

Infrator Kit

Der schlaue Schalter



Jugend Programmiert

Vorwort

Alle Schaltungen, Programme und Bauteile wurden vorher von uns geprüft. Trotzdem können wir leider Fehler im Buch, den Schaltungen oder Programmen nicht ausschließen. Bei Fragen oder Problemen gibt es unten die Kontaktmöglichkeiten. Mehr Informationen findest du auch auf der Internetadresse der Seitenzahl.

LEDs

LEDs sollten nicht aus kurzer Distanz direkt angesehen werden. Ein direkter Blick kann zu Schäden an den Netzhäuten führen. Dies kann gefährlich sein, auch wenn die Verletzungen nicht sofort klar erkennbar sind. Die LEDs dürfen nur wie in den Anleitungen beschrieben verwendet werden, höhere Ströme oder Stromspannungen sind zu vermeiden.

Dieses Produkt entspricht den geltenden Europäischen Richtlinien und trägt ein CE-Kennzeichen. Der richtige Gebrauch ist in dem beiliegenden Buch erklärt.



Baue Schaltungen immer wie beschrieben auf, achte auch auf die verschiedenen GPIO Pins vom Raspberry Pi.

Herausgeber

Jugend Programmiert

Coding World UG (haftungsbeschränkt) | Homwer.com

Obdrupstraße 23 A, 24986 Mittelangeln

www.codingworld.io

Support: support@cw42.de

Feedback: feedback@cw42.de

Moin Moin und Willkommen

In diesem Kit werden wir uns mit dem ganzen Themenkomplex Infrarot beschäftigen. Beispielsweise eine Fernbedienung und das dazu gehörende Empfangsgerät arbeiten mit Infrarot, aber auch ein Bewegungsmelder fällt in diesen Themenkomplex, da auch dieser mit Infrarot arbeitet. Wenn wir uns die Grundlagen erst einmal angeeignet haben, geht es dann um den Servo Motor und wie wir diesen so nutzen können, dass er wie ein schlauer Schalter funktioniert. So bauen wir mit dir eine automatische Mini-Home-Anlage.

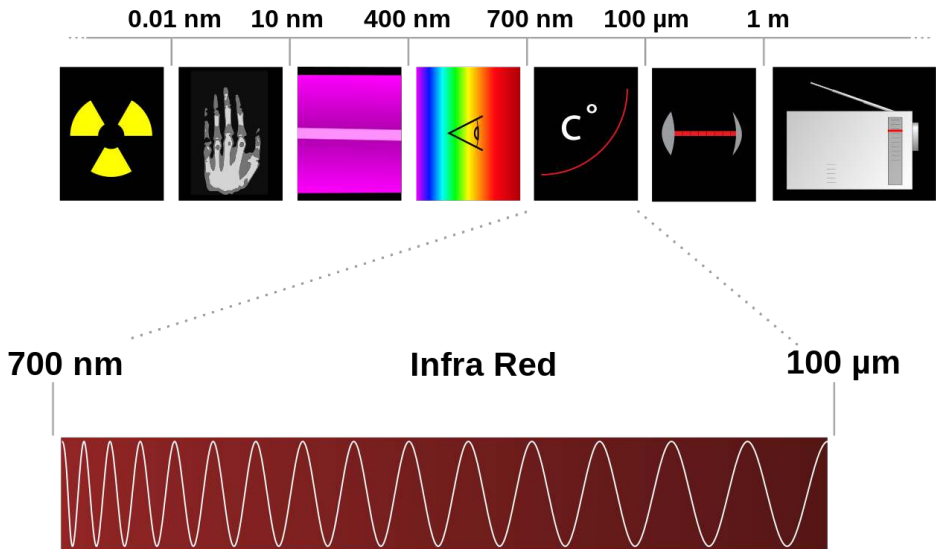
Inhalte	Seite
IR Grundlagen	Seite 2
IR Empfänger am Pi	Seite xx I
Python und der IR Empfänger	Seite xx
Servo am Pi	Seite xx
Der Bewegungssensor	Seite xx
Schlauer Schalter Projekt	Seite xx
Andere Projekte mit IR	Seite xx

