

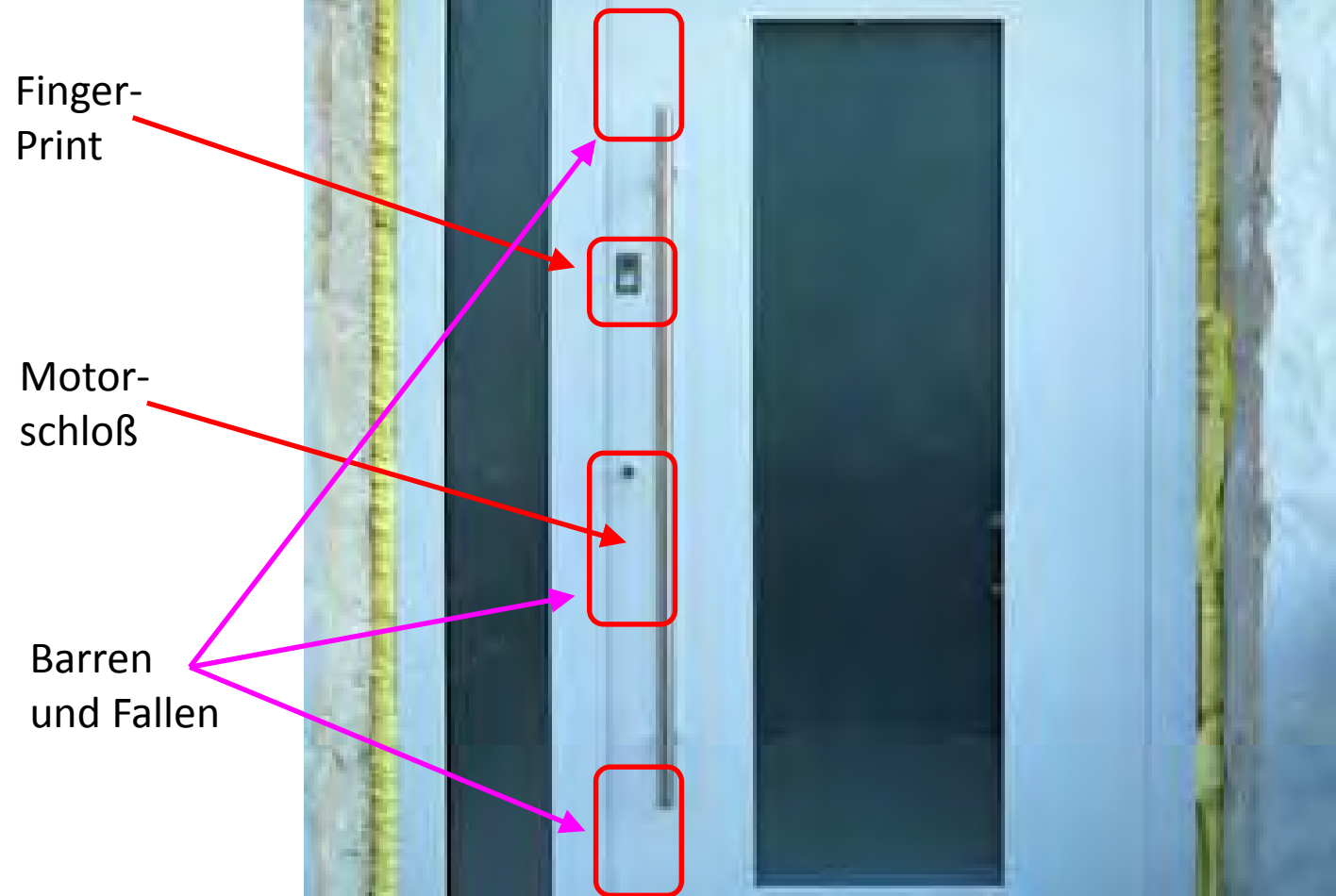
IP Symcon Stammtisch: Im Umkreis Nürnberg, Fürth und Erlangen

