

Haustechnik steuern
mit
Raspberry Pi

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Gliederung:

- Motivation
- Anlagearten/Anwendungsgebiete Sinnvolle Anwendungen
- Sicherheitsfunktionen/Sicherheitsaspekte, Anlagenschutz, Schutzmaßnahmen, Installationsrichtlinien
- Grundlagen Regler
- Sensoren, Aktoren, I/O-Module
- Vergleich mit professionellen Systemen, nicht mehr so weit auseinander IEC61131
- Anlagenbeispiel, Hydraulik, Schaltplan
- Softwareumsetzung, Module, Sicherheitsfunktionen, Monitoring
- Demonstration Sensor- Aktor
- Live-Monitorring

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Motivation:

- Kosten?
- Funktionen?
- ...

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Anlagenarten/Anwendungsgebiete:

Steuer- und Regelaufgaben ohne zeitkritische Prozesse, aber keine **baurechtlich notwendigen** Sicherheitsanlagen (BMA, SIBE, ...), keine Abnahme **pflichtigen** Anlagen (RWA).

- Heizungssteuerung (ohne Sicherheitskette), Mischerregler, Warmwasserbereitung, Sonnenkollektor-Steuerung, Zeitprogramme
- Lüftungs- Klimaanlage
- Lüftungs- Jalousieanlagen
- Beleuchtungssteuerung
- Bewässerungsanlagen, Gewächshaus-Steuerung
- Steuerung Medientechnik
- ...

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

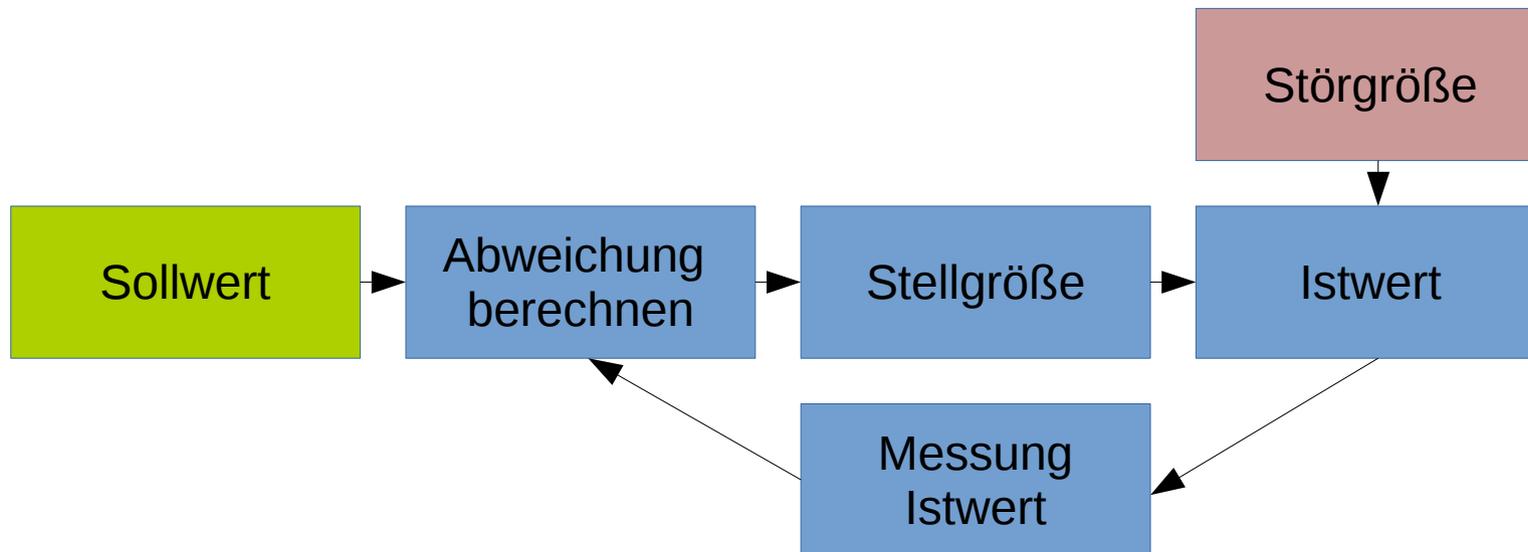
Sicherheitsfunktionen/Sicherheitsaspekte:

- Personenschutz! Anwendung FI-Schutzschalter xxA/0,03A
- Sicherheitsfunktionen in Hardware (STB, Wassermangel, Überdruck, ...)
- Berührungsschutz DIN EN 50 274
- Geprüfte Schaltgeräte für Anlagenteile 230/400V
- Vorsicherung / Leitungsschutz DIN VDE 0100-430
- Handbedienebene

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Regler/Grundlagen:

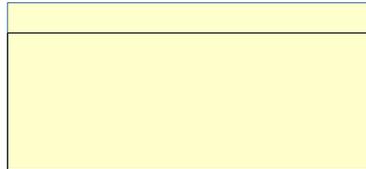


Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

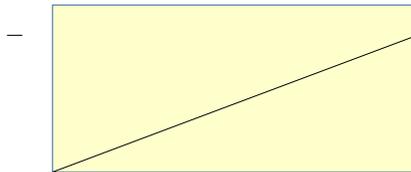
Regler/Grundlagen:

P Regler



$$y = K_p * e$$

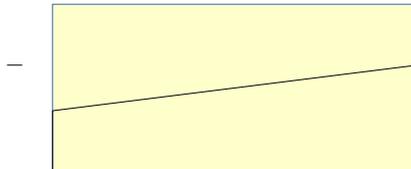
I Regler



$$e_{sum} = e_{sum} + e$$

$$y = K_i * T_a * e_{sum}$$

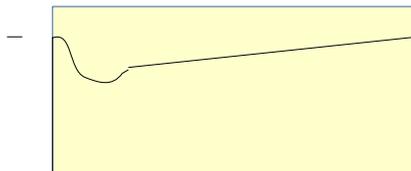
PI Regler



$$e_{sum} = e_{sum} + e$$

$$y = K_p * e + K_i * T_a * e_{sum}$$

PID Regler



$$e_{sum} = e_{sum} + e$$

$$y = K_p * e + K_i * T_a * e_{sum} + K_d * (e - e_{alt})/T_a$$

$$e_{alt} = e$$

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Sensoren, Aktoren und I/O-Module - Ankopplung

- 1Wire Bus

Leitung bis 150m z.B. J-Y(St)Y 2x2x08

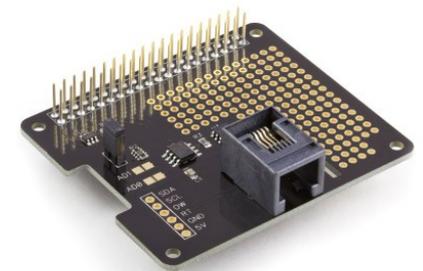
1Master maximal 100 Slaves (typisch bis 20) abhängig vom Stromverbrauch

Vorteile:

- Einfache Verkabelung
- „Feldbus“ → „aktive“ Sensoren und Aktoren entfernt vom Controller
- Einfach zu erweitern, Zuordnung Adresse / Typ

Nachteile:

- Langsam (Beispiel liefert Temperatursensor DS1820 erst ca. nach 15-20s neuen Wert)
- Spannungsversorgung, Stabilität bei vielen Busteilnehmern



Quelle: www.abelectronics.co.uk



Quelle: Maxim/Dallas DS9490R

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Sensoren, Aktoren und I/O-Module - Ankopplung

- I²C

2 Busleitungen SDA (serial data line), SCL (serial clock line)

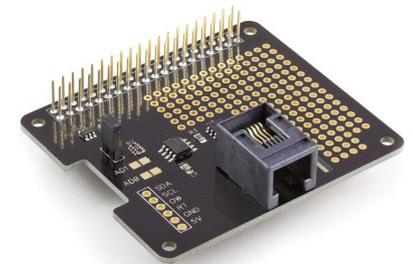
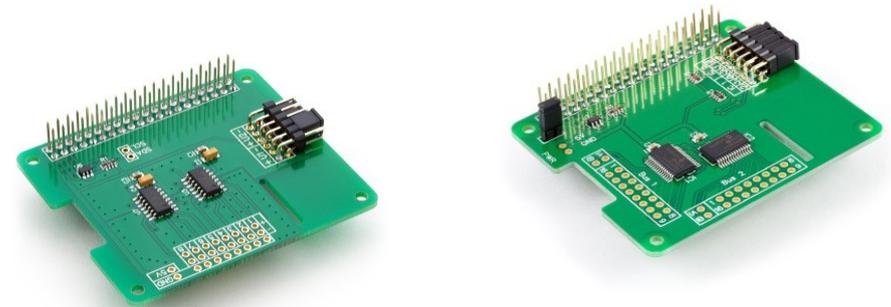
8 bit bidirektional Datentransfer bis zu 100 kbit/s
im Standard Mode

Module für

digital I/O, AD/DA Wandler, PWM ...

