wiring.

Component Required:

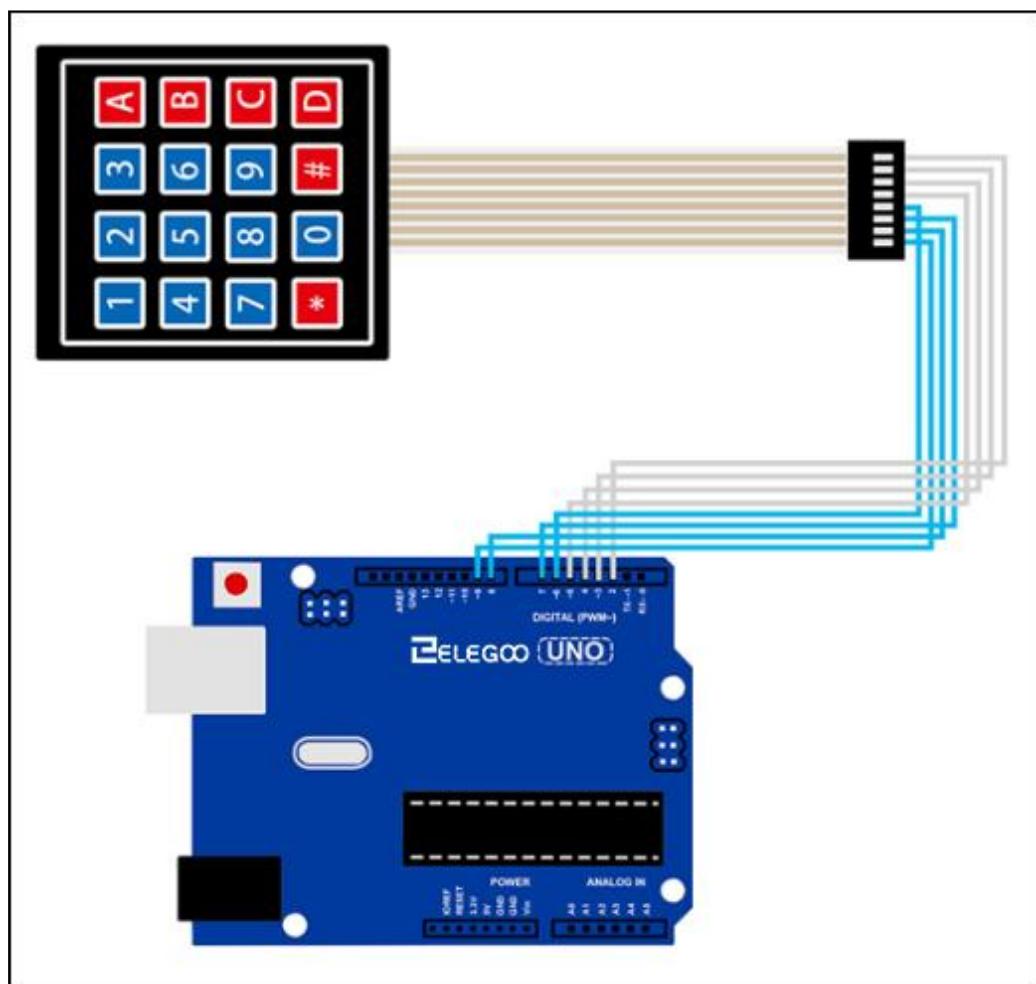
- (1) x Elegoo Uno R3
- (1) x Membrane switch module
- (8) x M-M wires

