



Raspberry Talking-Pi Set

INHALTSVERZEICHNIS

1. Einführung
2. Systeminstallation
3. Installation zusätzlicher Bibliotheken
4. Anschluss eines Lautsprechers
5. Verwendung von 433MHz Modulen
6. Servo/GPIO Anschlüsse
- 7 Drivers
8. Button
9. Gehäuse Zusammenbau
10. Einrichtung der Cloud Speech API
11. Benutzerdefinierte Wake-Words
12. Anlernen eigener Befehle
13. Sicherheitshinweise
14. Informations- und Rücknahmepflicht
15. Support

1. EINFÜHRUNG

Sehr geehrter Kunde,
vielen Dank, dass sie sich für unser Talking-Pi Produkt entschieden haben.
Die eigene Programmierung und Modifizierung des Systems ist ein
wesentlicher und wichtiger Teil dieses Projektes.

In den folgenden Schritten werden wir Ihnen Schritt für Schritt erklären und
beibringen, wie Sie die Google Assistant API einrichten können, ihren Talking-
Pi einsatzbereit machen, das Gehäuse zusammenbauen und wie Sie Ihren
Talking-Pi nach Ihren eigenen Wünschen programmieren und modifizieren
können.

Sollten dennoch, während der Arbeit mit Ihrem Talking-Pi, Probleme
auftreten, so können Sie uns selbstverständlich gerne kontaktieren.

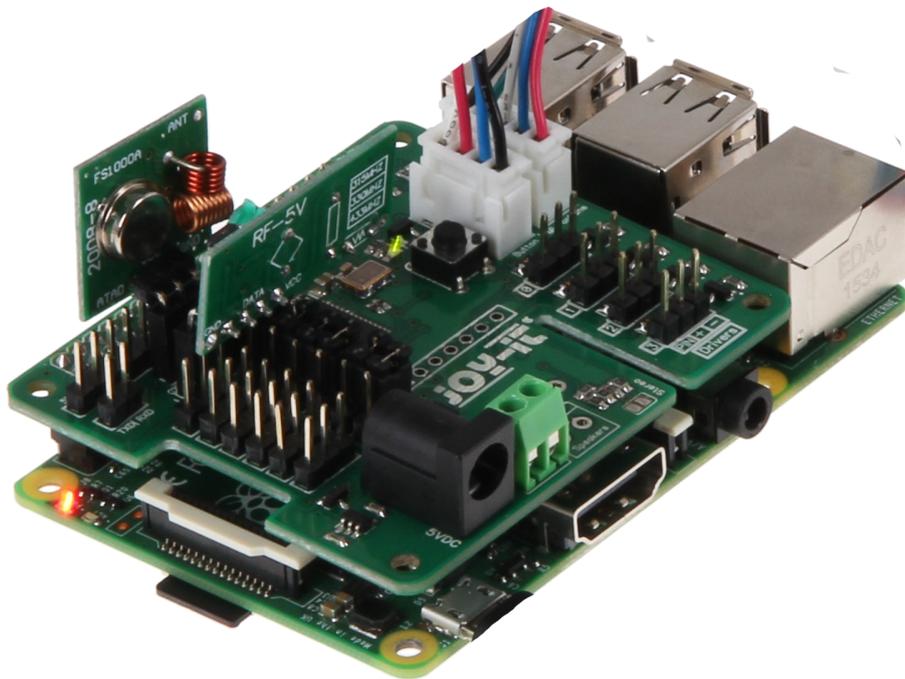
Aktuelle Anleitungen, Projekte und ein Community-Forum ist außerdem hier zu
finden:

[Talking-Pi Website](#)

2. SYSTEMINSTALLATION

Für die erste Systemeinrichtung sind nur wenige Schritte notwendig.
Bitte beachten Sie unbedingt, dass das Talking Pi Set nur mit einem Raspberry Pi 3 kompatibel ist.
Ältere Modelle können das System leider nicht verwenden.

Stecken Sie zuerst die Talking-Pi Erweiterungsplatine auf die GPIO-Leiste ihres Raspberry Pis auf, sodass sich die beiden Platinen übereinander befinden.



2. SYSTEMINSTALLATION

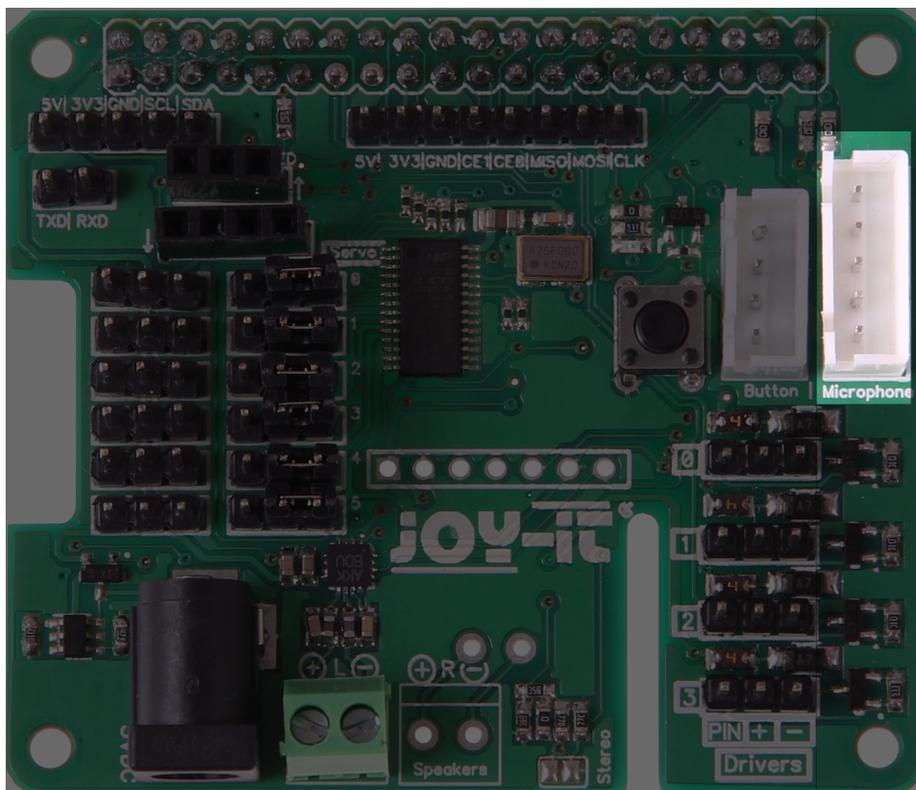
Da das Talking-Pi Set mit dem [Google AIY Project](#) vollständig kompatibel ist, empfiehlt sich die Verwendung des Voice Kit SD Images.

Dieses ist auf der [AIY Projektseite](#) verfügbar.

Laden Sie sich das Image herunter und schreiben Sie dieses mit einem dafür geeigneten Programm (z.B. Etcher) auf Ihre SD-Karte.

Nachdem die Übertragung abgeschlossen ist, können Sie die SD Karte in Ihren Raspberry Pi einlegen.

Sie können nun das beiliegende Mikrofon-Modul mit der Talking-Pi Erweiterungsplatine verbinden.



Das Talking-Pi System ist nun einsatzbereit und kann verwendet werden.

Um das Google Assistant SDK, und damit die Sprachsteuerung, zu aktivieren, sollten Sie der Anleitung auf der [AIY Projektseite](#) folgen.

3. INSTALLATION ZUSÄTZLICHER BIBLIOTHEKEN

Um alle Funktionen der Talking-Pi Erweiterungsplatine nutzen zu können ist die Installation und die Aktivierung von zusätzlichen Modulen bzw. Bibliotheken notwendig.

Folgen Sie den nachfolgenden Schritten, um alle notwendige Erweiterungen zu aktivieren und damit den vollen Funktionsumfang zu ermöglichen.

Denken Sie bitte daran Ihren Raspberry Pi nach der Installation der Bibliotheken neu zu starten.

Übersicht

Verschiedene Module werden für unterschiedliche Anwendungen benötigt.

Falls Sie nur ausgewählte Funktionen benötigen, oder nicht alle Module aktivieren möchten, so finden Sie nachfolgend eine Übersicht die Aufschluss darüber gibt, welche Module für welche Funktionen erforderlich sind.

Funktion	Benötigte Module
433 MHz Module	433Utils wiringPi
Servo Module	Servo-Motor-Erweiterung (modifizierte PCA9685 Bibliothek)
I2C Header	I2C

3. INSTALLATION ZUSÄTZLICHER BIBLIOTHEKEN

I²C

Die I2C-Funktionen sind bereits vorinstalliert und müssen nur noch aktiviert werden.

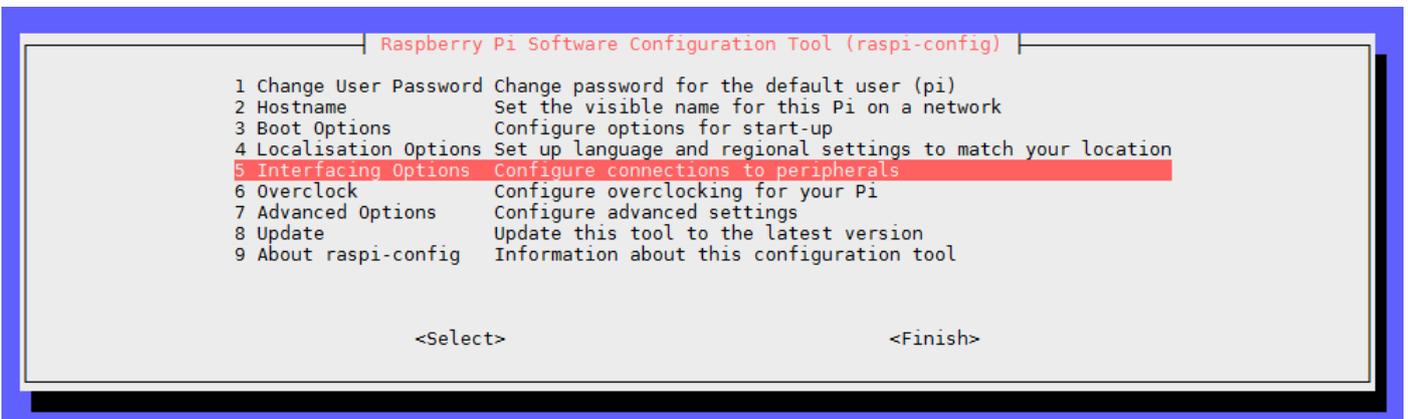
Öffnen Sie auf Ihrem Raspberry Pi ein Terminal Fenster und öffnen Sie das Raspberry-Konfigurationsmenü mit dem folgenden Befehl:

```
sudo raspi-config
```

