

1 Netzwerkeinrichtung und Temperatursensor im RB Pi

1.1 Netzwerkkonfiguration

Zunächst wurde die Verbindung zwischen Raspberry (RB) und Laptop (Mac OS) hergestellt und der erfolgreiche Verbindungsaufbau auf Wireshark beobachtet. Hierfür wurde das Netzwerk über den Ethernetport geteilt.

Folgend wurde folgender Befehl in der Konsole ausgeführt (Konfiguration aus Mac OS) um die eigene (Laptop) und die RB-IP-Adresse zu ermitteln:

- **ifconfig** → gibt die Netzwerkkonfiguration aus:

Informationen für Laptop:

```
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
```

```
ether a4:d1:8c:d3:dd:46
```

```
inet6 fe80::73:3dcd:fb18:f50f%en0 prefixlen 64 secured scopeid 0x5
```

```
inet 172.17.9.117 netmask 0xfffff800 broadcast 172.17.15.255
```

```
nd6 options=201<PERFORMNUD,DAD>
```

```
media: autoselect
```

```
status: active
```

Informationen für RB:

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
```

```
inet 169.254.10.178 netmask 255.255.0.0 broadcast 169.254.255.255
```

```
inet6 fe80::9849:f628:5588:c59d prefixlen 64 scopeid 0x20<link>
```

```
ether b8:27:eb:f1:34:ab txqueuelen 1000 (Ethernet)
```

```
RX packets 591 bytes 88527 (86.4 KiB)
```

```
RX errors 0 dropped 44 overruns 0 frame 0
```

```
TX packets 1056 bytes 74726 (72.9 KiB)
```

```
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Über Wireshark konnte der DHCP Handshake beobachtet werden. Nachdem dieser erfolgreich abgeschlossen war, konnte der RB mit seiner zugewiesenen IP-Adresse, über den Befehl:

ping 169.254.10.178

gepingt werden.

Ausgabe:

```
PING 169.254.10.178 (169.254.10.178): 56 data bytes
64 bytes from 169.254.10.178: icmp_seq=0 ttl=64 time=0.474 ms
64 bytes from 169.254.10.178: icmp_seq=1 ttl=64 time=0.438 ms
64 bytes from 169.254.10.178: icmp_seq=2 ttl=64 time=0.611 ms
64 bytes from 169.254.10.178: icmp_seq=3 ttl=64 time=0.463 ms
64 bytes from 169.254.10.178: icmp_seq=4 ttl=64 time=0.404 ms
64 bytes from 169.254.10.178: icmp_seq=5 ttl=64 time=0.522 ms
--- 169.254.10.178 ping statistics ---
6 packets transmitted, 6 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.404/0.485/0.611/0.067 ms
```

Als nächster Schritt wurde die RBkonfiguration angepasst, so dass der RB eine direkte Netzwerkverbindung, ohne DHCP Handshake, mit dem Laptop aufbauen konnte. Um Zugriff auf den RB zu erhalten, wurde der Konsolenbefehl:

ssh pi@169.254.10.178

verwendet. Die Verbindungsinformationen wurden bei dem Aufruf **ifconfig** (s.o.) abgelesen.

MAC-Adresse: b8:27:eb:f1:34:ab

Interfacename: eth0

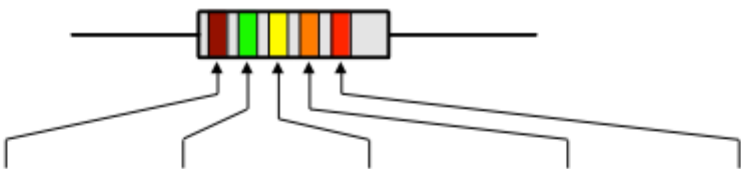
Folgende Befehle wurden auf dem RB ausgeführt:

- **micro /etc/dhcpd.conf** → Öffnen des Micro – Editors
 - interface name geändert:
Änderung des Interfacenamen geändert zu „grp6“. Dadurch war kein statischer Verbindungsaufbau zum RB mehr möglich.
Anschließend wurde „grp6“ wieder zu „eth0“ zurückbenannt. Ein statischer Verbindungsaufbau war nun möglich.
 - static ip addr RB geändert zu 10.6.1.2/24,l
 - standardgateway geändert zu 10.6.1.1
- **sudo reboot** → Neustart des RB

Nach dem Neustart konnte die SSH – Verbindung mit der neuen IP-Adresse erfolgreich hergestellt werden. Damit ist die Konfiguration des RB zunächst abgeschlossen.

