

Einrichtung des Betriebssystems des Raspberry Pi für die Benutzung mit der RasPiLC

Laden Sie sich zuerst die aktuelle Version des Betriebssystems „Raspbian Jessie lite“ unter <https://www.raspberrypi.org/downloads/raspbian/> herunter.

Dazu benötigen Sie einen Kartenleser und unter Windows das Programm „Win32DiskImager“, um das Betriebssystem auf die SD-Karte des Raspberry Pi zu übertragen. Dieses können Sie unter der folgenden Adresse herunterladen: <https://sourceforge.net/projects/win32diskimager/>

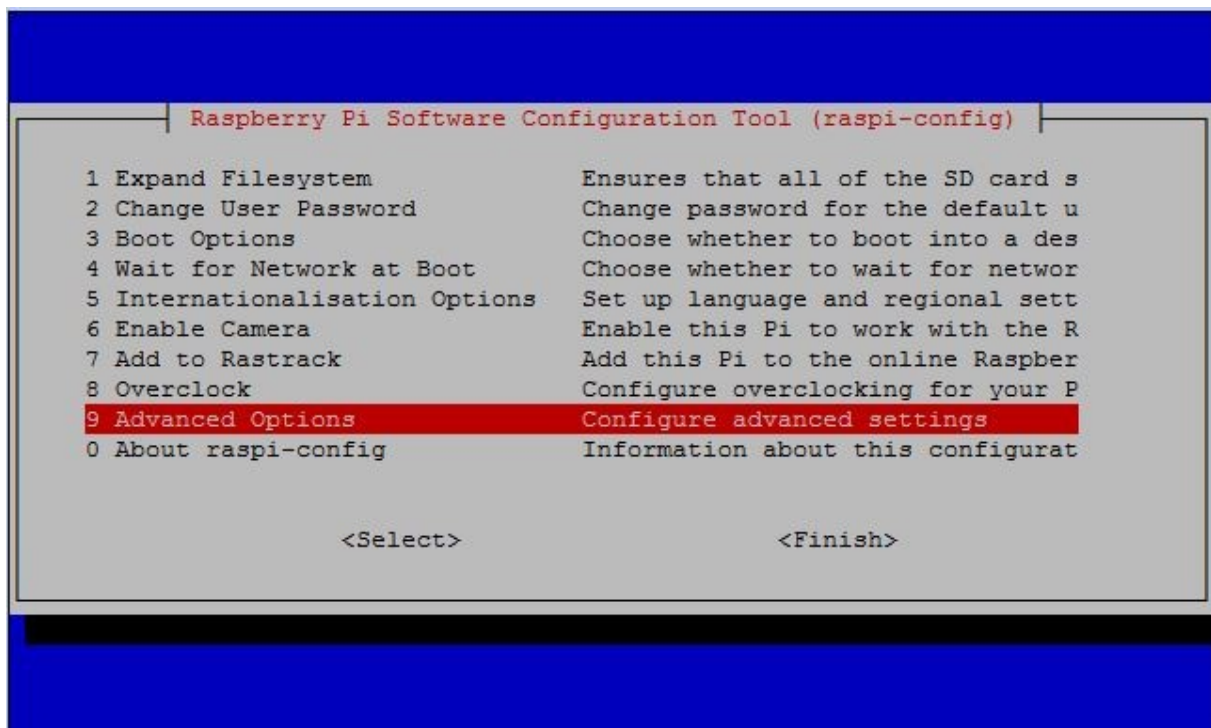
Entpacken Sie nun das heruntergeladene Image des Betriebssystems und übertragen Sie es auf die SD-Karte.

Danach können Sie die SD-Karte in Ihren Raspberry Pi stecken und ihn booten. Jetzt haben Sie zur Konfiguration zwei Möglichkeiten. Entweder Sie schliessen den Raspberry Pi direkt an Tastatur und Monitor an oder Sie benutzen einen Fernzugriff über eine SSH-Verbindung, welche allerdings eine Netzwerkverbindung voraussetzt. Die Vorgehensweise ist anschliessend für beide Varianten identisch.

Nach dem Start des Raspbian ist es zuerst notwendig, das Dateisystem auf die volle Größe der SD-Karte auszuweiten. Dazu (wenn noch nicht geschehen), loggen Sie sich im Betriebssystem ein (Standardbenutzer: „pi“, Passwort: „raspberrry“). Jetzt laden Sie das Konfigurationstool mit dem Befehl `sudo apt-get install raspi-config` herunter. Anschliessend starten Sie es mit `sudo raspi-config`.

