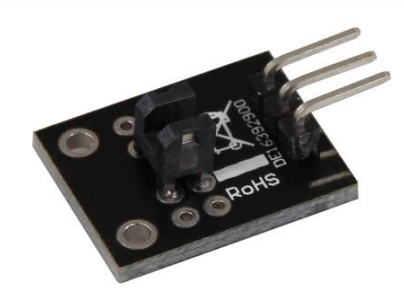
Sensor KY-010



Aplicación típica:

El módulo KY-010 Photo Interrupter consta de un emisor / detector óptico en la parte frontal y dos resistencias (1 kΩ y 33 Ω) en la parte posterior. El sensor usa un haz de luz entre el emisor y un detector para verificar si el camino entre ambos está siendo bloqueado por un objeto opaco.

Se utiliza para generar interrupciones por hardware en el programa en caso de detectar una interrupción física en el sensor.

Estos sensores se usan para medir las revoluciones por minuto de un motor colocando un disco perforado acoplado al motor y colocado entre el emisor y receptor del sensor para contar el número de interrupciones por segundo y así calcular las RPM.

Codigo de muestra

int Led = 13 ;// Declaración del LED de salida

int Sensor = 10; // Declaración de la entrada del sensor

int val; // Varible de apoyo

void setup () {

pinMode (Led, OUTPUT) ; // Inicializa pin LED de salida

pinMode (Sensor, INPUT) ; // inicializa el pin del sensor como entrada

}

void loop () {

val = digitalRead (Sensor) ; // Lee la señal del sensor

if (val == HIGH) //si se detecta señal , el LED encenderá

{

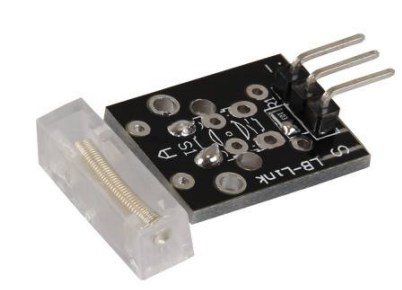
digitalWrite (Led, HIGH); }

else {

digitalWrite (Led, LOW); //sino se apagara

} }

Sensor KY-031 (Sensor de impacto)



Aplicación típica:

El Modulo KY-031 es mejor conocido como Sensor de Impacto, este sensor tiene la capacidad de percibir los impactos que este o una superficie sujeto a este pueda recibir, la información de impacto es transformada por el sensor y enviada a la placa Arduino o a un Microcontrolador.

Puede utilizarse para una gran cantidad de aplicaciones, puede servir como dispositivo de señal de detonación para dispositivos de control como sistemas de seguridad para bebés, objetos móviles, robot de batalla, etc.

Codigo de muestra

int Led = 13 ;// Declara el LED de salida

pin int Sensor = 10; // Declaración de la entrada del sensor

pin int val; // Variable de apoyo

void setup () {

pinMode (Led, OUTPUT) ; // Inicializa pin LED de salida

pinMode (Sensor, INPUT) ; // Inicializa pin del sensor

}

void loop () {

val = digitalRead (Sensor) ; // Lee la señal actual del sensor

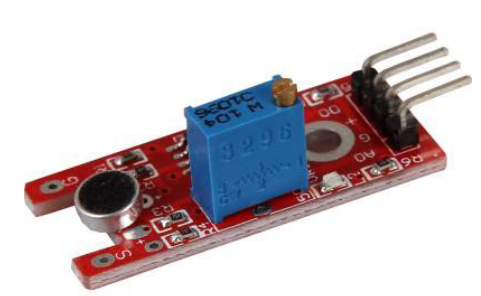
if (val == HIGH) // Si se detecta señal, el LED enciende

{ digitalWrite (Led, LOW); }

else {

digitalWrite (Led, HIGH); }//sino el LED se apagara

Sensor KY-038 (Sensor de sonido)



Aplicación típica:

El sensor KY-038 tiene como característica que es altamente sensible, por lo que su sensor de micrófono de condensador electret (EC) permite detectar un mínimo ruido producido en el ambiente. Cuenta con un LED indicadora de suministro de energía y por el otro extremo cuenta con otra que indica la salida.

Lo interesante de este sensor, es que la señal que nos entrega es digital y analógica, lo cual nos permite decidir cuál utilizar según nuestras necesidades. Si necesitamos saber el valor del sensor, podremos utilizar directamente la salida analógica para conseguir los datos crudos. Sino, podemos utilizar la salida digital, la cual se activa o se desactiva según si el sensor llega a medir la intensidad del sonido que le configuremos, mediante el potenciometro multi vuelta.

Se puede usar como micrófono o sensor de sonidos débiles.

Código de muestra

// Inicializacion de entradas

int Analog\_Eingang = A0; // Señal del eje X

int Digital\_Eingang = 3; // Boton

void setup () {

pinMode (Analog\_Eingang, INPUT);

pinMode (Digital\_Eingang, INPUT);

Serial.begin (9600); // Inicializa el puerto serial a 9600 baudios

}

// El programa lee los valores actuales de las entradas

// y los despliega en la salida serial

void loop () {

float Analog;

int Digital;

// Se convierten los valores de los sensores en voltage

Analog = analogRead (Analog\_Eingang) \* (5.0 / 1023.0);

Digital = digitalRead (Digital\_Eingang);

//y se despliegan aqui

Serial.print ("Analog voltage value:");

Serial.print (Analog, 4);

Serial.print ("V, ");

Serial.print ("Extreme value:");

if(Digital==1) {

Serial.println (" reached");

} else {

Serial.println (" not reached yet");

} Serial.println ("----------------------------------------------------------------");

delay (200); }