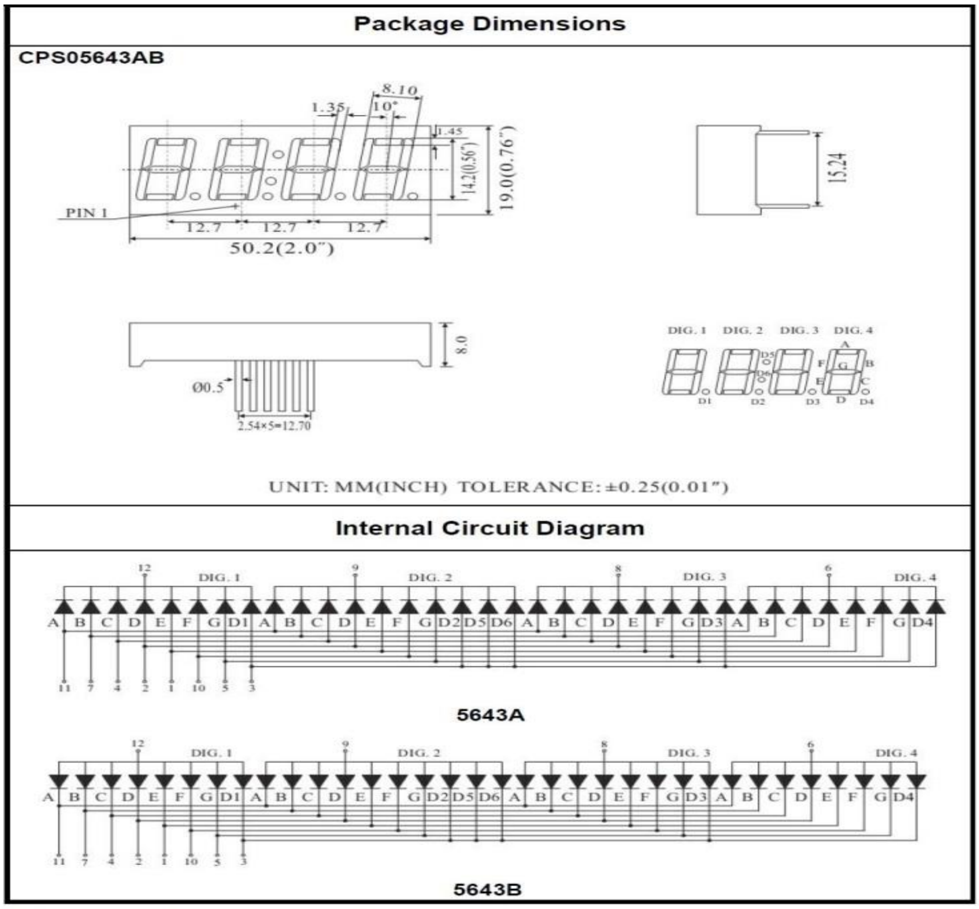
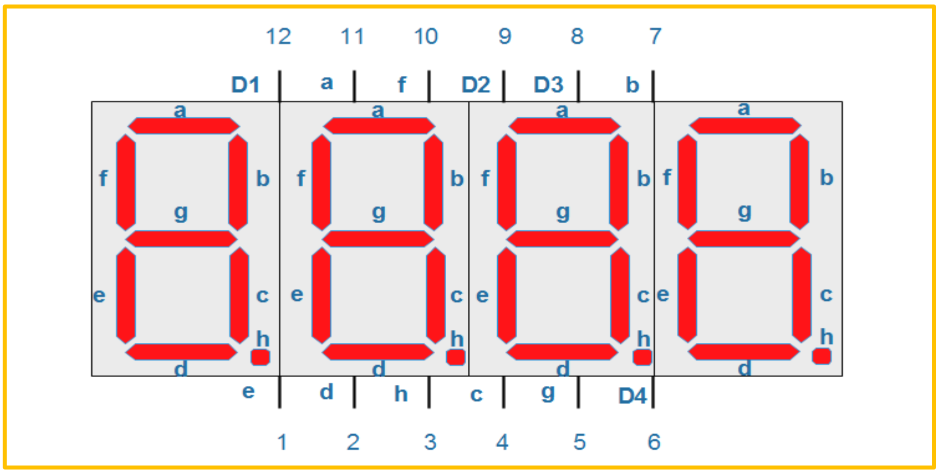
4-Digit 8-Segment Display

Introduction

We used a 8-segment tube before. When we want to display more than one number, then multidigit tube is required. Here we introduce four digital tube, actually each individual 8-segment tube is almost the same as the tube used above. In this experiment, we will use the Arduino to drive a common anode four digital tube.



