

IP-Symcon – SOLVIS SC-3 Koppelung

Geplant ist ein Modul, das die Installation auf wenige Minuten reduziert, deswegen derzeit noch die manuelle Installation

Ziel: Einrichtung einer *lesenden* Modbus-TCP Instanz um Anlagenwerte aus der SOLVIS RC-03 auszulesen.

Dauer: ~30 Minuten, lediglich müssen die verschiedenen Modbus-Adressen manuell angelegt (kopiert) werden.

BEREICH SOLVIS SC-3

Vorbereiten der [SC-03](#) auf die Modbus-Schnittstelle

Wechsel in den Installateur-Modus (Zugangscode über deinen Heizi oder SOLVIS)

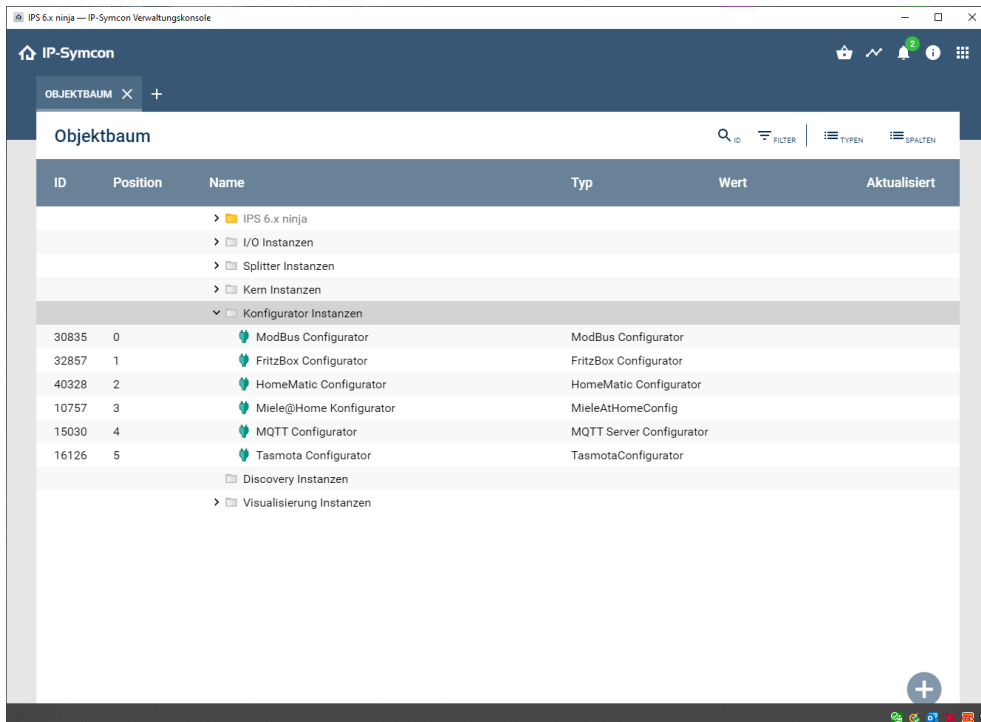
Unter „Sonstiges“ auf Punkt „Modbus“, hier die vorgegebene Adresse nutzen oder bei mehreren Anlagen entsprechend ändern. Der „Modus“ bleibt vorerst auf TCP(read) stehen.