15. April 2015 ab 17:30 - 18:00 Uhr bis ca. 20:00 - 21Uhr?

<http://www.hotel-restaurant-erlangen.de/>

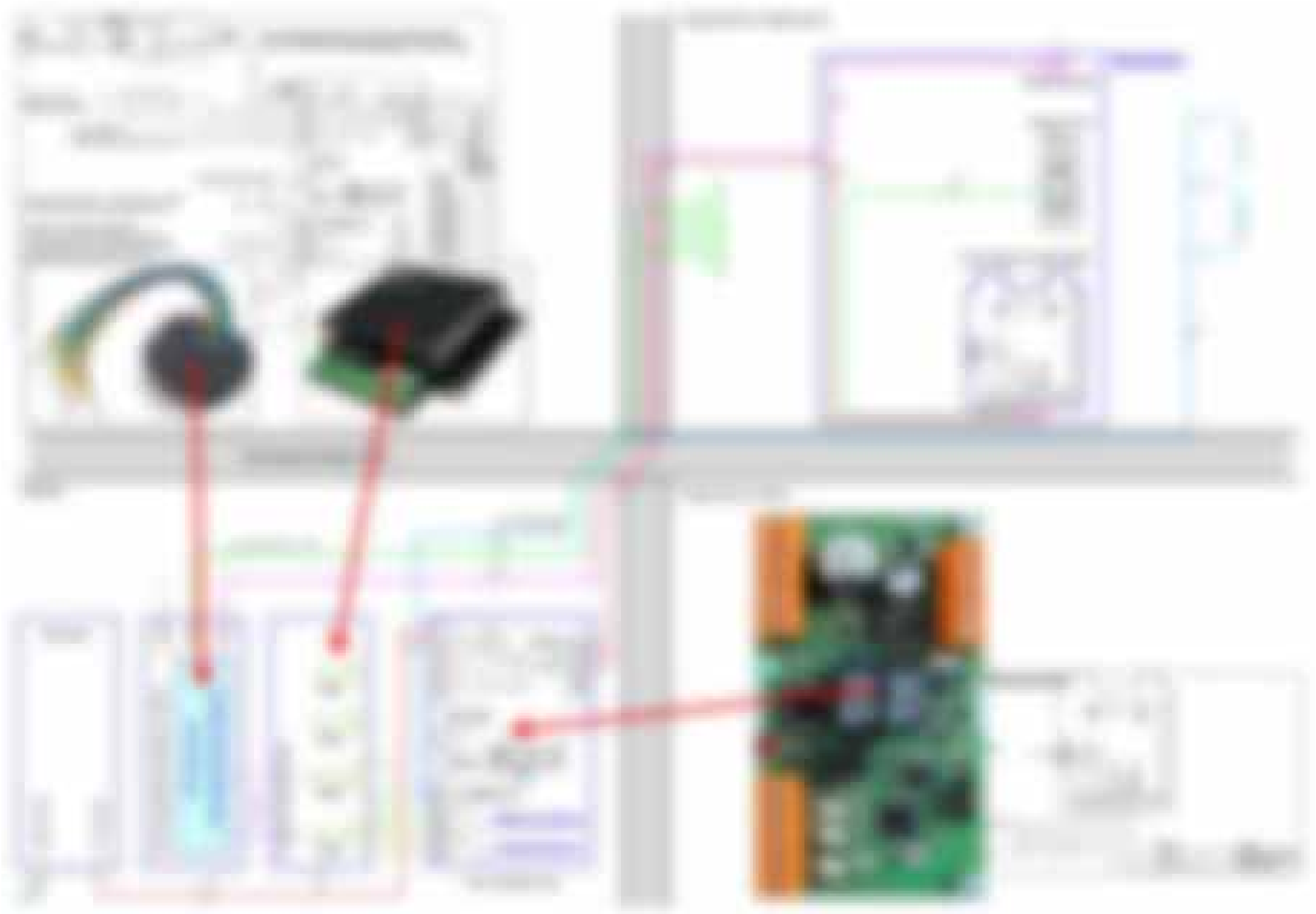
1. Haustürautomatisierung mit Fuhr Motorschloss und Fingerprint
2. Lichtstärkemessung mit HomeMatic Piri's
3. IPS auf RASPi mit JSON-Schnittstelle zu einem Windows-IPS
4. Temperaturerfassung mit 1-Wire auf RASPi ohne zusätzlichen 1-Wire Adapter
5. Ölstandsmessung mit RASPi auf Ultraschallbasis (*leider noch sehr experimentell*)

