

PlayGround

Nicht alles was in der MobaLedLib möglich ist, lässt sich auch im Dialogfeld unterbringen. Auch sind die Wünsche der einzelnen Anwender doch sehr unterschiedlich. In diesem Kapitel sollen Abläufe/Sketches vorgestellt werden, die funktionieren und selbst in den Prog_Generator zu erfassen sind. Positiver Nebeneffekt, durch die Erfassung wird eine gewisse Lernkurve erzielt, die im Idealfall dazu führt, dass man durch Änderung oder Verwendung einzelner Befehle eigene Abläufe generiert, die dann gerne wieder hier veröffentlicht werden können.

- Beispiel einer Discobeleuchtung:

<https://www.youtube.com/embed/UjEwNN2QuH8>

Mit einer oder mehreren RGB-LED kann der Discobetrieb sowie das „Arbeitslicht“ bei Partyende dargestellt werden. Für die Discobeleuchtung wird das Flashlight benutzt und bei jeder LED die 3 Kanäle mit verschiedenen Zeiten einzeln angesteuert. Die LED ist doppelt zugewiesen, es ist entweder nur Disco oder Tagesmodus aktiviert. Die Umschaltung wurde über einen einfachen Schalter gelöst, ist aber auch über DCC-Ansteuerung möglich. Damit Ihr das gleiche Ergebnis wie in dem Video bekommt, sind im Prog_Generator folgende Zeilen zu erfassen:

Aktiv	Filter	DCC Adresse	Typ	Start-wert	Beschreibung	Verteiler-Nummer	Stecker-Nummer	Beleuchtung, Sound, oder andere Effekte	Start LedNr	LEDs	InCnt	Loc InCh
✓		1	AnAus					#define TEST_TOGGLE_BUTTONS			0	0
✓		1	AnAus					Flash(#LED, C_ALL, #InCh, #LocInCh, 5 Sek, 120 Sek)	0	1	1	1
✓		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	1	C1-1	1	1
✓		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	1	C2-2	1	1
✓		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	1	C3-3	1	1
✓		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	2	C1-1	1	1
✓		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	2	C2-2	1	1
✓		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	2	C3-3	1	1
✓		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	3	C1-1	1	1
✓		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	3	C2-2	1	1
✓		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	3	C3-3	1	1
✓		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	4	C1-1	1	1
✓		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	4	C2-2	1	1
✓		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	4	C3-3	1	1
✓		1	AnAus					HouseT_Inv(#LED, #InCh, 4, 4, 0, 0, NEON_LIGHTL, NEON	5	4	1	0

Das Umschalten funktioniert auch ohne DCC. Der Trick ist die erste Zeile. Der Befehl „#define TEST_TOGGLE_BUTTONS“ aktiviert die drei Taster der Hauptplatine zum Simulieren der ersten drei DCC Befehlen. (funktioniert nicht, bei Minimalbelegung der Hauptplatine!) Das geht auch wenn kein zweiter (DCC) Arduino vorhanden ist.

Die Vorgaben für den Flash-Befehl werden über den Button Dialog in den Prog_Generator gezogen. Pro RGB-LED ist es notwendig drei Zeilen zu generieren (Rot / Grün / Blau). Je nach Größe der Disco können mehrere RGB-LED eingesetzt werden. Die Zeilen mit Flash... C1, C2, C3 sooft kopieren wie RGB-LED's angesteuert werden sollen Für das Arbeitslicht kann man auf den Dialog „House“ zurückgreifen. Wichtig ist hier, dass der Eingang invertiert wird. Dann ist entweder die Disko oder das normale Licht an.

House: Simulation eines "belebten" Hauses in dem zufällig und abwechselnd nur einige der Räume beleuchtet sind

Das ist vermutlich die am häufigsten genutzte Funktion auf einer Modelleisenbahn. Mit Ihr wird ein „belebtes“ Haus nachgebildet. In diesem Haus sind zufällig nur einige der Räume beleuchtet. Die Farbe und die Helligkeit der Beleuchtungen können individuell vorgegeben werden. Es lassen sich auch bestimmte Effekte wie Fernseher flackern oder ein offener Kamin für einzelne Räume konfigurieren. Außerdem kann das Einschaltverhalten angepasst werden (Neonröhrenflackern oder langsam heller werdende Gaslampen).

