# 2019

# Arduino Starter Kit

Arduino IDE Programming Tutorials





## C programming

1. Hello world	2
2. LED Twinkle	7
3. Analog value	10
4. Advertising lights	14
5. Traffic lights	18
6. Key control	22
7. Answering machine	26
8. Buzzer	30
8.1 Active buzzer	30
8.2 Passive buzzer	34
9. PWM dimming	40
10. Light controlled sound	44
11. Sensible heat light	48
12. 8x8 lattice dot matrix	52
13. Tilt switch	57
14. Flame alarm	61
15. Nixie tube	65
16. Four bit nixie tube	72
17.74HC595	78
18. Servo control	83
19. IR control	88
20. 1602 display	97



### **1-Hello World!**

### The purpose of the experiment:

This course is an experiment that allows Arduino and PC to communicate. The experimental results is to let Arduino say "Hello World!"

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1





### Experimental code analysis:

int val; //Defining variable val

int ledpin=13; //Defining the digital port 13 void setup()

{

Serial.begin(9600); //The baud rate is 9600, which is consistent with software settings. When accessing specific devices (such as Bluetooth), we also have to agree with the baud rate of other devices.

pinMode(ledpin,OUTPUT); //Set the number of 13 ports as output interfaces, and the I/O ports we use on Arduino have similar definitions.

}
void loop()

{

val=Serial.read(); //Read the instructions or characters sent to the Arduino by the PC machine, and assign the instruction or character to val

if(val=='R') //Determine whether the received instruction or character is "R"

{ //If the "R" character is received

digitalWrite(ledpin,HIGH); //Light the number of 13 ports of LED delay(500);

digitalWrite(ledpin,LOW); //Extinguish the number of 13 ports of LED delay(500);

Serial.println("Hello World!"); //Dsiplay "Hello World!"

```
<u>}</u>
```

### Experimental steps:

1. We need to open the code of this experiment: **code-Hello\_world.ino**, click" $\sqrt{}$ "under the menu bar to compile the code, and wait for the word "**Done compiling**" in the lower right corner, as shown in the figure below.





2. In the menu bar of Arduino IDE, we need to select 【Tools】----【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below.





Code-Hello work File Edit Sketch To	I Arduino 1.7.8 Joist Helo	and the first		
code-Hello_wor	Auto Format Archive Sketch Fix Encoding & Relo	Ctrl+T ad		<u>.</u>
int ledpin=	Board	Ctrl+Shift+M	hterface 13	1
void setup	Port Programmer		Serial ports COM1	
pinMode(led	Burn Bootloader	Set the numb	COM15 COM16	s as output
yoid loop()			COM17 COM18	
val=Serial.	read(). //Read	the instru	COM20 COM21	racters sent
* [ in	770etermine al	sether the r	COM22 COM23	nuction or che.

3. After the selection is completed, you need to click " $\rightarrow$ "under the menu bar to upload the code to the Arduino UNO board. When the word "**Done uploading**" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.



4. We need to turn on the serial port monitor in the upper right corner of the Arduino IDE, and a serial port printing box of the Arduino port will appear. The baud rate is set to 9600 in the lower right corner, as shown in the figure below.





5.According to the code, we should enter "R" in the sending box, as shown in the figure below.



6.After clicking "**send**" on the upper right corner , we can receive the Hello World! in the receiving box below, as shown in the figure below.



code-Hello_world   Arduino 1.7. le Edit Sketch Tools Help	8		
© COM6  R		Send	E S
		C( S	unsiste output
		tei	rs sen n or c
🖉 Autoscroll	No line ending	- eat	/ing

Send
Send
consis
s outpu
tare ca
ion or
• eaving



### **2-Led Twinkle**

### The purpose of the experiment:

This course is to use the I/O port on the Arduino UNO board and an external LED light to complete the experiment. The experiment is to make the LED light to twinkle, lights up for 1 second and turns off for 1 second.

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

LED\*1 (Color random)

220Ω Resistor \*1

Breadboard \*1

Dupont line \*1 bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.



#### Experimental code analysis:

int ledPin = 10; //Defining the digital port 10
void setup()

```
{
```

```
pinMode(ledPin, OUTPUT); //Defining the light port for the output port
}
void loop()
```

{

```
digitalWrite(ledPin, HIGH); //Lights up
delay(1000); //delay 1 second
digitalWrite(ledPin, LOW); //Lights out
delay(1000); //delay 1 second
```

```
Experimental steps:
```

### YAHBOOM

1. We need to open the code of this experiment: code-Led\_Twinkle.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below.





3.After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.



4. After the code is uploaded, we can see LED light twinkle every second, as shown in the picture below.





### **3-Analog value**

### The purpose of the experiment:

In this course, we learn how to use of simulate I/O interface by combining adjustable resistor. Arduino is equipped with 6 analog interfaces, port 0~part 5. They can be reused. In addition to the analog interface function, these 6 interfaces can be used as digital interfaces, No 14-19.

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

Adjustable resistor \*1

Breadboard \*1

Dupont line \*1bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.



#### Experimental code analysis:

int potpin=A0; //Defining the analog port A0
int ledpin=13; //Defining the led port 13
int val=0; //Declarations of temporary variables
void setup()
{
 pinMode(ledpin,OUTPUT); //Defining the light port for the output port
 Serial.begin(9600); //The baud rate is 9600
}
void loop()
{

val=analogRead(potpin); //Reading the voltage value of the A0 port and assign it to val



Serial.println(val); //Sending Val value by serial port

Experimental steps:

}

1. We need to open the code of this experiment: code-Analog\_value.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--- selecting the port that the serial number displayed by the device manager just now, as







3. After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload thecode to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.



4.After the code is uploaded. You need to open the serial port monitor on the top right corner of Arduino IDE, A serial port of Arduino port will appear, as shown in the following figure.When we rotate the adjustable resistor, we can see that the data printed in the serial port monitor will change accordingly, as shown in the following figure.



COME	Local Of Local		COM6		
	Sect	-	1	5 Sec. 2	
002	-		1010	-	
023 h03			1016		
023			1015		Ľ
023		r the outpu	1016		
023			1017		E.
023			1017		L
023			1016		L
023			1015		
022		the A0 port	1016		
023		part	1015		1
023			1017		r
923	-		1013		
	*	<b>1</b>	1016	e)	
Antennall	Fo line sading . 9000 hand	1 Report of the	1015	*	





### **4-Advertisement lights**

### The purpose of the experiment:

This course is to use the led lights programming to achieve the effect of the simulated advertising lights.

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

LED\*6 (Color random)

220Ω Resistor \*1

Breadboard \*1

Dupont line \*1bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.



### Experimental code analysis:

```
int BASE = 2 ; //The first LED I/O port
int NUM = 6; //The total number of LED
int i=0;
void setup()
{
   for (int i = BASE; i < BASE + NUM; i ++)
    {
      pinMode(i, OUTPUT); //Defining the digital I/O port for output port
   }
}</pre>
```



```
void loop()
{
    for (i = BASE; i < BASE + NUM; i ++)
    {
        digitalWrite(i, LOW); //Set the number I/O port to be "LOW", that is, the order
        is extinguished.
        delay(200);
    }
    for (i = BASE; i < BASE + NUM; i ++)
    {
        digitalWrite(i, HIGH); //Set the number I/O foot to be "HIGH", that is, the order is lit.
        delay(200);
    }
}
Experimental steps:
</pre>
```

1. We need to open the code of this experiment: code-Advertisement\_lights.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】--- 【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below.

YAHBOOM

12.16 日 2王 88		
文件(F) 操作(A) 重	11音(∨) 帮助(H)	
(= =>   📷   🖬 🗂	1   48	
🖌 🚔 Xiaozhen		
P C IDE ATA/AT	TAPI 控制器	
p 🔮 Jungo Con	nectivity	
▶ 🛄 处理器		
	(2 L DT)	
A MY BELL (COM	the Second Post (COM15)	
1 Bluetoc	oth Serial Port (COM16)	
Bluetoc	th Serial Port (COM17)	
Bluetoo	oth Serial Port (COM18)	
- 1 Bluetoo	th Serial Port (COM19)	
- 👘 Bluetoo	th Serial Port (COM20)	
Bluetoo	oth Serial Port (COM21)	
Bluetoo	th Serial Port (COM22)	
TT LICE.CE	RIAL CH340 (COM6)	
1030-36	COM1)	
24.000		
D 0 0 0 0 0 0 0 0 0		
6 2 tr t+0 0		
© code Advertisen		
code Advertisen	ent lights   Arduino 1.7.8.	
Code Advertisen	eent lights   Arduino 1.7.8 cols Help Auto Format Ctri+T	- 0 *
Code Advertisen Re Edit Sketch	eent lights   Archuino 1.7.8 Dols Help Auto Format Ctri+T Archive Sketch	
Code Advertisen	ant lights   Arduino 1.7.8 Dols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload	
code-Advertisen ile Edit Sketch To code-Advertiser ant BASE =	ant lights Arduino 1,7,8 Sols Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M	
Code Advertisen Re Edit Sketch To Code Advertiser Int BASE =	ent lights   Archaino 1.7.8 cols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M	
Code Advertisen Re Edit Sketch Code Advertiser int BASE = int NUM = 6	eent lights   Archuino 1.7.8 Dols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board ,	
Code Advertisen Re Edit Sketch To Code-Advertiser int BASE = int NUM = 6 int i=0;	eent lights   Archaino 1.7.8 Dols Help Auto Formsat Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port	Serial ports
2 code Advertisen Re Edit Sketch To Code Advertiser int BASE = int NUM = 6 int i=0 roid setup	eent lights Arduino 1.7.8 sols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board , Port Programmer	Serial ports COM1
© code Advertisen Re Edit Sketch Code Advertiser int BASE = int NUM = 6 int i=0 roid setup	annt lights   Arduino 1,7,8 bols   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootboader	Serial ports COM1 COM6
Code Advertisen Re Edit Sketch Code-Advertiser int BASE = int NUM = 6 int i=0, roid setup {	eent lights   Archaino 1.7.8 Dols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM6 COM15
Code Advertisen Re Edit Sketch Code-Advertiser int BASE = int NUM = 6 int i=0; roid setup for (int i	eent lights   Archaino 1.7.8 Dols Help Auto Formsat Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader = BASE 1 < BASE + NUM 1	Serial ports COM1 COM6 COM15 COM15
Code Advertisen Re Edit Sketch Code-Advertiser int BASE = int NUM = 6 int i=0 roid setup for (int i	eent lights   Arduino 1.7.8 sols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader = BASE, 1 < BASE + NUM, 1	Serial ports COM1 COM15 COM15 COM15 COM17
2 code Advertisen Re Edit Sketch T Code Advertiser int BASE = int NUM = 6 int i=0, roid setup [ for (int i { pinMode(	ient lights   Arduino 1.7.8 Sols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader = BASE, 1 < BASE + NUM, 1 1, OUTPUT) / Defining th	Serial ports COM1 COM5 COM15 COM15 COM15 COM17 COM18
D code Advertisen Re Edit Sketch To Code Advertiser int BASE = int NUM = 6 int i=0, roid setup { for (int i { pinMode( }	and lights   Arduino 1.7.8 bols   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader = BASE 1 < BASE + NUM 1 1, OUTPUT), //Defining th	Serial ports COM1 COM6 COM15 COM15 COM17 COM18 COM19
D code Advertisen in Edit Sketch Code-Advertiser int BASE = int NUM = 6 int i=0, roid setup { for (int i } pinMode( }	eent lights   Archaino 1.7.8 Dols Help Auto Formsat Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader = BASE, 1 < BASE + NUM, 1 1, OUTPUT), //Defining th	Serial ports COM1 COM6 COM15 COM15 COM16 COM17 COM18 COM18 COM19 COM20
© code Advertisen The Edit Sketch () Code Advertisen int BASE = int NUM = 6 int i=0; roid setup { for (int i } pinMode() }	eent lights   Arduino 1.7.8 DOIS Help Auto Formsat Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Skift+M Board Port Programmer Burn Bootloader = BASE, 1 < BASE + NUM, 1 1, OUTPUT), //Defining th	Serial ports COM1 COM6 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21
© code Advertisen Re Edit Sketch () Code Advertisen int BASE = int NUM = 6 int i=0; roid setup { for (int i { pinMode( }	eent lights   Arduino 1.7.8 DOIS   Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader = BASE, 1 < BASE + NUM, 1 1, OUTPUT), //Defining th	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21 COM21 COM22
code Advertisen Re Edit Sketch ( Code Advertiser Int BASE = Int NUM = 6 Int i=0, Toid setup for (int i { pinMode( }	and lights   Arduino 1.7.8 Sols Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader = BASE, 1 < BASE + NUM, 1 i, OUTPUT), //Defining th	Serial ports COM1 COM15 COM15 COM15 COM17 COM18 COM19 COM20 COM21 COM21 COM23

3. After the selection is completed, you need to click " $\rightarrow$ " "under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded, we can see that 6 LED lights are turned on successively and then turned off successively, as shown in the figure below.





### **5-Traffic lights**

### The purpose of the experiment:

This course uses led lights programming to realize the effect of simulated traffic lights.

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

LED\*3 (Color random)

220Ω Resistor \*3

Breadboard \*1

dupont line \*1bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.



### Experimental code analysis:

```
int redled =10; //Defining the digital port 10
int yellowled =7; //Defining the digital port 7
int greenled =4; //Defining the digital port 4
void setup()
{
    pinMode(redled, OUTPUT);//Defining the red light port for the output port
```

```
pinMode(realed, COTPOT),//Defining the realignt port for the output port
pinMode(yellowled, OUTPUT); //Defining the yellow light port for the output port
pinMode(greenled, OUTPUT); //Defining the green light port for the output port
}
```

```
void loop()
```



```
{
digitalWrite(redled, HIGH);//Light up red light
delay(1000);
digitalWrite(redled, LOW); //Extinguish red light
digitalWrite(yellowled, HIGH);//Light up yellow light
delay(200);
digitalWrite(yellowled, LOW);//Extinguish yellow light
digitalWrite(greenled, HIGH);//Light up green light
delay(1000);
digitalWrite(greenled, LOW);//Extinguish green light
}
```

Experimental steps:

1. We need to open the code of this experiment: code-Traffic\_lights.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--selecting the port that the serial number displayed by the device manager just now, as shown in the figure below.

