

# Sound Servoplatine

**++ Diese Seite wird derzeit überarbeitet ++**

Über die Servo-Platine 510 können drei Soundmodule angesteuert werden. Es werden dabei JQ6500 Module oder MP3-TF16-p/DFPlayer Mini Module verwendet. Dabei spielt es keine Rolle ob nur JQ6500 oder nur MP3-TF16-p/DFPlayer Mini oder diese beliebig gemischt, eingesetzt werden. Die drei Module können auch parallel je einen Sound abspielen.

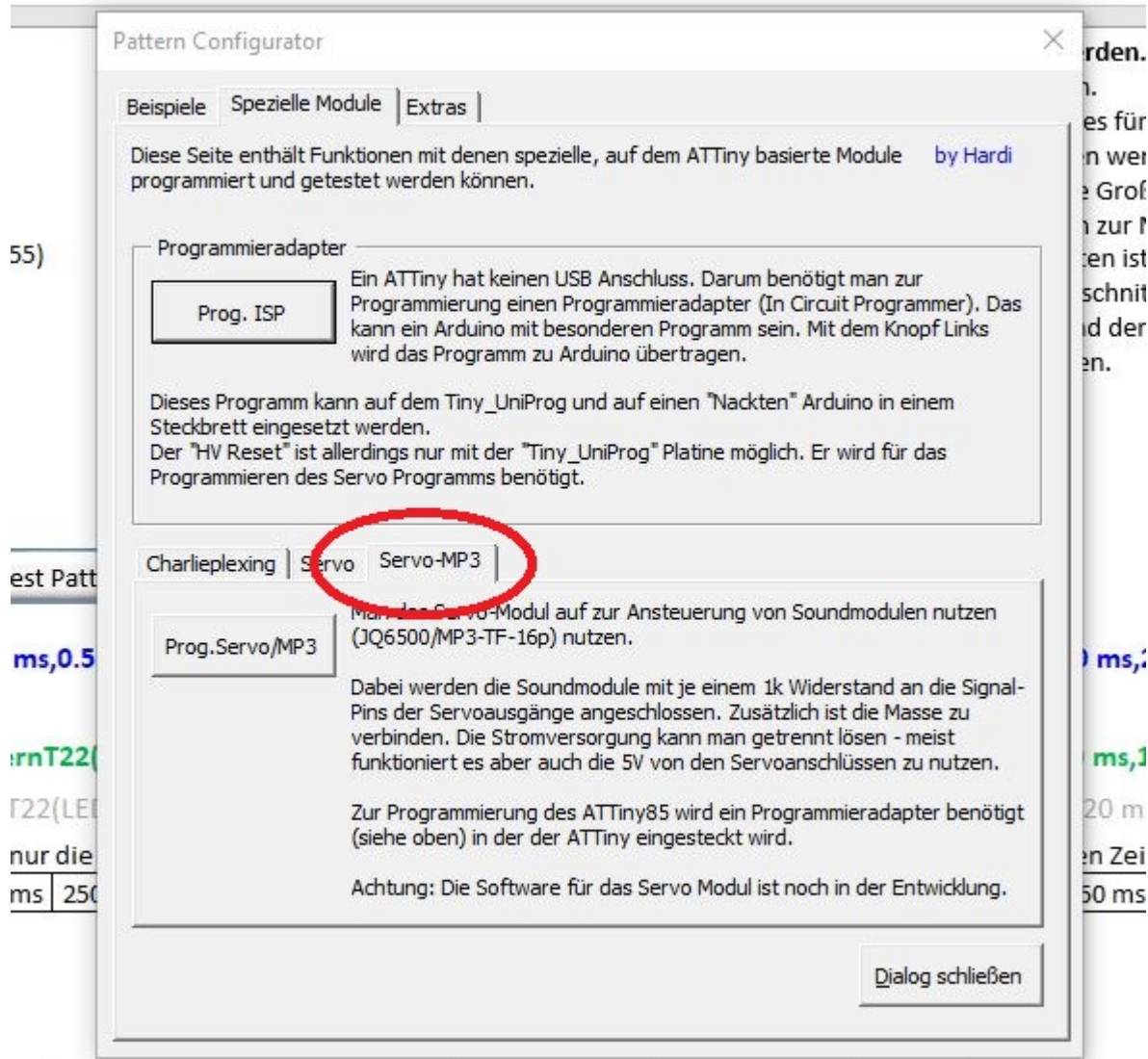
Das JQ6500 hat dabei den Vorteil das es bereits 2 MByte Speicher enthält, was für etliche Soundprojekte ausreichend sein dürfte.

