pyProgramGenerator - Z21 Simulator

Was macht der Z21 Simulator?

Die Z21 Simulator ermöglicht die direkte Steuerung des ARDUINOS von Modellbahnsteuerungsprogrammen aus (z.B. Rocrail, iTrain, Traincontroller, WinDigiPet). Die Simulation verhält sich gegenüber dem Steuerprogramm wie eine Roco Z21 Zentrale. Wenn das Steuerproramm auf demselben Rechner läuft, findet das Steuerprogramm die Z21 Simulation unter der IP Adresse 127.0.0.1. Das Textfeld zeigt alle gesendeten und empfangenen Kommmandos an.

Der LED-ARDUINO wird direkt über USB mit dem PC verbunden.

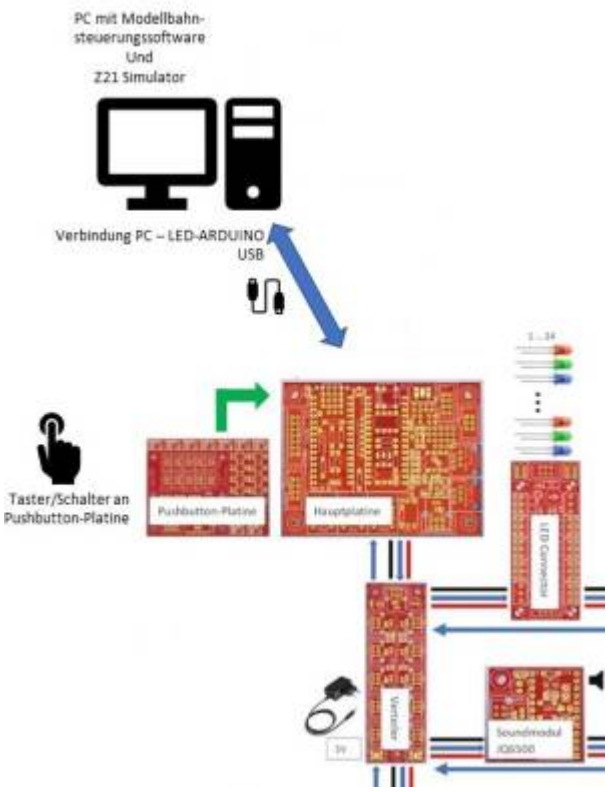
Es ist kein DCC-ARDUINO notwendig!

Direkte Steuerung über den USB-Anschluss

Einleitung

Bei der direkten Steuerung des LED-ARDUINOs ist der ARDUINO immer mit dem USB-Anschluss des Steuer-PCs verbunden. Nicht nur zum Programmieren.

Das folgende Bild zeigt die Grundkonfiguration:



Was sind die Vorteile der direkten Steuerung?

- es reicht die minimal Konfiguration für die MobaLedLib aus - ein ARDUINO, ein 100 Ohm

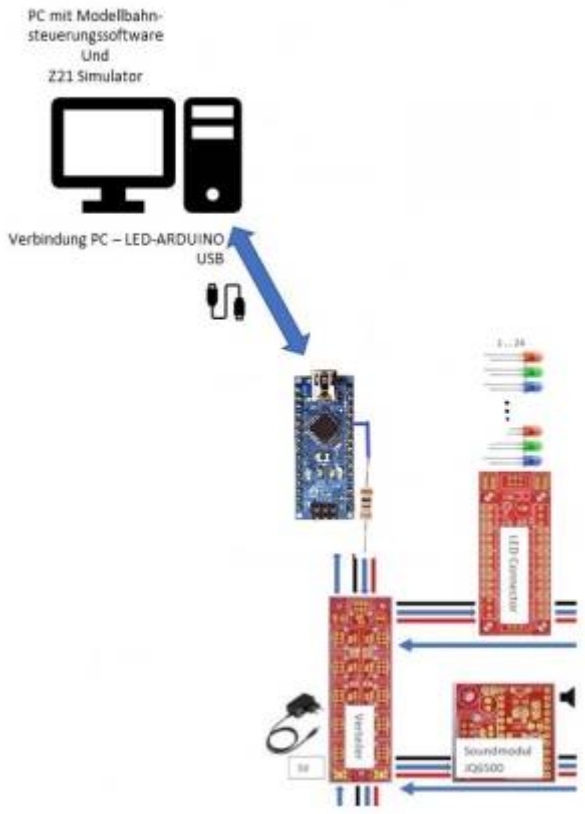
Widerstand und eine 4- oder 6-polige Buchse.

