

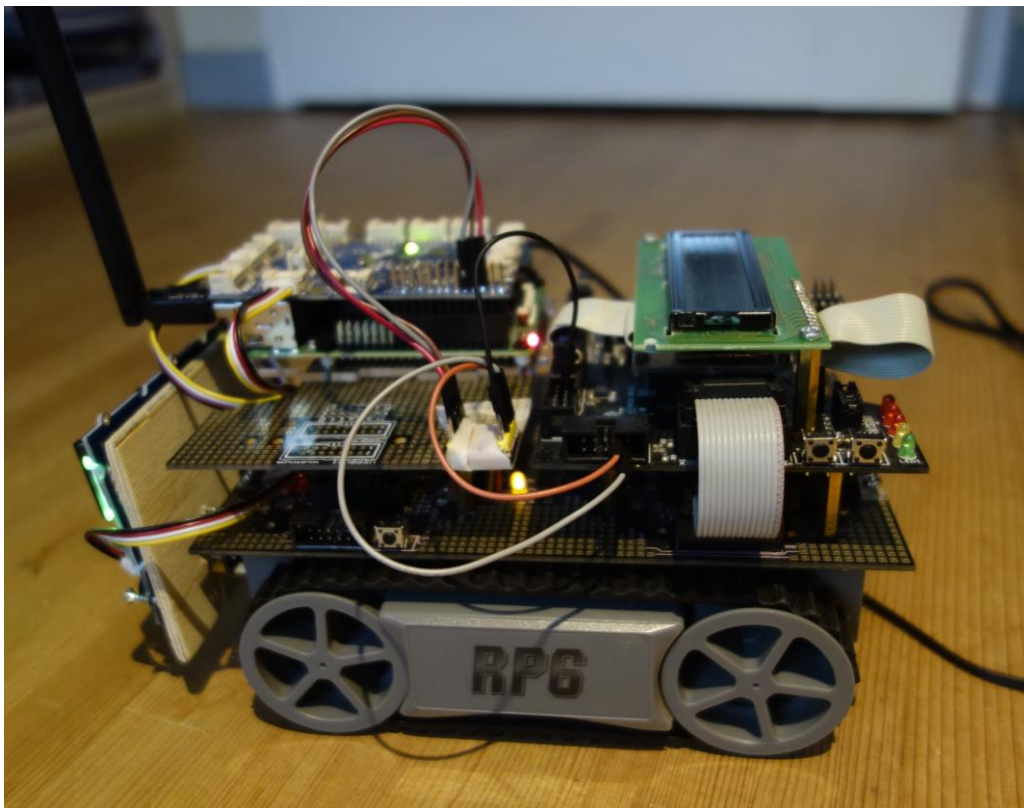
RP6 mit dem Raspberry Pi verbinden

Dieser Artikel zeigt, wie ein Raspberry Pi mit dem RP6 verbunden werden kann. Im Artikel werden Grundkenntnisse von Raspberry vorausgesetzt. Es geht nicht darum, am Schluss einen kompletten selbstfahrenden Roboter zu haben, bei dem alles funktioniert, sondern es geht darum, die Grundlagen zu liefern, um danach selber weiter zu entwickeln.

Sie finden alles Nötige dazu als ZIP Archiv zum Download.

Hardware

Als Hardware dient mir mein alter RP6 Roboter von Arexx.

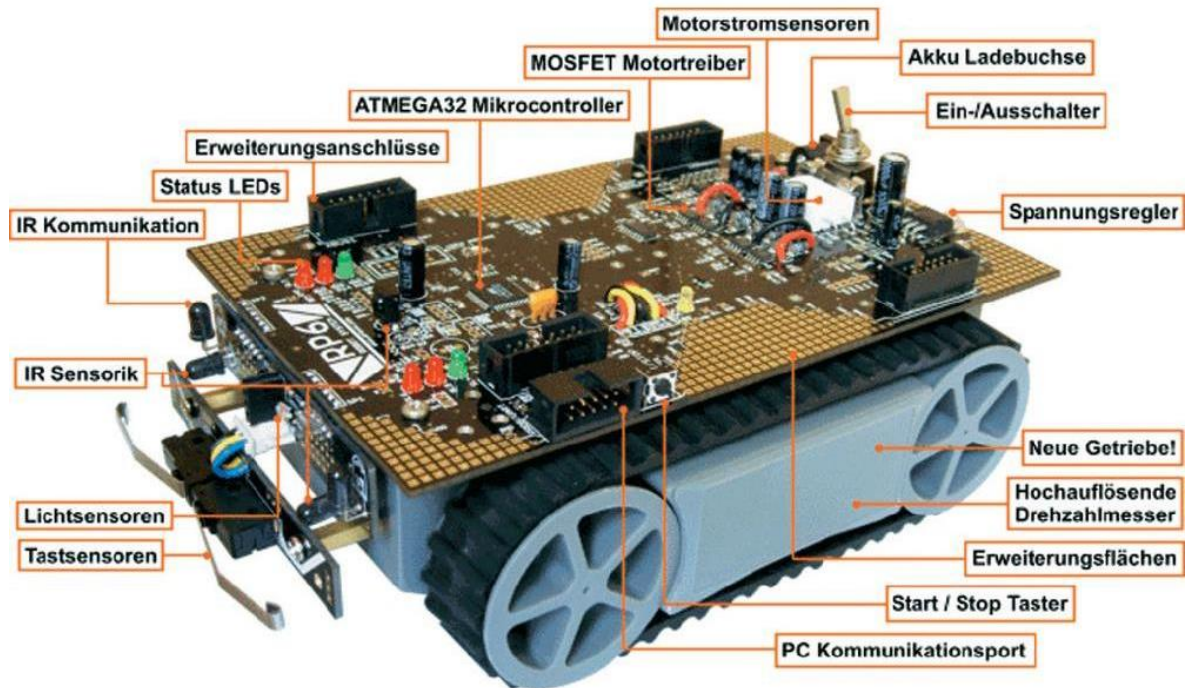


Links:

<http://www.arexx.com/rp6/html/de/>

Erhältlich ist dieser bei Conrad.

Zudem habe ich die M128 Erweiterung, diese braucht es aber eigentlich nicht, da auch alles mit dem Basisroboter gemacht werden kann.



Weiter habe ich einen Raspberry Pi mit einem GrovePi+ Starter Kit für Raspberry Pi.

