

# Test der MobaLedLib Hauptplatine

Zum Testen der Hauptplatine kann die Konfiguration „Mainboard\_Hardware\_Test.MLL\_pgf“ verwendet werden. Diese enthält die sieben Seiten zum testen der Hauptplatine und der PushButton-Platine. Mit dem Test können die meisten Funktionen der Platine überprüft werden. Zum Test benötigt man ein LED-Panel mit 64 LEDs, die Pushbutton Platine und möglichst eine Matrix mit 80 Tastern.



Die Tests der Hauptplatine stehen erst ab der Version 1.9.6J der MobaLedLib zur Verfügung. Manche Test sind nur möglich/sinnvoll wenn die entsprechenden Komponenten auf der Hauptplatine bestückt sind.

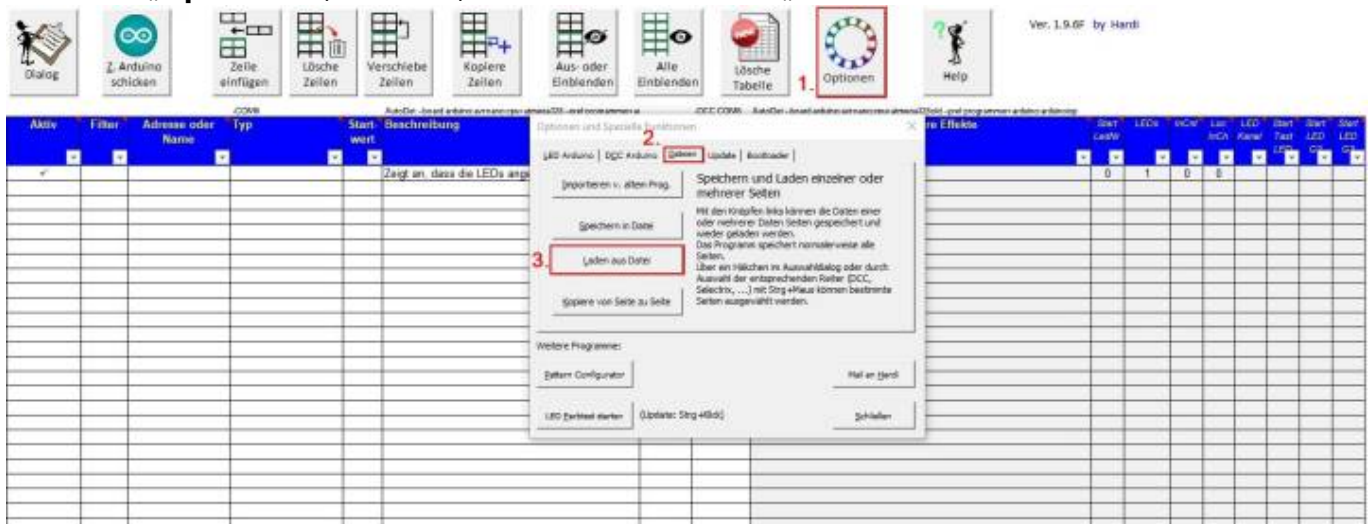
Die Tests zur Prüfung der Verbindung (DCC, MCAN-Bus oder Selectrix für das Andreaskreuz können nur mit der jeweiligen Digitalzentrale probiert werden. Die anderen Tests der Seite funktionieren auch ohne die entsprechenden Zentralen.



Bei einem Wechsel der Testseite, muss der jeweilige Test erst auf den Arduino hochgeladen werden, damit dieser auch zur Verfügung steht.

## Importieren der Testseiten

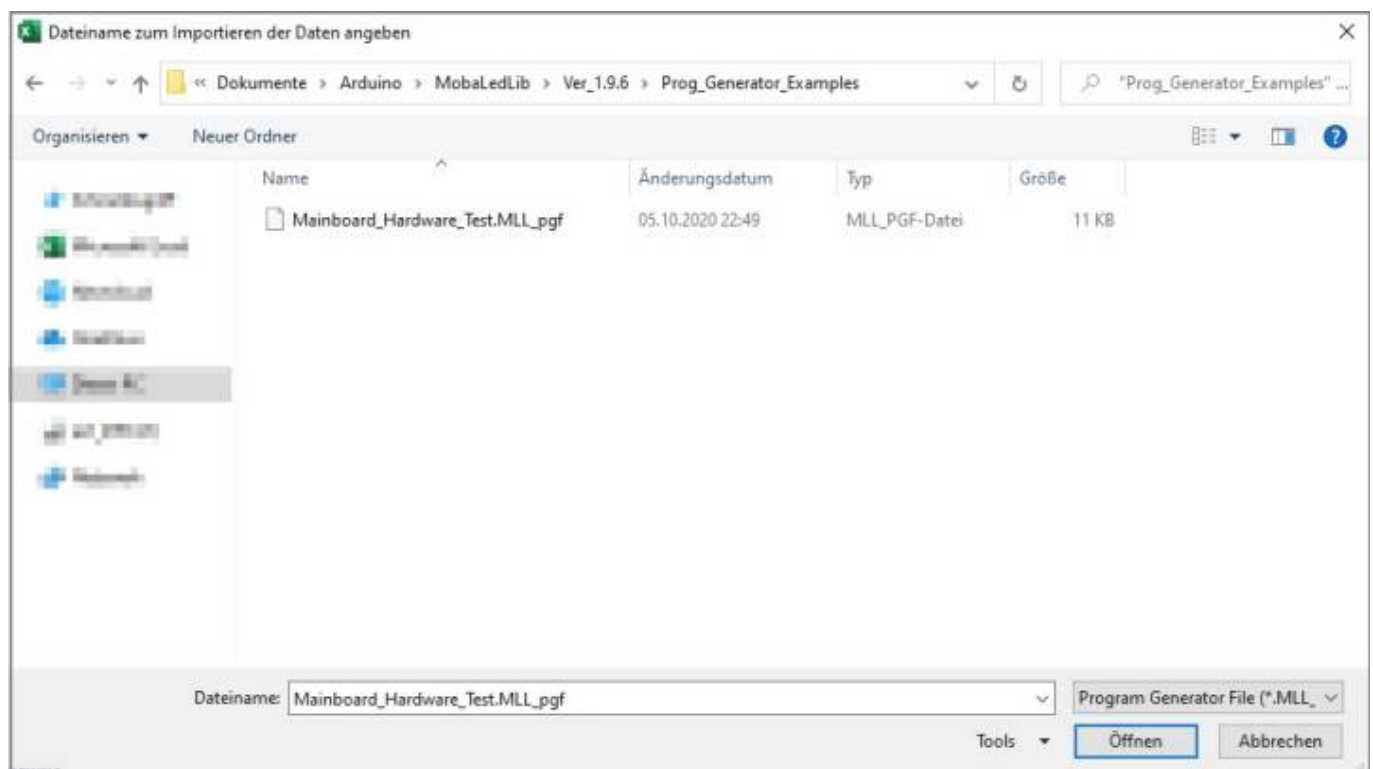
Das Importieren der Testseiten erfolgt über das Optionenmenü im Programm\_Generator. Dazu oben rechts auf „**Optionen**“ (LED-Kreis) klicken und den Reiter „**Dateien**“ auswählen.



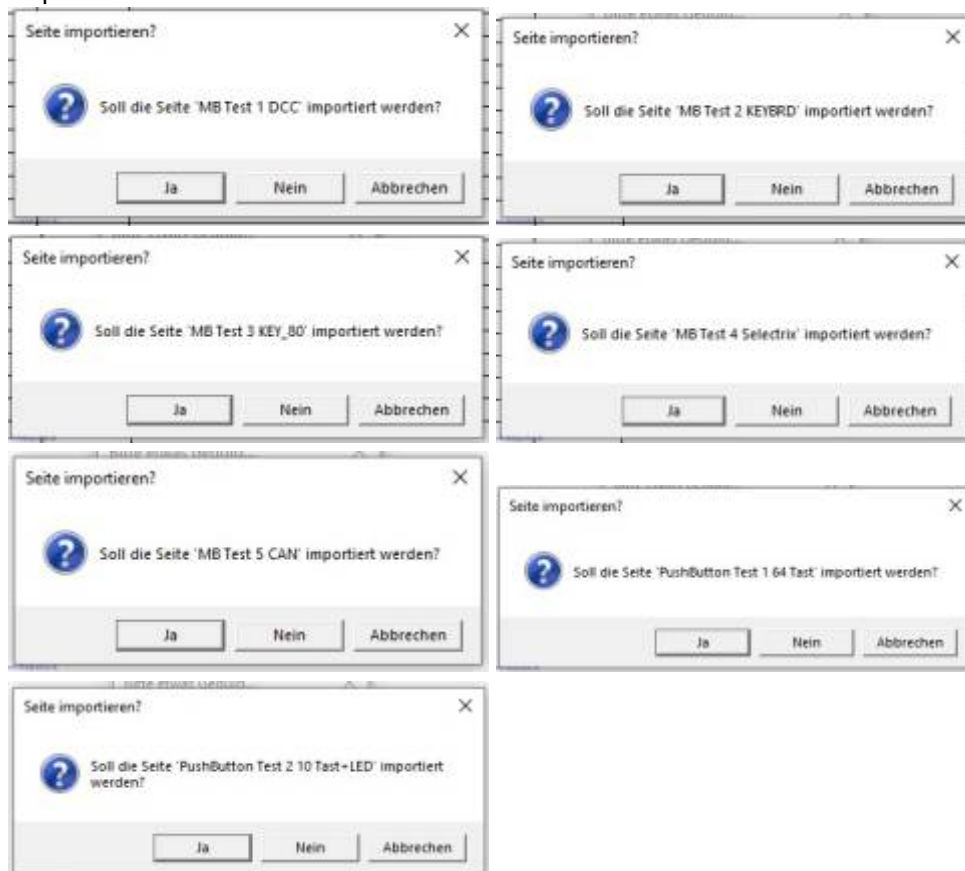
In dem Explorerfenster nun den Ordner

„C:\Users\BENUTZERNAME\Documents\Arduino\MobaLedLib\Aktuelle

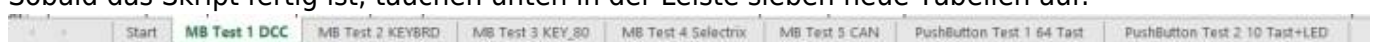
Versionsnummer\Prog\_Generator\_Examples“ suchen und öffnen. Darin befindet sich die Datei „Mainboard\_Hardware\_Test.MLL\_pgf“. Diese auswählen und auf „**Öffnen**“ klicken.



Direkt nach dem Öffnen kommen nacheinander sieben Dialogfenster, welche fragen, ob man die Seiten importieren möchte. Hier bitte immer „Ja“ auswählen, andernfalls werden die Seiten nicht importiert.



Sobald das Skript fertig ist, tauchen unten in der Leiste sieben neue Tabellen auf.



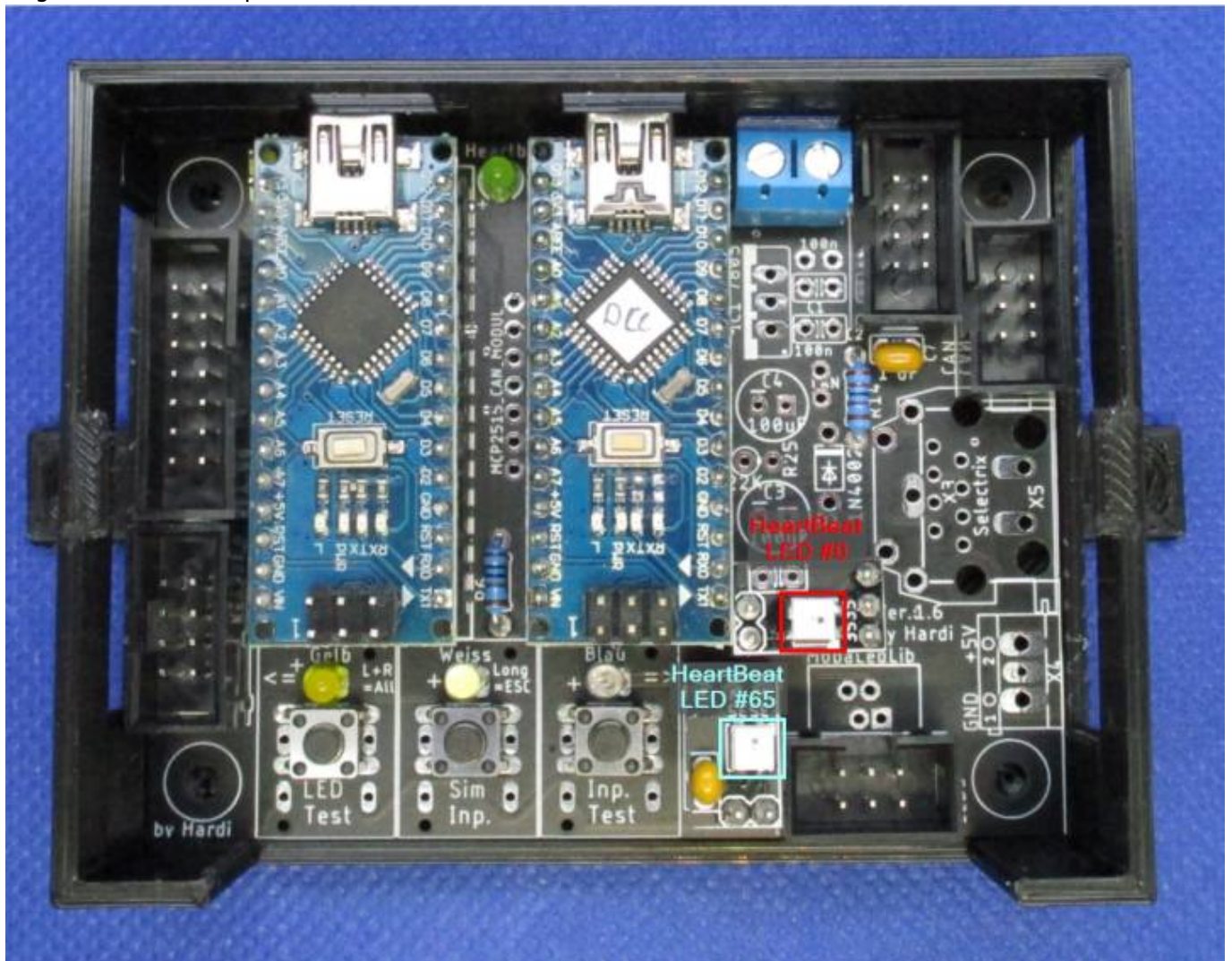
## MB Test 1 - DCC

Man beginnt mit der Seite „DCC Mainboard Test“ und installiert den neuen Bootloader auf den LED Arduino. (Optionen/Bootloader). Damit überprüft man die Verbindungen zwischen den beiden Arduinos

Anschließend spielt man die Konfiguration der „**MB Test 1 - DCC**“ Seite auf den LED Arduino. Im Anschluss kam die folgenden Tests machen