1.1 Haustürautomatisierung mit Fuhr Motorschloß und Fingerprint



1.2 Haustürautomatisierung mit Fuhr Motorschloss und Fingerprint



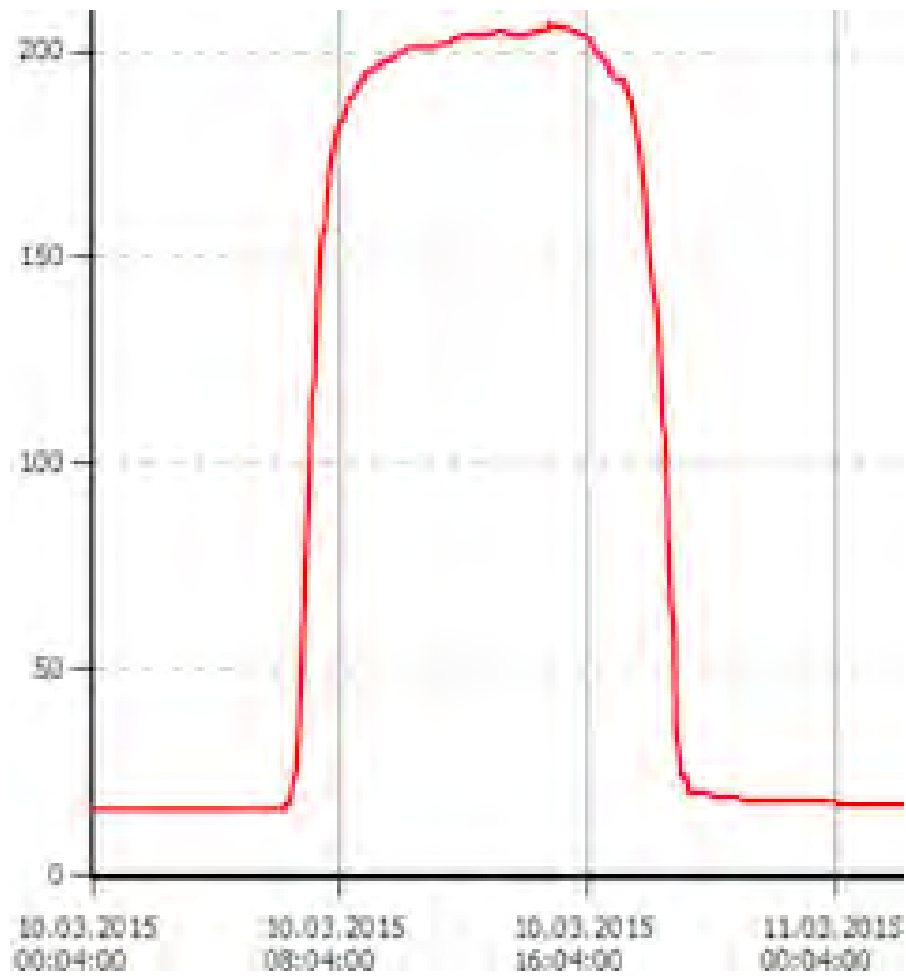


1.4 Haustürautomatisierung mit Fuhr Motorschloss und Fingerprint

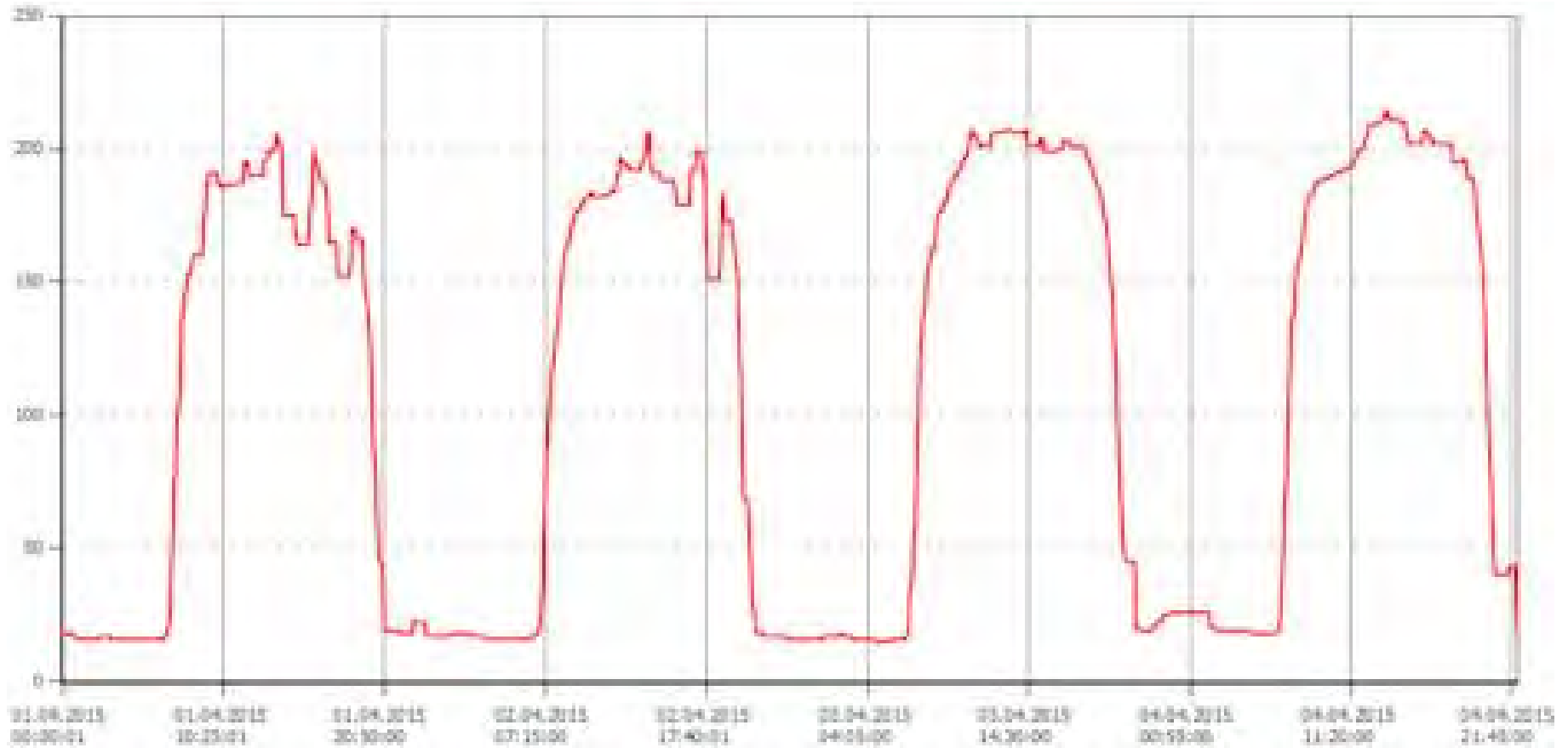
- Variablen und Scripts



2.1 Lichtstärkemessung mit HomeMatic Piri's



2.2 Lichtstärkemessung mit HomeMatic Piri's



2.3 Lichtstärkemessung mit HomeMatic Piri's

PIRI				
PIRIDrehbar	HomeMatic Device			N/A
BRIGHTNESS	Integer	18		22:08:10
INSTALL_TEST	Boolean	True		19:52:23
MOTION	Boolean	Untätig		19:52:58
PIRI_Status	Boolean	False		19:54:13

