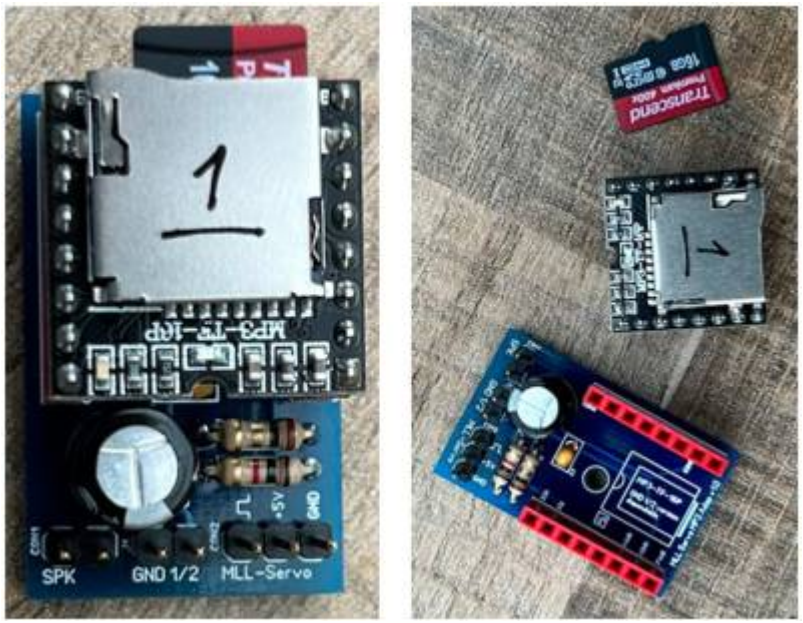
MLL ServoMP3Adapter

Adapter für MP3-TF-16P Modul

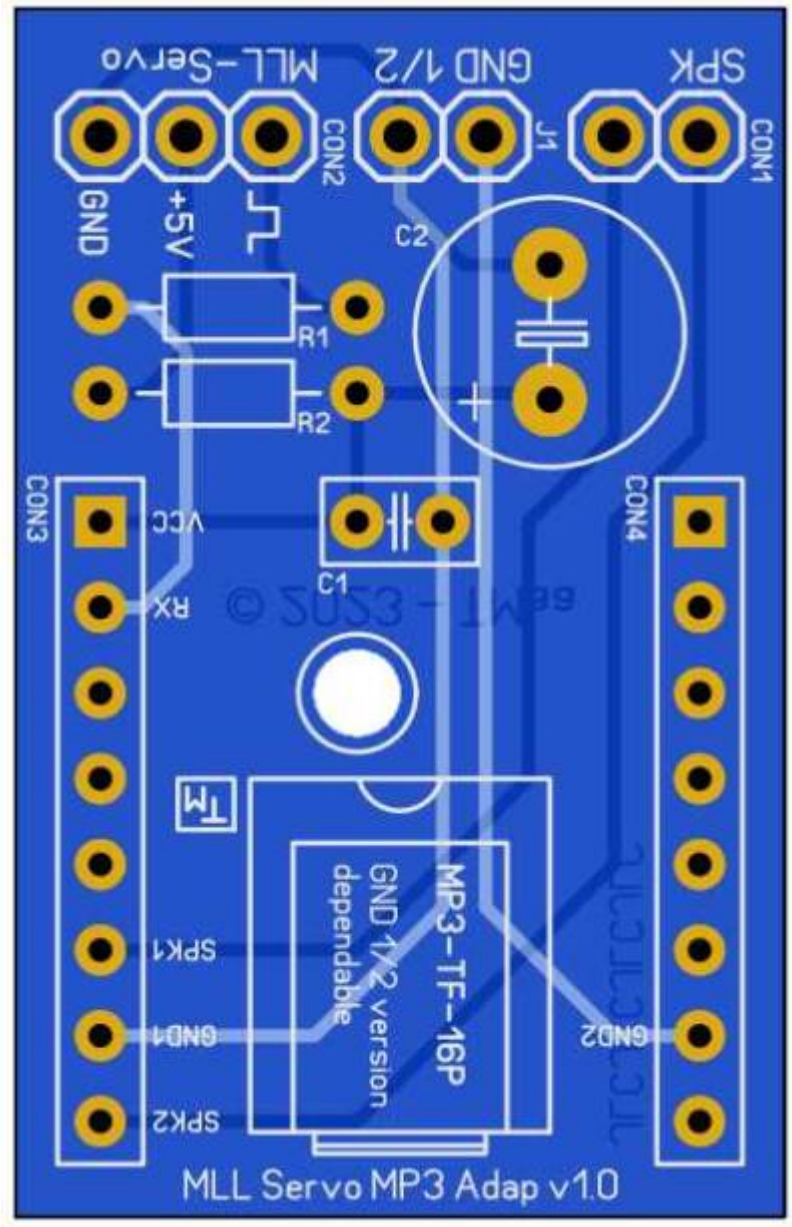
- Für das MLL Digital Output Modul
- Und das Servomodul (510DE - Variante „Servo“)

