

KY-027 Magic light cup module

Revision as of 15:01, 12 May 2017 (view source) **Latest revision as of 15:01, 12 May 2017 (view source)**
[Sensorkit wiki admin \(Talk | contribs\)](#) [Sensorkit wiki admin \(Talk | contribs\)](#)
 (→Code example Arduino) (→Code example Raspberry Pi)
 ← Older edit ← Older edit

Line 77:

```

==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 77:

```

==Code example Raspberry Pi==
<pre class="brush:py"># Needed modules will be
imported and configured.
import RPi.GPIO as GPIO
    
```

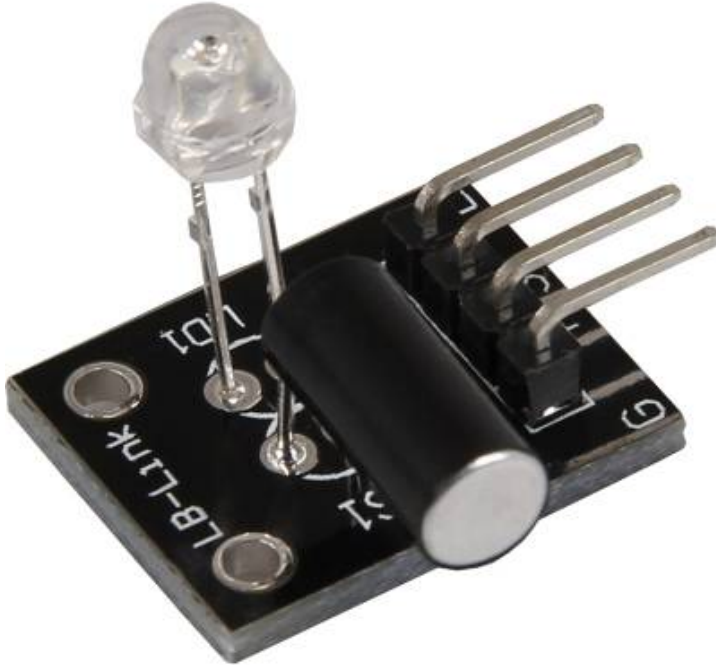
Latest revision as of 15:01, 12 May 2017

Contents

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

KY-027 Magic light cup module

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:

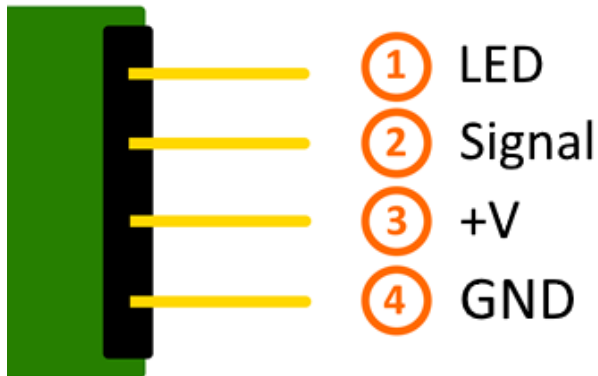
Rf (3,3V) [Red]= 120Ω

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

Rf (5V) [Red] = 220Ω

[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]

Example program download

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
```