Einführung in die Funktionen vom Charlieplexing/Multiplexing

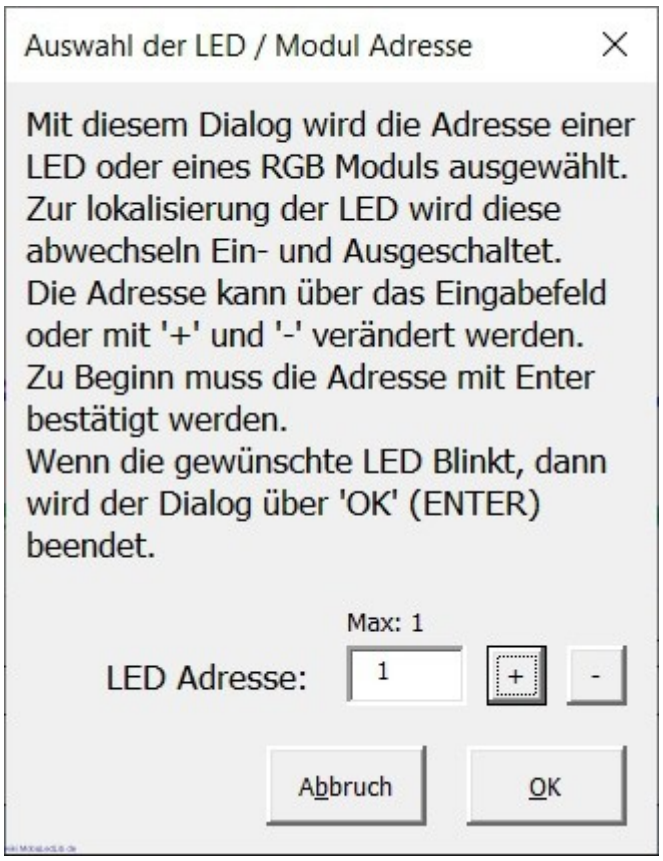
Hallo alle zusammen.
Hardi, der Erfinder und Programmierer der MobaLedLib, hat im Stummiforum, für alle Freunde und Mitbastler,
eine wunderbare Anleitung zum Thema „Charlieplexing“ geschrieben.

Hallo Charlieplexer,

Ich habe mal versucht ein Video zu drehen welches zeigt wie man eine Konfiguration zum Charlieplexing Modul schickt, weil es immer wieder Kollegen gibt die damit Probleme haben.

Nachdem der ATTiny das erste Mal über den TinyUniProg programmiert wurde enthält der EEPROM des ATtiny zunächst keine Konfiguration. Darum Blinken die an das Modul angeschlossenen LEDs. Das zeigt uns, dass das Programm auf dem ATtiny richtig läuft. Das sieht man gleich zu beginn in dem Video. Die LEDs blinken u.U. nicht wenn der ATTiny vorher schon mal konfiguriert wurde. Dann wird die entsprechende Konfiguration gezeigt. Diese kann auch so definiert sein, dass normalerweise keine LEDs leuchten ;-(Achtung: Die Daten im EEPROM werden nicht gelöscht, wenn das Programm nochmal per TinyUniProg hochgeladen wird.