Für meine Anwendung, Musik über ein MP3-Modul, habe ich eine Schnittstelle erstellt, die eigentlich eine reduzierte Version ist von die 8-fach MP3-Platine (502DE - 8x MP3-Multi-Soundplatine) die ich einfach in Einzelteile zerschnitten habe. Der ursprüngliche KEY_80-Stecker passte nicht zu meinem Modul (Arduino UNO MLL Shield) und ich hatte nur Platz/Bedarf für etwa 3 MP3 Module. (und ich verwende nicht das JQ6500-Musikmodul) Auf die Möglichkeit, ein JQ6500-Modul anzuschließen, habe ich verzichtet, da ich mit diesen Modulen keine so guten Erfahrungen gemacht habe. Ich kann sie selbst zum Laufen bringen, aber ich glaube nicht, dass der durchschnittliche MLL-Benutzer dies überhaupt zum Laufen bringen kann. Entweder sie funktionieren oder sie funktionieren nicht, und um sie wieder zum Laufen zu bringen, müssen alle Hebel in Bewegung gesetzt werden. (YouTube: JQ6500, recover a broken / unreadable module) Da es sich tatsächlich um ein Modul (502DE - 8x MP3-Multi-Soundplatine) handelt das bereits standardmäßig mit dem Program Generator programmiert werden kann, sind für meine Version keine abweichenden Schritte erforderlich. (Danke an Hardi, Jürgen und Dominik)



Das MP3-Modul komplett und in Einzelteilen.
Die „1“ auf dem MP3-Modul ist für mich ein Hinweis darauf, dass es sich um eine MH2024K-24SS-Version handelt.

Lage der Komponenten



Um die Teile zu platzieren, ist es ratsam, immer mit den kleinsten Teilen zu beginnen !
Widerstände > Kondensatoren > Anschlüsse und so weiter …

Platzieren Sie die Komponenten gemäß der Stückliste.

Stückliste v1.0

Anzahl	Bezeichnung	Beschreibung	Bestellnummer	Bemerkungen	Montagereihenfolge
1	Platine	MLL Servo MP3 Adapter	Theo [1)]		

Anzahl	Bezeichnung	Beschreibung	Bestellnummer	Bemerkungen	Montagereihenfolge
1	C1	Vielschicht-Kerko 100 nF, 50/100 V, Z5U 20%, RM 2,5	Z5U-2,5 100N		2
1	C2	Elko, radial, 470µF, 10V, RM3,5, 1000h, 105°C, 20%	NHG-A 470U 10	Achten Sie auf die richtige Ausrichtungsrichtung	5
1	CON1	Stiftleisten 2,54 mm, 1×02, gerade	MPE 087-1-002	Speaker	4
1		Kleinlautsprecher LSF-28M/NT-B, 2W, 8Ohm	LSF-28M/NT-B		-
1	CON2	Stiftleisten 2,54 mm, 1×03, gerade	MPE 087-1-003		4
2	CON3, CON4	Buchsenleisten 2,54 mm, 1×08, gerade	MPE 094-1-008	MP3 Module	3
1		MP3-TF-16P	AliExpress	Achten Sie auf die richtige Ausrichtungsrichtung	-
1	J1	Stiftleisten 2,54 mm, 1×02, gerade	MPE 087-1-002	GND 1/2	4
1		Jumper 2,54 mm, geöffnet, schwarz	MPE 149-1-002-F0	Einige MP3-Module GND2 statt GND1	-
1	R1	Widerstand, Kohleschicht, 1,0 kOhm, 0207, 250 mW, 5%	1/4W 1,0K	braun-schwarz-rot-gold	1
1	R2	Widerstand, Kohleschicht, 1,0 Ohm, 0207, 250 mW, 5%	1/4W 1,0	schwarz-braun-schwarz-gold	1

