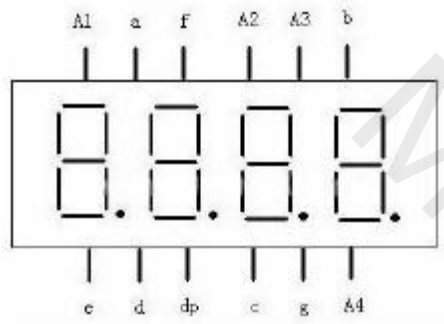
4 digit LED Segment Displays

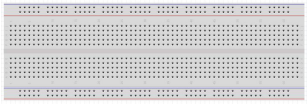
★ Overview

This experiment is similar to the LED experiment, the same is the control of LED, but the experiment can achieve time counting function.

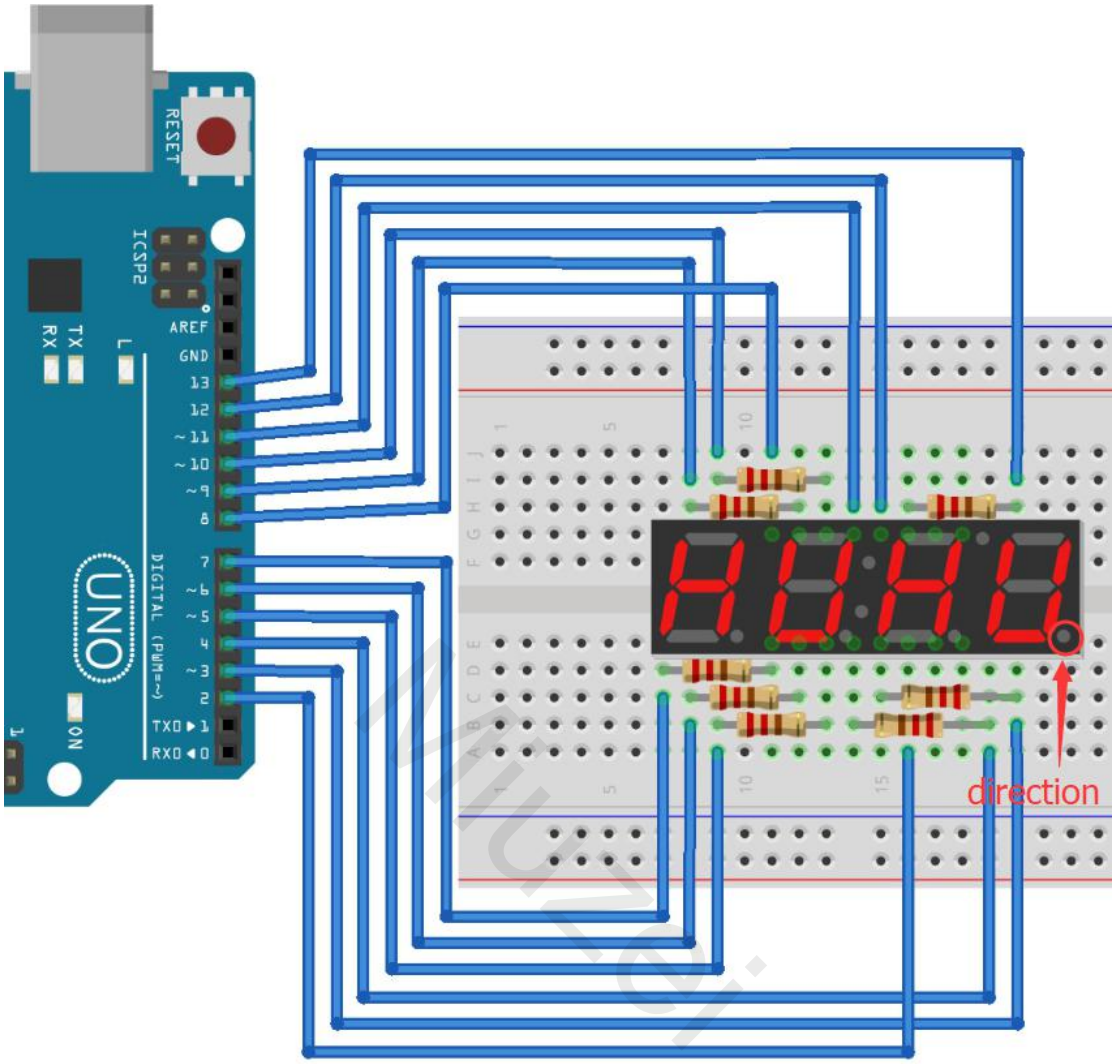
★ Pin definition



★ Hardware required

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

★ Connection diagram



Note: Pay attention to the direction of digital tube.