Mögliche Beleuchtungstypen:

ROOM_DARK

ROOM_BRIGHT

ROOM_WARM_W

ROOM_RED

ROOM_D_RED

*

ROOM_COL0

ROOM_COL1

ROOM_COL2

ROOM_COL3

ROOM_COL4

ROOM_COL5

ROOM_COL345

FIRE

FIRE2

FIRE3

ROOM_CHIMNEY

ROOM_CHIMNEY2

ROOM_CHIMNEY3

*

ROOM_TV0

ROOM_TV0_CHIMNEY

ROOM_TV0_CHIMNEY2

ROOM_TV0_CHIMNEY3

ROOM_TV1

ROOM_TV1_CHIMNEY

ROOM_TV1_CHIMNEY2

ROOM_TV1_CHIMNEY3

NEON_LIGHT

NEON_LIGHT1

NEON_LIGHT2

NEON_LIGHT3

NEON_LIGHT4

NEON_LIGHT1D

NEON_LIGHT2D

NEON_LIGHT3D

NEON_LIGHTM

NEON_LIGHT1M

NEON_LIGHT2M

NEON_LIGHT3M

NEON_LIGHTL

NEON_LIGHT1L

NEON_LIGHT2L

NEON_LIGHT3L

NEON_DEF_D

NEON_DEF1D

NEON_DEF2D

NEON_DEF3D

SINGLE_LED1

SINGLE_LED2

SINGLE_LED3

SINGLE_LED1D

SINGLE_LED2D

SINGLE_LED3D

GAS_LIGHT

GAS_LIGHT1

GAS_LIGHT2

GAS_LIGHT3

GAS_LIGHT4

GAS_LIGHT1D

GAS_LIGHT2D

GAS_LIGHT3D

SKIP_ROOM

* = Unveränderbare Farben
Alle anderen Farben können mit dem Set_ColTab Befehl und dem Farbtest Programm angepasst werden

Ausgewählte Beleuchtungen:

Mit einem Klick in das Feld unten kann die Position zum Einfügen / Löschen der Beleuchtungen gewählt werden.

ANZAHL: 4

NEON_LIGHTL, NEON_LIGHTL, NEON_LIGHTL, NEON_LIGHTL

Lösche Raum

RGB LED Kanäle: 4

4

Minimale Anzahl der zufällig aktiven Beleuchtungen

4

Maximale Anzahl der zufällig aktiven Beleuchtungen

☒

Eingang invertieren

☒

Individuelle Zeiten

0

Minimale Zeit bis zur nächsten Änderung [sek]

0

Maximale Zeit bis zur nächsten Änderung [sek]

Abbrechen

OK

Dadurch dass die „Minimale und Maximale Anzahl der zufällig aktiven Beleuchtungen“ gleich der Anzahl der LEDs ist werden alle LEDs angeschaltet. Durch die Verwendung der „Individuellen Zeiten“ von 0 werden die Lichter sofort aktiviert. Mit dem Schalter „Eingang invertieren“ ist die Beleuchtung an, wenn die Disco aus ist. Hier wurde als Beispiel der Effekt „Neon_LIGHTL“ verwendet, mit dem mehrere Neonlichter in einem größeren Raum simuliert werden. (Hier können auch andere Effekte ausgewählt werden-einfach mal ausprobieren!)

Wenn man auch einen „Tag“ Modus haben will bei dem weder Neonlichter noch Disco aktiv ist, dann kann man zwei DCC Adressen vergeben. Das sieht dann so aus:

