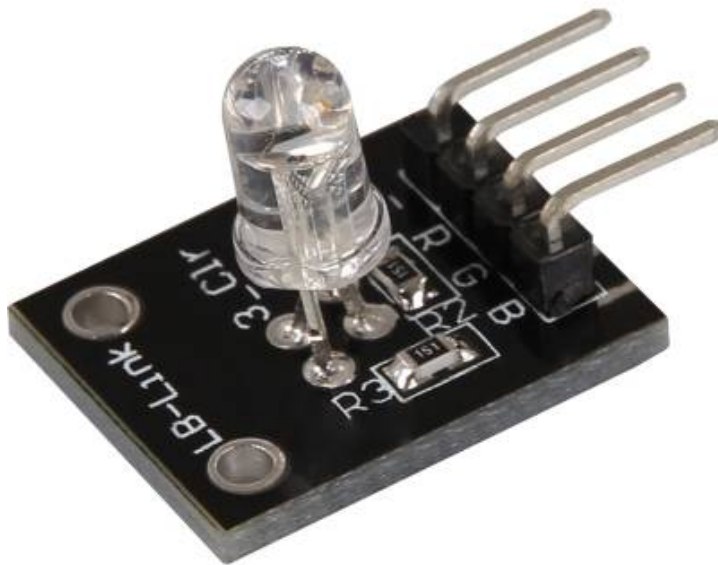


KY-016 Module led RGB 5mm

Sommaire

1 Photo	1
2 Données techniques / Description sommaire	1
3 Brochage	4
4 Exemple de code pour Arduino	4
5 Exemple de code pour Raspberry Pi	6

Photo



Données techniques / Description sommaire

Ce module est constitué d'une LED RGB qui est composée de 3 LEDS de couleurs différentes: rouge - vert - bleu. Celles-ci sont reliées par une cathode commune. En fonction de la tension d'entrée, des résistances en série sont nécessaires.

Vf [Rouge]= 1,8V

Vf [Verte, Bleue]= 2,8V

If= 20mA

KY-016 Module led RGB 5mm

Résistances de limitation:

Rf (3,3V) [Verte]= 100Ω

Rf (3,3V) [Rouge]= 180Ω

Rf (3,3V) [Bleue]= 100Ω

[Valeurs calculées lors de l'utilisation avec des microcontrôleurs ARM de base tel que la Raspberry Pi]

