



Das Geheimnis der *dev/hidraw0* bis *dev/hidraw3* Schnittstellen

Bei einem Raspberry 4 ist es z.B. so, daß es 4 USB-Ports gibt.
2 X USB3.0 und 2 X USB2.0

Bild links.

Die Reihenfolge ist ausgehend von der LAN-Buchse der obere 1.1, darunter 1.2, die beiden schwarzen, der obere 1.3, der untere 1.4

Wird ein USB-Hub verwendet, so setzt sich ein Mutter-Kind-Verhältnis fort.

Am Hub beginnt die Reihenfolge am Ausgangskabel zum Raspberry, wenn man die Ports selbst als Eingang wertet.

Im Bild recht sind demzufolge die {devpath} 1.1.1 und 1.1.2 belegt

Stöpselt man ein das erste Hidrawgerät an den oberen der beiden 3.0 Buchsen haben bei der Abfrage

```
udevadm info -a -n /dev/hidraw0 | grep '{devpath}' | head -n1
```

die Ausgabe

```
ATTRS{devpath}=="1.1"
```

beim Anstöpseln eines weiteren an die andere 3.0 Buchse und Abfrage

```
udevadm info -a -n /dev/hidraw1 | grep '{devpath}' | head -n1
```

die Ausgabe

```
ATTRS{devpath}=="1.2"
```

Analog hierzu geben die beiden anderen USB2.0-Buchsen bei den entsprechenden

Abfragen

```
udevadm info -a -n /dev/hidraw2 | grep '{devpath}' | head -n1
```

die Ausgabe

```
ATTRS{devpath}=="1.3"
```

und

```
udevadm info -a -n /dev/hidraw3 | grep '{devpath}' | head -n1
```

die Ausgabe

```
ATTRS{devpath}=="1.4"
```

Dies gilt solange der Rasperry eingeschalten und alle Geräte angestöpselt bleiben.

Zieht man ein Gerät ab, so bleiben die anderen weiter zugeordnet.

Stöpselt man es wieder ein, so fügt er sich auch wieder ein.

Startet man den Rasperry neu (z.B.Reboot) so würde beim Fehlen eines Gerätes zwar die Buchsen-Nummern 1.1 bis 1.4 erhalten bleiben, diese bekommen jedoch andere */dev/hidraw0 -3* zugeordnet.

Dieses kann man für die Solaranzeige.de umgehen. Ähnlich der individuellen Namensvergabe bei der Multireglerkonfiguration, fragt man nicht die {serial} ab sondern {devpath}

Durch den Konsolen-Befehl

```
sudo nano /etc/udev/rules.d/99-usb-serial.rules
```

erweitert man die seriellen Einträge um die devpath nach untenstehendem Muster, entsprechend seinen Ergebnissen.

```
SUBSYSTEM=="tty", ATTRS{serial}=="AB0KPCIS", SYMLINK+="geraet1", OWNER="pi"
SUBSYSTEM=="tty", ATTRS{serial}=="AQ00VCRQ", SYMLINK+="geraet2", OWNER="pi"
SUBSYSTEM=="tty", ATTRS{serial}=="AB0L3R4E", SYMLINK+="wunsch3", OWNER="pi"
SUBSYSTEM=="tty", ATTRS{serial}=="AB0L4GCR", SYMLINK+="shueco4", OWNER="pi"
SUBSYSTEM=="tty", ATTRS{devpath}=="1.1", SYMLINK+="hidraw0", OWNER="pi"
SUBSYSTEM=="tty", ATTRS{devpath}=="1.2", SYMLINK+="hidraw1", OWNER="pi"
SUBSYSTEM=="tty", ATTRS{devpath}=="1.3", SYMLINK+="hidraw2", OWNER="pi"
SUBSYSTEM=="tty", ATTRS{devpath}=="1.4", SYMLINK+="hidraw3", OWNER="pi"
```

Somit hat man zumindestens für die solaranzeige die Geräte eindeutig der USB-Buchse zugeordnet und für die entsprechende x.user.config.php den

```
$USBDevice = "geraet1";
oder
$USBDevice = "hidraw0";
usw.
definiert.
```

Was man als SYMLINK einträgt spielt absolut keine Rolle. Egal ob es sich beim Gerät um ein *dev/ttyUSB(x)* oder *dev/hidraw(x)* handelt. Bei der Aktivierung und Auslesen der `ATTRS{devpath}` wird generell die USB-Buchse gelesen.

In der Multireglervariante der Solaranzeige, funktioniert sogar, das Auslesen eines *dev/ttyUSB-* Gerätes wenn in der entsprechenden *x.user.config.php* ein solcher Eintrag

```
$USBDevice = "hidraw0";
```

vorhanden ist, wenn er in der

```
/etc/udev/rules.d/99-usb-serial.rules
```

als solcher deklariert wurde.

Diese Verfahrensweise funktioniert auch mit einer Kaskadierung per USB-Hub. Dort werden die Abfragen genauso gemacht und es erfolgen dann sicher solche Ausgaben

Im Browserformular des Webinstallers, kann man zwischen der namentlichen Vergabe und Hidraw wählen.