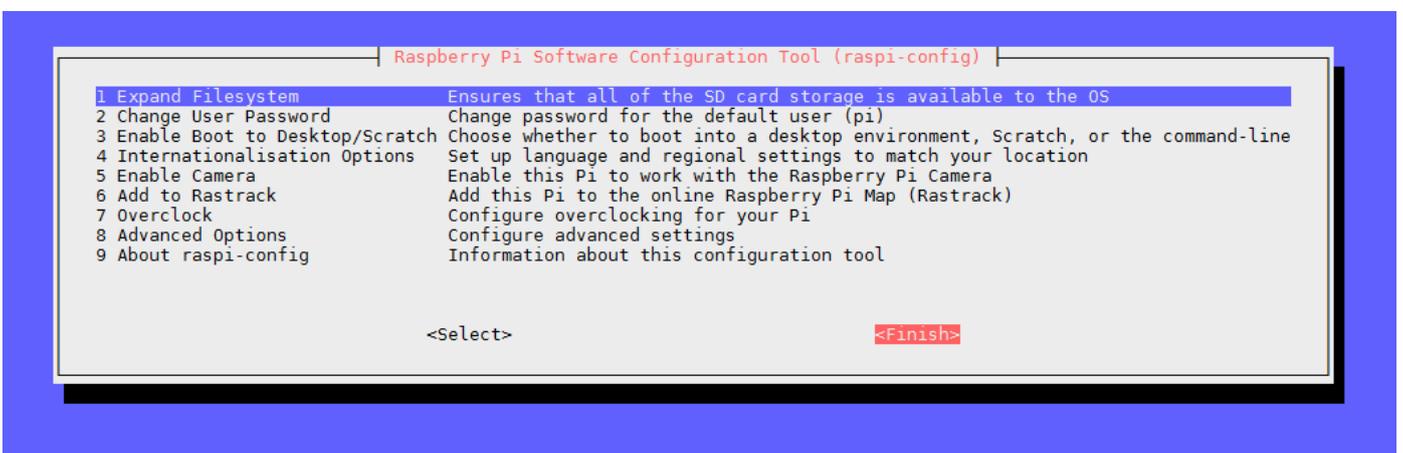
In dem sich nun öffnenden Fenster navigieren Sie in das Menü *Interfacing Options*.

Und aktivieren die **I2C** Option.

Die nächsten sich öffnenden Fenster bestätigen Sie bitte mit *Yes* bzw. *Ok*.



Verlassen Sie das Konfigurationsmenü nun mit *Finish* und starten Sie ihren Raspberry Pi neu.



```
sudo reboot
```

3. INSTALLATION ZUSÄTZLICHER BIBLIOTHEKEN

Servo-Motor Erweiterung

Für die Verwendung der Servomodule ist eine besondere Programmbibliothek erforderlich. Diese basiert auf der Adafruit_PCA9685 Python-Bibliothek, ist allerdings speziell auf unsere Platine angepasst.

Wir empfehlen unbedingt ausschließlich unsere eigene, abgestimmte, Bibliothek zu verwenden. Die angepasste Bibliothek können Sie [hier](#) herunterladen.

Kopieren Sie die entpackte Bibliothek bitte vollständig auf Ihren Raspberry Pi und navigieren Sie im Terminal in diesen Ordner.

Mit dem folgenden Befehl können Sie die Bibliothek dann installieren:

```
sudo python setup.py install
```

WiringPi

Die wiringPi Bibliothek ist notwendig um weitere Module (z.B. die 433MHz Funk-Module) anschließen zu können.

Um diese Bibliothek zu installieren führen Sie bitte die folgenden Befehle in einem Terminal-Fenster aus:

```
sudo apt-get install git git-core
git clone git://git.drogon.net/wiringPi
cd wiringPi
./build
```

3. INSTALLATION ZUSÄTZLICHER BIBLIOTHEKEN

433Utils

Zur Verwendung der Funk-Module ist die Installation einer zusätzlichen Bibliothek erforderlich. Hierfür eignet sich die [433Utils Bibliothek von ninjablocks](#).

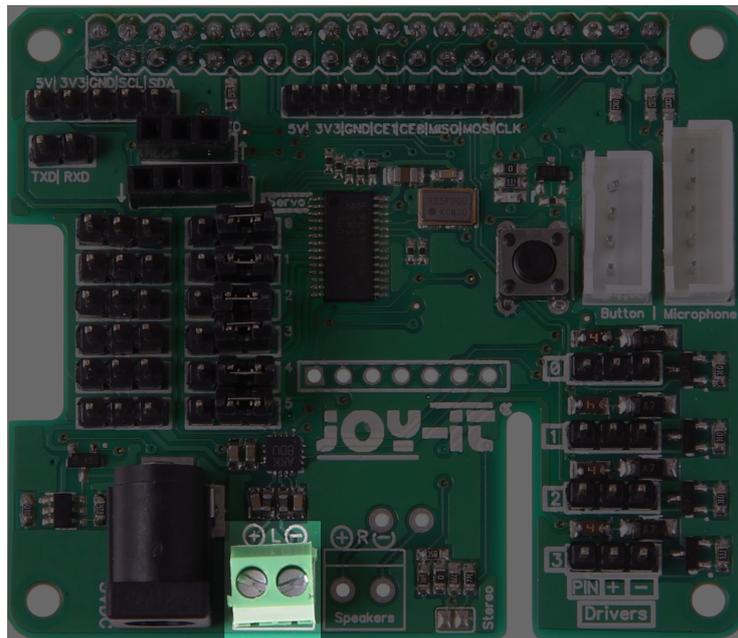
```
git clone https://github.com/ninjablocks/433Utils.git --recursive
cd 433Utils/RPi_utils
make all
```

Diese Bibliothek wurde unter der MIT Lizenz veröffentlicht und ermöglicht die einfache Verwendung von 433MHz Modulen.

4. ANSCHLUSS EINES LAUTSPRECHERS

Das Talking-Pi Modul verfügt über einen vorinstallierten Lautsprecheranschluss. Dieser ist geeignet für einen Lautsprecher mit einer Leistung von **3 Watt**.

Um einen Lautsprecher mit dem Modul zu verbinden, lösen Sie zuerst leicht die gekennzeichneten Schrauben der Schraubklemme des Lautsprecheranschlusses.



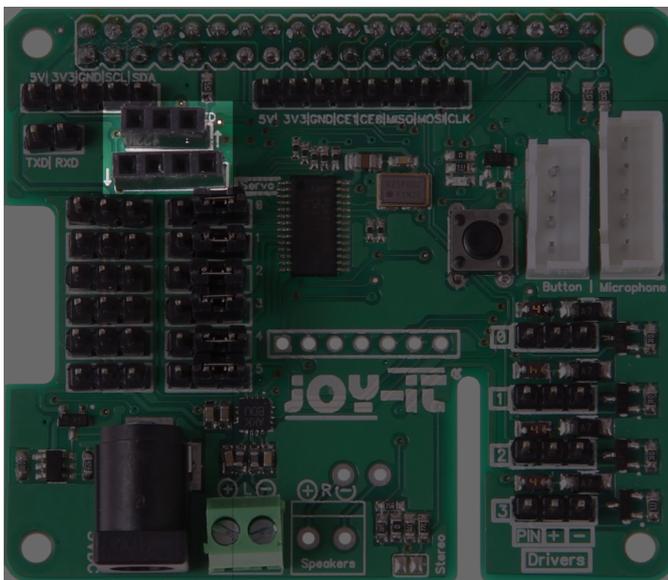
Führen Sie nun das rote Lautsprecherkabel in die linke Öffnung (gekennzeichnet durch das +) und das schwarze Lautsprecherkabel in die rechte Öffnung (gekennzeichnet durch das -) und fixieren Sie diese, indem Sie die oberen Schrauben festziehen.

5. VERWENDUNG VON 433 MHZ MODULEN

Anschluss der Module

Sie können an das Talking-Pi Modul auch Sender und Empfänger für 433 MHz Funk-Module anschließen, um damit Geräte (z.B. Funksteckdosen) zu steuern.

Setzen Sie die Module einfach in die gekennzeichneten Steckplätze (**433 TXD**: 3-Pin Sender, **433 RXD**: 4-Pin Empfänger) ein. Die Module sollten hierbei (erkennbar an den Pfeilen) voneinander weg zeigen.



Achtung! Bitte beachten Sie, dass eine Verwendung der 433MHz Module mit der gleichzeitigen Verwendung der **Driver1** und **Driver2** Anschlüsse nicht möglich ist.

Der Sender verweist auf den GPIO Port 17 und der Empfänger auf den GPIO Port 27. In der Regel bedarf es hier aber keiner weiteren Konfiguration.