Rf (5V) [Verte] = 100Ω

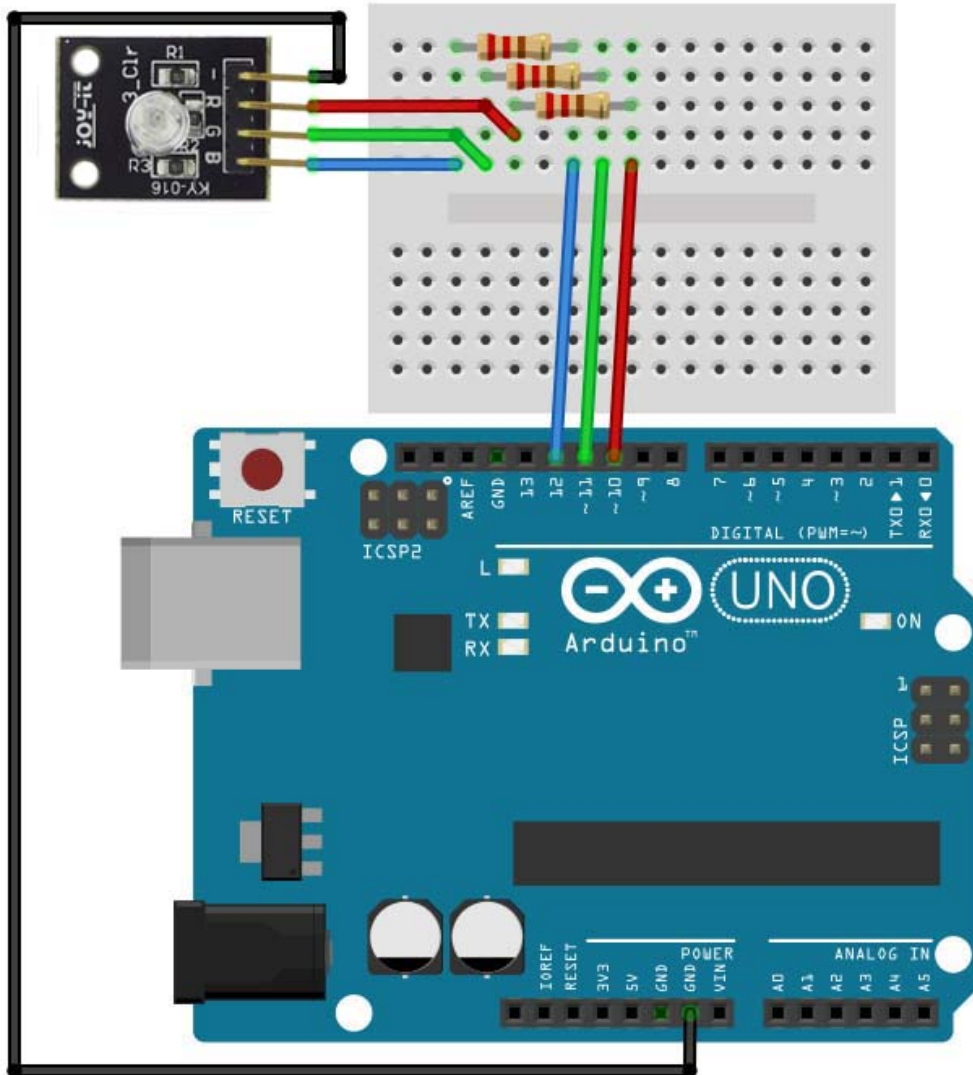
Rf (5V) [Rouge] = 180Ω

Rf (5V) [Bleue] = 100Ω

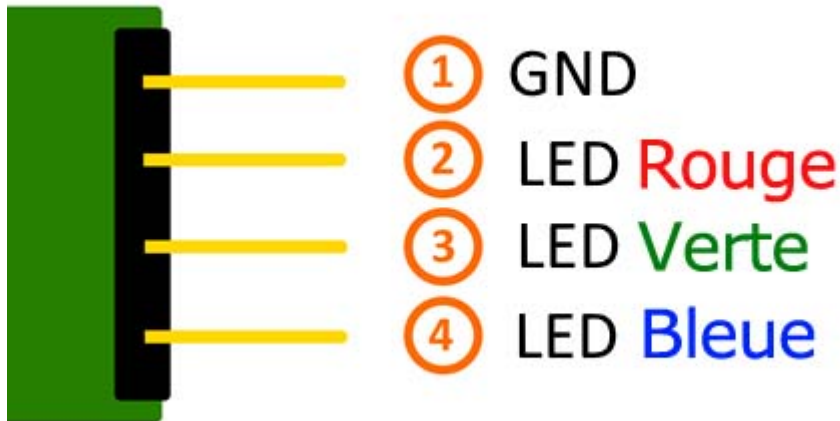
[Valeurs calculées lors de l'utilisation avec des microcontrôleurs Atmel Atmega tel que Arduino]

Exemple d'utilisation avec une carte Arduino (nous avons pris 220Ω dans l'exemple pour utiliser les mêmes résistances pour tous les montages):

KY-016 Module led RGB 5mm



Brochage



Exemple de code pour Arduino

Exemple de code ON/OFF

Cet exemple montre comment les LEDS intégrées sont alternativement allumées puis éteintes toutes les 3 secondes.

```
int Led_Rouge = 10;
int Led_Verte = 11;
int Led_Bleue = 12;

void setup ()
{
  // Initialisation des broches de sortie pour les LEDS
  pinMode (Led_Rouge, OUTPUT);
  pinMode (Led_Verte, OUTPUT);
  pinMode (Led_Bleue, OUTPUT);
}

void loop () //Boucle de programme principale
{
  digitalWrite (Led_Rouge, HIGH); // la Led s'allume
  digitalWrite (Led_Verte, LOW); // la LED commute
  digitalWrite (Led_Bleue, LOW); // la LED commute
  delay (3000); // Délai de 3 secondes

  digitalWrite (Led_Rouge, LOW); // la LED commute
  digitalWrite (Led_Verte, HIGH); // la Led s'allume
  digitalWrite (Led_Bleue, LOW); // la LED commute
  delay (3000); // Délai de 3 secondes supplémentaires pendant lequel les LEDS sont commutés

  digitalWrite (Led_Rouge, LOW); // la LED commute
  digitalWrite (Led_Verte, LOW); // la LED commute
  digitalWrite (Led_Bleue, HIGH); // la Led s'allume
  delay (3000); // Délai de 3 secondes supplémentaires pendant lequel les LEDS sont commutés
}
```