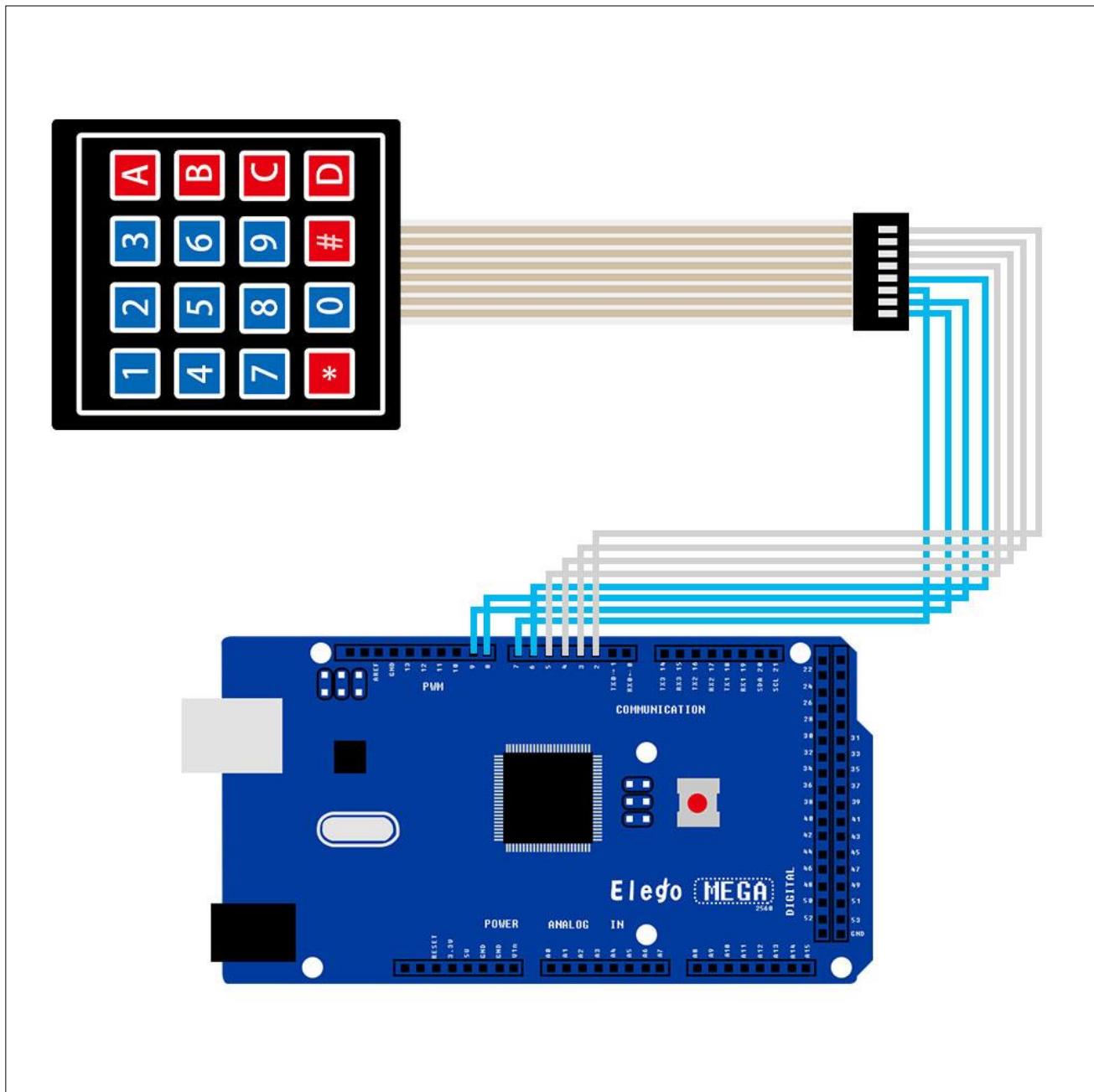
Connection

Schematic



Wiring diagram





When connecting the pins to the UNO R3 board, we connect them to the digital output pins, D9-D2. We connect the first pin of the keypad to D9, the second pin to D8, the third pin to D7, the fourth pin to D6, the fifth pin to D5, the sixth pin to D4, the seventh pin to D3, and the eighth pin to D2.

These are the connections in a table:

Keypad Pin	Connects to Arduino Pin...
1	D9
2	D8
3	D7
4	D6
5	D5
6	D4
7	D3
8	D2

Code

Now that we have the physical setup, all we need now is the code.

Before you can run this, you have to make sure that you have install the <Keypad> library.
Or you need to install again. If you do not do this, your code won't work.



With this code, once we press a key on the keypad, it should show up on the serial monitor of the Arduino software once the code is compiled and uploaded to the UNO R3 board.