- der Steuerbus der Zentrale und auf dem Gleis wird nicht belastet - gerade bei DCC kann dies ein Problem werden, wenn komplexe Beleuchtungsszenarien geschaltet werden, können Steuerkommandos für Lokomotive verzögert werden. Was beim Bremsen zu ungenauen Haltepunkten führen kann.
- die Informationsübertagung ist etwas schnelle, da die Verzögerung durch die Zentrale und das Protokoll entfallen
- es ist möglich Informationen vom ARDUINO an die Software zurückzumelden, z.B. Tasterdrücke, Schalterstellungen oder Reedkontakte (in Vorbereitung)

Die Lösung hat aber auch Nachteile:

- der ARDUINO muss dauerhaft mit dem USB Ausgang des Steuer-PCs verbunden sein. Das hat folgende Konsequenzen, die man beachten muss:
  - über den ARDUINO sind alle Komponenten der MobaLedLib mit der PC Masse verbunden. Die MobaLedLib darf daher auf keinen Fall irgendeine direkte Verbindung zu den Gleisen haben, z.B. für Besetztmeldungen, diese dürfen nur über Optokoppler laufen.
  - Beim Anschließen von zusätzlichen Netzteilen für die Stromversorgung ist darauf zu achten, dass die +5V Stromversorgung des ARDUINO von der zusätzlichen Stromversorgung auch wirklich getrennt ist. Bei einem Fehler kann der USB Anschluss des PCs zerstört werden.

Das folgende Bild zeigt die minimal Konfiguration:



Wie funktioniert das?

Die Modellbahnsteuerungssoftware muss jetzt irgendwie ihre Kommandos an den ARDUINO über die

USB-Schnittstelle schicken. Da keine Modellbahnsteuerungssoftware - zumindest bis jetzt nicht - die MobaLedLib kennt müssen wir das anders lösen.

Und da kommt der Z21-Simulator ins Spiel.

„Also, wat is en Z21 Simulator? Da stelle mehr uns janz dumm. Und da sage mer so: En Z21 Simulator, dat is ene jroße schwarze Box, die hat hinten un vorn e Schnittstell. Dat eine Schnittstell, dat is de Z21 Schnittstell. Und dat andere Schnittstell, dat krieje mer später.“

Die Z21-Schnittstelle

Wenn man eine Modellbahnsteuerungssoftware dazu bringen möchte, mit einer neuen Hardware zusammenzuarbeiten, ist es am einfachsten der Software eine Schnittstelle zu einer Zentrale anzubieten, die sie schon kennt. Hier gibt es mehrere Möglichkeiten. Früher wurde gerne die Intellibox-Schnittstelle dazu verwendet.

Seit ROCO seine Z21 Zentrale auf den Markt gebracht hat, die Beschreibung der Schnittstelle zu Modellbahnsteuerungssoftware veröffentlicht hat und erlaubt diese nachzubilden (was ESU z.B. ausdrücklich für die ECOS-Schnittstelle untersagt) ist die moderne Z21 Schnittstelle das Mittel der Wahl.

Wir müssen also eine Software auf dem PC laufen lassen, die die Z21-Schnittstelle simuliert und der Modellbahnsteuerungssoftware vorgaukelt sie wäre eine Roco Z21. Und diese Software ist der Z21-Simulator.

Was die Z21 Schnittstelle so interessant macht, ist, dass sie ein LAN Protokoll ist. Das heißt, der Z21-Simulator kann auf demselben PC wie die Modellbahnsteuerungssoftware laufen oder auf irgendeinem anderen Gerät im LAN. Damit ist es auch möglich den Z21-Simulator auf einem Raspberry laufen zu lassen und den ARDUINO an die USB-Schnittstelle des Raspberrys anzuschließen. Damit könnte man die begrenzte Länge der USB-Kabel von max 5m beliebig erweitern. Man könnte an den Raspberry auch mehrere ARDUINOS anschließen, die jeweils für einen Teilbereich der Modellbahn zuständig sind. (Diese Funktion ist in Vorbereitung)

Die ARDUINO-Schnittstelle

Die andere Schnittstelle des Z21-Simulators ist die USB-Schnittstelle. Genauer gesagt wird über die USB-Schnittstelle eine serielle Kommunikation simuliert. Diese serielle Kommunikation entspricht genau der Kommunikation zwischen dem DCC-ARDUINO und dem LED-ARDUINO, wenn eine DCC-Digital Zentrale verwendet wird. Für den LED-ARDUINO ist es deshalb überhaupt kein Unterschied, ob er über eine DCC-Digital-Zentrale und den DCC-ARDUINO die Kommandos erhält oder über den Z21-Simulator und den USB-Anschluss.