YVHSOON

HAR DOUBLE COMPANY AND AND A DOUBLE OF A				
文件(F) 操作(A) 3	查看(V) 帮助(H)			
Þ 🔿   🗊   🖬 🖬				
🖌 🚔 Xiaozhen				
P - IDE ATA/A	ATAPI 控制器			
P 🔮 Jungo Cor	nnectivity			
● ■ 处理器				
◎ 磁盘巡动器				
▲ 一学 )第□ (CON	A TU LPT)			
1 Blueton	oth Serial Port (COM15)			
1 Blueto	oth Serial Port (COM10)			
19 Blueto	oth Serial Port (COM18)			
- Bluetor	oth Serial Port (COM19)			
Blueto	oth Serial Port (COM20)			
Blueto	oth Serial Port (COM21)			
- P Blueto	oth Serial Port (COM22)			
Blueton	oth Serial Port (COM23)			
USB-SE	ERIAL CH340 (COM6)			
一下 通信端	COM1)			
▶ 📳 计算机				
Secole Tealfie Lat				
le Edit Skatch	ISTARdumo 175			×
Con occarr	ous nep		And in the owner of the owner owner owner own	(TTP)
	Charlen bergernaut	Ciri+1		
	hold format			9
code-Traffic lial	Archive Sketch			9
code-Traffic_lig	Archive Sketch Fix Encoding & Rek	oad		e. Y
code-Traffic_ligi pinMode (y	Archive Sketch Fix Encoding & Rek Serial Monitor	oad Ctrl+Shift+M	the yellow	•
code-Traffic_Hgf pinMode(y pinMode(g	Archive Sketch Fix Encoding & Rele Serial Monitor Board	oed Ctrl+Shift+M	the yellow the green li	1 · g
code-Traffic_Hgl pinMode (y pinMode (g	Archive Sketch Fix Encoding & Rek Serial Monitor Board Port	oad Ctrl+Shift+M	the yellow the green li Serial ports	1 ig
code-Traffic_Hol pinMode (y pinMode (g	Archive Sketch Fix Encoding & Rek Serial Monitor Board Port	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1	
code-Traffic_ligi pinMode (y pinMode (g oid loop	Archive Sketch Fix Encoding & Rek Serial Monitor Board Port Programmer	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1	
code-Traffic_Hgi pinMode (y pinMode (g oid loop	Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 COM1 COM15	
code-Traffic_Hof pinMode (y pinMode (g oid loop() digitalWrit	Archive Sketch Fix Encoding & Rek Serial Monitor Board Port Programmer Burn Bootloader c (redled, HICH)	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 COM15 COM16	
code-Traffic_Ho pirMode (y pinMode (g oid loop digitalWrit delay (1000)	Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader e (redled, HIGH)	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 COM15 COM16 COM17	
code-Traffic_Hgl pinMode (y pinMode (g oid loop() digitalWrit delay (1000) digitalWrit	Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader e (redled, HIGH)	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 COM15 COM15 COM16 COM17 COM18	
code-Traffic_Hgl pinMode (y pinMode (g oid loop digitalWrit delay(1000) digitalWrit	Archive Sketch Fix Encoding & Rek Serial Monitor Board Port Programmer Burn Bootloader e (redled, HICH) ; e (redled, LOW)	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 V COM6 COM15 COM16 COM17 COM18 COM19	
code Traffic_Hol pirMode (y pinMode (g oid loop digitalWrit delay(1000) digitalWrit digitalWrit	Archive Sketch Fix Encoding & Rek Serial Monitor Board Port Programmer Burn Bootloader C (redled, HIGH) () e (redled, LOW) e (yellowled, H)	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 COM15 COM15 COM16 COM17 COM18 COM19 COM20	
code-Traffic_Hg pirMode (y pinMode (g oid loop digitalWrit delay(1000) digitalWrit digitalWrit	Archive Sketch Fix Encoding & Rek Serial Monitor Board Port Programmer Burn Bootloader e (redled, HIGH) c (redled, LOW) e (redled, LOW)	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21	
code-Traffic_ligi pirMode (y pirMode (g oid loop digitalWrit delay(1000) digitalWrit digitalWrit	Archive Sketch Fix Encoding & Rek Serial Monitor Board Programmer Burn Bootloader e (redled, HICH) ( e (redled, LOW) e (yellowled, H	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 COM16 COM15 COM16 COM17 COM18 COM19 COM20 COM21	
code-Traffic_Hgi pinMode(y pinMode(g oid loop() digitalWrit delay(1000) digitalWrit digitalWrit	Archive Sketch Fix Encoding & Rek Serial Monitor Board Programmer Burn Bootloader e (redled, HICH) ( e (redled, LOW) e (yellowled, H	oad Ctrl+Shift+M	the yellow the green li Serial ports COM1 V COM6 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21 COM21 COM22	

3. After the selection is completed, you need to click " $\rightarrow$ " "under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded, we can see that the red light is on for 1 second, the yellow light is on for 0.2 seconds, and the green light is on for 1 second, as shown in the figure below.





### 6-Key control

### The purpose of the experiment:

In this course, we will learn how to use the input function of I/O port of Arduino, that is, to read the output value of the external device. We use a key and a LED light to complete a combination of the input and output experiments.

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

LED\*1 (Color random)

220Ω Resistor \*1

10kΩ Resistor \*1

Key switch \*1

Breadboard \*1

Dupont line \*1 bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.



#### Experimental code analysis:

```
int ledpin=11;//Defining the digital port 11
int inpin=7;//Defining the digital port 7
int val;//Defining variable
void setup()
{
```

pinMode(ledpin,OUTPUT);//Defining the light port for the output port



```
pinMode(inpin,INPUT);//Defining the key port for the input port
```

```
}
void loop()
```

```
{
```

```
val=digitalRead(inpin);//Read the value of the 7 port level to the val
if(val==LOW)//Check whether the button is pressed or not. When the button is
pressed, the light will turn on.
```

```
digitalWrite(ledpin,LOW);
else
digitalWrite(ledpin,HIGH);
```

Experimental steps:

1. We need to open the code of this experiment: code-Key control.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】--- 【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. for example:COM6,as shown in the following figure.

YAHBOOM

」、设备管理器		
文件(F) 摄作(A) 引	登着(V) 解助(H)	
🗢 🔿 🔟 🔟 🖬		
4 🔬 Xiaozhen		
D Ca IDE ATA/A	TAPI 控制器	
🕴 🔮 Jungo Con	nectivity	
2 🛄 处理器		
で 花田窓の話	-	
A ····································	HILLPT)	
T Bluetoc	wh Serial Port (COM15)	
T Bluetoc	sth Serial Port (COM17)	
- Bluetoc	oth Serial Port (COM18)	
- Bluetoc	oth Serial Port (COM19)	
- P Bluetoc	oth Serial Port (COM20)	
- T Bluetoc	oth Serial Port (COM21)	
- Bluetoc	oth Serial Port (COM22)	
- Bluetoc	oth Serial Port (COM23)	
USB-SE	RIAL CH340 (COM6)	
© code Key contro ile Edit Sketch T	Arduino 1.7.8 pols Help	
OO DE	Auto Format Ctrl+T	
	Archive Sketch	1
code-keA_couta	Fix Encoding & Reload	
{	Serial Monitor Ctrl+Shift	t+M
minWed- (1		
primode (1	Board	,light port for th
pinMode(i	Board Port	light port for th
pinMode(i }	Board Port	Light port for th Serial ports le in COM1
pinMode(i }	Board Port Programmer	Light port for th Serial ports we in COM1
pinMode(i pinMode(i } void loop()	Board Port Programmer Burn Bootloader	, light port for th Serial ports we in COM1 ✓ COM6 COM15
pinMode(i pinMode(i } void loop() {	Board Port Programmer Burn Bootloader	↓ light port for th Serial ports we in COM1 ✓ COM6 COM15 COM15
<pre>pinMode(i pinMode(i } void loop() { val=digital</pre>	Port Programmer Burn Bootloader	Light port for th Serial ports be in COM1 ✓ COM6 COM15 COM15 COM16
<pre>pinMode(i pinMode(i } void loop() { val=digital if(val==100000000000000000000000000000000000</pre>	Board Port Programmer Burn Bootloader Read (inpin) : //Read the v	Light port for the Serial ports be in COMI COMIS COMIS COMIS COMIS COMIS COMIS COMIS
<pre>pinMode(i pinMode(i } void loop() { val=digital if(val==L0% digital)</pre>	Board Port Programmer Burn Bootloader Rend (inpin) : //Read the y ) //Check whether the but	Light port for the Serial ports be in COM1 COM6 COM15 COM15 COM16 COM16 COM17 ort 1 tor COM18 or no COM19
<pre>pinMode(i pinMode(i } void loop() {  val=digital  if(val==L0W  digitalWg</pre>	Board Port Programmer Burn Bootloader Read (inpin) //Read the v () //Check whether the but Ite(ledpin,LGW) .	Light port for the Serial ports be in COM1 COM6 COM15 COM16 COM16 COM16 COM17 ort 1 COM18 or no COM19 COM20
<pre>pinMode(i pinMode(i } void loop() {  val=digital  if(val==L0W       digitalWe , also</pre>	Board Port Programmer Burn Bootloader Read (inpin) : //Read the v ) //Check whether the but Ite(ledpin,LOW) :	Light port for th Serial ports be in COM1 ✓ COM6 COM15 COM15 COM16 COM17 ort 1 tor COM18 or no COM19 COM20 COM21
<pre>pinmode(1 pinMode(i } void loop() {  val=digital  if(val==LOW       digitalWe       slime </pre>	Board Port Programmer Burn Bootloader Read (inpin) : //Read the w ) //Check shether the but Ite (ledpin, LOW) : "	Light port for th Serial ports be in COM1 V COM6 COM15 COM15 COM16 COM16 COM17 art 1 COM18 r no COM19 COM20 COM21 ,
<pre>pinMode(i pinMode(i } void loop() {  val=digital  if(val==LOW     digitalWe     else Done.compling</pre>	Board Port Programmer Burn Bootloader Bend (impin): //Read the v () //Check whether the but Ite (ledpin, LOW):	Light port for th Serial ports be in COM1 COM1 COM15 COM15 COM16 COM16 COM17 Tot COM18 COM19 COM20 COM21 COM21

3. After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded, when the button is pressed, the LED light will be on, and the LED light will be off when the button is released, as shown in the figure below.



### **7-Answering machine**

### The purpose of the experiment:

This experiment is to extend the experiment of button control LED lights in the previous class into 3 buttons corresponding to 3 LED lights, which requires 6 digital I/O interfaces of Arduino.

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

LED\*3 (Color random)

220Ω Resistor \*3

10kΩ Resistor \*3

Key switch \*3

Breadboard \*1

Dupont line \*1bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.



#### Experimental code analysis:

int redled=10; //set IO 10 to red LED int yellowled=9; //set IO 10 to yellow LED int greenled=8; //set IO 10 to green LED int redpin=7; //red key pin IO 7 int yellowpin=6; //yellow key pin IO 6 int greenpin=5; //green key pin IO 5



```
int red:
         //Declarations of variables
int yellow; //Declarations of variables
int green; //Declarations of variables
void setup()
{
 pinMode(redled,OUTPUT); //set as output
 pinMode(yellowled,OUTPUT); //set as output
 pinMode(greenled,OUTPUT); //set as output
 pinMode(redpin,INPUT); //set as input
 pinMode(yellowpin,INPUT); //set as input
 pinMode(greenpin,INPUT); //set as input
}
void loop()
{
 red=digitalRead(redpin);//Reading key state
 if(red==LOW) //Key state is LOW
 { digitalWrite(redled,LOW);}//LED turn off
 else //Key state is HIGH
 { digitalWrite(redled,HIGH);}//LED turn on
 yellow=digitalRead(yellowpin);
 if(yellow==LOW)
 { digitalWrite(yellowled,LOW);}
 else
 { digitalWrite(yellowled,HIGH);}
 green=digitalRead(greenpin);
 if(green==LOW)
 { digitalWrite(greenled,LOW);}
 else
 { digitalWrite(greenled,HIGH);}
}
```

Experimental steps:

1. We need to open the code of this experiment: code-Answering machine.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.





2.In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. for example:COM6,as shown in the following figure.

and the state		
	00   1Q	
🚔 Xiaozhen		
P CA IDE ATA	/ATAPI 控制器	
Jungo Ci	onnectivity	
	-	
Blueb	ooth Serial Port (COM15)	
- T Blueb	ooth Serial Port (COM16)	
- 1 Bluet	ooth Serial Port (COM17)	
-11 Bluet	ooth Serial Port (COM18)	
- Bluet	ooth Serial Port (COM19)	
Blueb	ooth Serial Port (COM20)	
THE Bluet	ooth Serial Port (COM21)	
T Bluet	ooth Serial Port (COM23)	
- TP USB-	SERIAL CH340 (COM6)	
一带 通信的	#□ (COM1)	
▶ 🚛 计算机		
code-Answering	g machine   Arduino 1.7.8	
code Answering Edit Sketch	g machine   Arduino 1.7,8 Fools_ Help	
e Edit Sketch	a machine   Arduino 1.7.8 Fools   Help Auto Format Ctrl+T	
code Answering e Edit Sketch	Auto Format Ctrl+T Archive Sketch	
code Answering e Edit Sketch COLOF Lode Answering	Auto format Archive Sketch Fix Encoding & Reload	
code Answering e Edit Sketch COLOT Code Answering at red,	g machine   Arduino 1.7.8 Tools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi	ift+M
code Answering e Edit Sketch COLLE code Answering at red, at yellow,	g machine   Arduino 1.7.8 Tools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board	ft+M
code Answering code Answering code Answering at red, at yellow, at green;/	Content of the second of the s	ift+M
code Answering e tdit Sketch College code Answering at red, at yellow, at green / rid setup	Auto format Ctrl+T Auto format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port	ft+M Serial ports COM1
code Answering E dit Skotch COLOFE code Answering at red, at yellow, at green, id setup	g machine   Arduino 1.7.8 Tools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Rootloadar	ft+M Serial ports COMI
code Answering e Edit Sketch COLOT code Answering nt red, nt yellow, nt green; / pid setup	g machine   Arduino 1.7.8 Tools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader	ft+M Serial ports COME COME COMIS
code Answering e Edit Sketch Collect code Answering at red, at yellow, at green./ Hid setup pinMode (res	a machine   Arduino 1.7.8 Tools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader dled, OUTPUT) //set of	ft+M Serial ports COMI COMIS COMIS COMIS
code Answering e Edit Skotch Collection code Answering at red, at yellow, at green,/ oid setup pinMode (res pinMode (ye	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader dled, OUTPUT) //set of Ilowled, OUTPUT) //set	ft+M Serial ports COM1 COM5 COM15 COM15 COM15 COM16 COM17
code Answering e tolt Sketch code Answering nt red, nt yellow, nt green, / oid setup pinMode (re pinMode (gre	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader dled, OUTPUT) //Set eenled. OUTPUT) //Set	ft+M Serial ports COM1 COM5 COM15 COM15 COM15 COM15 COM15 COM15 COM15 COM15
code Answering Edit Skotch Code Answering to red, it yellow, it green, / id setup pinMode (re pinMode (gre pinMode (gre	g machine   Arduino 1.7.8 Tools Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader dled, OUTPUT) //Set of Ilowled, OUTPUT) //Set of pin, INPUT) //Set of	ft+M Serial ports COM1 ✓ COM6 COM15 COM15 COM16 COM16 COM18 COM18 COM19
code Answering Edit Skotch Code Answering to the red; it yellow, it green; / id setup pinMode (rec pinMode (gre pinMode (gre pinMode (rec pinMode (rec	a machine   Arduino 1.7.8 Tools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader dled, OUTPUT) //Set of llowled, OUTPUT) //Set of dpin, INPUT) //Set of llowing, INPUT)	ft+M Serial ports COM1 COM15 COM15 COM15 COM16 COM16 COM17 COM18 COM19 COM20
code Answering e Edit Sketch Collection code Answering at red, at yellow, at green; / rid setup pinMode (res pinMode (gre pinMode (gre pinMode (gre pinMode (res pinMode (gre	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader dled, OUTPUT) //Set of Ilowled, OUTPUT) //Set of thenled. OUTPUT) //Set of	ft+M Serial ports COMI COMIS
code Answering e Edit Sketch Collection code Answering at red, at yellow, at green; / rid setup pinMode (rec pinMode (gre pinMode (gre pinMode (gre	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shi Board Port Programmer Burn Bootloader dled, OUTPUT) //set of Howled, OUTPUT) //set of thin, INPUT) //set of thompin, INPUT) //set of thompin, INPUT) //set of	H+M Serial ports COMI COMIS

3.After the selection is completed, you need to click " $\rightarrow$ "under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded, when different buttons are pressed, the LED lights of different colors will be turned on and the LED lights will be extinguished when the button is released, as shown in the figure below.





### 8.1-Active buzzer

### The purpose of the experiment:

In this course, we need to use active buzzer to make an experiment to make the circuit sound.

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω Resistor \*1

Active buzzer \*1

Breadboard \*1

Dupont line \*1bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.

Note: The active buzzer has positive and negative electrode. The actual object diagram below shows that the buzzer has positive and negative marks.





