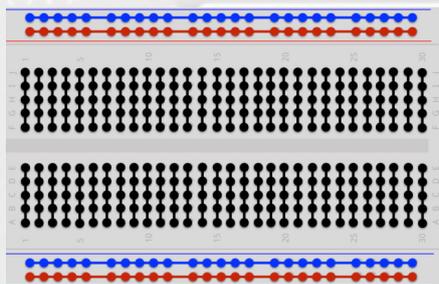


Den Raspberry Pi selbst bestücken - ohne SPIT_GPIO Platine

Für die folgenden Erläuterungen sollten Sie etwas Erfahrung mit dem RaspberryPi und etwas Bastel-Geschick haben. Es kann auch nicht schaden, wenn Sie sich ein klein wenig mit Elektronik auskennen.

ACHTUNG: Wenn Sie die Pins des RaspberryPi kurzschließen, kann der RaspberryPi dadurch zerstört werden. Der Autor übernimmt hierfür keine Haftung!

Für das Ausprobieren empfiehlt sich die Anschaffung eines Breadboard:



Im Internet finden Sie Anbieter für Breadboards, passende Kabel, Widerstände usw. Am besten kaufen Sie ein Breadboard-Kit, da sind passende Kabel und weiteres Zubehör meist enthalten.

Mit Hilfe des Breadboards können Sie, ohne Lötarbeiten, einfache Test-Schaltungen aufbauen, bzw. verdrahten. Entsprechend des obigen Bildes sind die Reihen und Spalten des Breadboard intern verbunden.

Im Folgenden wird das Anschließen von Hardware nur knapp erklärt, im Internet finden Sie weiterführende Anleitungen.

Das fertige SPIT_GPIO Image

Laden Sie sich die Image-Datei Raspberry_SPIT_GPIO.img herunter, z.B. auf <https://www.liveshowsoftware.de/index.php/raspberry.html>. Schreiben Sie das Image auf eine SD-Karte, z.B. mit [BalenaEtcher](#), [Win32DiskImager](#) oder einem anderen Image-Programm. Die SD-Karte sollte mindestens 8GB Speicherplatz besitzen.

Dann stecken Sie die SD-Karte in den RaspberryPi und starten diesen. Beim ersten Start wird das Filesystem automatisch auf die SD-Kartengröße erweitert. Das Image beinhaltet die SPIT_GPIO-Software, die liveSHOW_MIDI Software und es sind diverse Einstellungen schon getroffen (VNC ist aktiviert, es ist eine Bildschirmtastatur installiert, ..).

Lesen Sie bei dem Kapitel: 'Die Bedienung des RasperrPi' weiter!

Händische Installation der SPIT_GPIO Software

Wenn Sie nicht das fertige SPIT_GPIO-Image oder eine fertig vorbereitete SPIT_GPIO SD-Karte besitzen, können Sie die SPIT_GPIO Software auch selbst installieren.

Schritt 1:

Kopieren Sie die Datei SPIT_GPIO.zip auf ihren Rechner und entpacken Sie diese. Z.B. unter Windows klicken Sie mit der rechten Maustaste auf die Datei und

wählen 'Alle extrahieren' aus.

Schritt 2:

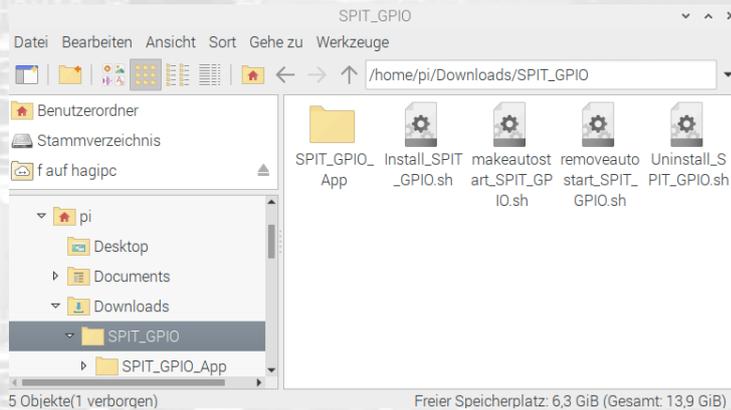
Den entpackten Ordner SPIT_GPIO kopieren Sie auf einen USB-Stick. Entfernen Sie den USB Stick und schließen diesen am RaspberryPi an.