Gerber-Dateien

- Zur Eigenfertigung der Leiterplatten, z. B. bei JLCPCB, stehen die Gerber-Dateien zur Verfügung:

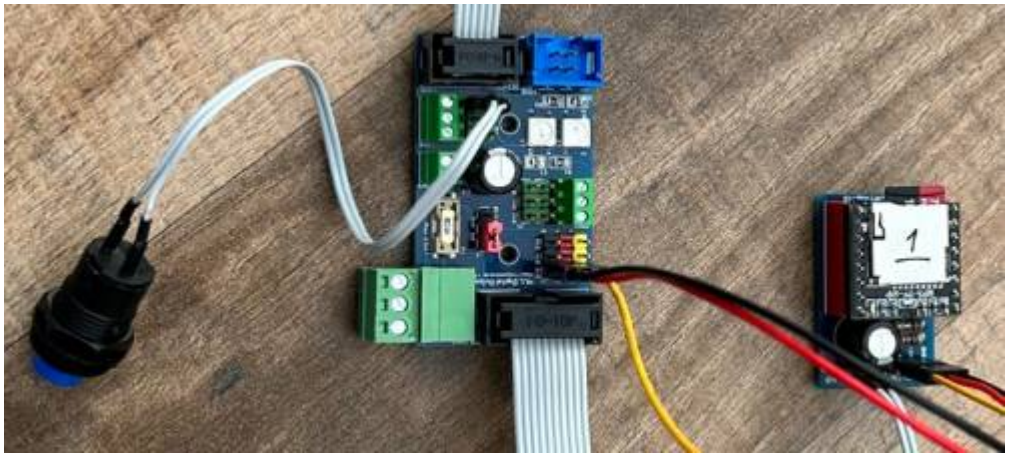
MLL_Servo_MP3_Adap_v10.zip

Erläuterung / Beispiel

MLL UNO Shield, Digital Output und Servo MP3 Adapter

Beim Anschluss des MP3-Moduls an die MLL Digital Output Erweiterung kann dieses mit dem Program Generator eingestellt werden, als Beispiel für das MP3-Modul auf D10, D11 oder D12, dies geschieht mit dem Befehl: **SOUND_CHANNEL_DEFINITON(10, MP3-TF-16P)** Wenn dies der erste Befehl ist, erhält das Modul (angeschlossen an Pin **D10**) den Namen **S0**. Wenn Sie mehrere Module verbinden möchten, muss dieser Befehl für die anderen Module wiederholt werden.
SOUND_CHANNEL_DEFINITON(11, MP3-TF-16P) dito, aber für ein Modul an Pin **D11**, das dann **S1** heißt ... usw.