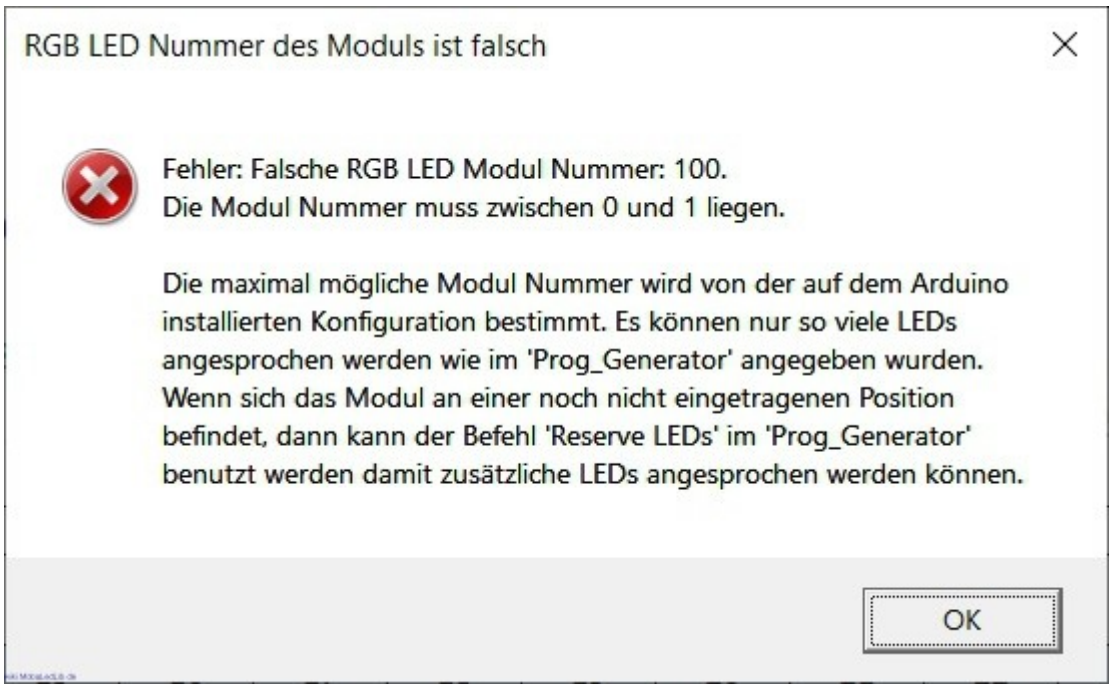
Die im Pattern_Configurator erstellte Konfiguration wird mit dem „Zum Modul schicken“ Knopf zum Charlieplexing Modul geschickt. Dazu muss die „RGB Modul Nummer“ passen. Wenn man eine WS2812 auf der Hauptplatine hat und das Charlieplexing Modul das erste angeschlossene Modul ist, dann muss hier eine 1 stehen. Wenn das Modul an einer anderen Stelle in der Kette steckt, dann trägt man bei „RGB Modul Nummer“ die entsprechende Nummer ein. Wenn die „RGB Modul Nummer“ leer ist, dann erscheint automatisch dieser Dialog:



Damit kann man „ausprobieren“ welche Modul Adresse das entsprechende Charlieplexing Modul hat. Das ist besonders dann interessant, wenn das Modul bereits auf der Anlage verbaut ist. Mit den „+“ und „-„ Tasten kann man eine Adresse auswählen. Die über diese Adresse angesprochene LED blinkt dann. So findet man die gewünschte Adresse einfach. Dabei hilft die auf dem Charlieplexing Modul vorhandene blaue LED welche anstelle des Widerstands R10 bestückt wird (Das lange Beinchen der LED zeigt zum Rand der Platine. R10 wird nur beim Servo Modul benötigt).

Der Dialog erlaubt nur Modul Adressen welche der LED Arduino kennt. Das bedeutet, dass auf dem LED Arduino bereits eine Konfiguration installiert sein muss bei dem die entsprechende LED benutzt wird. Zum Test kann man die Konfiguration „Charlie_But_Bin“ welche man im Pattern_Configurator unter „Optionen/Beispiele/Charlieplexing“ findet zum „Programm Generator“ schicken.

Wenn der LED Arduino zu falsch konfiguriert ist bekommt man das angezeigt:

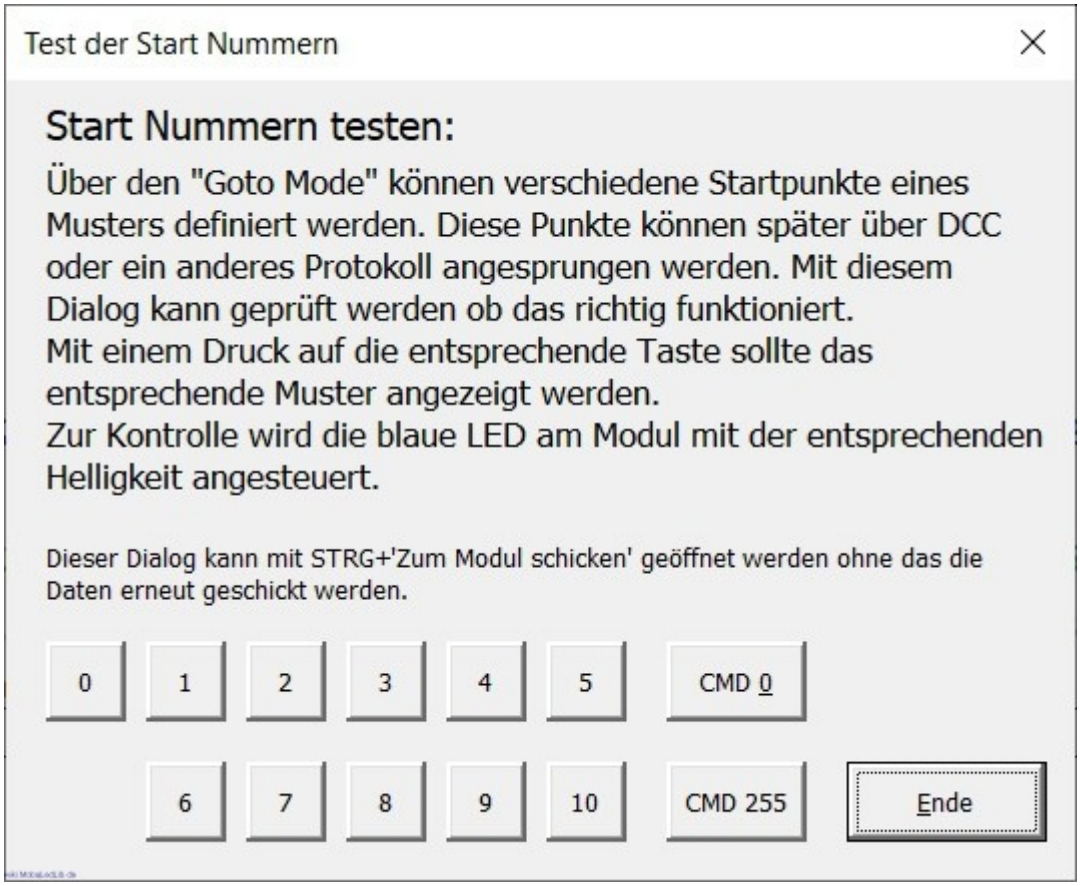


Ist die erste Hürde überwunden und die Übertragung wurde gestartet, dann verändert die blaue LED wie im Video gezeigt ihre Helligkeit während die Daten zum ATtiny geschickt werden. Gleichzeitig flackern die am Modul angeschlossenen LEDs. Im Video sieht man das leider nicht so richtig, weil meine Kamera nur 30 Bilder pro Sekunde kann ;-(
Wenn man keine blaue LED auf dem Board eingelötet hat dann kann man zum Test den ATtiny ausstecken und anstelle dessen eine LED in die Pins 3 und 4 stecken (Langes Beinchen in 3). Diese LED sollte leuchten solange keine Daten kommen (Test von R9) und ihre Helligkeit ändern, wenn auf „Zum Modul schicken“ geklickt wird. Die LED zeigt, dass die Daten ankommen und dass der WS2811 funktioniert.

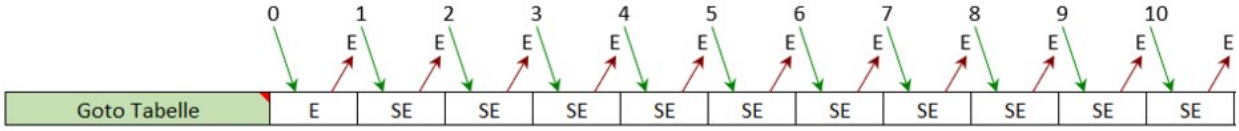
Die an das Charlieplexing Modul angeschlossenen LEDs müssen bei der Datenübertragung flackern, also ganz schnell und scheinbar zufällig ihre Helligkeit ändern. Das ist ein Zeichen, dass der ATtiny Daten bekommt und diese erkannt hat.

