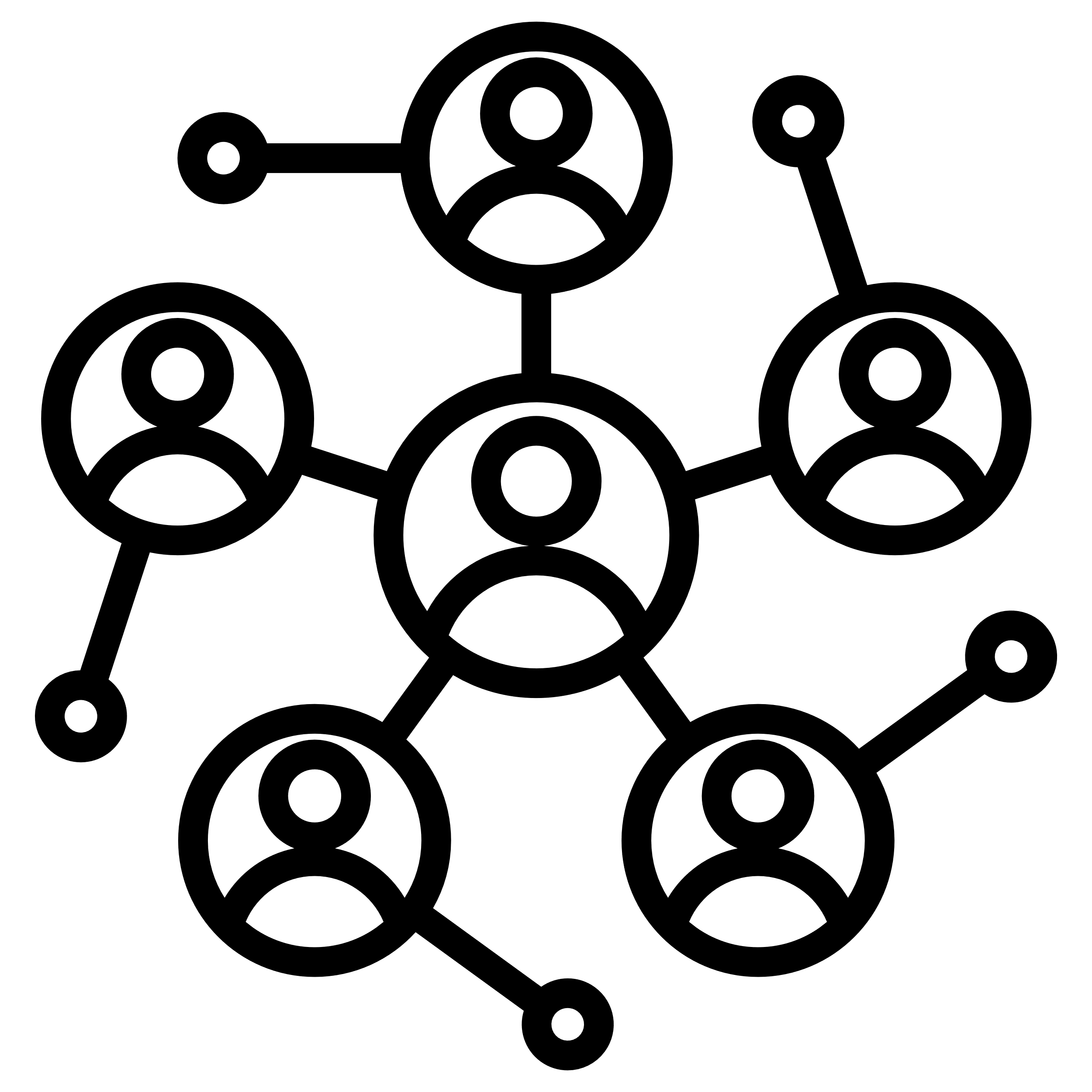


EchoPlay Chip ersatz

Projektplanung

EchoPlay



Tom Nielsen

Ausgabe v1.0

Inhalt.

[1 Planung. 3](#_Toc194302041)

[1.1 Informationen 3](#_Toc194302042)

[1.1.1 Komponente 3](#_Toc194302043)

[1.2 Planung 4](#_Toc194302044)

[1.2.1 Steckplan 4](#_Toc194302045)

[1.2.2 Vorgehen 4](#_Toc194302046)

[2 Verzeichnisse. 5](#_Toc194302047)

[2.1 Abbildungen 5](#_Toc194302048)

[2.2 Tabellen 5](#_Toc194302049)

[2.3 Quellen 5](#_Toc194302050)

# Planung.

## Informationen

### Ausgangslage

Ich muss für mein Projekt «EchoPlay» den vorhandenen Chip in der LED-Box ersetzen. Der vorhandene Chip ist nicht benutzerfreundlich und kann sehr schlecht programmiert werden, darum werde ich diesen mit einem Carrier Board mit ESP32 ersetzen.

### Komponente

**SparkFun MicroMod ATP Carrier Board**

Das Carrier Board von SparkFun ist die Hauptkomponente meines Projekts. Auf diesem werde ich all meinen Code schreiben und diesen auch ausführen. Die Pins, welche wir brauchen werden, sind GND, A0/1, G0 – G3, VIN, PWM0 und 3.3V.