3.1 IPS auf RASPi mit JSON-Schnittstelle zu einem Windows-IPS

Windows PC



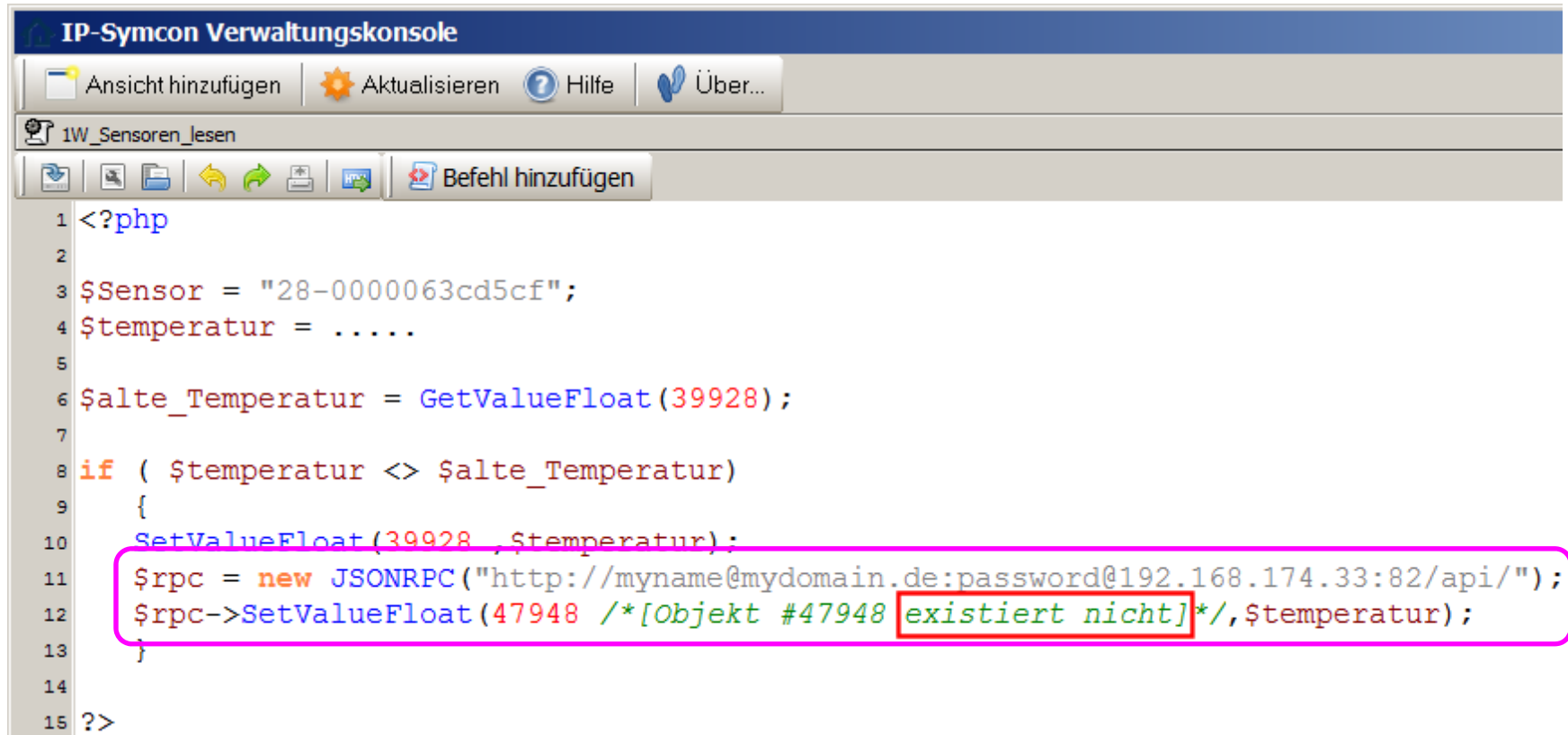
Raspberry PI



ObjektID	Standort\Objekt	Typ	Wert	Aktualisiert
	Home			
	Fussbodenheizung			
	NachMischer	Float	29,0 °C	22:32:06
	RL_Bad	Float	25,0 °C	22:32:02
	RL_Christina	Float	24,9 °C	22:29:04
	RL_Flur	Float	21,1 °C	22:32:01
	RL_Kueche_oben	Float	24,1 °C	22:29:03
	RL_Michael	Float	23,2 °C	22:30:00
	RL_Sauna	Float	26,8 °C	22:31:05
	Ruecklauf	Float	26,9 °C	22:31:07
	VorMischer	Float	30,3 °C	22:32:05
	GoogleChart			
	HomeMatic			

ObjektID	Standort\Objekt	Typ	Wert	Aktualisiert
	IP-Symcon			
	Fussbodenheizung			
	NachMischer	Float	29,0 °C	22:32:07
	RL_Bad	Float	25,0 °C	22:32:02
	RL_Christina	Float	24,9 °C	22:29:04
	RL_Flur	Float	21,1 °C	22:32:02
	RL_Kueche	Float	24,1 °C	22:29:03
	RL_Michael	Float	23,2 °C	22:30:01
	RL_Sauna	Float	26,8 °C	22:31:05
	Ruecklauf	Float	26,9 °C	22:31:08
	VorMischer	Float	30,3 °C	22:32:06
	1W_Sensoren_lesen	Skript	45194.ips.php	22:32:08
	Ereignis: Täglich alle ...	Ereignis		22:32:00

3.2 IPS auf RASPi mit JSON-Schnittstelle zu einem Windows-IPS



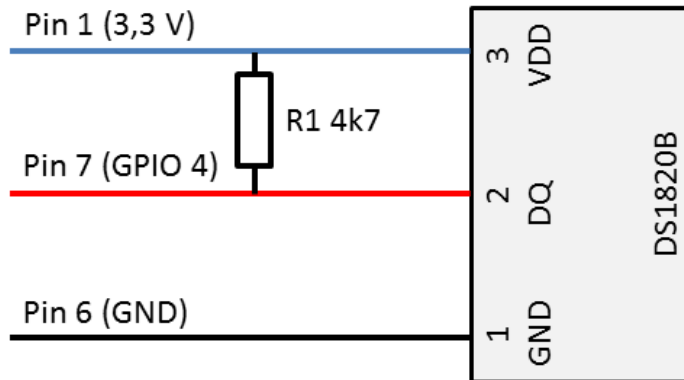
```
IP-Symcon Verwaltungskonsole
Ansicht hinzufügen Aktualisieren Hilfe Über...
1W_Sensoren_lesen
Befehl hinzufügen
1 <?php
2
3 $Sensor = "28-0000063cd5cf";
4 $temperatur = .....
5
6 $alte_Temperatur = GetValueFloat(39928);
7
8 if ( $temperatur <> $alte_Temperatur)
9 {
10     SetValueFloat(39928, $temperatur);
11     $rpc = new JSONRPC("http://myname@mydomain.de:password@192.168.174.33:82/api/");
12     $rpc->SetValueFloat(47948 /*[Objekt #47948 existiert nicht]*/, $temperatur);
13 }
14
15 ?>
```