Wichtig zu wissen ist, dass der LED-ARDUINO im DCC-Modus programmiert werden muss, damit er mit dem Z21-Simulator zusammenarbeiten kann.

Der Z21-Simulator

Wo finde ich den Z21-Simulator?

Wahrscheinlich haben die meisten MobaLedLib-Anwender der Z21-Simulator schon auf ihrem PC. Der **Z21-Simulator** ist nämlich ein Teil des **Farbtest-Programms**, dass zum Überprüfen und Einstellen der Farben verwendet werden kann. Der Farbtester

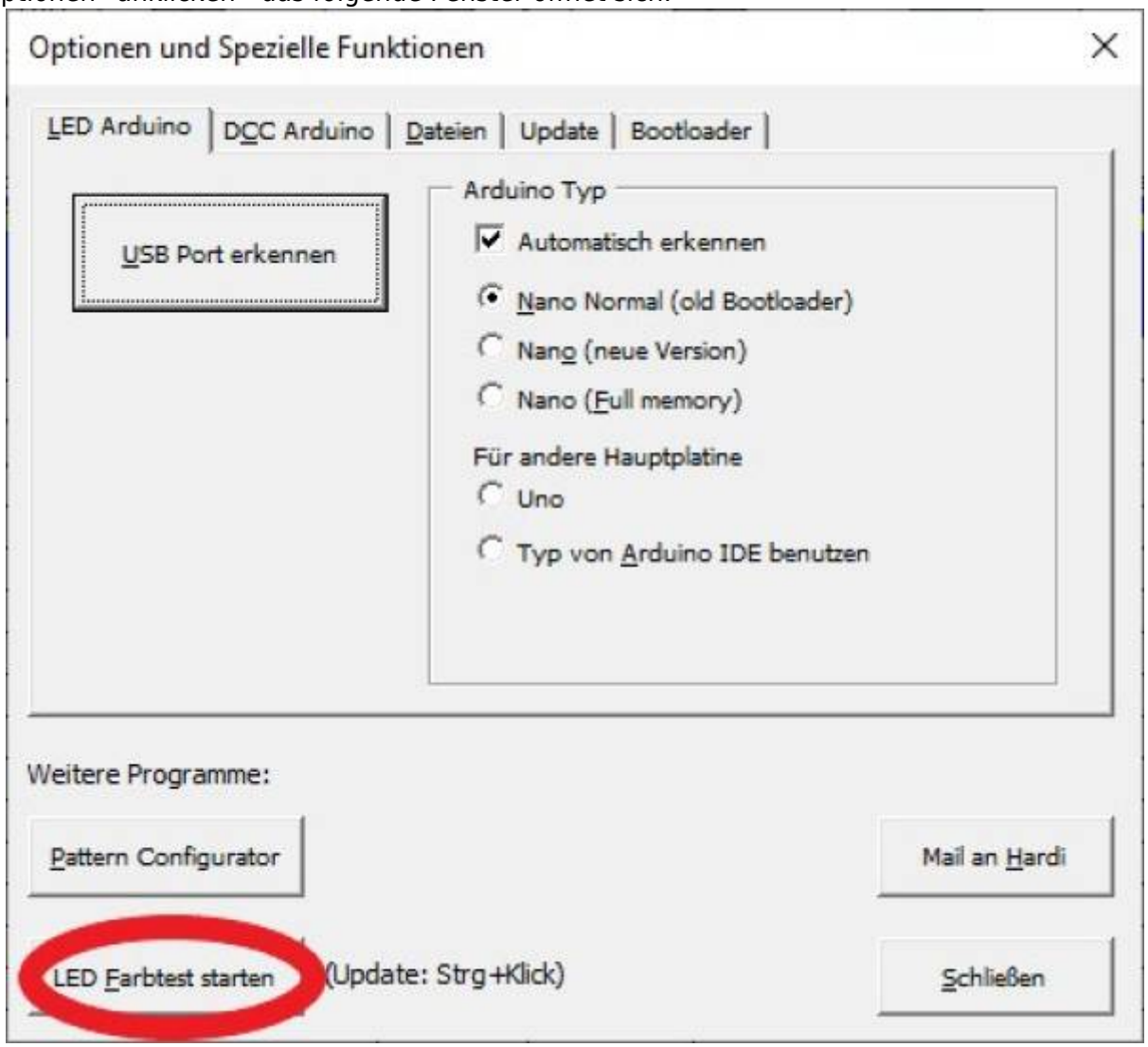
Genauer gesagt, Z21-Simulator und Farbtester sind Teil der **pyProgrammGenerators pyProg_Generator_MobaLedLib.exe**, die auch noch einige weitere Funktionen bietet. pyProgrammGenerator

Wie installiert man den Z21-Simulator?

