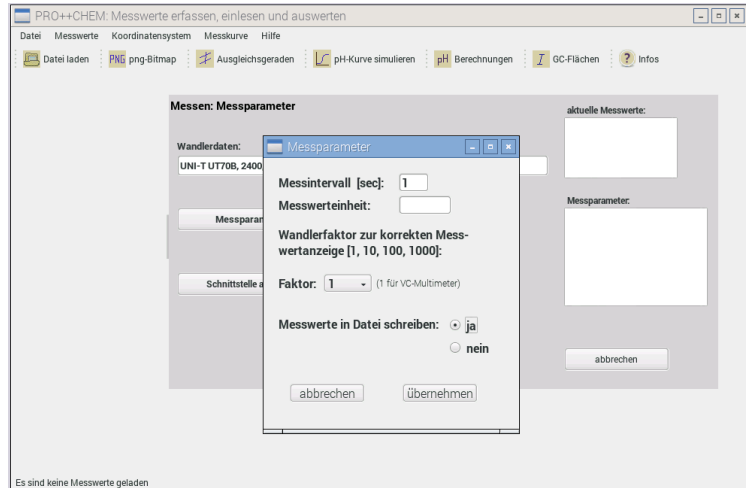


Raspberry Pi und pro++chem

Für den **Raspberry Pi** existiert eine [pro++chem-Version](#), die unter **Raspbian** auf einem Raspberry Pi 2 B getestet wurde. Diese Programmversion ist auch auf einem **pcDuinoV3** mit **Ubuntu 14.04** oder **Armbian Jessie** (Debian 8) als Betriebssystem lauffähig. Zur Aufzeichnung von Messwerten zumeist ein **USB-seriell-Adapter** erforderlich (→ abhängig vom Schnittstellenkabel des jeweiligen Messwandlers/Multimeters).

Messungen mit **pro++chem** lassen sich problemlos mit dem **Raspberry Pi 2 B** (1 GB Arbeitsspeicher, 900-MHz-ARMv7-Quad-Core-Prozessor) aufnehmen, ausgenommen 2-Kanalmessungen, für die ein Raspberry Pi 3 verwendet werden sollte.

Für *Messaufnahmen mit älteren Modellen* wie dem **Raspberry Pi Model B** und **Model B+** (512 MB Arbeitsspeicher, 700-MHz-Prozessor) existiert eine spezielle pro++chem-Version, die Messungen über einen Zeitraum von ca. 800 sec erlaubt (mit einer Messung/sec): [pro++chem_raspi2SD.zip](#) (in 2 Versionen, erstellt mit Qt4.8 bzw. Qt5). Mit einem **Raspberry Pi Zero W** können Messungen bis zu einer Dauer von ca. 1200 Sekunden aufgenommen werden.



Im Eingabebereich der Messparameter gibt es die Möglichkeit (s. Abbildung rechts), die Messwerte vom Programm in eine Datei (promess.p+c, im Programmverzeichnis) auf SD-Karte schreiben zu lassen. Während der Messung werden die Werte in Großanzeige dargestellt. Unmittelbar nach Beendigung der Messung werden die Daten aus der Datei eingelesen und in Form einer Messkurve präsentiert. Die Funktionalität dieser Version von pro++chem ist, abgesehen von der zusätzlichen Möglichkeit einer Messwertaufnahme, identisch mit der der anderen Programmversionen. Diese pro++chem-Variante ist **auf allen Modellen des Raspberry Pi ausführbar**.

Ausführung von pro++chem auf einem Raspberry Pi ohne eigenen Monitor und ohne eigene Tastatur, mit und ohne WLAN und DHCP-Server (sogeannter Headless-Betrieb)

Sinnvoll erscheint die im Folgenden beschriebene Vorgehensweise, wenn man **Messungen mit pro++chem auf einem Raspberry Pi** vornehmen möchte. Eine direkte ssh-Verbindung zwischen dem Raspi und einem Laptop ist dafür erforderlich.