1.2 Einrichtung des Temperatursensors

Die physische Installation wurde nach Anleitung durchgeführt



Farbe	1.Ring: 1.Wertziffer	2.Ring: 2.Wertziffer	3.Ring: 3.Wertziffer	4.Ring: Multiplikator	5.Ring: Toleranz
farblos	-	-	-	-	-
silber	-	-	-	$\times 10^{-2}$	-
gold	-	-	-	$\times 10^{-1}$	-
schwarz	-	0	0	$\times 10^0$	-
braun	1	1	1	$\times 10^1$	$\pm 1\%$
rot	2	2	2	$\times 10^2$	$\pm 2\%$
orange	3	3	3	$\times 10^3$	-
gelb	4	4	4	$\times 10^4$	-
grün	5	5	5	$\times 10^5$	$\pm 0,5\%$
blau	6	6	6	$\times 10^6$	-
violett	7	7	7	$\times 10^7$	-
grau	8	8	8	$\times 10^8$	-
weiß	9	9	9	$\times 10^9$	-

Abbildung 1: Farbcodierungstabelle 5-Ringe-Widerstand (Quelle:
https://www.google.com/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwilu_G824biAhXC_GewKHU8GCrEQjRx6BAgBEAU&url=http%3A%2F%2Felektronik-kurs.net%2Ffelektrotechnik%2Ffarbcode-toleranzen-normreihen-leistung%2F&psig=AOvVaw3UrbPq1tIRMRyhNGvm4Mx9&ust=1557225576640034)

Anhand von Abb. 1, wurde der passende Widerstand gewählt. In diesem Fall sollte der Widerstandwert 4,7 k Ω betragen. Folgende Farben sind auf dem Widerstand zu erkennen:

Grundwerte Widerstand (Ring 1-3):

Gelb = 4

Violett = 7

Schwarz = 0

➔ Daraus ergibt sich die Zahl 470 Ω

Multiplikator (Ring 4):

Braun = 10^1

➔ Der Wert 470 Ω * 10^1 = 4700 Ω = 4,7 k Ω

Toleranz (Ring 5):

Braun = $\pm 1\%$

Die softwareseitige Installation des Temperatursensors wurde mit folgenden Befehlen durchgeführt:

- **sudo nano /boot/config.txt** → öffnet die Boot-Konfiguration mit dem Editor „nano“
Am Ende der Datei die Zeile „dtoverlay=w1-gpio“ eingefügt
- **sudo reboot** → Neustart des RB
- **sudo modprobe w1-gpio** → Kernelmodul laden:
- **sudo modprobe w1-therm** → Kernelmodul laden
- **lsmod | grep w1** → listet Kernelmodule auf und filtert nach Modulen die „w1“ beinhalten:
Die Module w1-gpio und w1-therm wurden angezeigt, somit war die Installation der Kernelmodule erfolgreich.
- **micro /etc/modules** → zeigt eine Liste mit den Konfigurationen der geladenen Kernelmodulen:
Hier wurden die Module w1-gpio und w1-therm eingetragen.
- **cd /sys/bus/w1/devices/** → in das Verzeichnis der w1 Module wechseln
ls → Ordner/Dateien anzeigen lassen
cd 28-01143d2359aa → in das Verzeichnis 28-01143d2359aa wechseln
cat w1_slave → formatierte Ausgabe der Messwerte des Temperatursensors:
In der zweiten Zeile der Ausgabe wurde die aktuelle Temperatur angezeigt.

Da das Verzeichnis 28-01143d2359aa automatisch erstellt wurde und somit auch die aktuelle Temperatur ausgegeben werden konnte, ist die Einrichtung des RB vorläufig abgeschlossen.