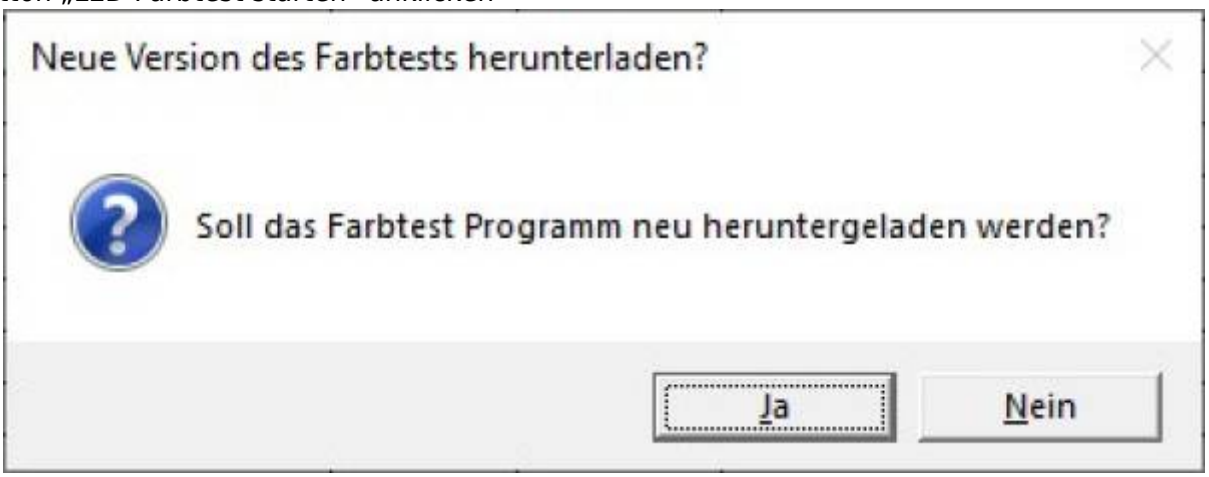
Der Z21-Simulator ist ein Teil des pyProgrammGenerators. Die Installation des pyPrgrammGenerator unter LINUX und Mac ist hier beschrieben: Installation auf einem Raspberry und Installation auf einem Mac

Wenn man den Excel „Programm-Generator“ benutzt ist die Installation viel einfacher:

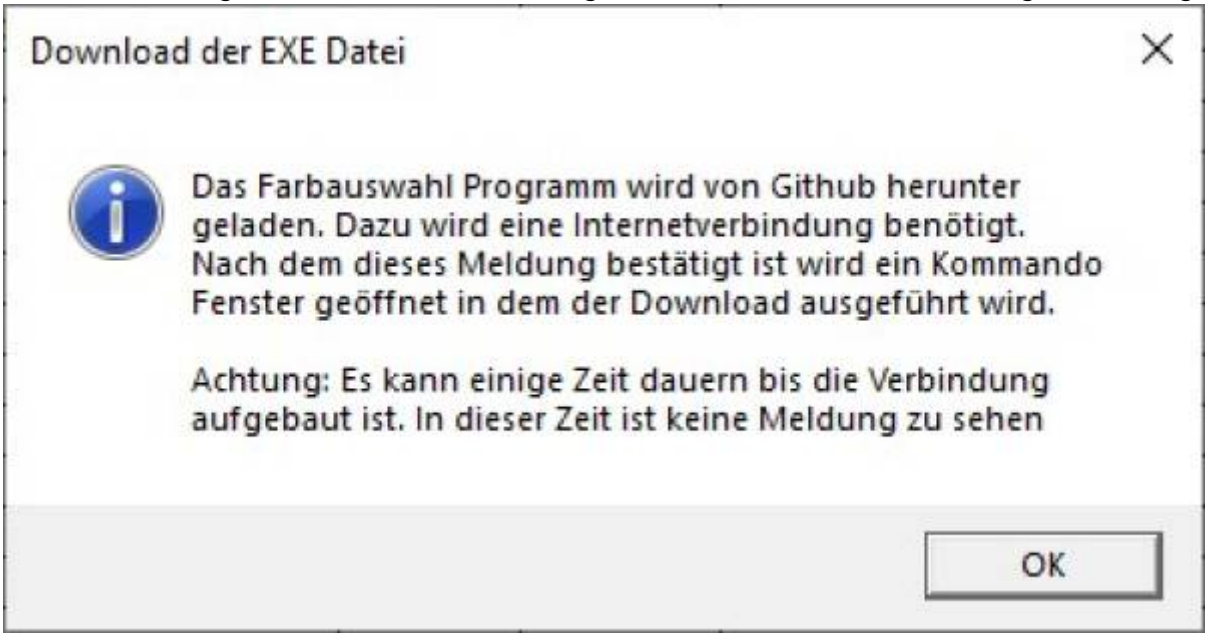
1. Excel „Programm-Generator“ öffnen
2. „Optionen“ anklicken - das folgende Fenster öffnet sich:



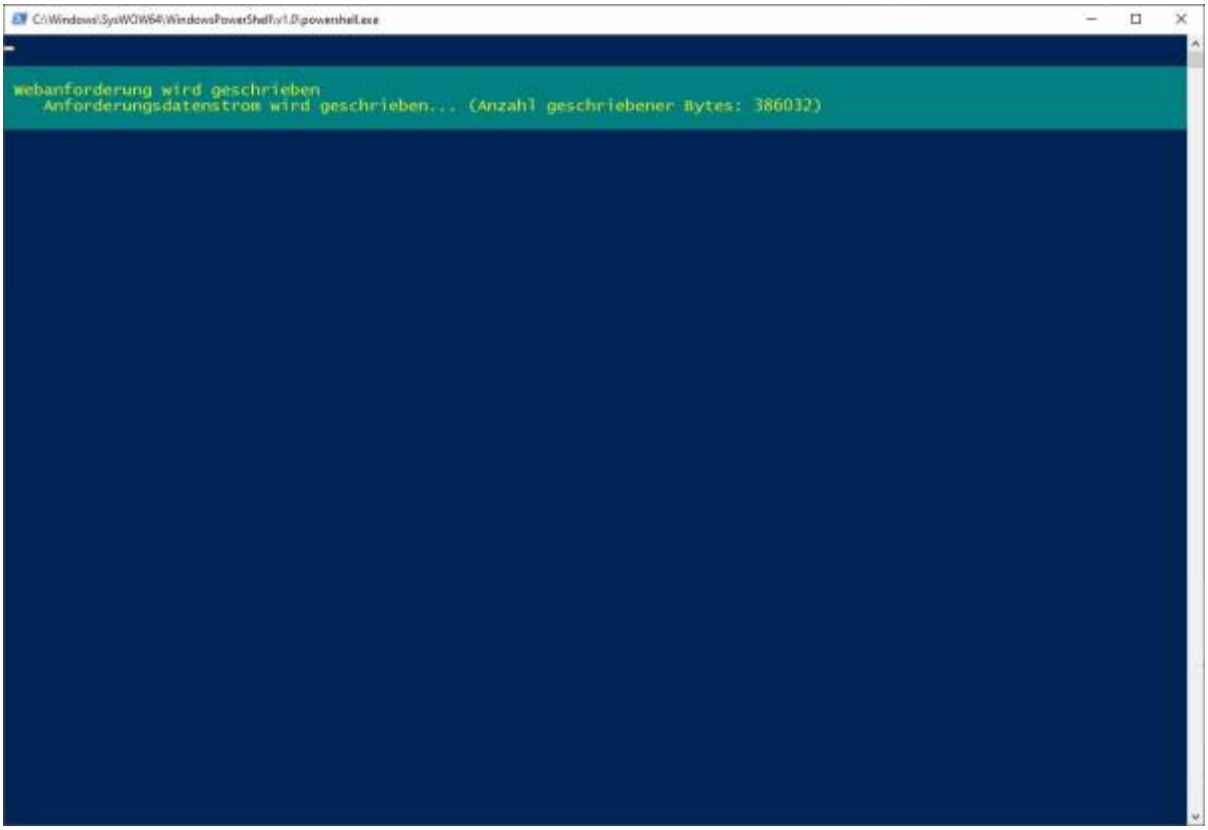
3. Button „LED Farbtest starten“ anklicken



