Installation auf einem Raspberry

ACHTUNG: Die Seite ist noch in ARBEIT - die Angaben hier beziehen sich auf ein alte Version

Die Installation der MobaLedLib auf einem Raspberry Pi wird folgendermassen durchgeführt:

Vorraussetzung

Raspberry Pi 3B, B+ oder 4 mit installiertem Rasbian. Ich empfehle die neueste Version zu installieren.

Schritt 1: Die ARDUINO IDE auf dem Raspberry Pi installieren

Im Prinzip erfolgt die Installation der ARDUINO IDE auf dem Raspberry ähnlich der Installation auf einem Windows PC. Da es einige kleine aber wichtige Unterschiede gibt, findest Du im folgenden die angepasste Anleitung:

Installation ARDUINO IDE

Los geht’s

Als Einstieg und zum Ausprobieren reichen

- ein Arduino (Original oder preiswerter Clone)
- ein paar WS2812 RGB-LEDs und
- drei Kabel
- ein Raspberry PI 3 B, B+, 4 oder 5

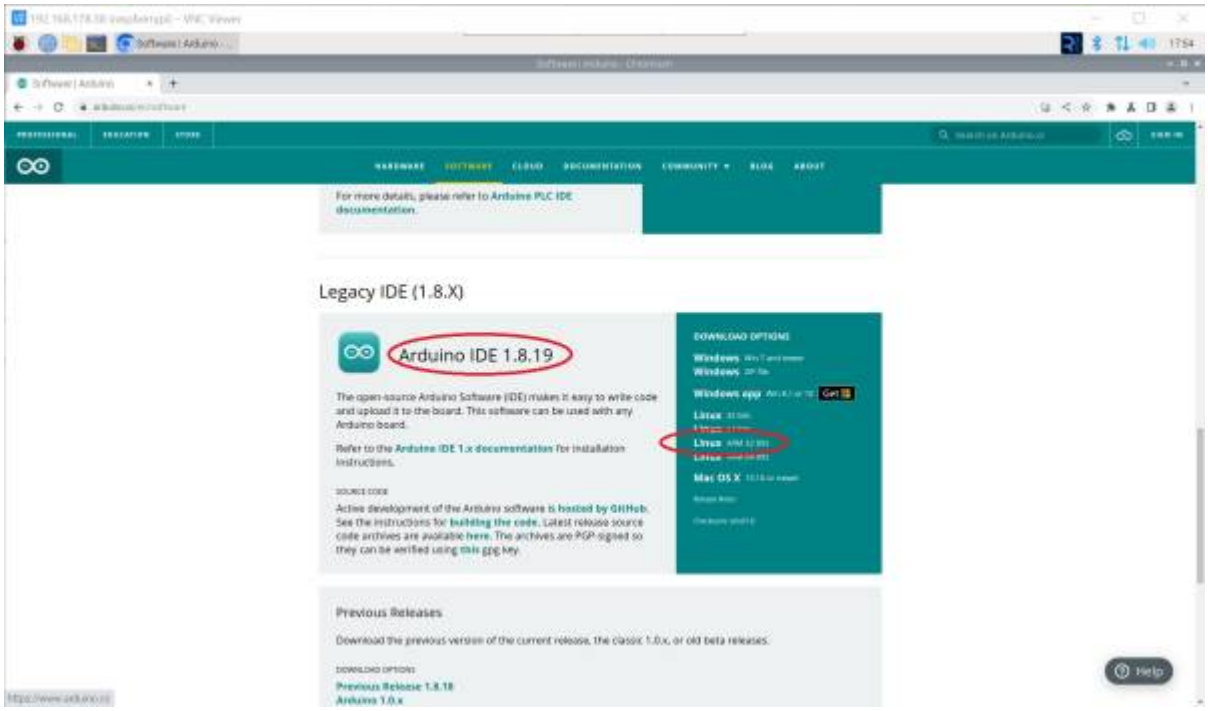
völlig aus. Der Arduino kann ein Uno, Nano oder alles, was die MLL unterstützt.

Diese Anleitung gilt für eine Raspberry Umgebung mit einem installierten Raspbian.

Zunächst muss man die aktuelle Arduino Entwicklungsumgebung IDE (Integrated Development Environment) herunterladen und installieren. Der Arduino WEB-Editor oder die Arduino App können für die MobaLedLib nicht genutzt werden.

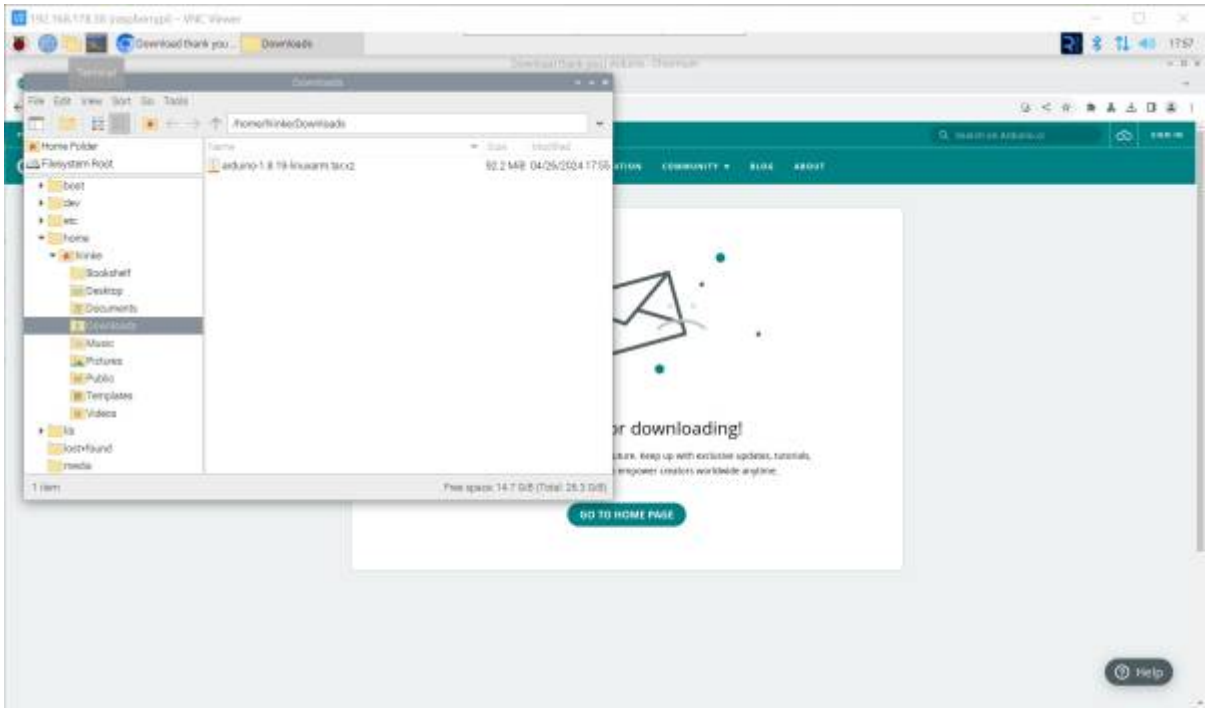
Die ARDUINO Download Seite erreicht man über den Link: https://www.arduino.cc/en/Main/Software

Es muß die **Version 1.8.19** verwendet werden. ACHTUNG: Die 2.xx Version geht leider nicht.