## HeartBeat

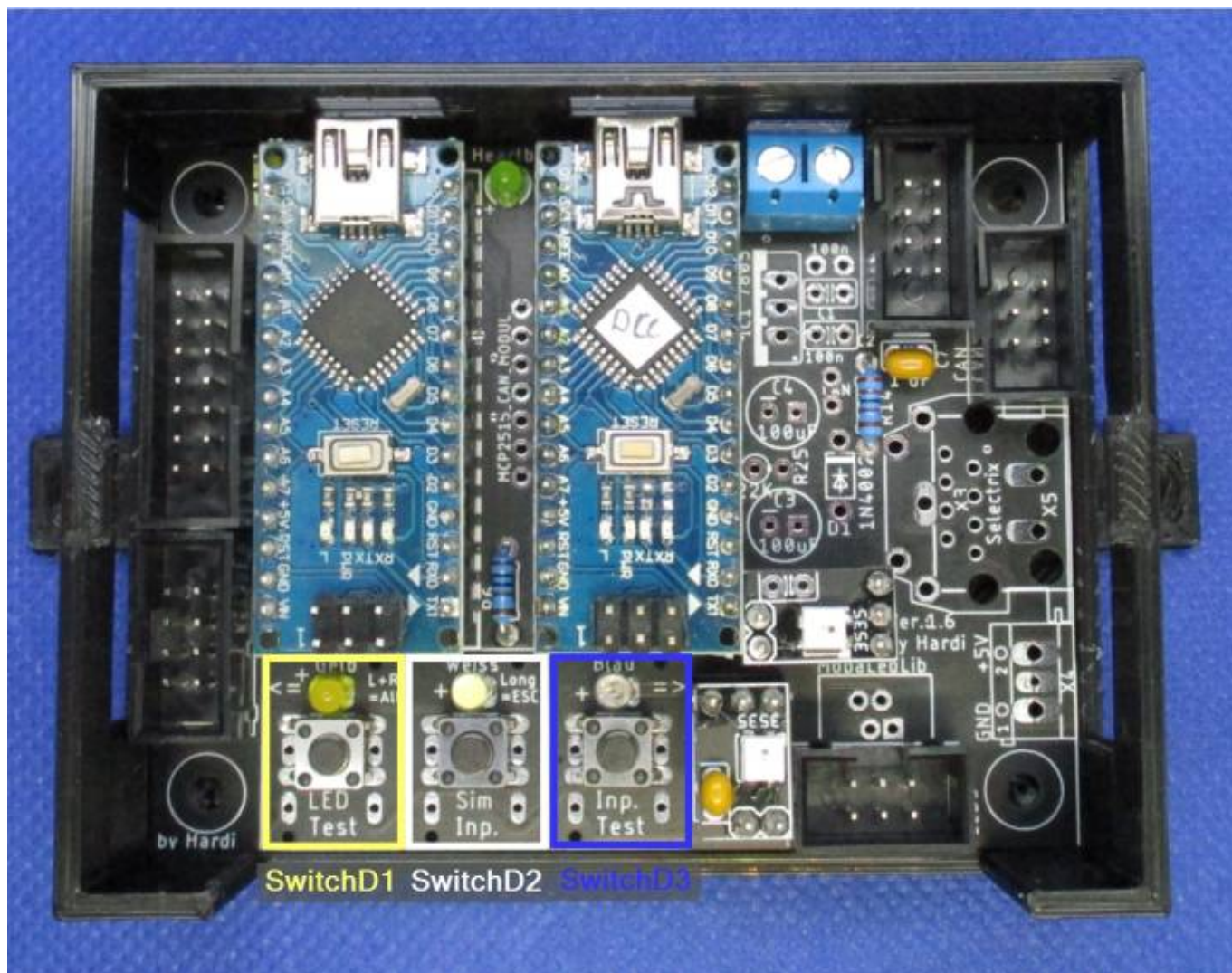
Die HeartBeat RGB LED (rot markiert) auf der Hauptplatine pulsiert farbig. Wenn genau 64 LEDs angeschlossen sind pulsiert auch die letzte LED (türkis markiert) der Kette auf dem Mainboard.

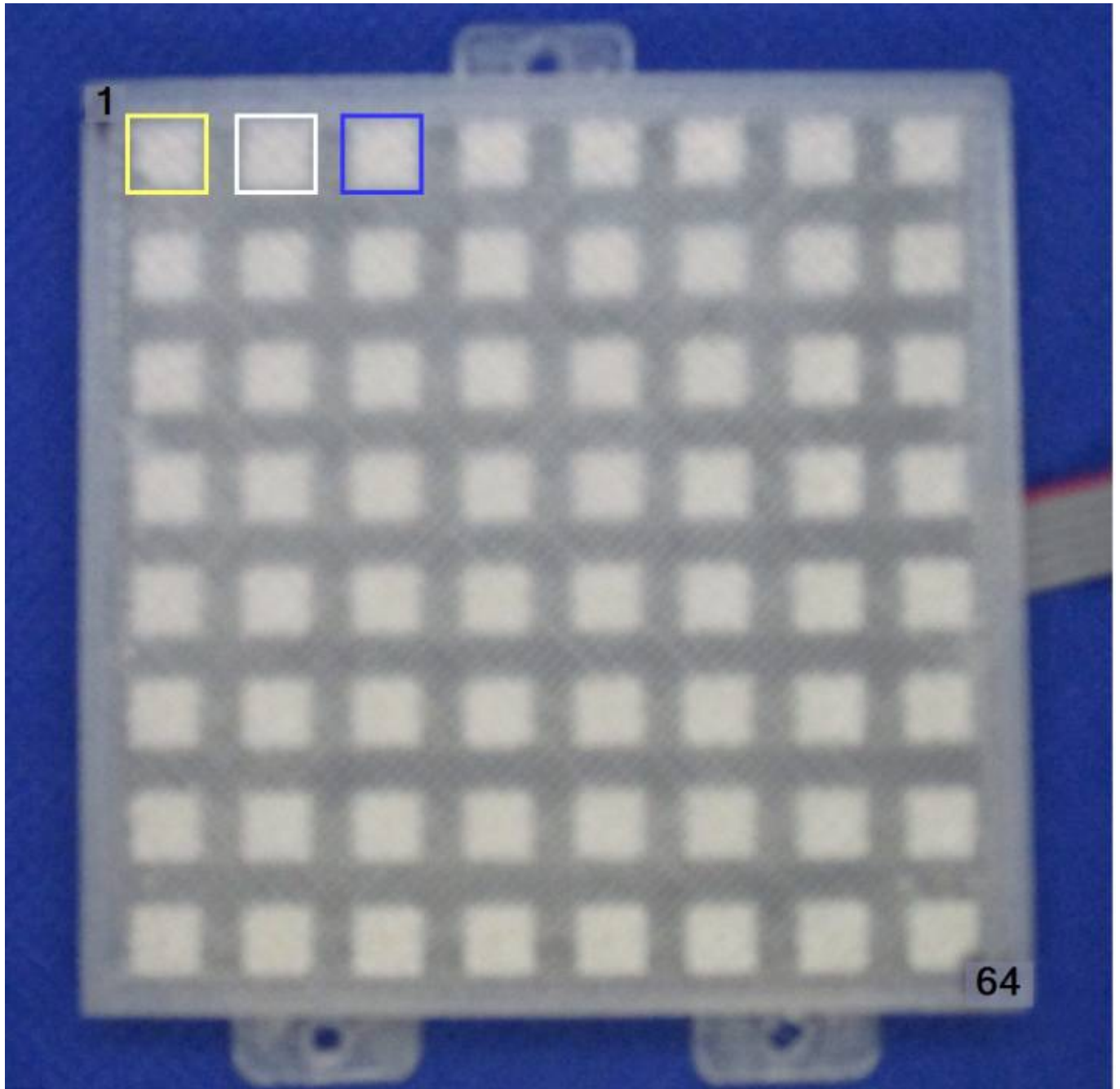




## Tastererkennung onBoard

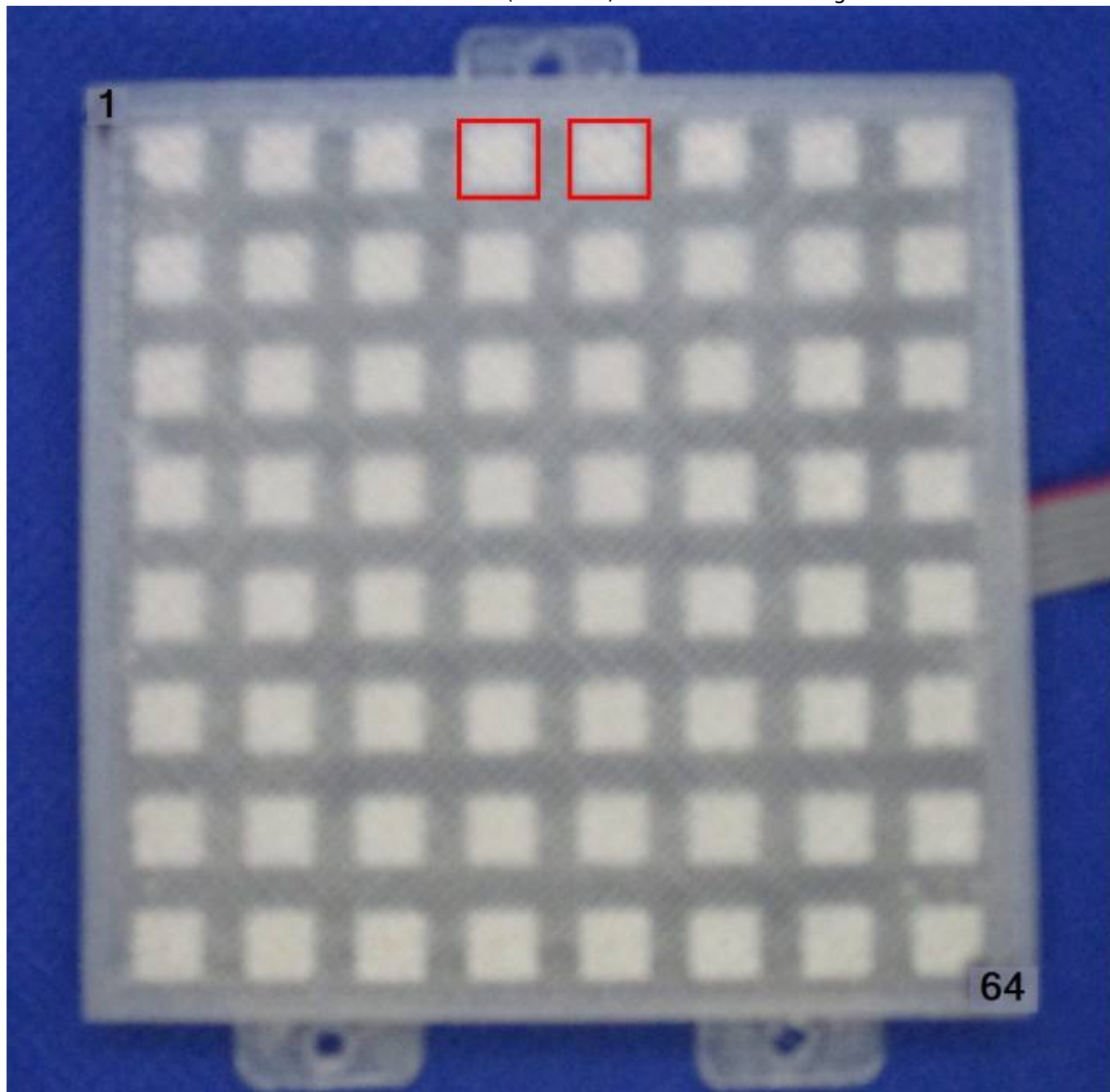
Mit den Tastern auf dem Mainboard können die LEDs über den Tastern geschaltet werden. Damit wird auch das Blinken der ersten drei LEDs des 64er Panels aktiviert.





## DCC-Verbindungstest

Mit der DCC Adresse 1 wird ein Andreaskreuz (LED 4&5) aktiviert und es fängt an zu blinken.



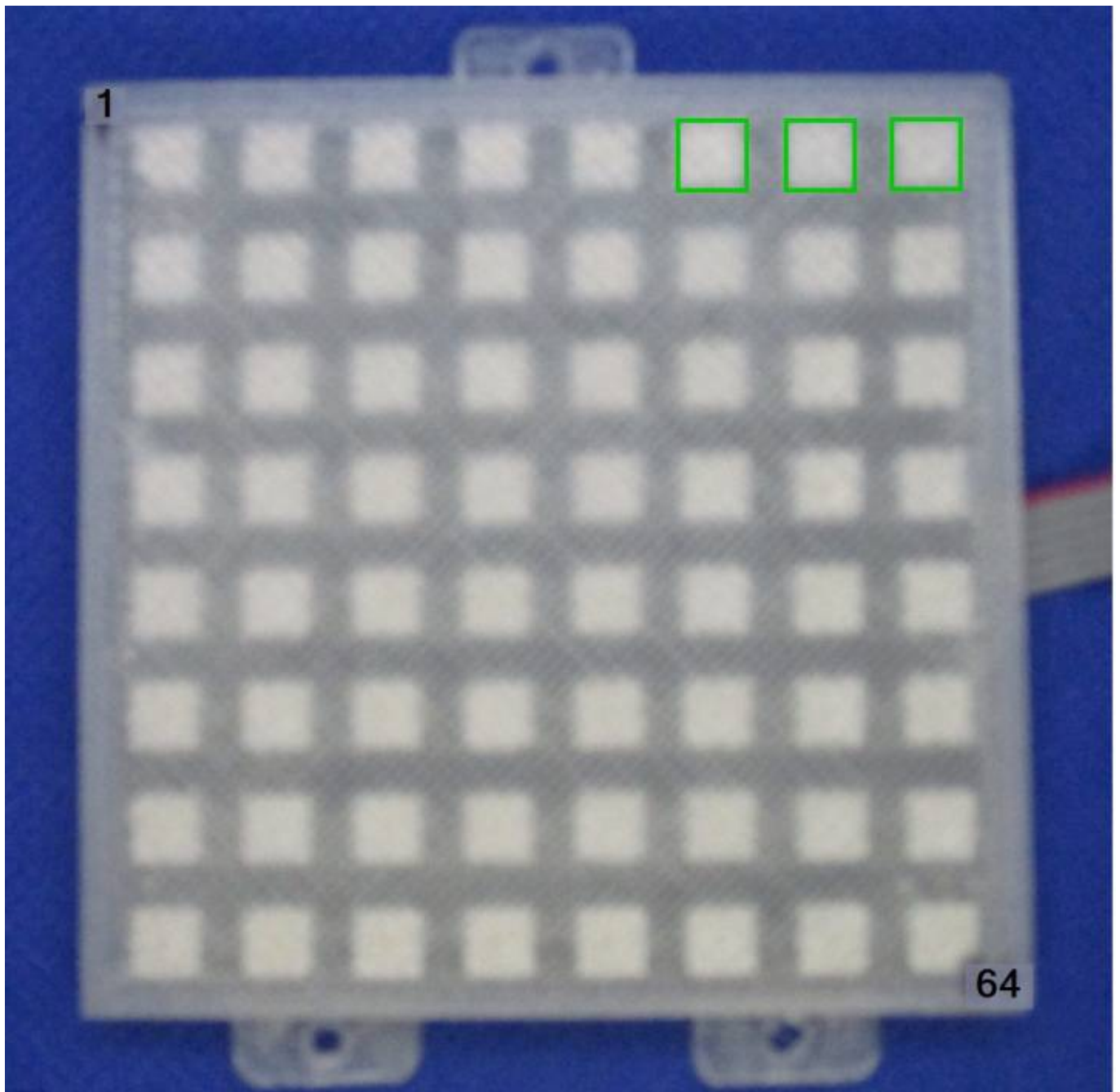


## Lichtsensor

Nach 1-2 Minuten leuchten LEDs 6 - 8 grün, wenn kein Helligkeitssensor angeschlossen ist. Wenn die LDR Pins mit einem Jumper überbrückt werden gehen die LEDs wieder aus (Tag). Wenn ein LDR angeschlossen ist kann man im seriellen Monitor der Arduino IDE (115200 Baud) die Messwerte überprüfen:

\ Inp: 27 damped: 25 LDR Der Stich am Anfang zeigt an ob es dunkler ,\' oder heller ,/' wird. Die Messwerte links geben den aktuellen Wert an. Große Werte bedeuten Dunkel, kleine Zahlen Hell. Der Messwert sollte sich zwischen 20 und 230 verändern, wenn der Sensor unterschiedlich beleuchtet wird. Für den Bereich ist die Kombination aus LDR und R13 entscheidend. Mit LDR5506 und R13=4.7K erreicht man etwa diesen Bereich. Wenn die Werte größer oder kleiner werden, dann interpretiert das Programm das Signal als Schalter und verändert die Dämpfung. Dadurch kann man mit einem Schalter mit drei Stellungen zwischen Automatik, Tag und Nacht umschalten.