<http://wiki.seeedstudio.com/wiki/GrovePi+>

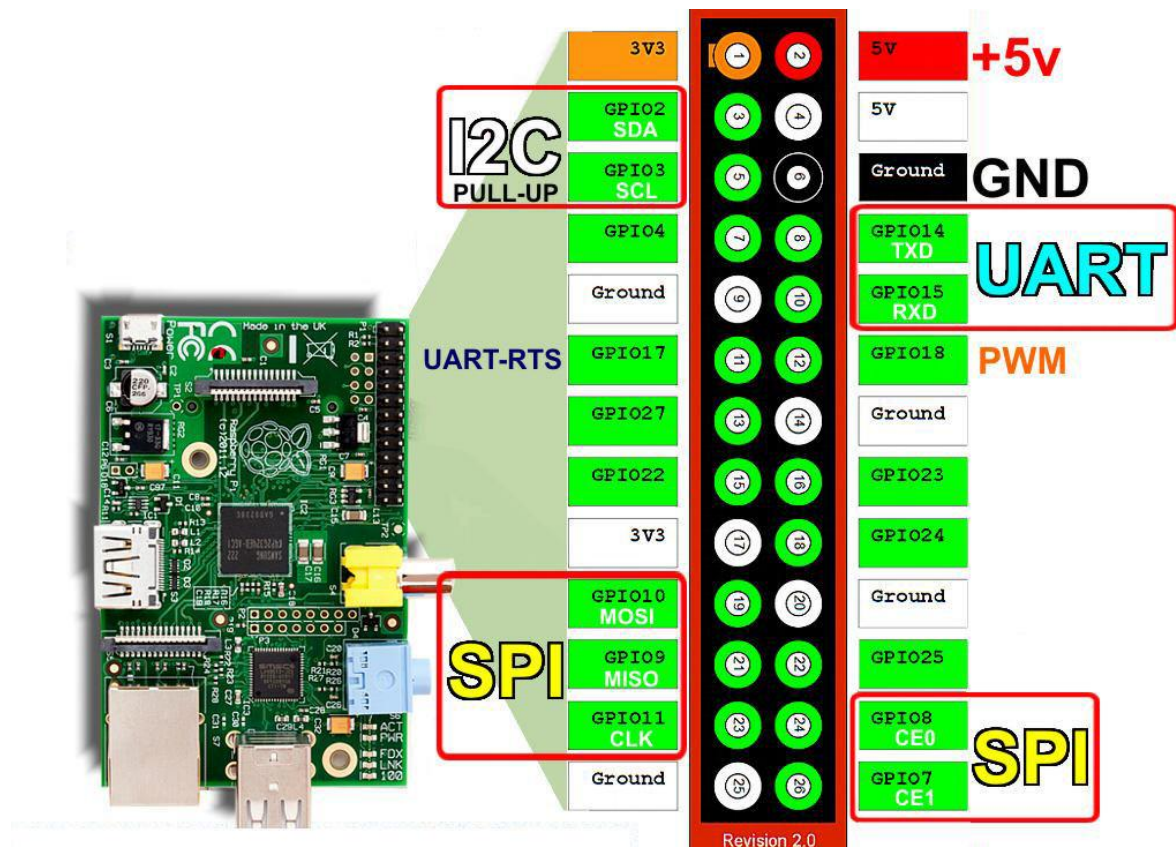
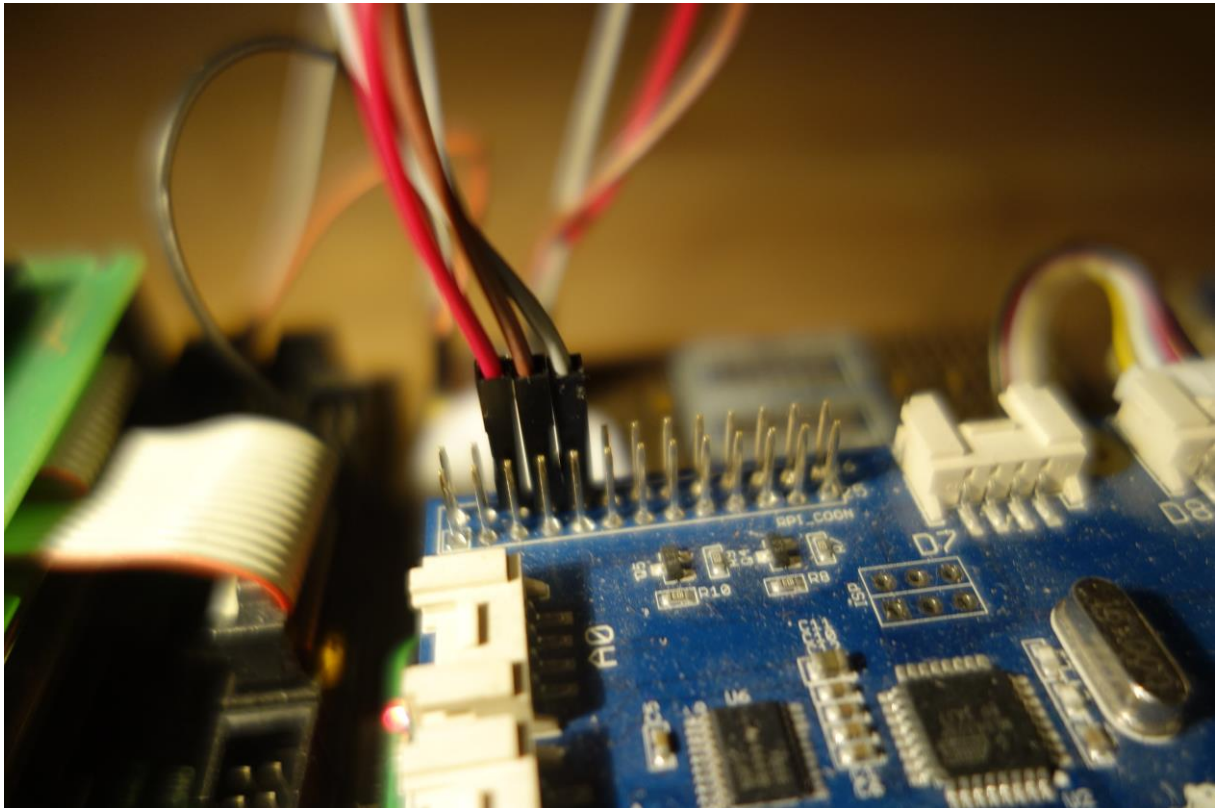
Da der Raspberry Pi mit 3.3V an den GPIO arbeitet und der RP6 mit 5V, brauche ich noch ein Logic Level Converter.



https://www.aliexpress.com/store/product/Freeshipping-Logic-Level-Converter-converts-3-3V-5V-TTL-logic-level-conversion/925845_932132709.html

Die HV Seite ist mit dem RP6 verbunden die LV Seite mit dem PI.

Die Verbindung ist auf den folgenden Fotos ersichtlich.



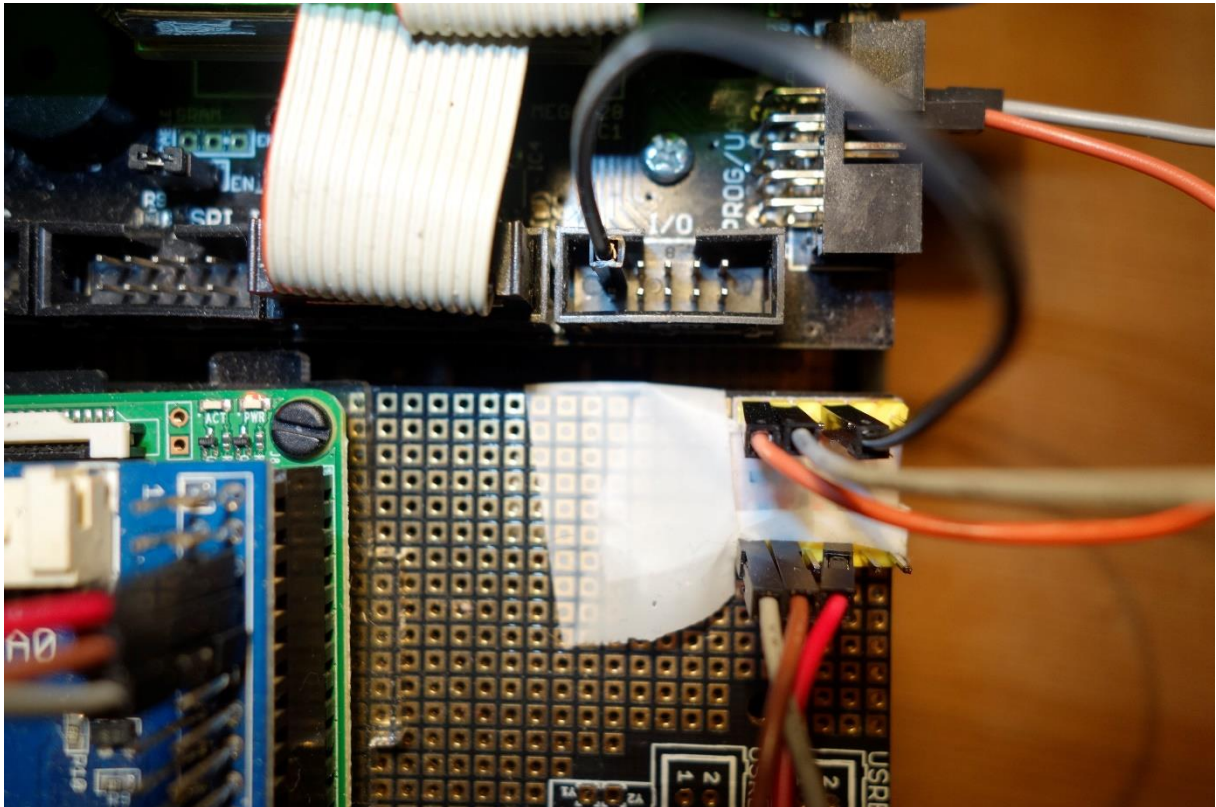
Rot=GND

Braun= TX

Grau= RX

LERNTOL.CH

Auf dem Logic Level Converter sind TX und RX gekreuzt, da das, was der Raspberry Pi sendet beim RP6 auf den Empfänger gehen muss.