Schritt 3:

Öffnen Sie am RaspberryPi den Dateimanager



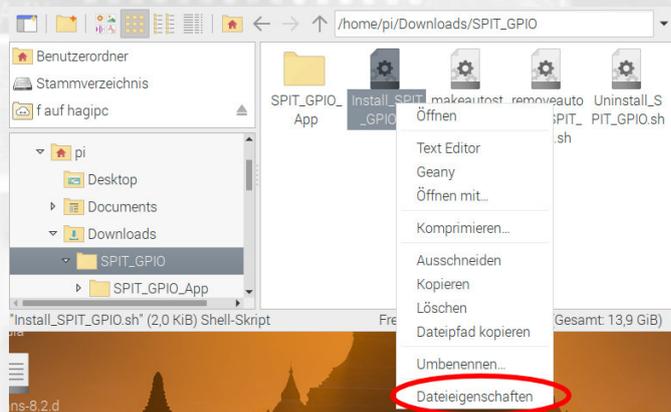
und kopieren Sie den Ordner SPIT_GPIO vom USB-Stick z.B. in das Download-Verzeichnis des RaspberryPi.



Schritt 4:

Öffnen Sie den Ordner SPIT_GPIO im Download-Verzeichnis auf dem RaspberryPi.

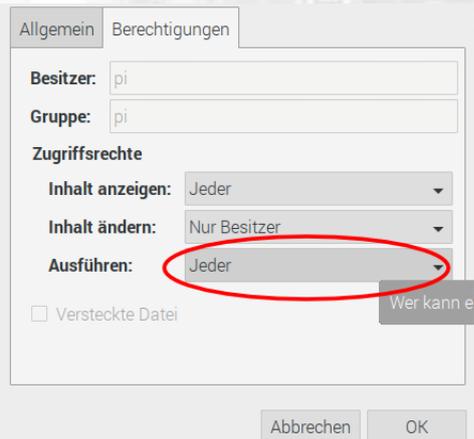
Klicken Sie mit der rechten Maustaste auf 'Install_SPIT_GPIO.sh' und dann mit der linken Maustaste auf Dateieigenschaften.



Schritt 5:

Es öffnet sich das Dateieigenschaften-Fenster, dort klicken Sie auf den Karteikartenreiter 'Berechtigungen' und wählen für 'Ausführen' den Eintrag 'Jeder' aus.

Dann können Sie das Dateieigenschaften-Fenster mit OK schließen.



Schritt 6:
Klicken Sie mit der linken Maustaste doppelt auf die Datei 'Install_SPIT_:GPIO.sh'. Es öffnet sich ein Fenster, dort klicken Sie auf 'Ausführen'

Jetzt ist die Software installiert.

Die Software wird in folgendes Verzeichnis installiert:

/home/pi/.local/share/liveSHOWApps/SPIT_GPIO_App

Wenn Sie dies das erste Mal machen, müssen Sie noch auf 'makeautostart_SPIT_GPIO.sh' doppelt klicken und dann wieder 'Ausführen' anklicken. Hiermit wird die SPIT_GPIO Software dem Autostart hinzugefügt, die Software startet beim Hochfahren des RaspberryPi automatisch.

Eventuell fehlen ein paar notwendige Programmbibliotheken, die die SPIT_GPIO Software benötigt. Diese können Sie folgendermaßen installieren:

Schritt 1:
Der Raspberry Pi benötigt eine Internetverbindung. Verbinden Sie den RaspberryPi mit einem Accesspoint, über den Sie einen Internetzugang haben.

Schritt 2:
Öffnen Sie das Terminalfenster



Dort geben Sie folgendes ein:

sudo apt-get install libqtwebkit4

dann drücken Sie die Return/Eingabetaste. Während des Installationsvorgangs werden Sie eventuell gefragt, ob Sie die entsprechenden Dateien installieren wollen, dies bestätigen Sie durch Drücken der Taste `j`.

WiringPi ist im Raspbian schon enthalten. Für den RaspberryPi 4 gibt es jedoch

eine notwendige Erweiterung auf WiringPi 2.52. Um diese zu installieren geben Sie im Terminalfenster folgendes ein und drücken nach jeder Zeile die Return-Taste:

```
cd /tmp
```

```
wget https://project-downloads.drogon.net/wiringpi-latest.deb
```

```
sudo dpkg -i wiringpi-latest.deb
```

Mit `gpio -v` können Sie kontrollieren, ob Sie die Version 2.52 installiert haben.

