

EchoPlay Chip Ersatz

Projektplanung

EchoPlay

* Informationen
* Planung



Tom Nielsen

Ausgabe v1.0

Inhalt.

[1 Planung. 3](#_Toc198107080)

[1.1 Informationen 3](#_Toc198107081)

[1.1.1 Komponente 3](#_Toc198107082)

[1.2 Planung 7](#_Toc198107083)

[1.2.1 Original 7](#_Toc198107084)

[1.2.2 Entfernen des PIC16F684 7](#_Toc198107085)

[1.2.3 Pins löten 8](#_Toc198107086)

[1.2.4 Verbindung mit dem ESP32 9](#_Toc198107087)

[2 Verzeichnisse. 10](#_Toc198107088)

[2.1 Abbildungen 10](#_Toc198107089)

[2.2 Tabellen 10](#_Toc198107090)

[2.3 Quellen 10](#_Toc198107091)

# Planung.

## Informationen

### Komponente

#### FREKVENS LED multi-use light

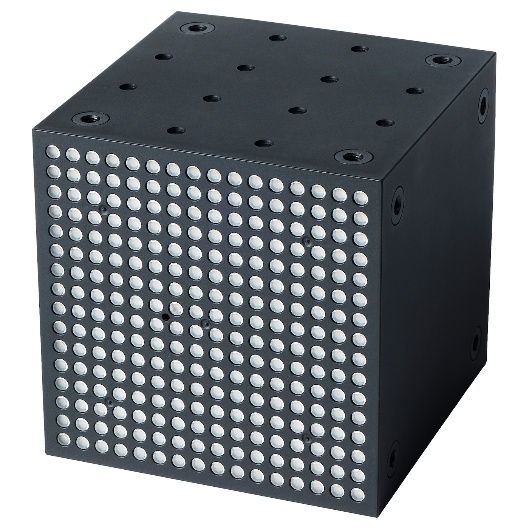
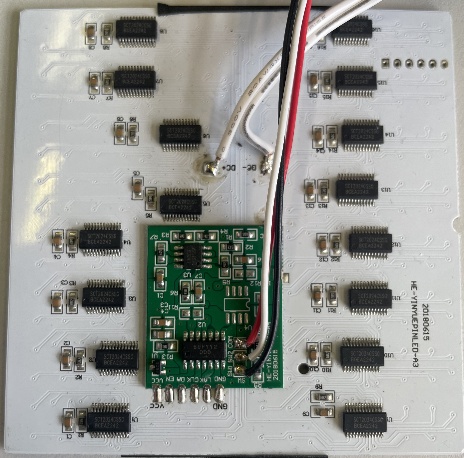


Abbildung LED-Box

In der LED-Box von Teenage Engineering befindet sich ein System, das Licht in verschiedenen Mustern ausstrahlt und auf äussere Reize wie Geräusche reagieren kann. Die Box enthält verschiedene elektronische Komponenten, darunter ein Mikrofon, ein LED-Screen, ein zentraler Steuerchip sowie eine Stromversorgung. Ich werde diese Hardware so modifizieren, dass man einfache Spiele darauf spielen kann. Dafür plane ich, den Chip zu ersetzen und eine zusätzliche Steuerungsmöglichkeit einzubauen. Die LED-Anzeige soll dabei nicht nur Muster, sondern auch einfache Spielformen grafisch darstellen können, wodurch aus der ursprünglichen Licht Box ein interaktives Spielgerät entsteht.

#### LED-Screen

Ein Bild, das Rechteck, Muster, Quadrat, Screenshot enthält.

KI-generierte Inhalte können fehlerhaft sein.

Abbildung LED-Screen

Der LED-Screen besitzt ein 16x16 Display, also insgesamt 256 einzelne LEDs. Auf der Rückseite ist ein Board montiert, auf dem sich zwei zentrale Chips befinden: der LM358, ein Operationsverstärker, und der **PIC16F684**, ein Mikrocontroller, der grundlegende Steuerfunktionen übernimmt. Neben diesem Board befinden sich zusätzlich 16 LED-Treiber-Chips, die dafür zuständig sind, die einzelnen LEDs gezielt anzusteuern. Diese Treiber ermöglichen das gezielte Ein- und Ausschalten der LEDs, wodurch unterschiedliche Muster und Animationen auf dem Display dargestellt werden können.

