

## # Junsi iCharger 4010Duo - Linux Setup Guide (Debian/LMDE)

Diese Anleitung beschreibt die Einrichtung des Systems für den Zugriff auf Junsi iCharger Hardware via USB. Aufgrund der Composite-Bauweise (HID & Massenspeicher) wird der Zugriff über **PyUSB** anstelle von hidraw/hidapi durchgeführt.

### ## 1. Systemvoraussetzungen (Root)

Zuerst müssen die notwendigen System-Bibliotheken für die USB-Kommunikation installiert werden:

```
sudo apt update
sudo apt install libusb-1.0-0 libusb-1.0-0-dev python3-venv
```

### ## 2. udev-Regeln konfigurieren

Die udev-Regel erfüllt zwei Zwecke:

- Gewährung von Schreibrechten für den User (via Gruppe plugdev).
- Zuweisung des uaccess-Tags für die aktuelle Session.

Erstellen Sie die Datei `/etc/udev/rules.d/99-junsi.rules`:

```
# Junsi iCharger 4010Duo - VID:0483 PID:5751
# Erlaubt Zugriff auf USB-Ebene und delegiert Rechte an den User
SUBSYSTEM=="usb", ATTR{idVendor}=="0483", ATTR{idProduct}=="5751",
MODE="0666", GROUP="plugdev", TAG+="uaccess"
```

Aktivieren Sie die Regeln ohne Neustart:

```
sudo udevadm control --reload-rules
sudo udevadm trigger
```

**Hinweis:** Das Gerät danach einmal physisch aus- und wieder einstecken.

### ## 3. Diagnose-Checkliste

Führen Sie diese Befehle aus, um den Status der Verbindung zu prüfen:

Prüfung	Befehl	Erwartetes Ergebnis
<b>Kernel-Treiber</b>	<code>lsusb -t</code>	Port für lf 0 sollte <code>Driver=[none]</code> oder <code>Driver=usbhid</code> zeigen.
<b>User-Berechtigung</b>	<code>ls -l /dev/bus/usb/xxx/yyy</code>	<code>crw-rw-rw-</code> oder <code>crw-rw---+</code> (Gruppe <code>plugdev</code> ).
<b>ACL Status</b>	<code>getfacl /dev/bus/usb/xxx/yyy</code>	Zeile <code>user:DEINNAME:rw-</code> muss vorhanden sein.

### ## 4. Python Framework Setup

Innerhalb der virtuellen Umgebung (venv) des Projekts muss PyUSB installiert sein:

```
pip install pyusb
```

Der Transport in `hardware/transport/hid_modbus.py` muss zwingend `detach_kernel_driver` nutzen, um Interface 0 (HID) exklusiv zu beanspruchen.

## ## 5. Bekannte Probleme

- **Massenspeicher-Blockade:** Interface 1 (SD-Karte) wird oft vom Kernel als Laufwerk gebunden. Der PyUSB-Transport löst dies automatisch beim Verbindungsaufbau.
- **SyntaxError bei Import:** Achten Sie darauf, dass keine Citations oder Metadaten-Tags im Python-Code von `hid_modbus.py` enthalten sind.

From:  
<https://www.drklipper.de/> - **Dr. Klipper Wiki**

Permanent link:  
[https://www.drklipper.de/doku.php?id=ionpy:hardware:junsi\\_icharger&rev=1769457653](https://www.drklipper.de/doku.php?id=ionpy:hardware:junsi_icharger&rev=1769457653)

Last update: **2026/01/26 21:00**