### Experimental code analysis:

int buzzer=8; //Defining the digital port 8 to control the buzzer int i = 0; void setup()



```
{
pinMode(buzzer,OUTPUT); //Defining the buzzer port for the output port
}
void loop()
{
  for(i=0;i<80;i++) //Output a frequency of sound
  {
   digitalWrite(buzzer,HIGH); //Sound
   delay(1);
   digitalWrite(buzzer,LOW); //Unsound
   delay(1);
  }
  for(i=0;i<100;i++) //Output another frequency of sound
  {
   digitalWrite(buzzer,HIGH); //Sound
   delay(2);
   digitalWrite(buzzer,LOW); //Unsound
   delay(2);
  }
```

Experimental steps:

1. We need to open the code of this experiment: code-Active\_buzzer.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2.In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

YVHSOON

文件(F) 操作(A)			
	查看(V) 帮助(H)		-
** 🖬 🖬	m   🛤		
A Xiaozhen			
De ata	/ATAPI 控制器		
👂 🔮 Jungo C	onnectivity		
🕨 🛄 处理器			
▶ 磁盘驱动	25 C		
▲ 🦙 第□ (CO	M和LPT)		
Bluet	ooth Serial Port (COM15)		
Bluet	ooth Serial Port (COM16)		
Bluet	ooth Serial Port (COM17)		
Till of a	ooth Serial Port (COM18)		
TE Pluet	ooth Serial Port (COM19)		
TT Bluet	noth Serial Port (COM21)		
Bluet	ooth Serial Port (COM22)		
- Bluet	ooth Serial Port (COM23)		
USB-	SERIAL CH340 (COM6)		
一下 通信的	無□ (COM1)		
▷ 📲 计算机			
code Active buzz	er Arduino 1.7.8		
ile Edit Sketch To	als Help		Contraction of Contra
OO BR	Auto Format C	trl+T	10
	Archive Sketch		
Internet of the second			
code-Active_bu	Fix Encoding & Reload		
code-Active_bu delay	Fix Encoding & Reload Serial Monitor C	trl+Shift+M	
code Active_but delay }	Fix Encoding & Reload Serial Monitor C Board	trl+Shift+M	
code-Active_bu delay } for(i=0	Fix Encoding & Reload Serial Monitor C Board Port	trl+Shift+M Serial po	ets found
code Active_bu delay } for(i=0 {	Fix Encoding & Reload Serial Monitor C Board Port	trl+Shift+M Serial po COMI	ets lound
code-Active_bun delay } for(i=0 { digit	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader	tri+Shift+M Serial po COMI V COM6	ets found
code Active_bun delay } for(i=0 { digit	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader	trl+Shift+M Serial po COMI ✓ COM6 COM15	rts Journd
code Active_bun delay } for(i=0 { digit delay(2	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader	trl+Shift+M Serial po COMI V COM6 COM15 COM15	ets found
code Active_bun delay } for(i=0 { digit delay(2 digital	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader !) ; Write (buzzer, LOW)	trl+Shift+M Serial po COM1 V COM6 COM15 COM15 COM16 COM17	rts found
code Active_bui delay } for(i=0 { digit delay(2 digital delay(2	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader 1) ; Write (buzzer, LOW) ; 2) ;	trl+Shift+M Setial po COMI V COM6 COM15 COM15 COM15 COM16 COM17 COM18	ets found
<pre>code Active_bui delay } for(i=0 { digit delay(2 digital delay(2 }</pre>	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader 2); Write (buzzer, LOW) . 2);	trl+Shift+M Serial po COMI COMIS COM15 COM15 COM15 COM16 COM16 COM17 COM18 COM19	ets lound
<pre>code Active_bui delay } for (i=0 { digit delay(2 digital delay(2 })</pre>	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader 2) ; Write (buzzer, LOW) ; 2) ;	trl+Shift+M Serial po COM1 V COM6 COM15 COM15 COM16 COM17 COM18 COM19 COM20	ets Journd
<pre>code Active_bui delay } for (i=0 { digit delay(2 digital delay(2 })</pre>	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader !) ; Write (buzzer, LOW) . !) ;	trl+Shift+M Serial po COM1 COM15 COM15 COM16 COM16 COM17 COM18 COM19 COM20 COM21	ete lound
<pre>code Active_bu delay } for (i=0 {     digit     delay(2     digital     delay(2     digital     delay(2     } }</pre>	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader !) ; Write (buzzer, LOW) ; ?) ;	trl+Shift+M Serial po COM1 ✓ COM6 COM15 COM15 COM16 COM17 COM18 COM19 COM28 COM28 COM28	ets found
<pre>code Active_bu delay } for (i=0 {     digit     delay(2     digital     delay(2     } }</pre>	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader !) ; Write (buzzer, LOW) . ?) ;	tri+Shift+M Serial po COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM17 COM18 COM19 COM20 COM21 COM21 COM23	ets found
<pre>code Active_bui     delay     delay     for (i=0     {         digit         delay(2         digital         delay(2         } } Cone complete </pre>	Fix Encoding & Reload Serial Monitor C Board Port Programmer Burn Bootloader !) ; Write (buzzer, LOW) . 2) ;	tri+Shift+M Serial po COMI V COM6 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21 COM22 COM23	ets found

3.After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded, we can hear the buzzer making two different frequencies in succession, as shown in the following figure.





### 8.2-Passive buzzer

### The purpose of the experiment:

In this course, we need to use passive buzzer to make an experiment to make the circuit sound.

#### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω Resistor \*1

Passive buzzer \*1

Breadboard \*1

Dupont line \*1bunch

Actual object connection diagram:

We need to connect the circuit as shown in the figure below.

Note: The passive buzzer has positive and negative electrode. The actual object diagram below shows that the buzzer has positive and negative marks.



	 	Passt	ive buzzer
	 +		
	 	********	
	 	*********	
legative electrode	T		
	 Press Press	ositive elec	trode
220Q Register	T	he object is	tagged with *
220** Resistor			
100 B			
Contract of the second			
dupont line		1	e
dupont line			0
dupont line			Breadboard
dupont line		[	Breadboard duino UNO boar
dupont line			Breadboard duino UNO boar
dupont line			Breadboard duino UNO boar
dupont line			Breadboard duino UNO boar
dupont line			Breadboard duino UNO boar
dupont line			Breadboard duino UNO hoar

Experimental code analysis:

#define BL1 248 #define BL2 278



#define BL3 294 #define BL4 330 #define BL5 371 #define BL6 416 #define BL7 467 #define B1 495 #define B2 556 #define B3 624 #define B4 661 #define B5 742 #define B6 833 #define B7 935 #define BH1 990 #define BH2 1112 #define BH3 1178 #define BH4 1322 #define BH5 1484 #define BH6 1665 #define BH7 1869 #define NTC1 262 #define NTC2 294 #define NTC3 330 #define NTC4 350 #define NTC5 393 #define NTC6 441 #define NTC7 495 #define NTCL1 131 #define NTCL2 147 #define NTCL3 165 #define NTCL4 175 #define NTCL5 196 #define NTCL6 221 #define NTCL7 248 #define NTCH1 525 #define NTCH2 589 #define NTCH3 661 #define NTCH4 700 #define NTCH5 786 #define NTCH6 882 #define NTCH7 990 #define NTD0 -1 #define NTD1 294 #define NTD2 330 #define NTD3 350 #define NTD4 393 #define NTD5 441


#define NTD7 556 #define NTDL1 147 #define NTDL2 165 #define NTDL3 175 #define NTDL4 196 #define NTDL5 221 #define NTDL6 248 #define NTDL7 278 #define NTDH1 589 #define NTDH2 661 #define NTDH3 700 #define NTDH4 786 #define NTDH5 882 #define NTDH6 990 #define NTDH7 1112 #define NTE1 330 #define NTE2 350 #define NTE3 393 #define NTE4 441 #define NTE5 495 #define NTE6 556 #define NTE7 624 #define NTEL1 165 #define NTEL2 175 #define NTEL3 196 #define NTEL4 221 #define NTEL5 248 #define NTEL6 278 #define NTEL7 312 #define NTEH1 661 #define NTEH2 700 #define NTEH3 786 #define NTEH4 882 #define NTEH5 990 #define NTEH6 1112 #define NTEH7 1248 int speakerPin= 8;

#define NTD6 495

int buzzer=8;//Defining the digital port 8 to control the buzzer int i = 0; /\*YeZi C\*/ int tune[]= //List the frequencies according to the simple spectrum { NTC3, NTC5, NTC5, NTC3, NTC6, NTC6, NTC7, NTC6, NTC6, NTC6, NTC5, NTC1, NTCH1, NTCH1, NTCH1, NTC6,NTCH1, NTC6, NTC5,NTC3, NTC5, NTC5, NTC5, NTC3, NTC6, NTC6, NTC7, NTC6, NTC6, NTC6, NTC5, NTC1, NTC41,

```
NTCH1,NTCH1, NTC6, NTC6, NTCH1,NTCH2,NTCH5, NTCH5, NTCH5,
NTCH5, NTCH5, NTCH3, NTCH2, NTCH1, NTCH1, NTC6, NTCH1, NTC6,
NTCH1,NTCH2,NTCH2, NTCH2, NTCH2,NTCH2, NTCH1, NTCH3, NTCH2,
NTCH2, NTCH3, NTCH3, NTCH3, NTCH3, NTCH2, NTCH2, NTCH1, NTCH1, NTCH1, NTCH1,
NTCH2, NTCH1, NTC6, NTC5, NTC5, NTC5, NTC5, NTC6, NTC5, NTCH2,
NTCH3,NTCH1,
};
float durt[]=
                       //List the tempo according to the simple spectrum
0.5, 0.5, 1.5, 0.5, 0.5, 0.5, 0.5, 1, 0.5, 0.5, 0.5, 1, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 2, 0.5, 0.5,
1, 0.5, 0.5, 0.5, 0.5, 0.5, 1, 0.5, 0.5, 0.5, 1, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 2,0.5, 1, 0.5, 0.5,
0.5, 0.5, 0.5, 0.5, 1, 1, 0.5, 0.5, 0.5, 1, 0.25, 0.5, 0.5, 0.5, 0.5, 1, 0.25, 2,0.5, 1, 0.5, 0.5,
0.5, 1, 0.5, 1, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 1,0.5, 0.5, 0.5, 0.5, 0.5, 2,
};
void PlayTest()
ł
int length = sizeof(tune)/sizeof(tune[0]); //Calculate length
for(int x=0; x < \text{length};x++)
 {
  tone(speakerPin,tune[x]);
  delay(500*durt[x]); //This is used to adjust the time delay according to the beat,500
this index can be adjusted by yourself. In this music, I find that 500 is more suitable.
  noTone(speakerPin);
 }
}
void setup()
pinMode(buzzer,OUTPUT); //Defining the buzzer port for the output port
}
void loop()
{
     PlayTest();
Experimental steps:
```

1.We need to open the code of this experiment: code-Passive\_buzzer.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.





2.In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

2件(F) 操作(A)	查看(V) 報助(H)			
Xiaozhen				
D Ca IDE ATA	/ATAPI 控制器			
👂 🔮 Jungo Cr	onnectivity			
▶ 👝 磁盘驱动	8			
• 7字 第日 (CO	M 和 LPT)			
- P Blueb	ooth Serial Port (COM	15)		
- Blueb	ooth Serial Port (COM	16)		
- Blueb	ooth Serial Port (COM	17)		
- Blueb	ooth Serial Port (COM	18)		
Blueb	ooth Serial Port (COM)	19)		
- Blueb	ooth Serial Port (COM)	20)		
- Blueb	ooth Serial Port (COM	21)		
- I Blueb	ooth Serial Port (COM	22)		
TIT LICE.	COMPANY CHIER COMPANY	23)		
1035				
100 100 00	the converse			
▶ (● 计算机 Code Passive bu	zam   Arduino 1.7.8			
● ☆ 计算机 code Passive bu e Edit Sketch T	zzar   Arduino 1.7.8 ools   Help Auto Format	Curl+T		
ode Passive bu e Edit Sketch T	zzm   Arduino 1.7.8 ools   Help Auto Format Archive Sketch	Corl+T		
ode Passive bu e Edit Sketch T Code Passive_t	zenr   Arduino 1.7.8 ools   Help Auto Format Archive Sketch Fix Encoding & Rel	Ctrl+T oad		
e Edit Sketch T code Passive bu e Edit Sketch T code-Passive_t define BL1	zzer Arduno 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor	Ctrl+T oad Ctrl+Shift+M		
code Persive Bu e Edit Sketch T COLLET code Passive_1 define BL1 define BL2	zzer Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board	Ctrl+T oad Ctrl+Shift+M		
code Pensive Bu e Edit Sketch T CODE-Passive_L define BL1 define BL2 define BL3	ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port	Ctrl+T oad Ctrl+Shift+M	Serial ports	
code Passive bu e Edit Sketch T Code-Passive_L define BL1 define BL2 define BL3 define BL4	zzer   Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1	
code Passive bu e Edit Sketch T Code-Passive_1 define BL1 define BL2 define BL2 define BL3 define BL4	zzer   Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1	
code Passive bu e Edit Sketch T Code-Passive_1 define BL1 define BL2 define BL3 define BL4 define BL5	zzer Arduno 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1 V COM6 COM15	
e Edit Sketch T code Passive bu e Edit Sketch T Code-Passive_t define BL1 define BL2 define BL3 define BL3 define BL5 define BL5	zzer Arduno 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader 416	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1 V COM6 COM15 COM16	
code Passive bu e Edit Sketch (7 Code-Passive_t define BL1 define BL2 define BL3 define BL3 define BL5 define BL5 define BL5	zzer Arduno 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader 416	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1 V COM6 COM15 COM15 COM16 COM17	
code Passive bu bit Sketch (7 Code-Passive_1 define BL1 define BL2 define BL3 define BL3 define BL3 define BL3 define BL3 define BL3 define BL3	zzer Arduno 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader 416 467	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1 V COM6 COM15 COM16 COM17 COM18	2
code Passive bu e Edit Sketch (P Code-Passive_1 define BL1 define BL2 define BL3 define BL3 define BL5 define BL5 define BL5 define BL5	zzer Arduno 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader 416 467	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1 COM15 COM15 COM15 COM15 COM18 COM19	2
code Passive bu e Edit Sketch (F Code-Passive I define BL1 define BL2 define BL3 define BL3 define BL3 define BL4 define BL5 define BL5 define BL5 define BL5	zzmr   Arduino 1.7.8 cols Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader 416 467 95	Oad Ctrl+Shift+M	Serial ports COM1 V COM6 COM15 COM15 COM15 COM17 COM18 COM19 COM20	
code Pessive bu e Edit Sketch T Code Passive 1 define BL1 define BL2 define BL3 define BL5 define BL5 define BL6 define BL6 define BL6 define BL7 define B1 4 define B2 5	azarr Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader 416 467 95	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1 V COM6 COM15 COM16 COM17 COM18 COM19 COM20 COM21	
code Passive Du e Edit Sketch T Code Passive I define BL1 define BL2 define BL3 define BL3 define BL4 define BL5 define BL5 define BL5 define BL5	azarr Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Rel Serial Monitor Board Port Programmer Burn Bootloader 416 467 95	Ctrl+T oad Ctrl+Shift+M	Serial ports COM1 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21 COM21 COM22	

3.After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded, We can hear the buzzer will sing according to the score written in the program, as shown in the following figure.





# 9-PWM dimming

#### The purpose of the experiment:

Arduino controller has six PWM (Pulse Width Modulation) interface, which are digital interface 3, 5, 6, 9, 10, 11. In this course, we input the different analog voltage by adjusting the adjustable resistor, and the microcontroller recognizes the corresponding proportional PWM wave to control the brightness of the LED lights.

#### Introduction of PWM:

PWM is used in many places, dimmable lighting, motor speed regulation, sound production and so on.

Its three basic parameters:

- 1. Pulse width
- 2. The pulse period (the reciprocal of the number of pulses in one second).
- 3. Voltage height (for example: 0V-5V)

#### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω Resistor \*1

Adjustable resistor \*1

Breadboard \*1

Dupont line \*1bunch

#### Actual object connection diagram:

We need to connect the circuit as shown in the figure below.





## Experimental code analysis:

```
int potpin=A0; //Defining the analog port A0
int ledpin=13; //Defining the led port 13
int val=0; //Declarations of temporary variables
void setup()
{
    pinMode(ledpin,OUTPUT); //Defining the light port for the output port
    Serial.begin(9600); //The baud rate is 9600
}
void loop()
{
    val=analogRead(potpin); //Read the voltage value of the A0 port and assign it to val
    Serial.println(val); //Sending Val value by serial port
}
```

## Experimental steps:

1.We need to open the program of this experiment: code-PWM-dimming.ino, click" $\sqrt{2}$ " under the menu bar to compile the program, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2.In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

YVHSOON

3.After the selection is completed, you need to click " $\rightarrow$  "under the menu bar to upload the program to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the program has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4.After the program is uploaded. When we adjust the adjustable resistance, we will see the change in the brightness of the LED. If you turn on the serial port monitor at the same time, you can also see the change of the printed values.