Aktiv	Filter	DCC Adresse	Typ	Start-wert	Beschreibung	Verteiler-Nummer	Stecker-Nummer	Beleuchtung, Sound, oder andere Effekte	Start LedNr	LEDs	InCnt	Loc InCh
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							#define TEST_TOGGLE_BUTTONS			0	0
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C_ALL, #InCh, #LocInCh, 5 Sek, 120 Sek)	0	1	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	1	C1-1	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	1	C2-2	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	1	C3-3	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	2	C1-1	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	2	C2-2	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	2	C3-3	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	3	C1-1	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	3	C2-2	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	3	C3-3	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C1, #InCh, #LocInCh, 100 ms, 750 ms)	4	C1-1	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C2, #InCh, #LocInCh, 150 ms, 800 ms)	4	C2-2	1	1
<input checked="" type="checkbox"/>		1	AnAus					Flash(#LED, C3, #InCh, #LocInCh, 75 ms, 900 ms)	4	C3-3	1	1
<input checked="" type="checkbox"/>		2	AnAus					HouseT(#LED, #InCh, 4, 4, 0, 0, NEON_LIGHTL, NEON_LIGHTL)	5	4	1	0

Man könnte die Disco noch weiter perfektionieren. Es könnten verschiedene Beleuchtungsvarianten erstellt werden welche zu verschiedenen Musikstücken passen (Langsam, Schnell, ...). Man könnte eine Spiegelkugel einbauen und, und, und ...

<https://wiki.mobaledlib.de/>

Printed on 2024/05/19 07:57

Tipp für eine Variante: Nutzt statt „Flash“ den Dialog „Blinker“ oder „Blinker-HD“, dann habt Ihr auch eine Variante für Schmuse-Songs!

Dieses Beispiel haben Dominik (Moba_Nicki) und Hardi zur Verfügung gestellt.

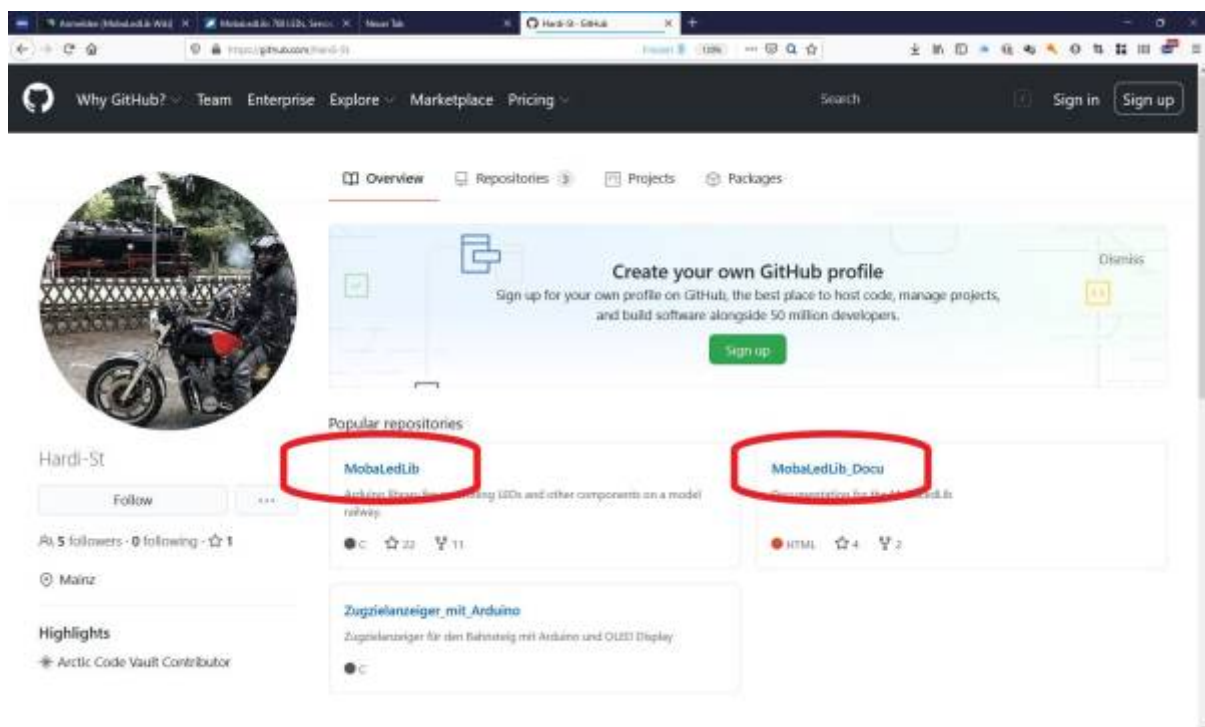
****Installation MobaLedLib aus GitHub****

