



BATMAN ÜNİVERSİTESİ
TEKNİK BİLİMLER
MESLEK YÜKSEKOKULU

(PIR) HAREKET SENSÖRÜ PROJE TANIMI



Hareket Sensörü (PIR) Nedir?

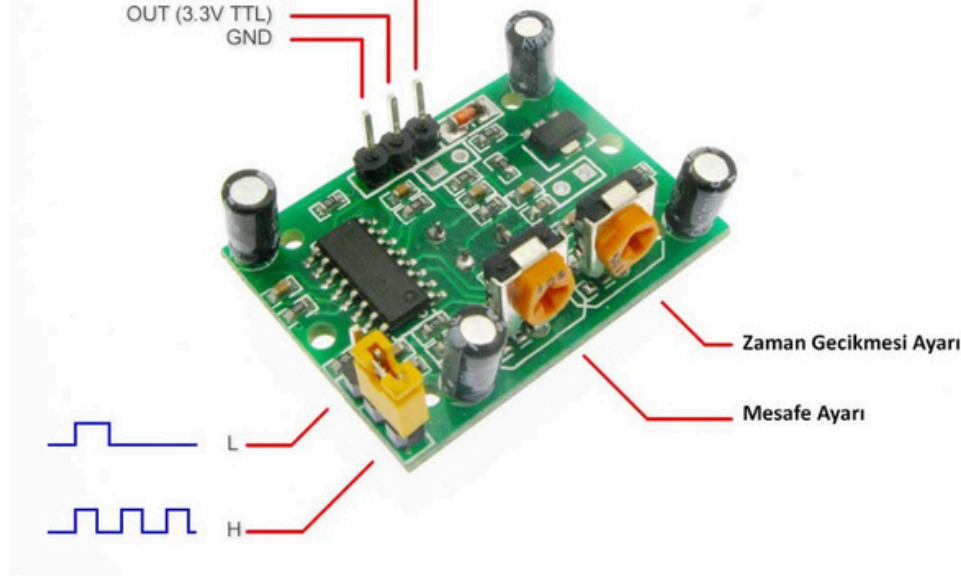
PIR (Pasif Kızılötesi) [Sensör](#)ü, çevredeki nesnelerin yaydığı kızılötesi (ısı) seviyelerindeki değişiklikleri ölçerek hareketi algılayan bir cihazdır. Hareket algılandığında PIR Sensörü çıkış piminde yüksek bir sinyal çıkarır. Bu mantık sinyali bir mikrodenetleyici tarafından okunabilir veya yüksek voltajlı cihazları değiştirebilen bir MOSFET'i tetikleyebilir. Bu, bir hareket alanını izlemek için iyi bir sensördür. Bu sensörün her iki versiyonu da aynı Fresnel lensi kullanır ve temel işlevler ikisi arasında aynı kalır.