Voraussetzungen:

- Das Betriebssystem (z.B. Raspbian) ist auf der SD-Karte des Raspberry Pi installiert ([Installation s. unten](#)).
- Der Benutzer 'pi' (oder ein vergleichbarer) existiert.
- Der Raspberry Pi muss einen ssh-Zugang (ssh, secure shell) ermöglichen, ein **ssh-Server muss also aktiv sein**.
entweder: Ob der ssh-Server läuft, kann durch Eingabe - in der Konsole/Terminal des Raspi - von '**service ssh status**' überprüft werden. Falls nötig, kann der Server durch das Kommando '**sudo service ssh start**' aktiviert werden. Eine anschließende Eingabe von '**sudo insserv ssh**' bewirkt, dass der ssh-Server bei jedem Neustart des Raspberry Pi automatisch gestartet wird.
oder: Bei Verwendung von Raspbian kann man SSH (den ssh-Server) auch wie folgt aktivieren: Nach der Übertragung der Raspbian-Image-Datei auf die SD-Karte öffnet man dort die boot-Partition und erstellt eine leere Datei namens 'ssh' (in einer Konsole mit der Anweisung '**touch ssh**' oder mit Hilfe des Dateimanagers). Beim Booten des Raspi von dieser SD-Karte wird der ssh-Dienst automatisch gestartet. [s. Raspberry Pi Geek, Heft 03-04/2017]
- Für eine ssh-Verbindung muss einem die IP-Adresse des Raspberry Pi bekannt sein (IP-Ermittlung s. unten: vom [LAN zugewiesene IP](#) oder [statische IP](#)).

- pro++chem befindet sich auf der SD-Karte des Raspberry Pi.

Einrichtung einer direkten **ssh-Verbindung** zwischen einem Laptop/PC und einem **Raspberry Pi Zero W** mit Hilfe von **WLAN (Headless-Betrieb)**

Der kleinste Vertreter der Pi-Familie, der **Pi Zero W**, ist auf seinem Board mit einem WLAN-Modul ausgestattet und stellt im aktiven Zustand, die ssh-Verbindung zu einem Zweitrechner her. Bei der hier beschriebenen Vorgehensweise wird davon ausgegangen, dass es sich bei dem Zweitrechner um einen Rechner mit einem Linuxsystem handelt (z.B. Debian, Ubuntu usw.). Mögliche Vorgehensweise zur Herstellung der Verbindung mit vorausgehender **Installation von Raspbian** auf einer SD-Karte:

- Aus dem Internet (<https://www.raspberrypi.org/downloads/raspbian/>) lädt man die **Lite-Version von Raspbian** (aktuell Debian 9, stretch, **ohne grafische Oberfläche**) auf den Zweitrechner herunter und entpackt sie in ein separates Verzeichnis (beliebiger Name).
- die Image-Datei schreibt man mit folgender Konsolen-Befehlskette auf eine SD-Karte (Größe: 4 - 16 GB, Class 10):
'sudo dd if=<Name-des-lite-Image>.img of=/dev/sd? bs=4M; sync'

Ganz wichtig: Es ist die exakte Laufwerksangabe für /dev/sd? herauszufinden. Das kann mit dem Befehl **'sudo fdisk -l'** geschehen, mit am Linux-Rechner angeschlossener SD-Karte.

Lautet eine Ausgabe z.B. **'Disk /dev/sda: 465,8 GiB, 500107862016 bytes, 976773168 sectors'** und **'Disk /dev/sdb: 29 GiB, 31104958464 bytes, 60751872 sectors'** (neben anderen Ausgaben), dann weiß man, dass die SD-Karte dem LW '/dev/sdb' entspricht. Die eingebaute Festplatte entspräche der LW-Angabe '/dev/sda'. Die obige Befehlseingabe müsste also 'sudo dd if=<Name-des-lite-Image>.img of=/dev/sdb bs=4M; sync' lauten. Würde ein anderes, bestehendes Laufwerk im Parameter of=/dev/sd? angegeben werden, so wären dessen Daten unwiederbringlich verloren! Das Image wird nämlich direkt auf das angegebene LW geschrieben!

