

Capteur PIR HC- SR501

Aperçu

Dans cette leçon, vous apprendrez à utiliser un détecteur de mouvement PIR avec un UNO. Le UNO est au cœur de ce projet. Il «écoute» le capteur PIR et, lorsqu'un mouvement est détecté, demande à la LED de s'allumer ou de s'éteindre.

Composant requis:

- 1 x arduino UNO
- 1 x capteur de mouvement PIR HC-SR501
- 3 x fils F-M (fils DuPont femelle à mâle)

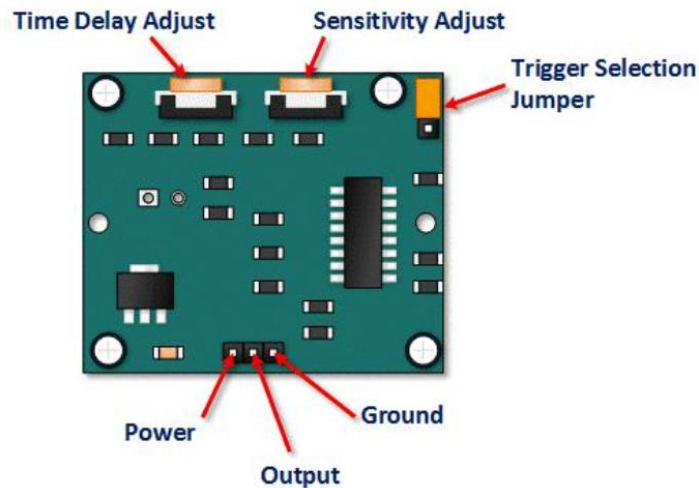
Component Introduction

CAPTEUR PIR:

Les capteurs PIR sont plus compliqués que la plupart des autres capteurs expliqués dans ce didacticiel (comme les cellules photoélectriques, les FSR et les commutateurs d'inclinaison) car plusieurs variables affectent l'entrée et la sortie des capteurs.

Le capteur PIR lui-même a deux fentes. Chaque fente est faite d'un matériau spécial sensible aux IR. L'objectif utilisé ici ne fait pas vraiment grand-chose et nous voyons donc que les deux fentes peuvent détecter au-delà d'une certaine distance (essentiellement la sensibilité du capteur). Lorsque le capteur est inactif, les deux fentes détectent la même quantité d'IR, la quantité ambiante rayonnée depuis la pièce, les murs ou l'extérieur. Lorsqu'un corps chaud comme un humain ou un animal passe, il intercepte d'abord la moitié du capteur PIR, ce qui provoque un changement différentiel positif entre les deux moitiés. Lorsque le corps chaud quitte la zone de détection, l'inverse se produit, le capteur générant un changement différentiel négatif. Ces impulsions de changement sont ce qui est détecté.





Pin ou contrôle	Fonction
Réglage de la temporisation	Définit la durée pendant laquelle la sortie reste élevée après la détection de motif N'importe où de 5 secondes à 5 minutes.
Réglage de la sensibilité	Règle la portée de détection ... de 3 mètres à 7 mètres
Cavalier de sélection de déclenchement	Défini pour des déclencheurs uniques ou répétables.
Broche de masse	Entrée au sol
Broche de sortie	Faible lorsqu'aucun mouvement n'est détecté. Élevé lorsqu'un mouvement est détecté. Haut est de 3,3 V
Broche d'alimentation	Entrée d'alimentation 5 à 20 VDC

Description fonctionnelle du HC SR501 PIR

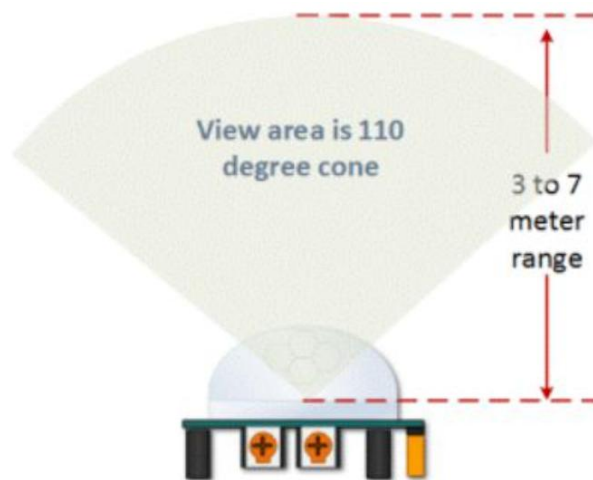
Le SR501 détectera les changements infrarouges et, s'il est interprété comme un mouvement, mettra sa sortie à un niveau bas. Ce qui est ou n'est pas interprété comme un mouvement dépend en grande partie des réglages et ajustements de l'utilisateur.