5. VERWENDUNG VON 433 MHZ MODULEN

Verwendung

Für die Verwendung der 433MHz Funkmodule navigieren Sie bitte in den Unterordner *433Utils/RPi_utils*

Um den Empfänger zu verwenden genügt der folgende Befehl:

Sobald ein Code vom Modul empfangen wird, wird dieser auf dem Display ausgegeben.

```
sudo ./RFSniffer
```

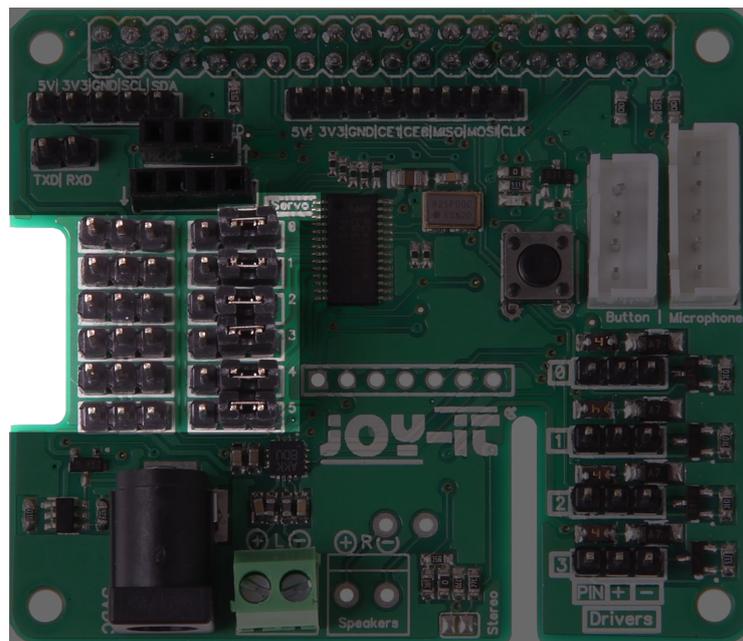
Der Sender kann mit diesem Befehl gestartet werden:

Die Zahl kann hierbei durch einen beliebigen Code ausgetauscht werden.

```
sudo ./codesend 1234
```

6. SERVO/GPIO ANSCHLÜSSE

Servo-Verwendung



Die markierten Anschlüsse sind für die Verwendung von bis zu 6 Servo-Motoren geeignet. Durch die daneben befindlichen Jumper lässt sich der Betrieb zwischen GPIO-Betrieb und Servo-Betrieb wechseln. Verbinden Sie einfach den mittleren und den rechten PIN durch den Jumper um in den Servo-Betrieb zu wechseln.

Bitte beachten Sie, dass für die Verwendung von Servo-Motoren eine zusätzliche Stromquelle benötigt wird. Verbinden Sie hierfür einfach einen Hohlstecker (5.5 / 2.1mm, 4.8 - 6V, 6A max.) in den dafür vorgesehenen Anschluss, neben dem Lautsprecheranschluss.

Um die Servos zu testen, navigieren Sie bitte in den Unterordner *Adafruit_Python_PCA9685/examples*. Hier können Sie das Beispiel mit dem folgenden Befehl starten:

6. SERVO/GPIO ANSCHLÜSSE

Dieses Beispiel wird den Servo auf Kanal 0 ansteuern.
Sollten Sie einen anderen Kanal verwenden möchten, so können Sie das Beispielprogramm selbstverständlich an Ihre Verwendung anpassen.

Öffnen Sie dieses dazu mit dem folgenden Befehl:

```
sudo nano simpletest.py
```

Hier finden Sie, ganz unten, die beiden Befehle ***set_servo_pulse(0,start_position)*** und ***set_servo_pulse(0,end_position)***.

Die 0 definiert in diesem Fall den angesprochenen Kanal.

Ändern Sie einfach die Kanalnummer oder fügen Sie neue Befehle, mit der passenden Kanalnummer, hinzu.

Ihre Änderungen können Sie mit der Tastenkombination **STRG+C** speichern und den Editor mit der Kombination **STRG+X** verlassen.

GPIO-Verwendung

Sollten Sie die vorhandenen Anschlüsse nicht für Servo-Motoren, sondern für den normalen GPIO-Betrieb verwenden wollen, so können Sie dies selbstverständlich tun.

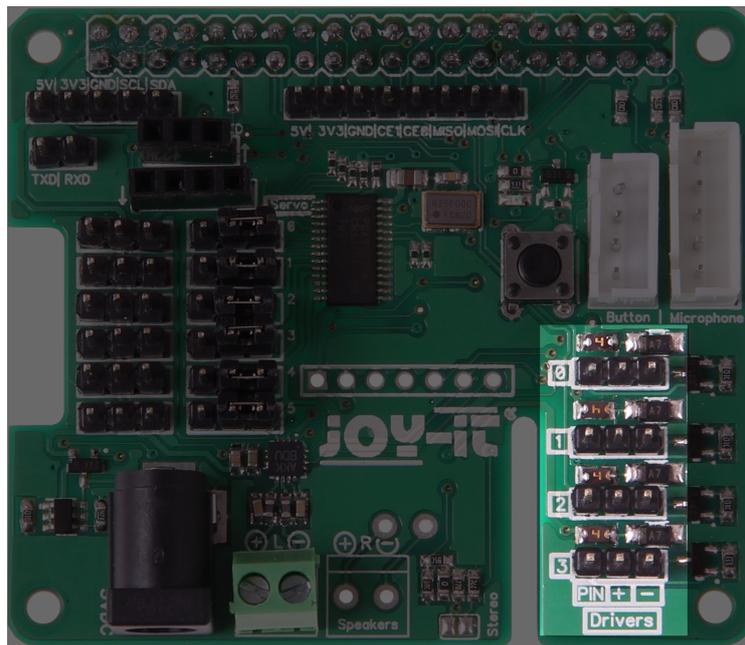
Stecken Sie den entsprechenden Jumper einfach auf den linken und den mittleren PIN auf, um in den GPIO-Modus zu wechseln.

Die einzelnen Kanäle sind mit folgenden GPIO-Nummern verwendbar:

Channel	GPIO
0	GPIO26
1	GPIO06
2	GPIO13
3	GPIO05
4	GPIO12
5	GPIO24

7. DRIVERS

Die Driver-Verbindungen verweisen direkt auf die GPIO-Verbindungen des Raspberry Pis. Neben den Pin-Verbindungen enthält jeder Driver-Anschluss außerdem einen Pin zur Stromversorgung, sowie einen Pin für die Masse-Verbindung.



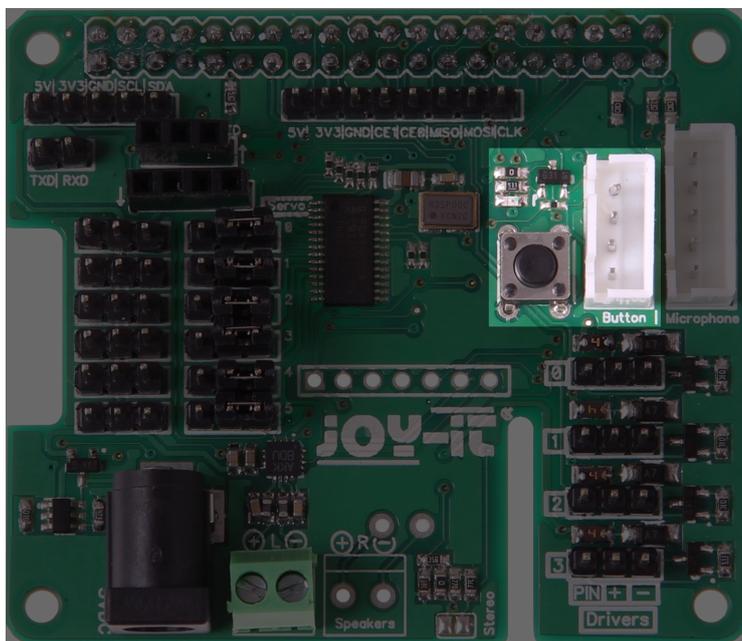
Die Driver verweisen hierbei auf die folgenden GPIO-Verbindungen:

Achtung! Driver1 und Driver2 können bei der Verwendung der 433MHz Funkmodule nicht verwendet werden!

Driver	GPIO
0	GPIO04
1	GPIO17
2	GPIO27
3	GPIO22

8. BUTTON

Die Talking-Pi Erweiterungsplatine verfügt bereits über einen Button. Optional lässt sich dennoch ein Button (z.B. ein Arcade-Button) über den dafür vorgesehenen Anschluss verbinden.



Die Verbindung verweist auf den GPIO Pin 23 und lässt sich somit frei verwenden. Der bereits eingebaute Schalter ist parallel zur optionalen Verbindung geschaltet, sodass beide Möglichkeiten gleichzeitig für den selben Zweck eingesetzt werden können.

9. GEHÄUSE ZUSAMMENBAU

Inhalt

- Gehäuse (Holz 7-teilig)
- Druckknopf mit LED
- Lautsprecher mit Kabel
- Kabelsatz Taster/ LED
- Kabel Mikrofon
- Schraubensatz

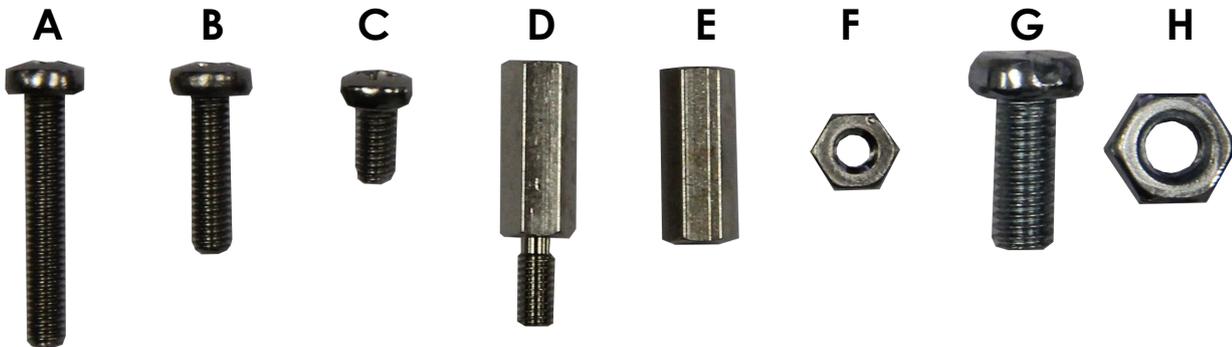


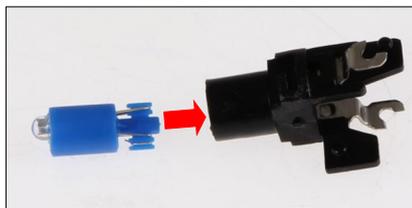
Abbildung	Stückzahl	Bezeichnung
A	4	Schraube M2,5 x 16
B	4	Schraube M2,5 x 10
C	8	Schraube M2,5 x 5
D	2	Abstandhalter M2,5 x 11 mit 1x Innen- & 1x Außengewinde
E	4	Abstandhalter M2,5 x 11 mit 2x Innengewinde
F	20	Mutter M2,5
G	4	Schraube M4 x 10
H	4	Mutter M4
	7	Gehäuseteile aus Holz
	1	Lautsprecher
	2	Kabel