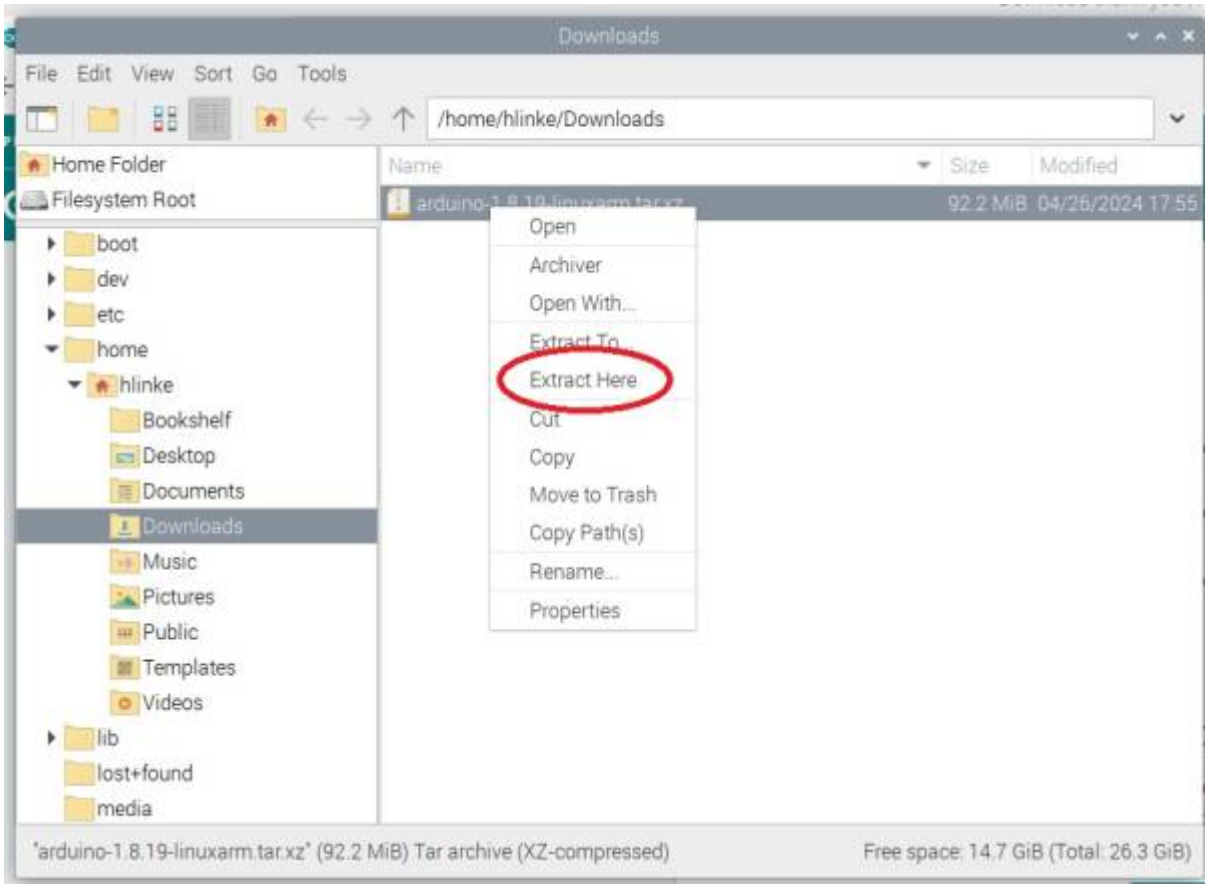
Für den Raspberry Pi den **LINUX ARM 32 Installer** auswählwen und anklicken.

Anschließend wird man aufgefordert, für das Arduino Projekt zu spenden. Es ist gut, wenn man das großartige Projekt auf diese Weise unterstützt. Das Programm kann aber problemlos auch ohne Spenden heruntergeladen werden. Das heruntergeladene Programm, hier „arduino-1.8.19-linux.arm.tar.gz“ findet man in dem „Downloads„-Ordner des Raspberry.

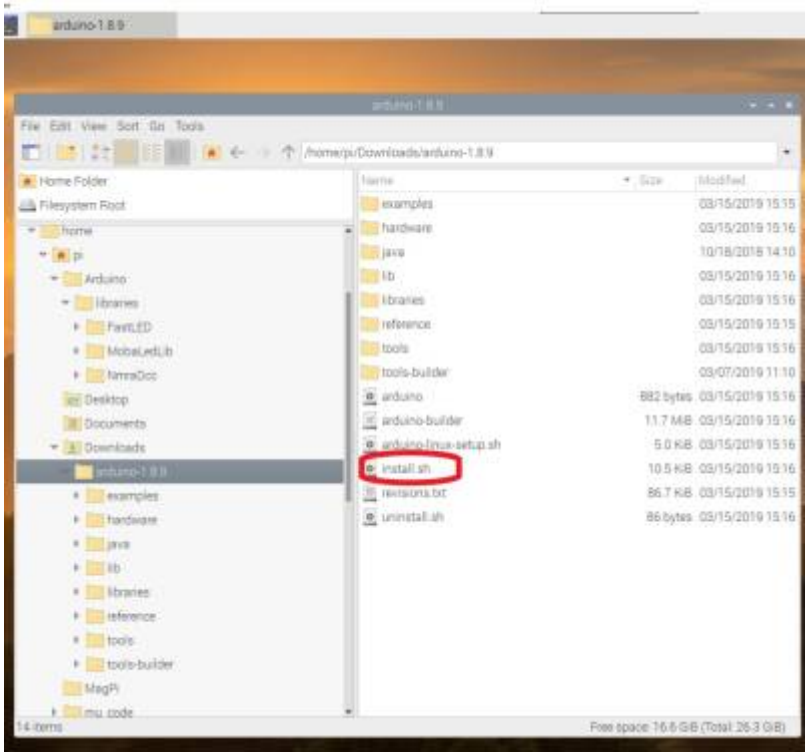


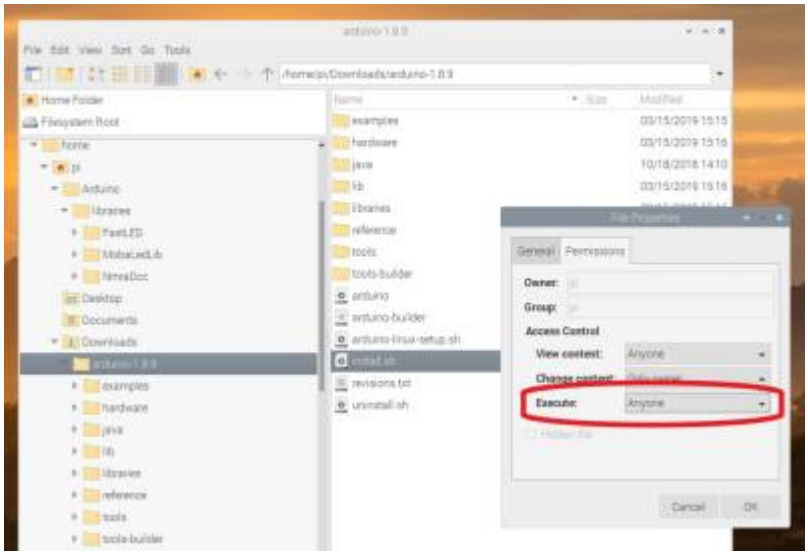
Zur Installation wird das Programm entpackt. Ich habe es gleich im Downloads Ordner entpackt. Ist nicht optimal, aber funktioniert.

Auf den Dateinamen Rechts-klicken und „Extract Here“ auswählen.



Nach dem Entpacken in das entpackte Verzeichnis wechseln und die Datei install.sh doppelt anklicken. Es kann sein, dass die Datei nicht ausgeführt werden kann. Dann auf die Datei Rechts-klicken. Die Eigenschaften (properties) auswählen. Und bei „Execute“ „Anybody“ auswählen.





Das Installationsscript erzeugt ein Desktop Icon für die ARDUINO Benutzeroberfläche.

Schritt 2: Die MobaLedLib installieren

Die Installation der MobaLedLib über die ARDUINO IDE erfolgt genauso, wie bei der original MLL, wie es hier beschrieben ist:

Anleitung für den ersten Schnelleinstieg

Die ARDUINO IDE aufrufen durch Doppelklick auf das ARDUINO-Icon auf dem Desktop.

Nun den Arduino an einen freien USB-Port anschließen. Der Raspi installiert nun bei erstmaliger Verbindung USB/Geräte-Treiber für den Arduino. Bei fehlerfreier Installation sollten Serielle Schnitstelle und Arduino, wie im folgenden Bild dargestellt, aufrufbar sein. Die Bezeichnung der Seriellen-Schnittstelle ist abhängt von der Belegung der Schnittstellen des Raspi mit anderen Geräten.