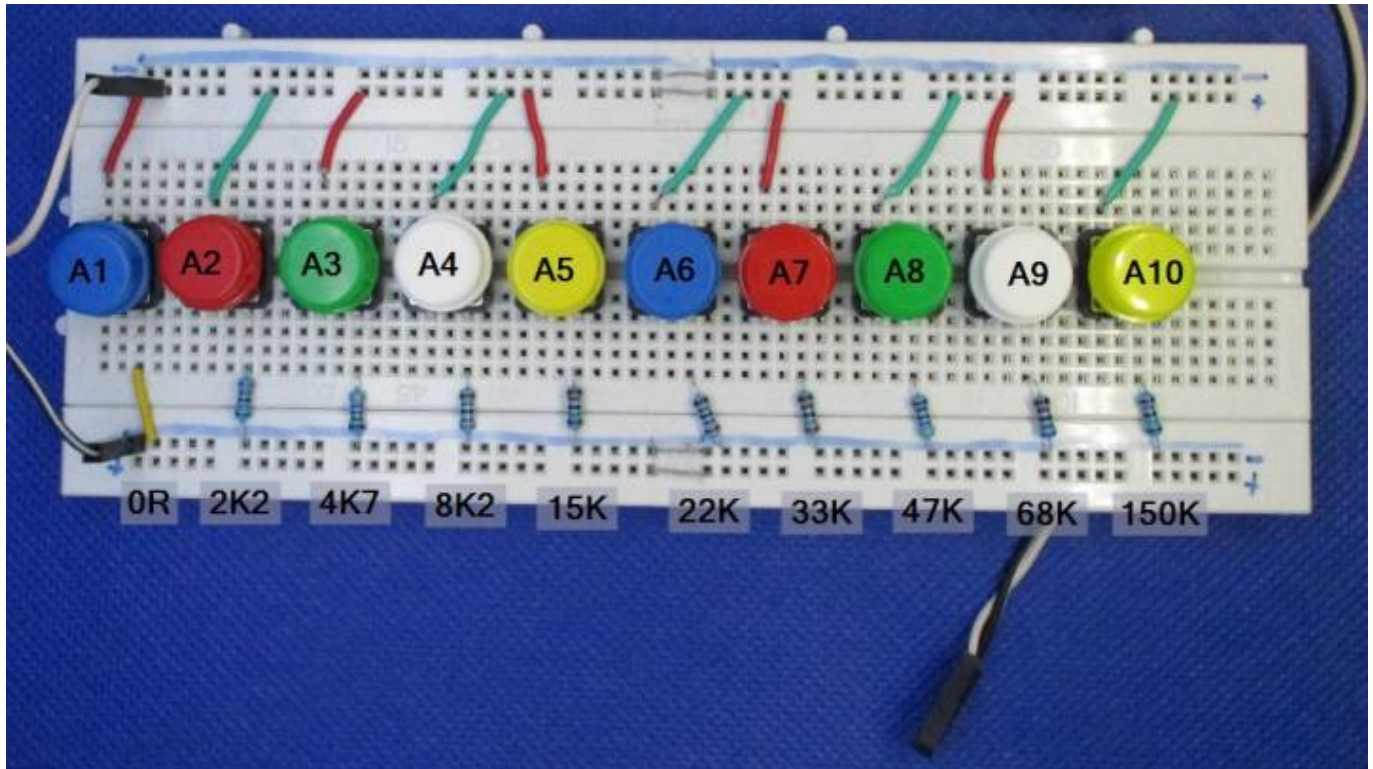


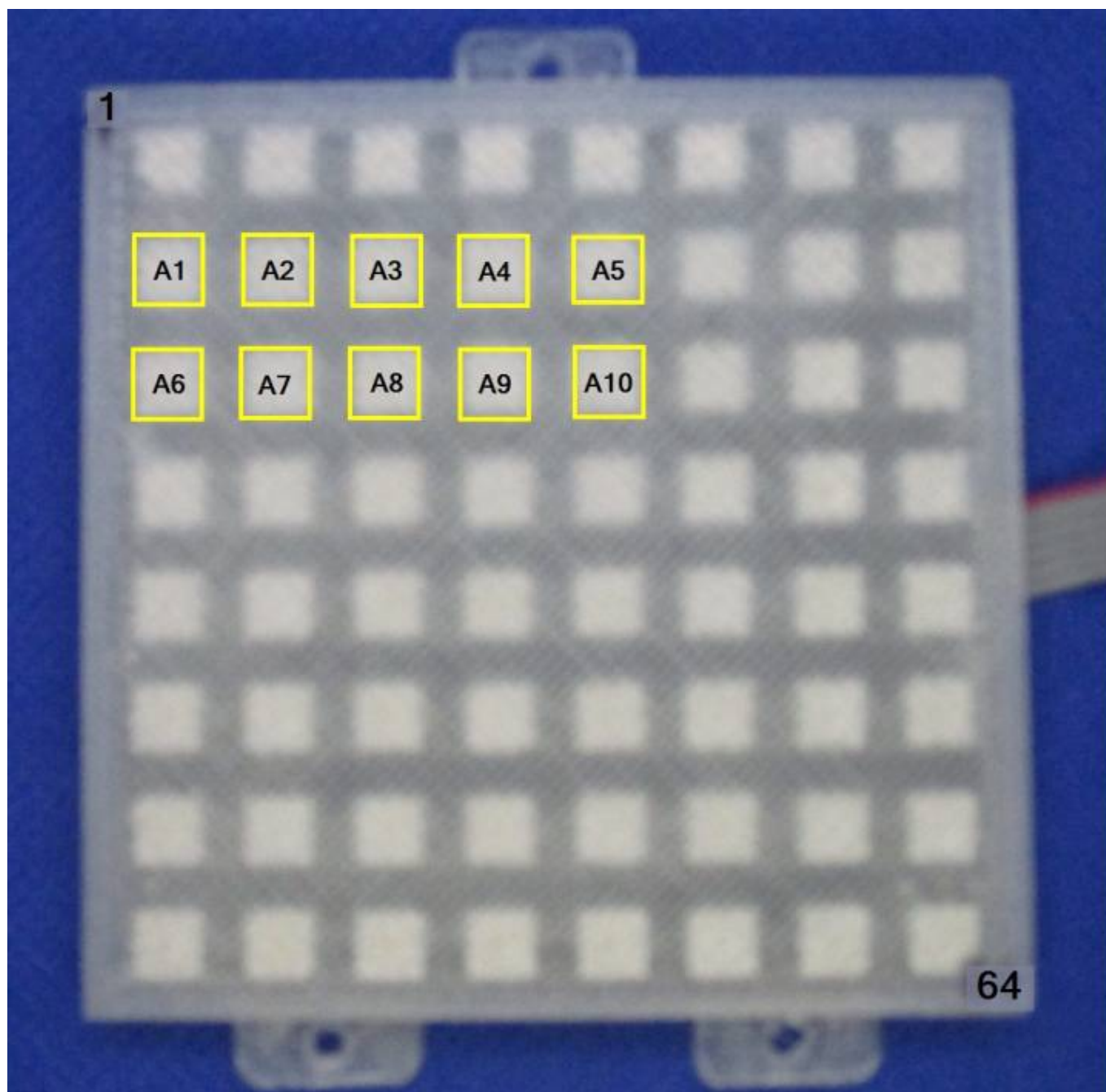




## Analoge Taster

Zehn widerstandskodierte Taster werden zwischen Pin 4 und 8 von Stecker KEYBOARD1 angeschlossen, die Polung ist dabei egal. Damit kann man 10 verschiedene gelbe LEDs steuern. Die Schalter werden über folgende Widerstände kodiert: 0 Ohm (Draht), 2K2, 4K7, 8K2, 15K, 22K, 33K, 47K, 68K, 150K



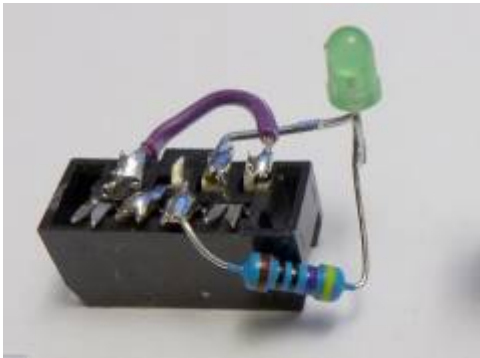


## MB Test 2 - KEYBRD



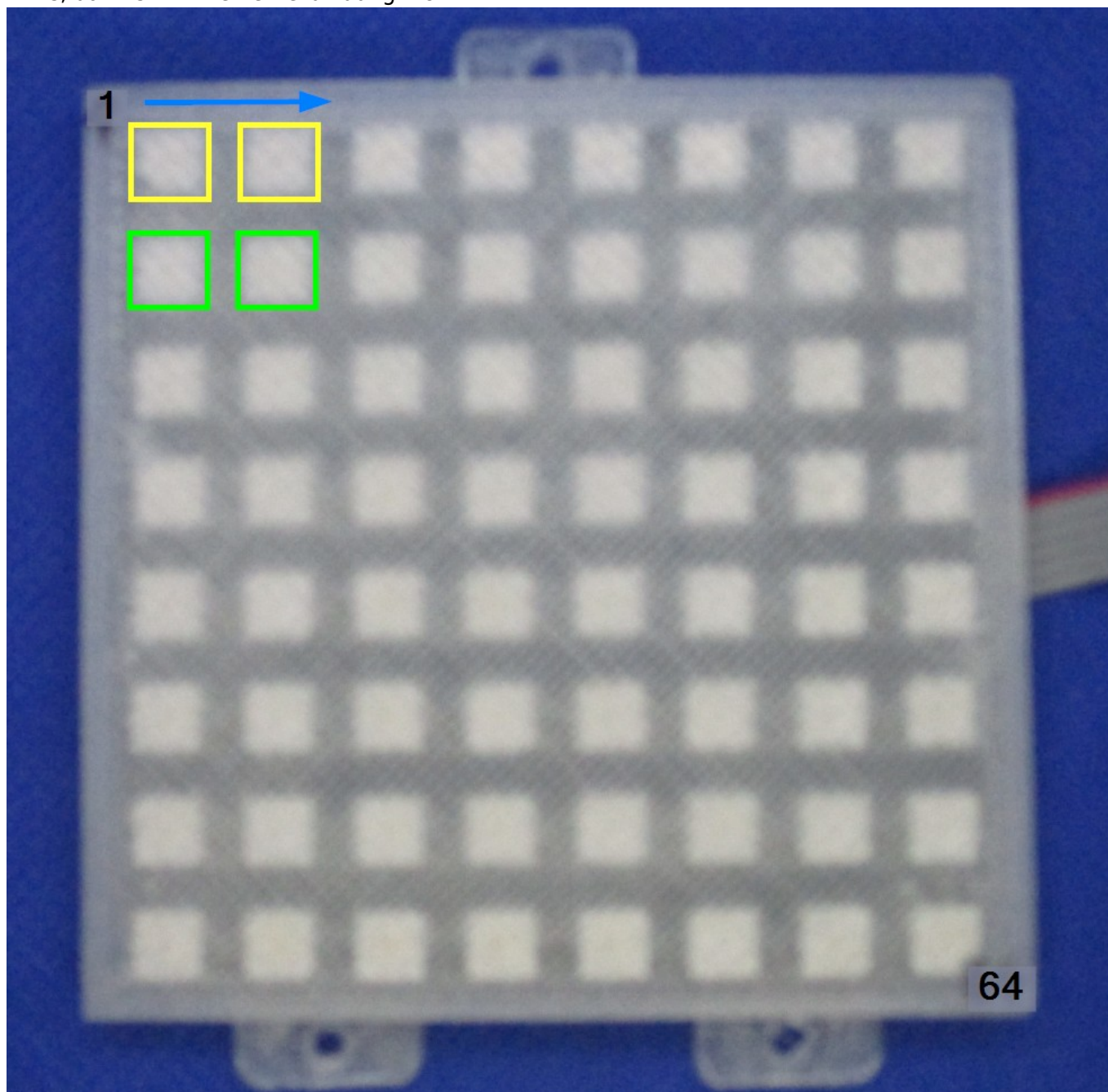
!! Achtung: Der Testadapter für den KEY\_80 - Anschluss darf nicht eingesteckt sein. !!

Für diesen Test benötigt man einen Testadapter, den man sich selber bauen kann oder verwendet zwei Jumperkabel. Mit dem Adapter/Jumperkabel müssen die Pins 1 & 7, sowie 5 & 6 verbunden werden. Zusätzlich kann man auch noch eine LED, zusammen mit einem Widerstand an die Pins 3 (VCC) & 4 (GND) anschließen. Diese LED prüft die Versorgungsspannung.





Auf der 64'er LED Matrix müssen zwei gelbe und zwei grüne LEDs durchlaufen. Fehlen die grünen LEDs, dann stimmt eine Verbindung nicht.