**LED-Screen**

Das LED-Screen, welches schon bei der LED-Box vorhanden war, hat ein 16x16 LED-Display, um verschiedene Sachen anzuzeigen. Auf dem LED-Screen hat es Folgende Pins VCC, EN, DA, CLK, LAG und GND

**Knöpfe**

Auf der LED-Box hat es 2 Knöpfe vorhanden, welche man benutzen kann, um Inputs zu erkennen.

|  |  |
| --- | --- |
| Begriff | Beschreibung |
| **Carrier Board** | |
| A0/1 | Analoge Pins welche einen ADC enthalten haben |
| G0 – G3 | General Input/Output Pins |
| VIN | Stromversorgung vom Carrier Board |
| PWM0 | Sendet digitales Signal, um analoge Geräte zu kontrollieren |
| 3.3V | Leitet einen 3.3V Strom |
| **LED-Screen** | |
| VCC | Stromversorgung (+) für die LED-Driver |
| EN | (ENABLE) Schalter zum Ein- oder Ausschalten einer Funktion |
| DA | (DATA) Datenleitung, um Informationen zu bekommen |
| CLK | (CLOCK) Taktgeber, sodass der Takt von der LED-Box gleich ist wie vom Chip |
| LAK | (LATCH) Speichert oder bestätigt Daten |
| GND | Minuspol (-) von der Stromversorgung |
| **Mikrofon** | |
| AUD | Audio Signal Output |
| VCC | Volt Eingang (2.3V – 3.6V). Braucht eine Spannung von 265 Mikro Ampere |
| **Knöpfe** | |
| Common | Gemeinsamer Bezugspunkt in einer Schaltung |

Tabelle Begriff und Ihre Bedeutung

## Planung

### Steckplan

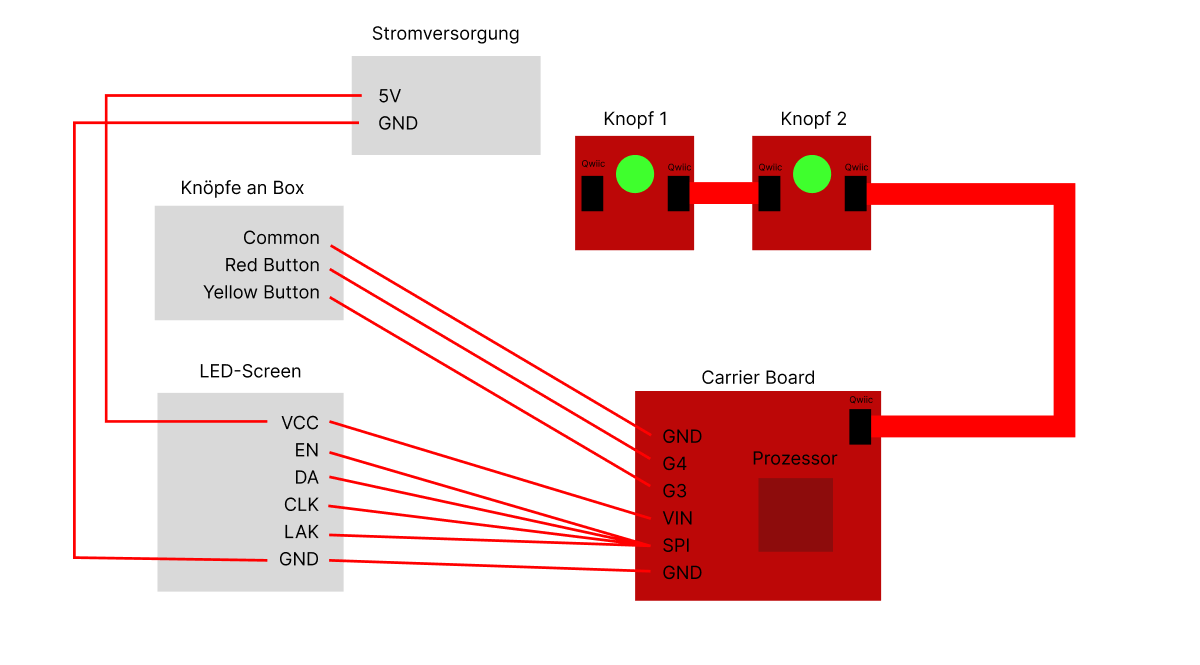


Abbildung Steckplan vom EchoPlay

### Vorgehen

Zuerst muss ich den vorhandenen Chip, welcher im LED-Screen eingelötet ist, entfernen.

Nachdem muss ich das Carrier Board mit dem LED-Screen wie im Steckplan zusehen zusammen lösten, ausser dem VCC zu VIN, weil ich später, wenn ich Programmieren will, mit dem USB-Kabel Strom am Carrier Board gebe. VCC und VIN muss ich erste verbinden, wenn ich fertig mit dem Programmieren bin.

Wenn das erledigt ist, muss ich noch die beiden Knöpfe, Common und das Mikrofon wie oben zusammenlöten.

Wenn dann alles zusammengelötet ist, kann ich ein USB-Kabel mit dem Carrier Board verbinden und den Code schreiben.

Zum Schluss, wenn der Code fertig ist, muss ich dann VCC mit VIN zusammenlöten.

# Verzeichnisse.

## Abbildungen

[Abbildung 1 Steckplan vom EchoPlay 4](#_Toc193975845)

## Tabellen

[Tabelle 1 Begriff und Ihre Bedeutung 3](#_Toc193975847)

## Quellen

s. (s. s s). *s*. Von https://github.com/attowatt/frekvensHack abgerufen