Nach einer gewissen Zeit sind die Daten übertragen und das Übertragungsfenster wird geschlossen. Der LED-Bus ist leider eine Einbahnstraße. Das bedeutet, dass das Charlieplexing Modul keine Antwort zum Pattern_Configurator schicken kann das die Übertragung richtig angekommen ist. Darum lasse ich die LEDs im Fehlerfall blinken. Damit wird angezeigt, dass keine oder ungültige Daten im EEPROM des ATTinys stehen.

Für das Video habe ich die „Charlieplex.LED Test“ Konfiguration verändert so dass ein Lauflicht generiert wird. Bei dem original Beispiel ist zusätzlich der „Goto Mode“ aktiviert. Dann kann man jede LED einzeln schalten. Nach dem hochladen der Konfiguration erscheint dann dieser Dialog:



Damit kann sofort überprüft werden ob der ATtiny richtig auf Befehle reagiert. Die Befehle werden ihm (wie könnte es anders sein) per Helligkeitswerte übertragen. Die Helligkeit 15 entspricht dem Befehl 0, 35 dem Befehl 1, 55 dem Befehl 3, ... Diese Werte findet man in der Konfiguration „Charlie_But_Bin“ wieder:

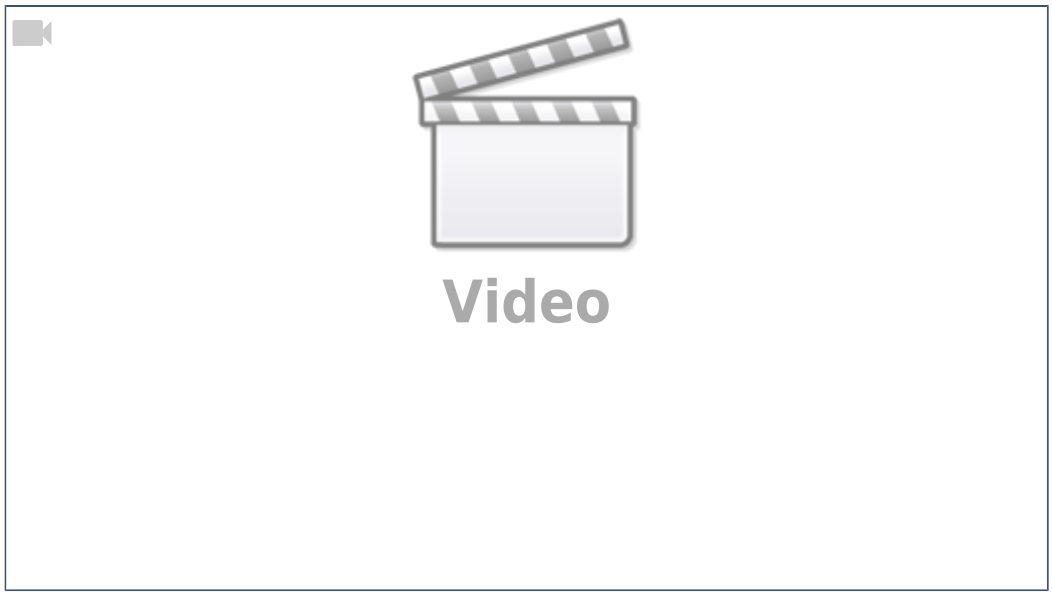


\+ - RGB LED

LED Nr	Spalte Nr ->	1	2	3	4	5	6	7	8	9	10	11
1	Red	15	35	55	75	95	115	135	155	175	195	215
2	Green	15	35	55	75	95	115	135	155	175	195	215
3	Blue (Status)	15	35	55	75	95	115	135	155	175	195	215

Die Werte größer 215 sind für die Programmierung reserviert. [color=#FF0000][b]Alle Klarheiten beseitigt?[/b][/color]

Nachdem ihr alles brav gelesen und auswendig gelernt habt dürft Ihr jetzt auch ein bisschen Video schauen:



Quelle: https://www.stummiforum.de/viewtopic.php?f=7&t=165060&sd=a&start=3191

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

Permanent link:
https://wiki.mobaledlib.de/anleitungen/spezial/charlieplexing1?rev=1602866484

Last update: **2020/10/16 17:41**