Four Digits Displays Series



Four digital tube has 12 pins. The upper left is the biggest number 12 pin. Besides the 8-segment we used to display “adbcdefg”, there are another 4 pins D1, D2, D3, D4 to be used as the “bit” pins. When the “bit” pins of common anode four digital tube is high level, the corresponding tubes light up. The display principle of four digital tube is that constantly scanning D1, D2, D3, D4, and then the corresponding eight-segment tubes will light up in turn. Due to the residual effect of human eye, so it looks like the four digital tube display at the same time.

With the principles described above, we can now make a stopwatch. Start timing now.

Experiment Principle

The most important purpose of this program is to display four digital tubes dynamically. In fact, the display of a single digital tube is very easy before the 4-digit display shows the experiment. Since it is an ordinary cathode tube, we first set D1, D2, D3, D4 to high level, all the LEDs are off, and then we output the value of “adbcdefg” to the corresponding gpio port and select the corresponding bit display.

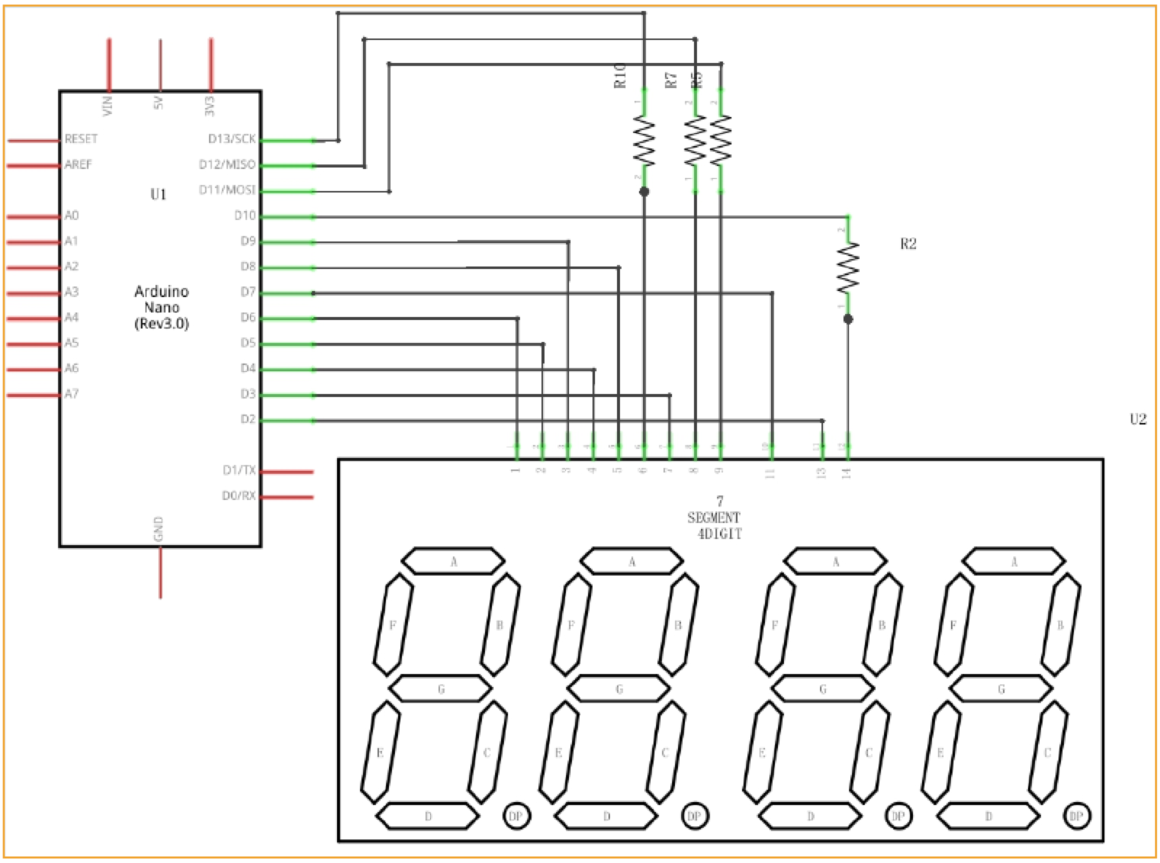
Experiment Purpose

The purpose is to display "1234" bit by bit through a 4-digit 8-segment display.