Infrarot - eine Einführung

Fernbedienungen finden sich häufig in elektronischen Apparaten wieder, egal ob im Fernseher oder in der Musikanlage. Sie machen unser Leben einfacher und erlauben uns auch größere Entfernungen zu überbrücken. So überwindet der Mensch seit Jahren die unglaublich, gar galaktisch große Entfernung vom Sofa bis zum Fernseher! Um diese Distanz zu überwinden benutzen wir infrarotes Licht. Auf der Gegenseite wird dieses Licht von einem Infrarot-Licht-Sensor erkannt und in elektronische Signale umgewandelt. Technik, die darauf basiert wird auch häufig mit IR abgekürzt (engl. Infrared (oder auch deutsch Infrarot)).

Dabei gibt es viele praktische Anwendungsfälle, in denen wir diese Technik für unsere Projekte benutzen können. Dazu gehört unter anderem, dass wir alte Fernbedienungen recyceln können. Falls du keine alte Fernbedienung hast, liegt in diesem Kit eine kleine Fernbedienung, die du benutzen kannst. Wenn du möchtest, kannst du aber auch eine vorhandene Fernbedienung nutzen. Wichtig ist nur, dass diese noch funktioniert. Das menschliche Auge kann Infrarotstrahlen nicht sehen, aber es gibt einen Trick wie wir Fernbedienungen ganz einfach testen können. Öffne die Kamera App auf deinem Handy und schaue dir die Infrarot LED der Fernbedienung auf deinem Handydisplay an, Wenn du jetzt anfängst die Fernbedienung zu bedienen, sollte die LED der Fernbedienung auf dem Handydisplay sichtbar blinken. Ein weiterer Vorteil ist der, dass wir mit diesen IR Fernbedienungen gleich einen Zugriff auf viele verschiedene Tasten haben und so unseren Projekten viel mehr Variabilität verleihen können. Gerade durch die Größe und Entfernung können wir nicht nur verschiedene Dinge steuern, die weit weg sind, sondern können diese Fernbedienung auch so verstecken, dass sie für die meisten Menschen nicht zugänglich ist. Damit haben wir die perfekte Erweiterung für unser Projekt.

Doch noch ein wenig mehr zu dem, was Infrarotstrahlen sind und wie wir diese konkret einsetzen werden.



Schon im Namen - Infrarot - erkennen wir die Farbe Rot wieder. Diese elektromagnetischen Wellen liegen im Spektrum zwischen dem sichtbaren Licht und der sogenannten Terahertzstrahlung (wird zum Beispiel in Ganzkörperscannern an Flughäfen benutzt). Und obwohl wir die Infrarotstrahlen nicht sehen können, können sie aber problemlos zur Kommunikation nutzen. Bekannt vorkommen dürften dir die Infrarotstrahlung, wie vorhin schon einmal erwähnt, von einem Bewegungssensor. Dort ist sie entscheidend, weil Wärme auch Infrarotstrahlen erzeugen kann. Darüber hinaus ist es für uns praktisch, dass es auch LEDs gibt, die diese Infrarotstrahlen erzeugen können. Denn nichts anderes befindet sich vorn an einer Fernbedienung, einfach eine LED.

Dich wie funktioniert das jetzt genau mit unserer Fernbedienung? Für die Kommunikation zwischen dem Fernsehgerät (Empfänger) und der Fernbedienung (Sender) werden Infrarotstrahlen mit einer Frequenz von 38kHz genutzt. Bei der Kommunikation gibt es nur zwei Zustände: entweder es gibt eine oder es gibt eben keine. Frei nach dem Motto: Ganz oder gar nicht. Aber wie können dann die verschiedenen Tasten der Tastatur erkannt werden? Jede Taste hat quasi einen eigenen Code, ähnlich wie ein

