

KY-027 Magic light cup module

Revision as of 14:56, 7 April 2017 (view source)

[Sensorkit wiki admin](#) (Talk | contribs)

(→Code example Arduino)

← Older edit

Latest revision as of 15:01, 12 May 2017 (view source)

[Sensorkit wiki admin](#) (Talk | contribs)

(→Code example Raspberry Pi)

(2 intermediate revisions by the same user not shown)

Line 21:

```
==Code example Arduino==
It is an example program wich lights up the LED from Magic=
Light Cup=module if the sensor detects gradient.
```

```

<pre class="brush:cpp">int Led = 13 ;// Declaration of the
LED-output pin
```

```
int Sensor = 10; // Declaration of the sensor input pin
```

Line 79:

```
==Code example Raspberry Pi==
Program example in the language python
```

```

The LED will light up, in this example, if a signal will be
detected.
```

```

<pre class="brush:py"># Needed modules will be imported
and configured.
```

```
import RPi.GPIO as GPIO
```

Line 142:

```
'''Example program download'''
```

```

[[http://sensorkit.joy-it.net/images/c/cb/KY-027-RPi-
MagicLightCup.zip KY-027-RPi-MagicLightCup.zip]]
```

~~– To start with the command:~~

```

<pre class="brush:bash">sudo python KY-027-RPi-
MagicLightCup.py
</pre>
```

Line 21:

```
==Code example Arduino==
```

```
<pre class="brush:cpp">int Led = 13 ;//
Declaration of the LED-output pin
```

```
int Sensor = 10; // Declaration of the
sensor input pin
```

Line 77:

```
==Code example Raspberry Pi==
```

```
<pre class="brush:py"># Needed
modules will be imported and configured.
```

```
import RPi.GPIO as GPIO
```

Line 136:

```
'''Example program download'''
```

```
[[Media:KY-027-RPi-MagicLightCup.zip|KY-
027-RPi-MagicLightCup]]
```

+ To start, enter the command:

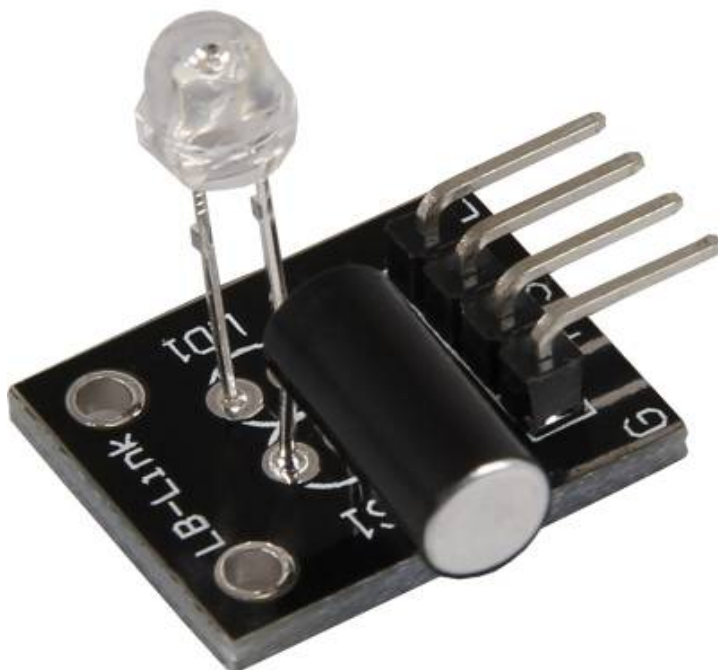
```
<pre class="brush:bash">sudo python
KY-027-RPi-MagicLightCup.py
</pre>
```

Latest revision as of 15:01, 12 May 2017

Contents

1 Picture	2
2 Technical data / Short description	2
3 Pinout	3
4 Code example Arduino	3
5 Code example Raspberry Pi	4

Picture



Technical data / Short description

The LED will be switched on and off by vibration. The signal will be send to the output if the LED is on. You need pre-resistors for some voltages.

Pre-resistor:

Rf (3,3V) [Red]= 120Ω

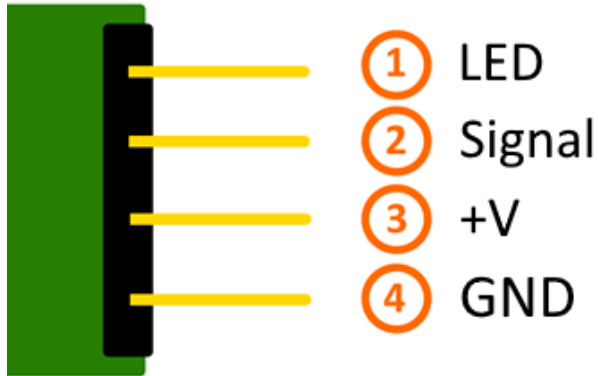
[used with ARM CPU-Core based microcontroller like Raspberry-Pi]

Rf (5V) [Red] = 220Ω

KY-027 Magic light cup module

[used with Atmel Atmega based microcontroller like Arduino]

Pinout



Code example Arduino

```
int Led = 13 ;// Declaration of the LED-output pin
int Sensor = 10; // Declaration of the sensor input pin
int val; // Temporary variable

void setup ()
{
  pinMode (Led, OUTPUT) ; // Initialization output pin
  pinMode (Sensor, INPUT) ; // Initialization sensor pin
  digitalWrite(Sensor, HIGH); // Activating of the internal pull-up resistor
}

void loop ()
{
  val = digitalRead (Sensor) ; // The current signal from the sensor will be read

  if (val == HIGH) // If a signal will be detected, the LED will light up.
  {
    digitalWrite (Led, LOW);
  }
  else
  {
    digitalWrite (Led, HIGH);
  }
}
```

Connections Arduino:

LED +	= [Pin 13]
LED -	= [Pin GND]
Sensor signal	= [Pin 10]
Sensor +V	= [Pin 5V]
Sensor -	= [Pin GND]

KY-027 Magic light cup module

Example program download

[SensorTest_Arduino](#)

Code example Raspberry Pi

```
# Needed modules will be imported and configured.
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

# Declaration of the LED and sensor pins
LED_PIN = 24
Sensor_PIN = 23
GPIO.setup(Sensor_PIN, GPIO.IN)
GPIO.setup(LED_PIN, GPIO.OUT)

print "Sensor-test [press ctrl+c to end the test]"

# This output function will be started at signal detection
def ausgabeFunktion(null):
    GPIO.output(LED_PIN, True)

# This output function will be started at signal detection
GPIO.add_event_detect(Sensor_PIN, GPIO.FALLING, callback=ausgabeFunktion, bouncetime=10)

# main program loop
try:
    while True:
        time.sleep(1)
        # output will be reseted if the switch turn back to the default position.
        if GPIO.input(Sensor_PIN):
            GPIO.output(LED_PIN, False)

# Scavenging work after the program has ended
except KeyboardInterrupt:
    GPIO.cleanup()
```

Connections Raspberry Pi:

LED	=	GPIO24	[Pin 18]
Signal	=	GPIO23	[Pin 16]
+V	=	3,3V	[Pin 1]
GND	=	GND	[Pin 6]

Example program download

[KY-027-RPi-MagicLightCup](#)

To start, enter the command:

```
sudo python KY-027-RPi-MagicLightCup.py
```