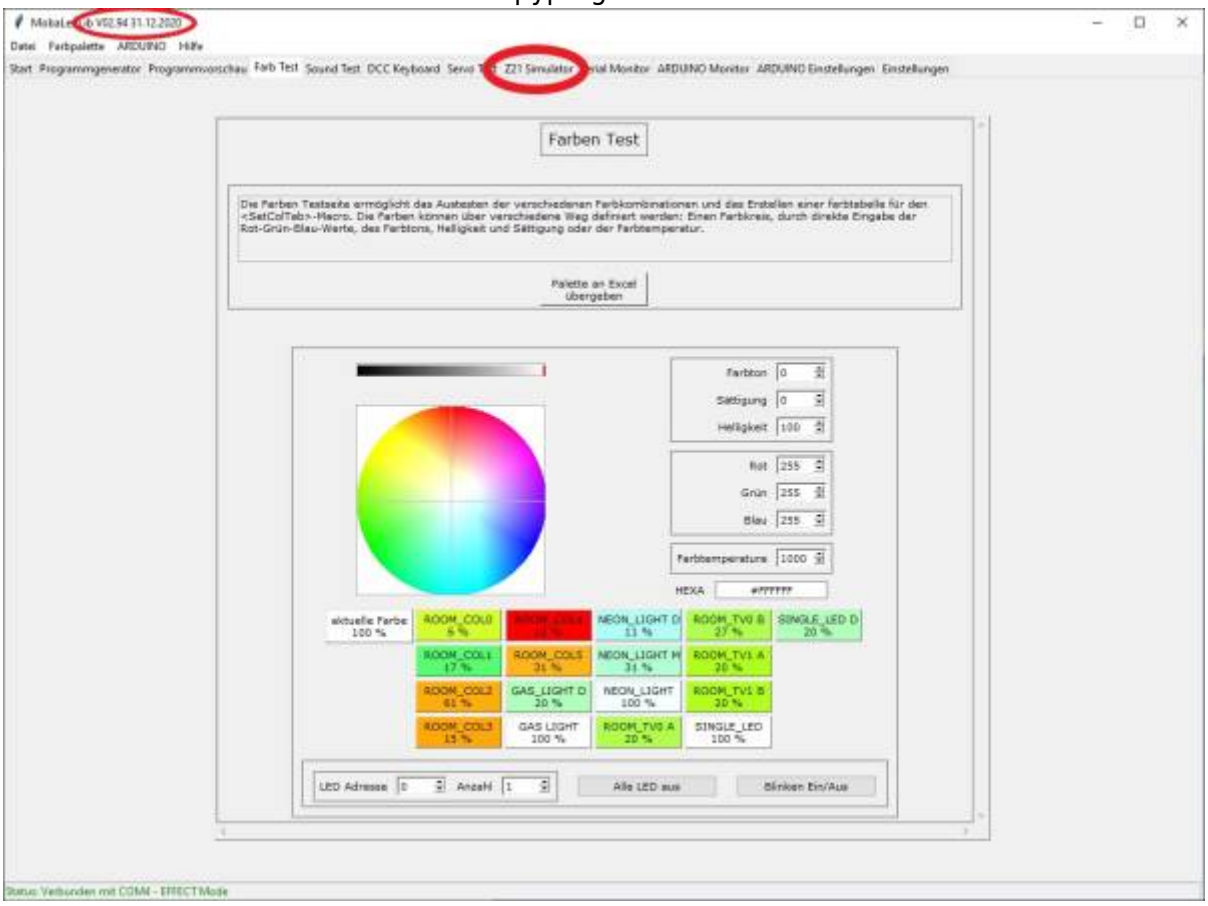
4. Wenn noch das Programm noch nicht heruntergeladen wurde öffnet sich der folgende Dialog:



5. herunterladen mit „ja“ bestätigen
6. nochmal bestätigen und der Download beginnt:



7. danach öffnet sich die Farbtest-Seite des pyprogramGenerators:



8. Bitte die Version oben Links am Fensterrand überprüfen. Es sollte mindestens Version V2.94 sein. Wenn nicht, dann das Programm schließen und den Button „LED Farbtest starten“ zusammen mit der STRG-Taste anklicken. Es wird dann die neueste Version heruntergeladen.
9. Den Reiter „Z21-Simulator“ anklicken
10. Die Z21-Simulator-Seite öffnet sich.