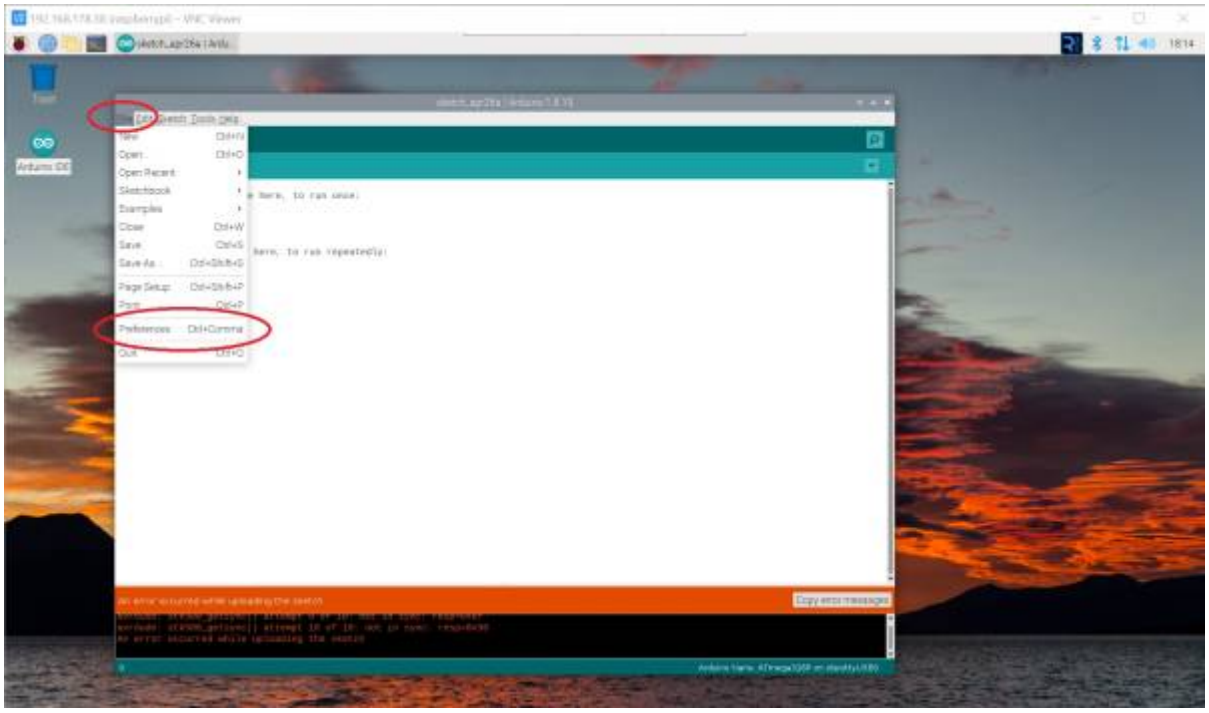
Für Einsteiger ist möglicherweise ein kleiner zusätzlicher Test interessant. Mehr dazu am Ende der Anleitung unter Test.

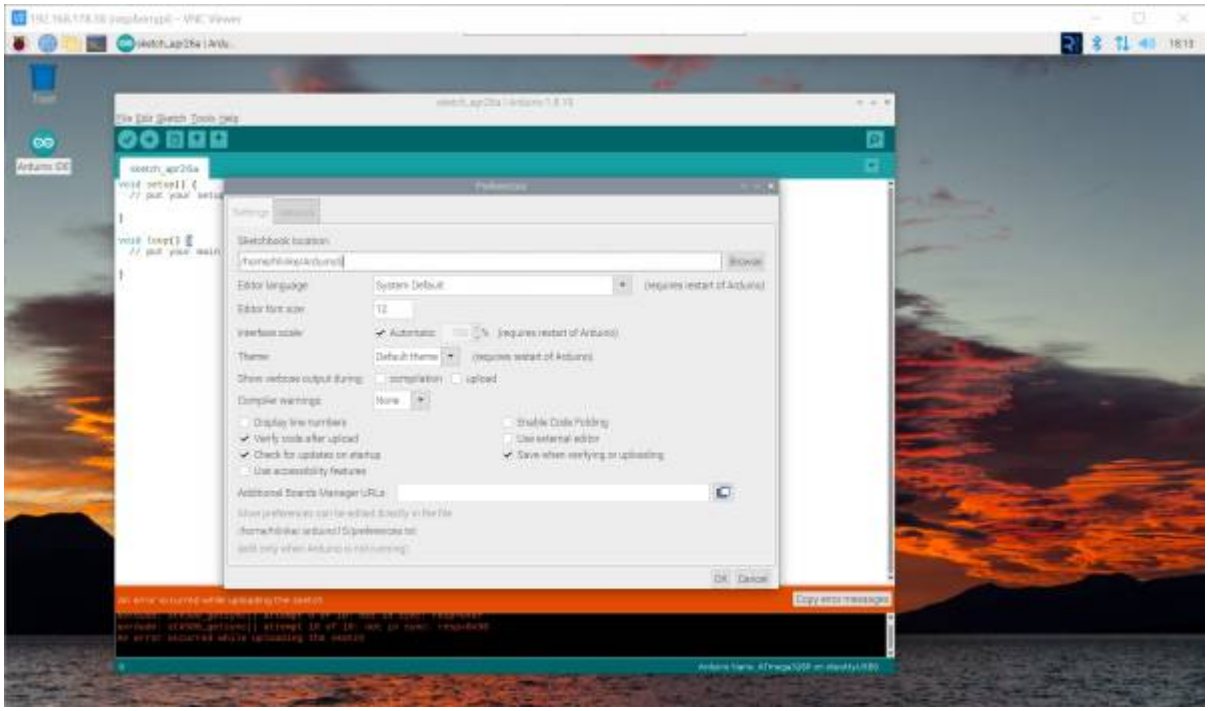
Überprüfung Sketch-Ordner

Damit alle Funktionen der MobaLedLib aufgerufen werden können, muss der Pfad zum Sketch-Ordner auf dem Standard-Pfad stehen.

```
/home/<user>/Arduino
```

Dies kann über das Menü innerhalb der Arduino IDE gemacht werden.



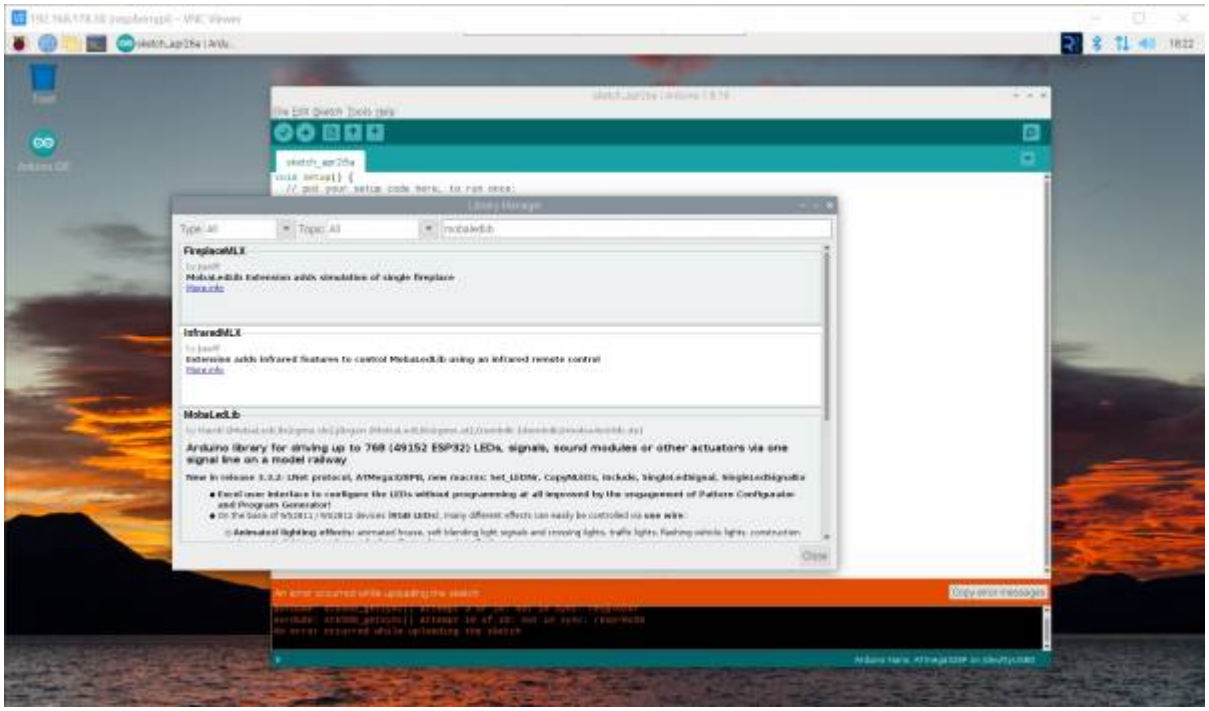


Zum Installieren der MLL in der ARDUINO IDE Tools - Manage Libraries aufrufen:

Es öffnet sich der Library Manager (der Start kann auf dem Raspi länger dauern, als Geduld ...).