9. GEHÄUSE ZUSAMMENBAU

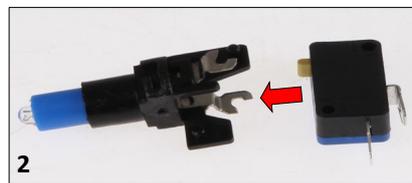
Montage

LED-Taster:

1. LED-Lampe in die Fassung stecken.
2. Mikroschalter einsetzen.



1



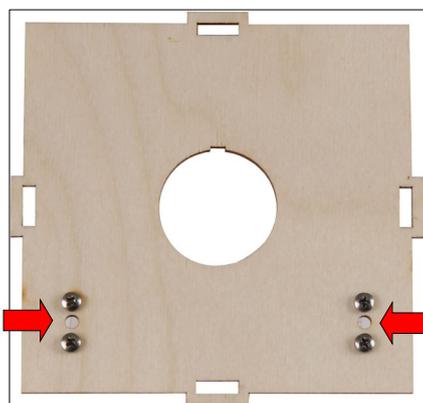
2



3

Mikrofon:

1. Schrauben (4x B) durch das Holzgehäuse stecken
2. und mit jeweils einer Mutter (4x F) verschrauben
3. Mikrofonplatine aufsetzen und diese mit Muttern (4x F) befestigen.



1



2

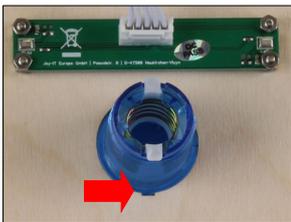


3

9. GEHÄUSE ZUSAMMENBAU

Druckknopf einsetzen

1. Druckknopf einsetzen. Hier auf die Kerbe (Pfeil) achten.
2. Kunststoffring aufsetzen.
3. Knopf mit Kunststoffmutter befestigen
4. LED- und Tastermodul einsetzen und mit einer leichten Drehbewegung einrasten lassen.
5. Darauf achten, dass der Mikroschalter dem weißen Tastkontakt gegenüber ist.



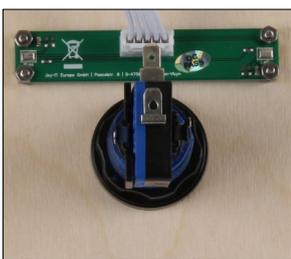
1



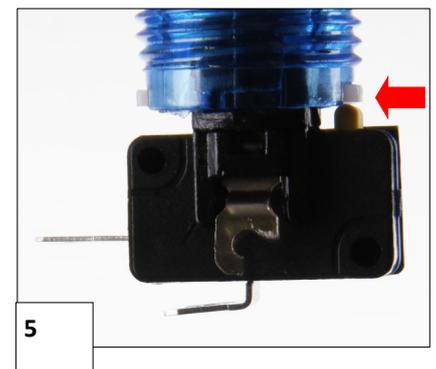
2



3



4

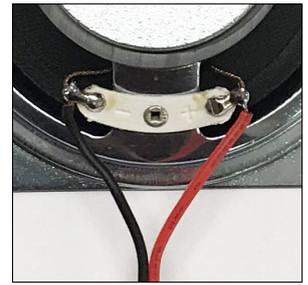


5

9. GEHÄUSE ZUSAMMENBAU

Lautsprecher

Als erstes wird das beiliegende Kabel an den Lautsprecher gelötet. Alles vier Kabelenden sollten verzinnt werden.



1. Holzplatte mit den 7 mittleren Löchern...
2. ... darauf die Distanzplatte legen.
3. Schrauben (4x G) durch beide Platten schieben.
4. Lautsprecher aufsetzen und mit Muttern (4x H) befestigen.



1



2



3

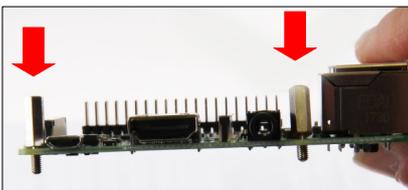


4

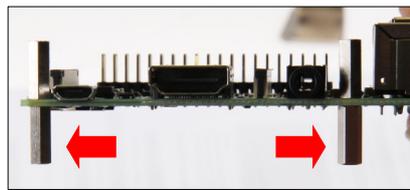
9. GEHÄUSE ZUSAMMENBAU

Raspberry Pi 3B

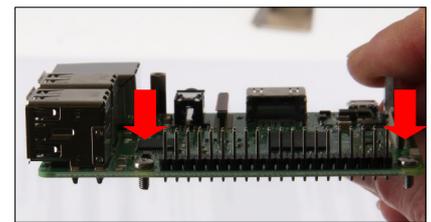
1. Abstandhalter (2x D) von oben einführen.
2. diese mit den Abstandshalter (2x E) auf der Unterseite verschrauben
3. Schrauben (2x C)...
4. ... mit Abstandshalter (2x E) verschrauben.
5. Talking-Pi Platine aufstecken mit verschrauben (2x C).
6. Raspberry-Pi auf die Holzwand auflegen und verschrauben (4x C).



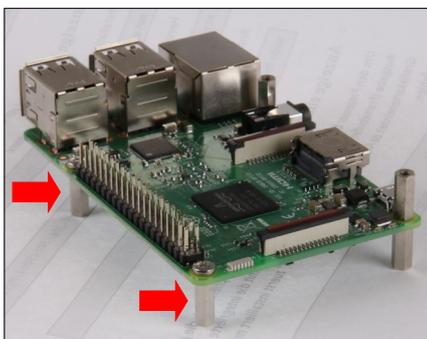
1



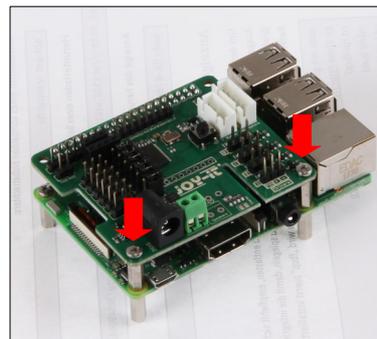
2



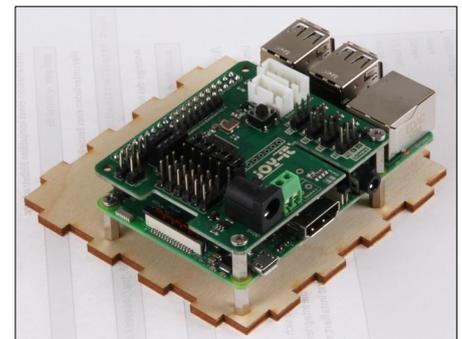
3



4



5



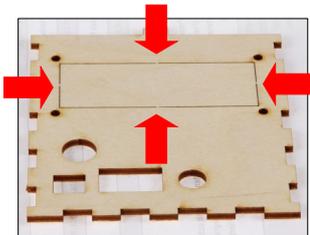
6

9. GEHÄUSE ZUSAMMENBAU

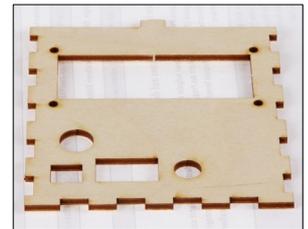
Display

Um das Display einzubauen, muss die davor vorgesehene Öffnung vorsichtig rausgebrochen werden.

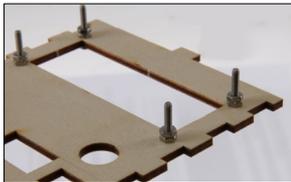
1. Hier mit einem Cutter Messer die Haltepunkte vorsichtig einschneiden...
2. ... und herausbrechen.
3. Schrauben (4x A) durch das Gehäuse und mit jeweils zwei Muttern (8x F) befestigen.
4. Display aufsetzen und mit Muttern (4x F) befestigen.