Die Bedienung des RaspberryPi

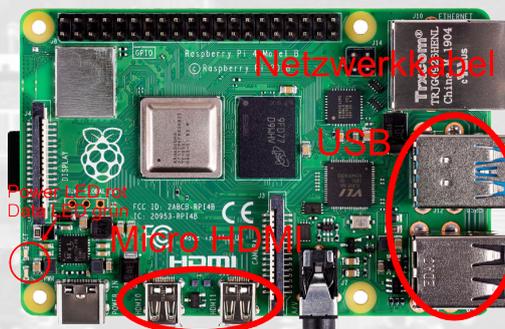
Um den RaspberryPi zu bedienen gibt es zwei Möglichkeiten:

1. Man schließt einen Monitor (HDMI) und eine USB-Tastatur/Maus an
2. Man steuert den Raspberry Pi mit einem anderen PC (via VNC) über ein Netzwerkkabel oder über eine WLAN-Verbindung.

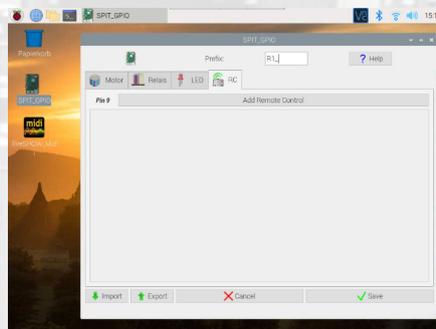
Ad 1.) Monitor und USB-Tastatur/Maus

An einem der HDMI-Ausgänge des RasperryPi schließen Sie einen Monitor/ Bildschirm mittels eines passenden HDMI-Kabel an.

An einen der USB Ausgänge schließen Sie eine USB-Tastatur/Maus an.



Wenn der Raspberry Pi startet sehen Sie die grafische Oberfläche des RaspberryPi.



Ad 2.) Fernbedienung des RaspberryPi über VNC

Da ja beim ersten Start eine Verbindung zu einem Accesspoint noch nicht auf der

SD-Karte gespeichert ist, können Sie versuchen den RaspberryPi über ein Netzwerkkabel mit einem PC zu verbinden.
Sollten die folgenden Anweisungen nicht funktionieren, müssen Sie an den RasperryPi einmal einen Monitor und eine USB-Tastatur/Maus anschließen.

Hinweis: Wenn Sie auf dem RaspberryPi eine WLAN Verbindung einstellen, können Sie die VNC Verbindung auch über das WLAN herstellen.

Schritt 1: VNCViewer Client-Software auf dem Computer installieren

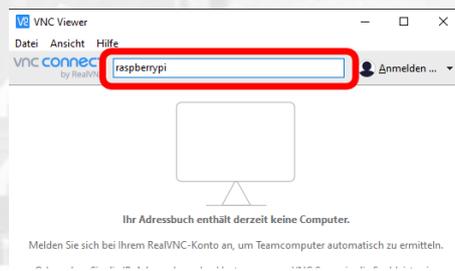
Installieren Sie auf dem PC die 'VNCViewer'-Software, diese gibt es für unterschiedliche Betriebssysteme:

<https://www.realvnc.com/de/connect/download/viewer/>

Schritt 2: Computer mit dem RaspberryPi verbinden

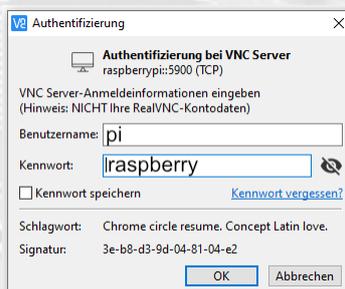
Verbinden Sie den Computer und den RaspberryPi direkt mit dem LAN-Netzwerkkabel.

Auf dem PC starten Sie nun die VNCViewer-Software.



In dem Eingabefeld geben Sie **raspberrypi.local** ein und drücken dann die Return/Eingabe-Taste.

Es müsste sich nach kurzer Zeit das Authentifizierungsfenster öffnen. Es kann sein, dass Ihr Betriebssystem (Firewall, Virens Scanner) ein paar Meldungen macht, die Sie alle positiv bestätigen.



