

Barriera IR con 2 sensori

Prezzo: 4.10 €

Tasse: 0.90 €

Prezzo totale (con tasse): 5.00 €



Confezione contenente un LED emettitore a infrarossi e un ricevitore. Posti frontalmente, ad una distanza massima di 25 centimetri, consentono di creare una barriera a infrarossi in grado di rilevare l'interruzione del fascio emesso dal trasmettitore. Rispetto ai sensori PIR, l'interruzione del fascio è molto più veloce e permette un migliore controllo. Il ricevitore è un transistor con uscita open collector al quale è possibile collegare un relè, un LED o altro dispositivo che assorba un massimo 100 mA. È possibile utilizzarli anche con Arduino o altro microcontrollore.

Specifiche tecniche

- **Distanza:** circa 25 cm
- **Alimentazione:** da 3,3 a 5,5 Vdc
- **Consumo Trasmettitore:** 10 mA @ 3.3V, 20 mA @ 5V
- **Carico applicabile all'uscita del ricevitore:** 100 mA
- **Angolo del LED Trasmettitore / Ricevitore:** 10°
- **Tempo di risposta:** <2 ms
- **Dimensioni (mm):** 20x10x8
- **Lunghezza del cavo:** 234 mm
- **Peso:** 3 grammi (singolo sensore)

Sketch di esempio per Arduino

```

/*
IR Breakbeam sensor demo!
*/

#define LEDPIN 13
// Pin 13: Arduino has an LED connected on pin 13
// Pin 11: Teensy 2.0 has the LED on pin 11
// Pin 6: Teensy++ 2.0 has the LED on pin 6
// Pin 13: Teensy 3.0 has the LED on pin 13

#define SENSORPIN 4

// variables will change:
int sensorState = 0, lastState=0; // variable for reading the pushbutton status

void setup() {
// initialize the LED pin as an output:
pinMode(LEDPIN, OUTPUT);
// initialize the sensor pin as an input:
pinMode(SENSORPIN, INPUT);
digitalWrite(SENSORPIN, HIGH); // turn on the pullup

Serial.begin(9600);
}

void loop(){
// read the state of the pushbutton value:
sensorState = digitalRead(SENSORPIN);

// check if the sensor beam is broken
// if it is, the sensorState is LOW:
if (sensorState == LOW) {
// turn LED on:
digitalWrite(LEDPIN, HIGH);
}
else {
// turn LED off:
digitalWrite(LEDPIN, LOW);
}

if (sensorState && !lastState) {
Serial.println("Unbroken");
}
if (!sensorState && lastState) {
Serial.println("Broken");
}
lastState = sensorState;
}

```

- [Datasheet](#)
- [Download PDF](#)