1



2



3

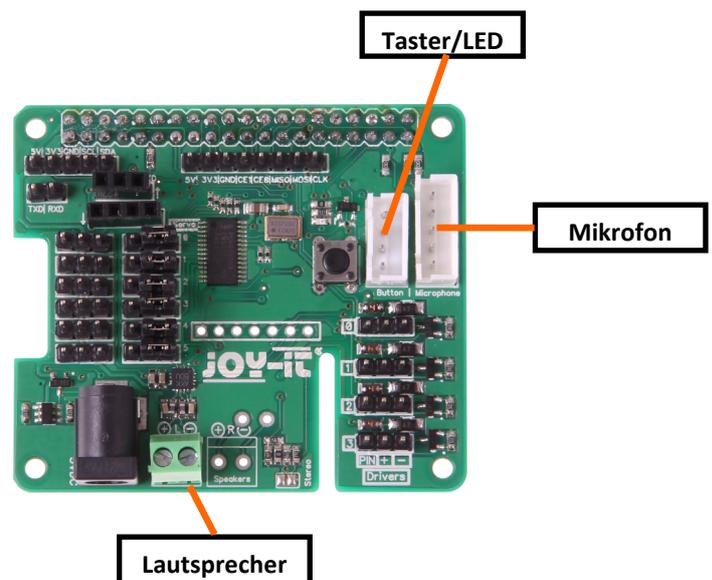
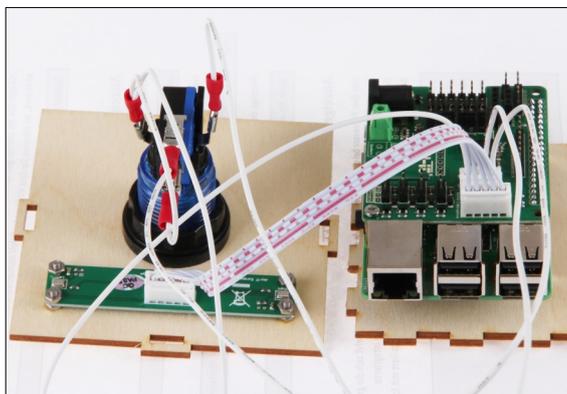
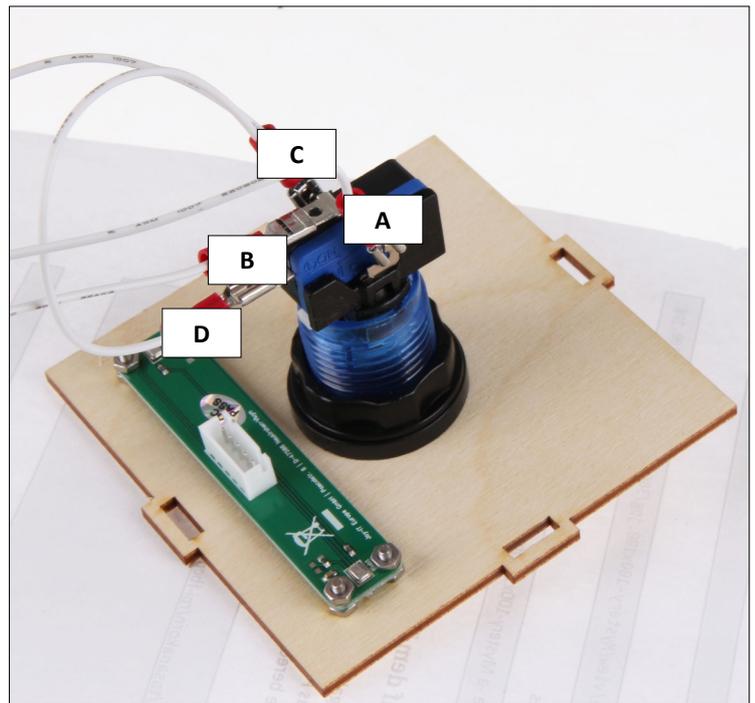
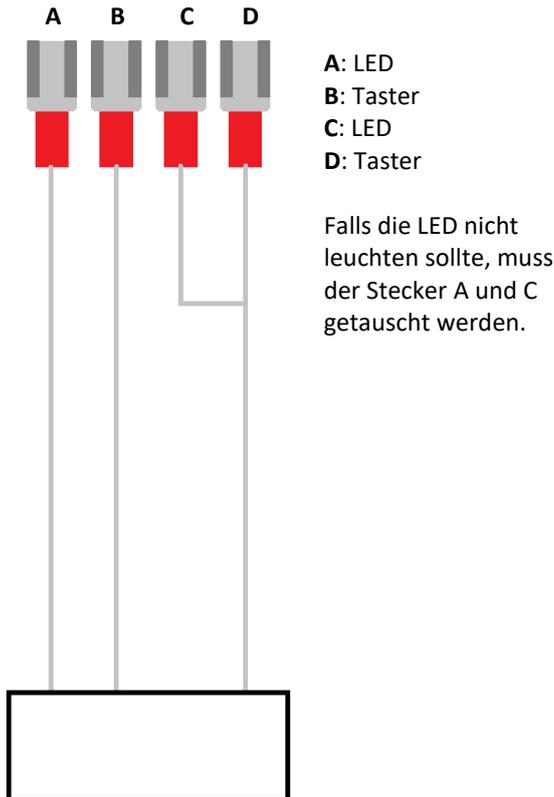


4

9. GEHÄUSE ZUSAMMENBAU

Verkabelung

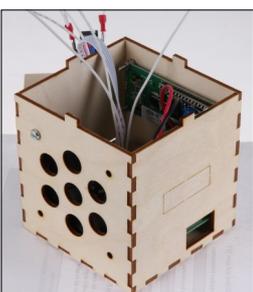
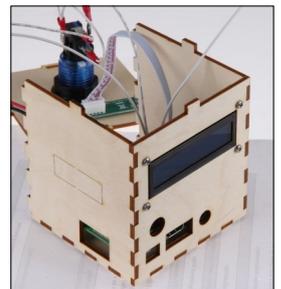
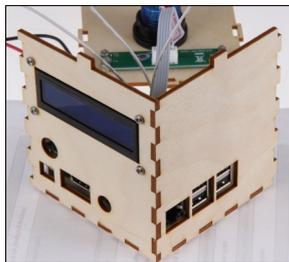
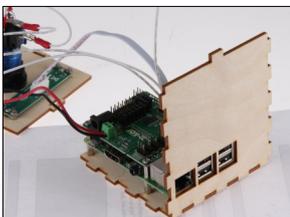
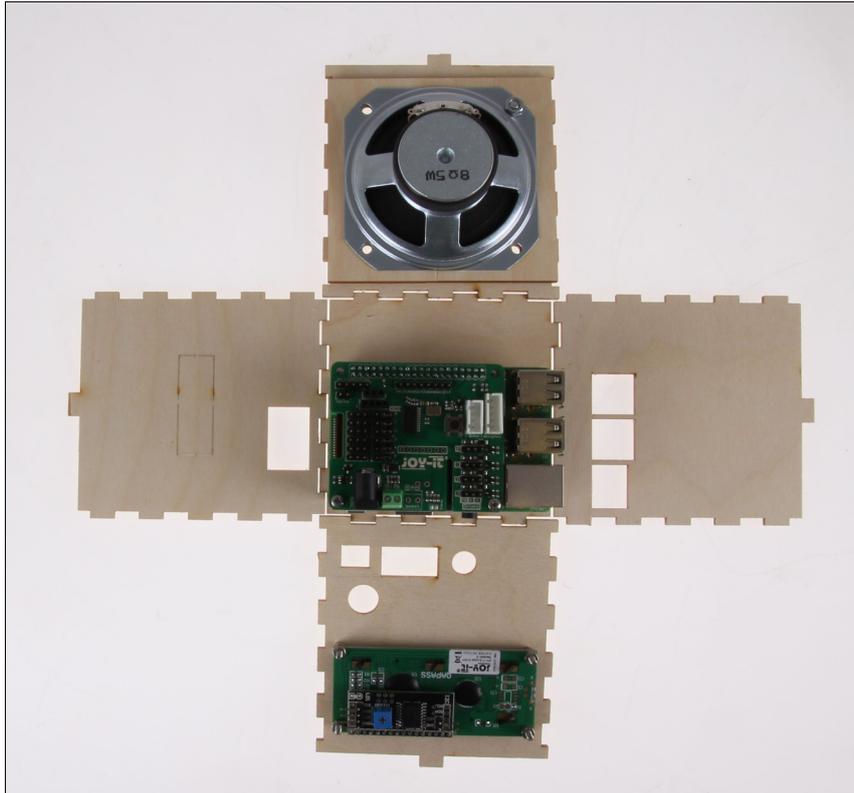
Nun wird das beiliegende Kabel mit dem Taster (LED) und dem Talking Pi (BUTTON) verbunden.
Auch Mikrophon, Lautsprecher und ggf. das Display können verbunden werden.



9. GEHÄUSE ZUSAMMENBAU

Zusammenbau

Nun kann das Gehäuse, wie auf den Bildern zu sehen ist, zusammengesteckt werden.



10. EINRICHTUNG DER CLOUD SPEECH API

Mit unserem vorinstallierten Talking-Pi Betriebssystem wartet der Talking-Pi geduldig darauf aktiviert zu werden.

Das Image können Sie direkt [hier](#) herunterladen.

Installieren Sie das Image auf einer Micro-SD-Karte (eine Kapazität von mindestens 8GB wird empfohlen) und legen Sie diese anschließend in Ihren Raspberry Pi ein.

Mit einer geänderten Cloud Speech API stehen Ihnen 80 Sprachen zur Verfügung.

Außerdem können Sie eigene Wake-Words und Befehle in das System einpflegen.



Achtung! Die Google Cloud Speech API ist kostenfrei, insofern Sie diese weniger als 60 Minuten pro Monat verwenden. Andernfalls wird Ihnen ein Betrag von 0,06\$ pro 15 Sekunden berechnet. Sie erhalten allerdings zuvor eine Benachrichtigung, falls Sie das Limit von 60 Minuten überschreiten sollten.

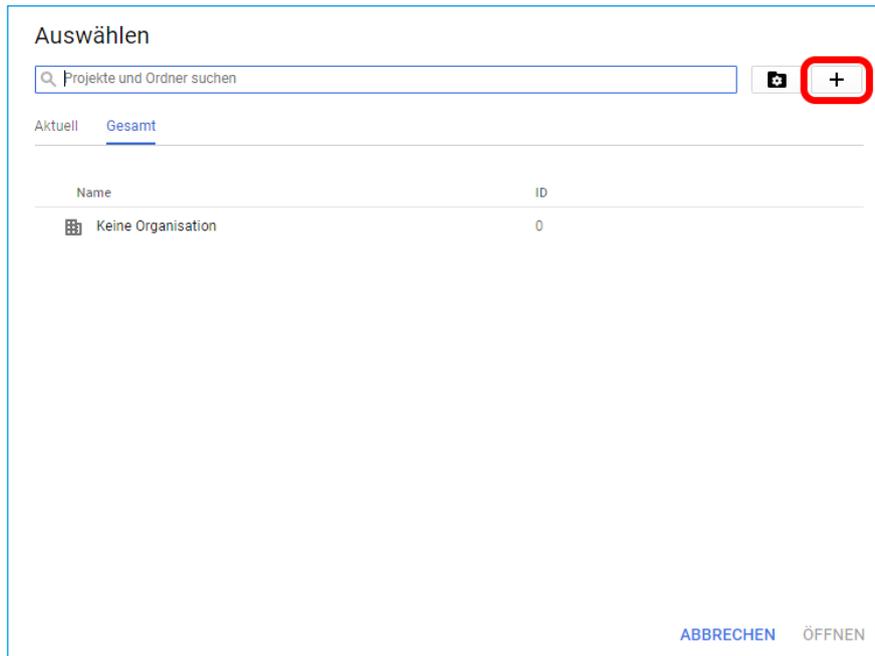
Öffnen Sie zunächst die [Google Cloud Console](#).

Loggen Sie sich mit Ihrem Google-Konto ein oder erstellen Sie ein neues, falls Sie noch keines besitzen.