Das MP3-TF-16p nutzt hingegen eine MicroSD-Karte mit bis zu 32 GByte und vereinfacht durch die mögliche Ordnerstruktur die Verwaltung der Sounddaten.

## Vorbereitung Hardware

### ATTiny 85

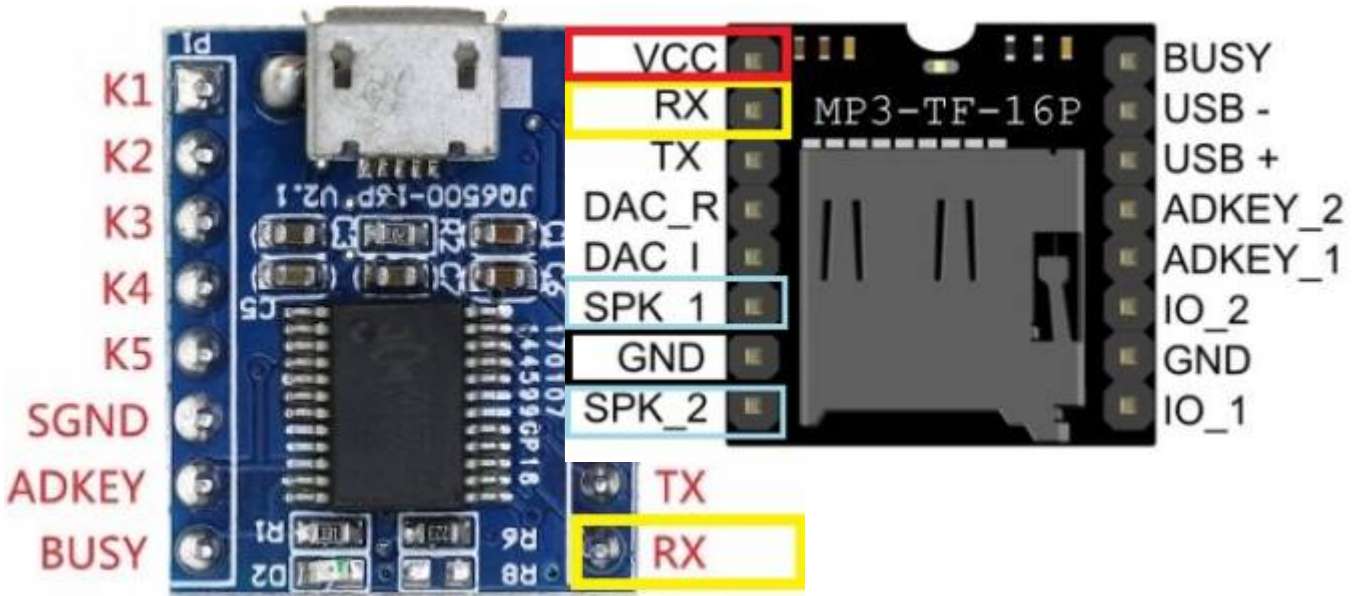
Als Servoplatine eignet sich jede 510DE-Platine, bestückt als Servoplatine. Die Lötjumper für den Servobetrieb SERVO, SERVO1, SERVO2, SERVO3 müssen geschlossen werden. Zunächst muss der ATTiny für die Servoplatine programmiert werden. Die Anleitung dazu [Programmierung von ATTinys für Servo, Charlieplexing oder Sound](#) ist im WIKI zu finden. Im Pattern-Configurator unter spezielle Module **Servo-MP3** auswählen.



Das folgende Bild zeigt die Pins am Ausgang der Servo-Platine. Der SIG-J1 - Pin ist mit dem Eingang des ersten Soundmodules zu verbinden, entsprechend J2 mit Modul 2 und J3 mit Modul3.



