lesson

1

Project Course: Dice

Overview

In this tutorial, we will use the Max7219 module and tilt ball switch to build an electronic dice. In the meanwhile, we will learn how to use the Arduino interrupt function.
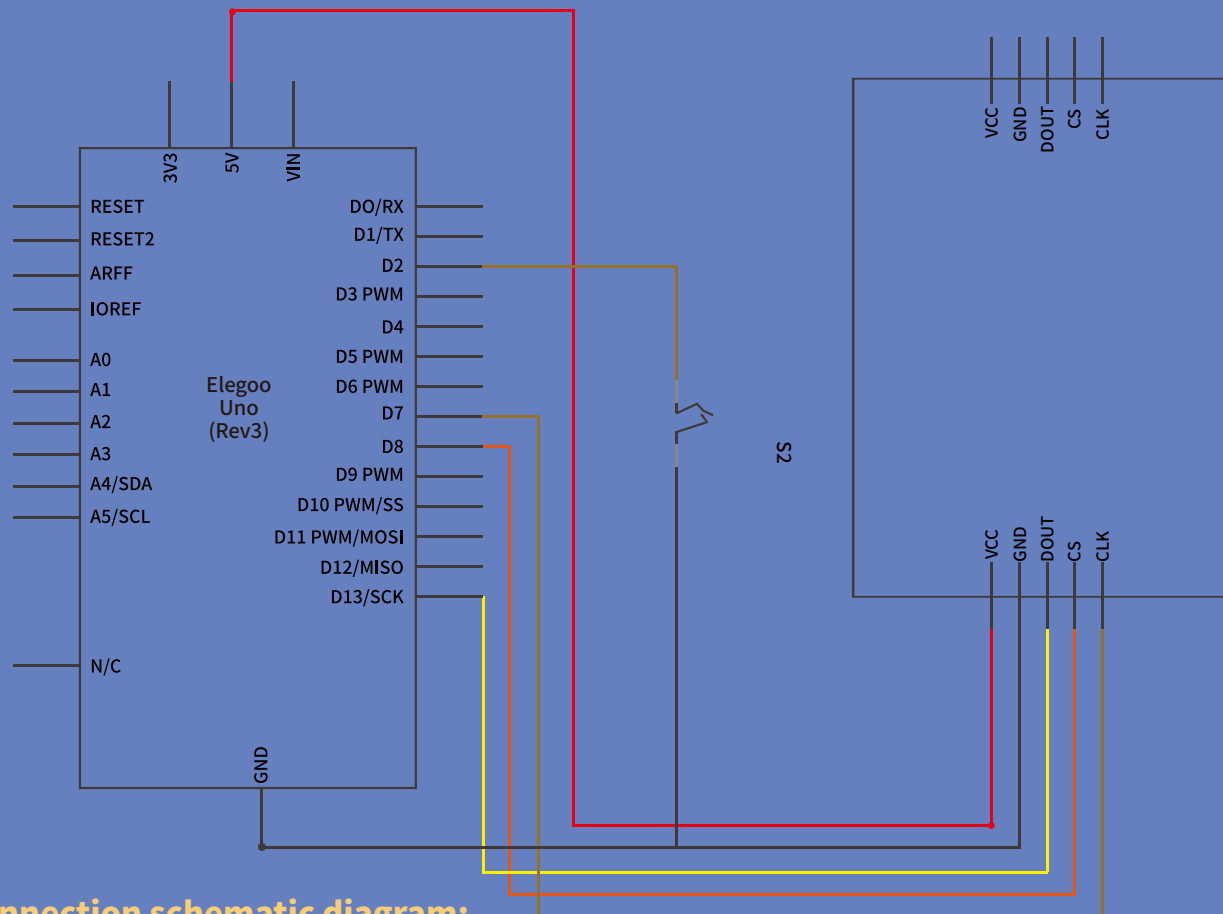
■ Component Required:

(1) X Elegoo UNO R3

(1) X ALL IN ONE Sensor Shield

(1) X MAX7219 module

(1) X Tilt Ball switch

Principle Description:

■ **When** shaking the circuit board, the tilt ball switch changes in the on and off states. This change will trigger an interrupt in the program, causing the MAX7219 module to change the point numbers and then stop at one of the point numbers.

Connection schematic diagram:

Tips:Please insert the extended board into UNO.
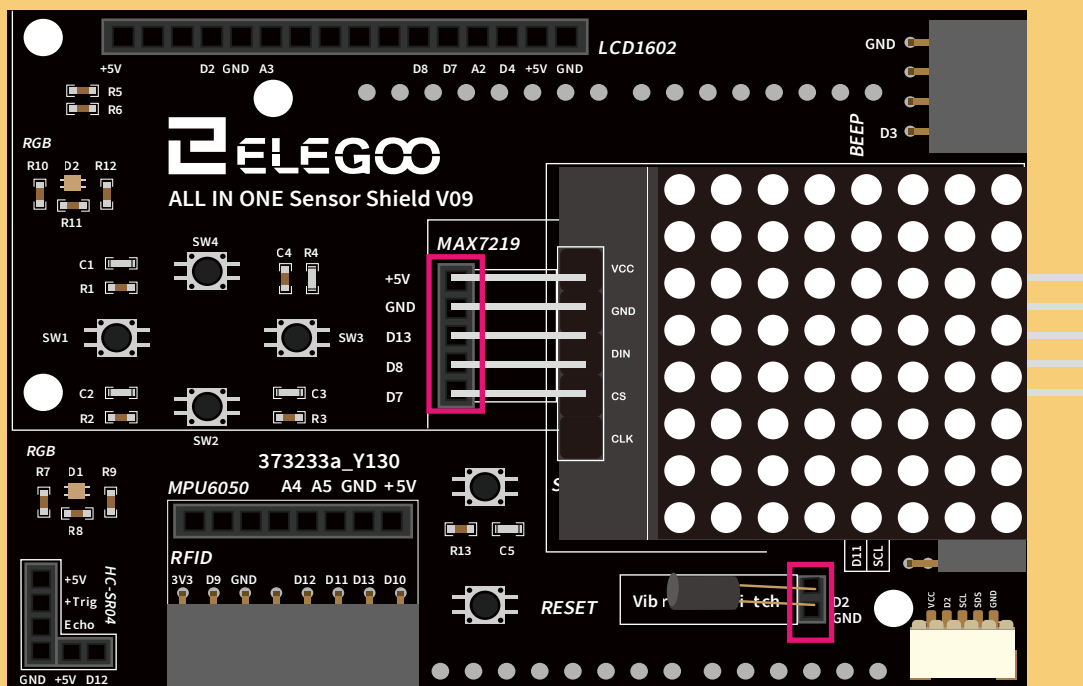
Wiring diagram

Code:

After wiring, please open the program in the code folder- Dice and click UPLOAD to upload the program. See Lesson 5 for details about program uploading if there are any errors.

Before you can run this, please make sure that you have installed the < LedControl> library or re-install it, if necessary. Otherwise, your code won't work. For details about loading the library file, see Lesson 5.

Part1.
Max7219 Display

1. Set the point number to display.
First create an array and write the graphics data of points into the array.
Here is an example of display point 2 on Max7219 module.
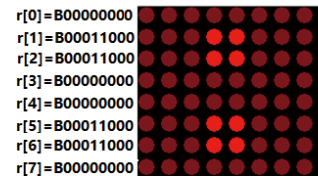

```
byte r[8]={B00000000,B00011000,B00011000,B00000000,B00000000,B00011000,B00011000,
B00000000};
```

Then use the lc.setRow() to display the points on the Max7219 module.

```
lc.setRow(0,0,r[0]);
lc.setRow(0,1,r[1]);
lc.setRow(0,2,r[2]);
lc.setRow(0,3,r[3]);
lc.setRow(0,4,r[4]);
lc.setRow(0,5,r[5]);
lc.setRow(0,6,r[6]);
lc.setRow(0,7,r[7]);
```

For the same reason, we set up different arrays to display different points.

1.We use the random() function to make the point number change randomly.
random(min, max)
min: lower bound of the random value,inclusive(optional).
max: upper bound of the random value, exclusive.

```
randNumber = random(1, 7);
```

random(1, 7) can return us a number from 1-6.

2. Through the numbers obtained in the first step, we can use the switch function to display the corresponding points.

```
Switch (randNumber)
{
    case 1:
      lc.setRow(0,0,a[0]);
      lc.setRow(0,1,a[1]);
      lc.setRow(0,2,a[2]);
      lc.setRow(0,3,a[3]);
      lc.setRow(0,4,a[4]);
  lc.setRow(0,5,a[5]);
      lc.setRow(0,6,a[6]);
      lc.setRow(0,7,a[7]);
  break;
    case 2:
      lc.setRow(0,0,r[0]);
    lc.setRow(0,1,r[1]);
    lc.setRow(0,2,r[2]);
    lc.setRow(0,3,r[3]);
    lc.setRow(0,4,r[4]);
    lc.setRow(0,5,r[5]);
    lc.setRow(0,6,r[6]);
    lc.setRow(0,7,r[7]);
  break;
  case 3: ......

  }
```

Part 3
Points Stop Changing and Start Changing

1.Points stop changing

```
void loop() {
  // put your main code here, to run repeatedly:
  writeRandomNumberOnMatrix();
delay(delaytime1);
  delaytime1=delaytime1+20;//The time for digital changes becomes larger
  /*delaytime1>300 Digital does not change*/
  while(delaytime1>300){
    writeRandomNumberOnMatrix();
    delay(delaytime1);
    Serial.println(delaytime1);
  }
  randNumber = random(1, 7);
}
```

In the loop, each time the loop is performed, the point changes one time and the delaytime1 is increased by 20ms. When the delaytime1 is greater than 300ms, the program will enter the while loop and the point will stop changing.

2.Points start changing
Using interrupt functions

```
attachInterrupt(digitalPinToInterrupt(pin), ISR, mode) (recommended)
SR: the ISR to call when the interrupt occurs; this function must take no parameters and
return nothing.This function is sometimes referred to as an interrupt service routine.
mode: defines when the interrupt should be triggered. Four constants are predefined as
valid values:
  · LOW to trigger the interrupt whenever the pin is low,
  · CHANGE to trigger the interrupt whenever the pin changes value
  · RISING to trigger when the pin goes from low to high,
  · FALLING for when the pin goes from high to low.
```

When shaking circuit board, the tilt ball switch triggers an interrupt, causing delaytime1=100. So the program will jump out of the while loop and change the number again.

```
attachInterrupt(digitalPinToInterrupt(Ball_Switch),blink,RISING);
void blink(){
  delaytime1=100;
  Serial.println(delaytime1);
}
```