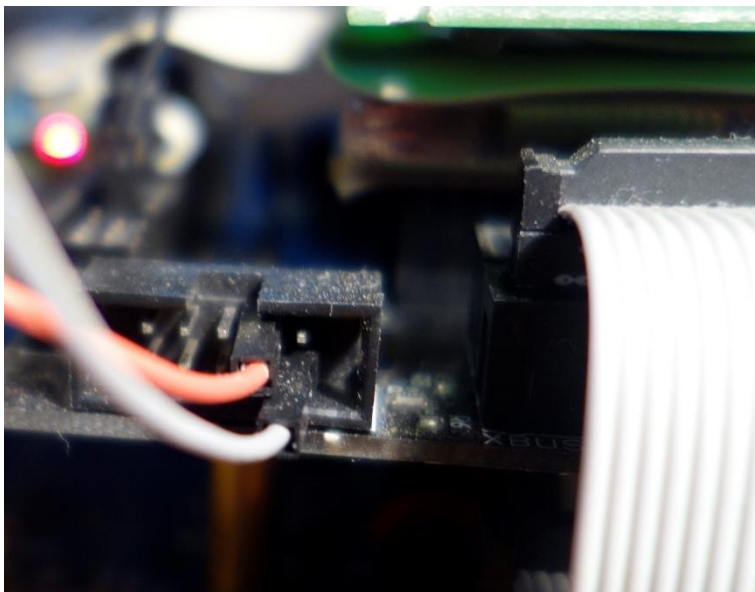
Hier sehen wir die Belegung auf dem Konverter:

Orange=TX

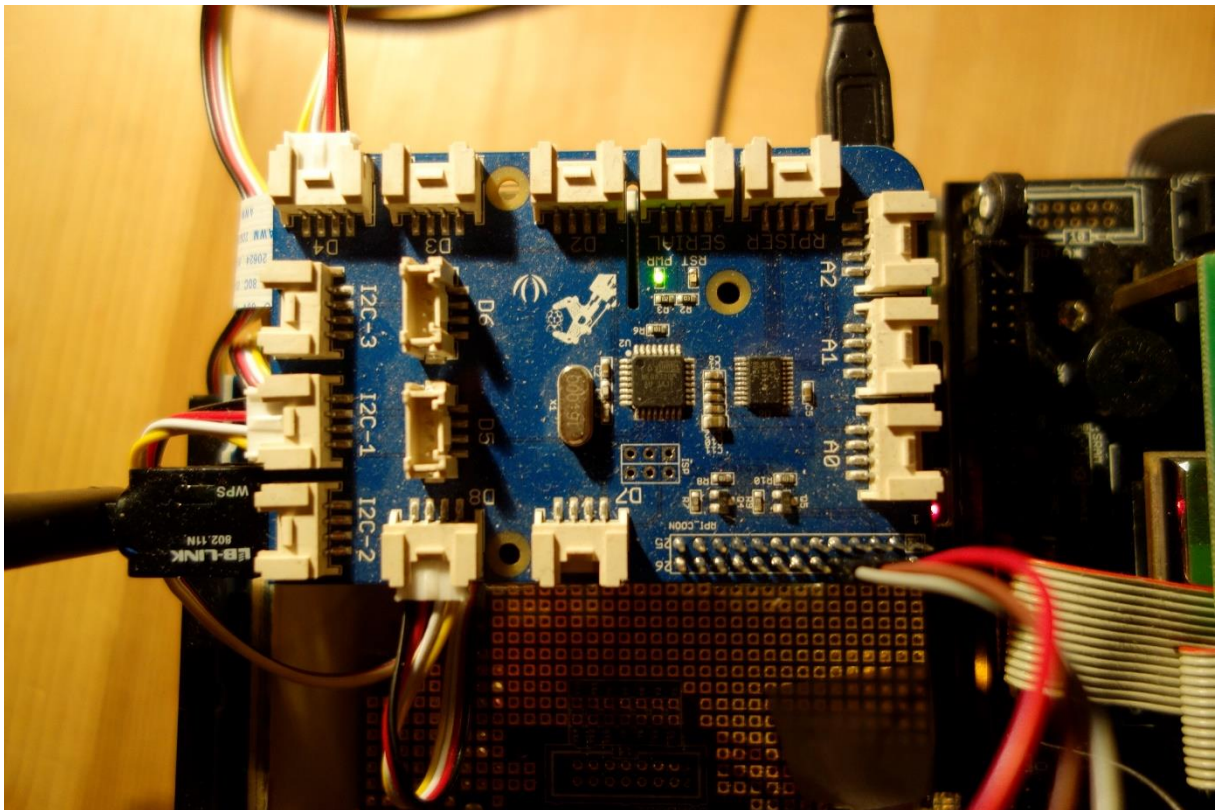
Grau=RX

Schwarz=GND

GND ist auf dem RP6 ebenfalls mit GND verbunden. Die anderen beiden gehen beim PROG/UART Stecker auf die Pins 2 und 3.



Auf dem PI Grove Board sind die Sensoren folgendermassen angeschlossen: Am I2C-1 ist das LCD Display angehängt. Am D4 hängt der Ultraschallsensor und am D8 der Piepser. Die Kamera ist direkt im Anschluss des Pi's montiert.