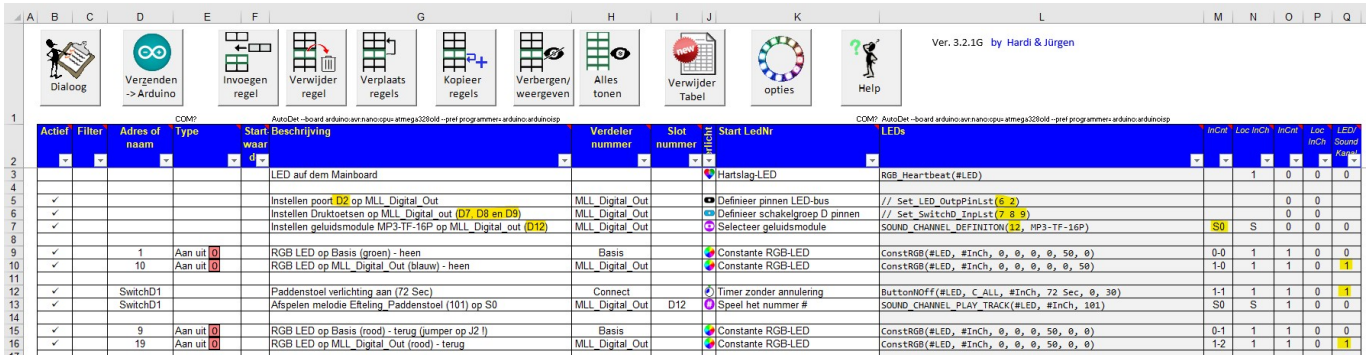
In diesem Beispiel habe ich einen Taster an Pin D7 = Switch**D**1 (Eselsbrücke „**D**“ = Direkt = Schalter auf Hauptplatine) und ein Musikmodul an die farbige Stiftleiste D12 angeschlossen. An den MLL-Bus (LED kanal 1) habe ich auch einen sogenannten Schokoladentafel angeschlossen, an dem drei LEDs den Pilz beleuchten.



Hier auf dem Foto ist das MP3-Modul an D12 sowie einen Druckknopf an D7 mit dem MLL Digital Output-Modul verbunden

NB! ... Ich habe vorübergehend einen Jumper auf J2 auf dem Arduino UNO MLL Shield gesetzt, um alles zu testen, sodass die „Heartbeat“-LED (L6), die vom Return kommt, direkt nach der „Heartbeat“-LED (L7) kommt. Dies dient dazu, zu testen, ob der MLL-Schaltkreis geschlossen ist und alles ordnungsgemäß funktioniert. An den MLL-Bus des Arduino UNO MLL Shield darf dann nichts angeschlossen werden. Vergessen Sie nicht, anschließend den Jumper auf J2 zu entfernen, bevor Sie zum Normalbetrieb zurückkehren !

Und hier ist das Beispielprogramm, das ich erstellt habe. Verwende eine SD-micro-Karte, die nicht größer als max. 32 GB ist. Größer ... dann kommt das MP3-Modul nicht damit zurecht. Um alles zu testen, habe ich 101 MP3-Sounds auf die Karte gelegt. Ich wollte wissen, wie weit mann gehen kann. Die Musikdatei, die ich dann abspiele, ist Melodie Nummer 101. Wichtig zu wissen ist, dass die erste kopierte Datei auf der SD-micro-Karte auch die erste Datei ist, auf die zugegriffen wird (unabhängig von der alphabetischen Reihenfolge), die zweite abgelegte Datei ist Nummer 2 und so weiter.



Dialoog | Verzenden -> Arduino | Invoegen regel | Verwijder regel | Verplaats regels | Kopieer regels | Verbergen/ weergeven | Alles tonen | Verwijder Tabel | opties | Help | Ver. 3.2.1G by Hardi & Jürgen