Morsecode. Für jede Taste gibt es sozusagen ein unterschiedliches Muster, dass sich aus dem Senden der Frequenz und Pausen zusammensetzt. Ein Zustand dauert dabei nur 13,5 Mikrosekunden, das gesamte Senden eines Signals ist also sehr schnell, und statt von Hand wie bei dem klassischen Morsen wird das hier alles automatisch gemacht. Der Empfänger funktioniert dabei ähnlich. Aber schauen wir uns das kurz mal genauer an: Unser Empfangsgerät (der Fernseher) empfängt nur die Infrarotstrahlen, die genau mit einer Frequenz gesendet werden, nämlich von 38kHz. Erreicht die Frequenz den Fernseher, gibt der Signalpin eine hohe Spannung ab. Dadurch ist es uns mit dem Raspberry Pi oder dem Arduino möglich, das Signal zu messen und wir können auswerten, welche Taste gedrückt wurde.

Also, mit den Grundlagen haben wir uns nun beschäftigt, jetzt geht's ans Eingemachte. Keine Sorge, so schlimm ist es gar nicht. Wir wollen unser Wissen ja nun aber auch praktisch am Raspberry Pi anwenden. Zu aller erst müssen wir dafür ein Programm installieren. Für das gesamte Arbeiten mit dem Raspberry Pi gibt es ein Programm mit dem wunderbaren Namen LIRC, dass es das für uns sehr einfach macht. LIRC steht für Linux Infrared Remote Control und heißt übersetzt: Linux Infrarot Bedienungskontrolle. Doch wie gesagt müssen wir dieses Programm erst einmal installieren:

```
1 | $ sudo apt-get install lirc
```

Bash

```
1 | $ sudo apt-get install python-lirc
```

Bash

Sobald wir damit fertig sind, müssen wir den Raspberry Pi noch richtig konfigurieren. Wenn du den IR Sensor an einem anderen GPIO Pin anschließen willst, musst du das in der folgenden Konfiguration natürlich auch ändern. Als erstes ändern wir die Konfiguration in **/etc/modules**.

```
1 | $ sudo nano /etc/modules
```

Bash

Da musst du die beiden folgenden Zeilen hinzufügen:

```
1 | lirc_dev
```

IR und Schläuer Schalter

```
2 | lirc_rpi gpio_in_pin=17 gpio_out_pin=27
```

Mit Strg+X kannst du das Programm beenden und mit y -> Enter speichern.

Als nächstes müssen wir noch die Konfiguration von LIRC anpassen. Diese befindet sich in **/etc/lirc/hardware.conf**

```
1 | $ sudo nano /etc/lirc/hardware.conf
```

Bash

Veränder dabei die folgenden Zeilen:

```
1 | DRIVER="default"
```

```
1 | DEVICE="/dev/lirc0"
```

```
1 | MODULES="lirc_rpi"
```

Als letztes müssen wir noch für die GPIO Pins folgende Zeile hinzufügen:

```
1 | $ sudo nano /boot/config.txt
```

Bash

```
1 | dtoverlay=lirc-rpi,gpio_in_pin=17,gpio_out_pin=27,gpio
```

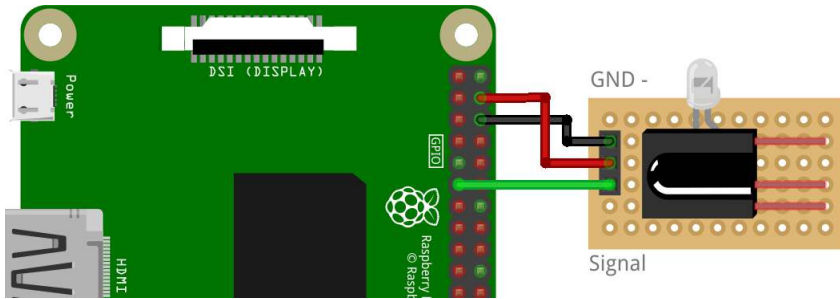
Im letzten Konfigurationsschritt muss der Pi neu gestartet werden, aber danach sind wir dann auch schon fertig! Also:

```
1 | $ sudo reboot
```

Bash

Daten mit Raspberry Pi auslesen

Bevor wir anfangen können, müssen wir zuerst unseren Infrarot Empfänger an unseren Raspberry Pi anschließen.