## JQ6500 und MP3-TF16-p/DFPlayer Mini



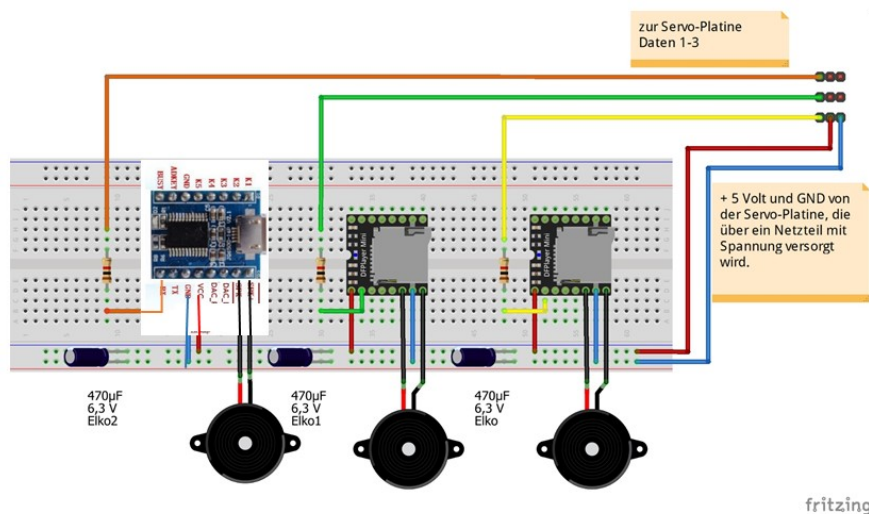
## Verdrahtung

Die drei Ausgänge (SIG) der Servo-Platine werden nun über je einen 1 kOhm Widerstand mit den Eingängen (RX) der Sound-Module verbunden.

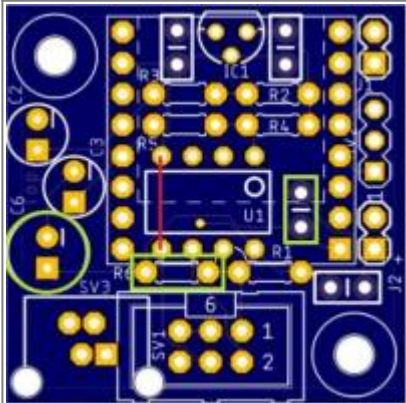
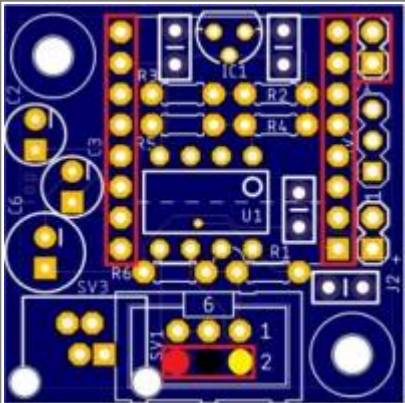
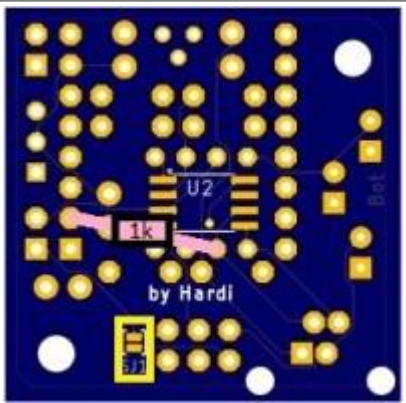
Die Versorgungsspannung sollte über eine Verteilerplatine mit angeschlossener stabiler Spannungsversorgung erfolgen, da die Leistung an der Hauptplatine zur Versorgung der Sound-Module nicht ausreicht.

Die 470µF Elektrolytkondensatoren dienen als Puffer für die recht hohen Einschaltströme der Soundmodule.

Eine separate 5V Spannungsversorgung ist ebenfalls möglich.