Anbindung Sensoren über

Messwandler z.B. PT1000 → AD Wandler



Quelle: www.piface.org.uk

Quelle: www.abelectronics.co.uk

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Sensoren und Aktoren

- 1Wire

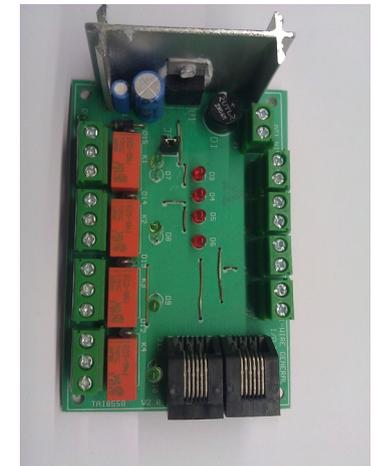
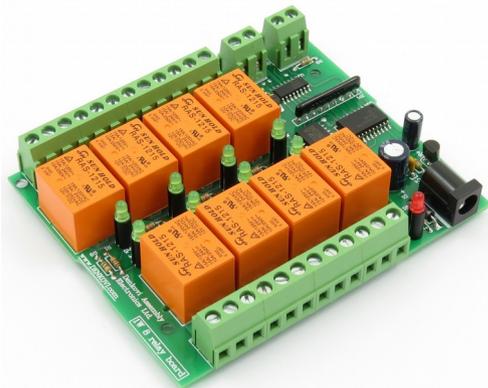
Sensoren:

Temperatur, Feuchte, Luftgüte, Licht, S0-Zähler, DI, AI 0-10V

Aktoren:

Relaiskarten, Dimmer,

Analog Ausgabe 0-10V, PWM



Quelle: Denkovi Assembly Electronics Ltd.

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

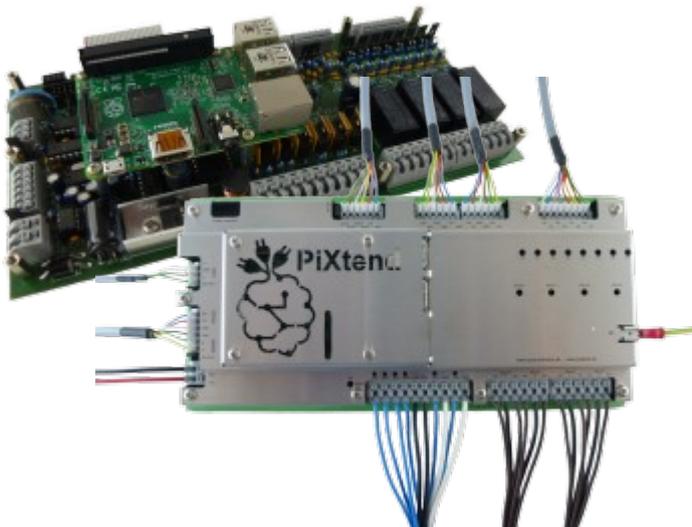
Vergleich mit professionellen Systemen



WAGO PFC 200
 CPU single core Cortex A8 600 MHz
 256 MB interne Flash-Speicher
 128 kB remanenter Speicherbereich
 256 MB RAM
 Betriebssystem Realtime Linux



EATON Modular PLC
 CPU NEC VR 4181 A MIPS 64bit
 MIPS RISC Prozessor 131MHz
 2 MB Programmspeicher
 512 kB Datenspeicher
 32 kB remanenter Speicherbereich