Initialisation de l'appareil.

L'initialisation de l'appareil nécessite près d'une minute. Pendant cette période, il peut et produira souvent de faux signaux de détection. La logique du circuit ou du contrôleur doit prendre en compte cette période d'initialisation.

Zone de détection de l'appareil

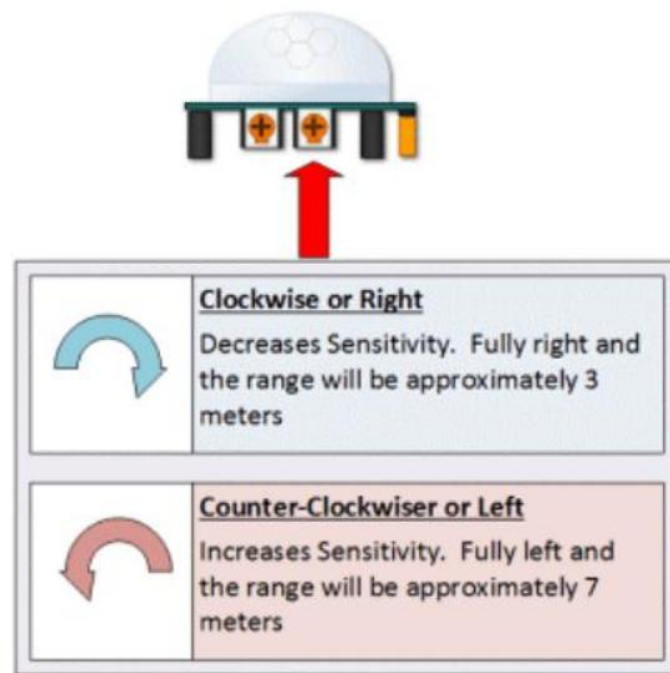
L'appareil détectera le mouvement à l'intérieur d'un cône de 110 degrés avec une portée de 3 à 7 mètres.



Zone de vue HC SR501

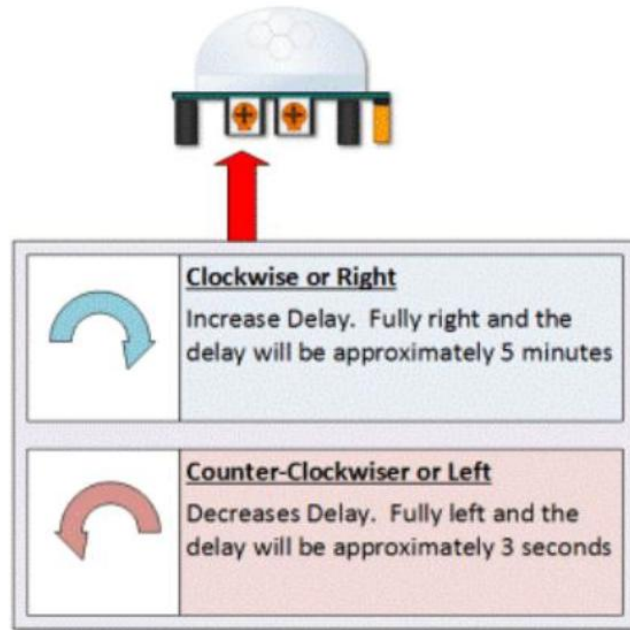
Réglage de la plage PIR (sensibilité)

Comme mentionné, la plage réglable est d'environ 3 à 7 mètres. L'illustration ci-dessous montre cet ajustement.



HC SR501 Réglage de la sensibilité Réglage de la temporisation

Le réglage de la temporisation détermine la durée pendant laquelle la sortie du module de capteur PIR restera élevée après la détection de mouvement. La plage est d'environ 3 secondes à cinq minutes.



Réglage de la temporisation HC SR501

3 secondes d'arrêt après la fin du délai - IMPORTANT

La sortie de cet appareil passera à FAIBLE (ou Éteint) pendant environ 3 secondes APRÈS la fin de la temporisation. En d'autres termes, TOUTE détection de mouvement est bloquée pendant cette période de trois secondes.

Par exemple:

Imaginez que vous êtes en mode de déclenchement unique et que votre délai est réglé sur 5 secondes.

Le PIR détectera le mouvement et le réglera à un niveau élevé pendant 5 secondes.

Après cinq secondes, le PIR règle sa sortie à un niveau bas pendant environ 3 secondes.

Pendant les trois secondes, le PIR ne détectera pas de mouvement.