Actief	Filter	Adres of naam	Type	Start-waard	Beschrijving	Verdeler nummer	Slot nummer	Richt	Start LedNr	LEDs	InCnt	Loc InCh	InCnt	Loc InCh	LED/ Sound Kanal
					LED auf dem Mainboard				Hartslag-LED	RGB_Heartbeat(#LED)		1	0	0	0
✓					Instellen poort D2 op MLL_Digital_Out	MLL_Digital_Out			Definieer pinnen LED-bus	// Set_LED_OutpPinLst(6 2)			0	0	
✓					Instellen Druktoetsen op MLL_Digital_out (D7, D8 en D9)	MLL_Digital_Out			Definieer schakelgroep D pinnen	// Set_SwitchD_InpLst(7 8 9)			0	0	
✓					Instellen geluidsmodule MP3-TF-16P op MLL_Digital_out (D12)	MLL_Digital_Out			Selecteer geluidsmodule	SOUND_CHANNEL_DEFINITON(12, MP3-TF-16P)	S0	S	0	0	0
✓		1	Aan uit 0		RGB LED op Basis (groen) - heen	Basis			Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 0, 50, 0)	0-0	1	1	0	0
✓		10	Aan uit 0		RGB LED op MLL_Digital_Out (blauw) - heen	MLL_Digital_Out			Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 0, 0, 50)	1-0	1	1	0	1
✓		SwitchD1			Paddenstoel verlichting aan (72 Sec)	Connect			Timer zonder annulering	ButtonNOff(#LED, C_ALL, #InCh, 72 Sec, 0, 30)	1-1	1	1	0	1
✓		SwitchD1			Afspelen melodie Efteling_Paddenstoel (101) op S0	MLL_Digital_Out	D12		Speel het nummer #	SOUND_CHANNEL_PLAY_TRACK(#LED, #InCh, 101)	S0	S	1	0	0
✓		9	Aan uit 0		RGB LED op Basis (rood) - terug (jumper op J2 !)	Basis			Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 50, 0, 0)	0-1	1	1	0	0
✓		19	Aan uit 0		RGB LED op MLL_Digital_Out (rood) - terug	MLL_Digital_Out			Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 50, 0, 0)	1-2	1	1	0	1

Die „gelben“ Markierungen beziehen sich auf die jeweiligen Anschlüsse und die dafür festgelegten Softwareeinstellungen.

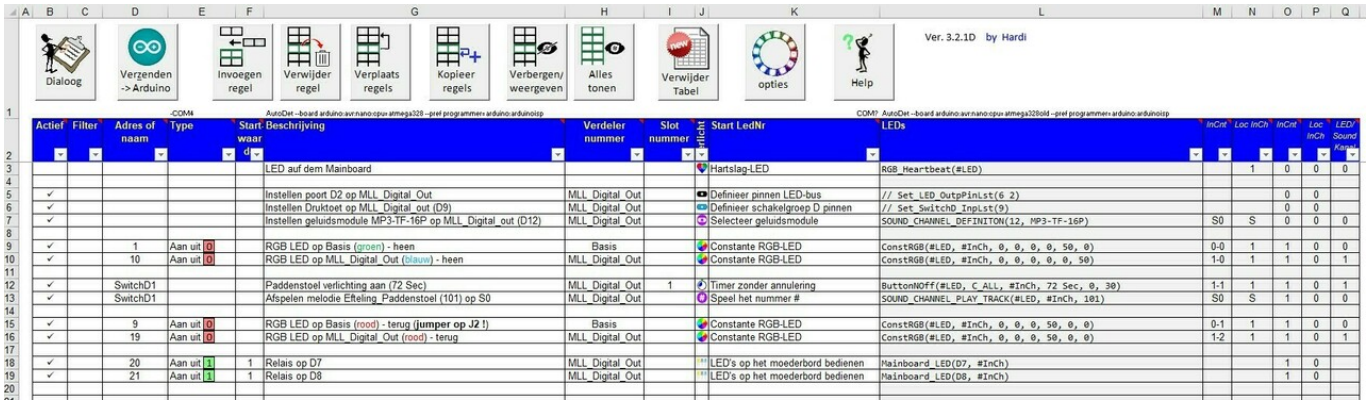
Großes Finale

Licht, Musik, Eingang (Taster) und Ausgang (Relais)

Ich habe auch etwas experimentiert und wurde durch etwas ausgelöst, das mich auf die Idee brachte, zu versuchen, herauszufinden, ob ich ein Relais über die neue MLL Digital-Output-Platine mit „einfachen“ MLL-Befehlen steuern kann.

Und das gelang mir, indem ich den Befehl verwendete, der normalerweise die LEDs auf der Hauptplatine steuert : **Mainboard_LED(D7, #InCh)** Dieser Befehl bietet die Möglichkeit anzugeben, an welchem Pin/Port die betreffende LED angeschlossen ist. Und in diesem Fall keine physische LED, sondern die LED in einem Optokoppler, der wiederum ein Relais aktiviert.

Hoffentlich verdeutlicht das Beispiel die Sache.