<u>A BOO</u>		10	CORDE		
COMB	Contraction in the local data		COM6	1010	-
	Seat		v 16	Servel	1
	-		4010		-
1023			1016		
1023			1015		
1023		· (*100 States)	1016		100
1023		a cuito carsta	1017		r 11
1023			1017		
1023			1016		
1023			1015		
1023			1016		
1022		THE AV DOLT	1015		the
1023		pert	1013		part
1023			1017		Contract of the
1023	-	-	1014		
	-		1016		•
2  Annasses stat	Fo line sading + 9600 hand	1 topological	1015		-
	Sector And	1 Teaving 1	Antenerol1	Be line enting - 9800 hand	1.0
ai variables, maxi	min 15 2. Van Dytes.			2 018 0 11	1.10
			local variables. Maxi	man is 2,0+0 bytes.	
and the second of the second					
fullt United Tracts Male	m s.c.a		Cride-Analog_value	Reduino 1.7.8	
ABBB			HAV OUR DAVEN TUBLE	NOTE:	
of the lot			CALCE IN LACE		_





## **10-Light controlled sound**

## The purpose of the experiment:

The purpose of this experiment is to make you learn how to use a special resistor -photosensitive resistor. This experimental result is: when the photosensitive resistance is connected in the circuit, the buzzer will make a small sound in the absence of light. In the case of illumination, the resistance of the photosensitive resistor will decrease, which causing the voltage across the buzzer to increase and the buzzer sound to become louder.

## List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

Photosensitive resistor \*1

Active buzzer \*1

Breadboard \*1

Dupont line \*1bunch

### Actual object connection diagram:

We need to connect the circuit as shown in the figure below.



## Experimental code analysis:

```
int buzzer=6;//Defining the digital IO port6 to control the buzzer
int i = 0;
void setup()
{
    pinMode(buzzer,OUTPUT);//Defining the led port for the output port
}
void loop()
{
```



```
for(i=0;i<80;i++)//Output a frequency of sound
{
    digitalWrite(buzzer,HIGH);//sound
    delay(1);
    digitalWrite(buzzer,LOW);//unsound
    delay(1);
}
for(i=0;i<100;i++)//Output another frequency of sound
{
    digitalWrite(buzzer,HIGH);//sound
    delay(2);
    digitalWrite(buzzer,LOW);//unsound
    delay(2);
}</pre>
```

## Experimental steps:

1.We need to open the code of this experiment:code-Light\_controlled\_sound.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2.In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

YAHBOOM

	10   1980		
A Xiaozhen	T & D7 +444/88		
DE ATA/	ATAPI 控制器		
ALTHER	nnecuvity		
「「「「「「「「」」」は、「「」」」は、「「」」」は、「「」」」」は、「「」」」」は、「」」」」は、「」」」」は、「」」」」は、「」」」」は、「」」」」」、「」」」」、「」」」、「」」」、「」」、「			
• 🐨 調日 (COM	M FO LPT)		
TT Blueto	oth Serial Port (COM15)		
T Blueto	oth Serial Port (COM16)		
- 1 Blueto	oth Serial Port (COM17)		
- P Blueto	oth Serial Port (COM18)		
- Blueto	oth Serial Port (COM19)		
Blueto	oth Serial Port (COM20)		
TH Blueto	oth Serial Port (COM21)		
T Blueto	oth Serial Port (COM23)		
USB-S	ERIAL CH340 (COM6)		
一帶 通信調	□ (COM1)		
> 1 计算机			
code light cont e Edit Sketch <mark>1</mark>	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T		
Code Light cont Re Edit Sketch T CODEC code-Light_cont int buzzer=	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M	ertô to cont	P P
Code Ught cont de Edit Sketch T Code-Light_cont int buzzer= int i = 0;	oollect sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fox Encoding & Reload Serial Monitor Ctrl+Shift+M Board	ertô to cont	rol tl
Code Light cont ile Edit Sketch T Code-Light_cont int buzzer= int i = 0; roid setup	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board • Port	Dertô to cont Serial ports	rol tl
© code light cont file Edit Sketch T Code-Light_cont int buzzer= int i = 0; void setup {	rollect sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port	ortô to cont Serial ports COMI	rol tl
Code Light cont ile Edit Sketch T code Light_con int buzzer= int i = 0; void setup {	oollect sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer	Serial ports COM1	rol tl
Code Ught cont ile Edit Skatch T code-Light_cont int buzzer= int i = 0; void setup { pinMode(buz	oolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1	rol tl
Code Light cont ile Edit Sketch T code-Light_cont int buzzer= int i = 0; void setup { pinMode (buz_)	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15 COM16	rol tl
© code Light cont File Edit Sketch T Code-Light_con int buzzer= int i = 0; void setup { pinMode(buz_) void Toop()	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15 COM16 COM17	rol tl
© code light cont The Edit Sketch T Code-Light_cont int buzzer= int i = 0; void setup { pinMode(buz_) void loop() {	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15 COM15 COM15 COM17 COM18	rol tl
Code Ught cont ile Edit Sketch T code-Light_cont int buzzer= int i = 0; roid setup { pinMode(buz_ roid Toop() { fun(i=0)	oolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15 COM15 COM16 COM17 COM18 COM19	rol tl he ou
Code Light cont le Edit Sketch T code-Light_cont int buzzer= int i = 0; roid setup [ pinMode (buz) roid loop() [ for (i=0;	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20	rol tl
Code Light cont ile Edit Skatch T code-Light_con int buzzer= int i = 0, void setup { binMode(buz } void loop() { for(i=0,1)	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21	rol tl
Code Light cont file Edit Sketch T code-Light_cont int buzzer= int i = 0, void setup { pinMode(buz_) void loop() { for(i=0, { digita	rolled sound   Arduino 1.7.8 ools   Help Auto Format Ctrl+T Archive Sketch Fox Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21 COM21	rol tl
Code Light cont ine Edit Sketch code-Light_cont int buzzer= int i = 0, void setup { pinMode(buz_ } void Toop() { for(i=0, { digita	i(80, i++)//Output a frequence Write(buzzer, HIGE) //sound	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM20 COM21 COM21 COM23	rol tl he ou

3.After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4.We need to give the photosensitive resistor different intensity of illumination, and as the light intensity changes, the size of the sound of the buzzer will also change. As shown in the figure below.





# **11-Sensible heat light**

### The purpose of the experiment:

In this course, we use sensor -- thermistor to control PWM by controlling the change of thermistor resistance value, so as to control the brightness of LED light.

Introduction of thermistor:

Thermistor is a kind of sensitive element, which can be divided into positive temperature coefficient thermistor (PTC) and negative temperature coefficient thermistor (NTC) according to different temperature coefficient. The typical characteristic of thermistors is that they are sensitive to temperature and show different resistance values at different temperatures. Resistance value of positive temperature coefficient (PTC) thermistor will become higher when the higher the temperature, Resistance value of negative temperature coefficient (NTC) thermistor will become lower when the higher the temperature. They are both semiconductor devices.

#### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

Negative temperature coefficient Thermistor\*1

220Ω resistor \*1

10kΩ resistor \*1

LED \*1

Breadboard \*1

Dupont line \*1 bunch

#### Actual object connection diagram :

We need to connect the circuit as shown in the figure below.





## Experimental code analysis:

```
int potpin=0; //Defining the analog port A0
int ledpin=11; //Defining the digital port 11(for output PWM)
int val=0; //Defining Variable val
void setup()
{
pinMode(ledpin,OUTPUT); //Defining the port11 for the output port
Serial.begin(9600); //The baud rate is 9600
}
void loop()
{
val=analogRead(potpin); //Reading the voltage value of the A0 port and assign it to val
val = 245 - val;
     if(val < 0)
      val = 0;
Serial.println(val);
analogWrite(ledpin,val); //PWM output is used to drive LED
delay(100);
}
```

## Experimental steps:

1.We need to open the code of this experiment: code-Sensible\_heat\_light.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】--- 【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

YAHBOOM

乙件(F) 摄作(A)	查看(V) 報助(H)		
-			
🔬 Xiaozhen			
D Ca IDE ATA/	ATAPI 控制器		
🕴 🔮 Jungo Co	onnectivity		
▶ □ 处理器			
一 磁盘驱动器	<b>5</b>		
• 響 第日 (CO)	M ROLPT)		
- Blueto	ooth Senial Port (COM15)		
TT Blueto	both Serial Port (COM17)		
Till Bluet	both Serial Port (COM18)		
Blueto	ooth Serial Port (COM19)		
- Blueto	ooth Serial Port (COM20)		
- Blueta	ooth Serial Port (COM21)		
- Blueta	ooth Serial Port (COM22)		
- T Blueto	ooth Serial Port (COM23)		
一帶 USB-S	SERIAL CH340 (COM6)		
一带 通信部	€□ (COM1)		
D 🐏 计算机			
▷ 0 · 1 · 并并有1			
code Sensible h	eat_light   Arcluino 1.7.8		
code Sensible In Edit Sketch	eut light   Ardoino 1.7.8 cols Help		- 0 ×
code-Sensible h	nat_light   Arduino 1.7.8 ools) Help Auto Format	Ctrl+T	
code-Sensible h Edit Sketch	eat light   Arduino 1.7.8 ools Help Auto Format Archive Sketch	Ctrl+T	
edit Sketch	eat light   Arduine 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reloa	Cn/+T d	2 •
edit Sketch	eat light   Arduine 1.7.8 ools   Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor	Crif+T d Crif+Shilt+M	2 0 ×
e de Sensible h Edit Sketch (f Code-Sensible code-Sensible did 100p ()	eut, light   Ardoino: 1.7.8 ools   Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Reard	Ctrl+T d Ctrl+Shift+M	
e dia Sketch ( code-Sensible ) code-Sensible oid loop ()	eat light   Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port	Ctrl+T d Ctrl+Shift+M	
e de Sensible h Edit Sketch ( Code-Sensible ode-Sensible did Loop ()	eat light   Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port	Ctrl+T d Ctrl+Shilt+M	Serial ports
e math code-Sensible h Edit Sketch code-Sensible oid loop () val	eut light   Arduines 1.7.8 cols Help Auto Format. Archive Sketch Fix Encoding & Relou Serial Monitor Board Port Programmer	Ctrl+T d Ctrl+Shift+M	Serial ports COMI
p m π#h code-Sensible h Edit Sketch code-Sensible oid loop() val val	eut light   Arduines 1.7.8 ools   Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port Programmer Burn Bootloader	Crrl+T d Crrl+Shilt+M	Serial ports COMI
P # π#h code-Sensible h Edit Sketch [] code-Sensible_ oid loop() val val if (wal	eut light   Arduine: 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port Programmer Burn Bootboader a1 < 0)	Ctrl+T d Ctrl+Shift+M	Serial ports COM1 COM6 COM15
P m #h code-Sensible h Edit Sketch [] code-Sensible_ oid loop () val val if (val	eut light   Arduine: 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port Programmer Burn Bootoader a1 < 0) 1 = 0	Ctrl+T d Ctrl+Shilt+M	Serial ports COM1 COM15 COM15 COM15
e dit Sketch ( code-Sensible h e Edit Sketch ( code-Sensible oid loop () val val if (va va	eut light   Arduine: 1.7.8 cols Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port Programmer Burn Bootoader a1 < 0) 1 = 0.	Ctrl+T d Ctrl+Shift+M	Serial ports COM1 COM15 COM15 COM15
P # ##N code Sensible code Sensible odd loop () val val if (val val Seri	eut light   Ardoino 1.7.8 ools   Help Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port Programmer Burn Bootoader al < 0) l = 0. ol. print le (val)	Ctrl+T d Ctrl+Shift+M	Serial ports COM1 COM15 COM15 COM15 COM17 COM18
Fit HAN code Sensible h r Edit Sketch (i code-Sensible) oid loop() val val if (va seri anal	eat light   Arduino 1.7.8 cols Help Auto Format. Archive Sketch Fix Encoding & Relou Serial Monitor Board Pot Programmer Burn Bootboader a1 < 0) 1 = 0. a1. printle (val) colorin. val	Ctrl+T d Ctrl+Shilt+M	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19
Fit HAN code Sensible h Edit Sketch (i) code-Sensible oid loop () val val if (val val if (val serianal delan	eat light   Arduines 1.7.8 cols   Help Auto Format. Archive Sketch Fix Encoding & Relou Serial Monitor Board Port Programmer Burn Bootloader a1 < 0) 1 = 0. ol. print in (val) selfinite (ledpin, val) (100) :	Ctrl+T d Ctrl+Shift+M	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM18 COM19 COM20
e da Sketch ( code-Sensible code-Sensible oid loop () val if (va seri analo delar	eat light   Ardumes 1.7.8 cols   telp Auto Format. Archive Sketch Fix Encoding & Relou Serial Monitor Board Port Programmer Burn Bootloader al < 0) l = 0. al. print ln (val) perint (ledpin, val) (100)	Ch1+T d Ch1+Shilt+M	Serial ports COM1 COM15 COM15 COM15 COM15 COM17 COM18 COM19 COM20 COM20 COM21
e eda Sketch ( e Eda Sketch ( code-Sensible oid loop () val if (va val if (va seri anald delat	eat light   Ardumes 1.7.8 cols   telp Auto Format Archive Sketch Fix Encoding & Reloa Serial Monitor Board Port Programmer Burn Bootloader al < 0) 1 = 0. al. print ln (val) perint (ledpin, val) perint (ledpin, val)	Crif+T d Crif+Shilt+M	Serial ports COMI COM6 COM15 COM15 COM15 COM16 COM17 COM18 COM18 COM19 COM20 COM20 COM21 COM21 COM21

3.After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded. When we do not heat the thermistor, the LED extinguish. When we heat the thermistor, the LED will bright, and the brightness of the LED will change with the change of the heat of the thermistor. At the same time, we can open the serial port monitor, and we can also see the change of the voltage value at both ends of the LED, as shown in the following figure.





## 12-8x8 dot matrix

### The purpose of the experiment:

In this course, we will learn how to use 8x8 dot matrix. The experimental effect is to light the LED on the 8x8 dot matrix.

### Introduction of 8x8 dot matrix:

The 8x8 lattice is composed of 64 LED, and each LED is placed at the intersection of line and line. When one line is high level(1) and a column is low level(0), the corresponding diode will be bright. If you want to light up the first line, the ninth pin need to high level, and (13, 3, 4, 10, 6, 11, 15, 16) these pins are low level. If you want to light up the first column, the thirteenth pin need low level, and (9, 14, 8, 12, 1, 7, 2, 5) these pins are low level.

Pin identification as shown in the two figures below.



## List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω resistor \*8

8x8 dot matrixLED\*1

Breadboard \*1

dupont line \*1bunch

## Actual object connection diagram :

We need to connect the circuit as shown in the figure below.