Component List

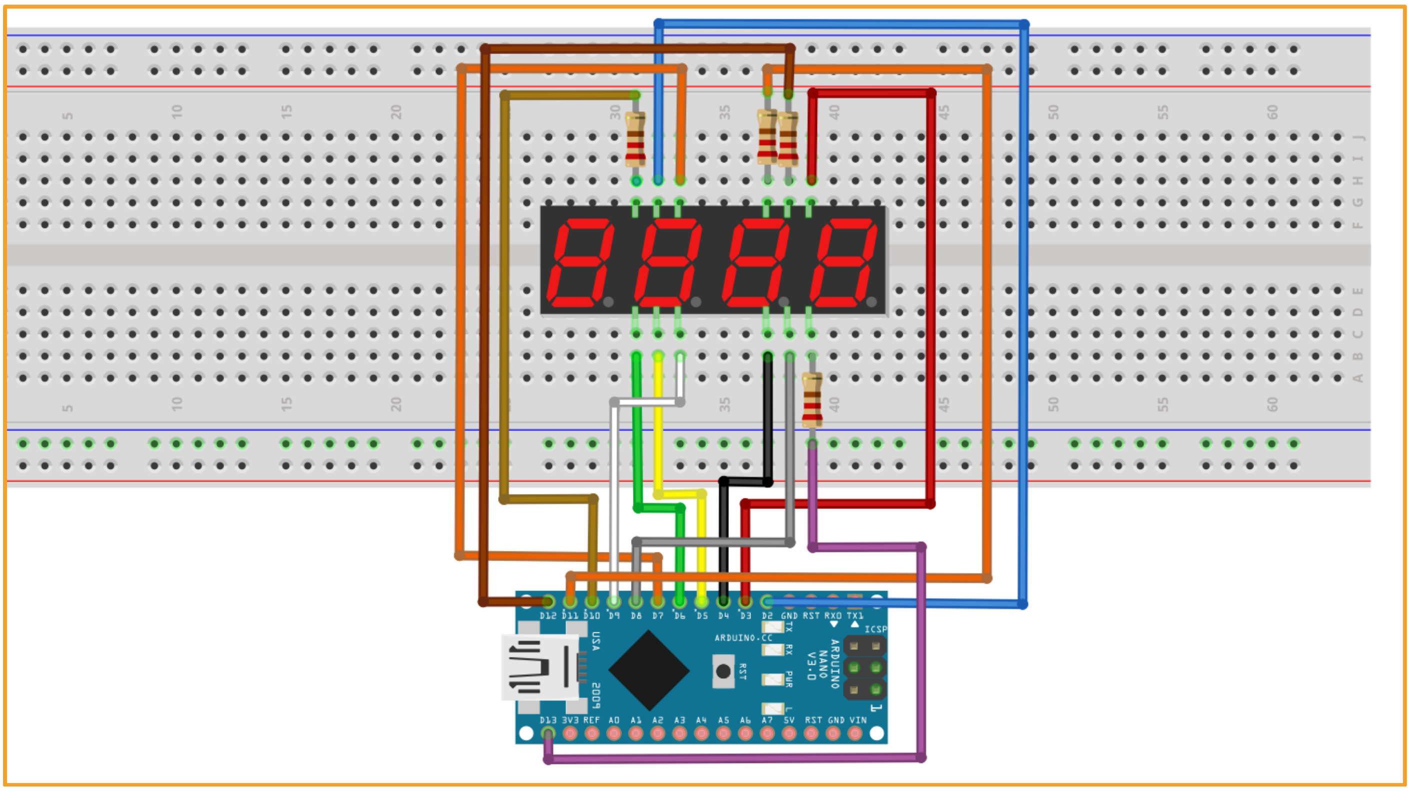
- Arduino Nano Mainboard
- Breadboard
- USB cable
- 4-Digit 8-Segment Display
- 1k Resistor * 4
- Several wires