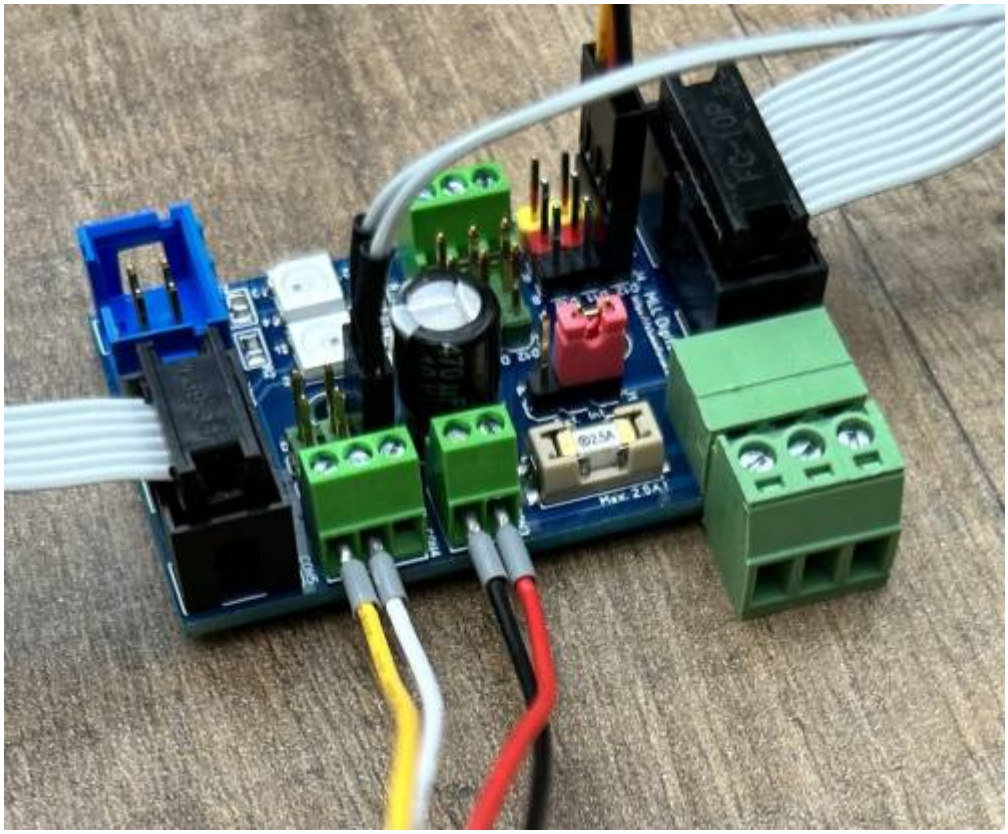


Actief	Filter	Adres of naam	Type	Start waar d	Beschrijving	Verdeler nummer	Slot nummer	Start LedNr	LEDs	InCnt	Loc InCh	InCnt	Loc InCh	LED/ Sound Kanaal
					LED auf dem Mainboard			Hartslag-LED	RGB_Heartbeat(#LED)		1	0	0	0
✓					Instellen poort D2 op MLL_Digital_Out	MLL_Digital_Out		Definieer pinnen LED-bus	// Set_LED_OutpPinLst(6 2)			0	0	
✓					Instellen Druktoet op MLL_Digital_out (D9)	MLL_Digital_Out		Definieer schakelgroep D pinnen	// Set_SwitchD_InpLst(9)			0	0	
✓					Instellen geluidsmodule MP3-TF-16P op MLL_Digital_out (D12)	MLL_Digital_Out		Selecteer geluidsmodule	SOUND_CHANNEL_DEFINITON(12, MP3-TF-16P)	S0	S	0	0	0
✓		1	Aan uit 0		RGB LED op Basis (groen) - heen	Basis		Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 0, 50, 0)	0-0	1	1	0	0
✓		10	Aan uit 0		RGB LED op MLL_Digital_Out (blauw) - heen	MLL_Digital_Out		Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 0, 0, 50)	1-0	1	1	0	1
✓		SwitchD1			Paddenstoel verlichting aan (72 Sec)	MLL_Digital_Out	1	Timer zonder annulering	ButtonNOff(#LED, C_ALL, #InCh, 72 Sec, 0, 30)	1-1	1	1	0	1
✓		SwitchD1			Afspelen melodie Efteling_Paddenstoel (101) op S0	MLL_Digital_Out		Speel het nummer #	SOUND_CHANNEL_PLAY_TRACK(#LED, #InCh, 101)	S0	S	1	0	0
✓		9	Aan uit 0		RGB LED op Basis (rood) - terug (**jumper op J2 !**)	Basis		Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 50, 0, 0)	0-1	1	1	0	0
✓		19	Aan uit 0		RGB LED op MLL_Digital_Out (rood) - terug	MLL_Digital_Out		Constante RGB-LED	ConstRGB(#LED, #InCh, 0, 0, 0, 0, 50, 0, 0)	1-2	1	1	0	1
✓		20	Aan uit 1	1	Relais op D7	MLL_Digital_Out		LED's op het moederbord bedienen	Mainboard_LED(D7, #InCh)			1	0	
✓		21	Aan uit 1	1	Relais op D8	MLL_Digital_Out		LED's op het moederbord bedienen	Mainboard_LED(D8, #InCh)			1	0	