4.1 Temperaturerfassung mit 1-Wire auf RASPi ohne zusätzlichen 1-Wire Adapter

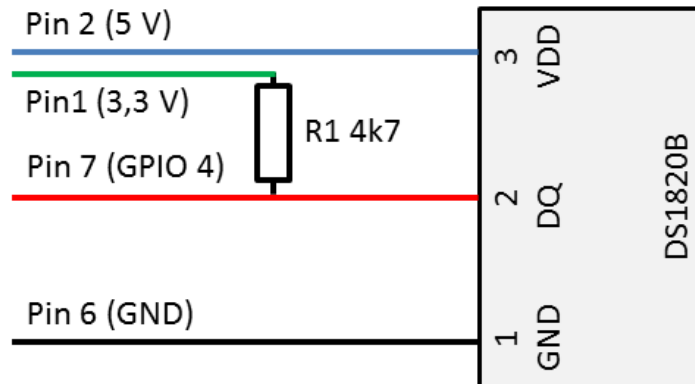


Pi Model B/B+			
3V3 Power	1	2	5V Power
GPIO2 SDA1 I2C	3	4	5V Power
GPIO3 SCL1 I2C	5	6	Ground
GPIO4	7	8	GPIO14 UART0_TXD
Ground	9	10	GPIO15 UART0_RXD
GPIO17	11	12	GPIO18 PCM_CLK
GPIO27	13	14	Ground
GPIO22	15	16	GPIO23
3V3 Power	17	18	GPIO24
GPIO10 SPI0_MOSI	19	20	Ground
GPIO9 SPI0_MISO	21	22	GPIO25
GPIO11 SPI0_SCLK	23	24	GPIO8 SPI0_CE0_N
Ground	25	26	GPIO7 SPI0_CE1_N
ID_SD I2C ID EEPROM	27	28	ID_SC I2C ID EEPROM
GPIO5	29	30	Ground
GPIO6	31	32	GPIO12
GPIO13	33	34	Ground
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
Ground	39	40	GPIO21

Für **kurze** Leitungen
(parasitäre Versorgung)



Für **längere** Leitungen



4.2 Temperaturerfassung mit 1-Wire auf RASPi ohne zusätzlichen 1-Wire Adapter

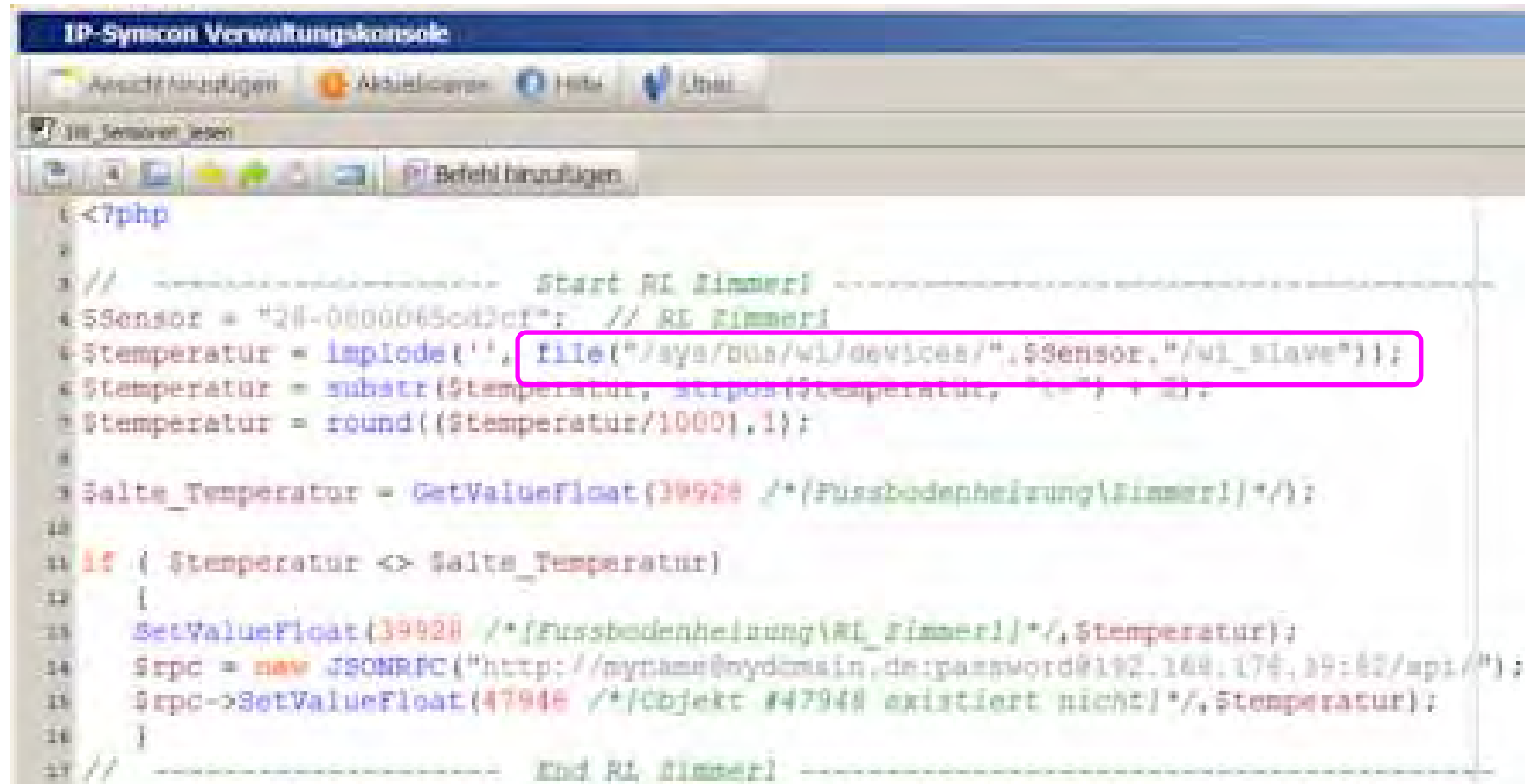
Notwendige Kernel Module

```
sudo nano /etc/modules  
w1-gpio pullup=1  
w1-therm
```

