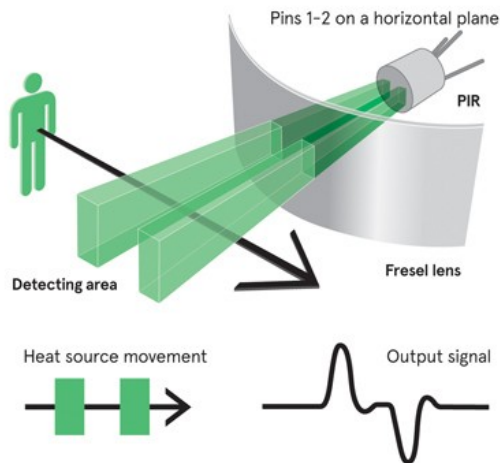


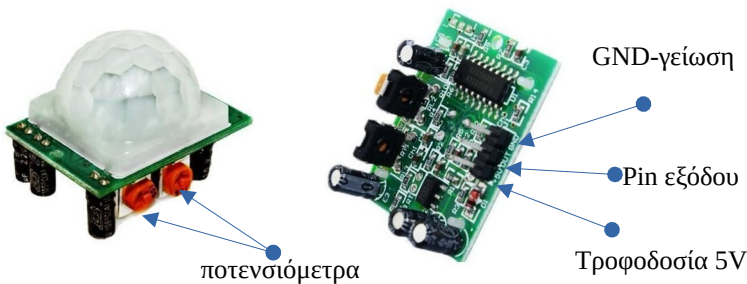
PIR sensor HC-SR501 : Αισθητήρας κίνησης και Arduino



Τα αρχικά **PIR** προέρχονται από τις λέξεις **Passive Infrared** (Παθητικό υπέρυθρο) αφού ο αισθητήρας κίνησης PIR μετρά το υπέρυθρο φως από αντικείμενα στο οπτικό του πεδίο. Έτσι, μπορεί να ανιχνεύσει κίνηση με βάση τις αλλαγές στο υπέρυθρο φως στο περιβάλλον. Είναι ιδανικό (βλ. **εικόνα 1**) να ανιχνεύσουμε εάν ένας άνθρωπος έχει μετακινηθεί μέσα ή έξω από το εύρος του αισθητήρα .

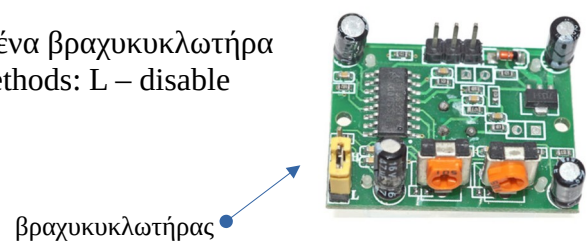
εικόνα 1

Ορισμένοι αισθητήρες κίνησης PIR διαθέτουν δύο ρυθμιζόμενα ποτενσιόμετρα για την αλλαγή της ευαισθησίας και της διάρκειας του σήματος ενεργοποίησης όπως βλέπουμε στην **εικόνα 2**. Σύμφωνα με το datasheet του αισθητήρα απαιτείται περίπου 1 λεπτό μετά την τροφοδοσία για να μπει σε κανονική λειτουργία.



εικόνα 2

Κάποια modules (**εικόνα 3**) διαθέτουν στο πίσω μέρος ένα βραχυκυκλωτήρα ανάλογα με τη θέση του οποίου επιτρέπουν Trigger methods: L – disable repeat trigger, H enable repeat trigger .

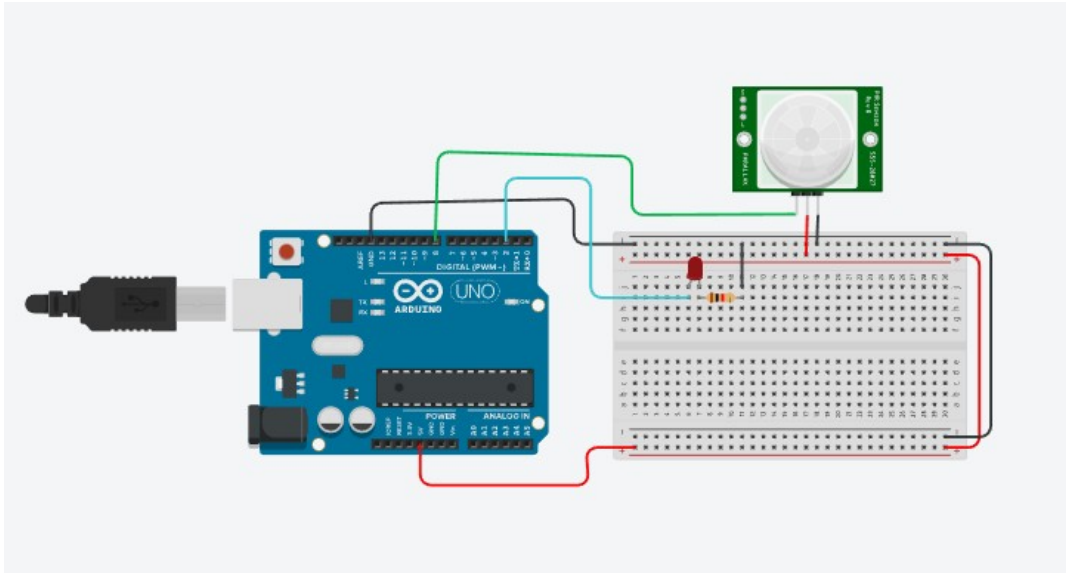


εικόνα 3

Παράδειγμα κατανόησης εφαρμογής στο Tinkercad : [pir sensor example](#)

Στην εικόνα που ακολουθεί φαίνεται το κύκλωμα.

Προσοχή!!! Οι ακροδέκτες του αισθητήρα στο tinkercad είναι με διαφορετική σειρά από ότι στον πραγματικό αισθητήρα Pir. Κάνετε την σωστή αντιστοιχία όταν μεταφέρετε τον κώδικα στην εφαρμογή Arduino IDE και κατ επέκταση στον μικροελεγκτή σας.



Ο κώδικας

```
// C++ code
int sensorState;
void setup()
{
  pinMode(8, INPUT);
  pinMode(2, OUTPUT);
  Serial.begin(9600);
}
void loop()
{
  // read the state of the sensor/digital input
  sensorState = digitalRead(8);

  // check if sensor pin is HIGH. if it is, set the
  // LED on.
  if (sensorState == HIGH) {
    digitalWrite(2, HIGH);
    Serial.println("Sensor activated!");
  } else {
    digitalWrite(2, LOW);
    Serial.println("Sensor is not activated!");
  }
  delay(10); // Delay a little bit to improve simulation performance
}
```