Schematic Diagram



Wiring of Circuit

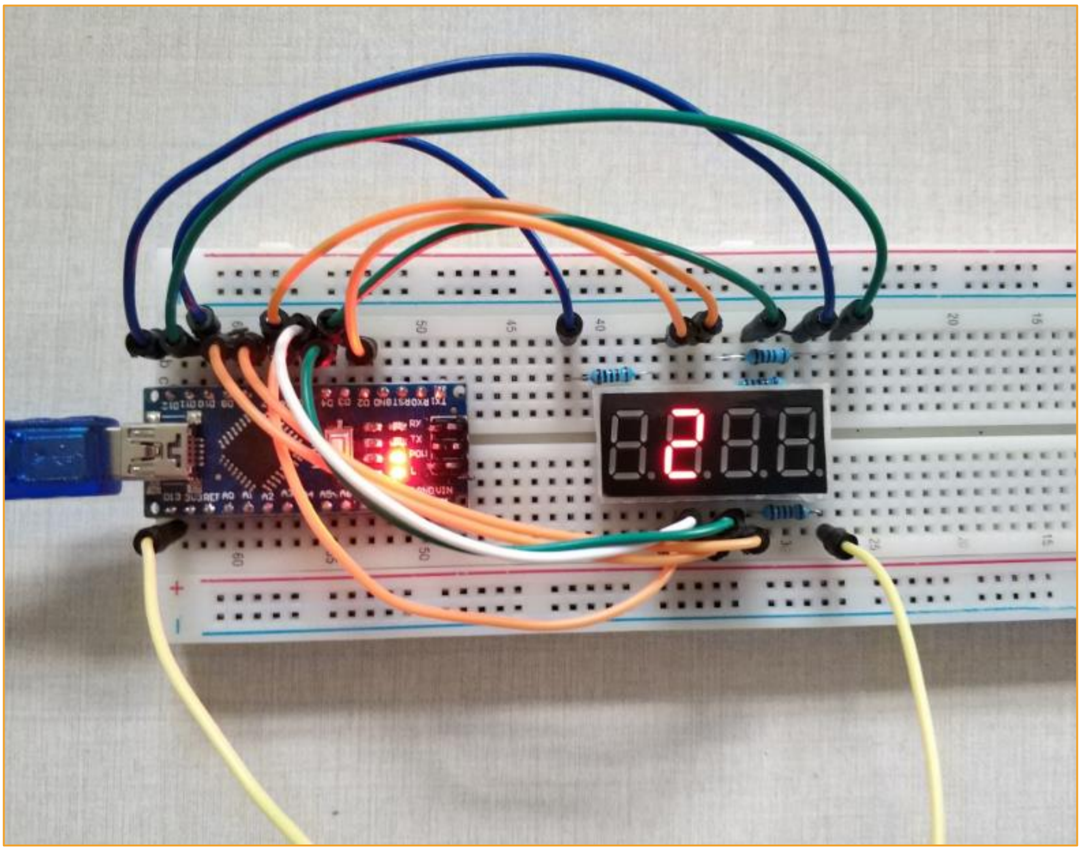
Arduino Nano	4-Digit 8-Segment Display
2	11(a)
3	7(b)
4	4(c)
5	2(d)
6	1(e)
7	10(f)
8	5(g)
9	3(h)
10	12(D1)
11	9(D2)
12	8(D3)
13	6(D4)


D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 GND RST RX0 TX1
ARDUINO.CC
ARDUINO NANO V3.0
ICSP
D13 3V3 REF A0 A1 A2 A3 A4 A5 A6 A7 5V RST GND VIN

Code

```
#define SEG_A 2
#define SEG_B 3
#define SEG_C 4
#define SEG_D 5
#define SEG_E 6
#define SEG_F 7
#define SEG_G 8
#define SEG_H 9
#define COM1 10
#define COM2 11
#define COM3 12
#define COM4 13

unsigned char table[10][8] =
{
   {0,0, 1, 1, 1, 1, 1, 1},         //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);     //Set as output pin
   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);        //The first digit shows 1
   delay(500);
   Display(2,2);        //The second digit shows 2
   delay(500);
   Display(3,3);        //The third digit shows 3
   delay(500);
   Display(4,4);        //The fourth digit shows 4
   delay(500);
}

void Display(unsigned char com,unsigned char num)
{
   digitalWrite(SEG_A,LOW);        //Remove the afterglow
   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_G,LOW);
   digitalWrite(SEG_H,LOW);

   switch(com)               //Strobe selection
   {
      case 1:
         digitalWrite(COM1,LOW);   //Select bit 1
         digitalWrite(COM2,HIGH);
         digitalWrite(COM3,HIGH);
         digitalWrite(COM4,HIGH);
         break;
      case 2:
         digitalWrite(COM1,HIGH);
         digitalWrite(COM2,LOW);   //Select bit 2
         digitalWrite(COM3,HIGH);
         digitalWrite(COM4,HIGH);
         break;
      case 3:
         digitalWrite(COM1,HIGH);
         digitalWrite(COM2,HIGH);
         digitalWrite(COM3,LOW);   //Select bit 3
         digitalWrite(COM4,HIGH);
         break;
      case 4:
         digitalWrite(COM1,HIGH);
         digitalWrite(COM2,HIGH);
         digitalWrite(COM3,HIGH);
         digitalWrite(COM4,LOW);   //Select bit 4
         break;
      default:break;
   }
   digitalWrite(SEG_A,table[num][7]);       //a Search code value table
   digitalWrite(SEG_B,table[num][6]);
   digitalWrite(SEG_C,table[num][5]);
   digitalWrite(SEG_D,table[num][4]);
   digitalWrite(SEG_E,table[num][3]);
   digitalWrite(SEG_F,table[num][2]);
   digitalWrite(SEG_G,table[num][1]);
   digitalWrite(SEG_H,table[num][0]);
}
```

Experiment Result



Notice : The 4 numeric digits are converted into the value of AscII by number2dis, say, we are going to convert “1234”, this should be as follows

Loop	numble	bit_base	disp
1	1234	1000	1
2	234	100	2
3	34	10	3
4	4	1	4