- SD-Karte dem Rechner entnehmen und erneut einstecken: die SD-Karte enthält 2 Partitionen, die beide in den Verzeichnisbaum des Rechners eingebunden (gemountet) werden. In der unter 'boot' eingebundenen Partition erstellt man eine Datei mit der Bezeichnung 'wpa_supplicant.conf' und schreibt folgenden *Inhalt* zeilenweise hinein:

```
country=DE
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
  ssid="Name-des-WLAN"
  psk="WLAN-Passwort"
}
```

Anm.: In Zeile 5 und Zeile 6 müssen der Name des WLAN-Netzes sowie das WLAN-Passwort den Gegebenheiten angepasst werden.

Beim Booten des Pi Zero W wird der so erstellte Inhalt der Datei nach '/etc/wpa_supplicant' ins Dateisystem der SD-Karte kopiert und der Pi Zero meldet sich automatisch im eingetragenen WLAN an. [s. Raspberry Pi Geek, Heft 09-10/2017, S. 20-24]

- Als nächstes erstellt man, wie oben beschrieben und ebenfalls in der boot-Partition, eine **leere Datei** mit der Bezeichnung **ssh** (auf der Konsole durch Eingabe von **'touch ssh'**).
- Nach dem sicheren Entfernen der SD-Karte wird der Pi Zero W damit zum ersten Mal gebootet. Durch die vorgenommenen Maßnahmen startet der ssh-Server des Kleinrechners automatisch und meldet sich auch selbsttätig im bestehenden WLAN an, erhält also eine IP-Adresse.
Die IP des Pi Zero W gilt es nun zu ermitteln. Hilfreich dafür kann der folgende Konsolen-Befehl (auf dem "großen" Linux-Rechner) sein:
'sudo nmap -sP 192.168.50.*', wobei das 3. Ziffernbündel auch anders als '50' lauten kann.
Die genaue "Zahlenfolge" findet man heraus, indem man am eigenen Linux-Rechner die

sogenannte Broadcast-Adresse herausfindet.

Das gelingt durch Eingabe von **'sudo ifconfig'**. In der Ausgabe sucht man die WLAN-Schnittstelle (oft 'wlan0') und dort den Eintrag **'broadcast xxx.xxx.xxx.255'** auf. Im obigen Befehl gibt man dann die broadcast-Adresse ein, wobei die '255' durch '*' zu ersetzen ist.

In der anschließenden Ausgabe kann man die IP-Adresse des Pi Zero W finden. Der 'Nmap scan report' gibt die IP-Adressen der im WLAN-Netz befindlichen Rechner aus, sowie deren 'MAC Address' (= Adressen der WLAN-Schnittstellen).

Einen Raspberry Pi erkennt man daran, dass hinter der Ausgabe seiner Mac-Adresse der Zusatz '(Raspberry Pi Foundation)' erscheint:

```
Nmap scan report for 192.168.50.77
```

```
Host is up (0.033s latency).
```

```
MAC Address: B8:27:EB:21:F6:DD (Raspberry Pi Foundation)
```

Der Raspberry Pi hat in diesem nmap-Ausgabebeispiel die IP-Adresse 192.168.50.77.

- Mit Kenntnis der IP des **Pi Zero W** stellt man die direkte Verbindung zwischen einem Laptop/PC und einem Pi Zero W durch Eingabe des Befehls **'ssh pi@<IP-des-Pi-Zero-W>'** (z.B. mit <IP-des-Pi-Zero-W>=192.168.50.77) und des Passwortes von 'pi' auf einer Konsole des Laptop/PC her.

Anmerkung zur Passworteingabe: Während der Eingabe des Passwortes werden **keinerlei Zeichen angezeigt** (auch keine Sternchen oder Punkte)!

Nun kann man vom Laptop aus (mit Hilfe dessen Tastatur und Display) auf dem Pi Zero W arbeiten. Z.B. das Passwort des users 'pi' ändern, das System auf den neuesten Stand bringen (**sudo apt-get update** mit anschließendem **sudo apt-get upgrade**) und 'qt5-default' installieren, mit **'sudo apt-get install qt5-default'**.

