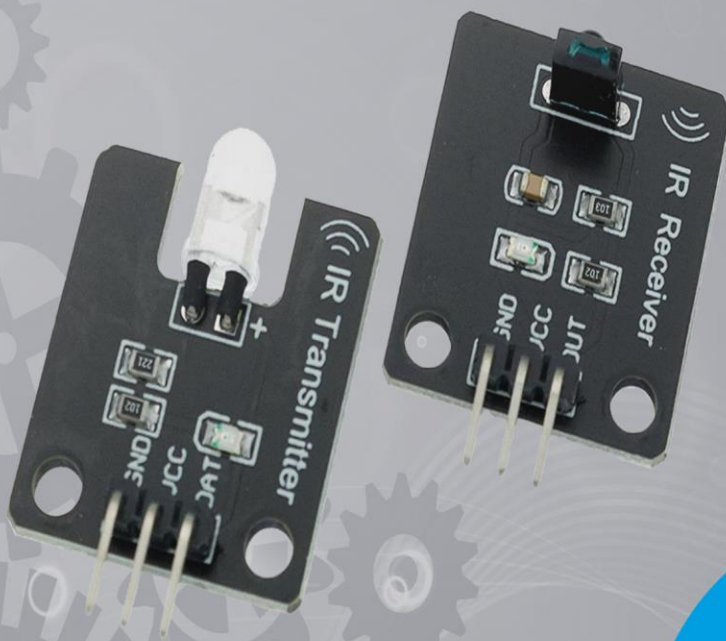


# BEDIENUNGSANLEITUNG

Infrarot Sender & Empfänger Set 5V



# Inhaltsverzeichnis

|   |   |
|---|---|
| Sicherheitshinweise .....               | 3 |
| Beschreibung .....                      | 4 |
| Details .....                           | 4 |
| Pinbelegung .....                       | 4 |
| Funktionsweise .....                    | 4 |
| Verschlüsselung mit NEC-Protokoll ..... | 5 |
| Beispiel mit Arduino .....              | 6 |

## Sicherheitshinweise

### *Betriebsspannung beachten:*

- Sowohl Sender als auch Empfänger benötigen eine Betriebsspannung von **5V**. Verwenden Sie keine höhere oder niedrigere Spannung, um Beschädigungen der Module zu vermeiden.

### *Betriebstemperatur einhalten:*

- Achten Sie darauf, die Module nur innerhalb des zulässigen Temperaturbereichs von **-25°C bis 85°C** zu betreiben. Temperaturen außerhalb dieses Bereichs können die Funktionsfähigkeit beeinträchtigen oder dauerhafte Schäden verursachen.

### *Winkel der Signalübertragung berücksichtigen:*

- Der Sender strahlt im Winkel von ca. **20°**, der Empfänger kann Signale im Winkel von ca. **35°** empfangen. Installieren Sie die Module entsprechend, um die beste Signalqualität zu erzielen.

### *Direkter Augenkontakt vermeiden:*

- Der Infrarotsender arbeitet mit einer Wellenlänge von **940 nm**, was für das menschliche Auge unsichtbar ist. Dennoch kann intensives IR-Licht auf Dauer die Augen schädigen. Richten Sie den Sender nicht direkt auf Ihre Augen oder die von anderen Personen.

### *Umgebungseinflüsse beachten:*

- Störquellen wie direkte Sonneneinstrahlung oder andere IR-Signale können die Funktion der Module beeinträchtigen. Achten Sie darauf, dass sich keine starken Infrarotquellen in der Nähe befinden, um Signalstörungen zu vermeiden.

### *Richtige Montage sicherstellen:*

- Verwenden Sie die vorhandenen 3-mm-Bohrungen der Module für eine sichere Befestigung. Eine stabile Installation stellt sicher, dass sich die Module nicht unbeabsichtigt verschieben und die Signalqualität konstant bleibt.

### *Korrekte Polarität bei der Stromversorgung beachten:*

- Stellen Sie sicher, dass die Stromversorgung korrekt angeschlossen wird, um eine Verpolung zu vermeiden, die zur Zerstörung der Module führen könnte.

### *Elektronische Bauteile vor Feuchtigkeit schützen:*

- Vermeiden Sie den Betrieb der Module in feuchten oder nassen Umgebungen. Feuchtigkeit kann zu Kurzschlüssen und Funktionsstörungen führen.