Çalışma Mantığı

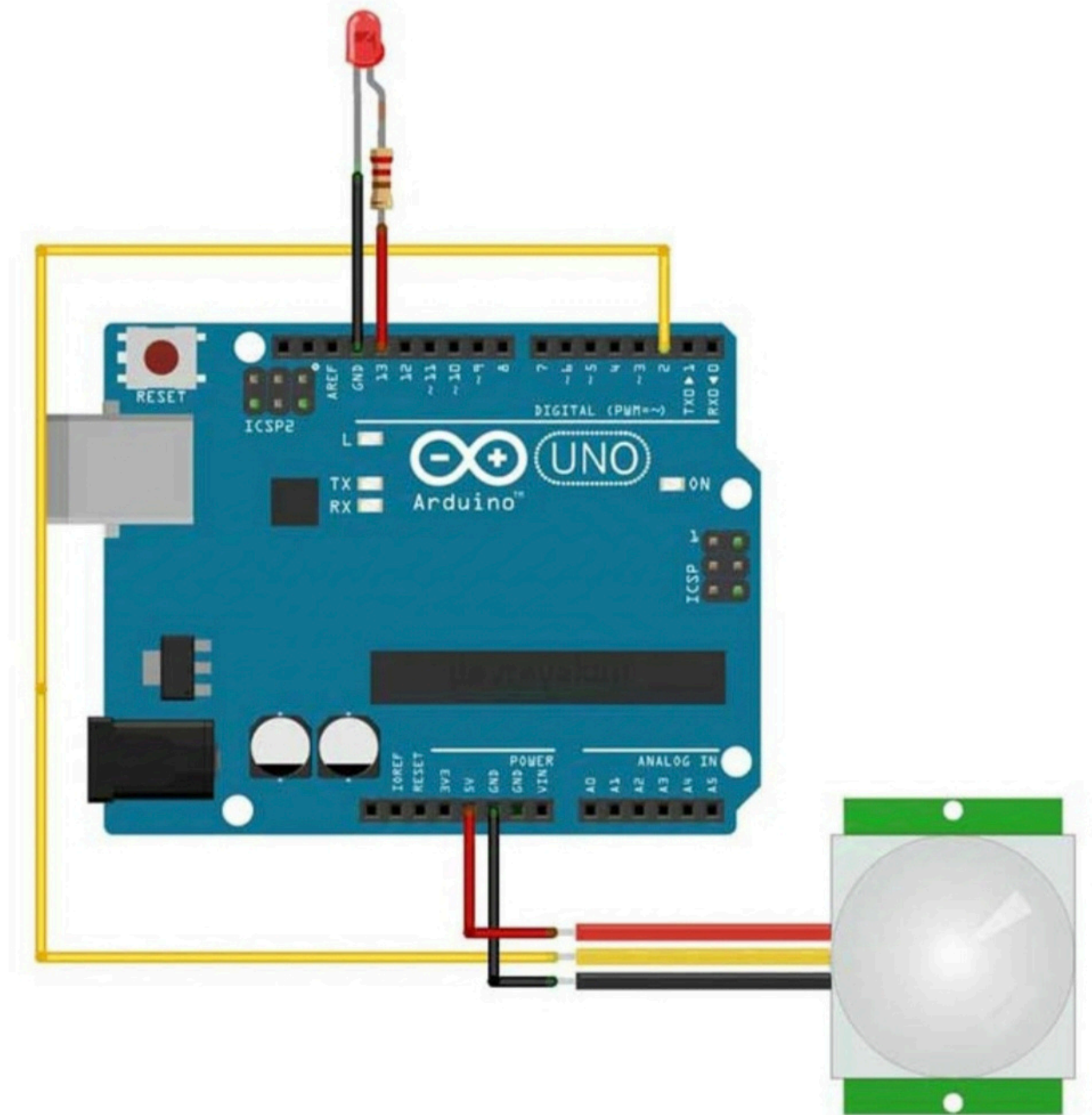
PIR sensörülerine "pasif" denir, çünkü algılanmaya yardımcı olmak için örneğin "radyasyon" veya "ışık" biçimini veya şeklini gönderecek herhangi bir "yardımcı" tarafından desteklenmezler. Tamamen sensörün çevreden neler alabildiğine, nesneler tarafından yayılana dayanır. PIR'ler aslında sadece iki sensör arasındaki farka bakar. Fark çok yüksekse devre tetiklenir yani hareketh algılar. Bu, örneğin kısa bir flaş veya oda sıcaklığındaki bir artıştan kaynaklanan yanlış pozitifleri önlemek için akıllı bir şekilde yapılır.



Gecikme Süresi



PIR'deki sensör, etrafımızdaki nesnelerden yayılan kızılötesi radyasyonu algılar veya okur. Mutlak sıfırın (-273.15° Celsius yani 0 Kelvin) üzerindeki bir sıcaklığa sahip her nesne, canlılar da dahil olarak kızılötesi yayar. Bununla birlikte, özel Termal Kızılötesi Kameralar ile bu, kızılötesi dalgalar insan gözü tarafından görülebilir hale getirilebilir. Kullandığımız modülün PIR'in sadece nispeten basit bir sensör kullandığına dikkat edin yani bir kamera olarak kullanamayız.




```
int led = 13;
int sensor = 2;
int state = LOW;
int val = 0;
void setup() {
  pinMode(led, OUTPUT);
  pinMode(sensor, INPUT);
  Serial.begin(9600);
}
void loop(){
  val = digitalRead(sensor);
  if (val == HIGH) {
    digitalWrite(led, HIGH);
    delay(100);
    if (state == LOW) {
      Serial.println("Hareket Basladi");
      state = HIGH;
    }
  }
  else {
    digitalWrite(led, LOW);
    delay(200);
    if (state == HIGH){
      Serial.println("Hareket Durdu");
      state = LOW;
    }
  }
}
```

Projenin orjinal kodu



sketch_dec18a\$

```
int ledPin = 13;           // LED'in bağlı olduğu pin
int sensorPin = 2;         // Sensörün bağlı olduğu pin
int sensorState = LOW;     // Sensörün durumu
int sensorValue = 0;       // Sensör değeri

void setup() {
  pinMode(ledPin, OUTPUT); // LED pini çıkış olarak ayarlandı
  pinMode(sensorPin, INPUT); // Sensör pini giriş olarak ayarlandı
  Serial.begin(9600);       // Seri haberleşme başlatıldı
}

void loop() {
  sensorValue = digitalRead(sensorPin); // Sensör değerini oku
  if (sensorValue == HIGH) {           // Sensör değeri HIGH ise
    digitalWrite(ledPin, HIGH);        // LED'i yak
    delay(500);                        // Kısa bir gecikme
    if (sensorState == LOW) {
      Serial.println("Hareket Başladı"); // Hareket başladığını bildir
      sensorState = HIGH;
    }
  } else {
    digitalWrite(ledPin, LOW);         // LED'i söndür
    delay(500);                        // Kısa bir gecikme
    if (sensorState == HIGH) {
      Serial.println("Hareket Durdu"); // Hareket durduğunu bildir
      sensorState = LOW;
    }
  }
}
```

Buda değiştirilmiş hali
(Bu kod kullanımda)

Yükleme tamamlandı.