Die Installation der MobaLedLib ist nicht nur über die ARDUINO IDE möglich, bei der es manchmal zu Fehlermeldungen kommt, sondern auch über eine gepackte (gezippte) Datei, die man leicht von GitHub herunterladen kann. Zu erreichen ist die GitHub-Seite über folgenden Link:

Zu erreichen ist die GitHub-Seite über folgenden Link:

<https://github.com/Hardi-St>

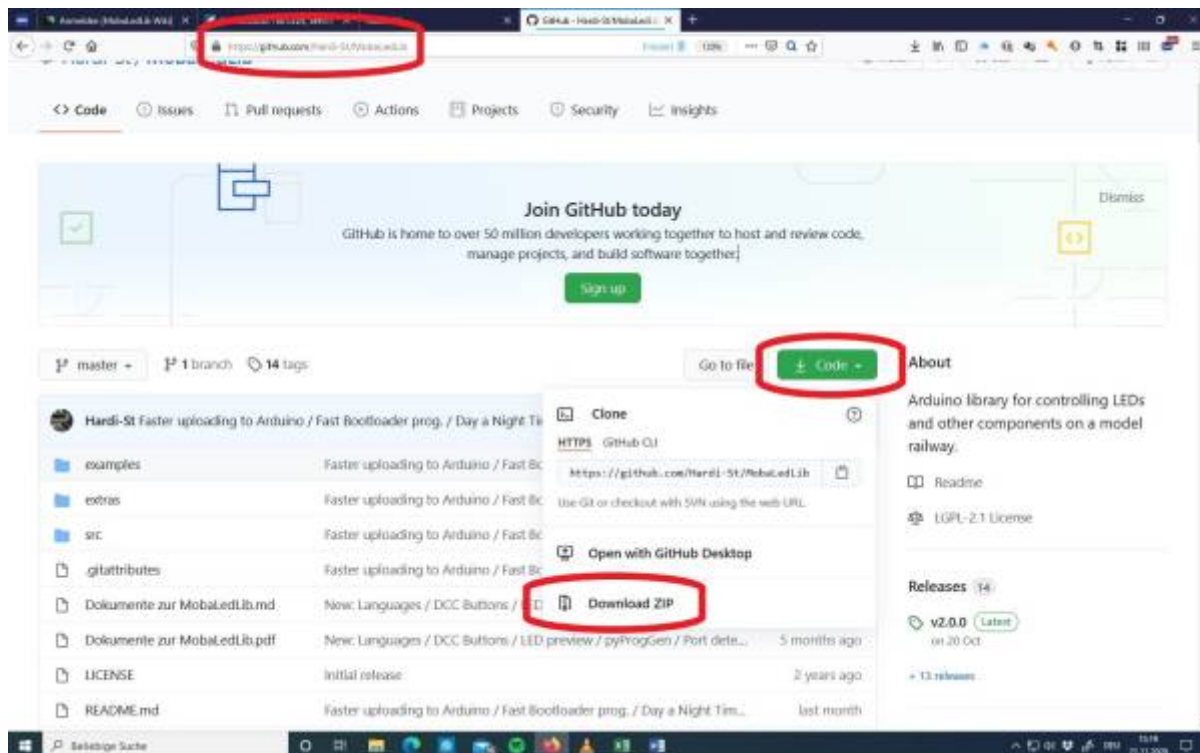
Hardi hat auf GitHub mehrere Projekte angelegt. Für die MobaLedLib sind in der folgenden Abbildung die Projekte MobaLedLib und MobaLedLib_Docu von Bedeutung.



In „MobaLedLib“ befinden sich die Dateien für die aktuelle Version, in „MobaLedLib_Docu“ die dazu gehörige Dokumentation. „MobaLedLib_Docu“ enthält außerdem ein Unterverzeichnis in dem eine **Betatest-Version** enthalten ist. Diese Testversion ist für Spezialisten und Tester vorgesehen, nicht für Einsteiger.