Im Authentifizierungsfenster geben Sie den Benutzernamen: **pi** und das Kennwort: **raspberry** ein und bestätigen dies mit OK

Jetzt müsste sich ein Fenster mit der Benutzeroberfläche des Raspberry Pi öffnen. Sie können nun mit der Tastatur und Maus des PC den Raspberry Pi bedienen.

Hinweis: Die VNCViewer Client-Software speichert die letzten Verbindungen. Dies kann manchmal zu Verbindungsproblemen führen. Löschen Sie einfach die gespeicherten Verbindungen und geben Sie wieder oben im Eingabefeld raspberrypi.local ein.

Das Ändern der Ini-Datei

Die SPIT_GPIO-Software wird beim Start durch eine ini-Datei initialisiert.

Im Ordner /home/pi/.local/share/liveSHOWApps/SPIT_GPIO_App finden Sie die Datei 'SPIT_GPIO_Init0.ini'.

*Hinweis: Es werden drei Pins verwendet, um die PlatinenTyp (Boardnummer) zu ermitteln.
Die Boardnummer wird verwendet um die richtige SPIT_GPIO_InitX.ini zu ermitteln.*

*Boardnummer = Pin29 * 4 + Pin28 * 2 + Pin25.
Falls ein neuer Platinentyp X gebaut wird, sollten die Pins (WiringPi-Nummerierung 25, 28, 29) fest verdrahtet werden und es muss eine neue Datei SPIT_GPIO_InitX.ini angelegt werden Für die erste Platine ist die Boardnummer 0 vorgesehen.*

Hinweis: Ordner und Dateien mit einem Punkt vor dem Namen sind versteckte Ordner/Dateien, diese werden normalerweise im Dateieexplorer nicht angezeigt. Im Dateieexplorer des RaspberrypiS gibt es unter dem Menü 'Ansicht' einen Punkt 'Versteckte Dateien anzeigen'.

Diese ini-Datei können Sie mit dem Texteditor öffnen, indem Sie im Dateieexplorer mit der rechten Maustaste auf die Datei klicken und dann 'Texteditor' wählen. Nachdem Sie Änderungen vorgenommen haben, müssen Sie die Datei speichern.

Anmerkung: Alternativ können Sie auch die 'SPIT_GPIO_InitX.ini' in dem Installations-Ordner ändern. Somit wird bei einer erneuten Installation auch die geänderte ini-Datei mit installiert!

Sie können in der ini-Datei einstellen, was Sie an Hardware installieren wollen und auf welchen GPIO-Pins die entsprechende Hardware angeschlossen werden soll oder angeschlossen ist. Die in der ini-Datei angegebenen Pin-Nummern sind die der **WiringPi Nummerierung**.



Es gibt fünf unterschiedliche Klassen an Hardware:

- GPIO_LED: Hier können Sie LED und kurze LED-Stripes steuern.
- GPIO_Motor: Hier können Sie Modellbau Servo Motoren steuern.
- GPIO_Relais: Hier können Relais angesprochen werden.
- GPIO_RC: Hier können Sie einen 433MHz Empfänger und Sender steuern. Der Empfänger wird zum Sniffen von Funk-Kommandos verwendet. Der Sender kann die erkannten Funk-Kommandos senden.
- GPIO_Input: Hier können Sie Eingangs-Kontakte schließen oder öffnen.

Soll eine Hardware-Klasse nicht verwendet werden, so kommentieren Sie die entsprechende Zeilen einfach aus, indem Sie am Anfang der Zeile zwei Schrägstriche // eingeben. Sollte etwas fehlen, so kopieren Sie eine passende Zeile und ändern diese ab.

Die Pin-Nummern müssen in der WiringPI-Nummerierung angegeben werden - NICHT in der GPIO(BCM)-Nummerierung!

Bei einem erneuten Start der SPIT_GPIO-Software passt sich die Programmoberfläche entsprechend der Einstellungen in der ini-Datei an.