## MB Test 3 - KEY\_80

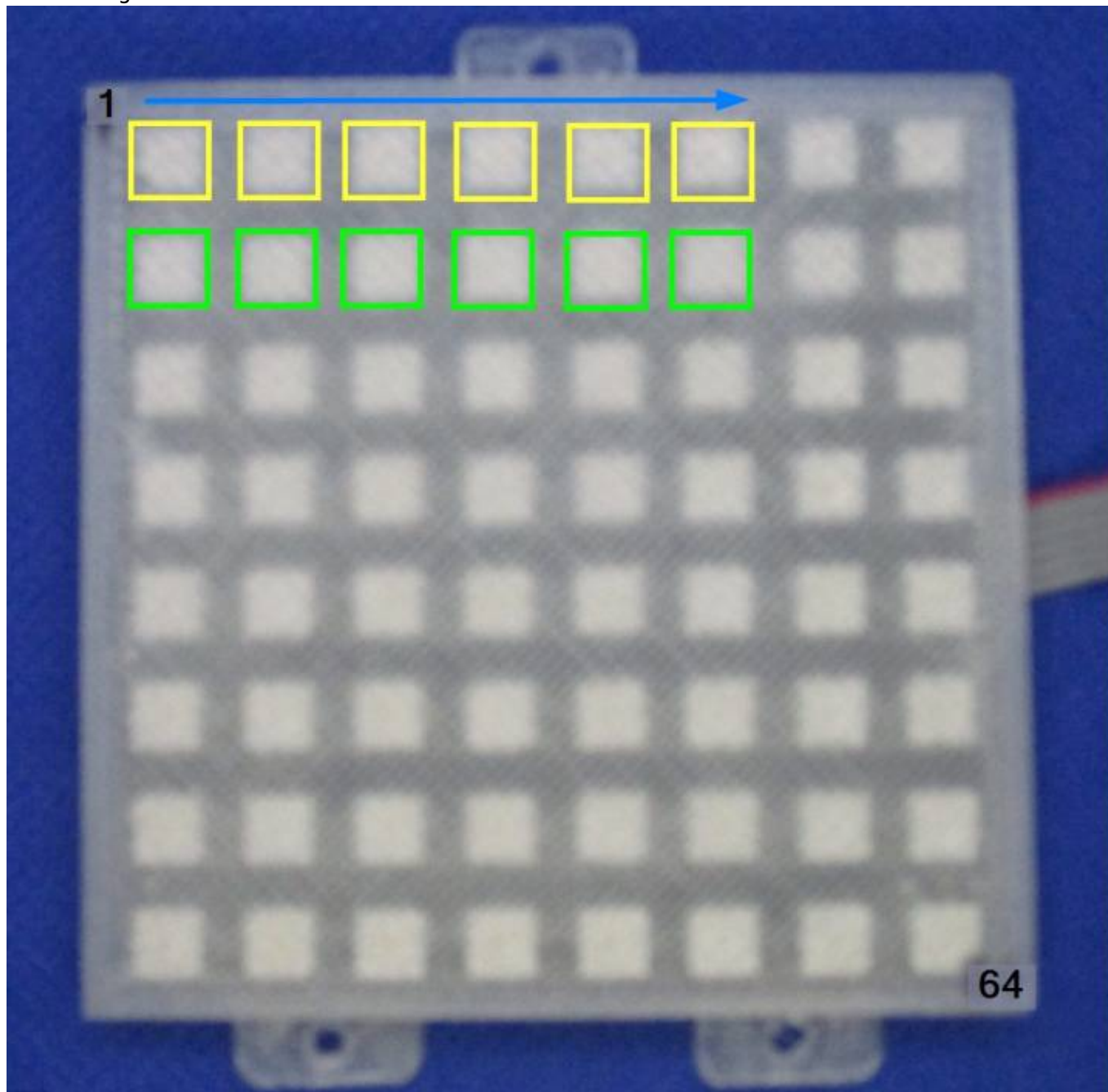


!! Achtung: Der Testadapter für den KEYBRD - Anschluss darf nicht eingesteckt sein. !!

In den KEY\_80 Stecker wird ein Testadapter gesteckt bei dem die folgenden Pins verbunden sind:  
1=2, 3=4, 5=6, 7=8, 9=10, 11=12. An Pin 13 und 14 ist eine grüne LED mit einem 470 Ohm Widerstand angeschlossen (+ = 13). Diese LED prüft die Versorgungsspannung.



Auf der 64'er LED Matrix müssen 6 gelbe und grüne LEDs durchlaufen. Fehlen eine oder mehrere der grünen LEDs, dann fehlt eine Verbindung. Laufen die Punkte nicht synchron, stimmen die Verbindungen nicht.

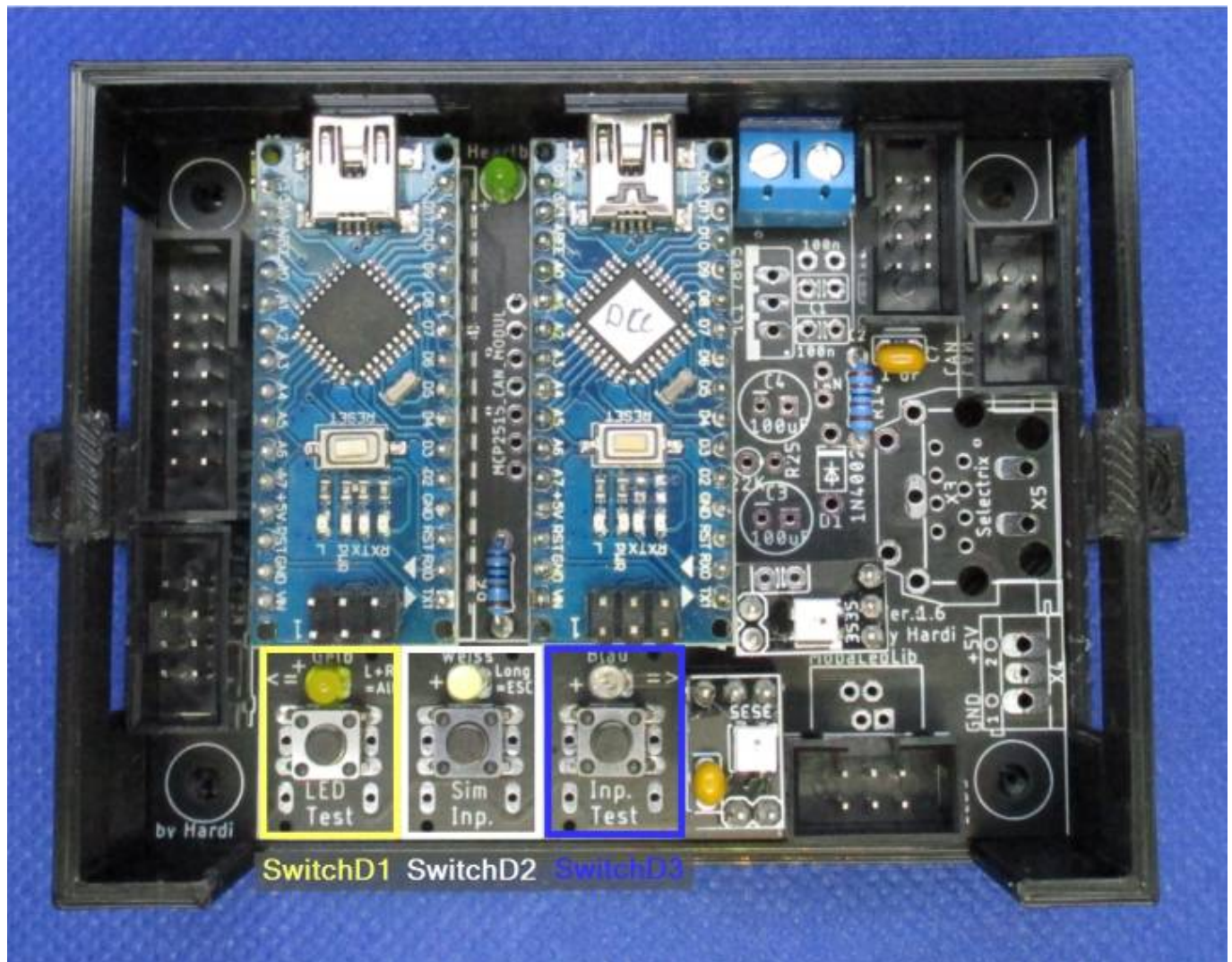


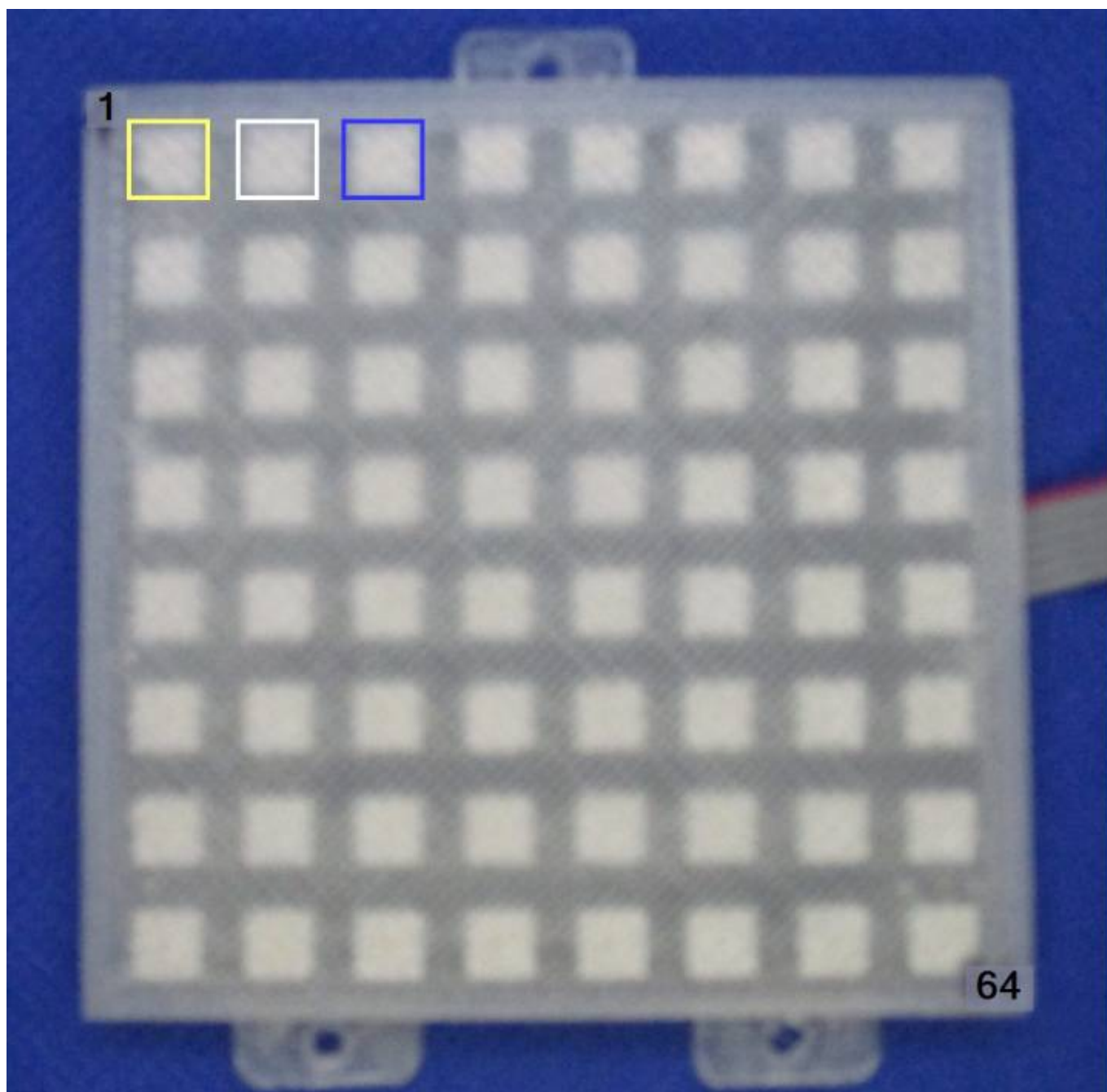


## MB Test 4 - Selectrix

### Tastererkennung onBoard

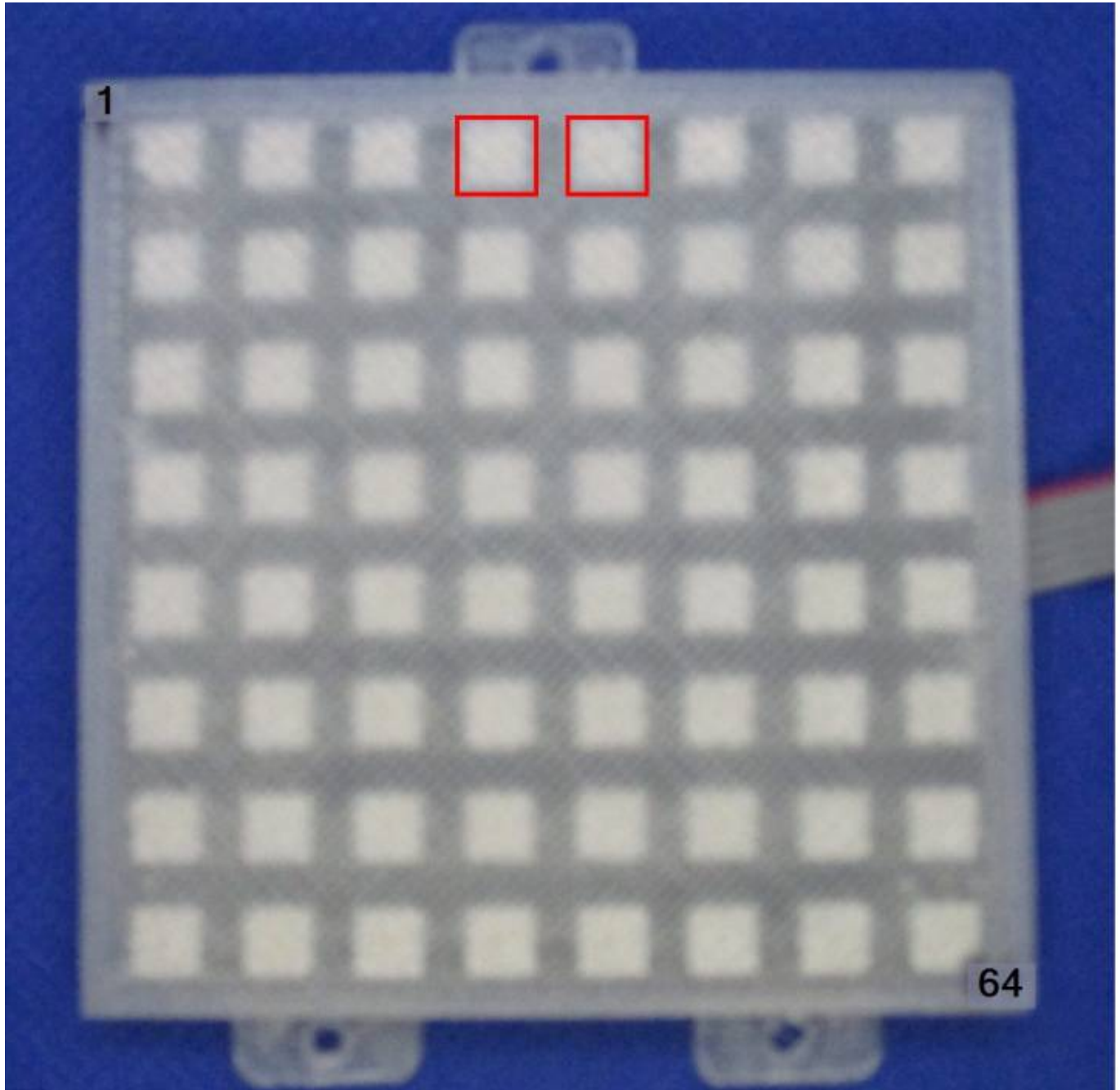
Mit den Tastern auf dem Mainboard können die LEDs über den Tastern geschaltet werden. Damit wird auch das Blinken der ersten drei LEDs des 64er Panels aktiviert.