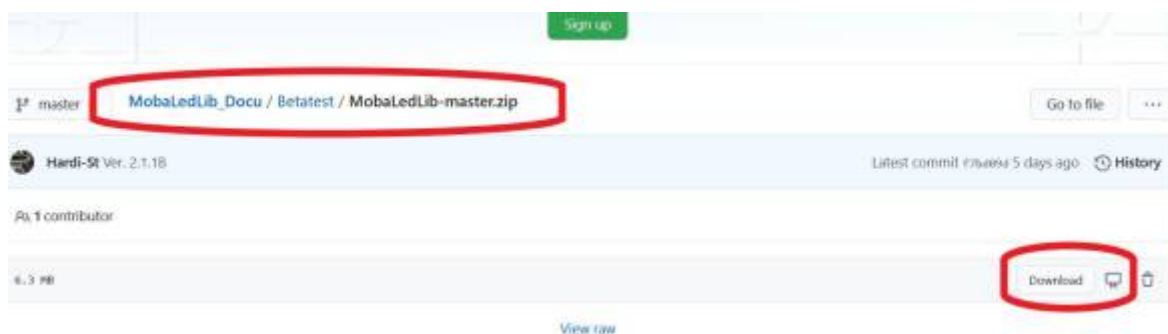
Herunterladen der aktuellen MobaLedLib-Version:

In „MobaLedLib“ die Schaltfläche „Code“ anklicken und in dem sich öffnenden Fenster „Download ZIP“ anklicken. Die gepackte Zip Datei wird in das Download-Verzeichnis des Nutzers heruntergeladen.



Herunterladen der MobaLedLib-Betatest-Version:

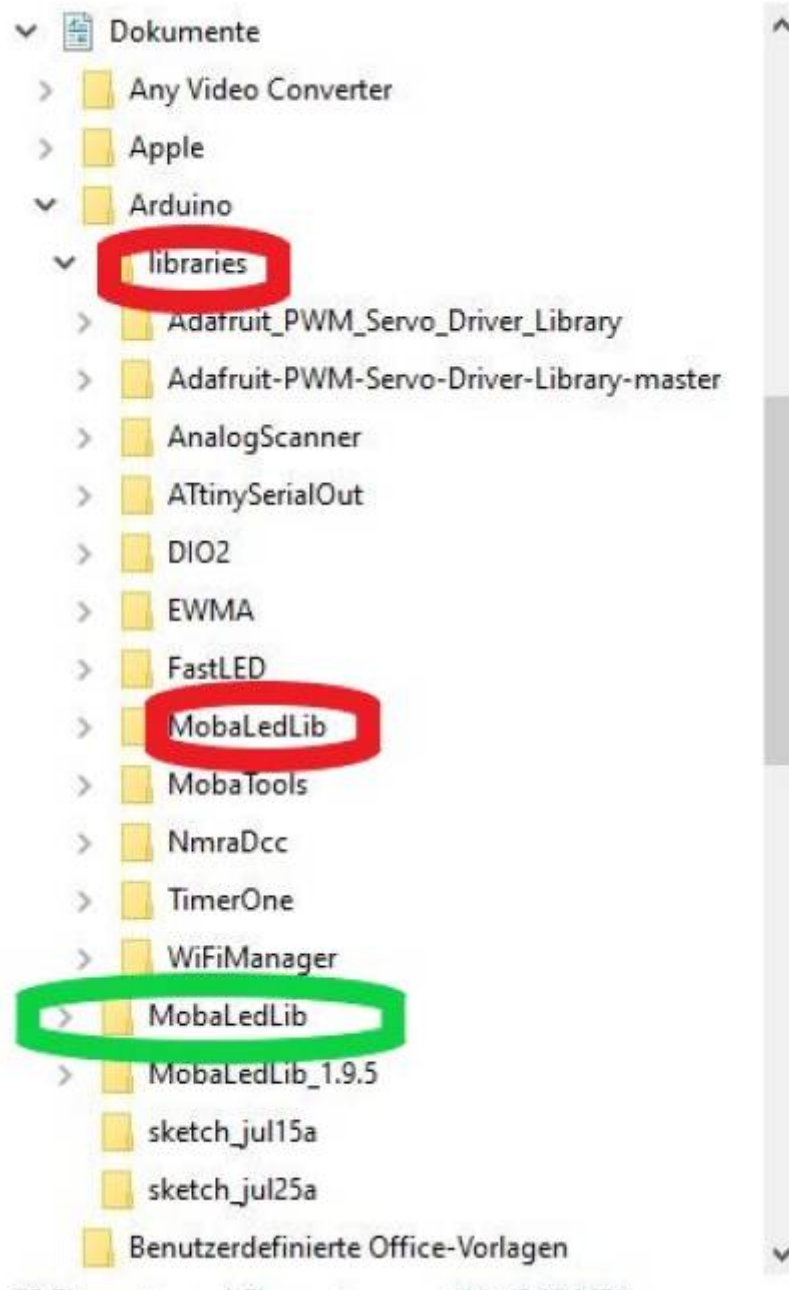
Im Unterverzeichnis MobaLedLib_Docu/Betatest/MobaLedLib-master.zip „Download“ anklicken.



