

# Infrarotfernbedienung und -sensor

## Benötigtes Material

- Arduino Uno
- USB-Kabel für den Arduino
- LED
- Infrarotsensor
- Infrarotfernbedienung
- 2x MW-Kabel (oder 2x MM-Kabel, je nach Modell des Infrarotsensors)

## Benötigte Software

- Bibliothek IRremote von shirriff

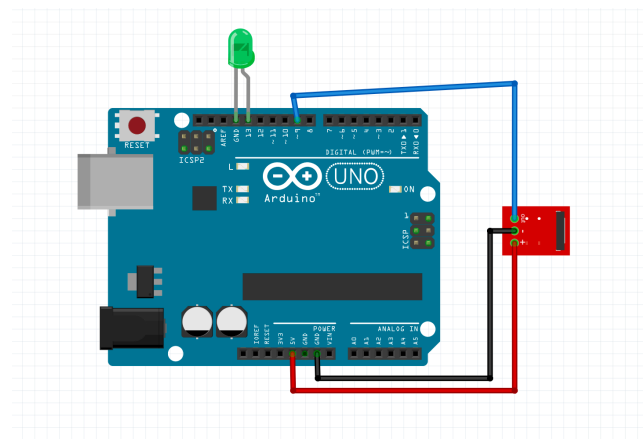
## Beschreibung

Der Infrarotsensor zusammen mit einer Infrarotfernbedienung dient dazu, den Arduino aus einigen Metern Entfernung zu steuern. Die eingesetzte infrarote Strahlung liegt dabei knapp außerhalb des für Menschen sichtbaren Spektrums.

Unterschiedliche Fernbedienungs-Modelle senden dabei unterschiedliche Signale aus. Die nicht gerade leichte Entschlüsselung dieser Signale wird glücklicherweise von der Bibliothek „IRremote“ übernommen, sodass die Nutzung eines Infrarotsensors sehr einfach ist (siehe Bibliotheken einbinden).

## Verkabelung


Arduino	LED	Infrarotsensor
13	langes Bein	Plus
5V		Minus
GND		out
9		



Schaltplan

## Code

Der Arduino wird wie oben beschrieben aufgebaut und dann wird der unten stehende Code hochgeladen.

Da jede Fernbedienung andere Knöpfe hat und unterschiedliche Signale sendet, müssen wir mit dem seriellen Monitor herausfinden, welchen Zahlencode die verschiedenen Knöpfe der Fernbedienung liefern. Dazu wird der serielle Monitor mit der Lupe  geöffnet. Drückt man nun Knöpfe auf der Infrarotfernbedienung, so sollte die im Infrarotsensor eingebaute LED aufleuchten und zehnstellige Zahlen auf dem seriellen Monitor erscheinen. Drückt man denselben Knopf auf der Fernbedienung, so sollte auch dieselbe Zahl auf dem seriellen Monitor erscheinen.

### ! Achtung!

Es kann sein, dass zwei verschiedene Zahlen beim mehrfachen Drücken desselben Knopfes erscheinen, etwa wenn man den Knopf mal länger und mal kürzer drückt. Man sollte dann die Zahl nehmen, die am häufigsten auftaucht.