★ Sample code

Note: sample code under the **Sample code** folder

```
int ASeg = 9;
int BSeg = 13;
int CSeg = 4;
int DSeg = 6;
int ESeg = 7;
int FSeg = 10;
int GSeg = 3;

int a1 = 8;
int a2 = 11;
int a3 = 12;
int a4 = 2;
// set variable
long n = 0;
int x = 100;
int del = 54; // fine adjustment for clock

void setup()
{
	pinMode(a1, OUTPUT);
	pinMode(a2, OUTPUT);
	pinMode(a3, OUTPUT);
	pinMode(a4, OUTPUT);
	pinMode(ASeg, OUTPUT);
	pinMode(BSeg, OUTPUT);
	pinMode(CSeg, OUTPUT);
	pinMode(DSeg, OUTPUT);
	pinMode(ESeg, OUTPUT);
	pinMode(FSeg, OUTPUT);
	pinMode(GSeg, OUTPUT);
}

void loop()
{
	clearLEDS(); // Eliminating ghost
	pickDigit(1);
	pickNumber((n/x/1000)%10); // Display number
	delayMicroseconds(del);

	clearLEDS();
	pickDigit(2);
	pickNumber((n/x/100)%10);
	delayMicroseconds(del);
```

```
    clearLEDS();
    pickDigit(3);
    pickNumber((n/x/10)%10);
    delayMicroseconds(del);

    clearLEDS();
    pickDigit(4);
    pickNumber(n/x%10);
    delayMicroseconds(del);
    n++;
}
// Select display position
void pickDigit(int x)
{
    digitalWrite(a1, LOW);
    digitalWrite(a2, LOW);
    digitalWrite(a3, LOW);
    digitalWrite(a4, LOW);

    switch(x)
    {
        case 1:
        digitalWrite(a1, HIGH);
        break;
        case 2:
        digitalWrite(a2, HIGH);
        break;
        case 3:
        digitalWrite(a3, HIGH);
        break;
        case 4:
        digitalWrite(a4, HIGH);
        break;
    }
}
// select display number
void pickNumber(int x)
{
    switch(x) {
    case 1: one(); break;
    case 2: two(); break;
    case 3: three(); break;
    case 4: four(); break;
    case 5: five(); break;
    case 6: six(); break;
    case 7: seven(); break;
```

```
        case 8: eight(); break;
        case 9: nine(); break;
        default: zero(); break;
        }
}

void clearLEDS()
{
        digitalWrite(ASeg, HIGH);
        digitalWrite(BSeg, HIGH);
        digitalWrite(CSeg, HIGH);
        digitalWrite(DSeg, HIGH);
        digitalWrite(ESeg, HIGH);
        digitalWrite(FSeg, HIGH);
        digitalWrite(GSeg, HIGH);
}

//Display function  '0-9'
void zero() {
        digitalWrite(ASeg, LOW);
        digitalWrite(BSeg, LOW);
        digitalWrite(CSeg, LOW);
        digitalWrite(DSeg, LOW);
        digitalWrite(ESeg, LOW);
        digitalWrite(FSeg, LOW);
        digitalWrite(GSeg, HIGH);
}

void one() {
        digitalWrite(ASeg, HIGH);
        digitalWrite(BSeg, LOW);
        digitalWrite(CSeg, LOW);
        digitalWrite(DSeg, HIGH);
        digitalWrite(ESeg, HIGH);
        digitalWrite(FSeg, HIGH);
        digitalWrite(GSeg, HIGH);
}

void two() {
        digitalWrite(ASeg, LOW);
        digitalWrite(BSeg, LOW);
        digitalWrite(CSeg, HIGH);
        digitalWrite(DSeg, LOW);
        digitalWrite(ESeg, LOW);
        digitalWrite(FSeg, HIGH);
        digitalWrite(GSeg, LOW);
```