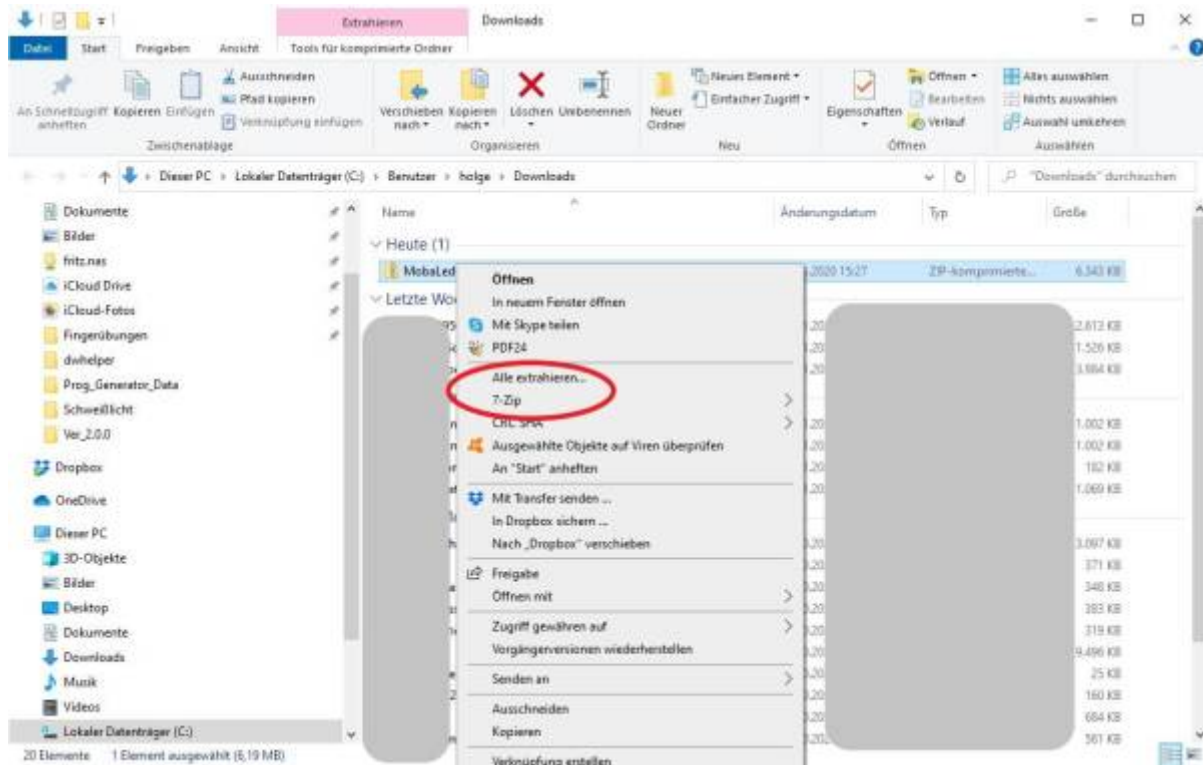
Die Zip-Datei wird in das Downloadverzeichnis heruntergeladen.

Das weitere Vorgehen ist bei beiden Versionen gleich.