Im Filter **MobaLedLib** eingeben und warten ...

Nach einiger Zeit erscheint die Beschreibung der MobaLedLib



Etwas nach unter scrollen, bis die „install“ Schaltfläche sichtbar ist und anklicken.

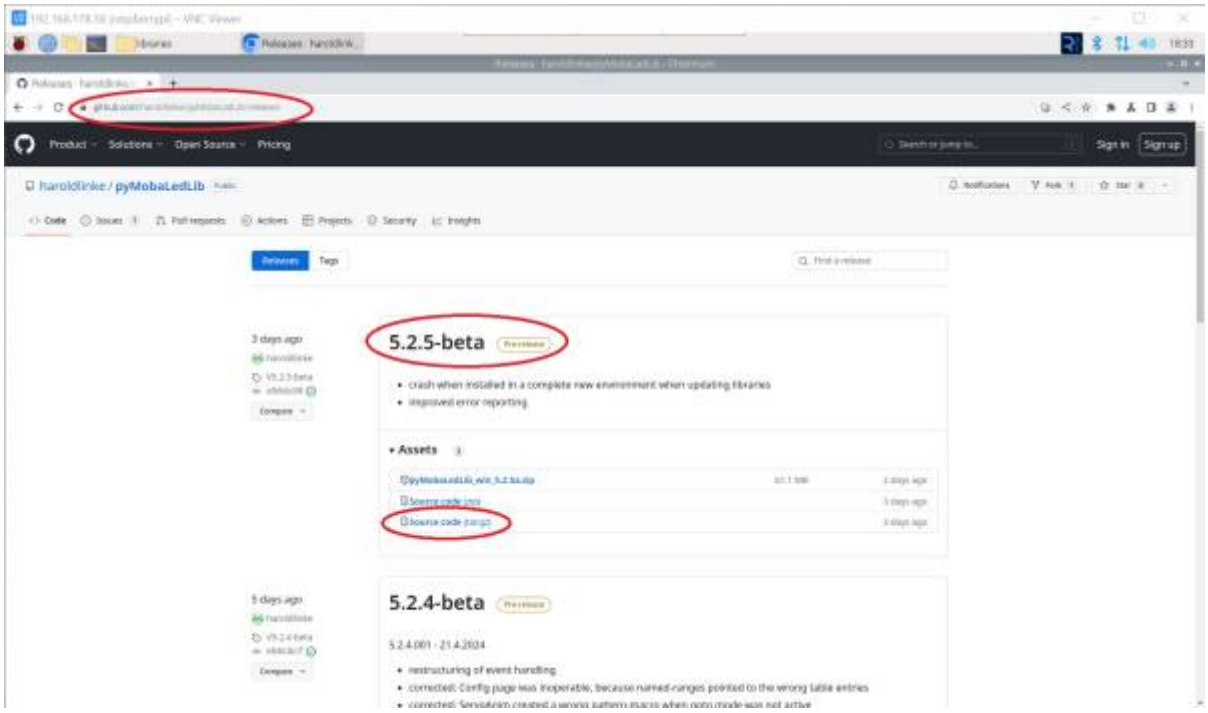
Es kommt die Frage ob die zusätzlichen Bibliotheken installiert werden sollen.

Diese Frage mit „Install all“ beantworten.

Jetzt ist die ARDUINO und MLL Installation. Normalerweise wird jetzt mit dem Excel-Programm weiter gemacht. Da Excel auf dem RASPI aber nicht läuft, benötigen wir jetzt die pyMobaLedllb.

PyMobaLedLib installieren

Den neuesten Release der PyMobaLedLib von GitHub https://github.com/haroldlinke/pyMobaLedLib/releases herunterladen.



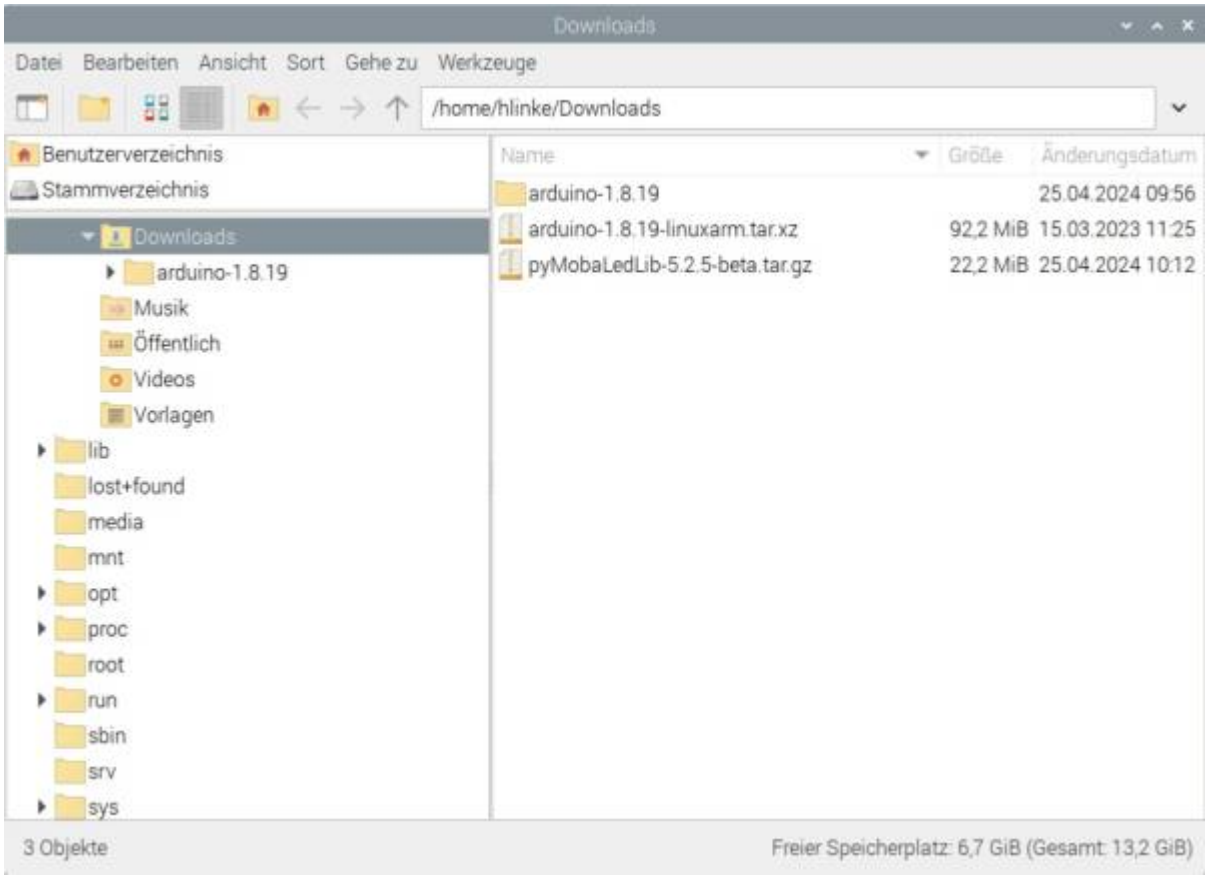
Hier ist die neueste Version die 5.2.5-beta.

Auf **Assets** klicken um die zu dem Release gehörenden Dateien zu sehen.