1-Wire Filesystem auf dem RASP

```
cd /sys/bus/w1/devices  
  
ls -l  
28-0000065bff68 28-0000065c1c55 ... ..  
  
cd 28-0000065bff68  
  
cat w1_slave  
b8 01 4b 46 7f ff 08 10 8a : crc=8a YES  
b8 01 4b 46 7f ff 08 10 8a t=27500
```

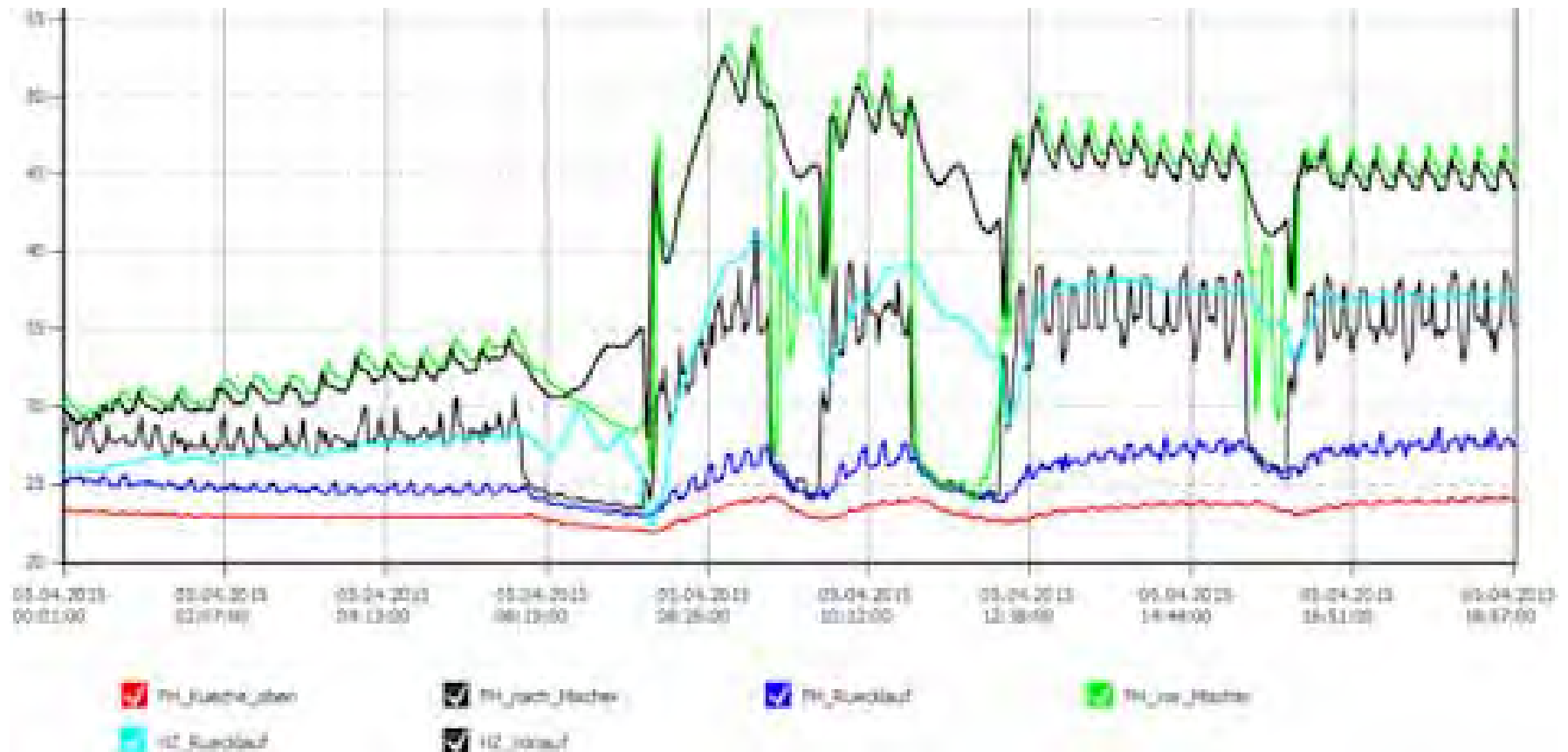
4.3 Temperaturerfassung mit 1-Wire auf RASPi ohne zusätzlichen 1-Wire Adapter