MobaLedLib-Dateien werden im Nutzer-Verzeichnis unter „Dokumente“ Unterverzeichnis „Arduino“ an zwei Stellen gespeichert:



- in dem Verzeichnis „MobaLedLib“ (grün) . Hier sind durch den **Nutzer keine Änderungen** vorzunehmen. Bei der Installation wird hier später automatisch ein weiteres Unterverzeichnis mit der entsprechenden Versionsnummer angelegt.
- und in „libraries“ (rot). In diesem Unterverzeichnis muss das Verzeichnis „MobaLedLib“ gelöscht und die grade heruntergeladene neue entpackte Version hineinkopiert werden. Zum Entpacken kann man das Microsoft-Tool oder, wenn vorhanden, ein anderes Programm wie „7-Zip“ nutzen.



Nun sollte sich in „libraries“ wieder ein MobaLedLib-Unterverzeichnis befinden, dass alle notwendigen Dateien enthält. Noch heißt diese Verzeichnis allerdings „MobaLedLib-master“. GitHub versieht beim Herunterladen standardmäßig Dateien mit dem Zusatz „-master“.

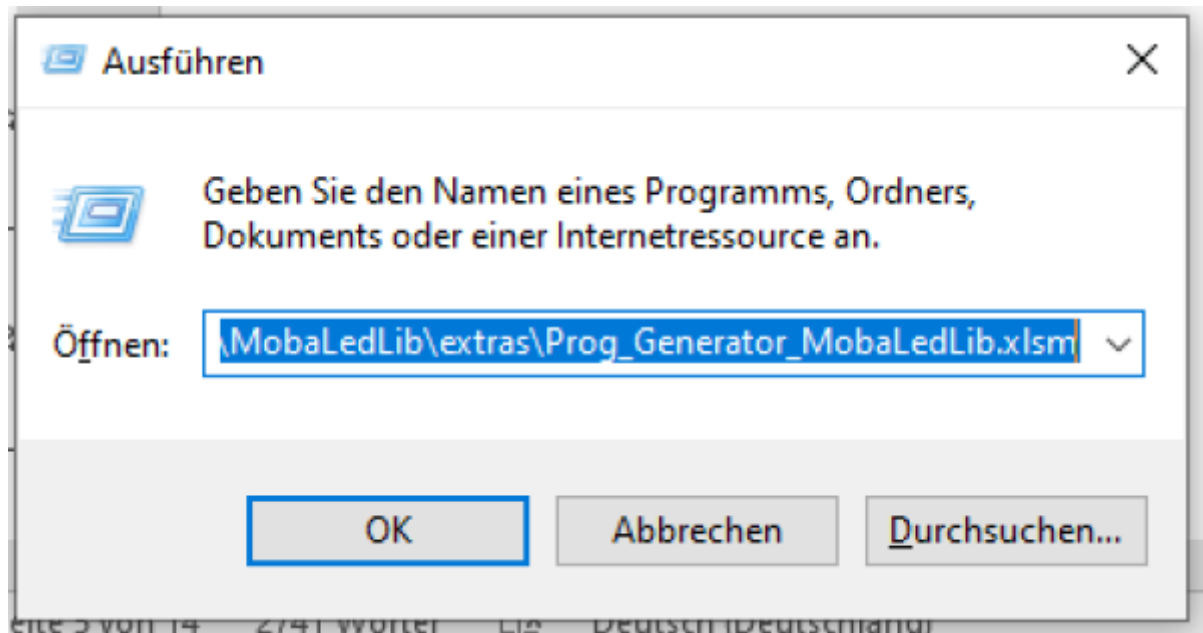


Der Name des Verzeichnisses muss von „MobaLedLib-master“ in „MobaLedLib“ geändert werden.

Nun wird wie gewohnt auf der Tastatur die WINDOWS und die „R“ Taste gleichzeitig gedrückt. In den erscheinenden „Ausführen“ Dialog die folgende Zeile kopieren:

%USERPROFILE%\Documents\Arduino\libraries\MobaLedLib\extras\Prog_Generator_MobaLedLib.xlsm

Achtung: Die Groß- und Kleinschreibung muss exakt stimmen.



Der weitere Ablauf entspricht der „Anleitung für den ersten Schnelleinstieg“.

From:

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

Permanent link:

<https://wiki.mobaledlib.de/playground/playground?rev=1606024766>

Last update: **2020/11/22 06:59**