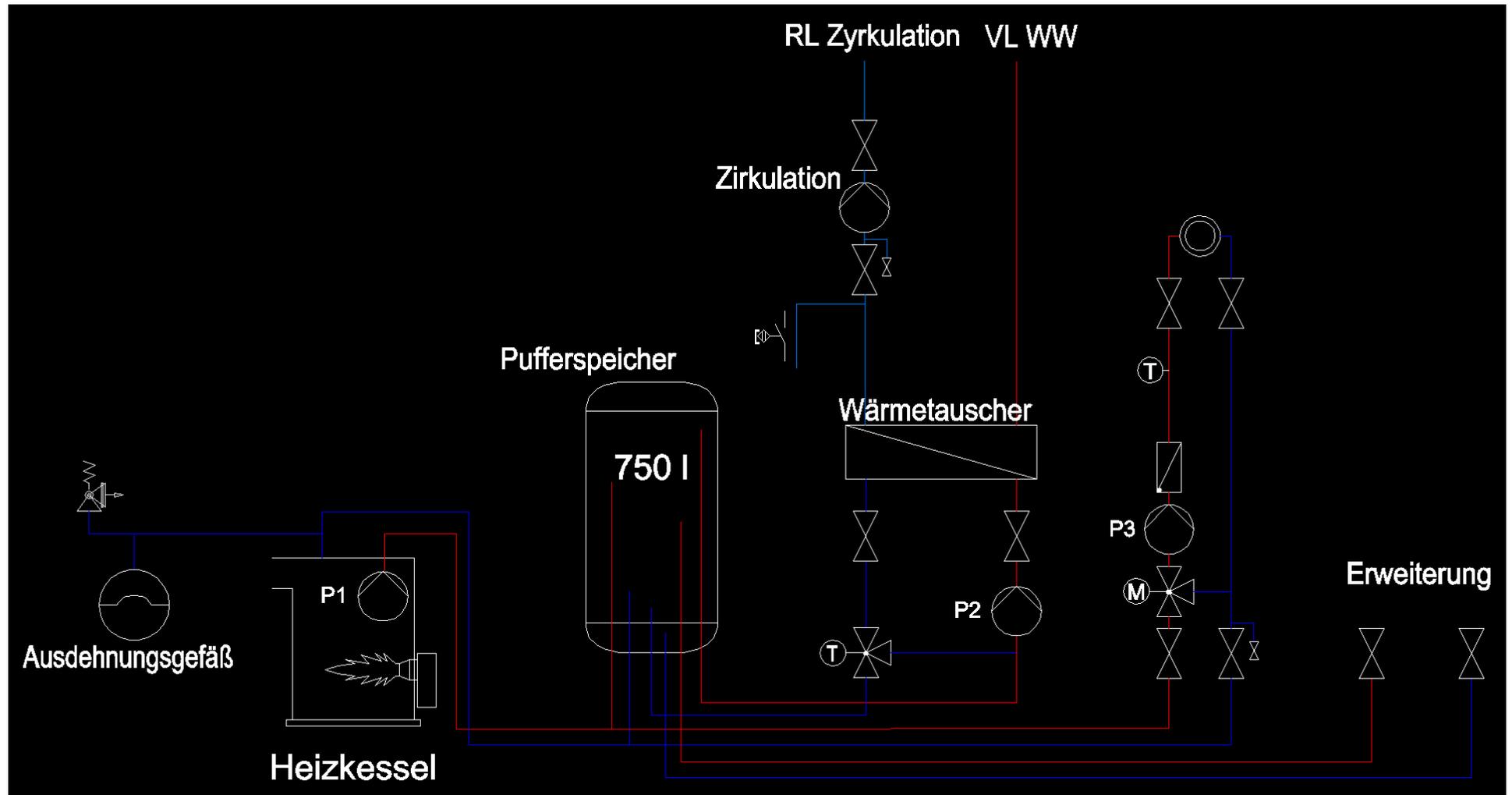


PIXten
 CPU 1,2 GHz Quad-Core
 ARM Cortex-A8 64bit
 SD Programmspeicher
 1GB RAM
 Betriebssystem Linux 32bit, (64bit)