## Verbindungstest Selectrix Zentrale

Über Channel 0, Bit 1 kann das Andreaskreuz geschaltet werden.  
Dieses aktiviert die LEDs 4&5.



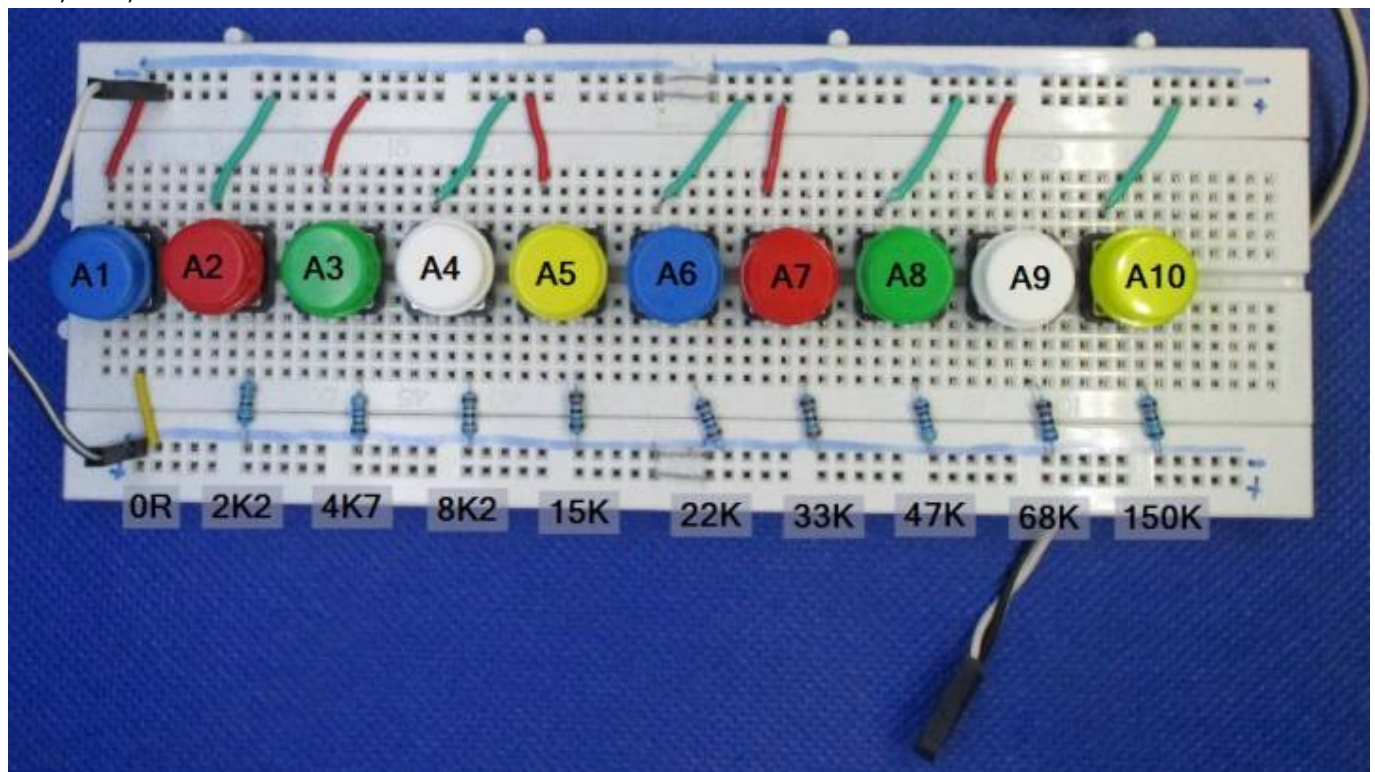


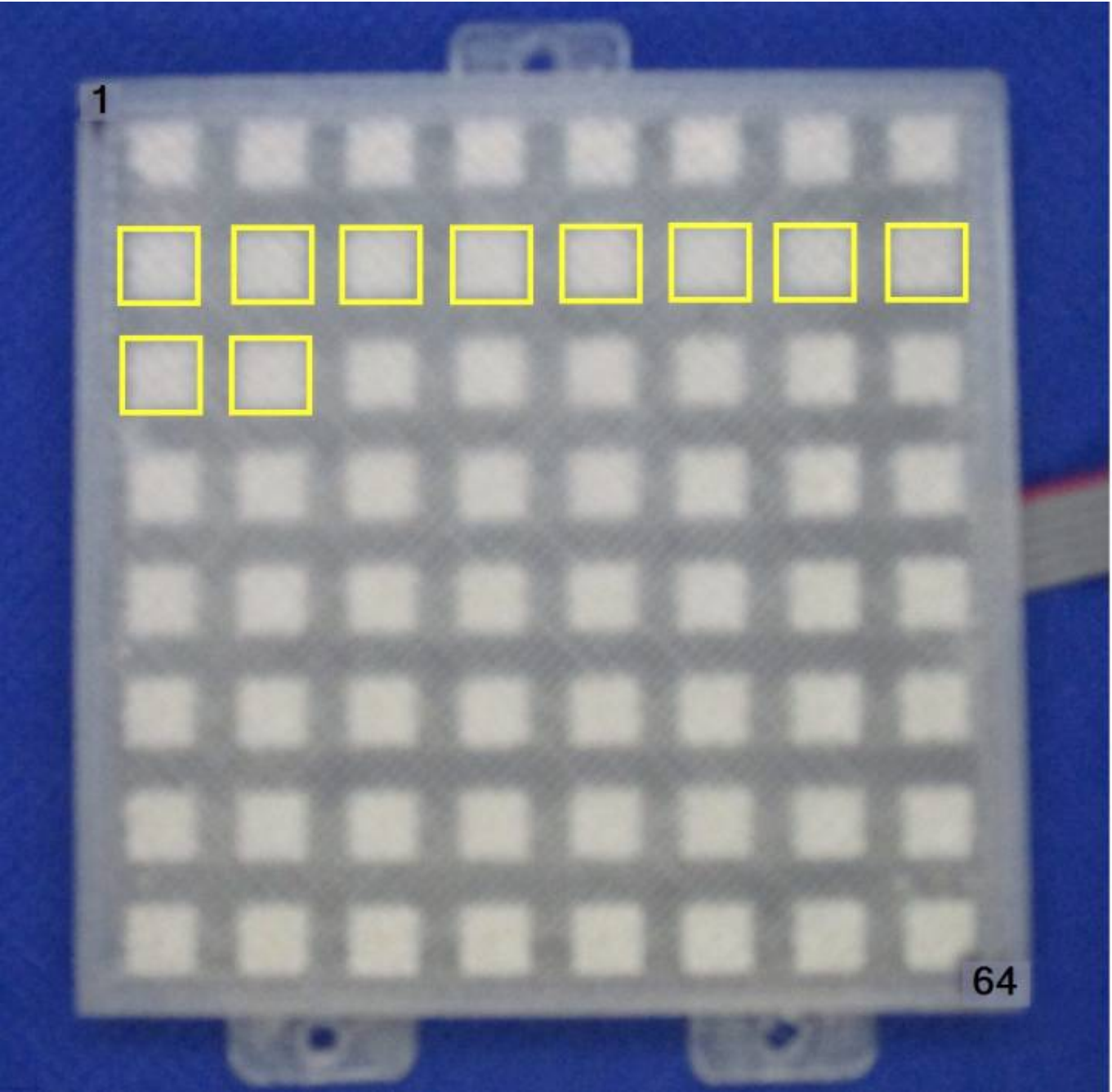
## Lichtsensor

Das selbe Testmuster wie bei dem Lichtsensor vom Blatt „MB Test 1 DCC“.

## Analoge Taster

Zehn widerstandskodierte Taster werden zwischen Pin 4 und 8 von Stecker KEYBOARD1 angeschlossen, die Polung ist dabei egal. Damit kann man 10 verschiedene gelbe LEDs steuern. Die Schalter werden über folgende Widerstände kodiert: 0 Ohm (Draht), 2K2, 4K7, 8K2, 15K, 22K, 33K, 47K, 68K, 150K



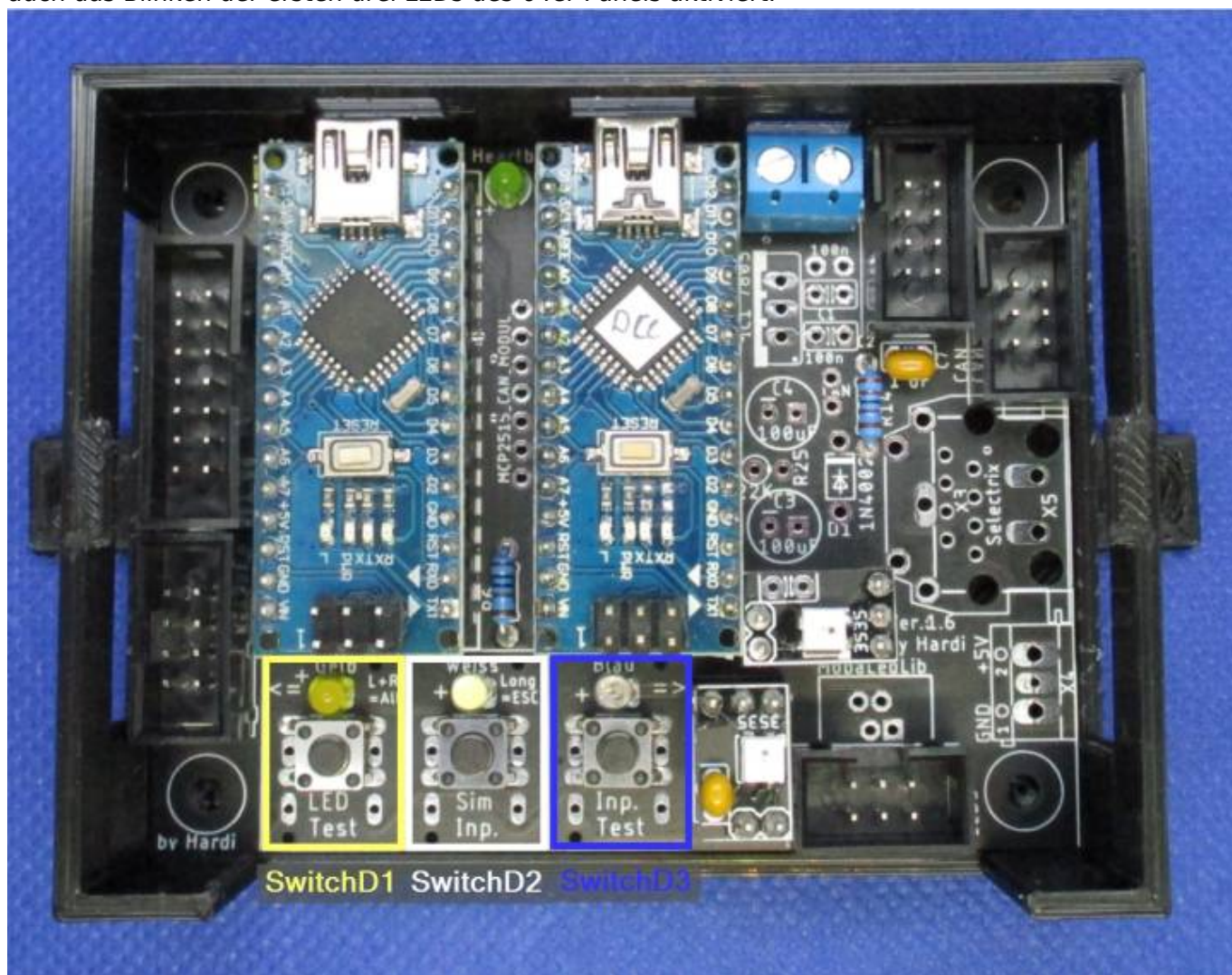


## MB Test 5 - CAN

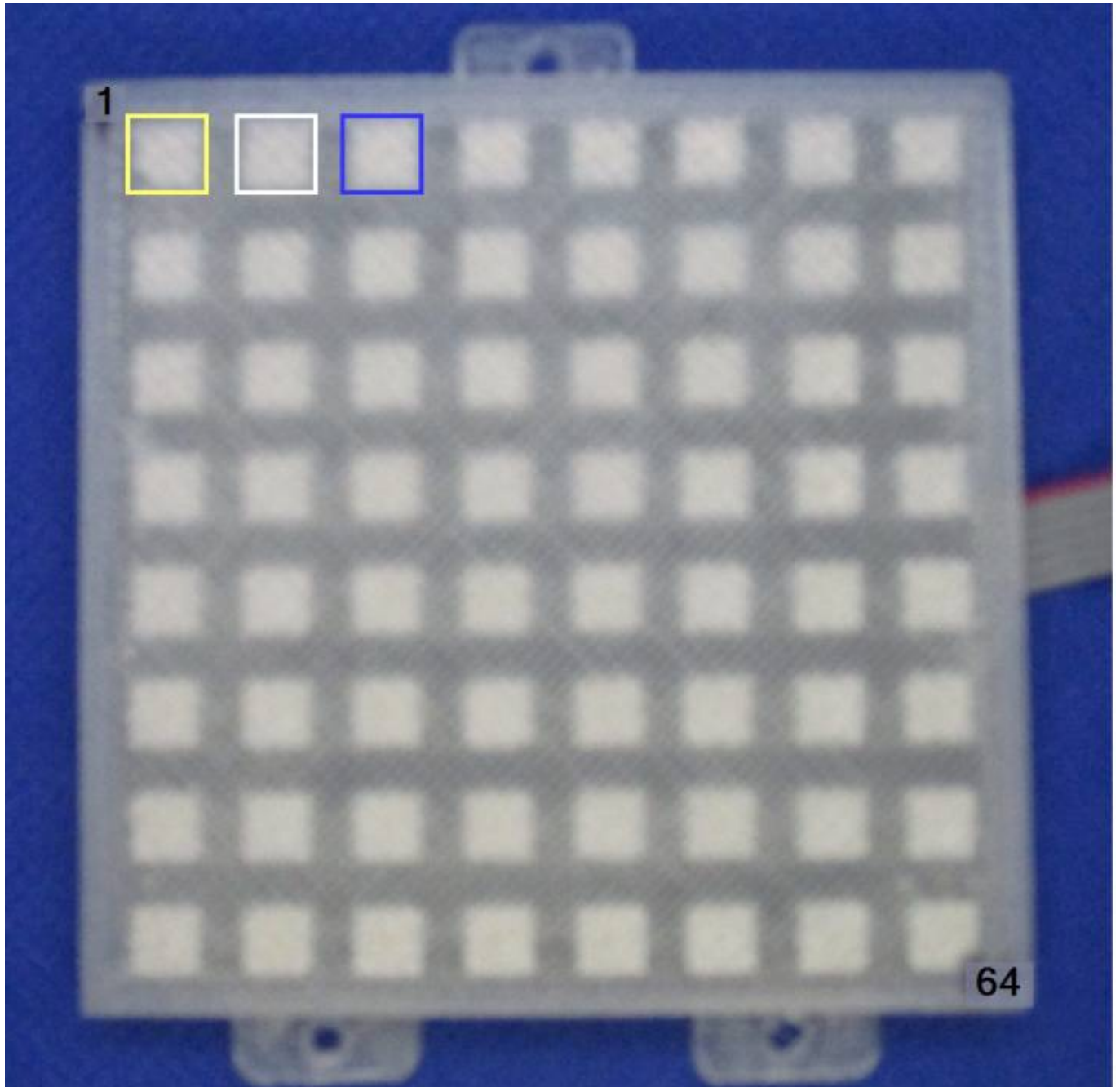
Mit der Konfigurationsseite „CAN Mainboard Test“ wird der CAN Bus und die restlichen Taster Pins geprüft. Dazu muss anstelle des DCC Arduinos ein CAN Modul eingesteckt werden und die Konfiguration hochgeladen werden.

