MAGIC CUP LIGHT MODULE - HR0025



| Pin Connections | | |
|-----------------|--|--|
| Pin - | 0 VDC | |
| Pin + | 3.3 or 5 VDC | |
| Pin S | Switch | |
| Pin L | LED positive | |
| LED Voltage | 1.9 V | |
| LED Current | 20 mA | |
| | Use current limiting resistor with LED | |
| | Use 160 Ohms for 5 V | |

The KY-027 Magic Light Cup project consists of two HR0025 boards. Each one has an LED and a mercury tilt switch.

Using PWM to drive the LEDs on each module, the effect of light being "magically" transferred from one module to the other when tilting the modules can be achieved. This is similar to pouring water from one cup to the other, hence the name.

These modules are compatible with Arduino, Raspbarry Pi, ESP32 and other popular electronic platforms.





KY-027 SPECIFICATIONS

The KY-027 project consists of two HR0025 modules. Each module is made up of a tilt switch, and LED, a 10 K-Ohm resistor and 4 male pin headers to connect the switch, LED, power and ground.

CONNECTION DIAGRAM

Connect each module using the following diagram.

| KY-027 (A) | Arduino | KY-027 (B) | Arduino |
|------------|---------|------------|---------|
| G | GND | G | GND |
| + | +5V | + | +5V |
| S | 8 | S | 7 |
| L | 9 | L | 6 |



KY-027 ARDUINO CODE

In this Arduino sketch both modules will be used to create the magic light cup effect. The mercury switches in each module will provide a digital signal that is used to regulate the brightness of the LEDs using PWM control.

The modules are places so that the mercury switches on each other are facing in the opposite directions. Tilting the modules will decrease the brightness on one module while increasing it on the other one, creating the illusion of light magically passing from one module to the other.

```
int ledPinA = 9;
     int switchPinA = 8;
    int switchStateA = 0;
    int ledPinB = 6;
    int switchPinB = 7;
    int switchStateB = 0;
    int brightness = 0;
    void setup()
    {
      pinMode(ledPinA, OUTPUT);
10
       pinMode(ledPinB, OUTPUT);
11
12
      pinMode(switchPinA, INPUT);
13
      pinMode(switchPinB, INPUT);
14
     }
15
    void loop()
16
    {
       switchStateA = digitalRead(switchPinA);
17
       if (switchStateA == HIGH && brightness != 255)
18
19
       {
20
       brightness ++;
21
      }
       switchStateB = digitalRead(switchPinB);
22
       if (switchStateB == HIGH && brightness != 0)
23
24
      {
25
       brightness --;
26
       }
       analogWrite(ledPinA, brightness); // A slow fade out
       analogWrite(ledPinB, 255 - brightness); // B slow bright
28
29
      delay(20);
    }
```