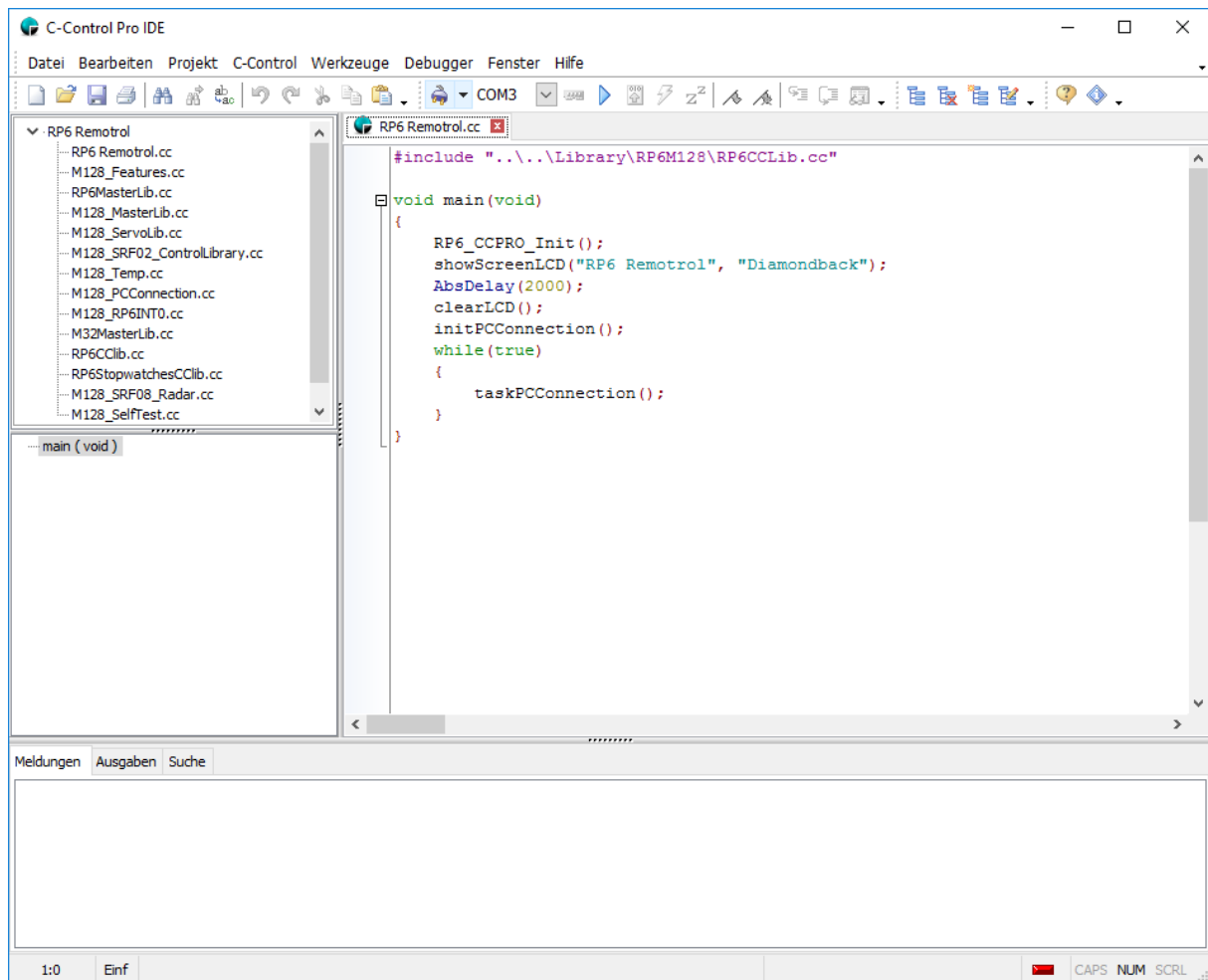
Auf dem RP6

Auf dem RP6 läuft die Software von Diamondback (<https://sourceforge.net/projects/rp6remotrol/>).

Diese wird mit der Software C-Control Pro IDE auf den RP6 in das Erweiterungsboard M128 geladen.

Die Software C-Control Pro IDE liegt im Archiv unter " Software für RP6\C-Control Pro IDE \C-ControlSetup.exe"

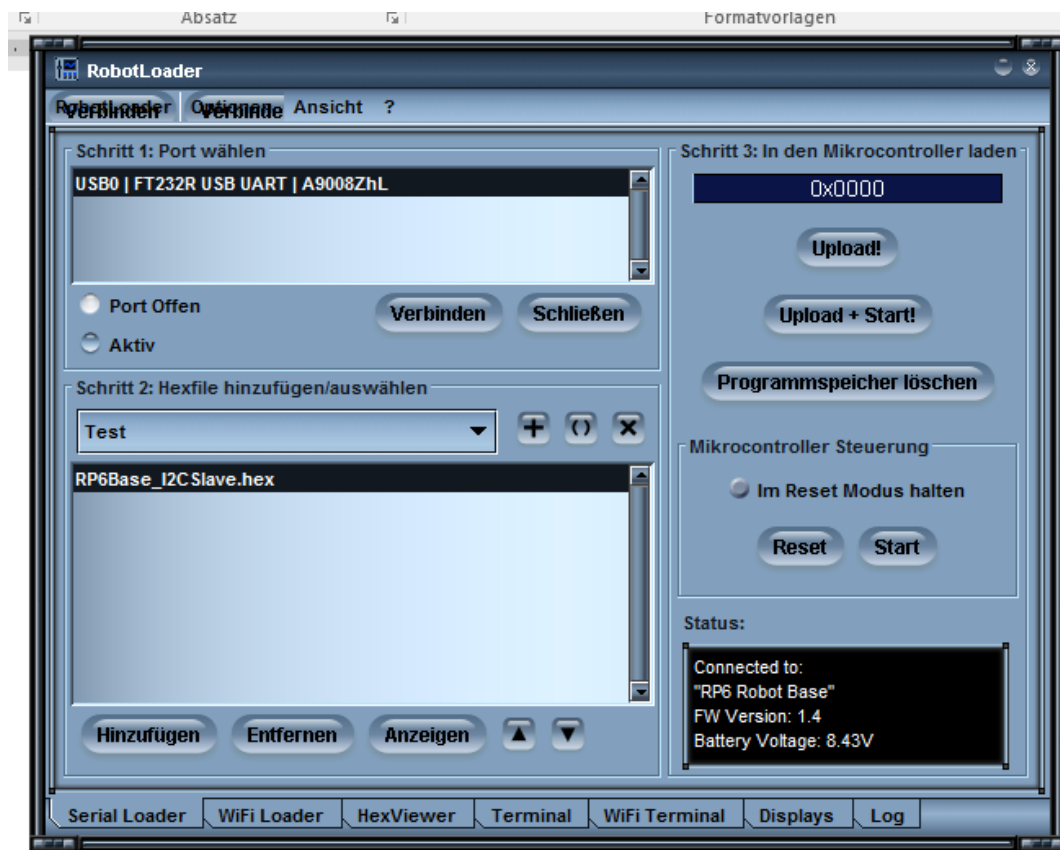
Die Software für den RP6 ist im Archiv unter " Software für RP6\rp6remotrol\Firmware auf RP6\M128 - Master\M128 – Master"



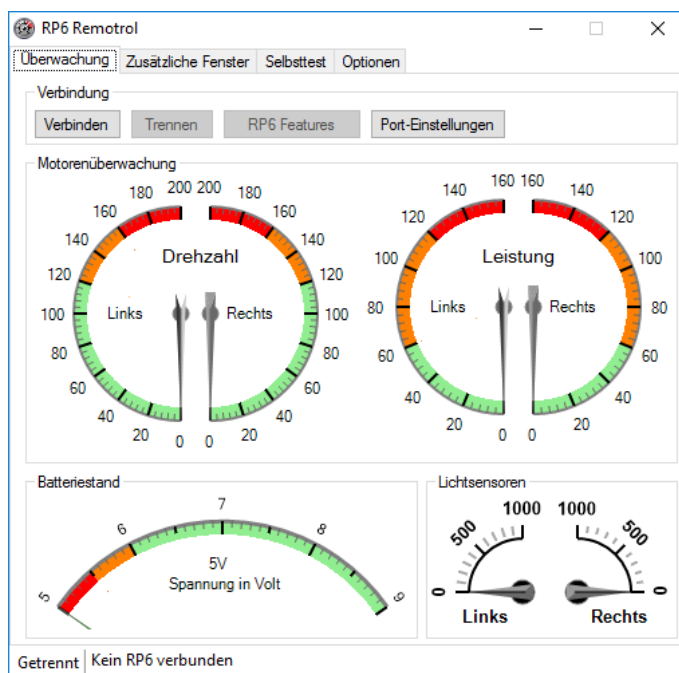
Auf dem Basisboard muss ebenfalls eine Software installiert werden. Dies liegt im Ordner "Software für RP6\\rp6remotrol\\Firmware auf RP6\\M128 - Master\\RP6 – Slave". (Verbindung via Kabel auf Basisplatine)

Um diese Software auf den RP6 zu bringen ist der RobotLoader nötig:

"Software für RP6\\RobotLoader_20130410\\RobotLoader_start.exe"



Nun kann der RP6 mit dem PC verbunden werden und mit dem Programm "RP6 Remotrol.exe" via serielle Schnittstelle gesteuert werden. Dazu muss das Serial Kabel wieder mit dem Erweiterungsboard verbunden werden.



Raspberry Pi

Auf dem Raspberry Pi muss das Image "Raspbian for Robots" von Dexterindustries geladen werden.