Nun sollte man sich die Zahlen notieren, welche zu den Knöpfen gehören, mit welchen man den Arduino später steuern möchte. Diese kann man auch mit der Tastenkombination STRG+C kopieren, nachdem man die Zahl mit der Maus markiert hat. Mit STRG+V fügt man diese an anderer Stelle ein. In diesem Beispiel nutzen wir die Zahlen für die Variablen INFRAROT\_SIGNAL\_AN und INFRAROT\_SIGNAL\_AUS im folgenden Programm:

```
// Skript zum Steuern einer LED mithilfe des Infrarot-Sensors.

// Binde die Bibliothek "IRremote" ein.
#include <IRremote.h>

int irPin = 9;
int ledPin = 13;

// Erstelle ein Objekt vom Typ IRrecv für den Pin namens irPin.
IRrecv infrarot_empfaenger(irPin);
// Erstelle ein Objekt vom Typ decode_results zum Speichern der empfangenen
// Signale.
decode_results signale;

// Ersetze diese Zahlen mit denen des seriellen Monitors.
long INFRAROT_SIGNAL_AN = 3772784863;
long INFRAROT_SIGNAL_AUS = 3772817503;

void setup() {
  // Erlaube es, dass Infrarot-Signale empfangen werden können.
  infrarot_empfaenger.enableIRIn();
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // Die Methode decode() gibt TRUE zurück, falls der Funktionsaufruf korrekt
  // ausgeführt wurde, also das Signal dekodiert werden konnte. &signale ist die
  // Adresse des Objektes signale im physischen Speicher des Arduinos.
  if (infrarot_empfaenger.decode(&signale)) {
    Serial.println(signale.value, DEC);
    // Entspricht das empfangene Signal dem an-Knopf?
    if (signale.value == INFRAROT_SIGNAL_AN) {
      // Schalte die LED ein.
      digitalWrite(ledPin, HIGH);
      // Entspricht das empfangene Signal dem aus-Knopf?
    } else if (signale.value == INFRAROT_SIGNAL_AUS) {
      // Schalte die LED aus.
      digitalWrite(ledPin, LOW);
    }
    // Erlaube es, dass weitere Infrarot-Signale empfangen werden können.
    infrarot_empfaenger.resume();
  }
}
```

Dieses Programm empfängt durch den Infrarotsensor Signale. Entsprechen die Signale dem Drücken der jeweiligen Knöpfe auf der Infrarotfernbedienung, so wird die LED an- oder ausgeschaltet.

Für den Einsatz eines Infrarotsensors sind der Phantasie keine Grenzen gesetzt. Beispielsweise lassen sich hiermit mit einer Fernbedienung für den Fernseher gleichzeitig die Lautstärke und die Helligkeit einer LED regeln, wenn man mithilfe eines Potentiometers den Widerstand des Stromkreises steuert.

## Aufgaben

- 1 Überlege dir, was man mit einem Arduino und einem Infrarotsensor steuern könnte!

Antwort: \_\_\_\_\_

- ② Versuche herauszufinden, aus welcher Entfernung du den Arduino mit deiner Fernbedienung steuern kannst.

Antwort: \_\_\_\_\_

- ③ Versuche herauszufinden, in welchem Winkel die Fernbedienung maximal zum Sensor gehalten werden darf, damit dieser noch die Signale empfängt.

Antwort: \_\_\_\_\_

- ④ Versuche herauszufinden, durch welche Materialien das infrarote Licht geblockt wird (z.B. Glas, durchsichtige Folien in verschiedenen Farben, Papier, Pappe, Metal usw.).

Antwort: \_\_\_\_\_

- ⑤ Stelle ein Hindernis zwischen dem Sensor und der Fernbedienung auf und versuche den Arduino mit Hilfe eines Spiegels zu steuern. Funktioniert das?

Antwort: \_\_\_\_\_

- ⑥ Wie könnte man das Programm abändern, sodass man mit nur einem Knopf die LED an- und ausschalten kann?

Antwort: \_\_\_\_\_

- ⑦ Schließe mehrere, verschiedenfarbige LEDs an den Arduino an und programmiere die Fernbedienung so, dass du jede einzeln an- und ausschalten kannst. Programmiere dann eine Taste, mit der du alle an- oder ausschalten kannst.

Antwort: \_\_\_\_\_

- ⑧ Programmiere eine Taste deiner Fernbedienung so, dass die LED drei mal blinkt. Dann programmiere zwei weitere Tasten so, dass du mit diesen einstellen kannst, wie schnell die LED beim Drücken der ersten Taste blinkt (schneller oder langsamer).

Antwort: \_\_\_\_\_

Das Material und dessen Inhalte sind - sofern nicht anders angegeben - lizenziert unter der Creative Commons Lizenz CC BY-NC-SA 4.0 (für den vollständigen Lizenztext siehe <https://creativecommons.org/licenses/by-sa/4.0/legalcode>)