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

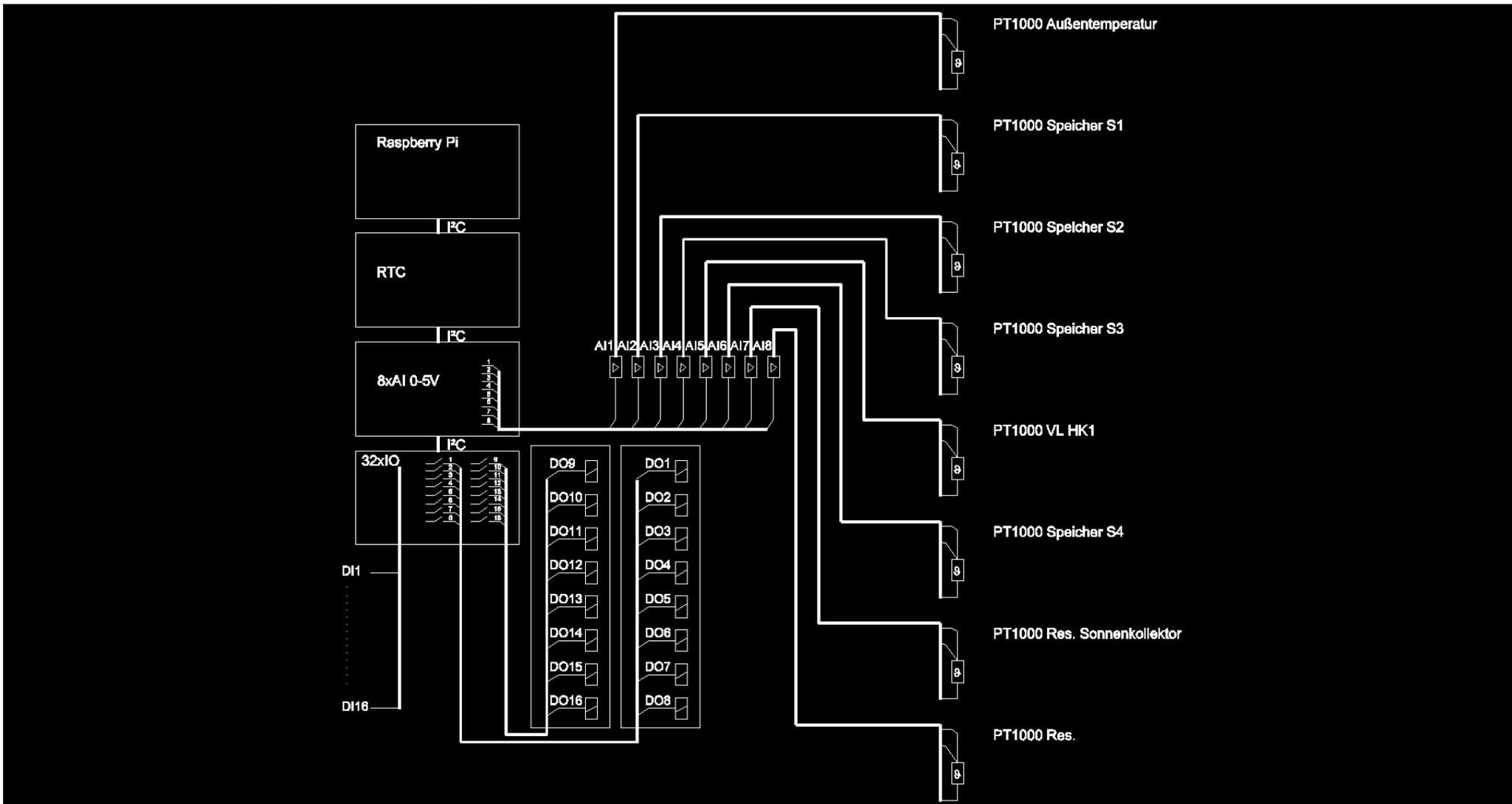
Anlagenbeispiel / Hydraulik



Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

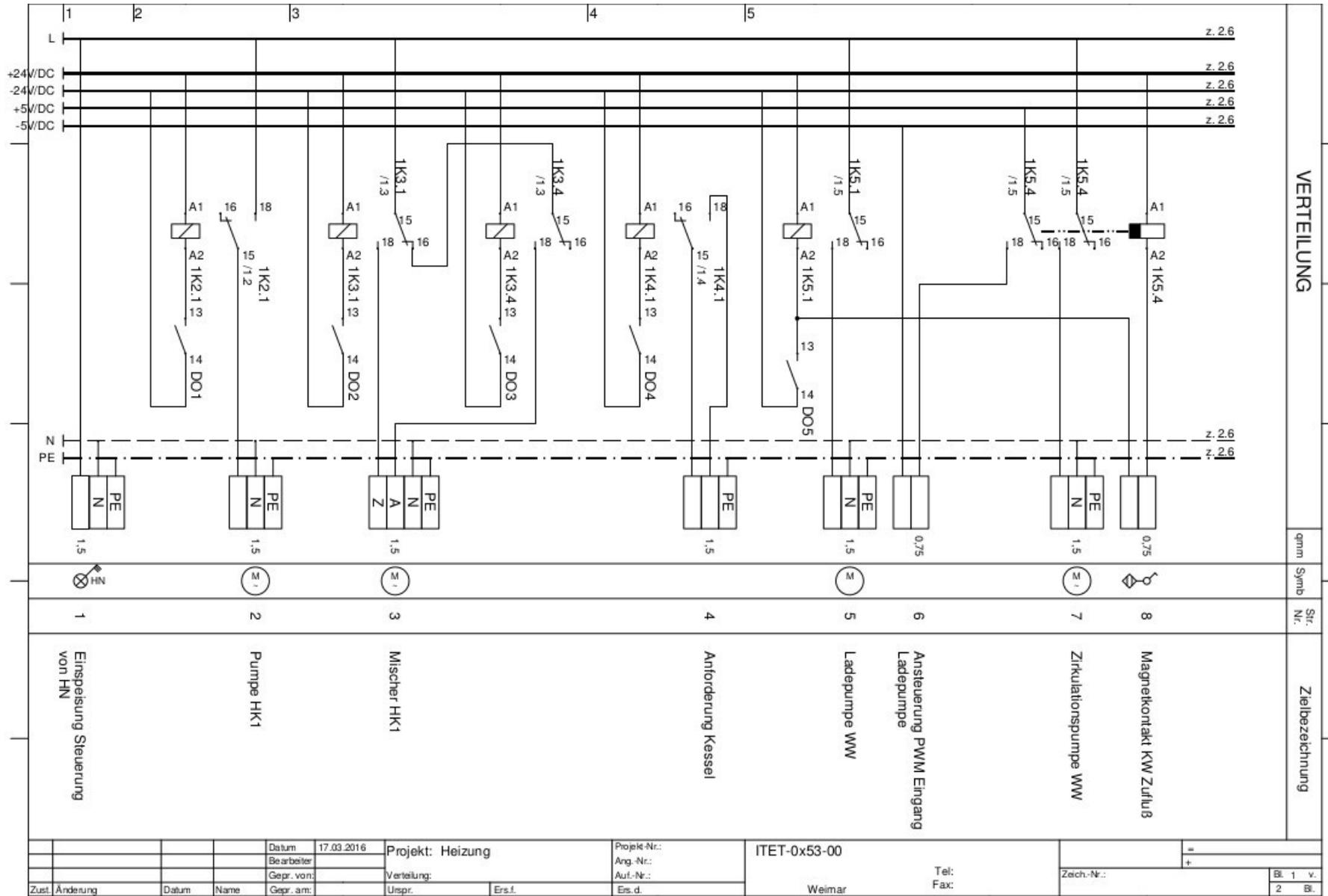
Anlagenbeispiel / Schaltplan



Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Anlagenbeispiel / Schaltplan