Als weitere Möglichkeit bietet sich der Umbau einer unbestückten 501de\_Soundplatine (MP3-TF-16p) an:

		
Drahtbrücke (rot) an der Oberseite .....	Buchsenleisten: 2x8 Soundmodul	Widerstand 1kOhm an der Unterseite
R6 Widerstand 10Ohm	1x3 rot(VCC), schwarz(GND), gelb(SIG)	Jumper SJ1 schließen
Keramikkondensator 100nf	<b>ACHTUNG-ServoAnschluss rot/schwarz vertauscht</b>	
C6 Elko 470µF	1x2 Lautsprecher	

## SD Karte

Die Verzeichnisstruktur der Sounddateien auf der SD-Karte ist relativ starr vorgeschrieben. Es können Sounddateien (wahlweise im mp3- oder wav-Format) in folgende Verzeichnisse der SD-Karte abgelegt werden:

- Wurzelverzeichnis (Dateinamen 4-stellig 0001.mp3 etc.)
- Verzeichnis ADVERT (Dateinamen 4-stellig)
- Verzeichnis mp3 (Dateinamen 4-stellig)
- Verzeichnisse 01 bis 32 - (Dateinamen 3-Stellig 001.mp3 etc.)

Es werden nur die ersten Zeichen der Dateinamen ausgewertet. Der tatsächliche Dateiname kann länger sein. Damit sind Dateinamen der Art

0023Yesterday .mp3

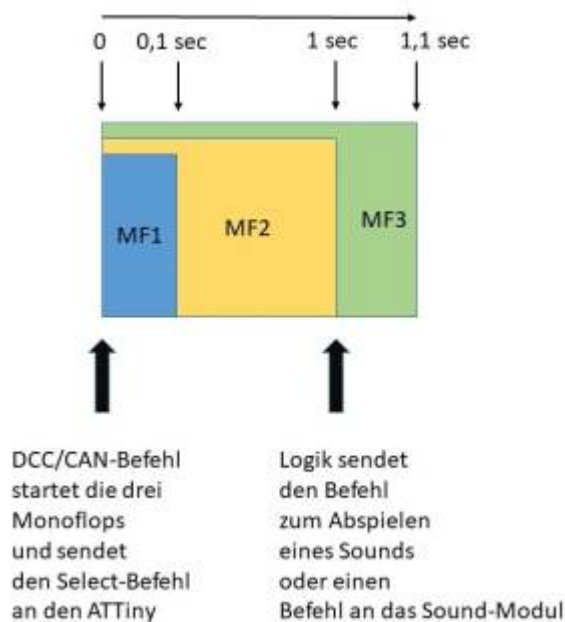
möglich. Dieser Dateiname wird vom Soundmodul als „0023.mp3“ behandelt. Das erleichtert den Umgang mit den Sounddateien deutlich und sollte ausgiebig genutzt werden.

## Programm-Generator

Im Prog-Gen gibt es für den Servo-Sound folgende Befehle:



# Steuerung über DCC/CAN-Befehle



Vor jedem Sound-Befehl muss das Modul ausgewählt werden auf dem sich die Sound-Datei befindet. So können Sound-Dateien in unterschiedlicher Reihenfolge von den drei Modulen abgespielt werden. Die Logik stellt sicher, dass zunächst über den ATTiny das Modul ausgewählt wird, der Befehl umgesetzt werden kann und dann, mit zeitlichem Verzug, der Track ausgewählt oder eine andere Funktion des Moduls aufgerufen wird.

## Beispiel:

Im folgenden Beispiel wird bei Aufruf des:

- DCC-Befehl „3“ der erste Sound des ersten Moduls (JP6500) abgerufen
- DCC-Befehl „4“ der erste Sound des zweiten Moduls (DFPlayer Mini) abgerufen
- DCC-Befehl „5“ der erste Sound des dritten Moduls (DFPlayer Mini) abgerufen