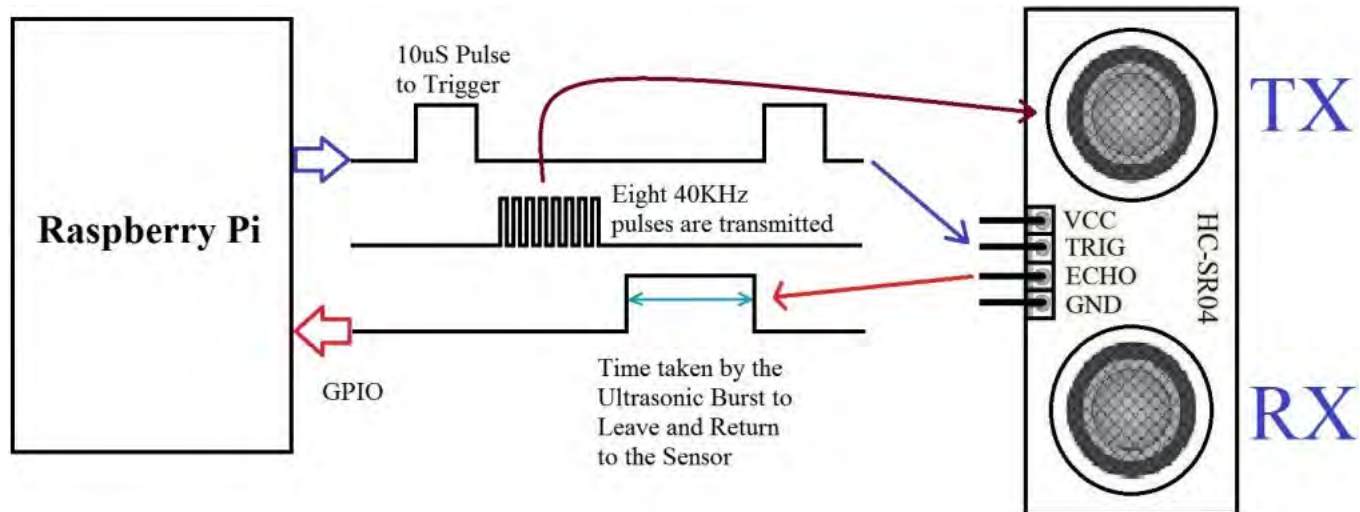
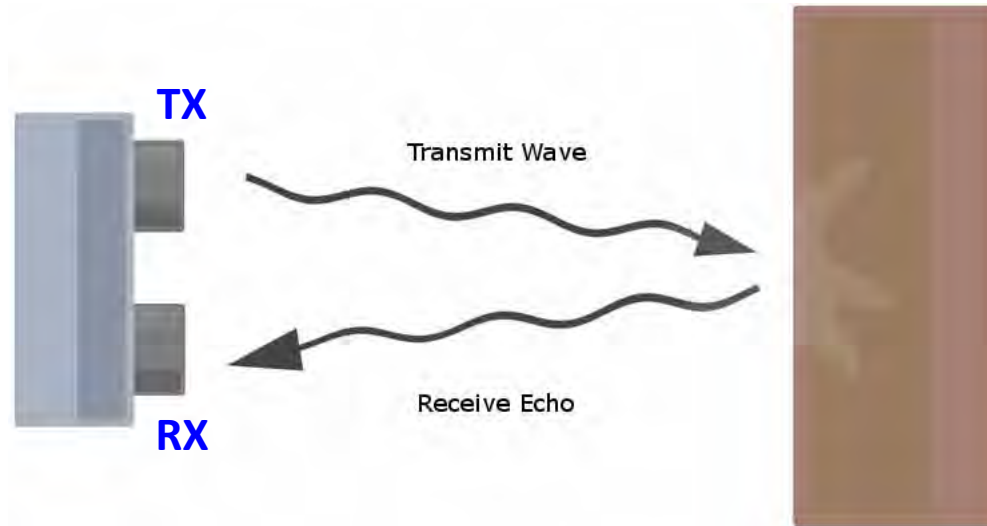
The screenshot shows the IP-Symcon Verwaltungskonsolle interface. The main content area displays PHP code for detecting a 1-Wire sensor. A red box highlights the `file()` function call used to check for the presence of the sensor.

```
1 <?php
2
3 // ----- Start RI_Simmerl -----
4 $Sensor = "28-0000065cd3ef"; // RI_Simmerl
5 $temperatur = implode('', file("/sys/bus/wi/devices/", $Sensor, "/w1_slave"));
6 $temperatur = substr($temperatur, strpos($temperatur, "t=") + 2);
7 $temperatur = round(($temperatur/1000), 1);
8
9 $alte_Temperatur = GetValueFloat(39928 /*[Fußbodenheizung\Simmerl]*/);
10
11 if ( $temperatur <> $alte_Temperatur)
12 {
13     SetValueFloat(39928 /*[Fußbodenheizung\RI_Simmerl]*/, $temperatur);
14     $rpc = new JSONRPC("http://myname@oydomain.de:password@192.168.178.19:62/api/");
15     $rpc->SetValueFloat(47948 /*[Objekt #47948 existiert nicht]*/, $temperatur);
16 }
17 // ----- End RI_Simmerl -----
```

4.4 Temperaturerfassung mit 1-Wire auf RASPi ohne zusätzlichen 1-Wire Adapter



5.1 Ölstandsmessung mit RASPi auf Ultraschallbasis (leider noch experimentell)