3v3 Power	1			2	5v Power
BCM 2 (WiringPi 8)	3			4	5v Power
BCM 3 (WiringPi 9)	5			6	Ground
BCM 4 (WiringPi 7)	7			8	BCM 14 (WiringPi 15)
Ground	9			10	BCM 15 (WiringPi 16)
BCM 17 (WiringPi 0)	11			12	BCM 18 (WiringPi 1)
BCM 27 (WiringPi 2)	13			14	Ground
BCM 22 (WiringPi 3)	15			16	BCM 23 (WiringPi 4)
3v3 Power	17			18	BCM 24 (WiringPi 5)
BCM 10 (WiringPi 12)	19			20	Ground
BCM 9 (WiringPi 13)	21			22	BCM 25 (WiringPi 6)
BCM 11 (WiringPi 14)	23			24	BCM 8 (WiringPi 10)
Ground	25			26	BCM 7 (WiringPi 11)
BCM 0 (WiringPi 30)	27			28	BCM 1 (WiringPi 31)
BCM 5 (WiringPi 21)	29			30	Ground
BCM 6 (WiringPi 22)	31			32	BCM 12 (WiringPi 26)
BCM 13 (WiringPi 23)	33			34	Ground
BCM 19 (WiringPi 24)	35			36	BCM 16 (WiringPi 27)
BCM 26 (WiringPi 25)	37			38	BCM 20 (WiringPi 28)
Ground	39			40	BCM 21 (WiringPi 29)

Legend

-  GPIO (General Purpose IO)
-  SPI (Serial Peripheral Interface)
-  I²C (Inter-integrated Circuit)
-  UART (Universal Asynchronous Receiver/Transmitter)
-  Ground
-  5V (Power)
-  3.3V (Power)



In der ini-Datei wird die WiringPi Nummerierung verwendet.

SPIT_GPIO_Init0.ini

m Ordner /home/pi/.local/share/liveSHOWApps/SPIT_GPIO_App

Die einzelnen Hardware-Bausteine

Die SPIT_GPIO-Software hat einen Hilfe-Button rechts oben. Wenn Sie diesen Anklicken bekommen Sie die Hilfe zur Bedienung der Software.

Daten zu den GPIO-Pins

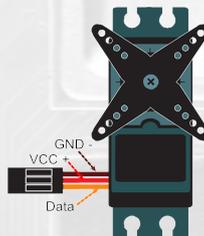
Spannung: Die Spannung an einem GPIO-Pin liegt für 'High' bei 3.3 Volt und für 'Low' bei 0 Volt.

Strom Faustregel: Der Gesamtstrom aller GPIOs sollte 50 mA nicht übersteigen. Der maximale Strom an einem GPIO sollte 16 mA nicht übersteigen.

GPIO_Motor

Modellbau Servo-Motoren drehen sich nicht dauerhaft, sondern können meist eine Stellung zwischen 0 - 180 Grad einnehmen. Diese Stellung kann über eine Datenleitung gesteuert werden.

Modellbau Servo-Motoren haben häufig eine Stromspeisung um die 5 Volt und eine Datenleitung, die von einem Pin des RaspberryPi versorgt werden kann.



Meist sind die Farben der Leitungen folgendermaßen:

Rot = V + (Plus)

Schwarz/Braun = Ground GND - (Minus)

Orange/Gelb = Datenleitung

Bitte vergleichen Sie dies mit der Spezifikation des Motor-Herstellers. Der RaspberryPi hat Spannung-Pins mit 3.3 Volt und 5 Volt. Die GPIO Pins geben +3,3V aus.

Haben Sie einen Servo-Motor mit einer anderen Spannung, als der RaspberryPi liefern kann oder einen Stromverbrauch, der 16 mA übersteigt, so müssen Sie eine externe Stromquelle verwenden. Den Minuspol der externen Stromquelle müssen Sie mit einem Ground GND Pins des RaspberryPi verbinden, sonst hat die Datenleitung kein eindeutiges Potential.

Da ein Servo-Motor beim Anfahren oder bei großer Last relativ viel Strom benötigen kann, macht es grundsätzlich Sinn den Motor an eine extra Stromquelle anzuschließen.

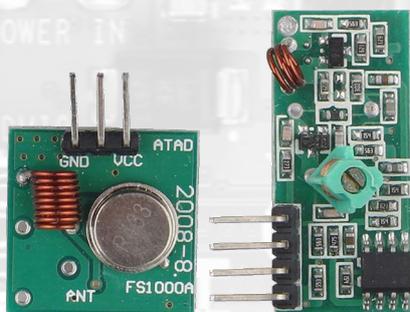
Servo-Motoren werden über eine Art Pulsweitenmodulation (PWM) gesteuert. Wird die PWM per Software erzeugt, so kann der Motor zittern, da eine Software PWM nicht immer genau arbeitet. Der RaspberryPi bietet aber zwei Pins mit Hardware PWM an.