Après trois secondes, le PIR détectera à nouveau le mouvement et le mouvement détecté réglera à nouveau la sortie à un niveau élevé.

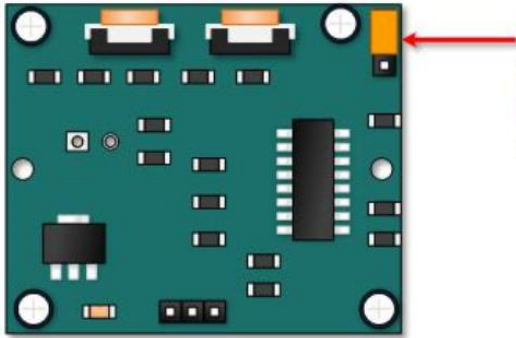
Cavalier de sélection du mode de déclenchement

Le cavalier de sélection du mode de déclenchement vous permet de choisir entre des déclencheurs simples et répétables.

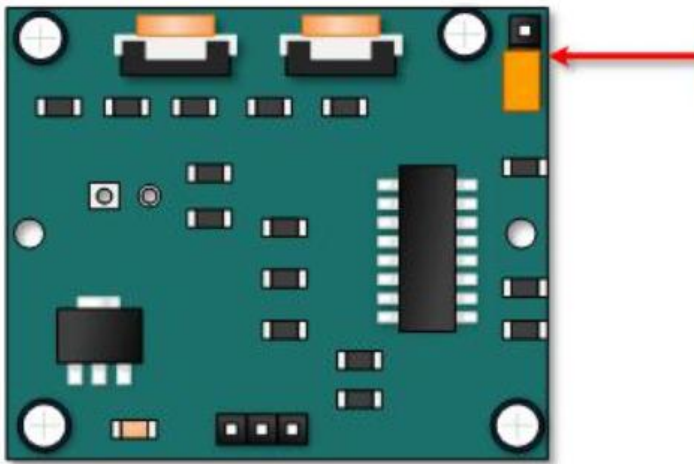
L'effet de ce réglage de cavalier est de déterminer quand commence la temporisation.

● **DÉCLENCHEMENT UNIQUE** - Le délai commence immédiatement lorsque le mouvement est détecté pour la première fois.

● **DÉCLENCHEMENT RÉPÉTABLE** - Chaque mouvement détecté réinitialise le délai. Ainsi, la temporisation commence avec le dernier mouvement détecté.



Mode de déclenchement unique -La temporisation démarre immédiatement après la détection d'un mouvement. La détection continue est bloquée.



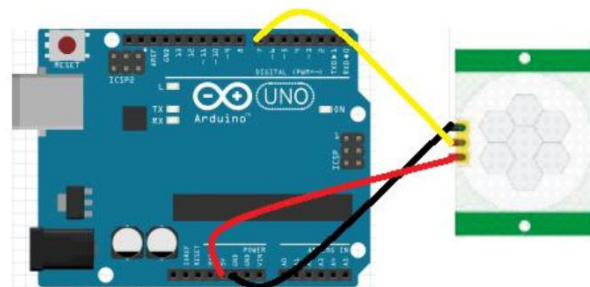
Mode de déclenchement répétable - La temporisation est redémarrée chaque fois qu'un mouvement est détecté.

Exemples d'application de piste de danse HC-SR501

Imaginez que vous vouliez contrôler l'éclairage sur une piste de danse en fonction de l'endroit où les danseurs dansent.

Comprendre comment la temporisation et le mode de déclenchement interagissent sera nécessaire pour contrôler cet éclairage de la manière que vous souhaitez.

Schéma de câblage



La connexion de capteurs PIR à un microcontrôleur est vraiment simple. Le PIR agit comme une sortie numérique, il vous suffit donc d'écouter que la broche bascule vers le haut (détecté) ou le bas (non détecté).

Il est probable que vous souhaitiez un redéclenchement, alors assurez-vous de mettre le cavalier

en position H! Alimenter le PIR avec 5V et connectez la terre à la terre. Connectez ensuite la sortie à une broche numérique. Dans cet exemple, nous utiliserons la broche 7.

Code

```
int ledPin = 13;  // LED on Pin 13 of Arduino
int pirPin = 7;  // Input for HC-S501
```

```
int pirValue; // Place to store read PIR Value
```

```
void setup() {
```

```
    pinMode(ledPin, OUTPUT);
    pinMode(pirPin, INPUT);
```

```
    digitalWrite(ledPin, LOW);
    Serial.begin(9600);
```

```
}
```

```
void loop() {
```

```
    pirValue = digitalRead(pirPin);
    digitalWrite(ledPin, pirValue);
    Serial.print(pirValue);
```

```
}
```