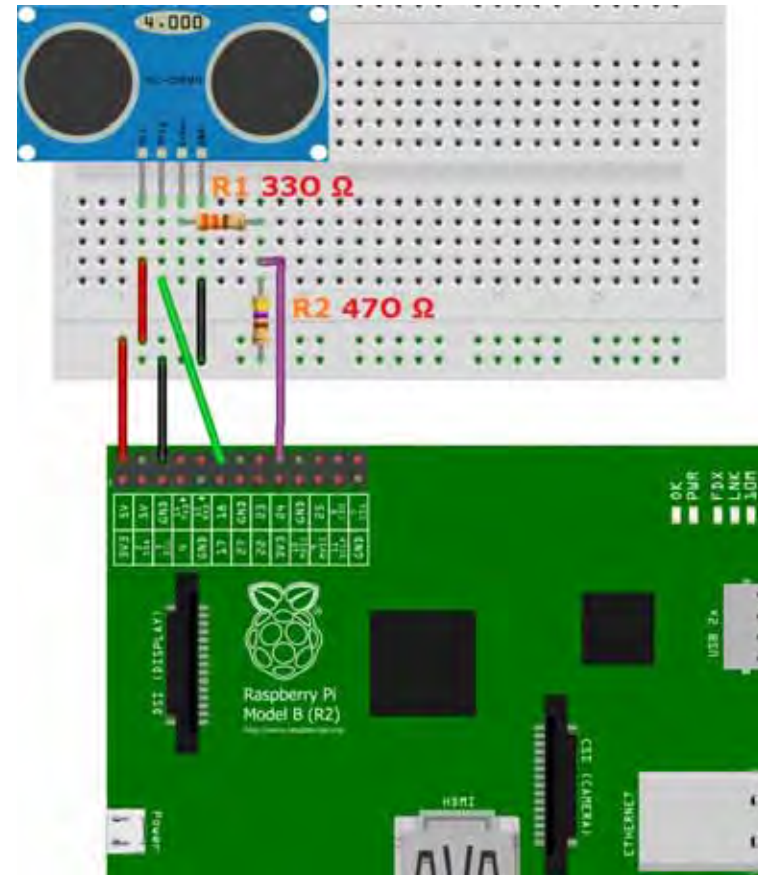
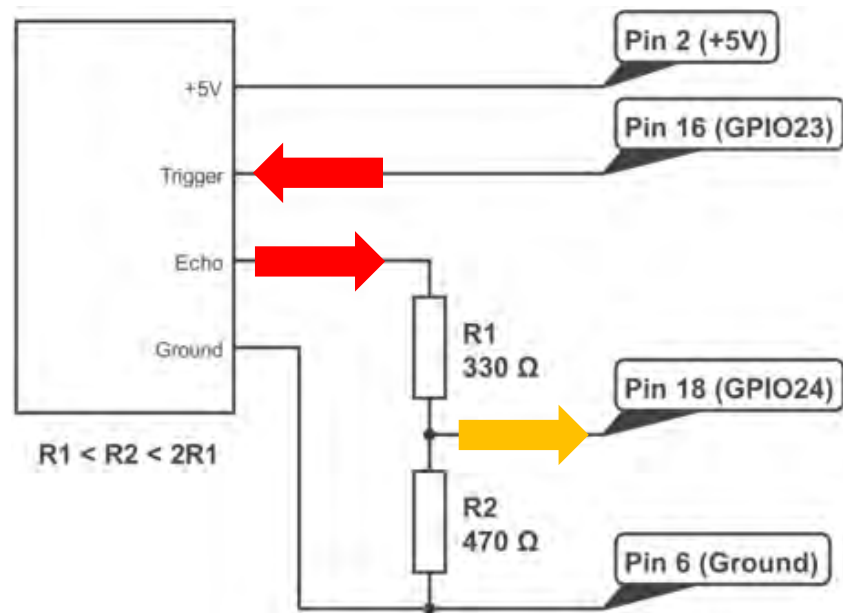
5.2 Ölstandsmessung mit RASPi auf Ultraschallbasis (leider noch experimentell)

HC-SR04



DYP-ME007Y

5.3 Ölstandsmessung mit RASPi auf Ultraschallbasis (leider noch experimentell)



5.4 Ölstandsmessung mit RASPi auf Ultraschallbasis (leider noch experimentell)

```
#GPIO Pins zuweisen
```

```
GPIO_TRIGGER = 23
```

```
GPIO_ECHO = 24
```

```
#Richtung der GPIO-Pins festlegen (IN / OUT)
```

```
GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
```

```
GPIO.setup(GPIO_ECHO, GPIO.IN)
```

```
def distanz():
```

```
    # setze Trigger auf HIGH
```

```
    GPIO.output(GPIO_TRIGGER, True)
```

```
    # setze Trigger nach 0.01ms aus LOW
```

```
    time.sleep(0.00001)
```

```
    GPIO.output(GPIO_TRIGGER, False)
```

```
    ....>
```

```
    StartZeit = time.time()
```

```
    StopZeit = time.time()
```

```
    # speichere Startzeit
```

```
    while GPIO.input(GPIO_ECHO) == 0:
```

```
        StartZeit = time.time()
```

```
    # speichere Ankunftszeit
```

```
    while GPIO.input(GPIO_ECHO) == 1:
```

```
        StopZeit = time.time()
```

```
    # Zeit Differenz zwischen Start und Ankunft
```

```
    TimeElapsed = StopZeit - StartZeit
```

```
    # mal der Schallgeschwindigkeit (34300 cm/s)
```

```
    # und durch 2 teilen, da hin und zurueck
```

```
    distanz = (TimeElapsed * 34300) / 2
```

```
    return distanz
```

5.4 Ölstandsmessung mit RASPi auf Ultraschallbasis (leider noch experimentell)

Anzahl