Öffnen Sie hier die Projektübersicht und erstellen Sie ein neues Projekt.



10. EINRICHTUNG DER CLOUD SPEECH API



Als nächstes wählen Sie aus dem linken Menü die Option „Abrechnung“.

Fügen Sie hier eine neue Bezahlmöglichkeit hinzu und folgen Sie den weiteren Anweisungen.

Kehren Sie nach dem Hinzufügen einer neuen Abrechnungsoption zum Abrechnungs-Menü zurück und stellen Sie sicher, dass Ihr neu erstelltes Projekt mit dieser Zahlungsoption verknüpft ist.

Um Ihr Projekt mit einer Zahlungsoption zur verknüpfen können Sie einfach die drei Punkte anwählen.

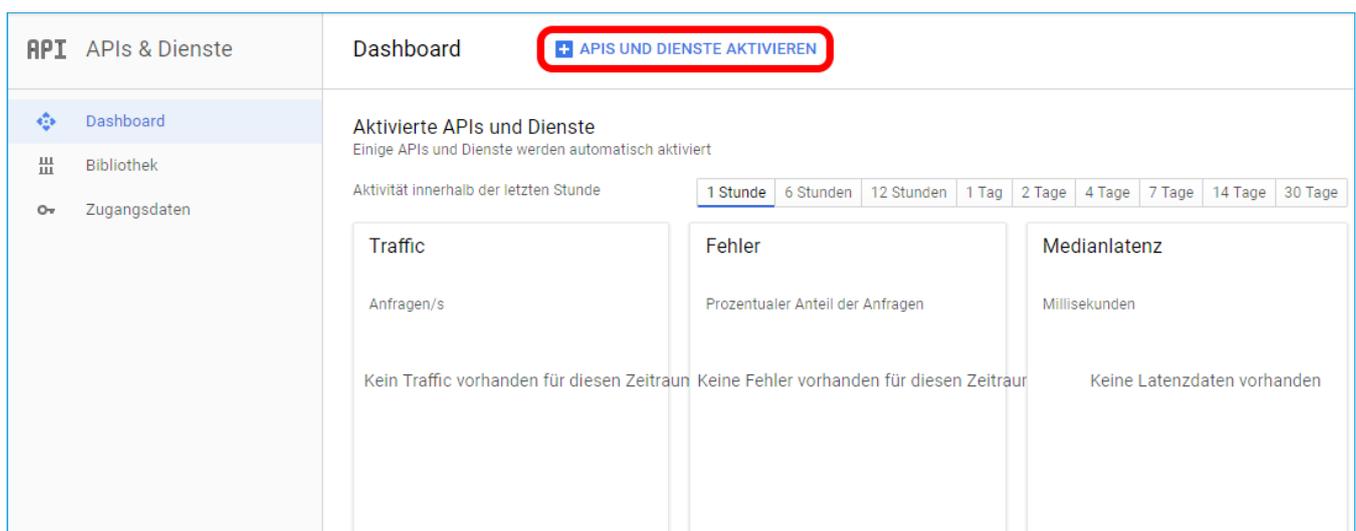
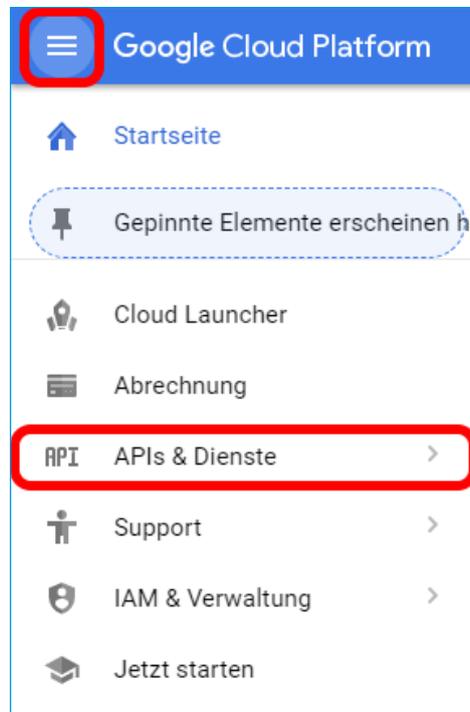
Als nächstes werden die benötigten Dienste aktiviert.

Verwenden Sie hier das im Bild markierte Symbol um ein neues Projekt zu Ihrer Liste hinzuzufügen.

Geben Sie dann einfach einen beliebigen Namen für Ihr Projekt ein und schließen Sie den Vorgang ab.

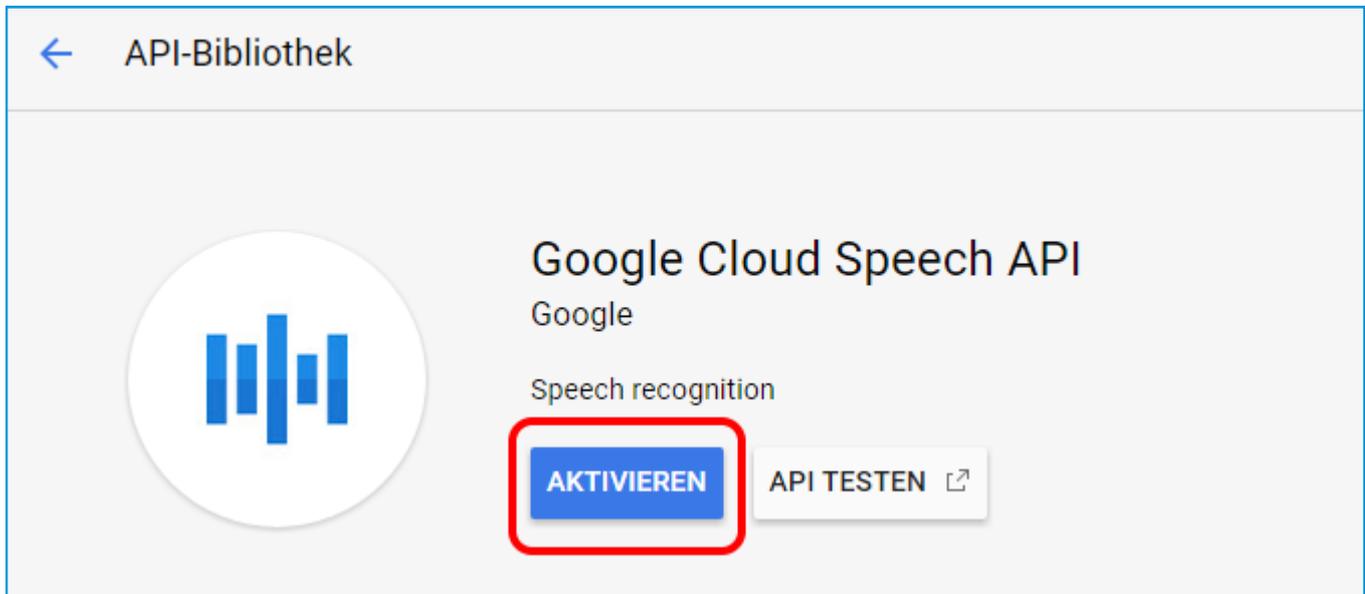
Öffnen Sie dann das Menü und wählen Sie hier die Option „**APIs & Dienste**“ und aktivieren Sie das Menü.

10. EINRICHTUNG DER CLOUD SPEECH API

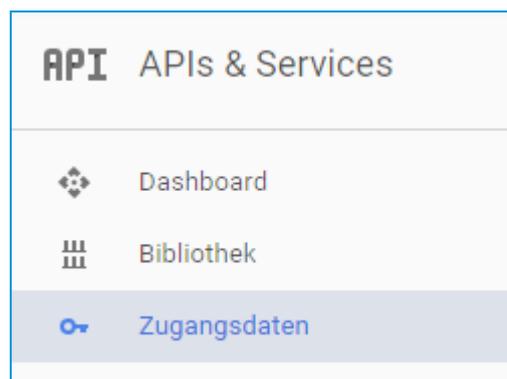


10. EINRICHTUNG DER CLOUD SPEECH API

In der sich nun öffnenden API-Bibliothek suchen Sie nach dem Begriff „**Google Cloud Speech API**“. Wählen Sie die gefundene API aus und aktivieren Sie diese.



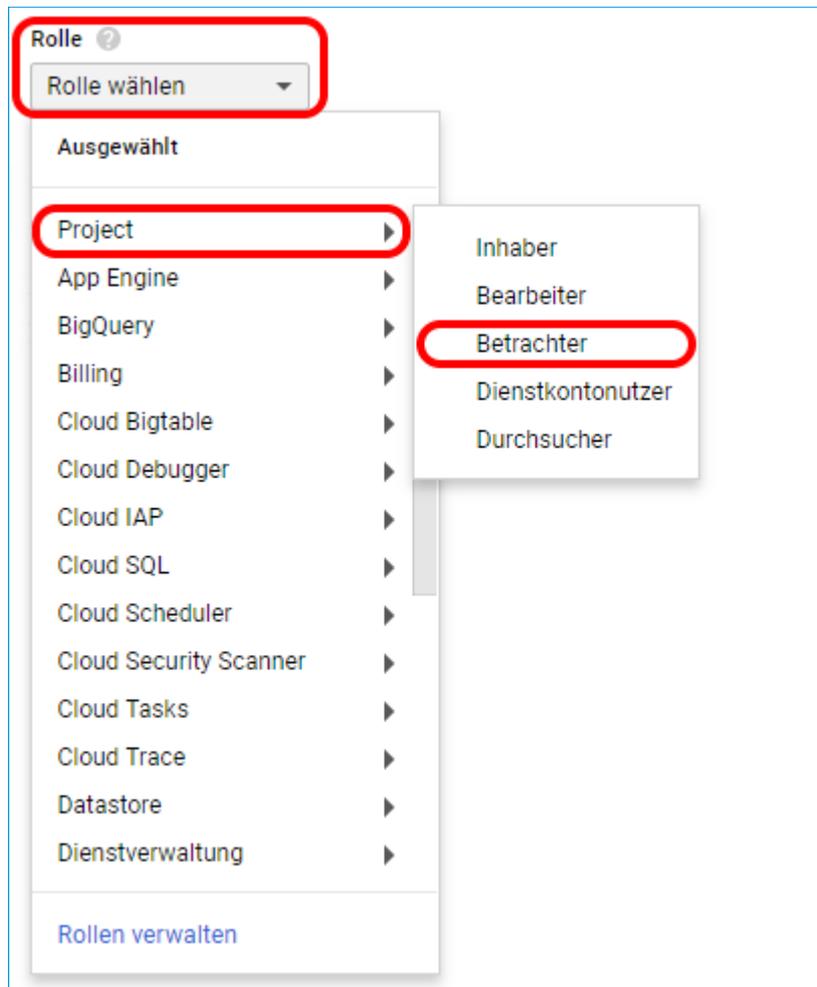
Wählen Sie nun in der „**API & Services**“ Übersicht den Punkt „**Zugangsdaten**“ aus.