GPIO Pins am Pi

Anschlüsse IR Empfänger

GND	GND oder auch -
VCC	VCC
GPIO 17	Signal oder auch S

Dazu müssen wir natürlich auch noch die IR Fernbedienung zum Laufen bekommen. Wenn du die beigelegte Fernbedienung benutzen willst, musst du dafür einfach die Knopfzellenbatterie einlegen. Wenn du dann die Kamera App auf einem Smartphone öffnest, solltest du die IR LED erkennen, wenn Tasten gedrückt werden. Dazu kommt, dass der IR Empfänger auch eine eingebaute LED hat, die anfängt zu leuchten, wenn der Empfänger etwas empfängt.

Jetzt sollten wir natürlich noch kontrollieren, ob wir diese Daten auch auf dem Raspberry Pi empfangen. Das können wir mit einem Befehl im Terminal tun. Dafür müssen wir aber zuerst den LIRC Dämon beenden. Dämon steht dabei einfach nur für ein Programm, welches im Hintergrund läuft und Sachen ausführt, in diesem Fall alles, was mit Infrarot zu tun hat.

```
1 | $ sudo /etc/init.d/lirc stop
```

Bash


```
1 | $ mode2 -d /dev/lirc0
```

Bash

Nach dem Ausführen des Befehls und dem Bedienen der IR Fernbedienung solltest du das Drücken der einzelnen Tasten im Terminal sehen. Das sind dann die Werte der einzelnen Tasten. Du kannst das Programm jederzeit mit Strg+C beenden. Doch da das Arbeiten mit diesen kryptischen Werten natürlich etwas anstrengend wäre, gibt es die Möglichkeit mit LIRC, die einzelnen Tasten von IR Fernbedienungen zu hinterlegen. So können wir auch ältere Fernbedienungen benutzen und korrekt konfigurieren. Es gibt dabei eine Vorgabe für die Benennung der Tasten, alle Vorgaben findest du, wenn du diesen Befehl ausführst.

```
1 | $ irrecord --list-namespace
```

Bash

Aber damit du nicht zu lange suchen musst, haben wir hier schon einmal ein paar für dich rausgesucht:

```
KEY_UP . KEY_DOWN . KEY_RIGHT . KEY_LEFT . KEY_1 ,....
```

Mit dem folgenden Befehl startest du das Programm zur Konfiguration und speicherst alle Angaben in der Datei: **lircd.conf**

```
1 | $ irrecord -d /dev/lirc0 ~/lircd.conf
```

Bash

Dazu folgst du einfach nur der Anleitung im Terminal. Zuerst musst du einfach gesagt, wild auf den Tasten herumdrücken, damit das Programm ungefähr weiß, worum es sich handelt. Im nächsten Schritt benennst du dann die einzelnen Tasten und dann immer diese Taste drücken. Wenn du damit fertig bist, kommst du mit einem einfachen Enter-Befehl zum letzten Punkt, bei dem du noch einmal Tasten drücken musst und schon bist du fertig.

Du kannst dir die Datei jetzt ansehen und gegebenenfalls im Kommentarbereich noch einmal konkreter benennen, um was für eine Fernbedienung es sich dabei überhaupt gehandelt hat.

```
1 | $ nano lircd.conf
```

Bash

Jetzt müssen wir diese Konfigurationsdatei noch in den richtigen Ordner kopieren und fertig ist die Tastenbelegung.

```
1 | $ sudo cp lircd.conf /etc/lirc/lircd.conf
```

Bash

Um zu testen, ob die Tasten jetzt auch richtig hinterlegt wurden, müssen wir zuerst den LIRC Dämon starten und das Programm "irw" ausführen.

```
1 | $ sudo /etc/init.d/lirc start
```

Bash

```
1 | $ irw
```

Bash

Wenn du jetzt die Tasten betätigst, solltest du auch die Tasten sehen, die du gerade eben selbst konfiguriert hast. Falls es da zu Problemen kommt, ist am besten, die Konfiguration noch einmal zu wiederholen.