Aktiv	Filter	Adresse oder Name	Typ	Startwert	Beschreibung	Verteiler-Nummer	Stecker-Nummer	Leuch	Name	Beleuchtung, Sound, oder andere Effekte	Start LedNr	LEDs	InCh	Loc InCh	LED/ Sound Kanal
					<b>Modul 1 JQ6500</b>										
✓		3	Rot		startet MonoFloP (MF) 1				! Mono-Flop	MonoFloP(#F11, #InCh, 0.1 Sek)					
✓		MF11	Rot		sendet Select Befehl für Modul 1				Pin MP3-Modul definieren	MP3_SELECT_MODULE(#LED, #InCh, 1)	1	^ C1-1			0
✓		3	Rot		startet MonoFloP (MF) 2				! Mono-Flop	MonoFloP(#F12, #InCh, 1 Sek)					
✓		3	Rot		startet MonoFloP (MF) 3				! Mono-Flop	MonoFloP(#F13, #InCh, 1.1 Sek)					
✓		MF12			logische Verknüpfung der drei MFs				! Logische Verknüpfung	Logic(#F12u13, NOT #InCh AND #F13)					
✓		MF12u13	Rot		spielt Track 1 vom JP6500 ab (Welding)				! Titel # abspielen	MP3_TRACK(#LED, #InCh, MP3_PLAY_TRACK, 1)	1	^ C1-1			0
					<b>Modul 2 DFPlayer Mini</b>										
✓		4	Rot		startet MF1				! Mono-Flop	MonoFloP(#F21, #InCh, 0.1 Sek)					
✓		MF21	Rot		sendet Select Befehl für Modul 2				Pin MP3-Modul definieren	MP3_SELECT_MODULE(#LED, #InCh, 2)	1	^ C1-1			0
✓		4	Rot		startet MF2				! Mono-Flop	MonoFloP(#F22, #InCh, 1 Sek)					
✓		4	Rot		startet MF3				! Mono-Flop	MonoFloP(#F23, #InCh, 1.1 Sek)					
✓		MF22			logische Verknüpfung der drei MFs				! Logische Verknüpfung	Logic(#F22u23, NOT #InCh AND #F23)					
✓		MF22u23	Rot		spielt Track 1 aus Root von DFPlayer 1 ab				! Titel # abspielen	MP3_TRACK(#LED, #InCh, MP3_PLAY_TRACK, 1)	1	^ C1-1			0
					<b>Modul3 DFPlayer Mini</b>										
✓		5	Rot		startet MF1				! Mono-Flop	MonoFloP(#F31, #InCh, 0.1 Sek)					
✓		MF31	Rot		sendet Select Befehl für Modul 3				Pin MP3-Modul definieren	MP3_SELECT_MODULE(#LED, #InCh, 3)	1	^ C1-1			0
✓		5	Rot		startet MF2				! Mono-Flop	MonoFloP(#F32, #InCh, 1 Sek)					
✓		5	Rot		startet MF3				! Mono-Flop	MonoFloP(#F33, #InCh, 1.1 Sek)					
✓		MF32			logische Verknüpfung der drei MFs				! Logische Verknüpfung	Logic(#F32u33, NOT #InCh AND #F33)					
✓		MF32u33	Rot		spielt Track 1 aus Root von DFPlayer 2 ab				! Titel # abspielen	MP3_TRACK(#LED, #InCh, MP3_PLAY_TRACK, 1)	1	^ C1-1			0
					<b>Welding Modul 1 JQ6500</b>										
✓		MF12u13			Imported_Pattern (pc)				Muster Pattern_Configurator	// Activation: N_ButtonsInCh_to_TapVar1(#InCh, 1)Pattern26	2	1			0

Über parallel angeschlossene Test-LEDs, Stichwort Mini-Verteiler, kann der Ablauf optisch sehr gut verfolgt werden. Andere/kürzere Zeitintervalle für die MonoFlops sind möglich und ggf. durch Tests zu ermitteln.

In Zeile 133 wird über die Variable MF12u13 zeitgleich mit dem dazu gehörigen Geräusch vom Sound-Modul 2 ein Schweißlicht ausgelöst. Geräuschlänge und Länge des Lichts können leicht durch Anpassung des **Schweisslicht** über den Pattern-Configurator angepasst werden.

Damit man den Schweißler nicht immer persönlich wecken muss, hier eine Lösung mit der Zufallsschaltung (Random-Funktion).