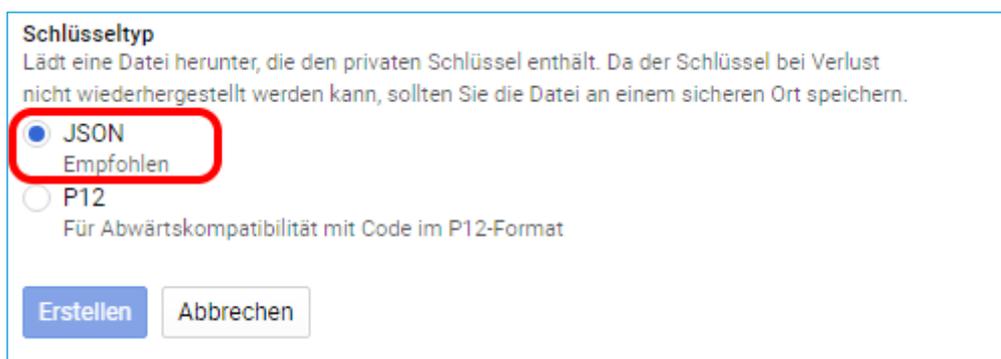
10. EINRICHTUNG DER CLOUD SPEECH API

Wählen Sie hier „Anmeldedaten erstellen“ und erstellen Sie hier einen „Dienstkontoschlüssel“.

Als Rolle definieren Sie hier **Project** und Betrachter:



Als Schlüsseltyp wählen Sie „JSON“ aus.

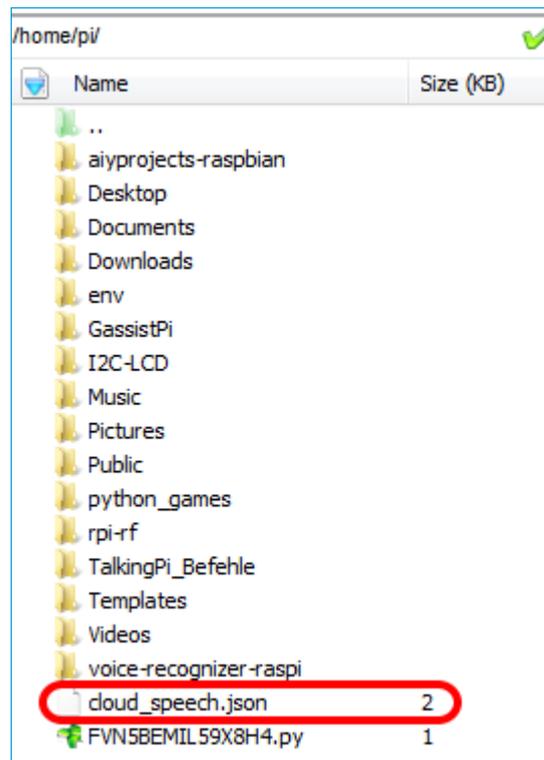


10. EINRICHTUNG DER CLOUD SPEECH API

Erstellen Sie nun Ihre Zugangsdaten, indem Sie auf „Erstellen“ klicken.

Der Download Ihrer Daten startet automatisch.

Nenne Sie die heruntergeladene Datei in **cloud_speech.json** um und kopieren Sie die Datei in das Benutzerverzeichnis Ihres TalkingPis.



Nach einem Neustart ist die Cloud Speech API auf Ihrem TalkingPi einsatzbereit.

11. BENUTZERDEFINIERTER WAKE-WORDS

Die Aktivierung kann entweder durch das Drücken des Knopfes erfolgen oder durch ein so genanntes Wake-Word, welches vorher in das System integriert wurde.

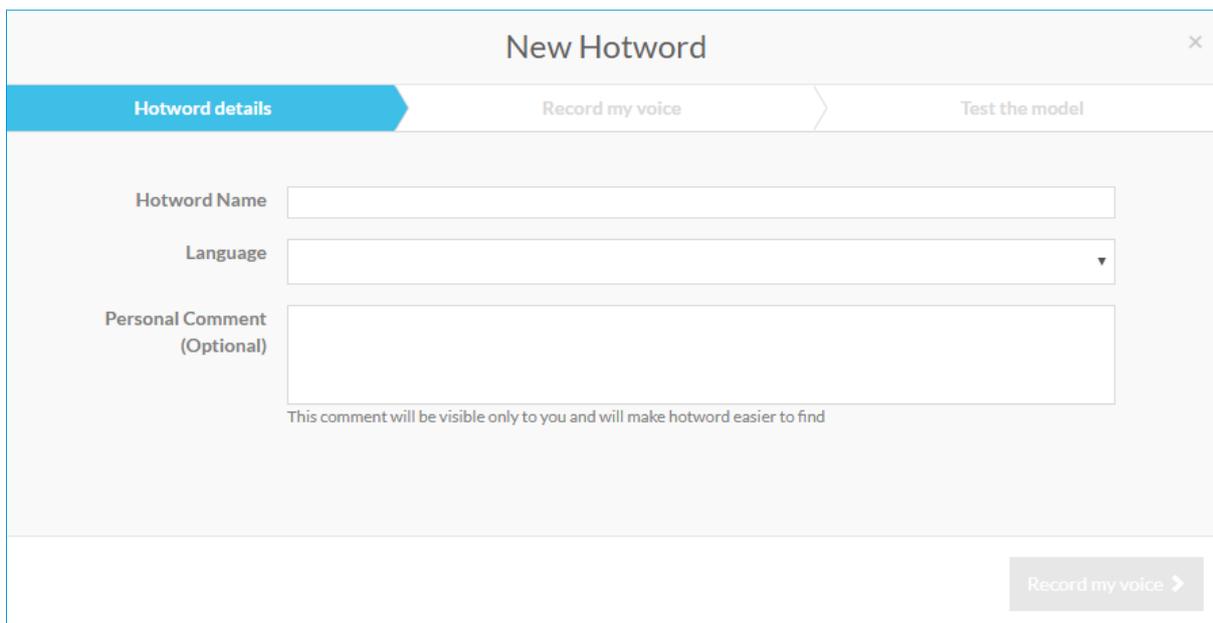
In dem von uns vorbereiteten System reagiert der Talking-Pi auf die beiden Wake-Words „**Talking-Pi**“ und „**Alexa**“.

Sollten Ihnen diese beiden Wake-Words nicht ausreichen, oder sollten Sie Ihren Talking-Pi etwas personalisieren möchten, so können Sie Ihrem Talking-Pi selbstverständlich auch eigene Wake-Words beibringen.

Für die benutzerdefinierte Wake-Word Erkennung verwenden wir die [Gassist Pi Bibliothek](#) in Verbindung mit der [Snowboy Hotword Detection](#).

Diese beiden Bibliotheken sind auf unserem vorbereiteten Betriebssystem selbstverständlich bereits installiert.

Für die benutzerdefinierten Wake-Words können Sie im [Snowboy Dashboard](#) entweder ein eigenes Hotword aufnehmen, oder ein bereits vorhandenes Wake-Word verwenden.



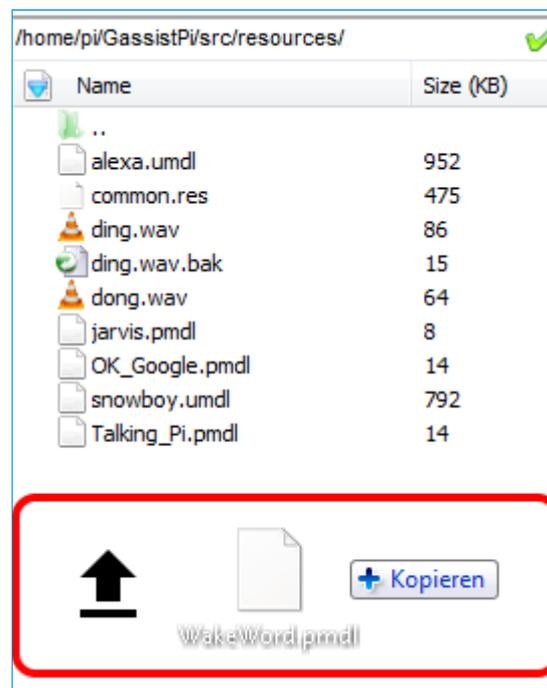
11. BENUTZERDEFINIERTER WAKE-WORDS

Hierbei nehmen Sie das Wake-Word drei mal auf, um es an Ihre Stimme zu adaptieren.

Im Anschluss können Sie die auf Ihr Wake-Word abgestimmte .pmdl-Datei herunterladen.

Kopieren Sie diese Datei anschließend in den folgenden Ordner Ihres Talking-Pi's:

`/home/pi/GassistPi/src/resources/`



Um nun das von Ihnen erstellte Wake-Word In Ihrem Talking-Pi zu aktivieren ist es nötig die Konfigurationsdatei der Gassist Pi Bibliothek zu modifizieren.

Geben Sie dafür den folgenden Befehl in ein Terminal auf Ihrem Talking-Pi:

```
sudo nano /home/pi/GassistPi/src/talkingPi_snowboy.py
```

11. BENUTZERDEFINIERTER WAKE-WORDS

Modifizieren Sie, wie im folgenden Beispiel zu sehen, das **models**-Array um eigene Wake-Word-Konfigurationsdateien hinzuzufügen oder zu entfernen.

```
models = ['/home/pi/GassistPi/src/resources/alexa.umdl', '/home/pi/GassistPi/src/resources/Talking_Pi.pmdl', 'home/pi/GassistPi/src/resources/[meinwakeword].pmdl']
```

Beachten Sie hierbei bitte, dass die Phrase „**[meinwakeword]**“ ein Platzhalter ist, der durch den Dateinamen Ihres eigenen Wake-Words zu ersetzen ist.