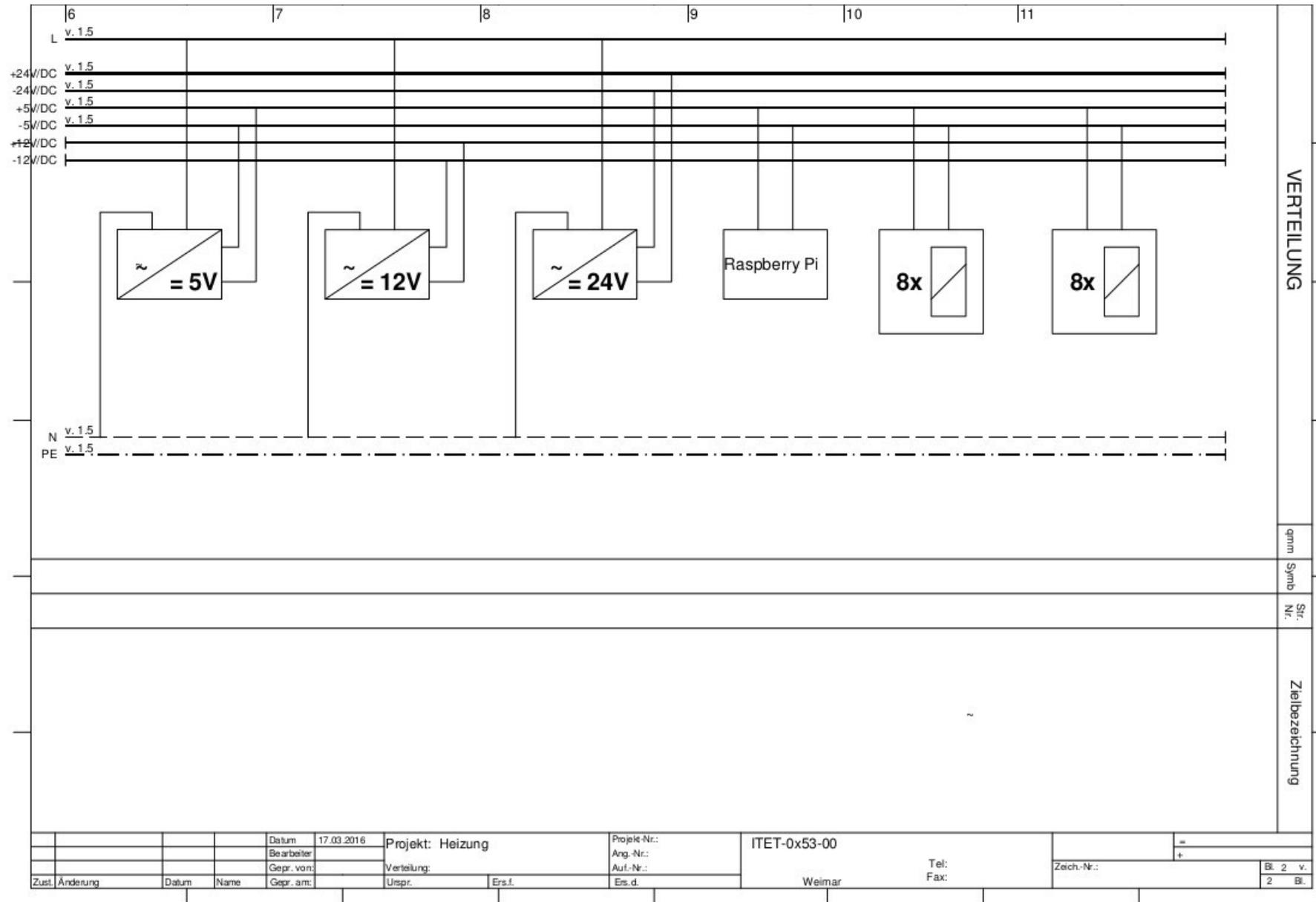


Datum: 17.03.2016		Projekt: Heizung		Projekt-Nr.: ITET-0x53-00		Bl. 1 v.	
Zust. Änderung		Verteilung:		Weimar		Bl. 2 v.	
Datum		Urspr.		Ers.f.		Tel: Fax:	
Name		Ers.d.		Ang.-Nr.:		Zech.-Nr.:	
Gepr. von:		Auf.-Nr.:		Weimar		Bl. 1 v.	