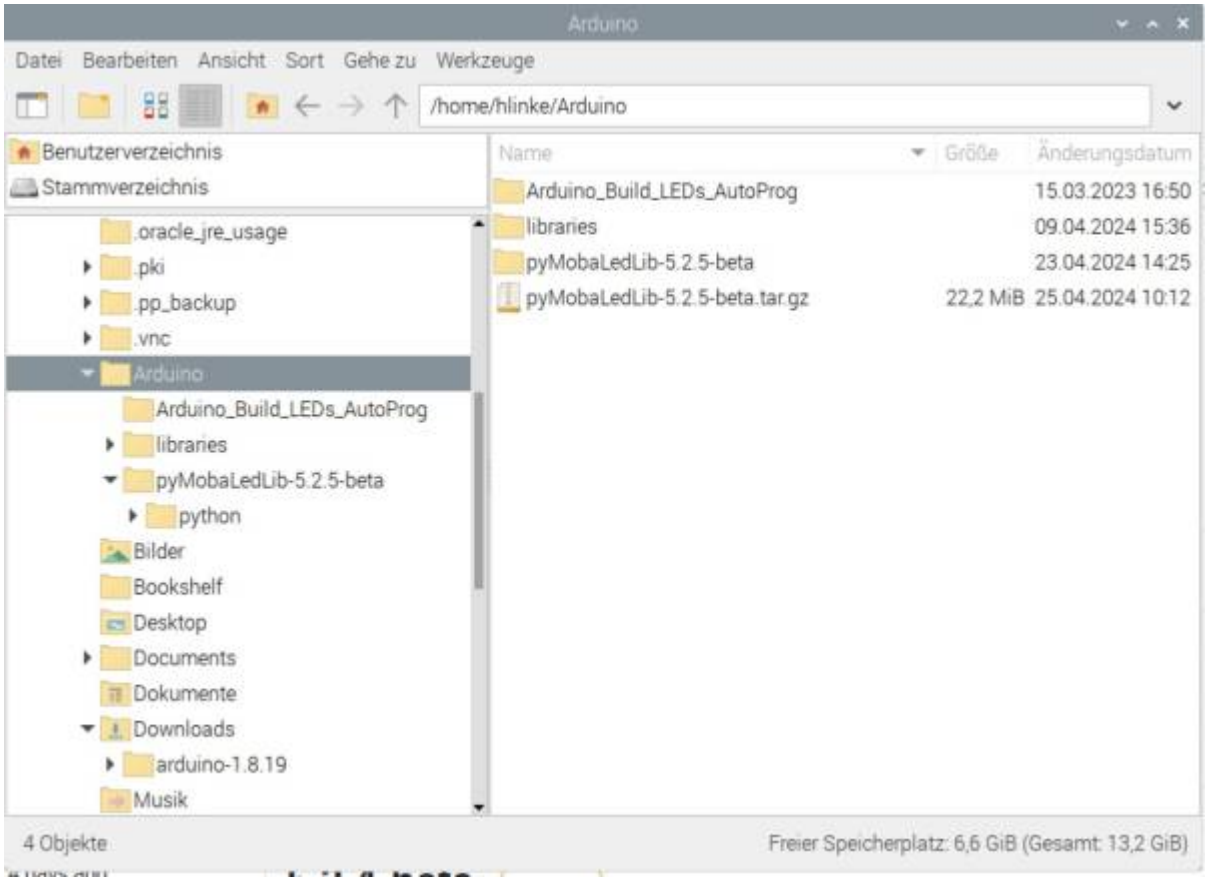
Für Linux benötigen wir die Souce code Datei vom Typ **tar.gz**

Für den Download auf den Link klicken.

Die Datei wird ins Downloadsverzeichnis heruntergeladen.



Von dort die Datei in das ARDUINO-Daten Verzeichnis kopieren. Bei einer Standard-Installation der ARDUINO SW sollte dies das Verzeichnis: home/<user>/Arduino sein. Siehe Bild:



Die ZIP-Datei „hier entpacken“, so daß sich das Verzeichnis pyMobaLedLib-xxxx in dem ARDUINO-Verzeichnis befindet. Siehe Bild oben.

Dieses Verzeichnis kann in „pyMobaLedLib“ umbenannt werden. Muß aber nicht, der Name ist beliebig

Die pyMobaledLib kann dann mit dem Kommando: Python3 /home/<user>/Arduino/pyMobaLedLib/python/pyMobaLedLib.py gestartet werden.

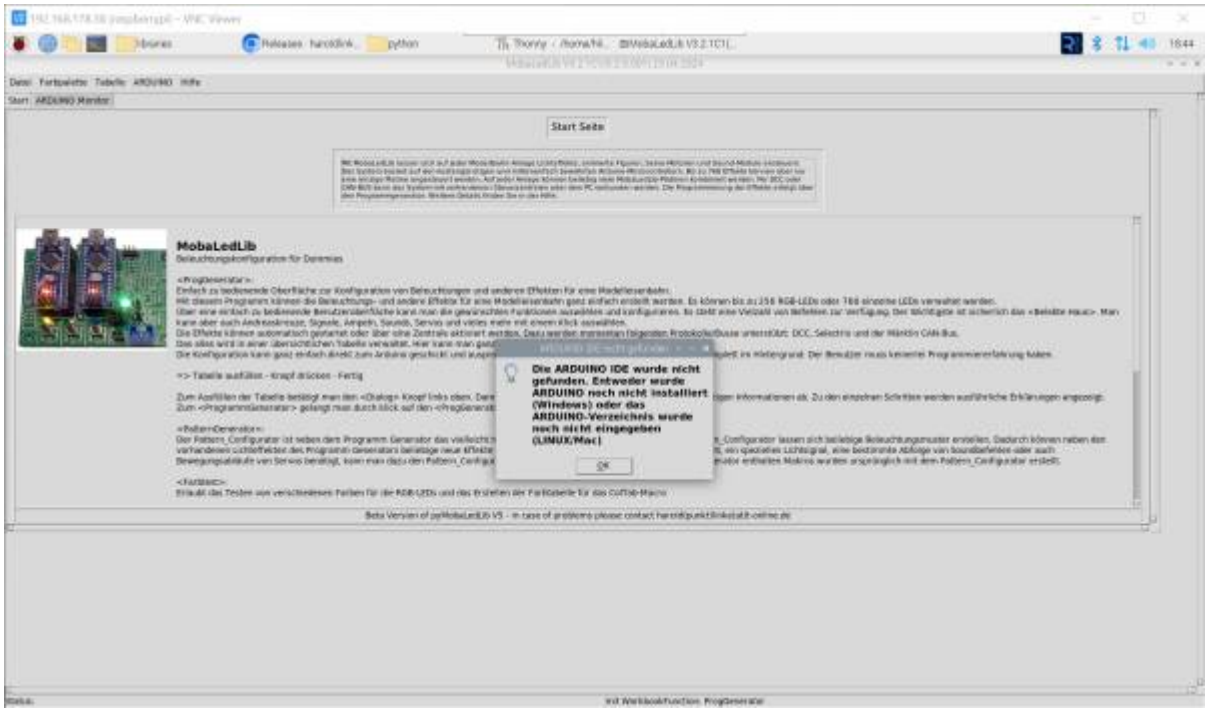
Desktop Icon erstellen

Da dieser Aufruf doch etwas kompliziert ist, kann man auf dem Desktop ein Icon installieren.

— noch in Arbeit —

Der erste Aufruf von pyMobaLedLib

Beim ersten Aufruf meldet sich das Programm mit einer fehlermeldung: Die ARDUINO Bilbilothek wurde nicht gefunden.