### Tastererkennung onBoard

Mit den Tastern auf dem Mainboard können die LEDs über den Tastern geschaltet werden. Damit wird auch das Blinken der ersten drei LEDs des 64er Panels aktiviert.

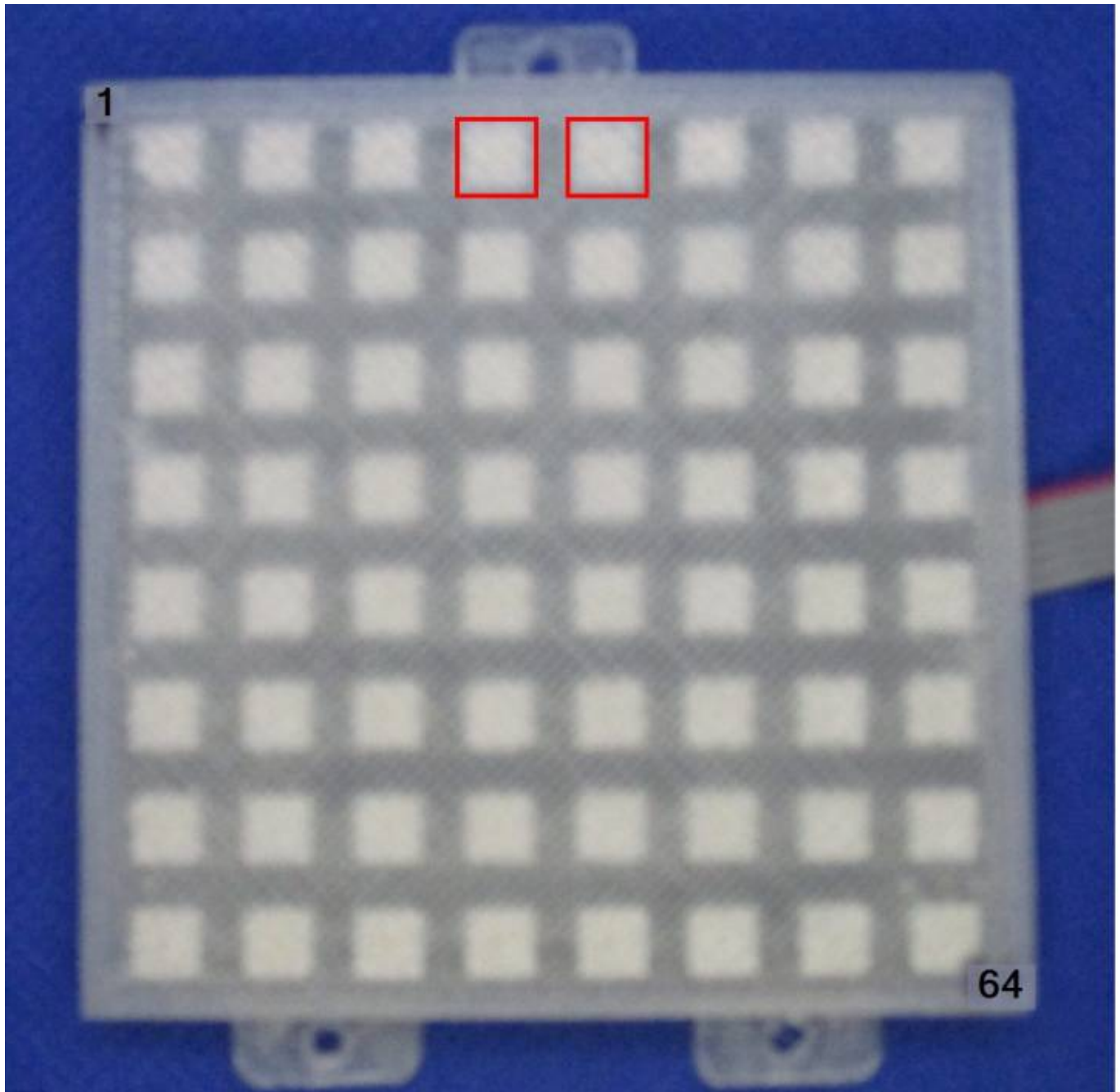






## Test CAN-BUS Verbindung

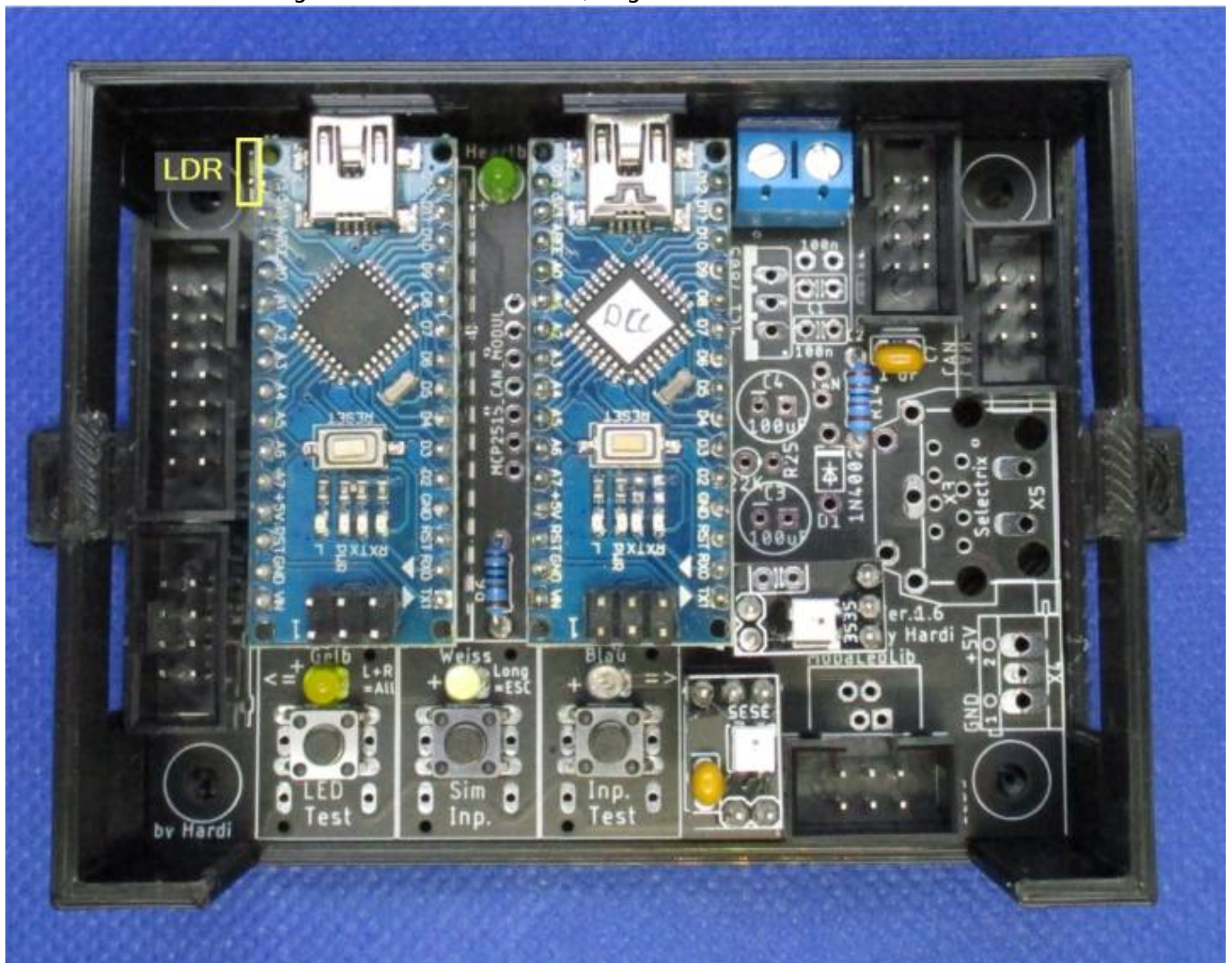
Mit der CAN Adresse 1 (Märklin MS2) wird auch hier ein Andreaskreuz aktiviert.



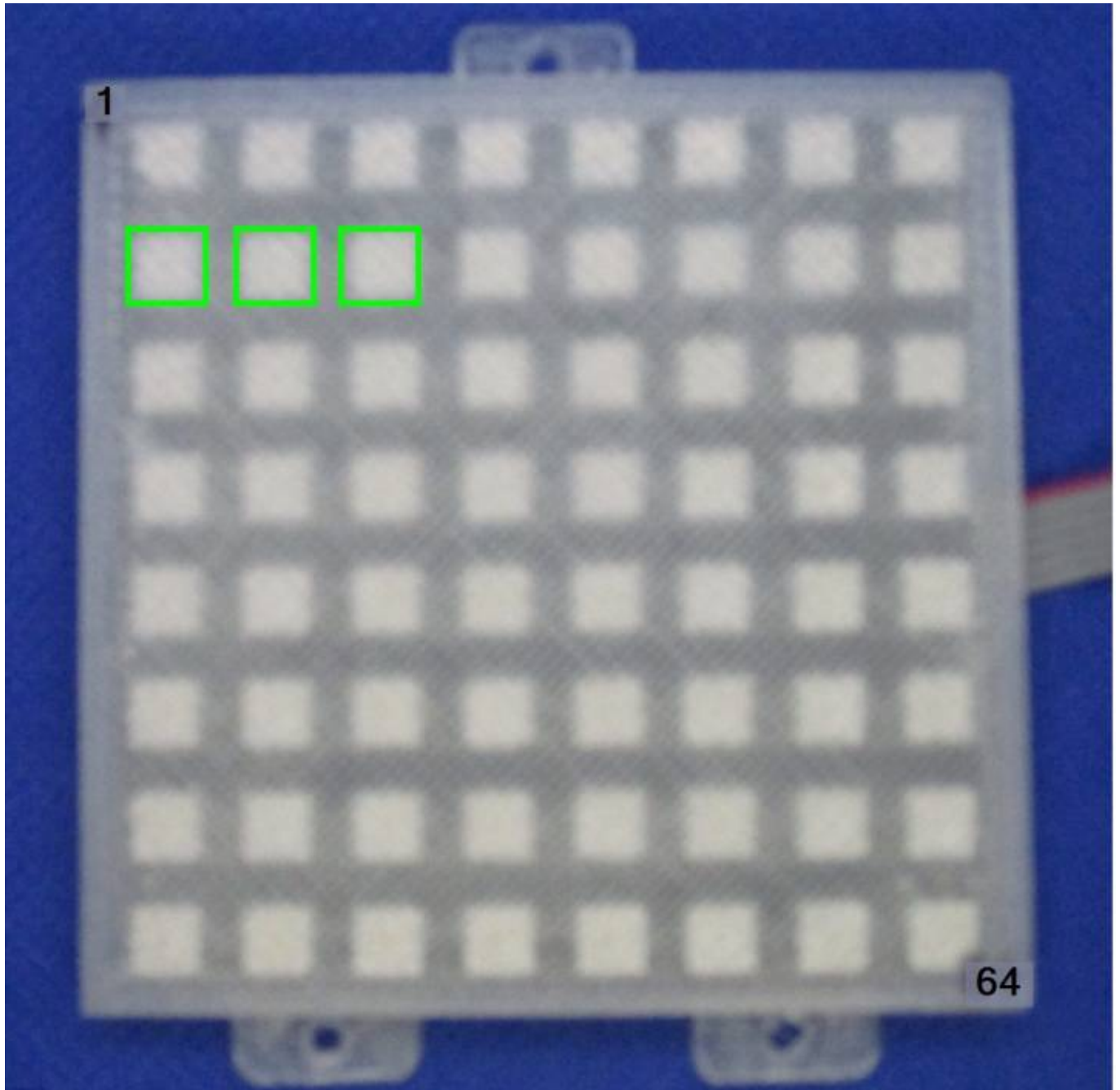
## Lichtsensord

Nach 1-2 Minuten leuchten LEDs 6 - 8 grün, wenn kein Helligkeitssensor angeschlossen ist. Wenn die LDR Pins mit einem Jumper überbrückt werden gehen die LEDs wieder aus (Tag). Wenn ein LDR angeschlossen ist kann man im seriellen Monitor der Arduino IDE (115200 Baud) die Messwerte überprüfen:

\ Inp: 27 damped: 25 LDR Der Stich am Anfang zeigt an ob es dunkler ,\' oder heller ,/' wird. Die Messwerte links geben den aktuellen Wert an. Große Werte bedeuten Dunkel, kleine Zahlen Hell. Der Messwert sollte sich zwischen 20 und 230 verändern, wenn der Sensor unterschiedlich beleuchtet wird. Für den Bereich ist die Kombination aus LDR und R13 entscheidend. Mit LDR5506 und R13=4.7K erreicht man etwa diesen Bereich. Wenn die Werte größer oder kleiner werden, dann interpretiert das Programm das Signal als Schalter und verändert die Dämpfung. Dadurch kann man mit einem Schalter mit drei Stellungen zwischen Automatik, Tag und Nacht umschalten.