Die Datenleitung schließen Sie an einen der GPIO-Pins des RaspberryPi an. Am besten eignen sich hier der Pin 26 (BCM 12) und für einen zweiten Motor der Pin 23 (BCM 13). Die ini-Datei müssen Sie entsprechend anpassen.

Wenn Sie die Hardware PWM für die Motoren wählen (siehe auch ini-Datei), so sind jetzt die Pins 1 (BCM 18) und 24 (BCM 19) außen vor, diese können Sie nicht mehr für etwas Anderes verwenden, da diese intern mit den Pins 26 und 23 gekoppelt sind.

GPIO_RC (433 MHz)

433 MHz Empfänger und Sender finden Sie über das Internet bei einschlägigen Händlern. Oft sehen diese folgendermaßen aus:



Die Empfänger / Sender haben folgende Anschlüsse:

VCC = meist 5 Volt +
Ground GND = Minuspol -
Data = Datenanschluss diesen kann es auch doppelt geben.

Diese Angaben sind meist auf der Platine aufgedruckt.

Bitte überprüfen Sie dies mit der Spezifikation des Herstellers.

Den VCC-Anschluss verbinden Sie mit einem der 5 Volt-Pins und den Ground GND-Anschluss verbinden Sie mit einem Ground-GND Pin des RaspberryPi.

Den Data-Anschluss verbinden Sie mit einem der GPIO-Pins des RaspberryPi.

Dies machen Sie mit dem Empfänger und dem Sender.

Der Empfänger wird zur Erkennung (Sniffen) von Funk-Kommandos verwendet. Diese erkannten Funk-Kommandos können Sie mit der SPIT_GPIO-Software speichern und mittels des Senders senden.

Jetzt müssen Sie noch die ini-Datei der SPIT_GPIO Software anpassen (GPIO_RC), also den Empfänger-Pin und den Sender-Pin abändern.

Hinweis: Für das Auslesen eines Funksignals sollten Sie den Sender sehr nahe an den Empfänger halten. Es werden viele Funksignale erkannt. Manche Hersteller verwenden jedoch eigene oder verschlüsselte Funkprotokolle, die nicht in der SPIT_GPIO-Software enthalten sind. Was genau erkannt wird, muss einfach ausprobiert werden.

GPIO_LED

Sie können einzelne LED direkt an einen Ground GND-Pin und einen GPIO-Pin des RaspberryPi anschließen, sie sollten jedoch einen Widerstand in Reihe (dazwischen) schalten, der Widerstand wird benötigt, damit die LED nicht zuviel Spannung abbekommt.



Der RaspberryPi gibt an einem GPIO-Pin 3.3 Volt aus.

Eine rote LED hat meist folgende Werte:

Spannung 2.3 Volt
Max. Stromfluss 20mA = 0.02A

Die 20mA geben den maximalen Stromfluss an. Die LED wird auch bei 10mA (0.01A) leuchten, daher verwenden wir den niedrigeren Wert.

Dann berechnet sich der Widerstand R folgendermaßen:

$$R = (\text{Spannung}_{\text{RaspberryPi}} - \text{Spannung}_{\text{LED}}) / \text{Stromfluss}_{\text{LED}} = (3.3\text{V} - 2.3\text{V}) / 0.01\text{A} = 100 \text{ Ohm}$$

Eine andere LED mit einer anderen Farbe hat eventuell andere Werte, dies hängt von dem verwendeten Halbleitermaterial der LED ab.

Eine LED hat einen längeren Draht und einen kürzeren Draht, der längere Draht ist der Pluspol der LED und der kürzere Draht ist der Minuspol der LED.

Den Minuspol der LED (kurzer Draht) verbinden Sie mit einem der Ground GND-Pins des RaspberryPi. An den langen Draht der LED schließen Sie den Widerstand an und das andere Ende des Widerstandes verbinden Sie mit einem der GPIO-Pins des RaspberryPi.

Eine LED kann nicht direkt gedimmt werden, sie wird in einem sehr schnellen Rhythmus ein- und ausgeschaltet. Je nachdem, wie lange sie in einer Phase an ist, erscheint Sie für das menschliche Auge heller oder dunkler. Diese Technik nennt sich Pulsweitenmodulation (PWM).