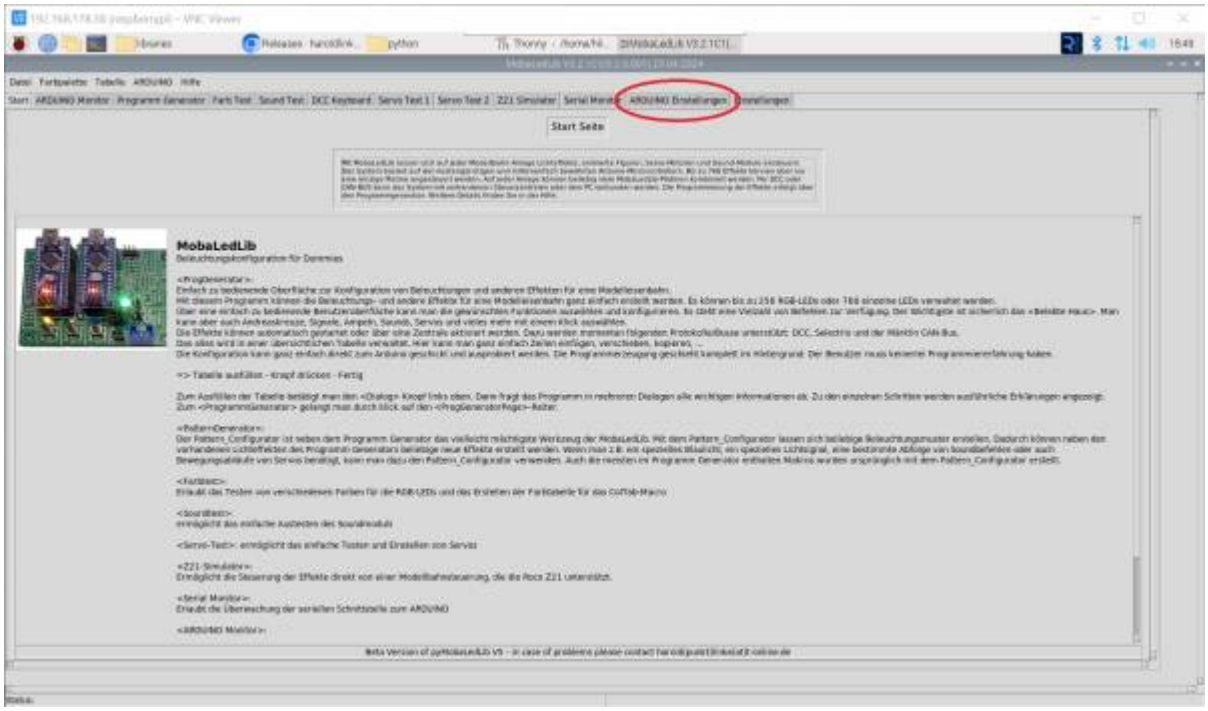
Das ist bei Linux normal, da es leider keine automatische Möglichkeit gibt, den Installationsort der ARDUINO IDE zu bestimmen. Diese Information müssen wir der pyMobaledLib von Hand mitteilen.

Zuerst diese Meldung mit „OK“ bestätigen.

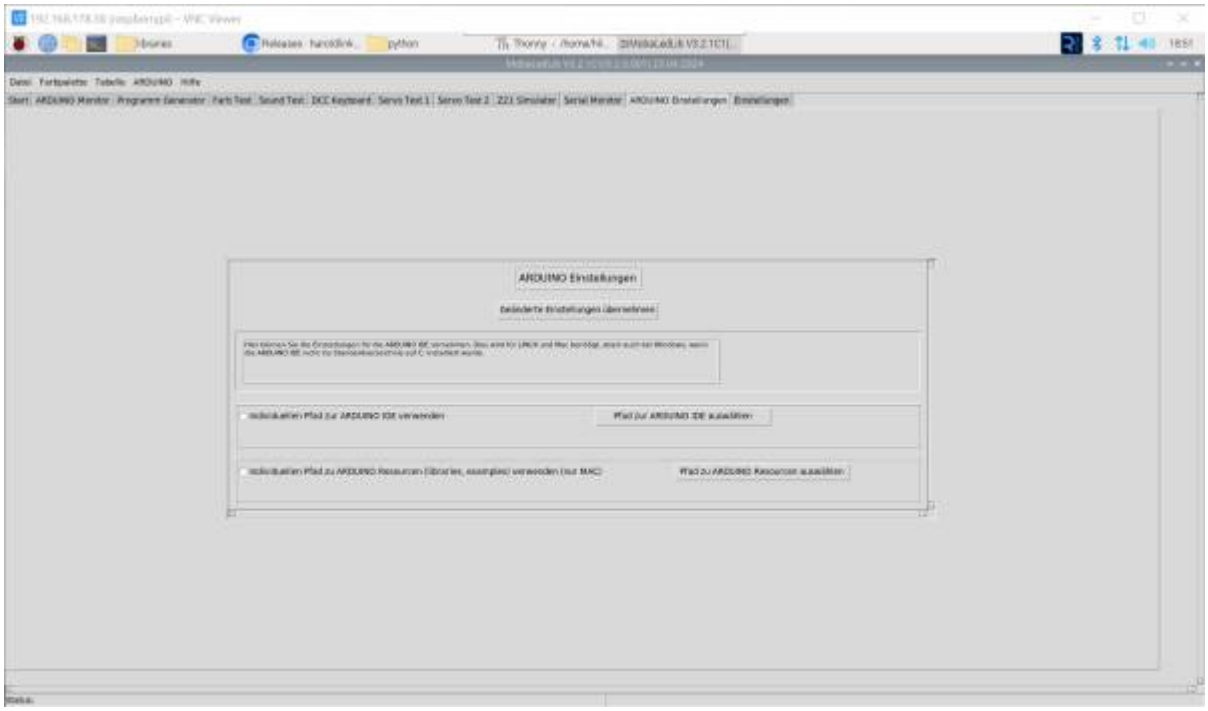
Dannach kommt eine weitere Meldung, dass das Programm eine Verbindung zum ARDUINO aufbauen möchte.

Zu dieser Meldung sagen wir „Nein“.

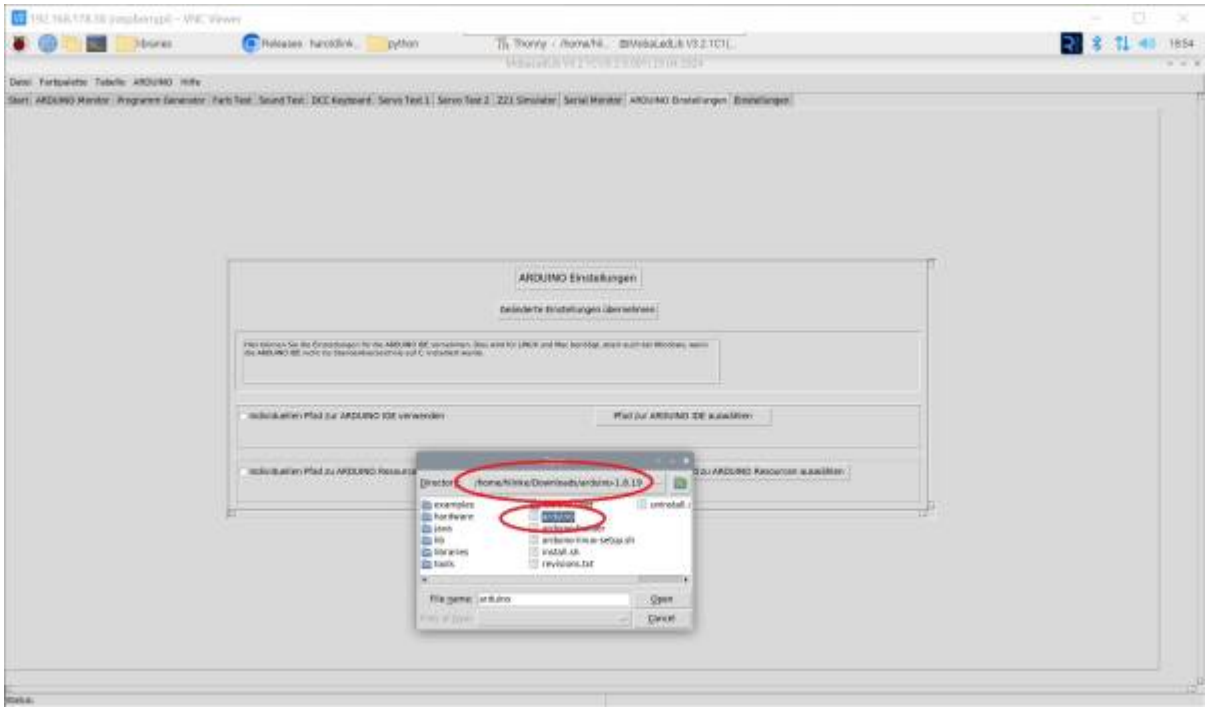
Danach sehen wir die Startseite der pyMobaLedLib.



Die Seite mit den **ARDUINO Einstellungen** öffnen.