## Experimental code analysis:

```
const int row1 = 2; // Arduino Pin2 connect pin9 of the dot matrix
const int row2 = 3; // Arduino Pin3 connect pin14 of the dot matrix
const int row3 = 4; // Arduino Pin4 connect pin8 of the dot matrix
const int row4 = 5; // Arduino Pin5 connect pin12 of the dot matrix
const int row5 = 17; // Arduino Pin17 (A3)connect pin1 of the dot matrix
const int row6 = 16; // Arduino Pin16 (A2)connect pin7 of the dot matrix
const int row7 = 15; // Arduino Pin15 (A1)connect pin2 of the dot matrix
const int row8 = 14; // Arduino Pin14 (A0)connect pin5 of the dot matrix
//the pin to control COI
const int col1 = 6; //Arduino Pin6 connect pin13 of the dot matrix
const int col2 = 7; // Arduino Pin7 connect pin3 of the dot matrix
const int col3 = 8; //Arduino Pin8 connect pin4 of the dot matrix
const int col4 = 9; // Arduino Pin9 connect pin10 of the dot matrix
const int col5 = 10; //Arduino Pin10 connect pin6 of the dot matrix
const int col6 = 11; //Arduino Pin11 connect pin11 of the dot matrix
const int col7 = 12; // Arduino Pin12 connect pin12 of the dot matrix
const int col8 = 13; // Arduino Pin13 connect pin13 of the dot matrix
void setup()
int i = 0;
for(i=2;i<18;i++)
ł
pinMode(i, OUTPUT);
for(i=2;i<18;i++) {
digitalWrite(i, LOW);
```



```
}
}
void loop()
{
int i;
//the row # 1 and col # 1 of the LEDs turn on
digitalWrite(row1, HIGH);
digitalWrite(row2, LOW);
digitalWrite(row3, LOW);
digitalWrite(row4, LOW);
digitalWrite(row5, LOW);
digitalWrite(row6, LOW);
digitalWrite(row7, LOW);
digitalWrite(row8, LOW);
digitalWrite(col1, LOW);
digitalWrite(col2, HIGH);
digitalWrite(col3, HIGH);
digitalWrite(col4, HIGH);
digitalWrite(col5, HIGH);
digitalWrite(col6, HIGH);
digitalWrite(col7, HIGH);
digitalWrite(col8, HIGH);
delay(1000);
//turn off all
for(i=2;i<18;i++) {
digitalWrite(i, LOW);
}
delay(1000);
}
```

## **Experimental steps:**

1.We need to open the code of this experiment: code-8x8\_dot\_matrix.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



YAHBOOM

2. In the menu bar of Arduino IDE, we need to select 【Tools】--- 【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

diese miles -	图看(V) 報助(H)		
	1 46		
🖌 📩 Xiaozhen			
D - CATA/A	TAPI 控制器		
👂 🔮 Jungo Con	nectivity		
▷ □ 处理器			
2 📷 磁盘驱动器			
• л 第日 (COM	和 LPT)		
- Bluetoo	th Serial Port (COM15)		
- Bluetoo	th Serial Port (COM16)		
- Bluetoo	th Serial Port (COM17)		
- Bluetoo	th Serial Port (COM18)		
- Bluetoo	th Serial Port (COM19)		
Bluetoo	th Serial Port (COM20)		
- Bluetoc	th Serial Port (COM21)		
T Bluetoo	th Serial Port (COM22)		
TET LICE.CE	RIAL CH240 (COM5)		
101 通信地口			
THE THE PARTY	T (COMI)		
Þ 👭 计算机			
▶ 📲 计算机			
▶ :● 计算机 D code-8x8 dot mat	ris   Arduino 1.7.8		x
▶ + ➡ 计算机 D code-8x8, dot mat le Edit Sketch [Top	rix   Arduino 1.7.8 Sis   Help		×
▶ (● 计算机 Coode-8x8 dot met le Edit Sketch Too	rix   Arduino 1.7.8 se Help Auto Format Ctrl+T		×
▶ 小 计并机 Coode-8x8 dot mat lie Edit Sketch Too	rix   Arcluino 1.7.8 Help Auto Format Ctrl+T Archive Sketch		×
▶ 小 计算机 Coode 8x8 dot mat le Edit Sketch Too Code 8x8_dot_r	rix   Archuino 1.7.8 Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload		×
P : IT	rix   Arduino 1.7.8 sis Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M		×
▶ :● 计算机 Coode-Baß, dot, mat lie Edit Sketch Too Coode-Bx8_dot_r digi digi	rix   Arduino 1.7.8 S Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M		×
P # it#in P code-8x8 dot mat Re Edit Sketch Too code-8x8 dot r digi digi digi	rix   Arduino 1.7.8 Sis Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board		
P f tr≢th P code-8x8 dot mat Receive the formation of th	ris   Arduino 1.7.8 Ss Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port	Serial ports	×
P # tr#t. P code-8x8 dot met Receive the transmission of tra	ris   Arduino 1.7.8 S Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer	Serial ports COM1	
# it#in code-8x8 dot met ie Edit Sketch Tor code-8x8_dot_r digi digi digi dela //ti for (	rix   Arduino 1.7.8 S Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1	×
P # it#in P code-8x8 dot met Re Edit Sketch Tor Code-8x8_dot_r digi digi digi digi dela //til for (	rix   Arduino 1.7.8 Second Second Ctrl+T Archive Sketch Fix Encoding & Reload Secial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM15	
<ul> <li>H # ##</li> <li>Coode-8x8. dot met lie Edit Sketch Toe</li> <li>Code-8x8. dot r digi digi dela //ti for</li> </ul>	ris   Arduino 1.7.8 S Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite(i, LOW).	Serial ports COM1 COM15 COM16	×
if if #1. Coode-8x8. dot met ier Edit Sketch Toe Code-8x8.dot_r Code-8x8.dot_r digi <pddigi< p=""> digi di di <pddigi< p=""> di &lt;</pddigi<></pddigi<>	ris   Arcluino 1.7.8 Serial Monitor Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite(1, LOW).	Serial ports COM1 COM15 COM16 COM17	
# it#ti code-8x8.dot.met ie Edit Sketch Toe code-8x8.dot.r digi digi dela //ti for } delar(	rix   Arcluino 1.7.8 Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite(1, LOW). 1000)	Serial ports COM1 COM6 COM15 COM16 COM17 COM18	
if if #1. Coode-8x8. dot. met Re Edit Sketch Tor Code-8x8_dot_r digi digi digi delay (	rix   Arcluino 1.7.8 Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite(1, LOW). 1000).	Serial ports COM1 COM15 COM15 COM16 COM17 COM18 COM19	
if if #1. Coode-8x8. dot. met Re Edit Sketch Tor Code-8x8.dot.r Code-8x8.dot.r digi digi digi delay (	rix   Arcluino 1.7.8 Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite(i, LOW). 1000).	Serial ports COM1 COM15 COM15 COM16 COM16 COM17 COM18 COM19 COM20	
<pre>&gt; f it # it code-8x8.dot met ie Edit Sketch Toe code-8x8.dot_r digi digi dela //ti for } delay(</pre>	rix   Arcluino 1.7.8 Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite(i, LOW). 1000).	Serial ports COM1 COM15 COM15 COM16 COM16 COM17 COM18 COM19 COM20 COM21	A latter
<pre>&gt; ** if#fil code-8x8_dot_met ile Edit Sketch for code-8x8_dot_r digi digi dela //ta for ( } delay (</pre>	rix   Archaino 1.7.8 Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite(i, LOW). 1000).	Serial ports COM1 COM15 COM15 COM16 COM16 COM17 COM18 COM19 COM19 COM20 COM21 COM21	

3. After the selection is completed, you need to click " $\rightarrow$ "under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, thecode has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded.We can see that the lights in the first row and first column of the dot matrix are twinkling, as shown in the following figure.





## **13-Tilt switch**

#### The purpose of the experiment:

This lesson is ball switch experiment, it also belongs to the tilt switch just name is different. It control the turning on or off of the circuit by the rolling contact pin of the beads in the switch, so the LED can be switched on and off.

#### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω resistor \*1

10kΩ resistor \*1

Tilt switch \*1

Breadboard \*1

Dupont line \*1 bunch

#### Actual object connection diagram :

We need to connect the circuit as shown in the figure below.





Experimental code analysis:



```
int switchpin = 5;
int ledpin = 8;
int val = 0;
void setup()
{
pinMode(ledpin,OUTPUT);//Defining the led port for the output port
Serial.begin(9600);//The baud rate is 9600
}
void loop()
{
     val = analogRead(switchpin);
     if(val>512)//The analog voltage value of 512 is exactly 2.5V
     digitalWrite(ledpin,HIGH);//If val Greater than 2.5 V
     else//If val less than or equal to 2.5 V
     digitalWrite(ledpin,LOW);
     Serial.println(val);
```

## Experimental steps:

1.We need to open the code of this experiment: code-8x8\_dot\_matrix.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】--- 【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

YVHSOON

文件(F) 操作(A) 计			
	查著(V) 帮助(H)		
A 🚔 Xiaozhen			
D CON IDE ATA/A	ATAPI 控制器		
D Ingo Cor	nnectivity		
2 1 文上注意語			
	A #O LPT)		
- Blueto	oth Serial Port (COM15)		
- P Blueto	oth Serial Port (COM16)		
- P Blueto	oth Serial Port (COM17)		
- P Blueto	oth Serial Port (COM18)		
- Blueto	oth Serial Port (COM19)		
Blueto	oth Serial Port (COM20)		
TP Blueto	oth Serial Port (COM22)		
Blueton	oth Serial Port (COM23)		
USB-SE	ERIAL CH340 (COM6)		
一型 通信調	(COM1)		
▷-1■ 计算机			
code_Tilt_switch	Arduino 1.7.8		
		Concession Property of the second	•
e Edit Sketch To	sols Help		
Contended In Conte	Auto Format Ctrl+T		
e Edit Sketch Te	Auto Format Ctrl+T Archive Sketch	Ø	
e Edit Sketch To Code_Titt_switc	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload	2	
code_Tilt_switc t switchpi	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shilt+M	<u></u>	
code_Tit_switc t switchpi t ledpin =	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board		
code_Titt_switc tt switchpi tt ledpin = tt val = 0;	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board	Serial ports	
code_Titt_switc at switchpi at ledpin = at val = 0. bid setup(	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port	Serial ports COM1	11
code_Tilt_switc at switchpi at ledpin = at val = 0, bid setup(	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer	Serial ports COM1 COM6	
code_Tit_switc nt switchpi nt ledpin = nt val = 0, oid setup(	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	Serial ports COM1 COM6 COM15	11
code_Titt_switc t switchpi at ledpin = at val = 0, oid setup( nMode(ledpin	Solis     Help       Auto Format     Ctrl+T       Archive Sketch     Fix Encoding & Reload       Fix Encoding & Reload     Serial Monitor       Serial Monitor     Ctrl+Shift+M       Board     Port       Port     Port       Programmer     Programmer       Burn Bootloader     Port	Serial ports COM1 COM6 COM15 COM16	
code_Titt_switc t switchpi at ledpin = at val = 0, oid setup( nMode(ledpin code_titt_switch)	Auto Format     Ctrl+T       Archive Sketch     Fix Encoding & Reload       Fix Encoding & Reload     Ctrl+Shift+M       Board     Port       Programmer     Programmer       Burn Bootloader     Post       AUTPUT)     //Defining the lei       9600)     //The baud rate is 96	Serial ports COM1 COM6 COM15 COM16 COM16	11
code_Tit_switc t switchpi at ledpin = at val = 0, bid setup( nMode(ledpin arial.begin(	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader A, OUTPUT) .//Defining the lea (9600) .//The baud rate is 96	Serial ports COM1 COM6 COM15 COM16 COM17 COM18	
code_Titt_switc t switchpi at val = 0, oid setup( nMode(ledpin trial.begin()	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader A, OUTPUT) : //Defining the let (9600) : //The baud rate is 90	Serial ports COM1 COM6 COM15 COM15 COM17 COM18 COM19	
code_Titt_switc t switchpi at ledpin = at val = 0, oid setup( nMode(ledpin arial.begin() arial.begin()	Solid       Help         Auto Format       Ctrl+T         Archive Sketch       Fix Encoding & Reload         Fix Encoding & Reload       Serial Monitor         Serial Monitor       Ctrl+Shift+M         Board       Port         Programmer       Programmer         Burn Bootloader       Pott         AUTPUT)       //Defining the leight         9600)       //The boud rate is 96	Serial ports COM1 COM6 COM15 COM15 COM15 COM17 COM18 COM19 COM20	
code_Tit_switc t switchpi at ledpin = at val = 0, bid setup( mMode(ledpin arial.begin() bid loop()	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader A, OUTPUT) //Defining the le- (9600) //The baud rate is 96	Serial ports COM1 COM6 COM15 COM15 COM15 COM17 COM18 COM19 COM20 COM21	
code_Tit_switc t switchpi at ledpin = at val = 0, oid setup( nMode(ledpin mial.begin() id loop()	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader A, OUTPUT) . //Defining the let (9600) . //The baud rate is 90	Serial ports COM1 COM6 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM20 COM21 COM21 COM22	
<pre>code_Titt_switc rode_Titt_switc nt switchpi nt ledpin = nt val = 0, oid setup( nMode(ledpin crial.begin() oid loop()</pre>	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader A, OUTPUT) //Defining the let (9600) //The baud rate is 90	Serial ports COM1 COM6 COM15 COM15 COM17 COM18 COM19 COM20 COM21 COM21 COM22 COM23	

3. After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4. After the code is uploaded. The LED lights up when the ball switch is in the horizontal position, and the LED turns off when we put the ball switch in the tilt position., At the same time, we can open the serial port monitor, we can also see the change of the voltage value at both ends of the ball switch, as shown in the figure below.





## 14-Flame alarm

### The purpose of the experiment:

In this lesson, we need to complete the experiment of fire alarm. The experimental effect is: when there is no fire source approaching, the circuit is normal. When there is a fire source approaching, the buzzer will make a sound.

#### Introduction of flame sensor:

The actual object is shown below. Flame sensor (Infrared receiving triode), Because infrared is very sensitive to flame, we use a special infrared receiver tube to detect the flame, and then convert the brightness of the flame into a level signal of high and low change, and we need to input these signals into the MCU. Finally the MCU makes corresponding program processing according to the change of the signals.



### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω resistor \*1

10kΩ resistor \*1

Tilt switch \*1

Breadboard \*1

Dupont line \*1 bunch

#### Actual object connection diagram :

We need to connect the circuit as shown in the figure below.





## Experimental code analysis:

```
int flame=A5; //Defining the analog port A5
int Beep=8; //Defining the digital port 8
int val=0; //Declarations of variables
void setup()
{
pinMode(Beep,OUTPUT); //Defining the digital port for the output port
pinMode(flame,INPUT); //Defining the analog port for the input port
Serial.begin(9600);//The baud rate is 9600
val=analogRead(flame); //Read analog port voltage
}
void loop()
Serial.println(analogRead(flame)); //The serial port sends the simulated voltage value
if((analogRead(flame)-val)>=600) //Determine whether the simulated voltage value is
greater than 600
digitalWrite(Beep,HIGH);
else
   digitalWrite(Beep,LOW);
}
Experimental steps:
```

1.We need to open the code of this experiment: code-Tilt\_switch.ino, click" $\sqrt{}$ " under the menu bar to compile the code, and wait for the word "Done compiling " in the lower right corner, as shown in the figure below.



2. In the menu bar of Arduino IDE, we need to select 【Tools】--- 【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.

YVHSOON

☆(生(F) 掃作(Δ)				
with with a	查看(V) 帮助(H)			
= =>   ==   🖬				
A Xiaozhen				
P CA IDE AT	A/ATAPI 控制器			
🖟 👰 Jungo 🕯	Connectivity			
▶ 🔲 处理器				
・ 🕞 磁盘驱荡	动器			
▲·徑 第□ (C	OM 和 LPT)			
- 🐨 Blue	etooth Serial Port (CC	OM15)		
- 🐨 Blue	etooth Serial Port (CC	OM16)		
- W Blue	stooth Serial Port (CC	OM17)		
Blue	etooth Serial Port (CC	DM18)		
- Blue	etooth Serial Port (CC	DM19)		
Blue	etooth Serial Port (CC	JM20)		
Blue	stooth Serial Port (CC	JM21)		
Blue	tooth Serial Port (CC	JW(22)		
Blue	etooth Serial Port (CC	JIV123)		
- WOSE	S-SERIAL CH340 (CON	VIO)		
▶ 🐙 计算机	满口 (COM1)			
code Flame_alarr	) 所日 (COM1) n   Acduino 1.7.8			
Code Flame alarr Edit Sketch II	前日 (COM1) n   Arduino 1.7.8 rols: Help Auto Format	Qrf+T		
Code-Flame_slatt Edit Sketch	前日 (COM1) n   Accluino 1.7.8 Roli: Help Auto Format Archive Sketch	Ctri+T		× (*) (*)
Code Flame slam Edit Sketch fre Ode Flame al	前日 (COM1) n   Atclaino 1.7.8 Note: Help Auto Format Archive Sketch Fot Encoding & Relo	Ctrl+T		<b>4</b>
ロック 通信 ・ 小事 计算机 code Flame_slam Edit Sketch for いのの Flame_ala pin	前日 (COM1) n   Arduino 1.7.8 iok: Help Auto Format Archive Sketch Fox Encoding & Relo Serial Monitor	Ctrl+T bad Ctrl+Shift+M	the digital	port for
Code-Flame_slarr Edit Sketch fr Ode Flame_ali pin pin	前日 (COM1) n   Acduino 1.7.8 Notic Help Auto Format Archive Sketch For Encoding & Relo Serial Monitor Board	Ctrl+T badi Ctrl+Shift+M	the digital	port for -
Code Finme alarr Edit Sketch fre Ode Finme al pin pin Ser	前日 (COM1) n   Arclaino 1.7.8 iolis Help Auto Format Archive Sketch For Encoding & Relc Serial Monitor Board Port	Ctrl+T sed Ctrl+Shift+M	the digital the snalog p Serial ports	part for for the
Code-Flame_atarr Edit Sketch To Ode-Flame_ata pin pin Ser val	画 (COM1) n   Actuino 1.7.8 zois: Help Auto Format Archive Sketch For Encoding & Relo Serial Monitor Board Port Processments	Ctri+T ved Ctrl+Shift+M	the digital the analog p Serial ports COM1	port for and
Code-Flame_slaff Edit Sketch To Doe Flame_sl pin pin Ser val	画 (COM1)  n (Arclaino 1.7.8  n) Arclaino 1.7.8  n) Help Auto Format Archive Sketch For Encoding & Relo Serial Monitor  Board Port Programmer Boun Bootheader	Ctrl+T add Ctrl+Shift+M	the digital the snalog p Serial ports COM1 COM6	port for the second sec
Code-Flame_slars Edit Sketch fre Def Flame_als pin pin Ser val	画 (COM1) n   Acduino 1.7.8 Note: Help Auto Format Archive Sketch Fox Encoding & Relo Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T ctrl+Shift+M	the digital the analog ports COM1 COM1 COM15	port for
Code Flame_slam Edit Sketch fre De Flame_ala pin pin Ser val	前日 (COM1) n   Actuino 1.7.8 Note: Help Auto Format Archive Sketch For Encoding & Reic Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T oad Ctrl+Shift+M	the digital the analog p Serial ports COM1 V COM6 COM15 COM16	port for the set of th
code Flame alar Edit Sketch Te Din pin pin Ser val	前日 (COM1) n   Arclaino 1.7.8 iols: Help Auto Format Archive Sketch For Encoding & Reic Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T oed Ctrl+Shift+M	the digital the analog p Serial ports COM1 COM15 COM15 COM16 COM17	part for for the second
code Flame alar Edit Sketch fre ode Flame_ala pin pin Ser val	前日 (COM1) n   Arduino 1.7.8 Note: Help Auto Format Archive Sketch Fox Encoding & Relc Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T xad Ctrl+Shift+M	he digital he analog r Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18	port for age
code Flame, atam r Edit Sketch fr code Flame_al pin pin pin Ser val	画 (COM1) n   Arclaino 1.7.8 Nois: Help Auto Format Archive Sketch For Encoding & Relo Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T oad Ctrl+Shift+M	the digital the analog ports COM1 COM15 COM16 COM16 COM17 COM18 COM19	port for oort for ti age
code Flame, atam e Edit Sketch fr code Flame, ata pin pin Ser val sid loop () Seria	画 (COM1) n   Acduino 1.7.8 Nois: Help Auto Format Archive Sketch For Encoding & Relo Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T ctrl+Shift+M Seed (flame) )-val) >=600)	the digital the analog r Serial ports COM1 V COM6 COM15 COM16 COM17 COM18 COM19 COM19 COM20	port for the set of th
code Flame, alam e Edit Sketch fis code Flame, ala pin pin Ser val sid loop() Seria if((a	im (COM1) n   Actuino 1.7.8 iols Help Auto Format Archive Sketch For Encoding & Reic Serial Monitor Board Port Programmer Burn Bootloader I. pr Intln (and lo millogRead (f lane	Ctrl+T oad Ctrl+Sbift+M eRead(flame) e) =val) >=600) (Heen, HIGH)	the digital the analog p Serial ports COM1 COM15 COM16 COM16 COM17 COM18 COM19 COM20 COM21	port for the set of th