Çalışmanız programın 2250 bayt (6 %) saklama alanını kullandı. Maksimum 32256 bayt.

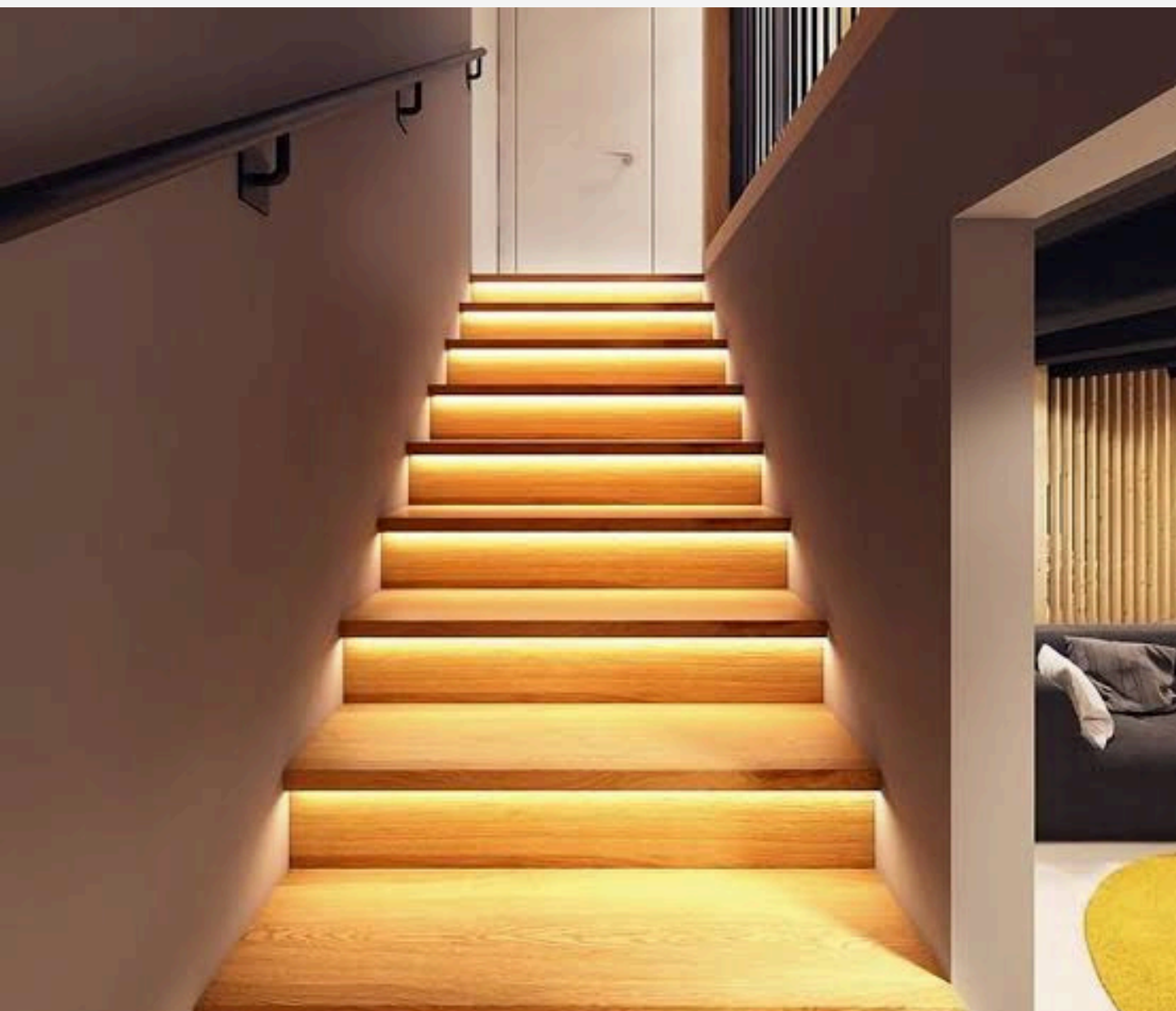
Global değişkenler belleğin 222 byte kadarını (10%) kullanıyor. Yerel değişkenler için 1826 byte yer kalıyor. En fazla 2048 byte kullanılabilir.

11



Kullanım amacı ve alanları

bir odaya giriş yapıldığında otomatik olarak ışıkların açılmasını sağlamak, güvenlik sistemlerinde hareketleri izlemek, akıllı ev sistemlerinde enerji tasarrufu sağlamak gibi birçok vb. fonksiyon özelliklerine sahiptir.



**BENİ DİNLEDİĞİNİZ İÇİN
TEŞEKKÜR EDERİM**

~ Furkan apraz ~