Gepr. am:		Ers.d.		Weimar		Bl. 2 v.	

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

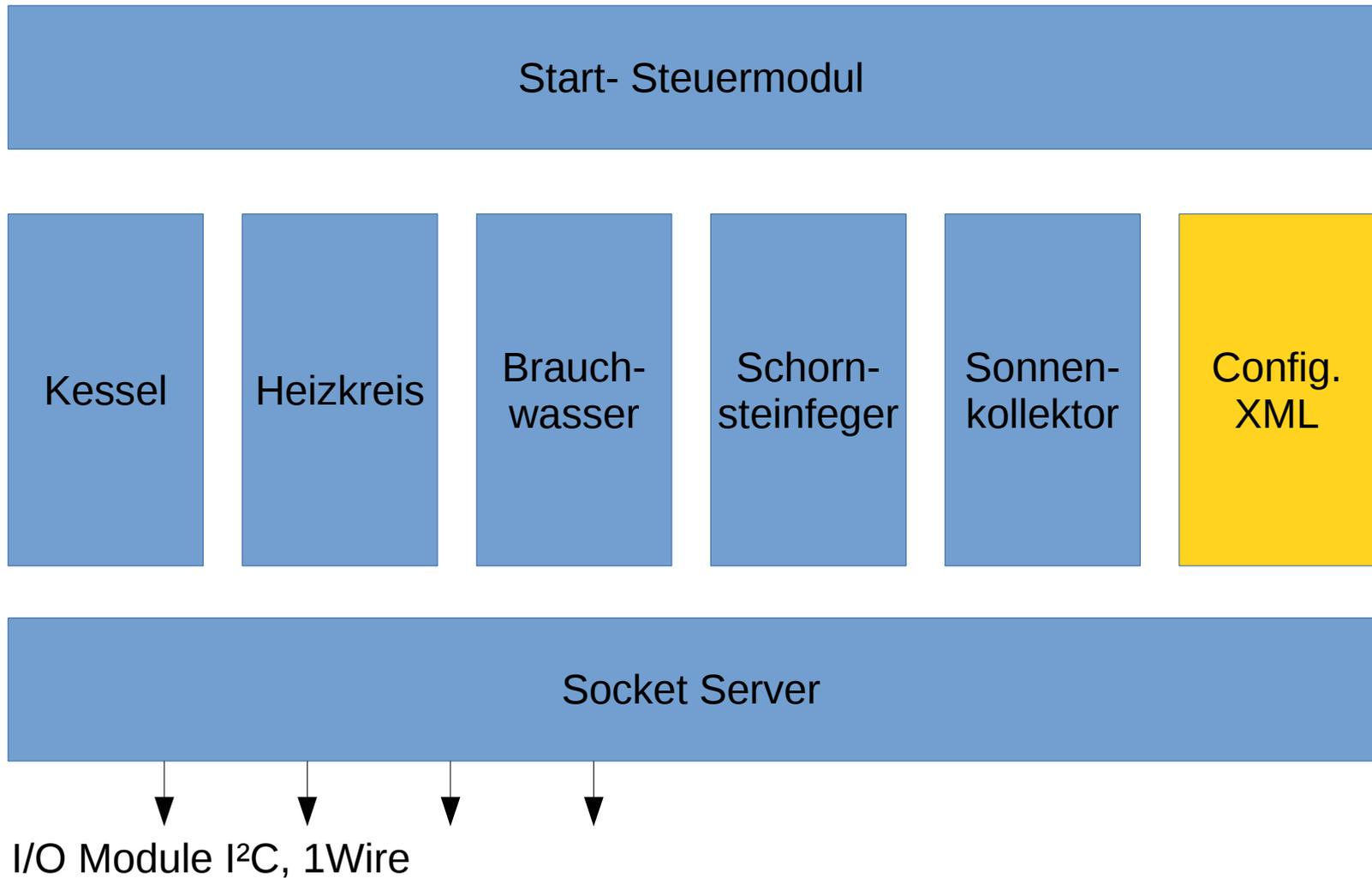
Anlagenbeispiel / Schaltplan



Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Softwaremodule:



Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Konfiguration in Form XML-File:

```
<?xml version="1.0"?>
<Daten_Heizung>
  <config_file>
    <file_name>heizung_conf.xml</file_name>
    <file_version>3</file_version>
    <file_status>aktiv</file_status>
  </config_file>
  <config_variablen_main>
    <!--      definierte Module      -->
    <module>i2c_io_server,kk1,hk1,ww1,di</module>
    <!--      Arbeitsverzeichnis fuer Statusdateien      -->
    <tmp_dir>/tmp/heizung</tmp_dir>
    <!--      OWServer IP:Port ,Standard 127.0.0.1:4304      -->
    <ow_ip_port>localhost:4304</ow_ip_port>
  </config_variablen_main>
  ...
```

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Konfiguration Module:

```

<config_variablen_kk1>
  <!-- Heizung Kesselkreis 1 aktiv -->
  <kk1_aktiv>1</kk1_aktiv>
  <!-- Aussentemperatur bei der Heizung ausgeschalten wird -->
  <a_temp_max_kk1>20</a_temp_max_kk1>
  <!-- Aussentemperatur bei der zwischen Absenkbetrieb und Ecobetrieb umgeschalten wird -->
  <a_temp_frost_kk1>0</a_temp_frost_kk1>
...
  <!-- Socket Adresse IO-Modul Schalter Anforderung Kessel Heizung -->
  <socket_io_adr_out_anforderung_heizung_kessel>do2</socket_io_adr_out_anforderung_heizung_kessel>
...
  <!-- Socket Adresse Außentemperatur -->
  <socket_adr_a_temp_kk1>ai1</socket_adr_a_temp_kk1>
  <!-- Socket Adresse Fuehler1 Pufferspeicher -->
  <socket_adr_s1_temp>ai2</socket_adr_s1_temp>
  <!-- Socket Adresse Fuehler2 Pufferspeicher -->
  <socket_adr_s2_temp>ai3</socket_adr_s2_temp>
...
  <!-- Fuehler1 min Temperatur -->
  <s1_min_temp>57</s1_min_temp>
  <!-- Fuehler2 min Temperatur -->
  <s2_min_temp>48</s2_min_temp>
...
  <!-- Betriebszeiten Tagbetrieb in der Form Startzeit,Endzeit ...,... im Format hhmm,hhmm;hhmm,hhmm ... -->
  <Betriebszeiten_kk1_Mo>0550,1200;1200,2300</Betriebszeiten_kk1_Mo>

```

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Module in Python:

i2c_io_server.py	Socket-Server
kk1.py	Kesselsteuerung
hk1.py	Heizkreissteuerung
ww1.py	Brauchwasser (Zeitsteuerung Freigabe)
di.py	Eingaben z.B. Schornsteinfeger, Störungen, ...

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Anlagenbilder / Demonstration /Live - Monitorring:

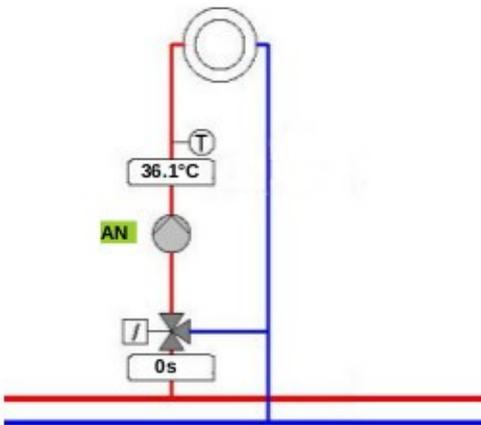


Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

Anlagenbilder / Demonstration /Live - Monitorring:

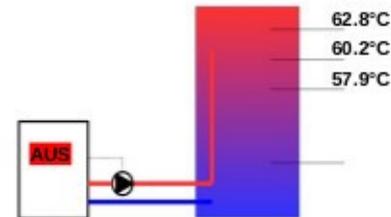
Heizungssteuerung Markt2
Markt 2
Anlagenbild Heizung HK1



Aussentemperatur	10.9°C
Modus	TAG
Aktualisierung	20:50:18
VL-Soll	37.5°C

[Startseite](#)

Heizungssteuerung Markt2
Markt 2
Anlagenbild Heizung KK1



Aussentemperatur	10.9°C
Modus	Tagbetrieb
Aktualisierung	20:49:38
	keine Anforderung

[Startseite](#)

Haustechnik steuern mit Raspberry Pi

ITET-0x53-00

ITET-0x53-00

IT und Elektrotechnik Seyfarth Weimar

ITET-0x53-00(at)t-online.de