#### Original Carrier Board

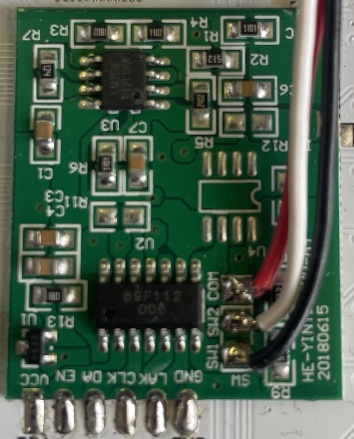


Abbildung Original Carrier Board

Dieses Carrier-Board ist das originale Trägerboard für die Chips PIC16F684 und LM358. Es dient dazu, diese Bauteile stabil zu halten und ihre Pins so miteinander zu verbinden, dass sie gemeinsam funktionieren können. Das Board stellt die nötigen Verbindungen zwischen dem Mikrocontroller, dem Operationsverstärker und den weiteren Komponenten wie den LED-Treibern her und sorgt dafür, dass Daten und Signale korrekt übertragen werden. So ermöglicht es die Steuerung des LED-Displays und die Zusammenarbeit aller Teile als ein funktionierendes System.

#### PIC16F684

Ein Bild, das Elektronik, Schaltung, Elektronisches Bauteil, Elektrisches Bauelement enthält.

KI-generierte Inhalte können fehlerhaft sein.

Abbildung PIC16F684

Der PIC16F684 ist ein 8-Bit-Mikrocontroller, der sich gut für einfache Steuerungsaufgaben eignet. Er enthält einen programmierbaren Speicher, verschiedene Ein- und Ausgänge sowie Funktionen wie PWM und analoge Signalverarbeitung. Er übernimmt die zentrale Steuerung und sorgt dafür, dass die LED-Treiber die richtigen Signale erhalten, um Muster oder Animationen auf dem Display darzustellen.

#### LM358

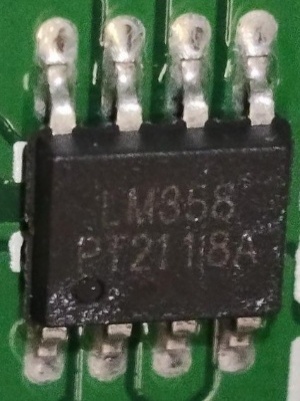
****

Abbildung LM358

Der LM358 ist ein integrierter Operationsverstärker (Op-Amp), der zwei unabhängige Verstärker in einem Chip enthält. Er verstärkt analoge Signale vom Mikrofon. Im Gegensatz zu digitalen Bauteilen verarbeitet der LM358 kontinuierliche Spannungen und kann dabei Signale glätten, verstärken oder vergleichen.

#### **SparkFun MicroMod ATP Carrier Board**



Abbildung Carrier Board

Das Carrier Board von SparkFun dient dazu, die kleinen Pins eines Prozessormoduls zugänglich zu machen, indem es sie auf grössere, standardisierte Anschlüsse überträgt. Dadurch wird es einfacher, Verbindungen zu anderen Komponenten herzustellen, etwa für Sensoren oder Aktoren, ohne direkt an die feinen Pins gehen zu müssen. Das erleichtert besonders den Aufbau von Prototypen und Testschaltungen.