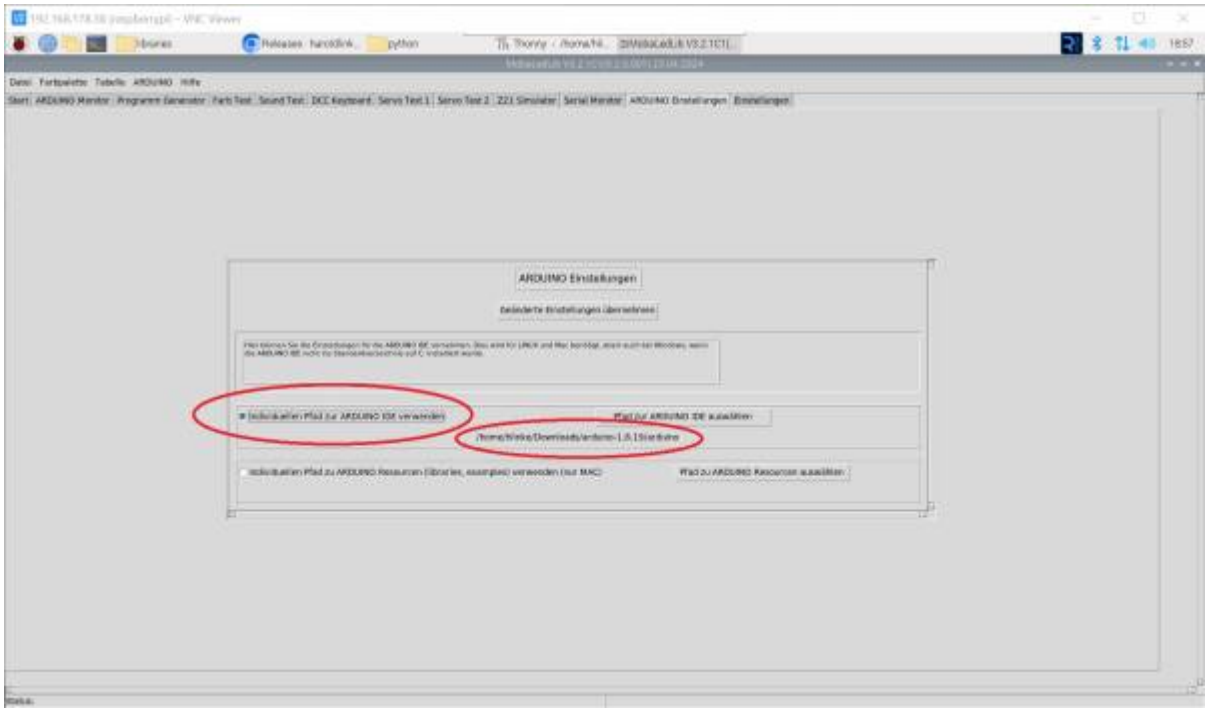


Auf den Button „**Pfad zur ARDUINO IDE auswählen**“ klicken.



Die ARDUINO IDE befindet sich in dem Ordner, in dem wir vorher die ARDUINO IDE-Datei entpackt haben. In unserem Fall /home/<user>/Downloads/arduino-1.8.19

Nicht vergessen, den Haken bei „**Individuellen Pfad zur ARDUINO IDE verwenden**“ zu setzen.

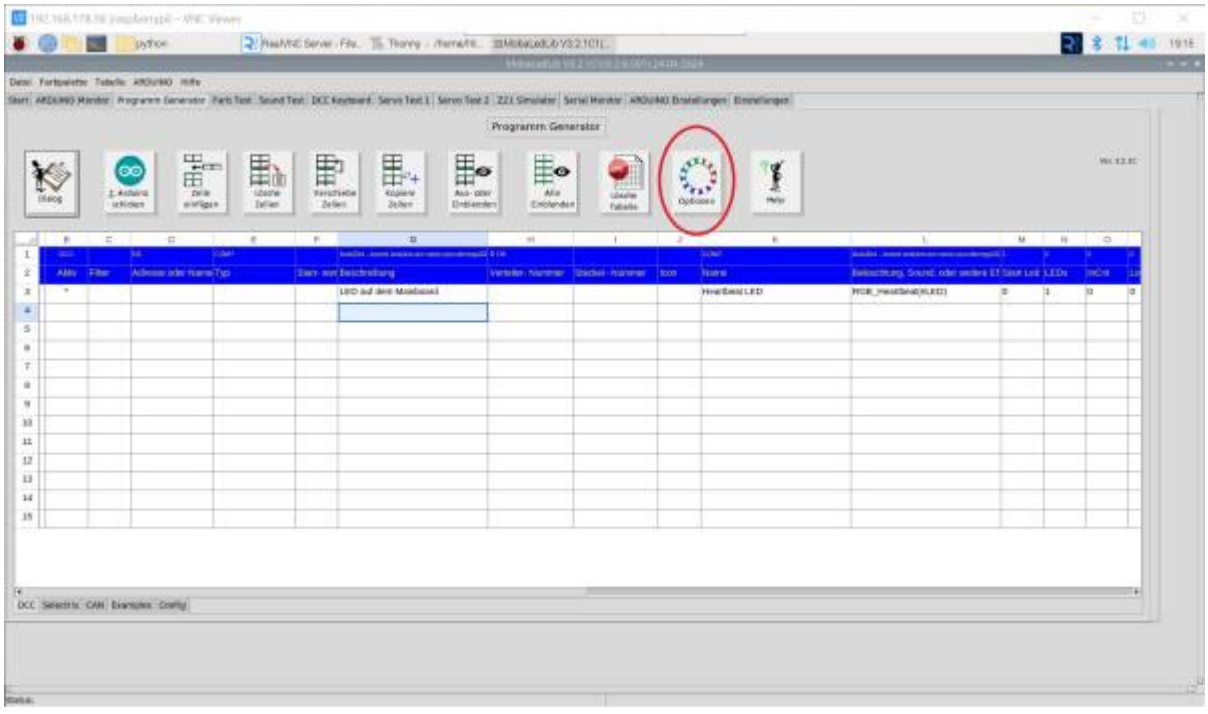


Am Besten die pyMobaLedLib jetzt neu starten.

Nach dem Start kannes sein, daß das Programm weitere Bibliotheken installieren möchte. Diese Frage mit „Ja“ beantworten.