<https://sourceforge.net/projects/dexterindustriesraspbianflavor/>

Eine Anleitung wie das geht gibt es hier:

<http://www.dexterindustries.com/howto/install-raspbian-for-robots-image-on-an-sd-card/>

Alternativ kann die GrovePi Erweiterung auch auf dem Raspian installiert werden:

<http://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/>

Wenn Sie sich beim "Raspbian for Robots" anmelden, benötigen sie den User "**PI**" und das Passwort "**robots1234**"

Es ist empfehlenswert, den Raspberry Pi anschliessend zu updaten:

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo rpi-update
```

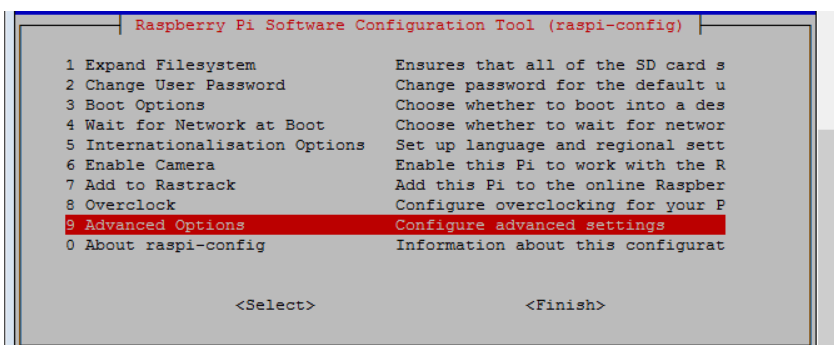
Anschliessend updaten wir die Grove Firmware. Dazu führen wir im Ordner Desktop/GrovePi/Firmware folgende Befehle aus:

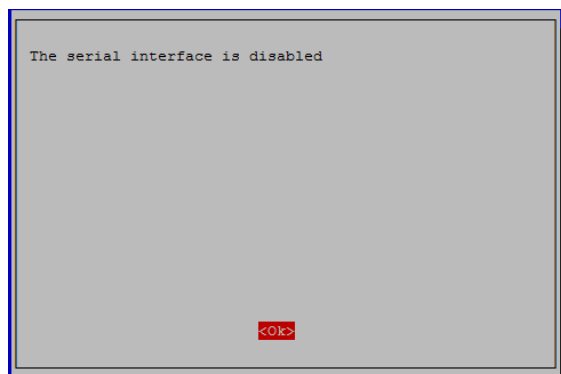
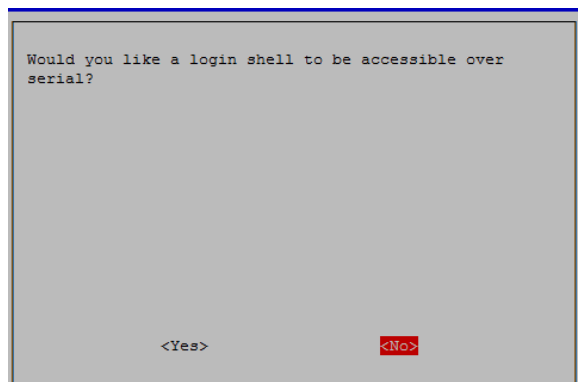
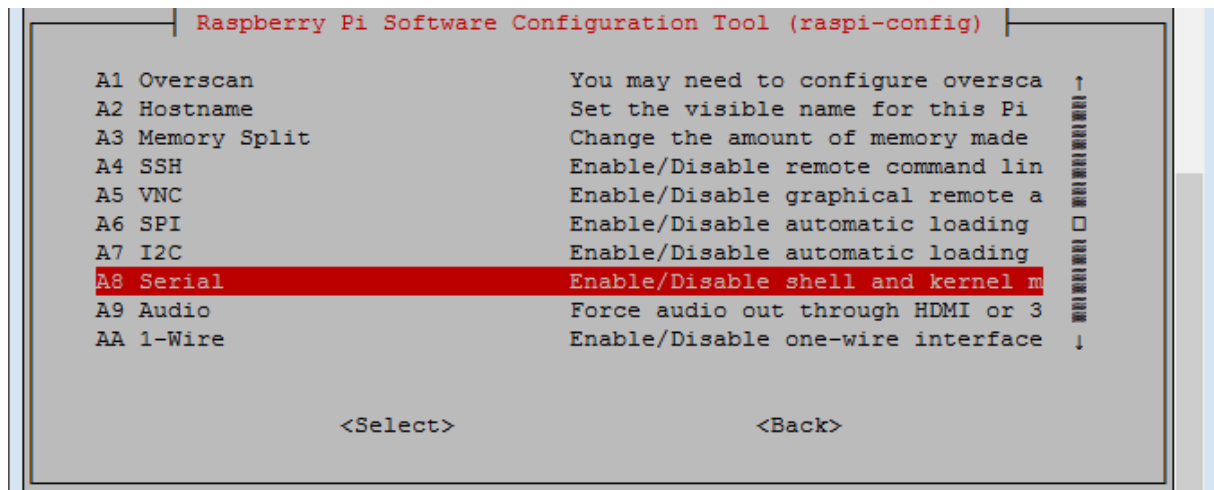
```
$ sudo chmod +x firmware_update.sh
$ sudo ./firmware_update.sh
```

Anschliessend müssen Sie die Konsolenausgabe des Raspberry Pi in die serielle Schnittstelle ausschalten.

```
$ sudo raspi-config
```

Advanced-Options->Serial->Would you like a login shell to be accessible over serial? ->No->OK



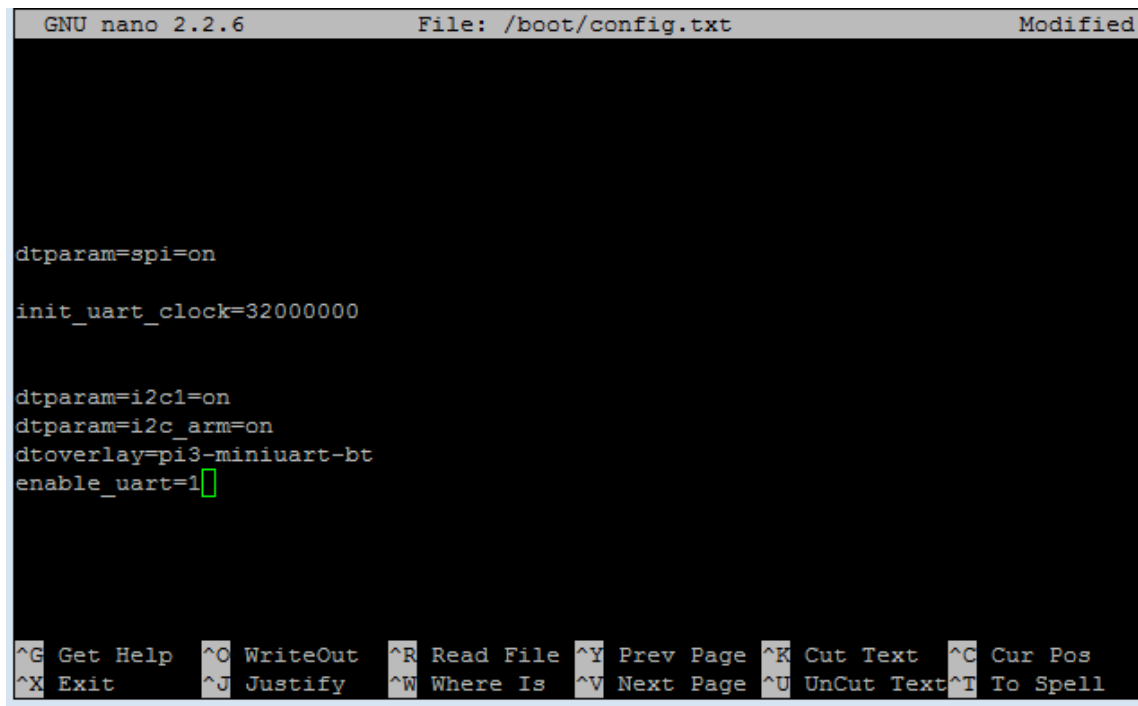


Anschliessend müssen wir die serielle Schnittstelle für einen Zugriff auf den RP6 konfigurieren.