#### **ESP32 Prozessor**

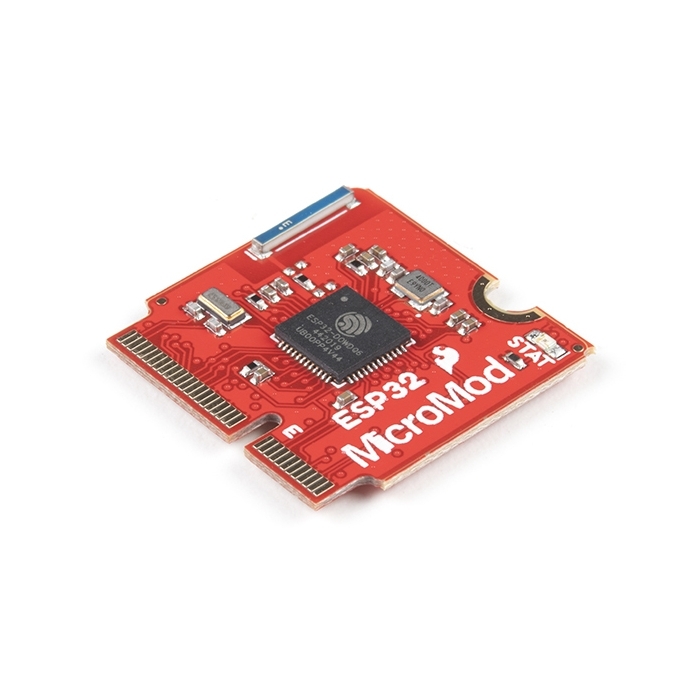


Abbildung ESP32

Der ESP32 ist ein kleiner Computerchip, der WLAN und Bluetooth hat und viele Aufgaben übernehmen kann zum Beispiel Daten messen, verarbeiten oder Geräte steuern. Wenn man ihn in ein Carrier Board einsetzt, werden seine feinen Pins grösser und besser erreichbar. So kann man einfacher Kabel anschliessen und mit anderen Geräten wie Sensoren, Tastern oder LEDs verbinden. Das macht den Aufbau von eigenen Projekten viel leichter.

#### DIP300-SOIC-16N



Abbildung DIP300-SOIC-16N

DIP300-SOIC-16N ist ein Adapter, mit dem man einen SOIC-16-Chip, also einen kleinen SMD-Baustein mit 16 Anschlüssen, auf ein Steckbrett oder eine Lochrasterplatine setzen kann. Der Adapter wandelt die feinen, eng beieinanderliegenden Pins in das grössere DIP300-Format um, das besser für Prototypenbau geeignet ist. So kann man den Chip einfacher testen und mit anderen Bauteilen verbinden.

#### HRS-1B-07-GA

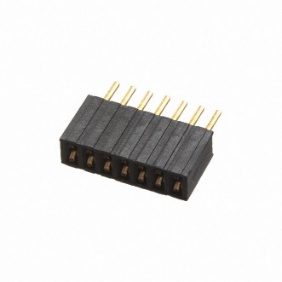


Abbildung HRS-1B-07-GA

Der HRS-1B-07-GA ist ein kleiner 7-poliger Steckverbinder, der auf eine Leiterplatte gelötet wird. Er wird verwendet, um Bauteile sicher elektrisch zu verbinden, zum Beispiel zwischen zwei Platinen. Die vergoldeten Kontakte sorgen für eine gute Leitfähigkeit.

|  |  |
| --- | --- |
| Begriff | Beschreibung |
| **Carrier Board** | |
| A0/1 | Analoge Pins welche einen ADC enthalten haben |
| G0 – G3 | General Input/Output Pins |
| VIN | Stromversorgung vom Carrier Board |
| **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 |
| **Knöpfe LED-Box** | |
| Common | Das gleiche wie GND |
| SW1/2 | Output von den Knöpfen |

Tabelle Begriffe und Ihre Bedeutung

## Planung

### Original

Um zu verstehen, wie die LED-Box am Ende korrekt aufgebaut und verdrahtet sein muss, betrachten wir zunächst den Originalzustand. Dabei analysieren wir, wie die einzelnen Komponenten im ursprünglichen Aufbau miteinander verbunden sind.

Ein Bild, das Text, Diagramm, Screenshot, Plan enthält.

KI-generierte Inhalte können fehlerhaft sein.

Abbildung Originalsteckplan

Der gezeigte Aufbau stellt die ursprüngliche Konfiguration der LED-Box dar. In dieser Box befinden sich sieben Hauptkomponenten:

* Stromversorgung
* Bildschirm (Display)
* Mikrofon
* Mikrocontroller PIC16F684
* Operationsverstärker LM358
* Zwei Taster

Die zentrale Steuerung wird durch den Mikrocontroller PIC16F684 übernommen. Da ich diesen durch einen modernen ESP32 ersetzen wollen, müssen wir dessen Verbindungspunkte übernehmen. Dazu entferne ich den PIC16F684 und erweitere die bestehenden Lötstellen, um die neue Verbindung mit dem ESP32 herstellen zu können.

### Entfernen des PIC16F684

Bevor ich den ESP32 einbauen können, muss der alte Mikrocontroller entfernt werden. Dazu erhitze ich die Lötstellen vorsichtig mit einem Lötkolben und lösen den PIC16F684 aus der Platine. Besonders wichtig ist es, die Leiterbahnen nicht zu beschädigen, da diese später zur Verbindung mit dem Adapter benötigt werden.