3.After the selection is completed, you need to click " $\rightarrow$ " under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, thecode has been successfully uploaded to the Arduino UNO board, as shown in the figure below.





4.After the code is uploaded. When there is no fire source approaching, the circuit is normal. When there is a fire source approaching, the buzzer will make a sound to indicate the alarm. We can also open the serial monitor to observe the change in the value of the flame sensor, as shown in the figure below.





## **15-Nixie tube**

### The purpose of the experiment:

In this experiment, we need to finish to display 1-9 on a single 8-segment Nixie tube.

### Introduction to digital tube:

Nixie tube is a semiconductor luminescent device, its basic unit is a light-emitting diode. It is divided into 7-segment Nixie tube and 8-segment Nixie tube. 8-segment Nixie tube more than 7-segment Nixie tube a light-emitting diode unit (more than a decimal point), this experiment we use the 8-segment Nixie tube. The actual object is shown below.



According to the light-emitting diode unit connection mode, it is divided into anode Nixie tubes and cathodeNixie tubes.

Anode Nixie tubes that connects the anodes of all light-emitting diodes together to form a common anode (COM). The common pole (COM) shall be connected to +5V when the common anode digital tube is applied. When the cathode of a certain field of light-emitting diode is low, the corresponding field will be light up. When the cathode of a field is high, the field does not light up.

Cathode Nixie tubes that connects the cathodes of all light-emitting diodes together to form a common cathode (COM). The common pole COM shall be connected to GND when the common cathode digital tube is applied. When the anode of a certain field of light-emitting diode is high , the corresponding field will be light up. When the anode of a field is low, the field does not light up.

#### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω resistor \*8

8-segment digital tube \*1

Breadboard \*1

Dupont line \*1bunch

#### Actual object connection diagram :

We need to connect the circuit as shown in the figure below.

220Ω resistor Nixie tube : ..... ..... ..... ..... .. 22 dupont line Breadboard JNO Arduino UNO board

## Experimental code analysis:

```
int a=7; // Digital port 7 is connected to digital tube section a
int b=6; // Digital port 6 is connected to digital tube section b
int c=5; // Digital port 5 is connected to digital tube section c
int d=11; // Digital port 11 is connected to digital tube section d
int e=10; //Digital port 10 is connected to digital tube section e
int f=8; //Digital port 8 is connected to digital tube section f
int g=9; //Digital port 9 is connected to digital tube section g
int dp=4; //Digital port 4 is connected to digital tube decimal point section
void digital_1(void) //Displaying 1
{
unsigned char j;
digitalWrite(c,HIGH); //Light digital tube section c
digitalWrite(b,HIGH); //Light digital tube section b
for(j=7;j<=11;j++) //The level is pulled low of tube section 7~11(a,f,q,e,d)
digitalWrite(j,LOW);
digitalWrite(dp,LOW); //Tube decimal point section is off
}
void digital_2(void) //Displaying 1
{
unsigned char j;
digitalWrite(b,HIGH);
digitalWrite(a,HIGH);
for(j=9;j<=11;j++)
digitalWrite(j,HIGH);
digitalWrite(dp,LOW);
digitalWrite(c,LOW);
digitalWrite(f,LOW);
```



```
}
void digital_3(void) //Displaying 3
{
unsigned char j;
digitalWrite(g,HIGH);
digitalWrite(d,HIGH);
for(j=5;j<=7;j++)
digitalWrite(j,HIGH);
digitalWrite(dp,LOW);
digitalWrite(f,LOW);
digitalWrite(e,LOW);
void digital_4(void) //Displaying 4
{
digitalWrite(c,HIGH);
digitalWrite(b,HIGH);
digitalWrite(f,HIGH);
digitalWrite(g,HIGH);
digitalWrite(dp,LOW);
digitalWrite(a,LOW);
digitalWrite(e,LOW);
digitalWrite(d,LOW);
}
void digital_5(void) //Displaying 5
{
unsigned char j;
for(j=7;j<=9;j++)
digitalWrite(j,HIGH);
digitalWrite(c,HIGH);
digitalWrite(d,HIGH);
digitalWrite(dp,LOW);
digitalWrite(b,LOW);
digitalWrite(e,LOW);
}
void digital_6(void) //Displaying 6
ł
unsigned char j;
for(j=7;j<=11;j++)
digitalWrite(j,HIGH);
digitalWrite(c,HIGH);
digitalWrite(dp,LOW);
digitalWrite(b,LOW);
void digital_7(void) //Displaying 7
{
unsigned char j;
for(j=5;j<=7;j++)
digitalWrite(j,HIGH);
digitalWrite(dp,LOW);
for(j=8;j<=11;j++)
digitalWrite(j,LOW);
}
```



```
void digital_8(void) //Displaying 8
{
unsigned char j;
for(j=5;j<=11;j++)
digitalWrite(j,HIGH);
digitalWrite(dp,LOW);
}
void digital_9(void) //Displaying 9
digitalWrite(a,HIGH);
digitalWrite(b,HIGH);
digitalWrite(c,HIGH);
digitalWrite(d,HIGH);
digitalWrite(e,LOW);
digitalWrite(f,HIGH);
digitalWrite(g,HIGH);
digitalWrite(dp,HIGH);
}
void setup()
{
int i; //Declarations of variables
for(i=4;i<=11;i++)
pinMode(i,OUTPUT); //Defining the port4-11 for the input port
}
void loop()
{
 while(1)
 {
   digital_1(); //Displaying 1
  delay(1000);
   digital_2(); //Displaying 2
   delay(1000);
   digital_3(); //Displaying 3
   delay(1000);
   digital_4(); //Displaying 4
   delay(1000);
   digital_5(); //Displaying 5
   delay(1000);
   digital_6(); //Displaying 6
   delay(1000);
   digital_7(); //Displaying 7
   delay(1000);
   digital_8(); //Displaying 8
  delay(1000);
   digital_9(); //Displaying 9
  delay(1000);
 }
}
```

#### **Experimental steps:**

1.We need to open the code for this experiment: code-Tilt\_switch.ino, click " $\sqrt{}$ "under the menu bar,compile the code, and wait for the words of Done compiling in the lower left corner, as shown in the following figure.





2. In the menu bar of Arduino IDE, we need to select 【Tools】--- 【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below. For example:COM6,as shown in the following figure.





Code Nine tube File Edit Sketch	Arduino 1.7.8			
code-Nuse_tube	Auto Format Archive Sketch Fix Encoding & Reloa	Ctrl+T		20
int a=7;//	Serial Monitor	Ctrl+Shift+M	b digital tu	ibe sec."
int b=6,//	Board	,	b digital tu	be sec
int c=5;//	Port		Serial ports	le sec
int d=11,//	Programmer		COM1	ube s
int e=10;//	Burn Bootloader		COM6	tine e
int f=8:// D	isital port 8 1	corpected.	COM15	the set
int e=9 // D	igital news 0 1		COM16	we se
int durt // n	ASISAL MARC - 10	si donu so ted	COM17	the se
1mt dp-4,// 1	feital port 4 4	connected	COM18	the de
void digital_	1(void) //Displa	Wire I	COM19	
1			COM20	
2.17	(Hr.)		COM21	. *
Done convoluna			COM22	
lobal variabl eaving 2,023 1.048 hytes	es use 25 bytes bytes for local	(1%) of dyr variables.	amic memory, Maximum is	

3. After the selection is completed, you need to click " $\rightarrow$ "" under the menu bar to upload the code to the Arduino UNO board. When the word "Done uploading" appears in the lower left corner, the code has been successfully uploaded to the Arduino UNO board, as shown in the figure below.

💿 code-Nixie_tube   Arduino 1.7.8
code-Nbie_tube
int a=7;// Digital port 7 is connected to digital tu
int b=6;// Digital port 6 is connected to digital tu
<pre>int c=5;// Digital port 5 is connected to digital tul</pre>
<pre>int d=11;// Digital port 11 is connected to digital -</pre>
<pre>int e=10;// Digital port 10 is connected to digital</pre>
int f=8;// Digital port 8 is connected to digital t
<pre>int g=9;// Digital port 9 is connected to digital to</pre>
<pre>int dp=4;// Digital port 4 is connected to digital to</pre>
<pre>void digital_1(void) //Displaying 1</pre>
×
Done uploading:
memory leaving 2 023 bytes for local variables
Maximum is 2.048 bytes
actual 13 2,040 0,003.
127 Arduine Une on COM5

4. After the code is uploaded, we can see that display 1-9 on a single 8-segment digital tube, as shown in the figure below.






## 16-4-Nixie tube

#### The purpose of the experiment:

In this experiment, arduino was used to drive a four-digit tube with a common Yin. Is the purpose of the experiment: the first Nixie tube display 1,the second Nixie tube display 2, the third Nixie tube display 3 and fourth Nixie tube display4 with such intervals of 0.5 seconds to display.

#### Introduction to digital tube:

Nixie tube is a semiconductor luminescent device, its basic unit is a light-emitting diode. According to the number of digital tube is divided into 7-segment Nixie tube and 8segment Nixie tube. 8-segment Nixie tube more than 7-segment Nixie tube a lightemitting diode unit (more than a decimal point), this experiment use the8segment Nixie tube.The actual object is shown below.



According to the light-emitting diode unit connection mode, it is divided into anode Nixie tubes and cathodeNixie tubes.

Anode Nixie tubes that connects the anodes of all light-emitting diodes together to form a common anode (COM). The common pole COM shall be connected to +5V when the common anode digital tube is applied. When the cathode of a certain field of light-emitting diode is low, the corresponding field will be light up. When the cathode of a field is high, the field does not light up.

Cathode Nixie tubes that connects the cathodes of all light-emitting diodes together to form a common cathode (COM). The common pole COM shall be connected to GND when the common cathode digital tube is applied. When the anode of a certain field of light-emitting diode is high , the corresponding field will be light up. When the anode of a field is low, the field does not light up.

#### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

220Ω resistor \*8

4bit 8-segment digital tube \*1

Breadboard \*1

dupont line \*1bunch

Actual object connection diagram :

We need to connect the circuit as shown in the figure below.



#### Experimental code analysis:

#define SEG\_A 2 //Arduino Pin2--->SegLed Pin11 #define SEG\_B 3 //Arduino Pin3--->SegLed Pin7 #define SEG\_C 4 //Arduino Pin4--->SegLed Pin4 #define SEG D 5 //Arduino Pin5--->SegLed Pin2 #define SEG\_E 6 //Arduino Pin6--->SegLed Pin1 #define SEG\_F 7 //Arduino Pin7--->SegLed Pin10 #define SEG G 8 //Arduino Pin8--->SegLed Pin5 #define SEG\_H 9 //Arduino Pin9--->SegLed Pin3 #define COM1 10 //Arduino Pin10--->SegLed Pin12 #define COM2 11 //Arduino Pin11--->SegLed Pin9 #define COM3 12 //Arduino Pin12--->SegLed Pin8 #define COM4 13 //Arduino Pin13--->SegLed Pin6 unsigned char table[10][8] =  $\{0, 0, 1, 1, 1, 1, 1, 1\}, //0$  $\{0, 0, 0, 0, 0, 0, 1, 1, 0\}, //1$  $\{0, 1, 0, 1, 1, 0, 1, 1\}, //2$ {0, 1, 0, 0, 1, 1, 1, 1}, //3  $\{0, 1, 1, 0, 0, 1, 1, 0\}, //4$  $\{0, 1, 1, 0, 1, 1, 0, 1\}, //5$ {0, 1, 1, 1, 1, 1, 0, 1}, //6  $\{0, 0, 0, 0, 0, 1, 1, 1\}, //7$ {0, 1, 1, 1, 1, 1, 1, 1}, //8 {0, 1, 1, 0, 1, 1, 1, 1} //9 }; void setup() { pinMode(SEG\_A,OUTPUT); //Defining the port for the output port pinMode(SEG\_B,OUTPUT); pinMode(SEG C,OUTPUT); pinMode(SEG D,OUTPUT); pinMode(SEG\_E,OUTPUT);



```
pinMode(SEG_F,OUTPUT);
pinMode(SEG_G,OUTPUT);
pinMode(SEG_H,OUTPUT);
pinMode(COM1,OUTPUT);
pinMode(COM2,OUTPUT);
pinMode(COM3,OUTPUT);
pinMode(COM4,OUTPUT);
void loop()
Display(1,1); //Displaying 1 on the first bit of the Nixie tube
delay(500);
Display(2,2); //Displaying 2 on the second bit of the Nixie tube
delay(500);
Display(3,3); //Displaying 3 on the third bit of the Nixie tube
delay(500);
Display(4,4); //Displaying 4 on the fourth bit of the Nixie tube
delay(500);
}
void Display(unsigned char com, unsigned char num)
digitalWrite(SEG_A,LOW); //This is to get rid of the shadow
digitalWrite(SEG B,LOW);
digitalWrite(SEG_C,LOW);
digitalWrite(SEG D,LOW);
digitalWrite(SEG E,LOW);
digitalWrite(SEG_F,LOW);
digitalWrite(SEG_G,LOW);
digitalWrite(SEG_H,LOW);
switch(com) //This is to select the display location
{
case 1:
digitalWrite(COM1,LOW); //First bit of the Nixie tube
digitalWrite(COM2,HIGH);
digitalWrite(COM3,HIGH);
digitalWrite(COM4,HIGH);
break:
case 2:
digitalWrite(COM1,HIGH);
digitalWrite(COM2,LOW); //Second bit of the Nixie tube
digitalWrite(COM3,HIGH);
digitalWrite(COM4,HIGH);
break;
case 3:
digitalWrite(COM1,HIGH);
digitalWrite(COM2,HIGH);
digitalWrite(COM3,LOW); //Third bit of the Nixie tube
digitalWrite(COM4,HIGH);
break:
case 4:
digitalWrite(COM1,HIGH);
```



digitalWrite(COM2,HIGH); digitalWrite(COM3,HIGH); digitalWrite(COM4,LOW); //Fourth bit of the Nixie tube break; default:break; } digitalWrite(SEG\_A,table[num][7]); digitalWrite(SEG\_B,table[num][6]); digitalWrite(SEG\_C,table[num][6]); digitalWrite(SEG\_C,table[num][5]); digitalWrite(SEG\_E,table[num][5]); digitalWrite(SEG\_E,table[num][2]); digitalWrite(SEG\_F,table[num][2]); digitalWrite(SEG\_G,table[num][1]); digitalWrite(SEG\_H,table[num][0]);

}

Experimental steps:

1.We need to open the code for this experiment: code-4-Nixie\_tube.ino, click " $\sqrt{}$ "under the menu bar,compile the code, and wait for the words of Done compiling in the lower left corner, as shown in the following figure.