### *Sicherer Umgang mit Kabeln und Verbindungen:*

- Achten Sie darauf, dass alle Verbindungen fest und sicher sind, um den korrekten Betrieb zu gewährleisten. Beschädigte Kabel sollten umgehend ersetzt werden, um Kurzschlüsse oder Funktionsfehler zu verhindern.

## Beschreibung

Mit dieser Kombination aus Infrarot-Empfänger und Infrarot-Sender Module bringen Sie schnell und einfach Infrarotsteuerungen in Ihr Raspberry Pi oder Arduino Projekt. Mit diesen Modulen können Sie zum Beispiel Daten per Infrarot zwischen zwei Arduinos Versenden und Empfangen.

Bei Verwendung mit Arduino empfehlen wir die IRremote Bibliothek, die Sie auch bequem über die Bibliothekverwaltung der Arduino IDE abrufen können.

## Details

### Sender

- Betriebsspannung: 5V
- Betriebstemperatur: -25°C bis 85°C
- Wellenlänge: 940nm
- Signal: Digital
- Reichweite: ca. 1,3 Meter (5V 38Khz)
- Winkel: ca. 20°
- Gewicht: 1,5 g
- Abmessungen: 23,5 x 20,8mm
- Bohrungen: 3mm

### Empfänger

- Betriebsspannung: 5V
- Betriebstemperatur: -25°C bis 85°C
- Frequenz: 38Khz
- Signal: Digital
- Reichweite: ca. 10 Meter
- Winkel: ca. 35°
- Gewicht: 1,5 g
- Abmessungen: 23,5 x 20,8mm
- Bohrungen: 3mm

## Pinbelegung

### Sender

- DAT - Dateneingang
- VCC - 5V DC
- GND – Masse

### Empfänger

- OUT - Datenausgang
- VCC – 5V DC
- GND - Masse

## Funktionsweise

Der IR-Empfänger wandelt das Lichtsignal, welches vom IR-Sender gesendet wird, in ein schwaches, elektronisches Signal um. Dieses Signal wird dann durch einen IC verstärkt, automatisch gefiltert, geglättet und demoduliert, bis das

ursprüngliche Signal des Senders wiederhergestellt ist. Dieses Signal wird dann an den Output-Pin des IR-Empfängers gesendet.

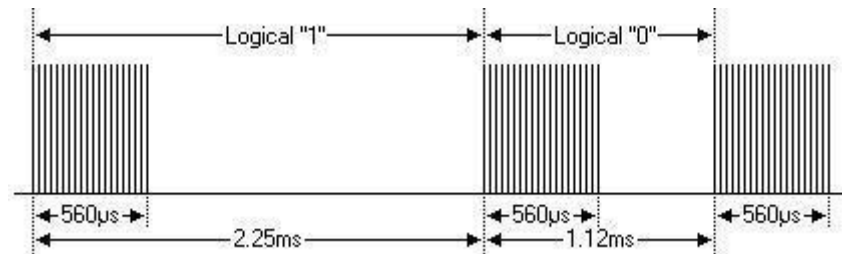
## Verschlüsselung mit NEC-Protokoll

Wenn Sie z.B. die Signale einer IR-Fernbedienung auslesen wollen, müssen Sie zuerst wissen, wie sie verschlüsselt sind. Die gängigste Verschlüsselung ist das NEC-Protokoll.

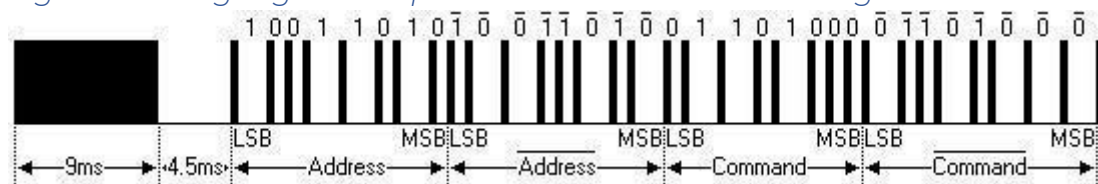
### Technische Daten des NEC-Protokolls

- 8-bit Adressen und 8-bit Befehlsängen.
- Adresse und Befehl werden zur Verlässlichkeit immer wiederholt.
- Puls Distance Modulation (PDM)
- Frequenz von 38 Khz
- Bit time von 1.125 ms oder 2.25 ms