Einige Fakten: (Siehe Foto unten)
Relais 1 ist mit D7 (Gelb) am MLL Digital-Output-Platine verbunden, Relais 2 mit D8 (Weiß) und ein Druckknopf mit D9. Der Pushbutton wurde ebenfalls auf Pin D9 umdefiniert und kann mit SwitchD1 aufgerufen werden, da er der erste und „einzige“ ist, der mit dem Befehl :**// Set_SwitchD_InpLst(9)**, Zeile 6 in Excel, gesetzt wird. Für D7 und D8 sind dies die Zeilen 18 und 19. D10 und D11 wurden nicht genutzt, der MP3-Player befindet sich jedoch auf D12. (Zeile 7) Die Versorgungsspannung für beide Relais wird ebenfalls von der MLL Digital-Output-Platine bezogen. (Rot ist +5V und Schwarz ist GND) In der mit dem Program Generator erstellten Datei wird durch Drücken des Tasters (auf D9) eine Melodie abgespielt und die LEDs im an Ausgang 2 angeschlossenen Pilz leuchten auf.

Auf der Hauptplatine befinden sich auch D7, D8 und D9, das hat hierauf aber keinen Einfluss. Es versteht sich von selbst, dass die Drucktasten auf dieser Platine dann nicht genutzt werden!

(Ausdruck auf eigene Gefahr 😎)



Beide Relais können ebenso wie die Heartbeats von Ausgang 1 (LED-Kanal 0) und 2 (LED-Kanal 1) sowie die Return-Beats mit DCC-Befehlen oder vom Program Generator aus gesteuert werden.

Eine Kuriosität besteht darin, dass bei der Steuerung des Relais eine „1“ das Relais ausschaltet und eine „0“ es einschaltet. (Daher habe ich im entsprechenden Befehl als Standard-Startwert eine „1“ eingetragen, damit diese beim Start deaktiviert sind)

Aus der kleinen Leiterplatte ist ein **Mehrzweckmodul** geworden... Musik, Schalter oder Taster, Relais, zusätzliche externe Stromversorgung mit Sicherung und ein zweiter LED-Kanal mit Heartbeat.

Gibt es Fragen oder Anmerkungen?