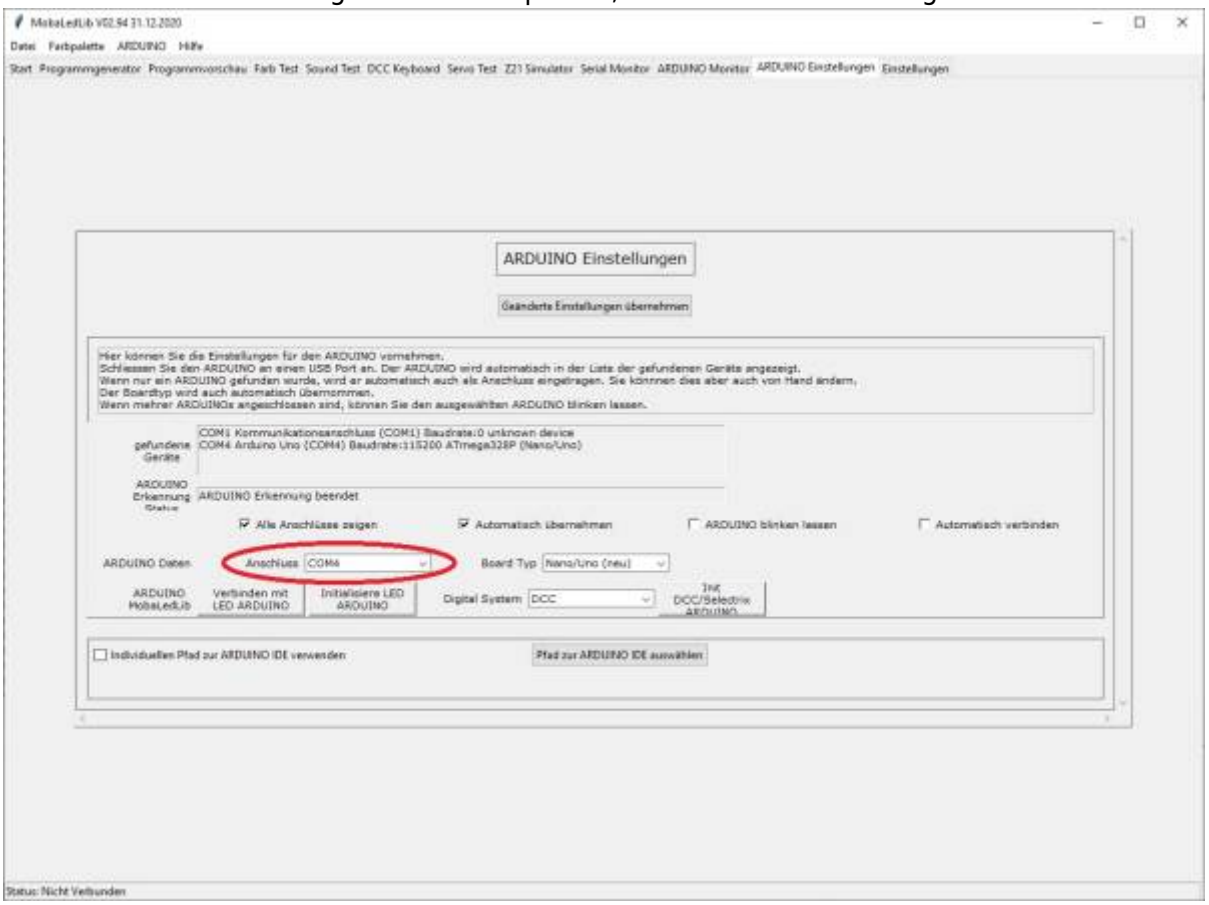
Wie startet man den Z21-Simulator?

Der Z21 Simulator kann folgendermaßen gestartet werden:

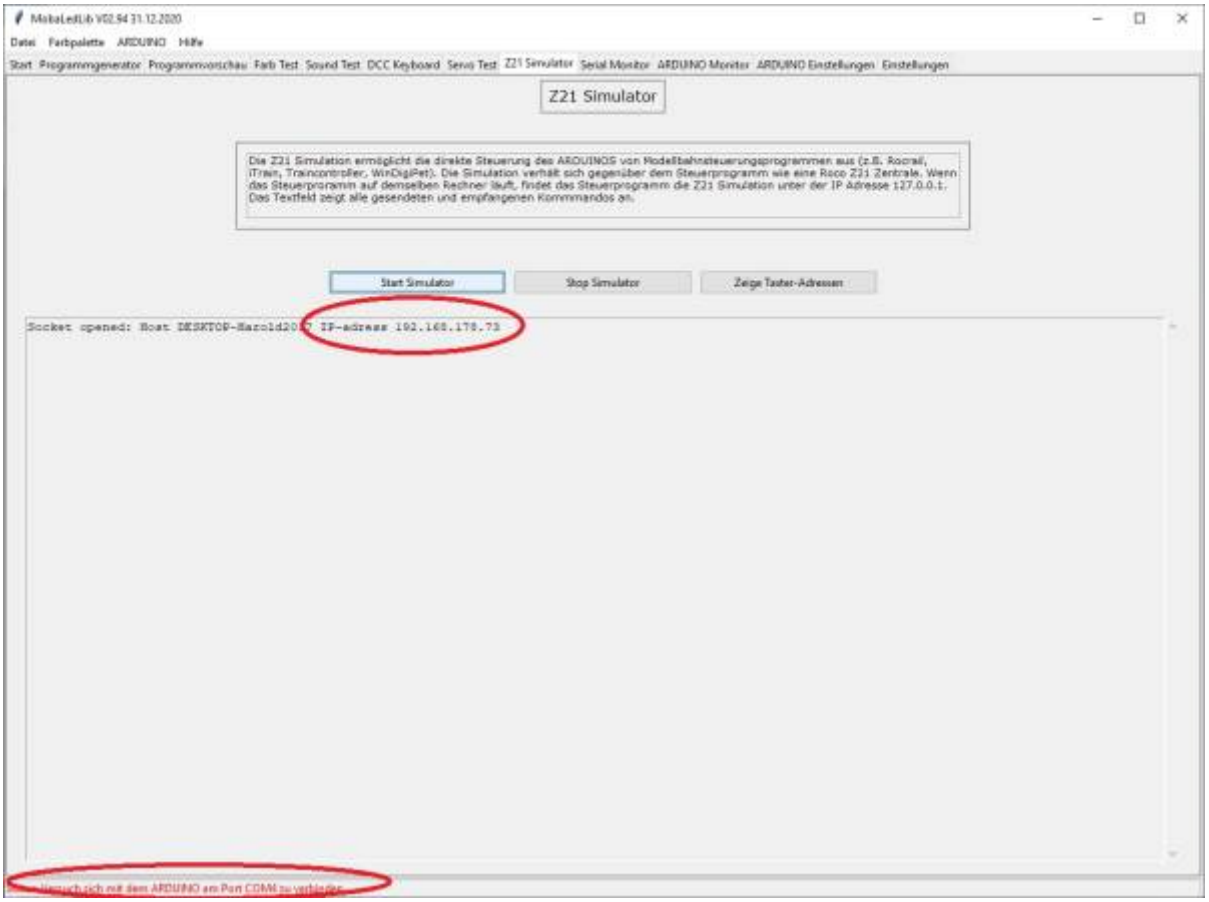
1. Über den Startbutton auf der Z21-Simulator-Seite
2. Automatisch beim Start des pyProgrammGenerators durch eine Kommandozeilenoption

Start des Z21-Simulators mit Startbutton

1. ARDUINO an den USB-Anschluss des PC anschließen
2. pyProgrammGenerator starten
3. auf der ARDUINO-Einstellungen-Seite überprüfen, ob der ARDUINO richtig erkannt wurde



4. die geänderten Einstellungen speichern
5. den Reiter Z21-Simulator auswählen
6. der Z21-Simulator wird gestartet. Im Meldungsfenster wird angezeigt, dass der Port für den Z21-Simulaor geöffnet wurde. Außerdem wird die IP-Adresse des PCs angezeigt. Diese IP-Adresse wird nachher beim Einrichten der Modellbahnsteuerungssoftware oder der Smartphone-App benötigt.In der Status-Zeile ganz unten wird die Meldung angezeigt, dass eine Verbindung zum ARDUINO aufgebaut wird.



7. Wenn alles funktioniert hat, und der Z21-Simulator auch die Verbindung zum ARDUINO herstellen konnte, wird die Meldung „Z21 Simulator started“ angezeigt.
8. Jetzt kann die Modellbahnsteuerungssoftware oder die Smartphone App die MobaLedLib steuern. Wie das genau geht, wird weiter unten beschrieben. -

Start des Z21-Simulators über Kommandozeilenoption

Da es nicht sehr komfortabel ist, jedes mal, wenn man die MobaLedLib mit der Modellbahnsteuerungssoftware steuern möchte, den Z21-Simulator von Hand zu starten, gibt es eine Kommandozeilenoption, mit der man das Programm anweisen kann, den Z21 Simulator automatisch zu starten.

Die Option zum einschalten des Z21-Simulators lautet: –z21simulator True

Die komplette Kommandozeile würde also so aussehen: pyProg_Generator_MobaLedLib –z21simulator True

ACHTUNG: Diese Funktion ist in Vorbereitung und noch nicht implementiert!

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

Permanent link:
https://wiki.mobaledlib.de/anleitungen/spezial/pyprogramgenerator/z21simulator?rev=1609684776

Last update: **2021/01/03 15:39**