### Definition von 0 und 1 im NEC-Protokoll



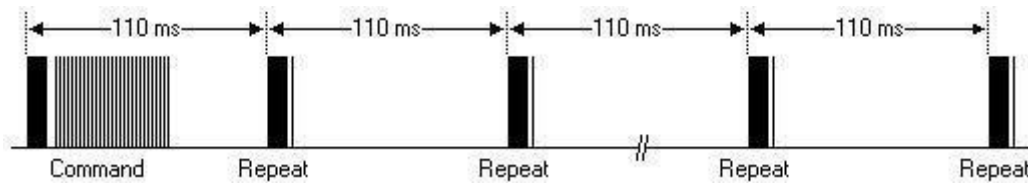
### Signalübertragung bei Knopfdruck auf der Fernbedienung



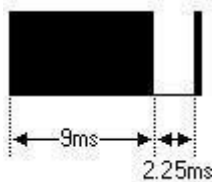
Diese Abbildung zeigt eine typische Impulsfolge des NEC-Protokolls. Bei diesem Protokoll wird das LSB (Least Significant Bit) zuerst übertragen. In diesem Fall wird die Adresse \$59 und der Befehl \$16 übertragen. Eine Nachricht wird durch einen 9 ms AGC-Burst eingeleitet, der bei den früheren IR-Empfängern zur Einstellung der Verstärkung verwendet wurde. AGC steht kurz für engl. „automatic gain control“ und dient dazu, den Ausgangspegel eines Verstärkers konstant zu halten.

Auf diesen AGC-Burst folgt ein 4,5 ms langes Leerzeichen, auf das die Adresse und der Befehl folgen. Adresse und Befehl werden zwei Mal übertragen, um die Fehlerquote zu verringern. Beim zweiten Übertragen sind alle Bits invertiert und können zur Überprüfung der empfangenen Nachricht verwendet werden. Die Gesamtübertragungszeit ist konstant, da jedes Bit mit seiner invertierten Länge wiederholt wird. Wenn Sie an diese Zuverlässigkeit nicht benötigen, können Sie die invertierten Werte auch ignorieren, oder die Adresse und den Befehl auf jeweils 16 Bit erweitern!

Ein Impuls wird gesendet, wenn eine Taste gedrückt und nach einer bestimmten Zeit losgelassen wird.



Ein Befehl wird nur einmal übertragen, auch wenn die Taste der Fernbedienung gedrückt bleibt. Alle 110ms wird ein Wiederholungscode gesendet, solange die Taste gedrückt bleibt. Dieser Wiederholungscode ist einfach ein 9ms AGC-Burst, gefolgt von einer 2,25ms Pause und einem 560µs Burst.



Wenn der Impuls im Empfänger eintrifft, findet eine Dekodierung, Signalverstärkung und Wellenlängung statt. Sie müssen sicherstellen, dass der Ausgangspegel genau das Gegenteil von dem des sendenden Signals ist. Das heißt, wenn kein Infrarotsignal vorhanden ist, ist der Ausgangspegel hoch; Wenn ein Infrarotsignal vorhanden ist, ist der Ausgangspegel low. Sie können das Signal des Empfängers mit einem Oszilloskop untersuchen, um das Signal besser zu verstehen.

## Beispiel mit Arduino

### Schritt 1: Verkabelung

#### IR-Sender Modul

1. VCC Pin des Senders an **5V** des Arduino anschließen.
2. GND Pin des Senders an **GND** des Arduino anschließen.
3. DAT Pin des Senders an einen digitalen Pin des Arduino (z.B. **Pin 3**) anschließen.

#### IR-Empfänger Modul

1. VCC Pin des Empfängers an **5V** des Arduino anschließen.
2. GND Pin des Empfängers an **GND** des Arduino anschließen.
3. OUT Pin des Empfängers an einen digitalen Pin des Arduino (z.B. **Pin 11**) anschließen.