Téléchargement de l'exemple de code ON/OFF:

[KY-016_LED_ON-OFF.zip](#)

Exemple de code en PWM

Un signal PWM (modulation de largeur d'impulsion) permet de faire varier la luminosité d'une LED. Le signal fait varier le temps pendant lequel il est à l'état haut et celui pendant lequel il est à l'état bas, ce qui fait varier la tension moyenne d'alimentation de la LED. La persistance rétinienne de l'oeil fait que nous visualisons cela comme un changement de luminosité.

Plusieurs LEDS sont intégrées dans ce module et la superposition de différents niveaux de luminosité de ces LEDS permet d'obtenir différentes couleurs. Ceci est illustré dans l'exemple de code suivant.

```
int Led_Rouge = 10;
int Led_Verte = 11;
int Led_Bleue = 12;

int val;

void setup () {
  // Initialisation des broches de sortie pour les LEDS
  pinMode (Led_Rouge, OUTPUT);
  pinMode (Led_Verte, OUTPUT);
  pinMode (Led_Bleue, OUTPUT);
}
void loop () {
  // Dans une boucle For, différentes valeurs PWM sont envoyées aux 3 LEDS
  for (val = 255; val > 0; val--)
  {
    analogWrite (Led_Bleue, val);
    analogWrite (Led_Verte, 255-val);
    analogWrite (Led_Rouge, 128-val);
    delay (1);
  }
  // Les signaux sont ensuite inversés
  for (val = 0; val < 255; val++)
  {
    analogWrite (Led_Bleue, val);
    analogWrite (Led_Verte, 255-val);
    analogWrite (Led_Rouge, 128-val);
    delay (1);
  }
}
```

Exemple de programme à télécharger:

[KY-016_PWM.zip](#)

Affectation des broches Arduino:

LED Rouge	= [Pin 10]
LED Verte	= [Pin 11]
LED Bleue	= [Pin 12]
Sensor GND	= [Pin GND]

Exemple de code pour Raspberry Pi

Exemple de code ON/OFF

Cet exemple montre comment les LEDS intégrées sont alternativement allumées puis éteintes toutes les 3 secondes.

```
# coding=utf-8
# Les modules nécessaires sont importés et mis en place
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

# Initialisation des broches de sortie pour les LEDS
LED_ROUGE = 6
LED_VERTE = 5
LED_BLEUE = 4

GPIO.setup(LED_ROUGE, GPIO.OUT, initial= GPIO.LOW)
GPIO.setup(LED_VERTE, GPIO.OUT, initial= GPIO.LOW)
GPIO.setup(LED_BLEUE, GPIO.OUT, initial= GPIO.LOW)

print "LED-Test [Appuyez sur Ctrl + C pour terminer le test]"

# Boucle de programme principale
try:
    while True:
        print("LED ROUGE allumée 3 secondes")
        GPIO.output(LED_ROUGE,GPIO.HIGH) #la Led s'allume
        GPIO.output(LED_VERTE,GPIO.LOW) #la LED commute
        GPIO.output(LED_BLEUE,GPIO.LOW) #la LED commute
        time.sleep(3) #Délai de 3 secondes
        print("LED VERTE allumée 3 secondes")
        GPIO.output(LED_ROUGE,GPIO.LOW) #la LED commute
        GPIO.output(LED_VERTE,GPIO.HIGH) #la Led s'allume
        GPIO.output(LED_BLEUE,GPIO.LOW) #la LED commute
        time.sleep(3) #Délai de 3 secondes
        print("LED BLEUE allumée 3 secondes")
        GPIO.output(LED_ROUGE,GPIO.LOW) #la LED commute
        GPIO.output(LED_VERTE,GPIO.LOW) #la LED commute
        GPIO.output(LED_BLEUE,GPIO.HIGH) #la Led s'allume
        time.sleep(3) #Délai de 3 secondes

#remise en place de tous les GPIO en entrées
except KeyboardInterrupt:
    GPIO.cleanup()
```