Jetzt ist es notwendig, die Unterstützung für SPI, I2C und den Onewire-Bus zu aktivieren. Dazu navigieren Sie bitte mit den Pfeiltasten auf die jeweiligen Punkte wie in den folgenden Bildern gezeigt.

Zuerst „Advanced Options“ auswählen:

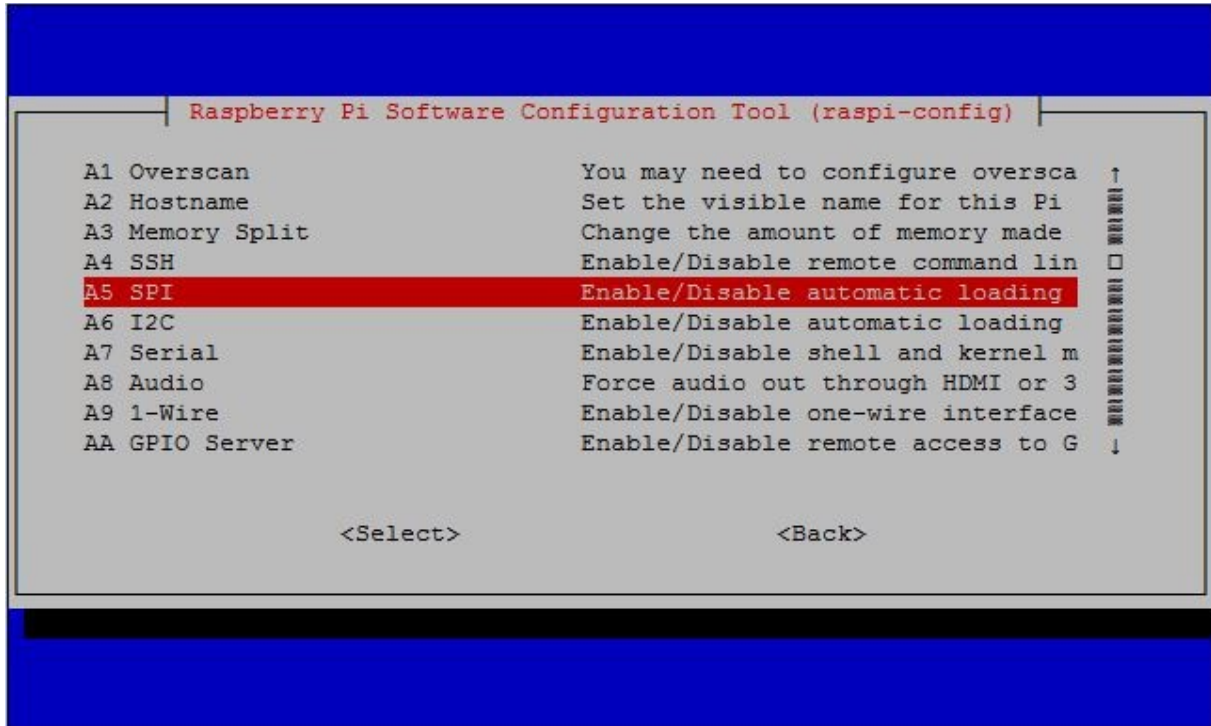


```
Raspberry Pi Software Configuration Tool (raspi-config)

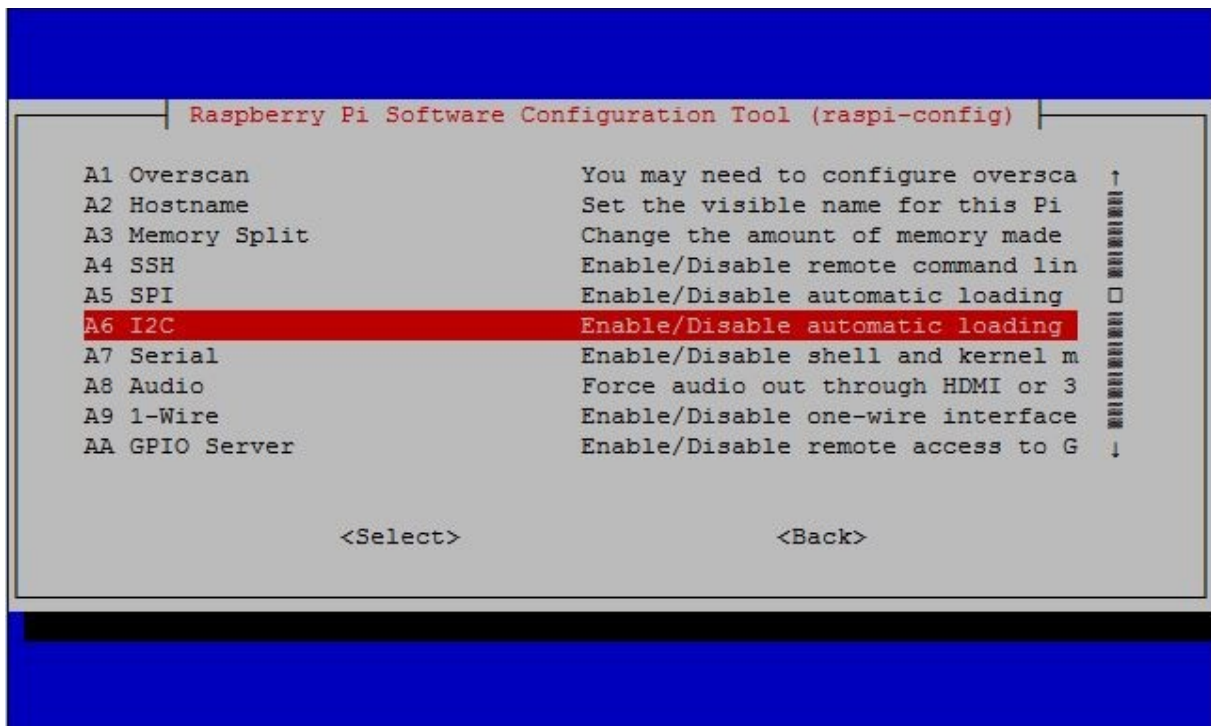
1 Expand Filesystem           Ensures that all of the SD card s
2 Change User Password        Change password for the default u
3 Boot Options                 Choose whether to boot into a des
4 Wait for Network at Boot     Choose whether to wait for networ
5 Internationalisation Options Set up language and regional sett
6 Enable Camera                Enable this Pi to work with the R
7 Add to Rastrack              Add this Pi to the online Raspber
8 Overclock                    Configure overclocking for your P
9 Advanced Options             Configure advanced settings
0 About raspi-config           Information about this configurat

<Select>                       <Finish>
```