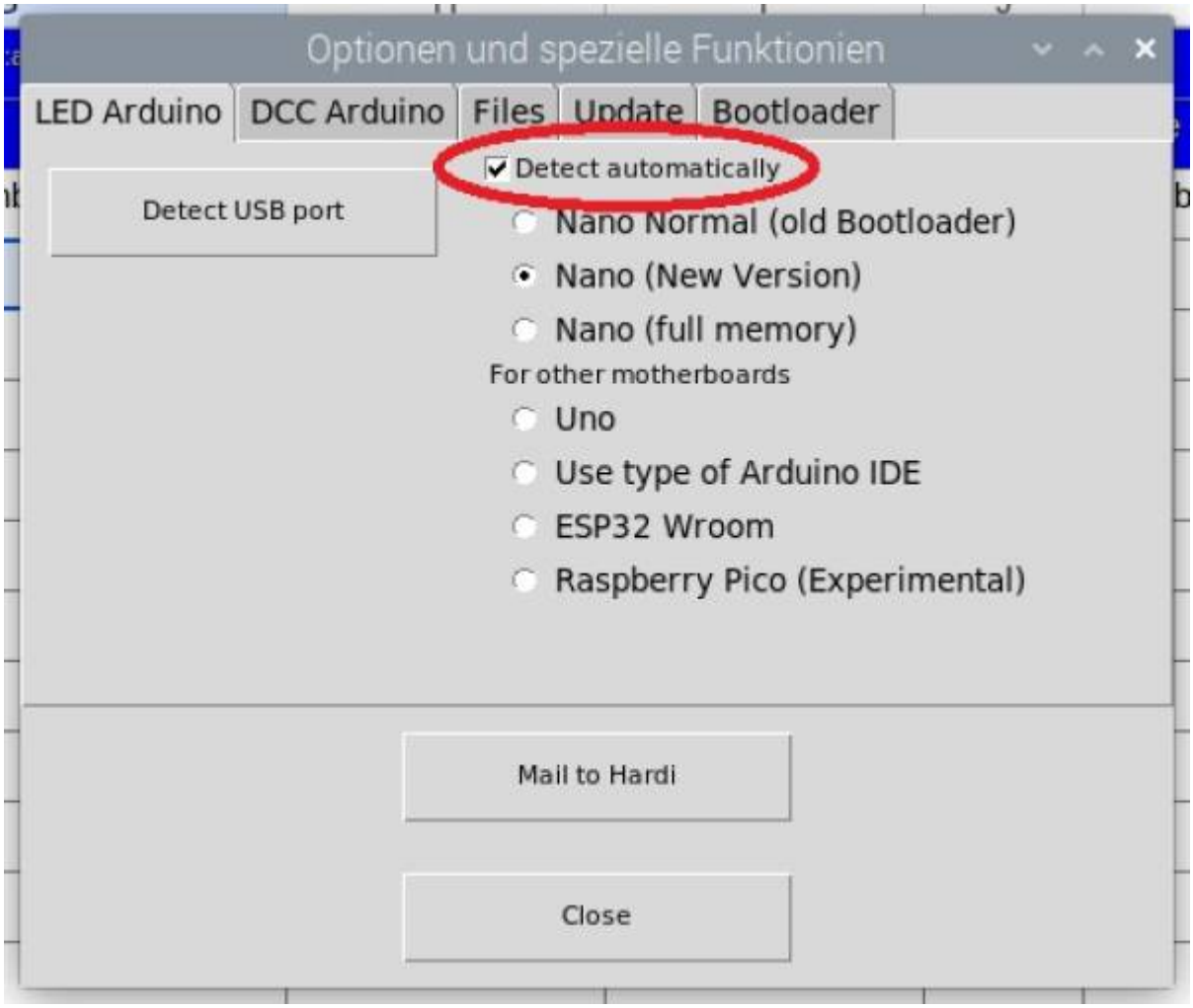
Die Frage nach der Verbindung mit dem ARDUINO sollte wieder mit „Nein“ beantwortet werden, da das Programm unter Linux den Boardtyp nicht automatisch erkennen kann und dieser sehr wahrscheinlich falsch eingestellt ist. Wir müssen den Boardtyp von Hand einstellen.

Dazu den Programm Generator öffnen:



Auf den Button „Optionen“ klicken.

Es öffnet sich das Fenster mit dem man den Boardtyp festlegen kann.



Den Haken bei „Automatisch erkennen“ löschen und den Boardtyp einstellen. Beim nano gibt es verschiedene Typen:

- neue offizielle Nanos haben meist den Typ Nano (New Version)
- China Nachbauten den Typ Nano Normal (old Bootloader)

Sie können die verschiedenen Typen auch durch probieren und sehen ob es klappt.

Nach der Einstellung des Boardtyps, den Button „**Detect USB Port**“ anklicken:

Es erscheint der USB-Port Auswahl Dialog

Hier den Port auswählen, bei dem der Nano erkannt wird entweder mit einem Namen oder als USB Serial:

Mit OK bestätigen.

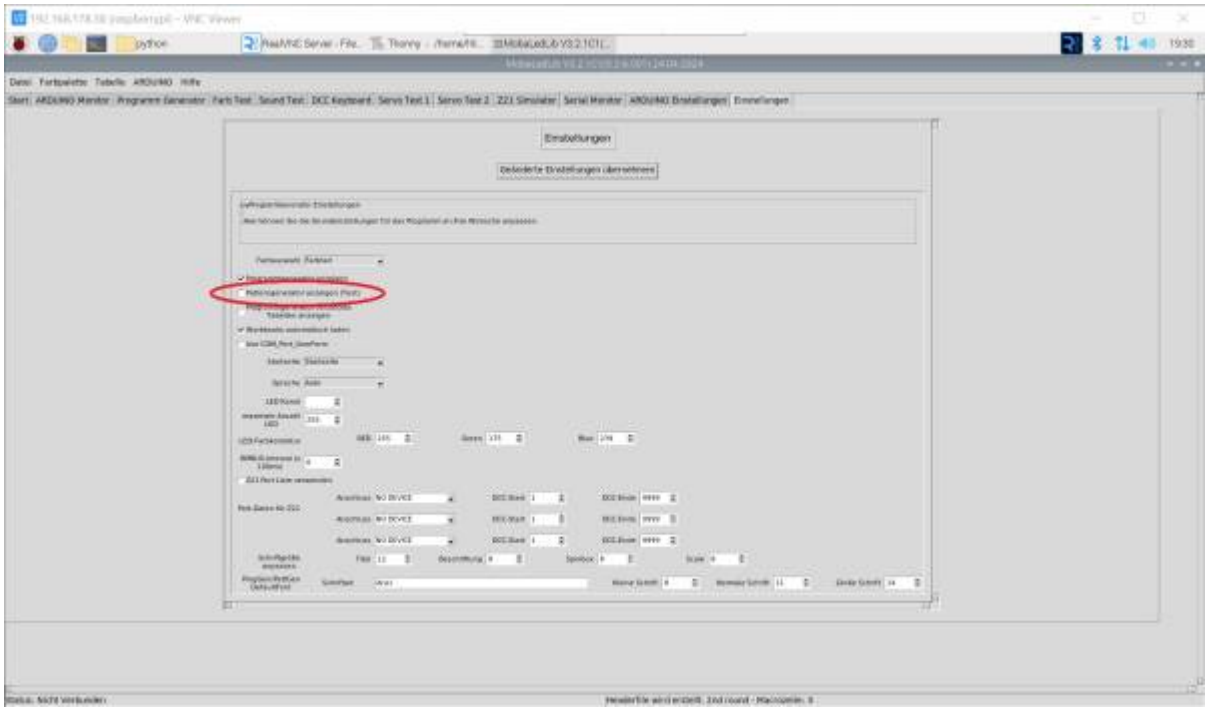
Es kommt wahrscheinlich die Meldung, dass kein ARDUINO erkannt wurde. Da die automatische Erkennung bei Linux meistens nicht funktioniert, diese Meldung mit „Nein“ ignorieren.

Dieselbe Meldung kommt auch beim Hocvhladen eines Programms zum ARDUINO. Diese dann auch ignorieren.

Sollte das Hochladen nicht klappen, liegt es meistens daran, dass der falsche Typ eingestellt wurde. Die obige Prozedur dann bitte wiederholen und den richtigen Typ einstellen.

Pattern Configurator aktivieren

Wenn die **Pattern Configurator** Seite nicht angezeigt wird, kann sie einfach durch eine Einstellung auf der „**Einstellungen**“ Seite aktiviert werden.



Bei der Option „Patterngenerator anzeigen muß ein Haken sein, damit die Pattern Configurator Seite angezeigt wird. Nach dem Setzen des Hakes, die Einstellungen speichern und pyMobaledLib neu starten.

Desktop Icon

Wenn Sie ein Desktop Icon zum Aufruf der pyMobaledLib haben möchten, müssen Sie folgendermaßen vorgehen:

Erstellen Sie eine Textdatei mit folgendem Inhalt: <webcode name=„Default“ frameborder=0 width=100% scrolling=yes externalResources=“,“ renderingMode=story >

[Desktop Entry]

Type=Application

Name=pyMLL

GenericName=pyMll

Comment=Open-source electronics prototyping platform

Exec=python3 /home/<user>/Arduino/pyMobaLedLib/python/pyMobaLedLib.py

Icon=/home/<user>/Arduino/pyMobaLedLib/python/images/06_Michael_MLL.png

Terminal=false

Categories=Development;IDE;Electronics;

MimeType=text/x-arduino;

Keywords=embedded electronics;electronics;avr;microcontroller;

StartupWMClass=processing-app-Base

Name[en_US]=pyMll

Path=home/<user>/Arduino/pyMobaLedLib/python/

<user> muß durch Deinen usernamen ersetzt werden.

Diese Textdatei unter einem beliebigen Namen mit der Endung .desktop speichern Die Datei auf den Desktop ziehen.

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

Permanent link:
https://wiki.mobaledlib.de/anleitungen/spezial/pyprogramgenerator/raspberryinstallation?rev=1714154007

Last update: **2024/04/26 17:53**