- Installation von pro++chem auf dem Pi Zero W:
 - a) Ein Verzeichnis für pro++chem mit dem Befehl **'mkdir pro++chem'** erstellen und mit der Anweisung **'cd pro++chem'** in das Verzeichnis wechseln. Der Verzeichnisname ist frei wählbar.
 - b) Die gewünschte Version von pro++chem (z.B. pro++chem_raspi2SD.zip) mit folgender Anweisung von gprot.de herunterladen:
'wget http://gprot.de/index.html/download/qt5/noscreenanpassung/pro++chem_raspi2SD.zip'.
Die zip-Datei befindet sich nun im Verzeichnis 'pro++chem' und kann dort mit dem Befehl **'unzip pro++chem_raspi2SD.zip'** entpackt werden. Mit der Anweisung **'ls'** oder **'ls -l'** kann man sich den Verzeichnisinhalt anzeigen lassen. Enthalten ist u.a. das ausführbare Programm 'pro++chem'. Um das Programm auszuführen meldet man sich mit **'exit'** vom Pi Zero W ab, beendet also die ssh-Verbindung.
- Man öffnet eine neue ssh-Verbindung mit **'ssh -X pi@<IP-des-Pi-Zero-W>'** zum Pi Zero W. Wichtig ist der **Parameter '-X'**, der die Anzeige der grafischen Programmoberfläche von pro++chem erst ermöglicht.
Mit **'cd pro++chem'** wechselt man in das pro++chem-Verzeichnis und **führt** mit der Eingabe von **'./pro++chem'** und Betätigung der **Eingabetaste das Programm aus**. Es erscheint die Programmoberfläche von pro++chem auf dem Laptop/PC.
Sollte nach der Eingabe von **'./pro++chem'** die **Fehlermeldung './pro++chem: Keine Berechtigung'** erscheinen, dann muss das Programm mit folgender Anweisung ausführbar gemacht werden: **'sudo chmod +x pro++chem'**. Ein Programmstart sollte dann möglich sein.
- Nach Beendigung der Arbeiten auf dem **Pi Zero W** sollte man diesen mit der Anweisung **'sudo init 0'** bzw. **'sudo poweroff'** sicher herunterfahren.

Einrichtung einer direkten *ssh-Verbindung* zwischen einem Laptop/PC und einem Raspberry Pi mit Hilfe eines *Netzwerkkabels* (Headless-Betrieb)

A) Laptop/PC mit (Debian-) Linux als Betriebssystem

Die Netzwerkschnittstellen des Raspberry Pi und des Laptops/PC (*hier mit Debian-Linux als Betriebssystem*) müssen jeweils eine **feste IP-Adresse** aus demselben Adressbereich

besitzen.

Anmerkung: Das Debian-Betriebssystem befindet sich auf der SD-Karte des Raspberry Pi ([Raspbian-Installation s. oben](#))!

Um die notwendigen Veränderungen an der Netzwerkkonfiguration des Raspberry Pi (Raspi) vorzunehmen, öffnet man dessen SD-Karte über einen Kartenleser/Kartenslot am Laptop/PC und verfährt wie folgt beschrieben:

- Man öffnet eine **Konsole/Terminal** und wechselt (auf der root-Partition der SD-Karte) in das Unterverzeichnis **'network'** von Verzeichnis **'/etc'**. Der Befehl lautet **'cd /etc/network'**.
- Im Fall einer **Raspbian-Version auf Basis von Jessie (Debian 8)**: Die Datei **'interfaces'** enthält die eigentlichen Netzwerkeinstellungen und muss nun **bearbeitet** werden (als Systemverwalter). Mit dem Editor **'nano'** wird die Datei durch Eingabe von **'sudo nano interfaces'** **geöffnet** und durch das Einfügen der folgenden 4 Anweisungen erweitert:

```
allow-hotplug eth0
iface eth0 inet static
address 10.0.2.15
netmask 255.255.255.0
```