2. In the menu bar of Arduino IDE, we need to select the 【Tools】---【Port】--- select the port that the serial number displayed by the device manager just now.for example:COM6,as shown in the following figure.

YAHBOOM

设备管理器		
3件(F) 操作(A) 查	(著(V) 帮助(H)	
+ 🖬 🖬 🖬	14	
A Xiaozhen		
De ATA/AT	API 控制器	
👂 🔮 Jungo Conn	nectivity	
▶ ■ 处理器		
一直磁量驱动器		
·小小 第日 (COM )	和 LPT)	
TO Bluetoot	th Serial Port (COM15)	
- Bluetoot	th Serial Port (COM17)	
Bluetoot	th Serial Port (COM18)	
- Bluetoot	th Serial Port (COM19)	
- T Bluetoot	th Serial Port (COM20)	
- Bluetoot	th Serial Port (COM21)	
Bluetoot	th Serial Port (COM22)	
Bluetoot	th Senai Port (COM23)	
TO 通信法门	(COM1)	
「三日日日」	(0001)	
code 4 Nixie tub	e (Arduino 1.7.8	
code 4 Nixe tub r Edit Sketch To	ools Help	
coste 4 Nisie tub r Edit Sketch <mark>T</mark> e	Archuino 1.7.8 Dools Help Auto Format Ctrl+T	
code 4 Noie tub r Edit Sketch Fr	ools Help Auto Format Ctrl+T Archive Sketch	
coste 4 Niose tub Edit Sketch To COLLER code-4 Nizie_tu	ools Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload	
code 4 Nixe tub e Edit Sketch Tr COLORE code 4 Nixie_tu	oots Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M	
coste 4 Note tub Edit Sketch To Colore code-4-Nizie_tu	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board	3, HIGH) ;
code 4 Nixe tub r Edit Sketch Tr COLOR code 4 Nixie_tu	er Archaine 1.7.8 oots Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board	B, HIGH) ; 1, HIGH) ; 5 serial ports
code-4-Nixie_tu	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port	3, HIGH) : Serial ports COMI
code-4-Nixie_tu	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port	3, HIGH) : Serial ports COMI
code 4 Nixie_tu Edit Sketch Code 4 Nixie_tu	be   Archuino 1.7.8 bols   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	3, HIGH) : Serial ports COMI COMI
code 4 Nixie_tu	be   Archaine 1.7.8 bools' Help Auto Format Ctri+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	A HIGH) : A HIGH) : A HIGH) : COMI ✓ COM6 COM15 COM15
code-4-Nixie_tu	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader	3, HICH) : 4, HICH) : 5, HICH) : 5, HICH) : 5, HICH) : 5, COM1 ✓ COM6 COM15 COM15 COM16 COM17
code-4-Nixie_tu	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Ctrl+Shift+M Board Port 9 Programmer 9 Burn Bootloader digital Write (CON digital Write (CON	3, HIGH) : Serial ports COMI COM15 COM16 COM17
code-4-Nixie_tu e Edit Sketch Tr Code-4-Nixie_tu	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite (COM digitalWrite (COM	3, HIGH) : 1, HIGH) : 1, HIGH) / COM1 COM15 COM15 COM16 COM17 COM18
code 4 Nixie_tu code 4 Nixie_tu	be   Archaine 1.7.8 bools' Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Ctrl+Shift+M Board Port Programmer Burn Bootloader digital Write (CON digital Write (CON break	A, HIGH) : A, HIGH) : A, HIGH) : A, HIGH) : COM1 ✓ COM6 COM15 COM15 COM16 COM17 COM18 COM19
code 4 Nixe tub e Edit Sketch Tr Code 4 Nixie_tu	be   Archuino 1.7.8 boots   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Ctrl+Shift+M Board Port Programmer Burn Bootloader digitalWrite (COM digitalWrite (COM digitalWrite (COM digitalWrite (COM	A, HICH) : A, HICH) : A, HICH) : COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20
code-4-Nixie_tu	be   Archuino 1.7.8 boots   Help Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Ctrl+Shift+M Board Port Programmer Burn Bootloader digital Write (COM digital Write (COM digital Write (COM break ; case 1:	<ul> <li>→ HIGH) :</li> <li>→ HIGH) :</li> <li>→ HIGH) :</li> <li>→ HIGH) :</li> <li>→ COM1</li> <li>→ COM1</li> <li>→ COM16</li> <li>→ COM16</li> <li>→ COM17</li> <li>→ COM18</li> <li>→ COM20</li> <li>→ COM21</li> </ul>
code-4-Nixie_tu	Auto Format Ctrl+T Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload Serial Monitor Ctrl+Shift+M Board Ctrl+Shift+M Board Port Programmer Burn Bootloader digital Write (CON digital Write (CON digital Write (CON break. case 1: digital Prite (CON	3, HIGH) : 3, HIGH) : 4, HIGH) : 5erial ports COM1 COM1 COM15 COM16 COM17 COM18 COM19 COM20 COM21 COM21 COM22

3. After the selection is completed, click " $\rightarrow$ " under the menu bar,and upload the code to the Arduino UNO board, when appears to Done uploading on the lower left corner, that means that the code has been successfully uploaded to the Arduino UNO board, as shown in the following figure.



code-4-Nixie_tube   A	rduino 1.7.8	
COEDD		2
code-4-Nixie_tube		
#define SEG_A 2	//Arduino Pin2>SegLed Pin11	•
#define SEG_B 3	//Arduino Pin3>SegLed Pin7	=
#define SEG_C 4	//Arduino Pin4>SegLed Pin4	
#define SEG_D 5	//Arduino Pin5>SegLed Pin2	
#define SEG_E 6	//Arduino Pin6>SegLed Pin1	
#define SEG_F 7	//Arduino Pin7>SegLed Pin10	
#define SEG_G 8	//Arduino Pin8>SegLed Pin5	
#define SEG_H 9	//Arduino Pin9>SegLed Pin3	
#define COM1 10	//Arduino Pin10->SegLed Pin12	
#define COM2 11	//Arduino Pinl1>SegLed Pin9	
<	m	•
Done uploading.		
memory, leaving	1,959 bytes for local variables.	*
Maximum is 2,048	bytes.	
		-
107	Arduino Uno en C	OMO

4. After the code is uploaded, the first Nixie tube display 1,the second Nixie tube display 2, the third Nixie tube display 3 and fourth Nixie tube display4 with such intervals of 0.5 seconds to display.





### 17-74HC595

#### The purpose of the experiment:

74HC595 is an 8-bit serial input and parallel output displacement buffer: the parallel output is three-state output. In this course, we use three digital I/O ports of Arduino to control 8 LED lights by 74HC595, so that they were lit in 8-bit binary (0-256) order.

The actual object is shown below.



Binary order: 00000001 00000010 00000011 00000100 00000101 00000110 00000111 00001000 00001001 00001010 00001011 00001100





number of pin	name of pin	Description
1,2,3,4,5,6,7,15	QB,QC,QD,QE,QG,QH,QA	Tri-state output pin
8	GND	GND
9	SQH	Serial port data output pin
10	SCLR	Shift register clear
11	SCK	Data input clock line
12	RCK	Output memory latch clock line
13	OE	Output enable
14	SI	Data line



16 VCC

VCC

### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

74HC595 \*1

 $220\Omega$  resistor \*8

LED \*8

Breadboard \*1

Dupont line \*1bunch

### Material object connection diagram :

We need to connect circuit as shown in the following figure.



### Experimental code analysis:

```
//connect 74hc595 pin10:MR--->VCC; Pin13:OE--->GND
int latchPin = 5; //to 595 pin12
int clockPin = 4; //to 595 pin11
int dataPin = 2; //to 595 pin14
void setup ()
{
    pinMode(latchPin,OUTPUT); //Defining the port5 for the output port
    pinMode(clockPin,OUTPUT); //Defining the port4 for the output port
    pinMode(dataPin,OUTPUT); //Defining the port2 for the output port
}
```



```
void loop()
```

{

for(int a=0; a<256; a++) //The meaning of this loop is to let a variable increase by 1 until it is equal to 256.

{

//The following activities are performed every cycle.

digitalWrite(latchPin,LOW); //Giving a low level to the port ST\_CP indicates that the chip is ready to receive data.

shiftOut(dataPin,clockPin,MSBFIRST,a);

/\*

dataPin : Data output pin, each bit of data will be output sequentially. Mode of pin needs to be set to output.

clockPin : Clock output pin. Mode of pin needs to be set to output

bitOrder : Data shift order selection bit. The type of this parameter is byte,

High-level first-entry MSBFIRST or low-level first-entry LSBFIRST Can be selected by youself.

a:The data value to be output.

```
*/
```

}

```
digitalWrite(latchPin,HIGH); //Giving a low level to the port ST_CP delay(1000); //Pause for 1 second to make you see the effect
```

### Experimental steps:

1.We need to open the code for this experiment: code-74HC595.ino, click " $\sqrt{}$ "under the menu bar,compile the code, and wait for the words of Done compiling in the lower left corner, as shown in the following figure.

💿 code-74HC595   Arduino 1.7.8	
File Edit Sketch Tools Help	
code-74HC595	
//connect 74hc595 pin10:MR>VCC; Pin13:OE>GN	*
<pre>int latchPin = 5;//to 595 pin12</pre>	
<pre>int clockPin = 4;//to 595 pin11</pre>	
<pre>int dataPin = 2; //to 595 pin14</pre>	1
void setup ()	
{	
<pre>pinMode(latchPin, OUTPUT); //Defining the port5</pre>	
<pre>pinMode(clockPin, OUTPUT); //Defining the port4</pre>	
pinMode(dataPin,OUTPUT); //Defining the port2 f	
}	-
	_
Done compiling	
memory, leaving 2,033 bytes for local variables.	*
Maximum is 2,048 bytes.	
	-
	*

2. In the menu bar of Arduino IDE, we need to select the 【Tools】---【Port】--- select the port that the serial number displayed by the device manager just now.for example:COM6,as shown in the following figure.



and an an and a second	10000 A.A		
(中) (F) 我们(A)	登禮(V) 報助(H)		
	111   1 <b>1</b> 4		
Xiaozhen			
D-CALIDE ATA	ATAPI 控制器		
Jungo Ci	onnectivity		
2 SCORES			
	M RI LDTI		
TT Bluet	ooth Serial Port (COM15)		
- TOP Bluet	ooth Serial Port (COM16)		
T Bluet	ooth Serial Port (COM17)		
-17 Bluet	ooth Serial Port (COM18)		
- Bluet	ooth Serial Port (COM19)		
- T Bluet	ooth Serial Port (COM20)		
- Blueb	ooth Serial Port (COM21)		
- Bluet	ooth Serial Port (COM22)		
Bluet	poth Serial Port (COM23)		
USB-	SERIAL CH340 (COMb)		
1.11.11.11.11.11.11.11.11.11.11.11.11.1	ET (COMI)		
1. 1. 11 24 D.F.			
and a party state	And and a set of		The state of the s
ode-74HC595	Arduino 1.7.8		
eode-74HC595 Edit Sketch	Arduino 1.7.8 ools Help	Child	
code-74HC595 Edit Sketch	Archuino 1.7.8 ools Help Auto Format Archive Sketch	Ctrl+T	
ode-74HC595 Edit Sketch 1	Arduino 1.7.8 ools Help Auto Format Archive Sketch	Ctrl+T	
ode-74HC595	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor	Ctrl+T	
ode-74HCS95 Edit Sketch Ode-74HC595 CONDECT	Arcluino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor	Ctrl+T Ctrl+Shift+M	In13:08
ede-74HC595 Edit Sketch COLDE ode-74HC595 connect t latchP	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board	Ctrl+T Ctrl+Shift+M	In13:0E
ede-74HC595 Edit Sketch Ode-74HC595 Connect t latchP t clockP	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port	Ctrl+T Ctrl+Shift+M	In13:0E Serial ports
ede-74HC595 Edit Sketch i ode-74HC595 connect t latchP t clockP t dataPi	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer	Ctrl+T Ctrl+Shift+M	In13:08 Serial ports COM1
ede-74HC595 Edit Sketch Connect t latchP t clockP t dataPi	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T Ctrl+Shift+M	In13:0B Serial ports COM1
t dataPi	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T Ctrl+Shift+M	In13:0E Serial ports COM1 COM15
ede-74HC595 Edit Sketch Ode-74HC595 Connect t latchP t clockP t dataPi id setup	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T Ctrl+Shift+M	In13:0E Serial ports COM1 COM15 COM15 COM16
ode-74HC595 Edit Sketch T ode-74HC595 connect t latchP t clockP t dataPi id setup	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T Ctrl+Shift+M	COM1 COM15 COM15 COM15
ode-74HC595 Edit Sketch College ode-74HC595 connect t latchP t clockP t dataPi id setup	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader	Ctrl+T Ctrl+Shift+M	Serial ports COM1 COM15 COM15 COM16 COM17 COM18
ode-74HC595 Edit Sketch Collect ode-74HC595 Connect t latchP t clockP t dataPi id setup cinMode(1 pinMode(c	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader V atchPin, OUTPUT) lockPin, OUTPUT)	Ctrl+T Ctrl+Shift+M //Defini //Defini	In13:0E Serial ports COM1 COM15 COM15 COM16 COM17 COM18 COM19
ode-74HC595 Edit Sketch Ode-74HC595 connect t latchP t clockP t dataPi id setup pinMode(l pinMode(c	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader V atchPin, OUTPUT) lockPin, OUTPUT)	Ctrl+T Ctrl+Shift+M //Defini //Defini	Senial ports COM1 COM15 COM15 COM16 COM15 COM16 COM17 COM18 COM19 COM20
ode-74HC595 Edit Sketch T Dide-74HC595 connect t latchP t clockP t dataPi id setup inMode(l sinMode(d	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader V atchPin, OUTPUT) lockPin, OUTPUT) ataPin, OUTPUT)	Ctrl+T Ctrl+Shift+M //Defini //Defini	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21
ede-74HC595 Edit Sketch Connect t latchP t clockP t dataPi id setup pinMode(l pinMode(d	Arduino 1.7.8 ools Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor Board Port Programmer Burn Bootloader V atchPin, OUTPUT) lockPin, OUTPUT) ataPin, OUTPUT)	Ctrl+T Ctrl+Shift+M //Defini //Defini //Definir	Serial ports COM1 COM15 COM15 COM15 COM16 COM17 COM18 COM19 COM20 COM21 COM21 COM22

3. After the selection is completed, you need to click " $\rightarrow$ "under the menu bar,and upload thecode to the Arduino UNO board, when appears to Done uploading on the lower left corner , that means that the code has been successfully uploaded to the Arduino UNO board, as shown in the following figure.





4. After the code is uploaded, We can see that 8 LEDs will be lit from 00000001 to 10000000, as shown in the following figure.(Just an example)





## **18-servo control**

#### The purpose of the experiment:

Based on Arduino UNO, a code is written to rotate the servo to the angle corresponding to the user's input number, and the angle print is displayed on the serial monitor of the Arduino IDE.

About the servo :

The actual object is shown below. Servo rotation angle is by adjusting the duty ratios of PWM (pulse width modulation) signal. The standard PWM (pulse width modulation) signal has a fixed period of 20ms (50Hz). Theoretically, pulse width distribution should be between 1 ms to 2 ms, but in fact between pulse width can be 0.5 ms and 2.5 ms. Pulse width and the servo rotation angle  $0^{\circ} \sim 180^{\circ}$  corresponds, as shown in the figure below.





Servo have many specifications, but all of the servo possess external three lines, with brown, red, orange, three kinds of color to distinguish. Due to brand is different, color is different, brown for the grounding line, red for positive line, orange for signal lines.

Note: Due to brand is different, for the same signal, different brands of servo rotation angle will be different.