LIRC mit Python

Jetzt haben wir die einzelnen Tasten schon richtig konfiguriert und können so endlich vernünftig anfangen zu programmieren. Die Einbindung in Python ist auch recht einfach. Doch dafür müssen wir noch eine weitere Konfigurationsdatei anlegen. Diesmal in **/etc/lirc/lircrc**

```
1 | $ sudo nano /etc/lirc/lircrc
```

Bash

```
1 |
2 | begin
3 |     button = KEY_DOWN
4 |     prog = slidepuzzle
5 |     config = down
6 | end
```

```

7 | begin
8 |     button = KEY_UP
9 |     prog = slidepuzzle
10 |    config = up
11 | end
12 | begin
13 |     button = KEY_LEFT
14 |     prog = slidepuzzle
15 |     config = left
16 | end
17 | begin
18 |     button = KEY_RIGHT
19 |     prog = slidepuzzle
20 |     config = right
21 | end

```

Wenn bei dir später die Fehlermeldung *lirc.IntiError: unable to load default config [...]* auftritt, hast du vermutlich in dieser Datei ein Formatierungsfehler. Das kann vor allem beim reinen Kopieren der Datei passieren.

Warum die Datei genauso aufgebaut sein muss, erklärt sich am besten, wenn wir gleich auch das Python schreiben und ausführen.

test.py

```

1 | import lirc
2 | sockid = lirc.init("sidepuzzle")
3 |
4 | while True:
5 |     codeIR = lirc.nextcode()
6 |     if "up" in codeIR:
7 |         print("UP")
8 |     print codeIR

```

Python

```

1 | $ sudo python test.py

```

Bash

Wenn du das Programm jetzt ausführst und die Tasten auf der

→ cw42.de/ir/10

Fernbedienung drückst, sollten jetzt die Ausgaben zu sehen sein (vorausgesetzt du hast auch genau die Tasten konfiguriert). Bei der Konfiguration steht `button` für den Namen des hinterlegten Buttons, `prog` für den Namen des Programms. Dieser wird auch als Parameter bei `lirc.init()` übergeben. Mit `confi` wird der Befehl hinterlegt, welcher nachher im Programm ausgegeben wird.

Da das aber natürlich etwas langweilig ist, gibt es noch ein etwas größeres Python Programm, welches wir noch einmal vorstellen wollen. Da das Programm eine grafische Oberfläche hat, musst du dafür auch den Pi über ein Display erreichen. Das Programm kannst du dir einfach mit dem folgenden Befehl herunterladen und ausführen

```
1 | $ wget cw42.de/p/slidepuzzle.py
```

Bash

```
1 | $ sudo python slidepuzzle
```

Bash

Das Spiel ist recht simpel. Es geht darum, dass wir in einem Puzzle die Zahlen anzuordnen. Die Bewegung erfolgt über die Pfeiltasten. Wenn du dich an die obige Anleitung zur Einrichtung gehalten hast, sollte alles ohne Probleme funktionieren.

Der Servo Motor

Im nächsten Teil wollen wir uns mit dem Servo Motor beschäftigen. Da es sich hierbei um einen Motor handelt, können wir mit dessen Hilfe verschiedene Dinge in der wirklichen, physischen Welt bewegen und ausrichten. Eine Besonderheit ist, dass wir bei diesem Motor genau die Position festlegen können, auf die sich der Motor bewegen soll. Bei DC Motoren gibt es z.B. den großen Nachteil, dass wir diese nur "anschalten" können, sie aber nicht wirklich auf eine Position ausrichten können. Ein Nachteil ist an Servo Motoren ist, dass sich die meisten nicht frei über 360° bewegen können, sondern nur einen Winkel von 180° haben. Beim Servo Motor gibt es insgesamt nur drei Anschlüsse. Das rote und braune Kabel dienen beide für die Stromversorgung. Mit dem orangefarbenen Kabel können wir Daten auf den Servo übertragen und damit die Ausrichtung des