In der SPIT_GPIO-Software wird die PWM über einen Algorithmus erzeugt. Eine PWM, die softwareseitig erzeugt wird, ist nicht immer stabil, aber für die Zwecke hier ist es ausreichend.

Jetzt müssen Sie noch die ini-Datei der SPIT_GPIO-Software anpassen (GPIO_LED), die verwendeten Pins ändern.

Mehrere LEDs in Reihe schalten: Die Spannungen der LED addieren sich hierbei. Man erkennt schon an der Berechnung des Widerstandes, dass man nicht mehrere LED an einem Pin des RaspberryPi anschließen kann:

$$R = (\text{Spannung}_{\text{RaspberryPi}} - (\text{Spannung}_{\text{LED1}} + \text{Spannung}_{\text{LED2}})) / \text{Stromfluss}_{\text{LED}}$$

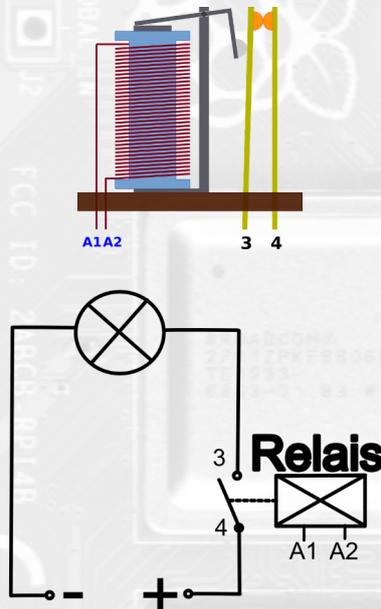
ergibt einen negativen Wert. Und negative Widerstände gibt es nicht.

Mehrere LEDs parallel schalten: Hier benötigt jede LED einen Widerstand. Die Stromstärke addiert sich. Wieviele LEDs verwendet werden können, hängt davon ab, wieviel Ampere ein Pin liefern kann - siehe weiter oben Strom-Faustregel.

Anleitungen für die Verwendung von LED-Stripes mit dem RaspberryPi gibt es im Internet, z.B. <https://dordnung.de/raspberrypi-ledstrip/>.

GPIO_Relais

Ein Relais ist im Prinzip ein Schalter, der durch einen Elektromagneten ein-, bzw. ausgeschaltet werden kann. Somit kann mit einer kleinen elektrischen Leistung ein Stromkreis mit hoher elektrischer Leistung geschaltet werden. Wird im Bild unten ein kleiner Strom an A1 und A2 angeschlossen, so wird der Kontakt zwischen 3 und 4 geschlossen.



Beispiel:

Der Anschluss A1 wird mit einem Ground GND-Pin verbunden, der Anschluss A2 wird einem GPIO-Pin verbunden. Der RaspberryPi liefert 3.3 Volt an den GPIO-Pins. Sie müssen also ein passendes Relais heraussuchen. Dies ist keine Lösung für die Dauer, eigentlich müsste man das Relais mit einer eigenen Spannungsversorgung betreiben!

Relais gibt es in den unterschiedlichsten Spielvarianten auch ohne Elektromagnet, sondern mit Transistoren.

Da es sehr viele unterschiedliche Relais gibt und somit auch viele Fehlerquellen entstehen, wird das Thema hier nicht weiter erörtert. Wer sich auskennt weiß sowieso, was er machen muss.

Sie können aber auch die GPIO_Relais-Technik nutzen um z.B. eine LED ein- bzw. auszuschalten. Die Vorgehensweise ist ähnlich, wie es bei GPIO_LED erklärt wurde, nur dass hier die LED nicht gedimmt werden kann.

GPIO_Input (Eingang)

Wird einer GPIO_Pins mit einem 3,3 Volt-Pin verbunden, so gilt er als geschlossen (connected). Wird die Verbindung wieder getrennt, so gilt der GPIO-Pin als offen (disconnected). Die SPIT_GPIO-Software erkennt dies und man kann in der Software einem Eingang eine Standalone-Aktion und/oder ein Fernsteuer-Befehl (z.B. der liveSHOWsoftware) zuweisen.

In der ini-Datei der SPIT_GPIO-Software können Sie festlegen, welche GPIO-Pins als Eingang verwendet werden sollen. Bitte achten Sie darauf, dass in der ini-Datei die WiringPi-Pinnummer angegeben werden muss.