List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

Servo \*1

Dupont line \*1 bunch

Actual object connection diagram :

We need to connect the circuit as shown in the figure below.

```
YAHBOOM
```



#### Experimental code analysis:

```
//UART send 1~9==>20~180 degree
int servopin=9;//Defining the port 9 for the servopin
int myangle;//Define Angle variable
int pulsewidth;//Define the pulse width variable
int val:
void servopulse(int servopin, int myangle)
/*A pulse function is defined to generate PWM values by simulation */
 pulsewidth=(myangle*11)+500://Convert the Angle to 500-2480 pulse width
 digitalWrite(servopin,HIGH);//Giving a high level to the servo interface
 delayMicroseconds(pulsewidth);//The number of microseconds of delay pulse width
 digitalWrite(servopin,LOW);//Giving a low level to the servo interface
 delay(20-pulsewidth/1000);//The remaining time in the delay period
}
void setup()
{
 pinMode(servopin,OUTPUT);//Defining the servopin port for the output port
 Serial.begin(9600);//The baud rate is 9600
 Serial.println("servo=o_seral_simple ready" );
}
void loop()
{
val=Serial.read();//Reading the data received by the serial port
if(val>'0'&&val<='9')//Determineing whether the received data values conform to the
range
{
val=val-'0';//Convert ASCII code to a value, for exmaple : '9'-'0'=0x39-0x30=9
val=val*(180/9);//Convert Numbers into angles, for exmaple : 9* (180/9) =180
Serial.print("moving servo to ");
```



```
Serial.print(val,DEC);
Serial.println();
for(int i=0;i<=50;i++)
//Generate the number of PWM, equivalent delay to ensure that the response Angle
can be turned
{
servopulse(servopin,val);//Generate PWM values by simulation
}
}
Experimental steps:
```

1.We need to open the code for this experiment: code-servo\_control.ino, click " $\sqrt{}$ "under the menu bar, compile the code, and wait for the words of Done compiling in the lower left corner, as shown in the following figure.



2. In the menu bar of Arduino IDE, you need to select the 【Tools】---【Port】--select the port that the serial number displayed by the device manager just now.for example:COM6,as shown in the following figure.







3. After the selection is completed, you need to click " $\rightarrow$ "under the menu bar,and upload the code to the Arduino UNO board, when appears to Done uploading on the lower left corner, that means that the code has been successfully uploaded to the Arduino UNO board, as shown in the following figure.



4. You can open the serial port monitor on the top right corner of Arduino IDE, A serial port of Arduino port will appear, and the baud rate is set to 9600 on the lower right corner, as shown in the following figure.





5.After the code is uploaded, we open the serial port monitor of Arduino IDE, you can see the words "servo=o\_seral\_simple ready" written in the program. And then input a number between 1 ~ 9 randomly in the send box, servo will turn the corresponding angle. Moreover, the serial port monitor will print out the corresponding angle, a comment in the program: "UART send 1~9= >20~180 degree" as shown in the figure below (for example only).





## **19-IR control**

#### The purpose of the experiment:

In this experiment, we will make the IR remote controller communicate with the IR receiver sensor.

#### About the infrared remote control :

The signal from the IR remote controller is a series of binary pulse codes. In order to protect it from other infrared signals during wireless transmission. It is modulated on a specific carrier frequency ,and then transmitted by infrared emission sensor. The infrared receiving device need to filter out other waveform and receive the signal of the specific frequency and restore it to binary pulse code, this process is called demodulation.

The IR receiver sensor converts the optical signal emitted by the infrared emission sensor to a weak electrical signal. These signals are restored to the original encode by various circuits, finally outputs the signal to the control circuit.





#### List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

IR receiver sensor \*1

IR remote controller \*1

Breadboard \*1

Dupont line \*1 bunch

#### Actual object connection diagram :

We need to connect the circuit as shown in the figure below.

			IR rece	iver sensor
		to sold its sold and the sold sold sold its sold		area ounder
	007			
			*******	
	and the second	A CARGE CONTRACTOR		
		TTT		
		TALL AND ADDRESS OF T	anna B	
	<u>.</u>		ACC 1	*************
			Concernation of the	************
	CHD			
	GND	and a second second		Dupont line
	* ***** *****		**** ***	
		AND ADDRESS		** *****
		Ц.,		/
Breadboar			ONL	
Breadboar rduino UNO board				
Breadboar rduino UNO board				
Breadboar duino UNO board			<u>іі іііі</u> _ Оли	
Breadboar			ONU ONU	
Breadboar			ONL ONL	
Breadboar				
Breadboar				

#### Experimental code analysis:

#include <IRremote.h>//Including infrared library int RECV\_PIN = 11; // Declarations of port int LED1 = 2; int LED2 = 3: int LED3 = 4; int LED4 = 5: int LED5 = 6; int LED6 = 7: long on 1 = 0x00FF6897;//Code the example to match the send long off1 = 0x00ff30CF; long on  $2 = 0 \times 00 \text{FF} 9867$ ; long off2 = 0x00FF18E7; long on 3 = 0x00FFB04F; long off3 = 0x00FF7A85; long on  $4 = 0 \times 00 \text{FF10EF}$ ; long off4 = 0x00FF42BD; long on  $5 = 0 \times 00 \text{FF} \times 38 \text{C7}$ ; long off5 = 0x00FF4AB5; long on  $6 = 0 \times 00 \text{FF5AA5}$ ; long off6 = 0x00FF52AD; IRrecv irrecv(RECV\_PIN); decode\_results results;//Declarations of struct // Dumps out the decode results structure. // Call this after IRrecv::decode() // void \* to work around compiler issue //void dump(void \*v) { // decode\_results \*results = (decode\_results \*)v void dump(decode\_results \*results) { int count = results->rawlen; if (results->decode\_type == UNKNOWN) {



```
Serial.println("Could not decode message");
}
else
if (results->decode_type == NEC)
Serial.print("Decoded NEC: ");
else if (results->decode_type == SONY)
Serial.print("Decoded SONY: ");
else if (results->decode_type == RC5)
Serial.print("Decoded RC5: ");
else if (results->decode_type == RC6)
Serial.print("Decoded RC6: ");
Serial.print(results->value, HEX);
Serial.print(" (");
Serial.print(results->bits, DEC);
Serial.println(" bits)");
Serial.print("Raw (");
Serial.print(count, DEC);
Serial.print("): ");
for (int i = 0; i < \text{count}; i++)
if ((i \% 2) == 1)
Serial.print(results->rawbuf[i]*USECPERTICK, DEC);
}
else
Serial.print(-(int)results->rawbuf[i]*USECPERTICK, DEC);
Serial.print(" ");
Serial.println("");
}
void setup()
pinMode(RECV_PIN, INPUT); //Defining the RECV port for the input port
pinMode(LED1, OUTPUT);//Defining the LED1 port for the output port
pinMode(LED2, OUTPUT);//Defining the LED2 port for the output port
pinMode(LED3, OUTPUT);//Defining the LED3 port for the output port
pinMode(LED4, OUTPUT);//Defining the LED4 port for the output port
pinMode(LED5, OUTPUT);//Defining the LED5 port for the output port
pinMode(LED6, OUTPUT);//Defining the LED6 port for the output port
pinMode(13, OUTPUT);//Defining the port13 for the output port
```



```
Serial.begin(9600); //The baud rate is 9600
irrecv.enableIRIn(); // Start the receiver
}
int on = 0;
unsigned long last = millis();
void loop()
{
 if (irrecv.decode(&results)) //Calling the library function: decode
  {
  // If it's been at least 1/4 second since the last
  // IR received, toggle the relay
  if (millis() - last > 250)
    {
    on = !on;
    digitalWrite(13, on ? HIGH : LOW);
    dump(&results);
    }
  if (results.value == on1)
    digitalWrite(LED1, HIGH);
  if (results.value == off1)
    digitalWrite(LED1, LOW);
  if (results.value == on2)
    digitalWrite(LED2, HIGH);
  if (results.value == off2)
    digitalWrite(LED2, LOW);
  if (results.value == on3)
    digitalWrite(LED3, HIGH);
  if (results.value == off3)
    digitalWrite(LED3, LOW);
  if (results.value == on4)
    digitalWrite(LED4, HIGH);
  if (results.value == off4)
    digitalWrite(LED4, LOW);
  if (results.value == on5)
    digitalWrite(LED5, HIGH);
  if (results.value == off5)
    digitalWrite(LED5, LOW);
  if (results, value == on6)
    digitalWrite(LED6, HIGH);
  if (results.value == off6)
    digitalWrite(LED6, LOW);
  last = millis();
  irrecv.resume(); // Receive the next value
 }
```

#### **Experimental steps:**

1.You need to open the code for this experiment: code-IR\_control.ino, click " $\sqrt{}$ "under the menu bar, compile the code, and wait for the words of Done compiling in the lower left corner, as shown in the following figure.





2. In the menu bar of Arduino IDE, select the 【Tools】---【Port】--- select the port that the serial number displayed by the device manager just now.for example:COM6,as shown in the following figure.





3. After the selection is completed, you need to click " $\rightarrow$ "under the menu bar,and upload the code to the Arduino UNO board, when appears to Done uploading on the lower left corner, that means that the code has been successfully uploaded to the Arduino UNO board, as shown in the following figure.



4. After the code is uploaded, we need to open the serial monitor of Arduino IDE, and set the baud rate to 9600. When we press the button on the infrared remote controller, we can see the code value of the corresponding button on the serial monitor, as shown below (Just for example).







#### Steps to add a library file

Note:Before you compile the code, you must look at this steps.

1.We need to add IRremote file, as shown in the figure below.

名称	修改日期	类型	大小
code-IR control	2018/7/9 17:15	文件夹	
📕 IRremote	2018/7/13 17:56	文件夹	
19.IR control.docx	2018/7/13 17:56	DOCX 文档	1,306 KB
ReadMe.docx	2018/7/13 16:25	DOCX 文档	1,826 KB

2.You need to find the installation path of Arduino. As shown in the figure below.( just for example)

This is my Arduino installation path.

##00	тва	າ ສະສະດະກ	di.						
Electron v	117	/ ++++ac(ri) T + = ====+++++++++++++++++++++++++++++							
2044-10		名称 名称	修改日期	英型	大小				
		📕 drivers	2018/7/24 10:50	文件夹					
		📕 examples	2018/7/24 10:50	文件夹					
0 <b>9</b>		🍌 hardware	2018/7/24 10:51	文件夹					
1000		📙 java	2018/7/24 10:51	文件夹					
		🔒 līb	2018/7/24 10:51	文件夹					
		🎍 libraries	2018/8/9 9:52	文件夹					
		🗼 reference	2018/7/24 10:51	文件夹					
		👃 tools	2018/7/24 10:51	文件夹					
	II.	<b>II</b> .		ш.	11	💿 arduino.exe	2015/11/4 0:44	应用程序	850 KB
								📄 arduino.l4j.ini	2015/11/4 0:44
		🔝 Arduino	2018/7/24 10:52	Internet 快捷方式	1 KB				
		💿 arduino_debug.exe	2015/11/4 0:44	应用程序	389 KB				
		📄 arduino_debug.l4j.ini	2015/11/4 0:44	Configuration Se	1 KB				
		🚯 libusb0.dll	2015/11/3 22:20	应用程序扩展	43 KB				
1		🗟 msvcp100.dll	2015/11/3 22:21	应用程序扩展	412 KB				
		🚳 msvcr100.dll	2015/11/3 22:21	应用程序扩展	753 KB				
	10	an revisions.txt	2015/11/3 22:20	文本文档	63 KB				
		💿 uninst.exe	2018/7/24 10:52	应用程序	394 KB				

You need to copy this file into the libraries folder in the Ardunio installation path.
 As shown in the figure below.





4.You need to open Arduino IDE and click [Sketch] --- [Import library] --- [Add library] . As shown in the figure below.



5.You need to add 【IRremote】 to here. As shown in the figure below.



6.After the addition is completed, the words "Library added to your libraries." will appear in the lower right corner of the Arduino IDE. As shown in the figure below.



7. You can see these library files on the Arduino IDE. As shown in the figure below.



8.After completing the above steps, you can compile and upload this code successfully .



# **20-1602 display**

#### The purpose of the experiment:

In this experiment, we use Arduino UNO to directly drive 1602 display letters.

Introduction of 1602 :

The actual object is shown below.



#### Main specification of 1602LCD:

Display capacity: 16 x 2 characters; Working current: 2.0mA Operating voltage: 5.0v Size of character: 2.95 \* 4.35 (W \* H) mm. 1602 possess 16 pins: Pin 1: VSS is ground power Pin 2: VDD is connected to 5V positive power supply Pin 3: V0 is the LCD contrast adjustment pin, which can be adjusted by a 10K adjustable resistor.

Pin 4: RS is the register selection pin, data register is selected at high voltage and instruction register is selected at low voltage.

Pin 5: R/W is the signal line for reading and writing. Reading operation is carried out at high level and writing operation is carried out at low level.

Pin 6: E pin is the enable pin. When this pin changes from high level to low level, the LC D module executes the command.

Pin 7 ~ Pin 14: D0 ~ D7 is 8-bit two-way data line.

Pin 15: power positive pole of backlight.

Pin 16: power negative pole of backlight.

List of components required for the experiment:

Arduino UNO board \*1

USB cable \*1

1602 \*1

Dupont line \*1 bunch

Breadboard \*1

#### Actual object connection diagram :

We need to connect the circuit as shown in the figure below.



	-

#### Experimental code analysis:

```
#include <LiquidCrystal.h>
//Declaring the Arduino digital port connected to the 1602 LCD pin,
//8-wire or 4-wire data mode, either one
LiquidCrystal lcd(12,11,10,9,8,7,6,5,4,3,2);
//LiquidCrystal lcd(12,11,10,5,4,3,2);
int i;
void setup()
{
 lcd.begin(16,2);
                     //Initialization of 1602
                    //The 1602 LCD display range is defined as 2 lines and 16 columns
characters
 while(1)
 {
  lcd.home();
                    //Moving the cursor back to the upper left corner,output from the
beginning
  lcd.print("Hello World");
  Icd.setCursor(0,1); //The cursor is positioned on line 1, column 0
  lcd.print("Welcome to Yahboom-Arduino");
  delay(500);
  for(i=0;i<3;i++)
  {
    lcd.noDisplay();
    delay(500);
    lcd.display();
    delay(500);
  }
  for(i=0;i<24;i++)
  {
    lcd.scrollDisplayLeft();
    delay(500);
   }
  lcd.clear();
```

```
lcd.setCursor(0,0); //Moving the cursor back to the upper left corner,output from the
beginning
    lcd.print("Hi,");
    lcd.setCursor(0,1); //The cursor is positioned on line 1, column 0
    lcd.print("Arduino is fun");
    delay(2000);
  }
}
void loop()
{}//Initialization is complete and the main loop is not need to do anythings
```

YAF

#### Experimental steps:

1.We need to open the code for this experiment: code-1602\_display.ino, click " $\sqrt{}$ "under the menu bar,compile the code, and wait for the words of Done compiling in the lower left corner, as shown in the following figure.



2. In the menu bar of Arduino IDE, you need to select the **[**Tools**]** --- **[**Port**]** --- select the port that the serial number displayed by the device manager just now.for example:COM6,as shown in the following figure.





code-1602 dist	Auto Format Archive Sketch	Ctrl+T		Э.
lcd.	Fix Encoding & Reload Serial Monitor	Ctrl+Shift+M		×
}	Port		Serial ports	-
lcd. cl lcd. sec	Burn Bootloader	THUTTE	✓ COM6 COM15	
lcd.prin lcd.set@	rt ("Hi, "); ursor(0, 1); //	The curse	COM16 COM17 COM18	Contraction of the
lot.ortg	t("Arduino is f		COM19 COM20	
torage space	e Maximum is 3	2 256 byrt	COM21 COM22	

3. After the selection is completed, you need to click " $\rightarrow$ " under the menu bar, and upload the program to the Arduino UNO board, when appears to Done uploading on the lower left corner, that means that the code has been successfully uploaded to the Arduino UNO board, as shown in the following figure.



4.After the code is uploaded. First, the 1602 screen will display "Hello World, Welcome to yahboom-arduino" and flash three times. Then, "Hello World, Welcome to yahboom-arduino," is displayed from the right to the left. Next, "Hi, Arduino is fun." is displayed on the 1602. Finally, itclear the screen, and continue the endless cycle.

As shown in the figure below.