Téléchargement de l'exemple de code ON/OFF:

[images/3/3a/KY009_RPi_ON-OFF.zip KY016_RPi_ON-OFF.zip]

Pour débiter avec la commande:

```
sudo python KY016_RPI_ON-OFF.py
```

Exemple de code PWM

KY-016 Module led RGB 5mm

L'utilisation de signaux PWM (modulation de largeur d'amplitude) permet de faire varier la luminosité d'une LED. La LED est alimentée par des impulsions, le ratio entre la durée des impulsions et la durée au repos correspondant à une luminosité relative. En raison de la persistance rétinienne, l'oeil humain assimile ces changements d'alimentation de la LED à une variation de luminosité.

Vous trouverez plus d'informations sur http://www.mon-club-elec.fr/pmwiki_reference_arduino/pmwiki.php?n=Main.ApprendrePWM

Plusieurs LEDS sont intégrées dans ce module et la création de couleurs différentes est produite par la superposition de différents niveaux de luminosité. Ceci est illustré dans l'exemple de code suivant.

```
# coding=utf-8
# Les modules nécessaires sont importés et mis en place
import random, time
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)

# Déclaration des broches de sortie sur lesquelles les LEDS sont raccordées
LED_Rouge = 6
LED_Verte = 5
LED_Bleue = 4

# Configuration des broches en sortie
GPIO.setup(LED_Rouge, GPIO.OUT)
GPIO.setup(LED_Verte, GPIO.OUT)
GPIO.setup(LED_Bleue, GPIO.OUT)

Freq = 100 #Hz

# Les couleurs respectives sont initialisées
ROUGE = GPIO.PWM(LED_Rouge, Freq)
VERTE = GPIO.PWM(LED_Verte, Freq)
BLEUE = GPIO.PWM(LED_Bleue, Freq)
ROUGE.start(0)
VERTE.start(0)
BLEUE.start(0)

# Cette fonction génère la couleur réelle
# L'intensité de la couleur peut être modifiée grâce à la variable de couleur
# Après réglage de la couleur, la durée d'allumage est définie par 'time.sleep'

def LED_Couleur(Rouge, Verte, Bleue, pause):
    ROUGE.ChangeDutyCycle(Rouge)
    VERTE.ChangeDutyCycle(Verte)
    BLEUE.ChangeDutyCycle(Bleue)
    time.sleep(pause)

    ROUGE.ChangeDutyCycle(0)
    VERTE.ChangeDutyCycle(0)
    BLEUE.ChangeDutyCycle(0)

print "LED-Test [Appuyez sur Ctrl + C pour terminer le test]"

# Boucle de programme principale:
# Cette boucle doit faire varier l'intensité de chaque couleur de 0 à 100% en utilisant un
# Les mélanges des différentes luminosités permettent de créer un gradients de couleurs di
try:
    while True:
        for x in range(0,2):
            for y in range(0,2):
                for z in range(0,2):
                    print (x,y,z)
                    for i in range(0,101):
```

KY-016 Module led RGB 5mm

```
LED_Couleur((x*i),(y*i),(z*i),.02)

# remise en place de tous les GPIO en entrées
except KeyboardInterrupt:
    GPIO.cleanup()
```

Exemple de programme à télécharger:

[KY-016_RPi_PWM.zip](#)

Commande pour lancer le programme:

```
sudo python KY-016_RPi_PWM.py
```

Brochage Raspberry Pi:

LED **Rouge** = GPIO6 [Pin 22]
LED **Verte** = GPIO5 [Pin 18]
LED **Bleue** = GPIO4 [Pin 16]
Sensor GND = Masse [Pin 6]