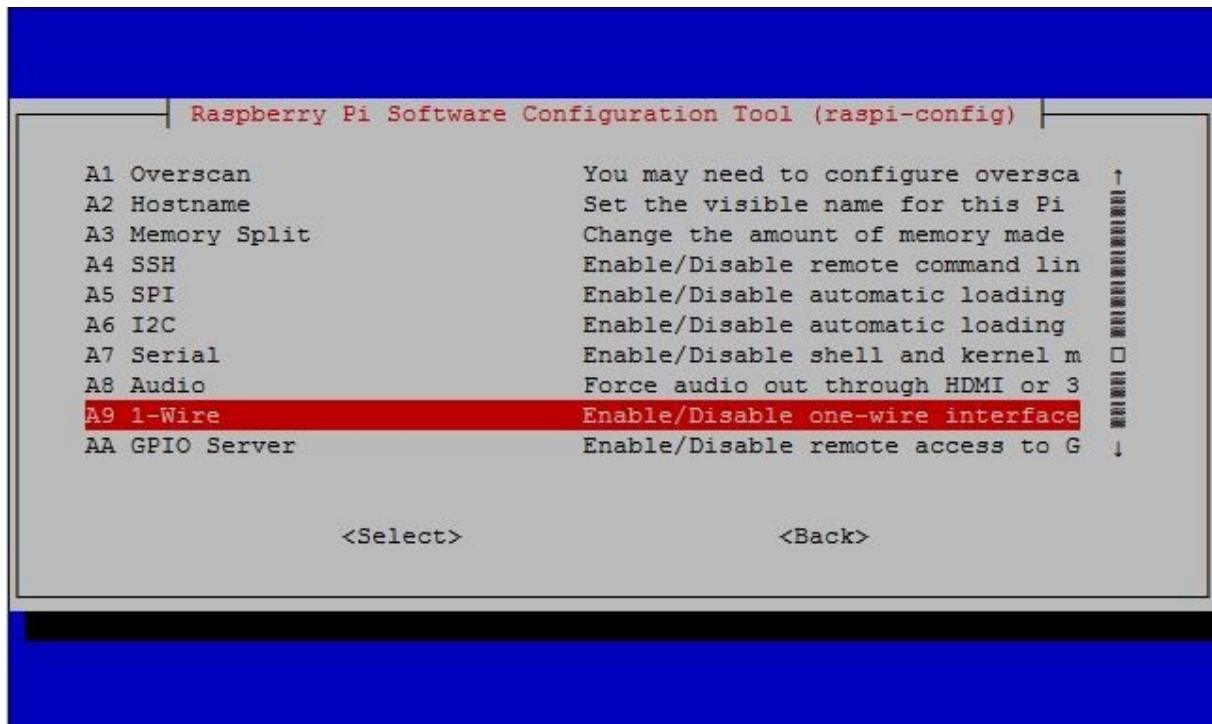
Anschließend SPI aktivieren:



Danach die I2C-Bus-Unterstützung einschalten:



... und den Onewire-Bus aktivieren.



Anschließend das Programm mit „Back > Finish“ verlassen und mit dem Befehl **sudo reboot** neu starten.

Jetzt müssen wir dem System mitteilen, dass wir (wenn gewünscht) die CAN-Bus-Unterstützung brauchen. Dafür müssen wir die Datei „/boot/config.txt“ anpassen und ergänzen.

Öffnen Sie hierfür diese mit einem Texteditor (wir empfehlen „nano“) und beziehen uns im Folgenden darauf:

sudo nano /boot/config.txt und ergänzen bzw. bearbeiten Sie die Datei wie folgt im unteren Bereich:

```
dtparam=spi=on  
dtoverlay=spi-bcm2835-overlay  
dtoverlay=spi-dma-overlay  
dtoverlay=mcp2515-can0-overlay,oscillator=2000000,interrupt=25,spimaxfrequency=1000000  
  
dtoverlay=w1-gpio-pullup
```

Das Ergebnis sollte wie folgt aussehen:

```
GNU nano 2.2.6 File: /boot/config.txt
#dtoverlay=lirc-rpi
# Additional overlays and parameters are documented /boot/overlays/README
# Enable audio (loads snd_bcm2835)
#dtparam=audio=on
dtparam=i2c_arm=on
dtparam=spi=on
dtoverlay=spi-bcm2835-overlay
dtoverlay=spi-dma-overlay
dtoverlay=mcp2515-can0-overlay,oscillator=2000000,interrupt=25,spimaxfrequency=1000000
dtoverlay=w1-gpio-pullup
#core_freq=500
#sdram_freq=500
#over_voltage=2
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^U Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Speichern Sie anschliessend die Datei mit STRG-X und bestätigen Sie die Änderungen mit „J“ bzw. „Y“.

Jetzt muss eine Datei angelegt werden, die dem CODESYS-System die Parameter der CAN-Bus-Kommunikation mitteilt. Das tun wir mit dem Befehl `nano /home/pi/rts_set_baud.sh`.

Achtung! Bei Verwendung der neuen Codesys-Runtime 3.5.11 muss sich die Datei im Verzeichnis `/var/opt` befinden! Dem entsprechend lautet der Befehl hier: `nano /var/opt/rts_set_baud.sh`

Es öffnet sich nun eine leere Datei, in der wir Folgendes einfügen:

```
#!/bin/sh
BITRATE=`expr $2 \\* 1000`
ifconfig $1 down
echo Interface $1 set to $BITRATE kbps
ip link set $1 type can bitrate $BITRATE
ifconfig $1 up
```

Danach wieder mit STRG-X speichern und mit „J“ bzw. „Y“ bestätigen. Anschließend mit „`sudo reboot`“ den Pi neustarten. Damit sind alle Einrichtungsaufgaben erledigt und Sie können das RasPiLC-System in Betrieb nehmen. Wir empfehlen, den Raspberry Pi nicht zu übertakten, weil dadurch das Timing des SPI-Busses geändert wird und die CAN-Kommunikation möglicherweise beeinträchtigt wird.