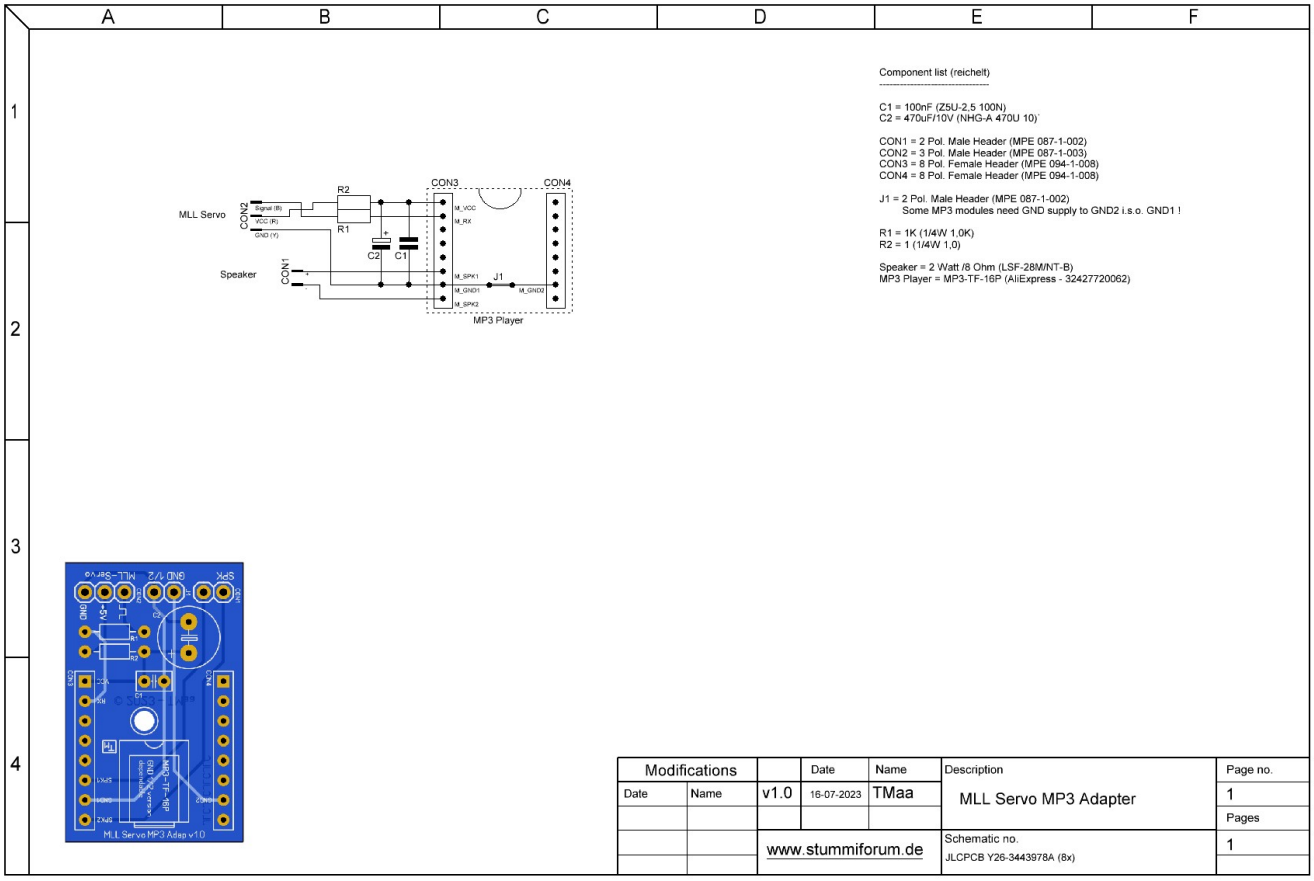
Klicken Sie einfach auf den Link und Sie werden auf die entsprechende Seite weitergeleitet, auf der Sie diese Fragen und Kommentare melden können.

Zum Forum - MLL ServoMP3Adapter

Geben Sie im Problemfall bitte möglichst viele Informationen an: Worin besteht das Problem, wie wurde es bemerkt und wann ist es aufgetreten. Fotos (bei Hardwareproblemen Fotos von der Ober- und Unterseite der Leiterplatte) sind hilfreich und bei Steuerungsproblemen stellen Sie bitte einen Screenshot des betreffenden Problems aus dem Program_Generator zur Verfügung.

Aber wenn Sie einen Tipp für uns haben, würden wir uns auch darüber freuen.

Schaltplan



1)

Anfrage für eine Platine kann per PN an Theo gesendet werden (**TMaa**)

From:
https://wiki.mobaledlib.de/ - **MobaLedLib Wiki**

Permanent link:
https://wiki.mobaledlib.de/spezial/user/theo/servomp3adapter_tmaa

Last update: **2025/07/03 11:22**