### Schritt 2: Arduino IDE vorbereiten

1. Öffnen Sie die **Arduino IDE** und verbinden Sie Ihr Arduino-Board mit dem Computer.
2. Installieren Sie die Bibliothek "**IRremote**", die Ihnen das Senden und Empfangen von Infrarotsignalen ermöglicht:
  - Gehen Sie zu „**Werkzeuge > Bibliotheken verwalten.**“
  - Suchen Sie nach **IRremote** und installiere sie.

### Schritt 3: Programmierung des Arduino

Senden von IR-Signalen:

```
#include <IRremote.h> // Die IRremote.h-Bibliothek einbinden

const int irSenderPin = 3; // Der Pin, an dem der IR-Sender angeschlossen
ist
IRsend irsend(irSenderPin); // Erstellt ein IRsend-Objekt und legt den Pin
fest

void setup() {
  // Keine speziellen Setups erforderlich für das Senden von IR-Signalen
}

void loop() {
  irsend.sendNEC(0xFFA25D, 32); // Sendet ein IR-Signal im NEC-Protokoll,
Code: 0xFFA25D, 32-Bit
  delay(2000); // Wartet 2 Sekunden, bevor das nächste
Signal gesendet wird
}
```

Dieses Programm sendet ein Infrarotsignal mit dem NEC-Protokoll. Sie können den Code **0xFFA25D** durch einen anderen Code ersetzen, je nach Gerät oder Fernbedienung.

### Empfangen von IR-Signalen

```
#include <Arduino.h>
#include <IRremote.hpp> // IRremote Bibliothek einbinden

const int irReceiverPin = 11; // Der Pin für den IR-Empfänger

void setup() {
  Serial.begin(115200);
  Serial.println(F("IR-Empfänger gestartet"));

  // IR-Empfänger an Pin 11 starten
  IrReceiver.begin(irReceiverPin, ENABLE_LED_FEEDBACK);

  Serial.print(F("Bereit, IR-Signale zu empfangen an Pin "));
  Serial.println(irReceiverPin);
}

void loop() {
  // Überprüfen, ob ein IR-Signal empfangen wurde
  if (IrReceiver.decode()) {

    // Wenn das Protokoll unbekannt ist
    if (IrReceiver.decodedIRData.protocol == UNKNOWN) {
      Serial.println(F("Unbekanntes oder Rauschen erkannt"));
      IrReceiver.printIRResultRawFormatted(&Serial, true);
    }
  }
}
```

```
    } else {  
        // Gültige IR-Daten anzeigen  
        IrReceiver.printIRResultShort(&Serial);  
    }  
  
    IrReceiver.resume(); // Bereit für das nächste Signal  
    Serial.println();  
  
    // Beispielaktionen basierend auf empfangenen Befehlen  
    if (IrReceiver.decodedIRData.command == 0x10) {  
        // Aktion 1 ausführen  
    } else if (IrReceiver.decodedIRData.command == 0x11) {  
        // Aktion 2 ausführen  
    }  
}  
}
```

Dieses Programm empfängt Infrarotsignale im NEC-Protokoll und gibt den empfangenen Code im **seriellen Monitor** der Arduino IDE aus.

#### *Schritt 4: Hochladen und Testen*

- Lade den **Sender-Sketch** auf das Arduino-Board, das mit dem IR-Sender verbunden ist, um das Senden von IR-Signalen zu testen.
- Lade den **Empfänger-Sketch** auf das Arduino-Board, das mit dem IR-Empfänger verbunden ist, um das Empfangen von IR-Signalen zu testen.
- Öffne den **seriellen Monitor** in der Arduino IDE (9600 Baudrate), um die empfangenen Signale zu sehen.

#### *Schritt 5: Fehlerbehebung*

- **Keine Signale empfangen:** Überprüfe, ob der Empfänger richtig verkabelt ist und die Fernbedienung oder das sendende Arduino korrekt funktioniert.
- **Zu kurze Reichweite:** Achte darauf, dass der IR-Sender und Empfänger in der richtigen Entfernung (ca. 1,3 m für Sender und bis zu 10 m für Empfänger) und im richtigen Winkel (Sender: 20°, Empfänger: 35°) zueinander stehen.
- **Fehlende Signalübertragung:** Teste den Sender und Empfänger in einer Umgebung mit wenig Umgebungslicht, da Sonnenlicht oder andere Lichtquellen die IR-Kommunikation stören können.