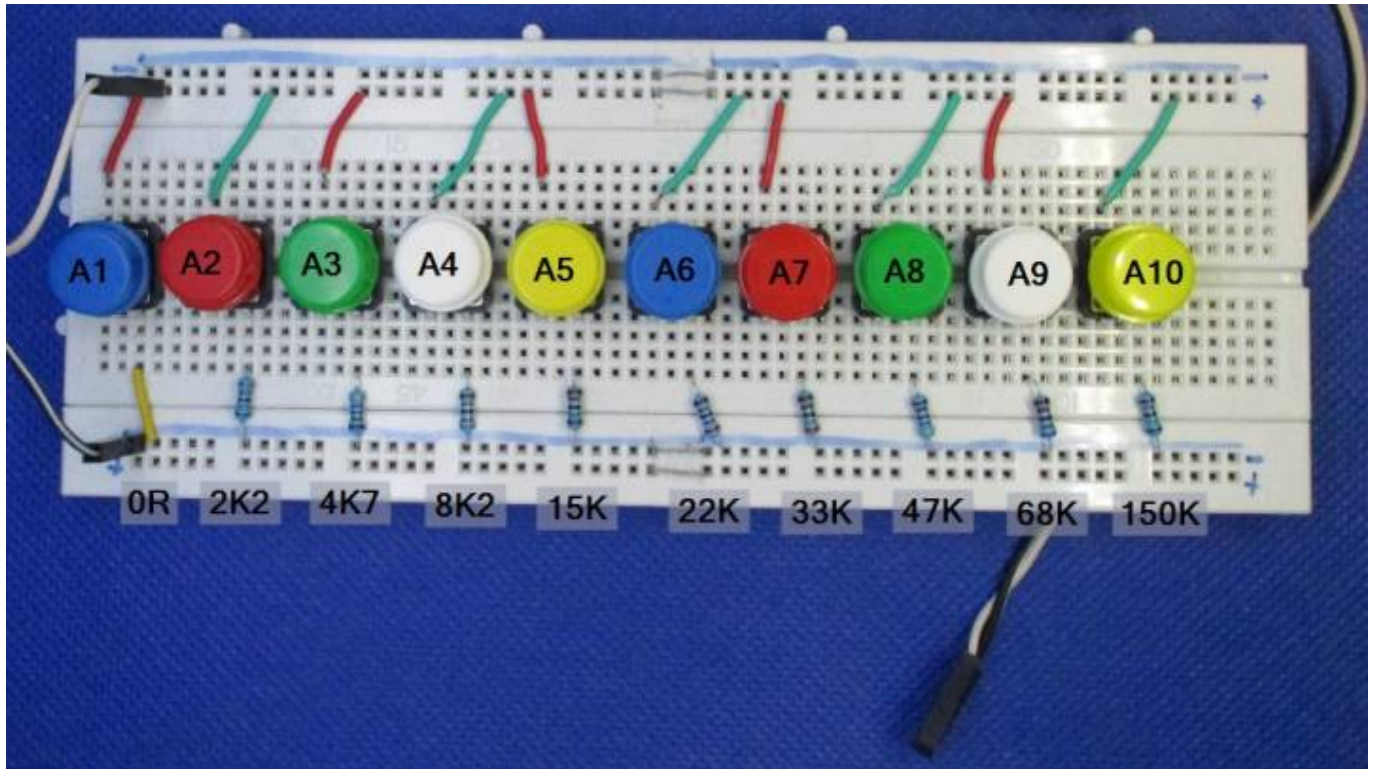


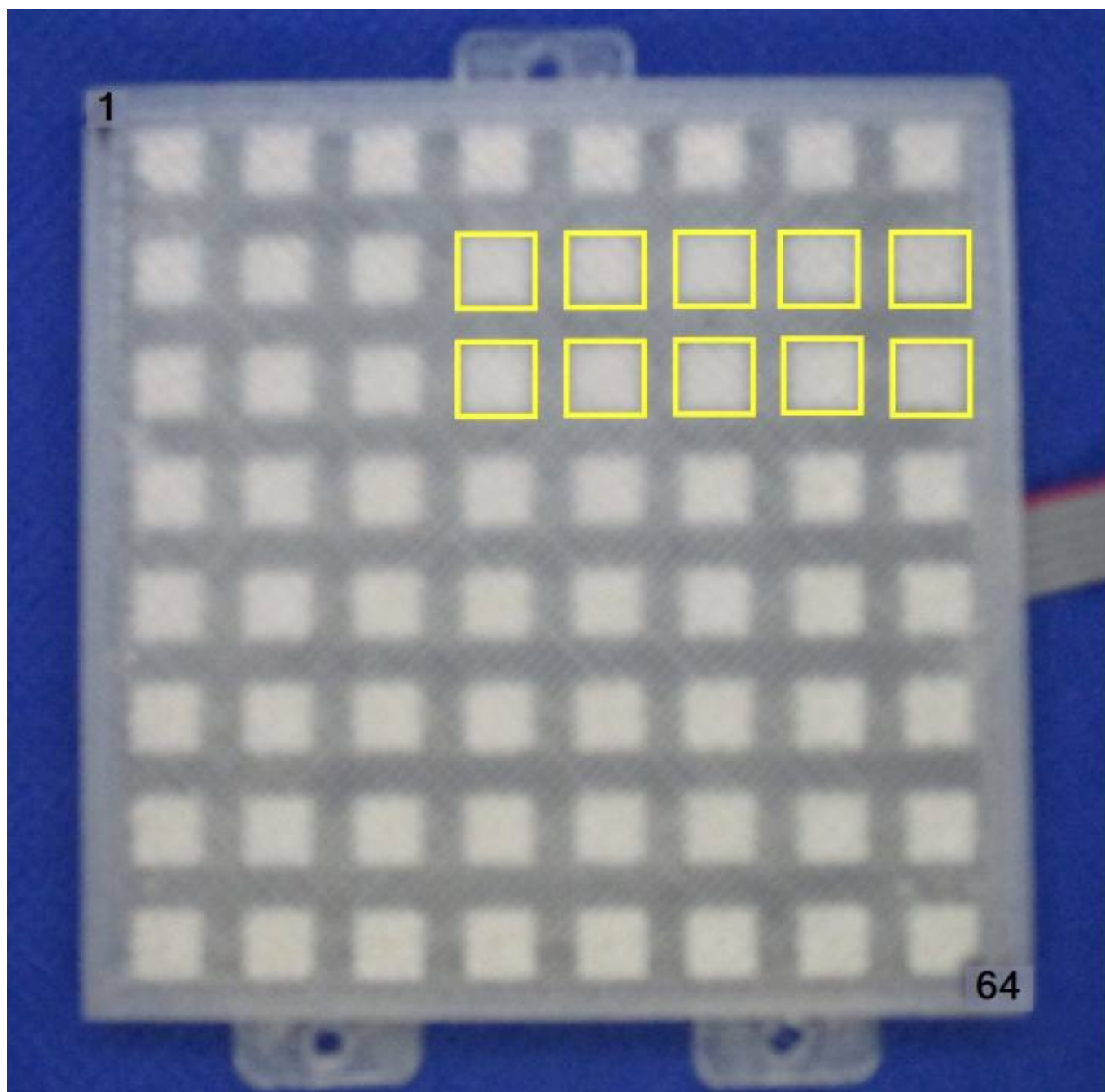




## Analoge Taster

Zehn widerstandskodierte Taster werden zwischen Pin 4 und 8 von Stecker KEYBOARD1 angeschlossen, die Polung ist dabei egal. Damit kann man 10 verschiedene gelbe LEDs steuern. Die Schalter werden über folgende Widerstände kodiert: 0 Ohm (Draht), 2K2, 4K7, 8K2, 15K, 22K, 33K, 47K, 68K, 150K