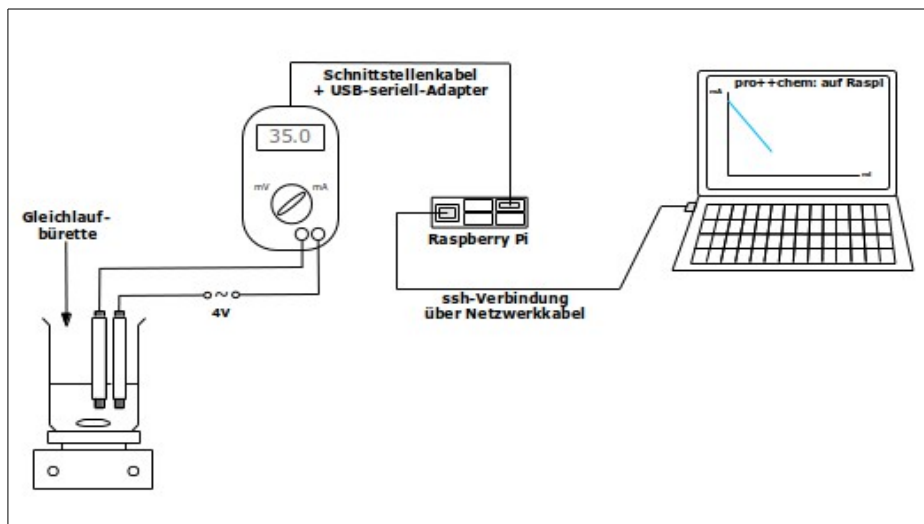
Man speichert die Neueingaben der Datei **'interfaces'** durch die Tastenkombination **'Strg+O'** und Drücken der **Eingabetaste**. Mit **'Strg+X'** verlässt man den Editor **'nano'**. Als Ergebnis erhält der Raspi (dessen Netzwerkschnittstelle **'eth0'**) nach einem Neustart die feste/statische IP-Adresse 10.0.2.15.

Im Fall einer **Raspbian-Version auf Basis von Stretch (Debian 9)**: Man öffnet mit **'nano'** im Verzeichnis **'/etc'** die Datei **'dhcpcd.conf'** mit der Anweisung **'sudo nano /etc/dhcpcd.conf'** und fügt am Ende der Datei zwei Zeilen ein:

```
interface eth0
static ip_address=10.0.2.15/24
```

Die Änderungen speichert man wie oben mit **'Strg+O'** + **Eingabetaste**. Mit **'Strg+X'** verlässt man den Editor. Nach einem Neustart des Raspberry Pi erhält dessen Netzwerkschnittstelle **'eth0'** die statische IP-Adresse 10.0.2.15.

- Der **Netzwerkschnittstelle eth0** des **Laptops/PC**, der die Verbindung zum Raspi herstellen soll, muss nun ebenfalls eine **feste IP-Adresse** aus demselben Adressbereich wie dem Raspi zugeordnet werden, z.B. **10.0.2.10** oder **10.0.2.20** (oder 10.0.2.x mit x = 1 - 24 und 26 - 254).
- **Laptop/PC** und **Raspi** hochfahren und beide über ein **Netzwerkkabel** direkt **miteinander verbinden**.
- Auf dem **Laptop/PC** ein **Terminal** öffnen und durch Eingabe von **'ssh -X pi(at)10.0.2.15'** ('(at)' = @) eine **sichere Verbindung zum Raspi** aufbauen. Nach Eingabe des Passwortes des Users **'pi'** besteht eine direkte Verbindung vom Laptop/PC zum Raspi und man kann pro++chem starten, wie oben beschrieben.
- Falls keine Verbindung zum Internet besteht, ist es sinnvoll, **Datum und Systemzeit** des Raspi dem realen Datum und der Uhrzeit anzupassen. **Startwerte** für beide kann man mit der Anweisung **sudo date --set="2018-05-23 13:30:00"** setzen. Festgelegt wären damit das Datum auf den 23. Mai 2018 und die Zeit auf 13:30 Uhr. Nach ihrer Eingabe werden die Startwerte vom System fortlaufend aktualisiert. Durch Eingabe von **'date'** kann die Systemzeit abgefragt werden.



Die **Abbildung** stellt die Situation dar, in der nach vorangegangener Netzwerkkonfiguration (s. vorherigen Abschnitt) am Laptop über das Netzwerkkabel eine physikalische Verbindung zum Raspberry Pi hergestellt wurde.

In einer Konsole erfolgt durch Eingabe von **ssh -X pi@<IP-des-Raspi>** die Anmeldung am Raspberry Pi. **Wichtig** ist der **Parameter -X**, denn er ermöglicht erst die Übertragung grafischer Elemente. Nach einem Wechsel in das Verzeichnis, in dem sich pro++chem befindet (z.B. mit `cd pro++chem`), wird das Programm mit der Eingabe von **./pro++chem** auf dem Raspi gestartet.