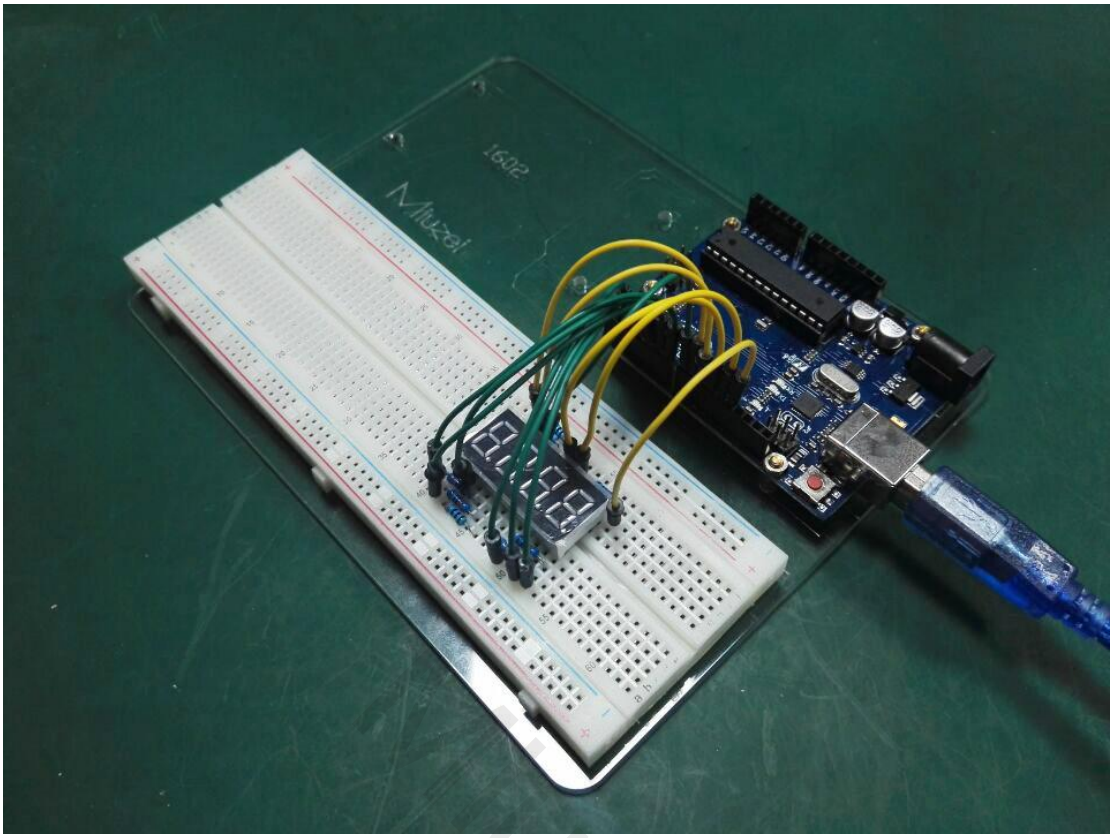
```
}
void three() {
	digitalWrite(ASeg, LOW);
	digitalWrite(BSeg, LOW);
	digitalWrite(CSeg, LOW);
	digitalWrite(DSeg, LOW);
	digitalWrite(ESeg, HIGH);
	digitalWrite(FSeg, HIGH);
	digitalWrite(GSeg, LOW);
}
void four() {
	digitalWrite(ASeg, HIGH);
	digitalWrite(BSeg, LOW);
	digitalWrite(CSeg, LOW);
	digitalWrite(DSeg, HIGH);
	digitalWrite(ESeg, HIGH);
	digitalWrite(FSeg, LOW);
	digitalWrite(GSeg, LOW);
}
void five() {
	digitalWrite(ASeg, LOW);
	digitalWrite(BSeg, HIGH);
	digitalWrite(CSeg, LOW);
	digitalWrite(DSeg, LOW);
	digitalWrite(ESeg, HIGH);
	digitalWrite(FSeg, LOW);
	digitalWrite(GSeg, LOW);
}
void six() {
	digitalWrite(ASeg, LOW);
	digitalWrite(BSeg, HIGH);
	digitalWrite(CSeg, LOW);
	digitalWrite(DSeg, LOW);
	digitalWrite(ESeg, LOW);
	digitalWrite(FSeg, LOW);
	digitalWrite(GSeg, LOW);
}
void seven() {
	digitalWrite(ASeg, LOW);
	digitalWrite(BSeg, LOW);
	digitalWrite(CSeg, LOW);
	digitalWrite(DSeg, HIGH);
	digitalWrite(ESeg, HIGH);
```

```
    digitalWrite(FSeg, HIGH);
    digitalWrite(GSeg, HIGH);
}

void eight() {
    digitalWrite(ASeg, LOW);
    digitalWrite(BSeg, LOW);
    digitalWrite(CSeg, LOW);
    digitalWrite(DSeg, LOW);
    digitalWrite(ESeg, LOW);
    digitalWrite(FSeg, LOW);
    digitalWrite(GSeg, LOW);
}

void nine() {
    digitalWrite(ASeg, LOW);
    digitalWrite(BSeg, LOW);
    digitalWrite(CSeg, LOW);
    digitalWrite(DSeg, LOW);
    digitalWrite(ESeg, HIGH);
    digitalWrite(FSeg, LOW);
    digitalWrite(GSeg, LOW);
}
```

★ Example picture



★ Language reference

Long
switch()
case

★ Application effect

In order to achieve the time counting function, you will see the number of digital tube display increasingly.

About Miuzei:

Miuzei found in 2011，which is a professional manufacturer and exporter that concerned with open-source hardware research & product development, We have more than hundred engineers devote to developing open source hardware like Arduino, Raspberry pi ,3d printers，robots.

Miuzei committed to make more creative open source products and provide richer knowledge for enthusiasts worldwide.　No matter what your ideas are, we provide various mechanical parts and electronic modules to turn your ideas into success.

Would you like to experience our new release products for Free ? If you are intersted with that you could feel free contact with us by email: support@miuzeipro.com
Or join our facebook:
https://www.facebook.com/miuzeipro
Twitter:
https://twitter.com/miuzei_offical