```
$ sudo nano /boot/config.txt
```

Zuunterst fügen wir diese Zeile ein:

```
enable_uart=1
```



```

GNU nano 2.2.6      File: /boot/config.txt      Modified

dtparam=spi=on

init_uart_clock=32000000

dtparam=i2c1=on
dtparam=i2c_arm=on
dtoverlay=pi3-miniuart-bt
enable_uart=1

```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
 ^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

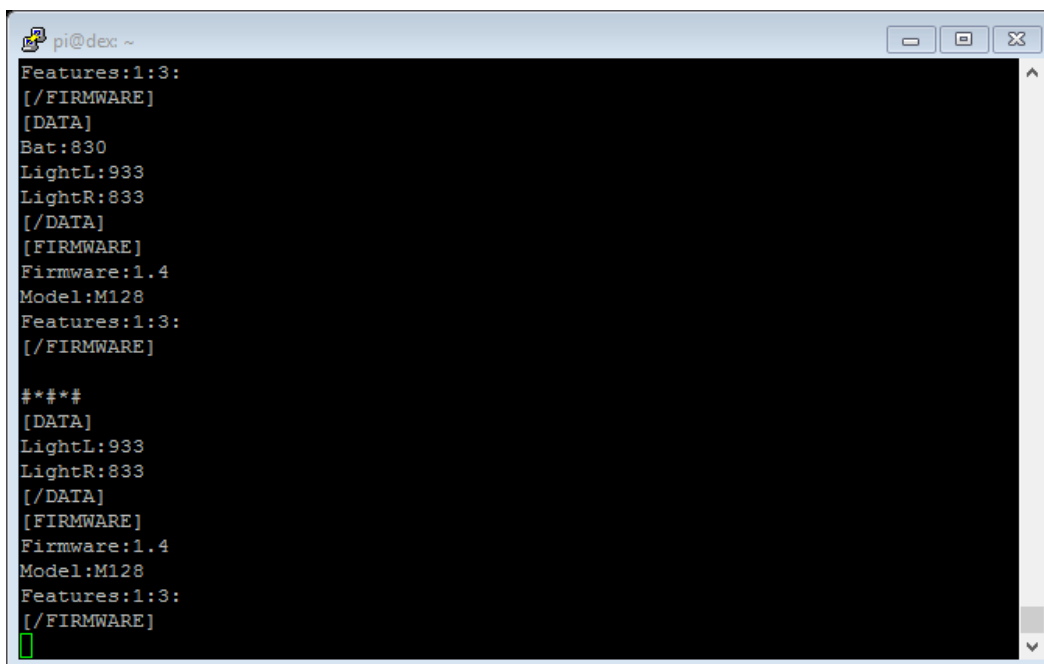
Nun installieren wir noch die Software Minicom:

```
$ sudo apt-get install minicom
```

Danach verbinden wir uns via serielle Schnittstelle mit dem RP6. Dazu muss auf dem RP6 das Programm ebenfalls gestartet werden.

```
$ sudo minicom -b 38400 -D /dev/ttyAMA0
```

Wenn alles klappt, sehen Sie nun sehr viele Zeichen auf dem Bildschirm. Mit: **Ctrl+A** und danach **u** können Sie diese in ein vernünftiges Format bringen.



```

pi@dex ~
Features:1:3:
[/FIRMWARE]
[DATA]
Bat:830
LightL:933
LightR:833
[/DATA]
[FIRMWARE]
Firmware:1.4
Model:M128
Features:1:3:
[/FIRMWARE]

####
[DATA]
LightL:933
LightR:833
[/DATA]
[FIRMWARE]
Firmware:1.4
Model:M128
Features:1:3:
[/FIRMWARE]

```

Mit **Ctrl+A** und **q** verlassen Sie das Terminal wieder.

Allfällige Rechtprobleme lösen Sie mit:

```
$ sudo chmod a+rw /dev/ttyAMA0
```

Dies vergibt Lese- und Schreibrechte auf der UART.

Installieren von xrdp

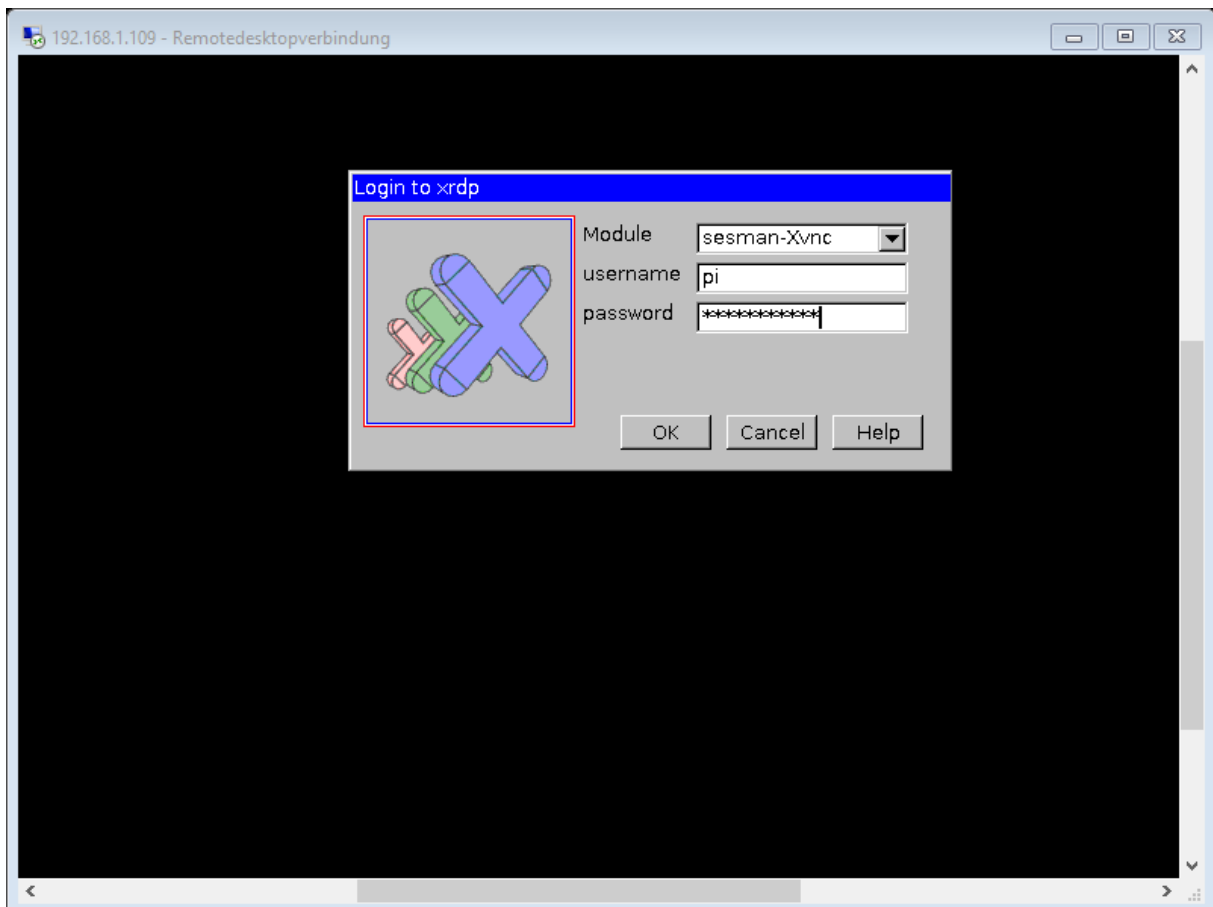
Nun installieren Sie xrdp damit Sie nicht nur in der Konsole, sondern auch via Remote Desktop auf die grafische Oberfläche zugreifen können.

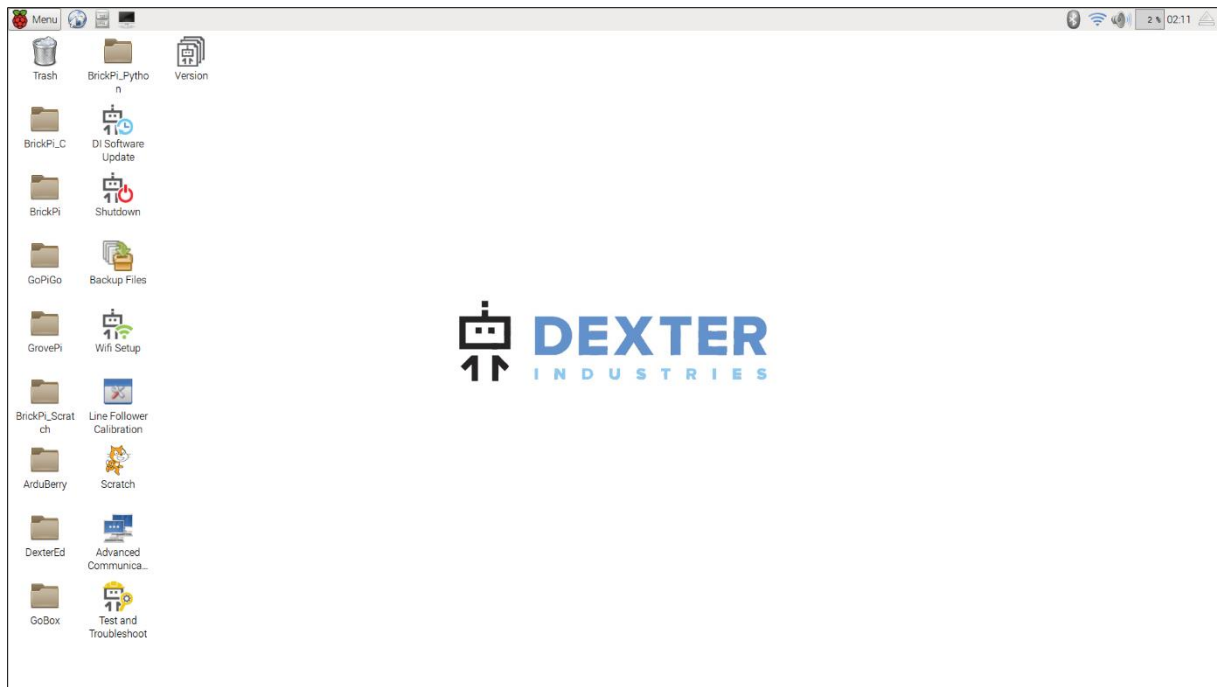
```
$ sudo apt-get install xrdp
```

Es ist möglich, dass XRDP noch nicht das CH-Tastaturlayout verwendet. Folgende Befehle schaffen Abhilfe:

```
$ cd /etc/xrdp
$ sudo mv km-0409.ini alt-km-0409.ini
$ sudo wget http://www.mail-archive.com/xrdp-devel@lists.sourceforge.net/msg00614/km-0807.ini
$ sudo mv km-0807.ini km-0409.ini
$ sudo chown xrdp:xrdp km-0409.ini
$ sudo service xrdp restart
```

Nun können Sie via Remote Desktop auf den Raspberry Pi zugreifen.





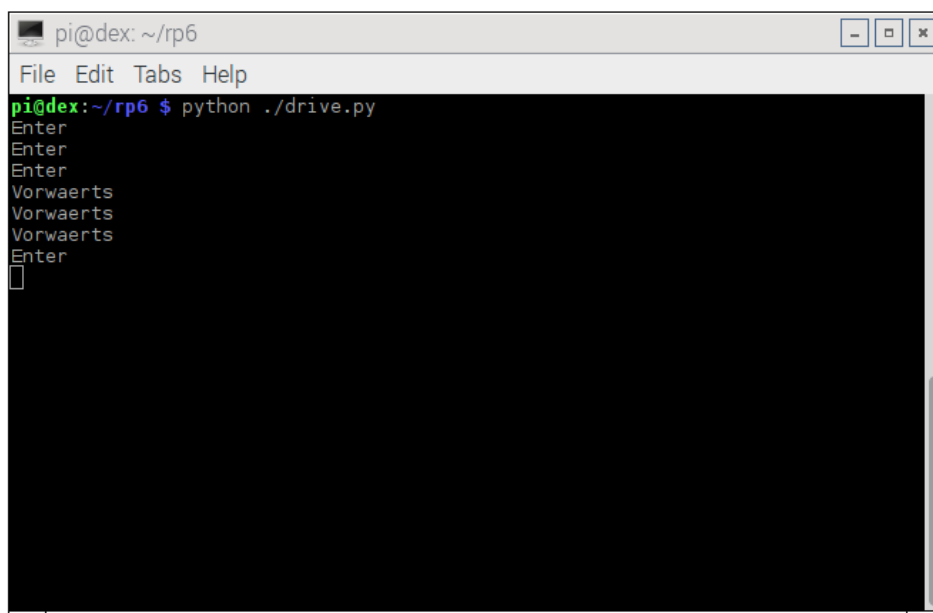
Installieren von Python

Um mit Python auf die serielle Schnittstelle zuzugreifen, installieren Sie noch das folgende Paket:

```
$ sudo apt-get install python-serial
```

Nun kopieren Sie den Python Script aus dem Ordner "\demo Script\drive.py" via FTP auf den Raspberry Pi.

Dieses können Sie in der grafischen Oberfläche in der Shell ausführen. Ein Ausführen via Shell ohne die grafische Oberfläche geht nicht, da ich die Bibliothek "pygame" benutze, um die Pfeiltasten zu gebrauchen.



Ich gehe hier nicht näher auf den Code ein. Dieser ist in der Datei "\demo Script\drive.py" kommentiert.

Kommunikation über die Serielle Schnittstelle

Das Programm von Diamondback empfängt die Informationen über ein eigenes Protokoll.

Zum Beispiel:

#3:0:1:1 * # Schaltet die rechte grüne LED am RP6 ein.

#3:0:0:2 * # Schaltet die rechte grüne LED am RP6 wieder aus.

Wir sehen, dass die zweithinterste Stelle für das ein- oder ausschalten zuständig ist. Die hinterste Stelle ist ein Counter. Dieser muss hochzählen, sonst wird der Befehl nicht als gültig akzeptiert.

In meinem Programm umgehe ich das, indem ich mit #99:0* den Counter nach jedem Befehl zurücksetze.

Hier nun eine Übersicht über die Befehle, immer mit Counter = 1:

#99:0* = Reset counter

LED RP6 rechts #3:0:1:1 * #LED 1 Grün ein #3:0:0:1 * #LED 1 Grün aus #3:0:2:1 * #LED 2 Rot ein #3:0:0:1 * #LED 2 Rot aus #3:0:4:1 * #LED 3 Rot ein #3:0:0:1 * #LED 3 Rot aus	LED RP6 links #3:0:8:1 * #LED 4 Grün ein #3:0:0:1 * #LED 4 Grün aus #3:0:16:1 * #LED 5 Rot ein #3:0:0:1 * #LED 5 Rot aus #3:0:32:1 * #LED 6 Rot ein #3:0:0:1 * #LED 6 Rot aus	LED M128 #3:2:1:1* #LED 1Grün ein #3:2:2:1* #LED 2 Gelb ein #3:2:4:1* #LED 3 Rot ein #3:2:8:1* #LED 4 Rot ein #3:2:16:1* #LED 5 Rot ein #3:2:0:1* Löschen
---	--	--

Kombinationen der LEDS wird durch binäres Hochzählen erreicht.

Fahren vorwärts links #1:0:0:0:1* Stopp #1:6:0:0:1* Langsam #1:200:0:0:1* Schnell	Fahren vorwärts rechts #1:0:0:0:20* Stopp #1:0:6:0:1* Langsam #1:0:200:0:1* Schnell	Miteinander #1:6:6:0:23* Langsam #1:200:200:0:1* Schnell
---	---	---

Fahren rückwärts links #1:0:0:0:1* Stopp #1:6:0:1:1* Langsam #1:200:0:1:1* Schnell	Fahren rückwärts rechts #1:0:0:0:20* Stopp #1:0:6:1:1* Langsam #1:0:200:1:1* Schnell	Miteinander #1:6:6:1:23* Langsam #1:200:200:1:1* Schnell
--	--	---

#7:1* Notstop

Aktivieren Sensoren

Lichtsensoren #6:1:1:1* Aktivieren #6:1:0:1* Deaktivieren	Interruptsensoren #6:3:1:1* Aktivieren #6:3:0:1* Deaktivieren	Temperatursensoren #6:2:1:1* Aktivieren #6:2:0:1* Deaktivieren	Allgemeine Sensoren #6:0:1:23* Aktivieren #6:0:0:24* Deaktivieren
---	---	--	---

Rückgabewerte