Nach kurzer Zeit erscheint die Programmoberfläche auf dem Laptopbildschirm; nach Auswahl des Messgerätes bzw. des AD-Wandlers und Festlegung der Messparameter kann die Messwertaufnahme erfolgen. Die Messwerte werden vom Messgerät über den USB-seriell-Adapter an einer der USB-Schnittstellen des Raspi empfangen und von pro++chem zu einer Messkurve (o. Großanzeige der Werte) verarbeitet. Die gesamte Messwertaufzeichnung kann am Laptop verfolgt, gespeichert und ausgewertet werden. Nach erfolgter Arbeit beendet man pro++chem (Strg+Q) und fährt den Raspberry Pi - vom Laptop aus - durch Eingabe von **sudo init 0** (sudo init Taste-**Null**, nicht großer Buchstabe O) oder **sudo poweroff** sicher herunter. Die ssh-Sitzung wird dabei beendet.

weitere Headless-Möglichkeiten: s. Raspberry Pi Geek, Heft 09-10/2018, S. 22-25.

B) Laptop/PC mit MS-Windows als Betriebssystem

Unter MS-Windows gibt es das Problem, dass das System keine **ext4-Dateisysteme** einbindet! Man benötigt also ein Hilfsprogramm, das die Einbindung solcher Dateisysteme erledigen kann, denn die **root-Partition** der Raspi-SD-Karte ist vom **Typ ext4**. Ein derartiges Freeware-Programm ist '**ExtFS for Windows**'[®] von [Paragon Software](#). **ExtFS** läßt sich unter MS-Windows in der Version 7 (und höher) installieren, aber nicht unter Windows-XP. Nach dem Einbinden (Mounten) der root-Partition kopiert man die '**interfaces**'-Datei zur Bearbeitung mit dem Editor auf den Desktop und kopiert sie nach dem Abspeichern der Veränderungen in das Verzeichnis '**/etc/network**' zurück.

Für die Einrichtung einer direkten ssh-Verbindung unter MS-Windows kann man die Open-Source-Programme '**Notepad++**' (als Editor), '**Xming**' (eine Implementierung eines X-Servers für Windows) und '**PuTTY**' (als 'secure shell' zum Herstellen der ssh-Verbindung) verwenden. Die drei Programme können aus dem Internet heruntergeladen und auf dem Laptop/PC installiert werden.

- Die feste IP-Adresse der Raspi-Netzwerkschnittstelle stellt man, wie oben beschrieben, ein; nur jetzt mit '**Notepad++**' als Editor für die Bearbeitung der '**interfaces**'-Datei.
- Zur Einrichtung der statischen IP des Laptops/PC benutzt man das windowseigene Werkzeug '**Netzwerkverbindungen**'.
- Mit '**PuTTY**' wird die ssh-Verbindung zum Raspi (über Netzwerkkabel mit dem Laptop/PC verbunden) wie folgt hergestellt:
 - PuTTY öffnen. In der rechten Spalte in die Zeile '**Host Name**' die IP des Raspi (z.B. `10.0.2.25`) eingeben. In der Zeile '**Port**' sollte schon 22 stehen.
 - In der linken Spalte die Kategorie '**SSH**' öffnen und '**X11**' anklicken; im rechten Fensterteil das Rechteck vor '**Enable X11 forwarding**' aktivieren. Ein Klick auf die

Schaltfläche *'open'* leitet die Anmeldung des Benutzers *'pi'* am Raspi ein.

- *'Xming'* starten.

In beiden Fällen (A und B) ist man nun in der Lage, vom Laptop/PC aus auf dem Raspi zu arbeiten und dort auch Programme zu starten. Zur Ausführung von *pro++chem*, um z.B. eine Messung vorzunehmen, wechselt man in das Verzeichnis, in dem das Programm zu finden ist (z.B. *'cd pro++chem'*) und gibt als Befehl *'./pro++chem'* ein. Das Programm und seine Ausgaben erscheinen dann auf dem Bildschirm des Laptops/PC, der mit dem Raspi verbunden ist und kann nun von dort aus uneingeschränkt genutzt werden.