Ihre Änderungen können Sie anschließend mit der Tastenkombination **STRG+O** speichern und den Editor mit der Kombination **STRG+X** verlassen.

Nach einem Neustart des Systems sollte Ihr persönliches Wake-Word im System integriert sein und der Talking-Pi auf sein neues Kommando reagieren.

12. ANLERNEN EIGENER BEFEHLE

Um eigene Sprachkommandos hinzuzufügen, die der Talking-Pi als eigenen Befehl erkennt, ist es nötig den Befehl zuerst als erwartete Eingabe zu definieren.

Dies erhöht die Erkennungsrate.

Öffnen Sie dazu die Datei „**googlecloudtalkingpi.py**“ mit Hilfe des folgenden Befehls:

```
sudo nano /home/pi/voice-recognizer-raspi/src/googlecloudtalkingpi.py
```

Fügen Sie hier den folgenden **recognizer.expect_phrase**-Befehl zu den bereits vorhandenen Phrasen hinzu:

```
recognizer.expect_phrase('[Ihr eigenes Sprachkommando]')
```

Tauschen Sie hierbei den Platzhalter durch Ihr eigenes Sprachkommando aus.

```
def main():
    # Definition aller zu erwartenden Befehle
    # Diese Erwartungen erhoehen die Erkennungsgenauigkeit des Talking-Pis
    # Aehnlich klingende Befehle, leichte Abweichungen oder auch Akzente in der Sprache
    # koennen durch die Definierung der Erwartungen besser erkannt werden
    recognizer = aiy.cloudspeech.get_recognizer()
    # Sie koennen hier weitere Phrasen hinzufuegen!
    recognizer.expect_phrase('mach das Licht an')
    recognizer.expect_phrase('mach das Licht aus')
    recognizer.expect_phrase('mach das Licht oben an')
    recognizer.expect_phrase('mach das Licht oben aus')
```

Da wir nun die zu erwartende Spracheingabe definiert haben, kann nun der Befehl aus eigentliche Erkennung definiert werden.

Fügen Sie dazu, im weiteren Verlauf der bereits geöffneten Datei, den folgenden Inhalt zu den bereits existierenden Definitionen hinzu.

12. ANLERNEN EIGENER BEFEHLE

```
elif '[Ihr eigenes Sprachkommando]' in text:
    subprocess.call(["python", "/home/pi/I2C-LCD/EigenerBefehl.py"])
    subprocess.call(["bash", "/home/pi/TalkingPi_Befehle/Receiver/EigenerBefehl"])
    subprocess.Popen(["aplay", "/home/pi/GassistPi/src/resources/dong.wav"], stdin=subprocess.PIPE,
stdout=subprocess.PIPE, stderr=subprocess.PIPE)
    time.sleep(2)
```

Achten Sie bitte darauf auch hier den Platzhalter durch Ihren eigenen Befehl auszutauschen.

Auch die Einrückungen müssen Sie unter Umständen nach dem Kopieren korrigieren.

Orientieren Sie sich hierfür einfach an den bereits vorhandenen Befehls-Definitionen.

```
print('You said "', text, "'")
# Prüfe auf folgenden Befehl
if 'mach das Licht an' in text:
    # Wenn der Befehl passt, dann führe die folgenden Befehls-Dateien aus
    if Display:
        # Der folgende Befehl fuer das Display wird nur ausgefuehrt wenn der entsprechende Wert oben gesetzt wurde. Dieser ist standardmaessig aktiv.
        subprocess.call(["python", "/home/pi/I2C-LCD/Befehl1.py"])
        subprocess.call(["bash", "/home/pi/TalkingPi_Befehle/Licht/LichtAn.sh"])
    # Spiele ausserdem folgende Tondatei ab
    subprocess.Popen(["aplay", "/home/pi/GassistPi/src/resources/dong.wav"], stdin=subprocess.PIPE, stdout=subprocess.PIPE, stderr=subprocess.PIPE)
    # Warte danach 2 Sekunden
    time.sleep(2)
elif 'mach das Licht aus' in text:
    if Display:
        subprocess.call(["python", "/home/pi/I2C-LCD/Befehl2.py"])
        subprocess.call(["bash", "/home/pi/TalkingPi_Befehle/Licht/LichtAus.sh"])
        subprocess.Popen(["aplay", "/home/pi/GassistPi/src/resources/dong.wav"], stdin=subprocess.PIPE, stdout=subprocess.PIPE, stderr=subprocess.PIPE)
    time.sleep(2)
```

Innerhalb der neu definierten Befehlsaufrufe, werden drei verschiedene Dateien aufgerufen.

Die ersten beiden Dateien ("/home/pi/I2C-LCD/EigenerBefehl.py" und "/home/pi/TalkingPi_Befehle/Receiver/EigenerBefehl") definieren die Aktionen die beim Ausführen des entsprechenden Sprachbefehls aufgerufen werden.

Dabei stellt in diesem Beispiel die erste Datei die Ausgabe auf dem Display und die zweite Datei die eigentliche Befehlsfolge dar.

Die dritte Datei definierten den Ton der ausgegeben wird, sobald der Sprachbefehl erkannt wurde.

Sie können selbstverständlich beliebig viele aufzurufende Dateien hier eintragen.

Die von uns angelegte Struktur der drei Dateien (Display, Befehl, Ton) dient nur der Übersicht.

12. ANLERNEN EIGENER BEFEHLE

Bitte beachten Sie, dass die auszuführenden Dateien von Ihnen noch in den entsprechenden Ordnern angelegt werden müssen.

Sie können natürlich so viele Dateien definieren wie Sie benötigen.

In unserer ausgelieferten Standardkonfiguration haben wir allerdings jeweils die folgenden Dateien zur Ausführung definiert.

An diesen können Sie sich für ihre persönliche Konfiguration orientieren.

Display-Ansteuerung:

Dateiort: /home/pi/I2C-LCD/[IhreDatei].py

Beispiel:

```
import lcddriver
from time import *

lcd = lcddriver.lcd()
lcd.lcd_clear()

lcd.lcd_display_string("+++ Ausgabe +++", 1)
lcd.lcd_display_string("+Ich bin eine Ausgabe+", 2)
```

Ton-Ausgabe:

Dateiort: /home/pi/GassistPi/src/resources/[IhreTonDatei].wav

Weitere auszuführende Befehlsdateien:

Dateiort: /home/pi/TalkingPi_Befehle/

Dateiformat: .sh

Hinweis: Hier sind die einzelnen Befehlsdateien, um eine Übersicht zu ermöglichen, nochmals in die Ordner *Beamer*, *Licht*, *Computer* und *Receiver* unterteilt.

Sie können selbstverständlich weitere Kategorien hinzufügen.

Bitte achten Sie jedoch darauf, diese dann bei der Dateireferenzierung richtig anzugeben.

Nach einem Neustart Ihres Talking-Pi's sollte Ihr neuer Sprachbefehl erfolgreich erkannt und verarbeitet werden können.

13. SICHERHEITSHINWEISE

Bitte beachten Sie unbedingt die folgenden Sicherheitshinweise zum Schutz vor elektrischen Schlägen und Verletzungen.

- Bedienungsanleitung und Sicherheitshinweise aufbewahren!
Bei einer Weitergabe des Gerätes an Dritte müssen diese unbedingt mitgegeben werden.
- Verwenden Sie dieses Gerät nur für den jeweiligen Einsatzbereich und entsprechend der Bedienungsanleitung.
Halten Sie sich bei der Montage und Nutzung an unsere Anleitung, nur so ist sichergestellt das der Zusammenbau richtig funktioniert, lesen Sie die Anleitung vollständig bevor Sie mit der Montage / Nutzung beginnen.
- Überzeugen Sie sich vor Inbetriebnahme, dass die vorhandene Netzspannung der angegebenen Netzspannung entspricht.
Bei der Verwendung darf unter keinen Umständen von der angegebenen Netzspannung abgewichen werden.
- Fassen Sie den Netzstecker niemals mit nassen Händen an.
- Alle Anschlusskabel zum Raspberry Pi und Talking-Pi müssen unversehrt sein.
Bei Beschädigungen ist das entsprechende Kabel unverzüglich auszutauschen.
- Bei Arbeiten am Bausatz muss die Stromversorgung vorher getrennt werden und gegen Wiedereinschalten gesichert (z.B. durch Entfernen des Netzteils).
- Schützen Sie das Gerät vor Feuchtigkeit und Regen. Benutzung nur in trockenen Räumen.
Eine Verwendung bei Regen, in feuchten Räumen oder sogar in oder unter Wasser darf unter keinen Umständen ausgeführt werden.
- Durch den Genuss von Alkohol, Medikamenten und Drogen, sowie durch Krankheit, Fieber und Müdigkeit kann ihre Reaktionsfähigkeit und Ihre Wahrnehmung beeinträchtigt werden.
In diesem Fall darf das Gerät nicht verwendet werden.
- Nach der gesetzlichen Regelung ist dieses Produkt kein Spielzeug und darf somit auch nicht von Kindern bedient werden.
Halten Sie das Produkt daher stets von Kindern fern und bewahren Sie es an einem für Kinder unzugänglichen Ort auf.

14. INFORMATIONEN- UND RÜCKNAHMEPFLICHTEN

Symbol auf Elektro- und Elektronikgeräten



Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte nicht in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

Rückgabemöglichkeiten

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in Haushaltsüblichen Mengen abgeben werden.

Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten

Simac GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

Möglichkeit Rückgabe in Ihrer Nähe

Wir senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an Service@joy-it.net oder per Telefon an uns.

Informationen zur Verpackung

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

15. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: service@joy-it.net

Ticket-System: <http://support.joy-it.net>

Telefon: +49 (0)2845 98469 – 66 (10- 17 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net