Aktiv	Filter	Adresse oder Name	Typ	Startwert	Beschreibung	Verteiler-Nummer	Stecker-Nummer	Leuch	Name	Beleuchtung, Sound, oder andere Effekte	Start LedNr	LEDs	InCh	Loc InCh	LED/ Sound Kanal
					<b>Modul 1 JQ6500</b>										
✓		3	AnAus	0					! Zufallsschaltung 1 Ausgang	Random(WL1, #InCh, RN_NORMAL, 3 sec, 15 sec, 1 sec, 1 sec)					
✓		WL1	Rot		startet MonoFloP (MF) 1				! Mono-Flop	MonoFloP(#F11, #InCh, 0.1 Sek)					
✓		MF11	Rot		sendet Select Befehl für Modul 1				Pin MP3-Modul definieren	MP3_SELECT_MODULE(#LED, #InCh, 1)	1	^ C1-1			0
✓		WL1	Rot		startet MonoFloP (MF) 2				! Mono-Flop	MonoFloP(#F12, #InCh, 1 Sek)					
✓		WL1	Rot		startet MonoFloP (MF) 3				! Mono-Flop	MonoFloP(#F13, #InCh, 1.1 Sek)					
✓		MF12			logische Verknüpfung der drei MFs				! Logische Verknüpfung	Logic(#F12u13, NOT #InCh AND #F13)					
✓		MF12u13	Rot		spielt Track 1 vom JP6500 ab (Welding)				! Titel # abspielen	MP3_TRACK(#LED, #InCh, MP3_PLAY_TRACK, 1)	1	^ C1-1			0
✓		MF12u13			Imported_Pattern (pc)				Muster Pattern_Configurator	// Activation: N_ButtonsInCh_to_TapVar1(#InCh, 1)Pattern26	2	1			0
✓									! LED Nummer manipulieren	// Next_LED(-2)	3	-2			0
					<b>Modul 2 DFPlayer Mini</b>										
✓		4	Rot		startet MF1				! Mono-Flop	MonoFloP(#F21, #InCh, 0.1 Sek)					
✓		MF21	Rot		sendet Select Befehl für Modul 2				Pin MP3-Modul definieren	MP3_SELECT_MODULE(#LED, #InCh, 2)	1	^ C1-1			0
✓		4	Rot		startet MF2				! Mono-Flop	MonoFloP(#F22, #InCh, 1 Sek)					
✓		4	Rot		startet MF3				! Mono-Flop	MonoFloP(#F23, #InCh, 1.1 Sek)					
✓		MF22			logische Verknüpfung der drei MFs				! Logische Verknüpfung	Logic(#F22u23, NOT #InCh AND #F23)					
✓		MF22u23	Rot		spielt Track 1 aus Root von DFPlayer 1 ab				! Titel # abspielen	MP3_TRACK(#LED, #InCh, MP3_PLAY_TRACK, 1)	1	^ C1-1			0
					<b>Modul3 DFPlayer Mini</b>										
✓		5	Rot		startet MF1				! Mono-Flop	MonoFloP(#F31, #InCh, 0.1 Sek)					
✓		MF31	Rot		sendet Select Befehl für Modul 3				Pin MP3-Modul definieren	MP3_SELECT_MODULE(#LED, #InCh, 3)	1	^ C1-1			0
✓		5	Rot		startet MF2				! Mono-Flop	MonoFloP(#F32, #InCh, 1 Sek)					
✓		5	Rot		startet MF3				! Mono-Flop	MonoFloP(#F33, #InCh, 1.1 Sek)					
✓		MF32			logische Verknüpfung der drei MFs				! Logische Verknüpfung	Logic(#F32u33, NOT #InCh AND #F33)					
✓		MF32u33	Rot		spielt Track 1 aus Root von DFPlayer 2 ab				! Titel # abspielen	MP3_TRACK(#LED, #InCh, MP3_PLAY_TRACK, 1)	1	^ C1-1			0

From: <https://wiki.mobaledlib.de/> - MobaLedLib Wiki

Permanent link: [https://wiki.mobaledlib.de/anleitungen/spezial/sound\\_servoplatine?rev=1727427395](https://wiki.mobaledlib.de/anleitungen/spezial/sound_servoplatine?rev=1727427395)

Last update: 2024/09/27 08:56