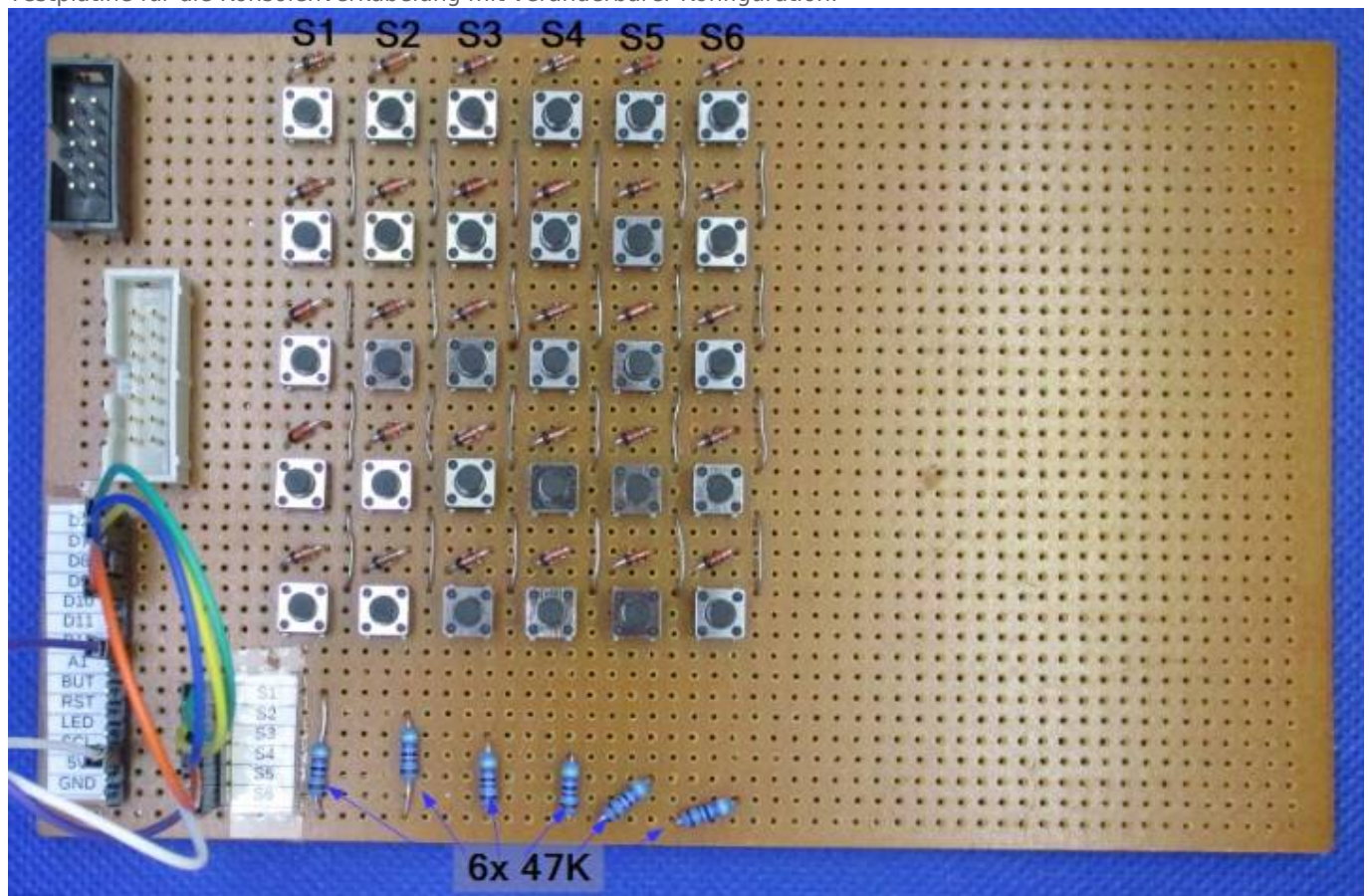


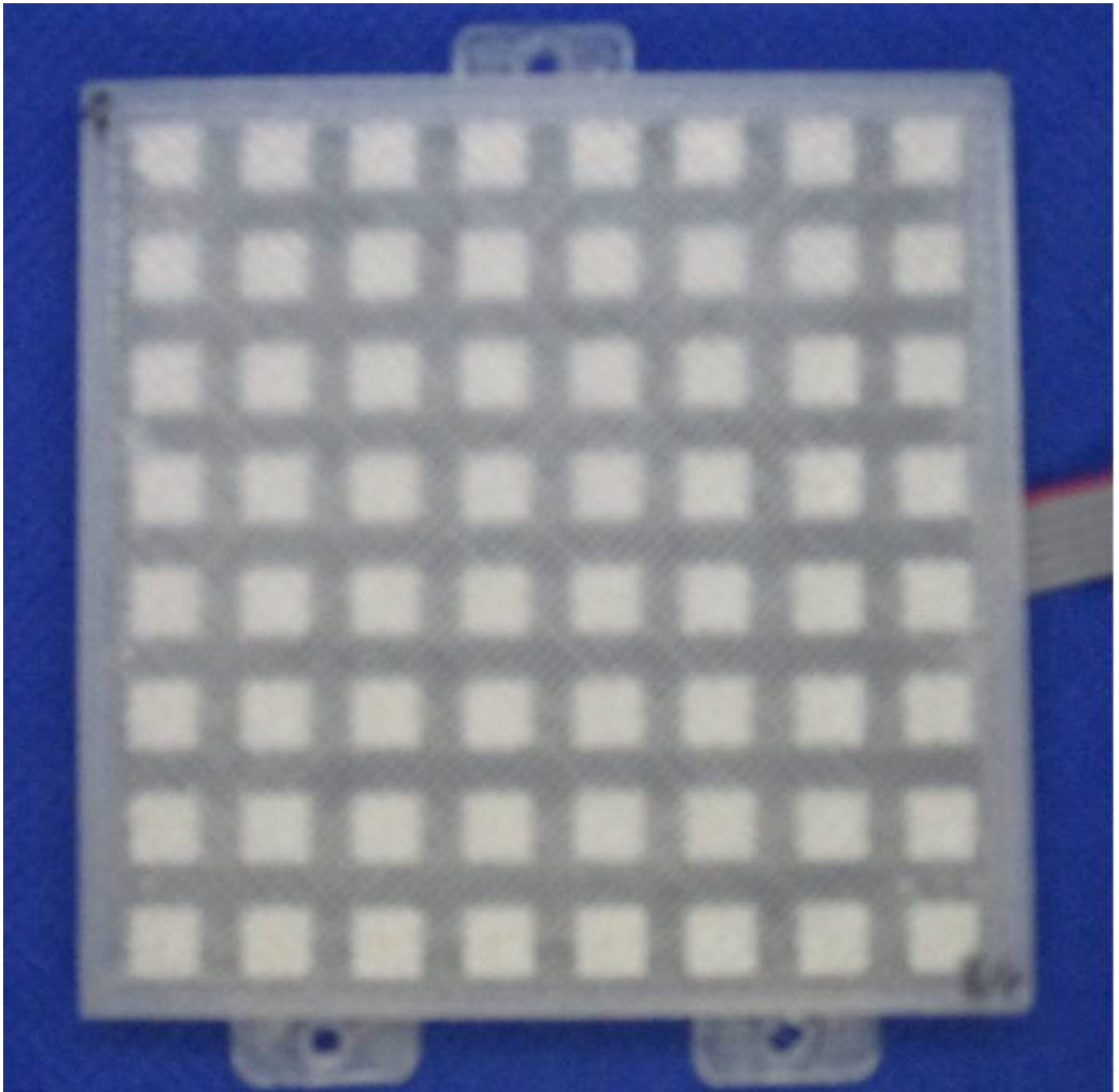
# PushButton Test 1 - 64 Tast

## Taster im Weichenstellpult

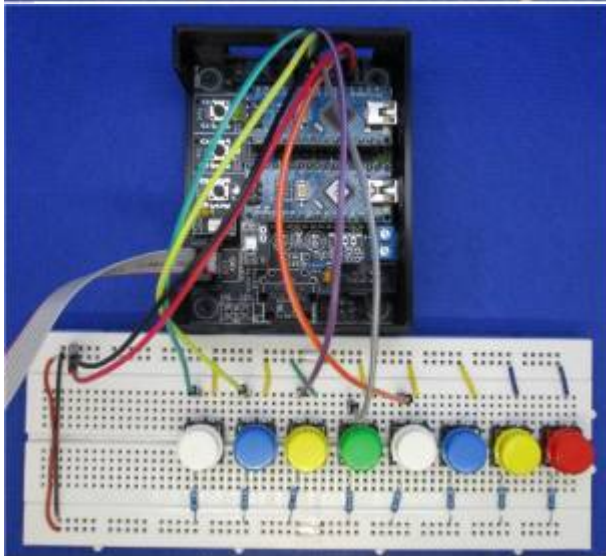
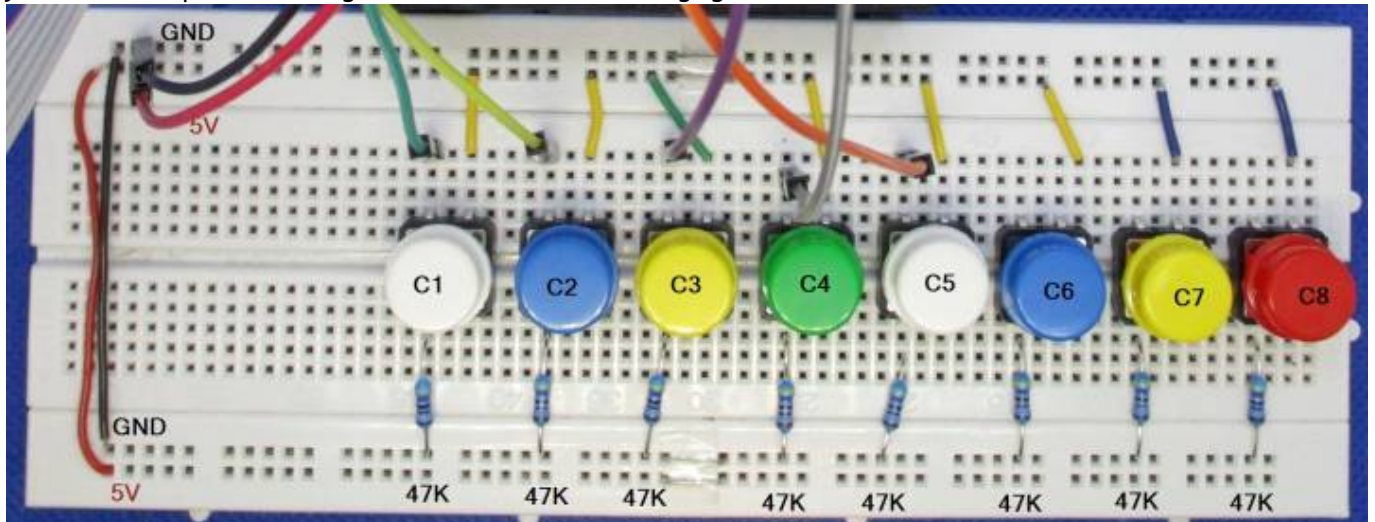
Mit diesem Test wird die PushButton4017 Platine und eine angeschlossene Taster Matrix getestet. Die Taster Matrix besteht aus 64 Tastern welche in 8 Zeilen und 8 Spalten angeordnet sind. Die Zeilen werden an die Pins 1-7 und 12 des Key\_80 Steckers angeschlossen. **Achtung C9 und R27 der Hauptplatine dürfen nicht bestückt sein.** Jeder Taster aktiviert eine LED auf der LED Matrix. Mit einer PushButton Platine können bis zu 80 Taster abgefragt werden. Für diesen kompletten Test müssten dann allerdings zusätzliche LEDs angeschlossen und die Zeilen erweitert werden.

Testplatine für die Konsolenverkabelung mit veränderbarer Konfiguration.

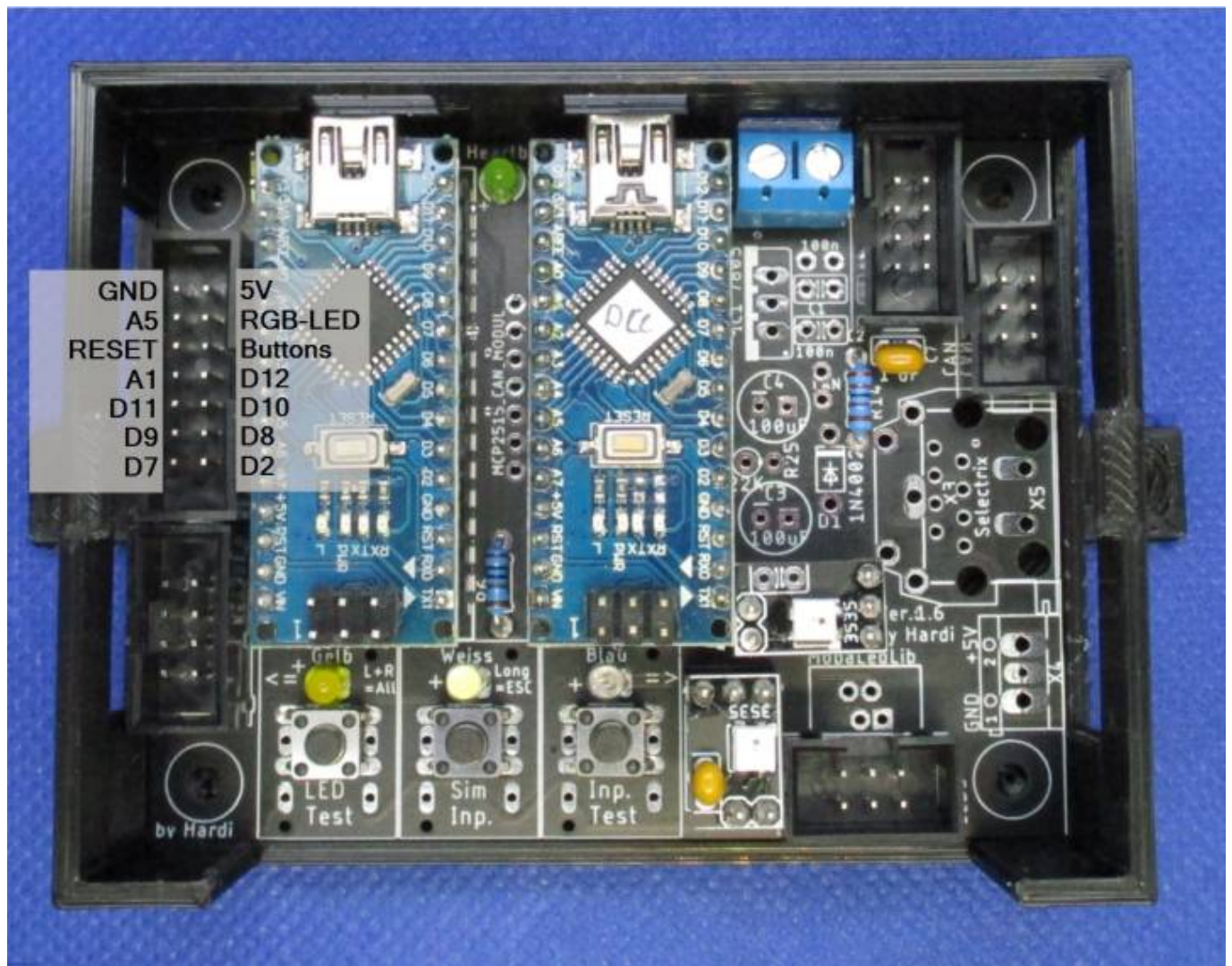




Anstelle der Pushbutton Platine können auch 8 Taster verwendet werden welche gegen +5V schalten. Jede Taster-Spalte benötigt einen 47K Widerstand gegen Masse.





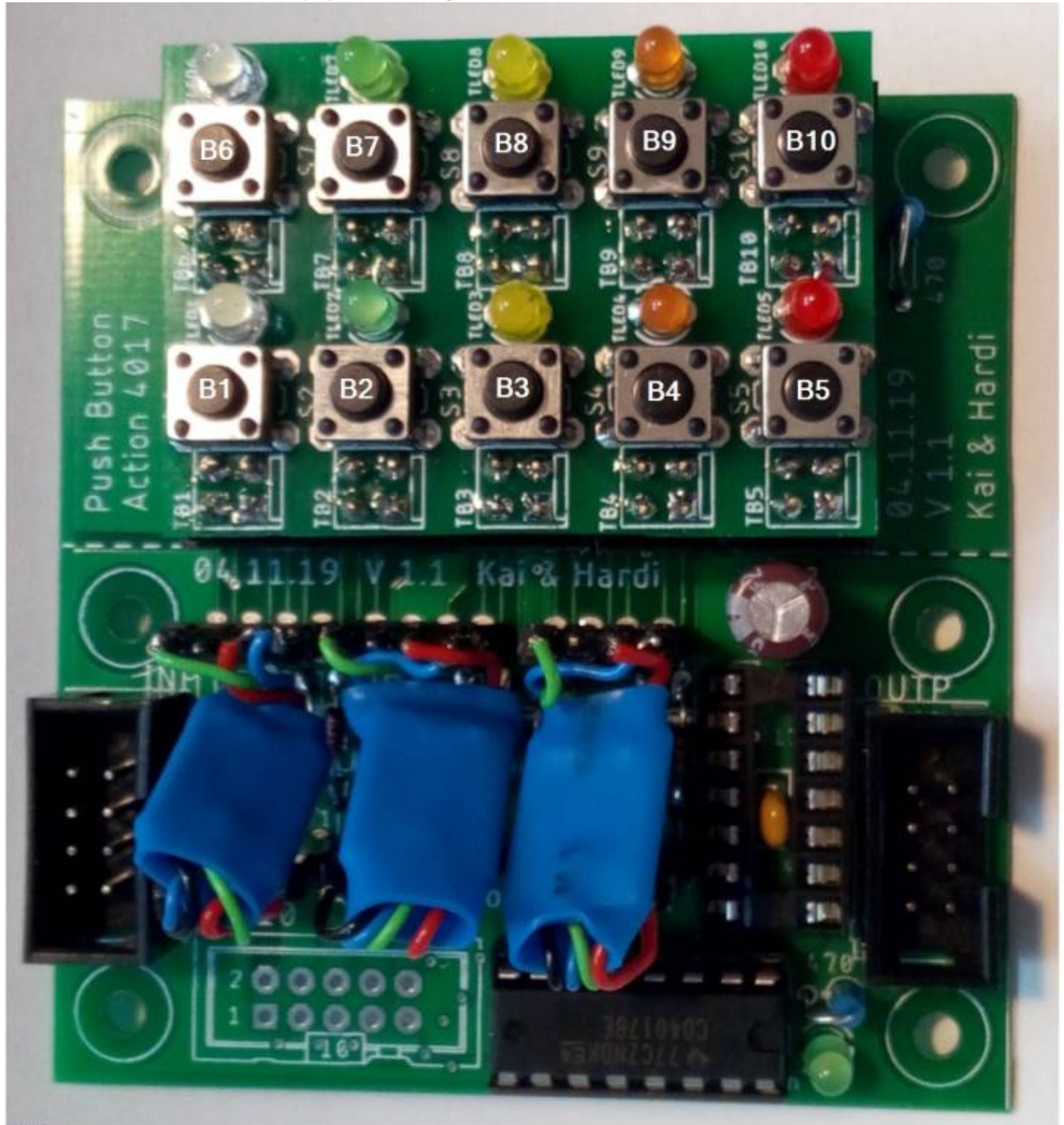


## PushButton Test 2 - 10 Tast+LED

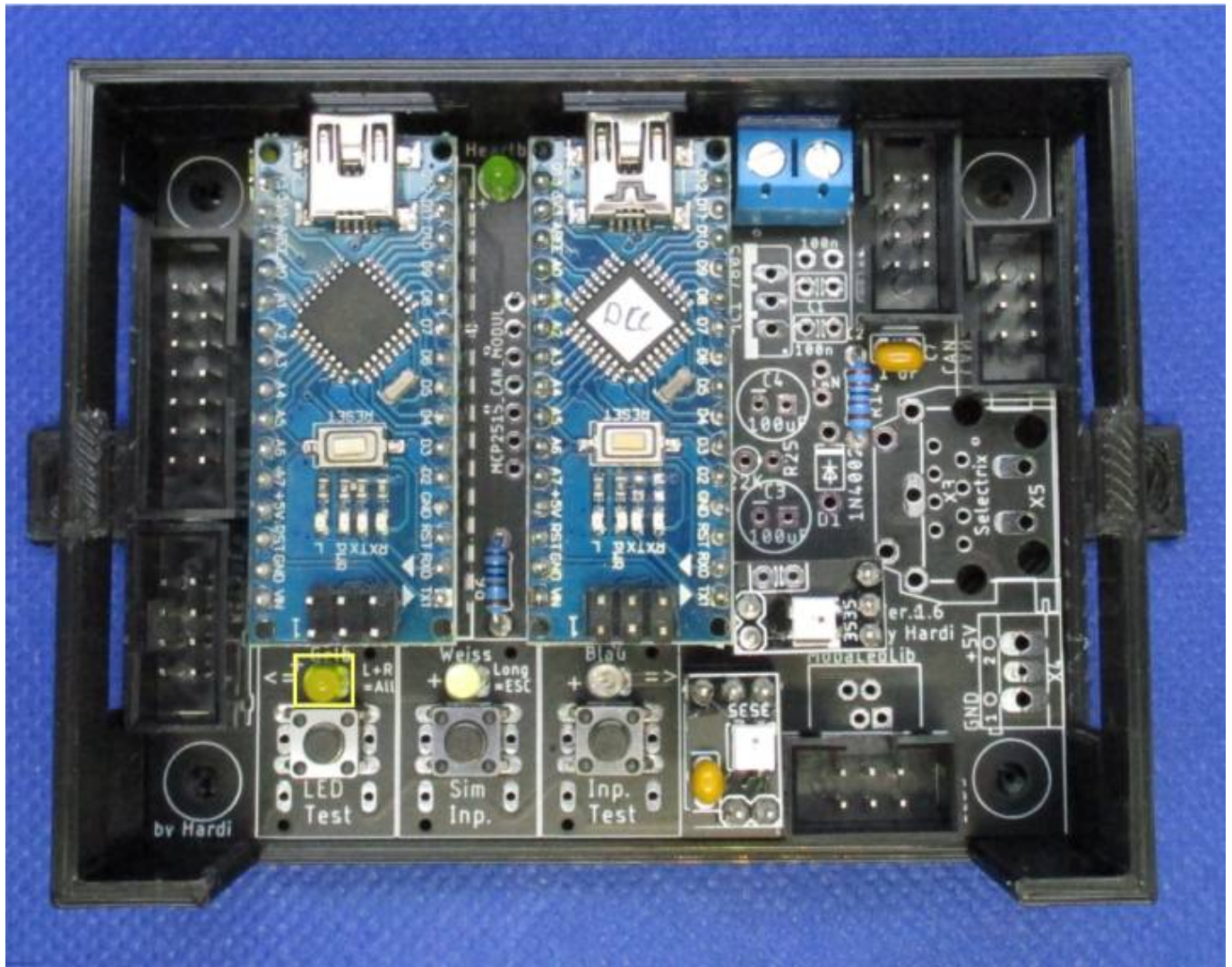
### Push-Button-Platine

Prüft die PushButton Platine in der Konfiguration für den Anlagenrand mit 10 Tastern und integrierten LEDs. Jede Taste steuert die dazu gehörige LED an.

**LED #10 leuchtet dauerhaft und kann nicht angesteuert werden. Daher wird bei Taster 10 die erste LED auf der Hauptplatine angesteuert.**







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

Permanent link:  
[https://wiki.mobaledlib.de/anleitungen/platinen/testanleitung\\_hauptplatine\\_100\\_16de\\_sw200?rev=1603211444](https://wiki.mobaledlib.de/anleitungen/platinen/testanleitung_hauptplatine_100_16de_sw200?rev=1603211444)

Last update: 2020/10/20 17:30