```
####  
[DATA]  
Bat:831  
LightL:976  
LightR:938  
[/DATA]
```

Geschwindigkeit der Motoren SpeedL: SpeedR:	Motorleistung PowerL: PowerR:	Lichtsensor LightL: LightR:	RP6 Infos Firmware: Model:
---	-------------------------------------	-----------------------------------	----------------------------------

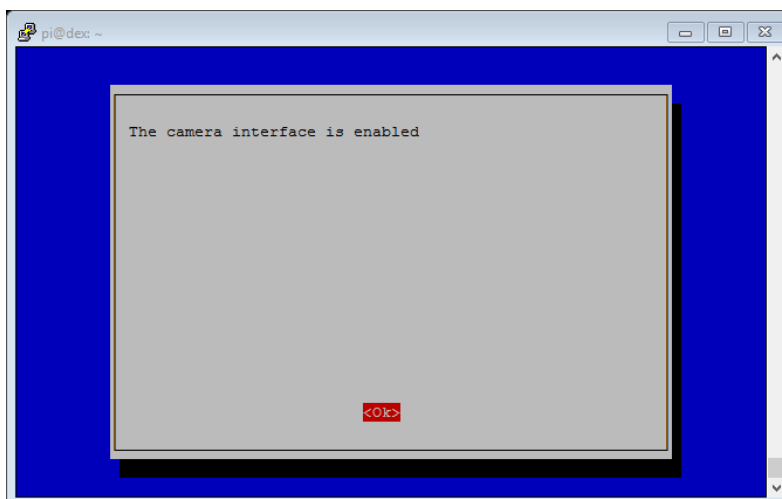
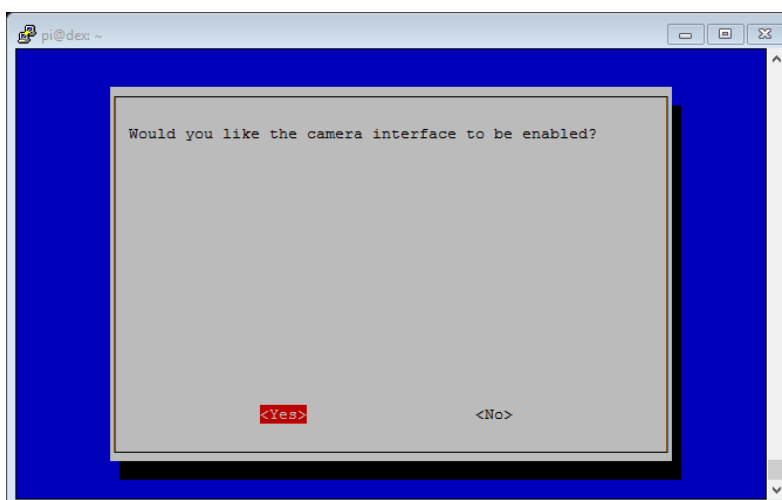
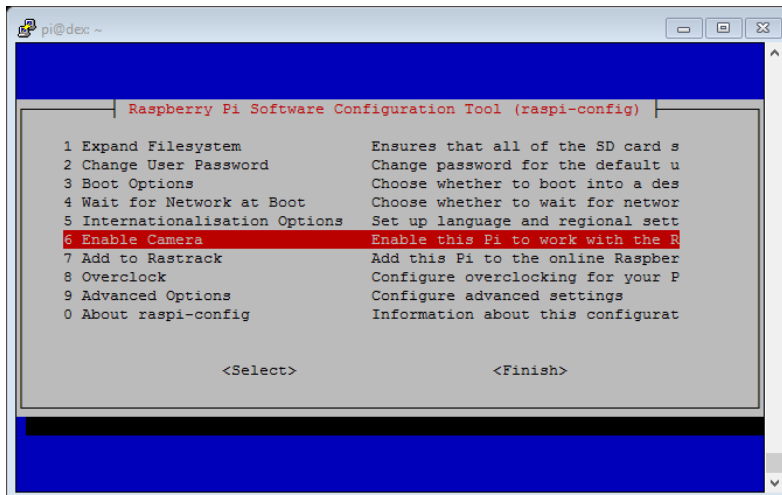
Hindernisse ObsL: ObstR:	Bumper BumpR: BumpL:	Diverses Distance: Mic: Temp: Features:	
--------------------------------	----------------------------	---	--

Einige Werte konnte ich mit meinem Model des RP6 nicht ansprechen, da diese möglicherweise defekt sind. Aus diesem Grund habe ich auch die Pi-Grove Erweiterung montiert.

Kamera und Webcam

Nun aktivieren wir die Kamera.

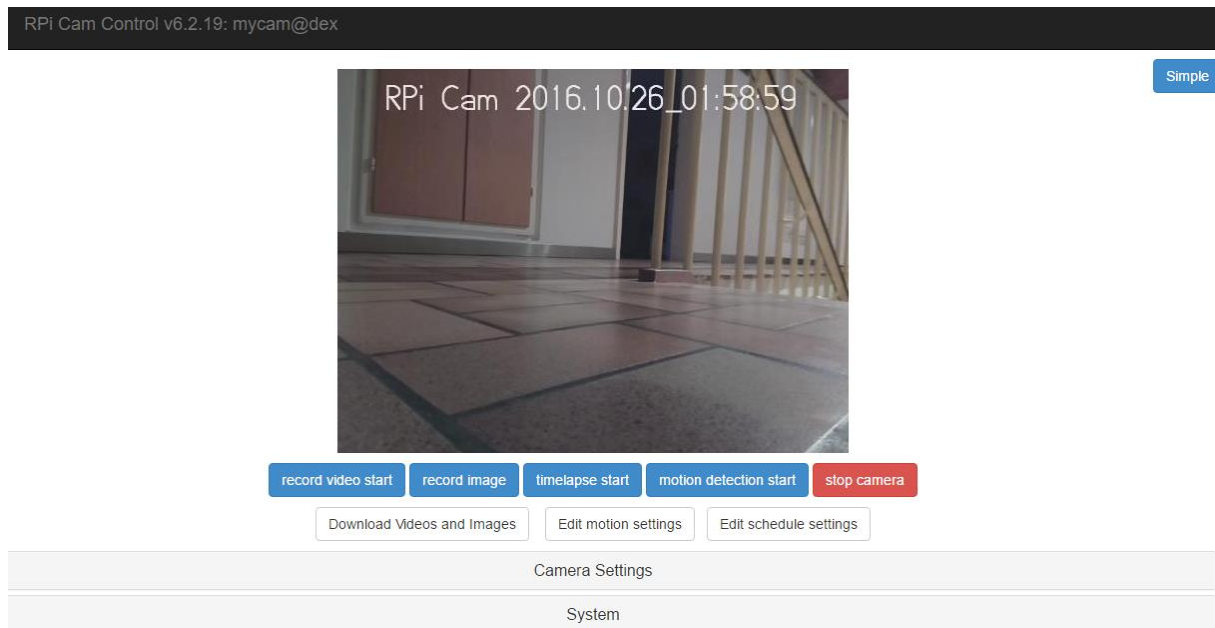
```
$ sudo raspi-config
```



Danach installieren wir die Webcam Software von Silvan Melchior:

```
$ git clone https://github.com/silvanmelchior/RPi_Cam_Web_Interface.git
$ cd RPi_Cam_Web_Interface
$ chmod u+x *.sh
$ ./install.sh
```

Nun können wir im Browser auf die IP von unserem Raspberry Pi zugreifen und sehen das Bild aus der Kamera:



Ein Fahren über die Weboberfläche habe ich nicht realisiert, denke aber das dies durchaus möglich ist.