Ein Bild, das Elektronik, Schaltung, Elektronisches Bauteil, Elektrisches Bauelement enthält.

KI-generierte Inhalte können fehlerhaft sein.



Abbildung PIC16F684

### Pins löten

Um den ESP32 mit der LED-Box zu verbinden, müssen die ehemaligen Lötstellen des PIC16F684 in steckbare Pins umgewandelt werden. Dafür verwende ich einen DIP300-SOIC-16N Adapter, den ich auf die alten Lötstellen auflöten. Dieser Adapter erlaubt es, eine Verbindung zwischen der SMD-Bauform des alten Controllers und standardmässigen Steckverbindungen herzustellen.



Abbildung DIP300-SOIC-16N

Zusätzlich löte ich 14 männlich-zu-weiblich Pins an die Adapterausgänge, um eine flexible Verbindung mit dem ESP32 zu ermöglichen. Dies mache ich mit 2 HRS-1B-07-GA, welche ich in den DIP300-SOIC-16N stecke und diese dann löte.

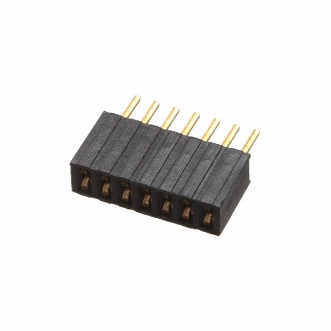


Abbildung HRS-1B-07-GA

Wenn diese Schritte abgeschlossen sind, kann der ESP32 problemlos über männlich-zu-männlich Jumperkabel mit den neu geschaffenen Pin-Verbindungen verbunden werden.

### Verbindung mit dem ESP32

Im letzten Schritt wird der ESP32 mit den vorbereiteten Pins verbunden. Die Jumperkabel werden entsprechend dem Schaltplan so gesteckt, dass jede Funktion (z. B. Datenleitung, Stromversorgung, Eingänge) korrekt zugeordnet ist.

Ein Bild, das Diagramm, Text, Screenshot, Plan enthält.

KI-generierte Inhalte können fehlerhaft sein.

Abbildung Modifizierter Steckplan

Im letzten Schritt wird der ESP32 mit den vorbereiteten Pins verbunden. Die Jumperkabel werden entsprechend dem Schaltplan so gesteckt, dass jede Funktion korrekt zugeordnet ist.

1. VIN
2. G1
3. G2
4. G3
5. G4
6. G5
7. G6
8. –
9. –
10. –
11. A1
12. –
13. –
14. GND

# Verzeichnisse.

## Abbildungen

[Abbildung 1 LED-Box 3](#_Toc198114483)

[Abbildung 2 LED-Screen 3](#_Toc198114484)

[Abbildung 3 Original Carrier Board 4](#_Toc198114485)

[Abbildung 4 PIC16F684 4](#_Toc198114486)

[Abbildung 5 LM358 4](#_Toc198114487)

[Abbildung 6 Carrier Board 5](#_Toc198114488)

[Abbildung 7 ESP32 5](#_Toc198114489)

[Abbildung 8 DIP300-SOIC-16N 5](#_Toc198114490)

[Abbildung 9 HRS-1B-07-GA 6](#_Toc198114491)

[Abbildung 10 Originalsteckplan 7](#_Toc198114492)

[Abbildung 11 PIC16F684 8](#_Toc198114493)

[Abbildung 12 DIP300-SOIC-16N 8](#_Toc198114494)

[Abbildung 13 HRS-1B-07-GA 8](#_Toc198114495)

[Abbildung 14 Modifizierter Steckplan 9](#_Toc198114496)

[Abbildung 15 offene LED-Box 10](#_Toc198114497)

[Abbildung 16 offene LED-Box mit Carrier Board 10](#_Toc198114498)

[Abbildung 17 Qwiic Konnektoren 10](#_Toc198114499)

[Abbildung 18 Hüllen Löcher 11](#_Toc198114500)

[Abbildung 19 Qwiic Kabel nach aussen 11](#_Toc198114501)

[Abbildung 20 Offene LED-Box mit Kabel 11](#_Toc198114502)

## Tabellen

[Tabelle 1 Begriffe und Ihre Bedeutung 6](#_Toc198189966)

## Quellen

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