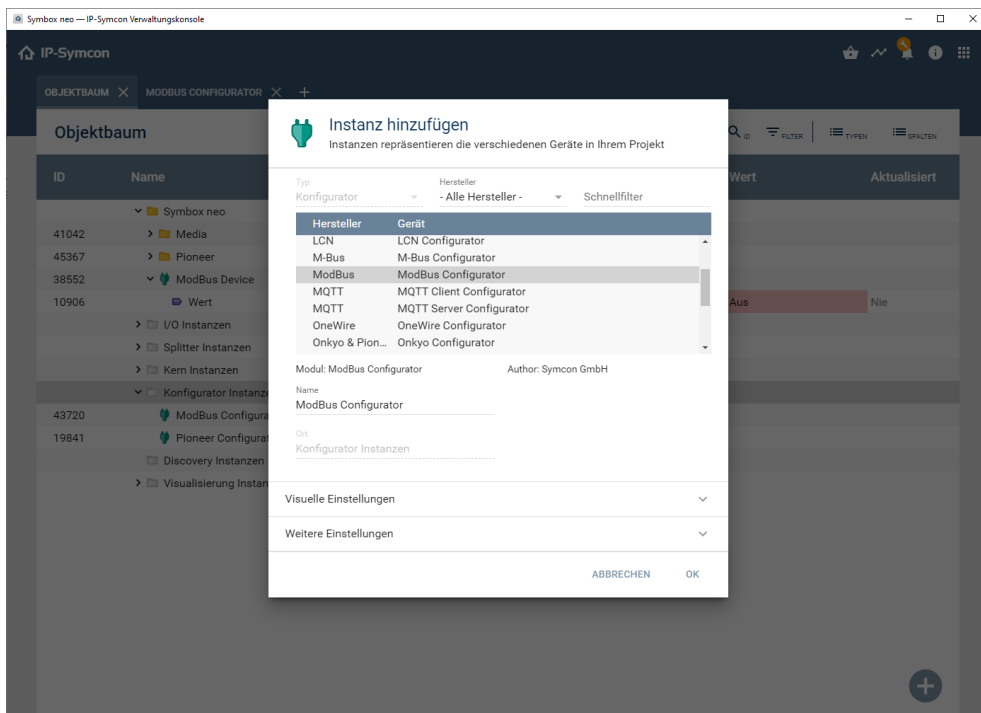
BEREICH IP-SYMCON

KONFIGURATOR

Die Installation erfolgt über die [Verwaltungs-Konsole](#) / [Verwaltungs-Konsole Pro](#) von IP-Symcon.
Hierzu wird der Objektbaum geöffnet

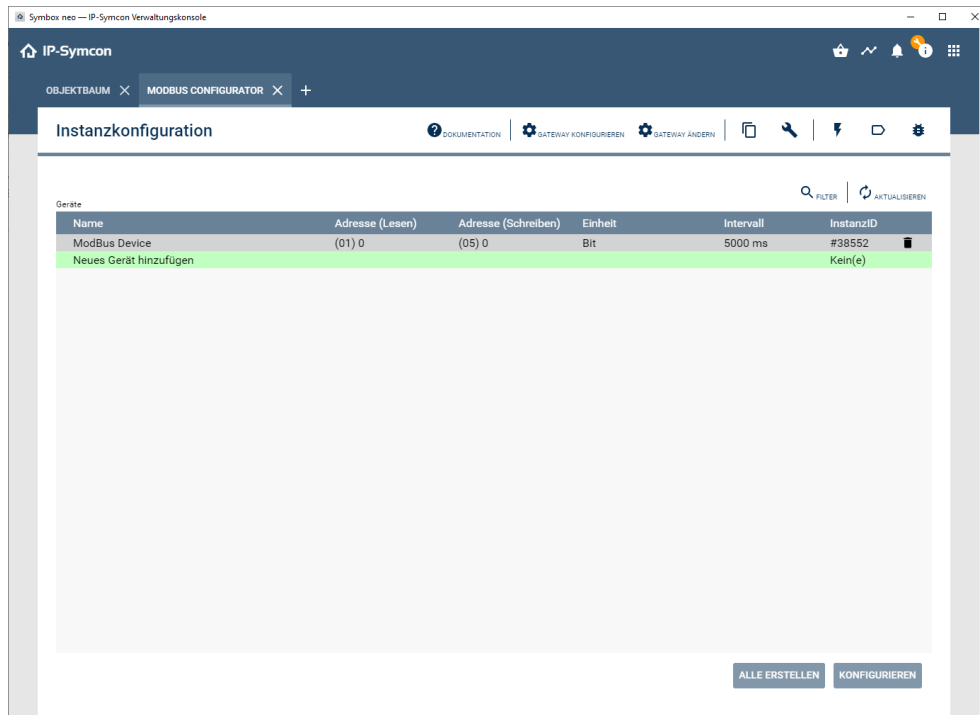


Im Bereich der Konfigurator Instanzen wird der Modbus Configurator als Instanz angelegt.



Danach wird der „Modbus Configurator“ gestartet.

Die grün unterlegte Zeile „Neues Gerät hinzufügen“ muss markiert werden, danach auf Erstellen klicken.

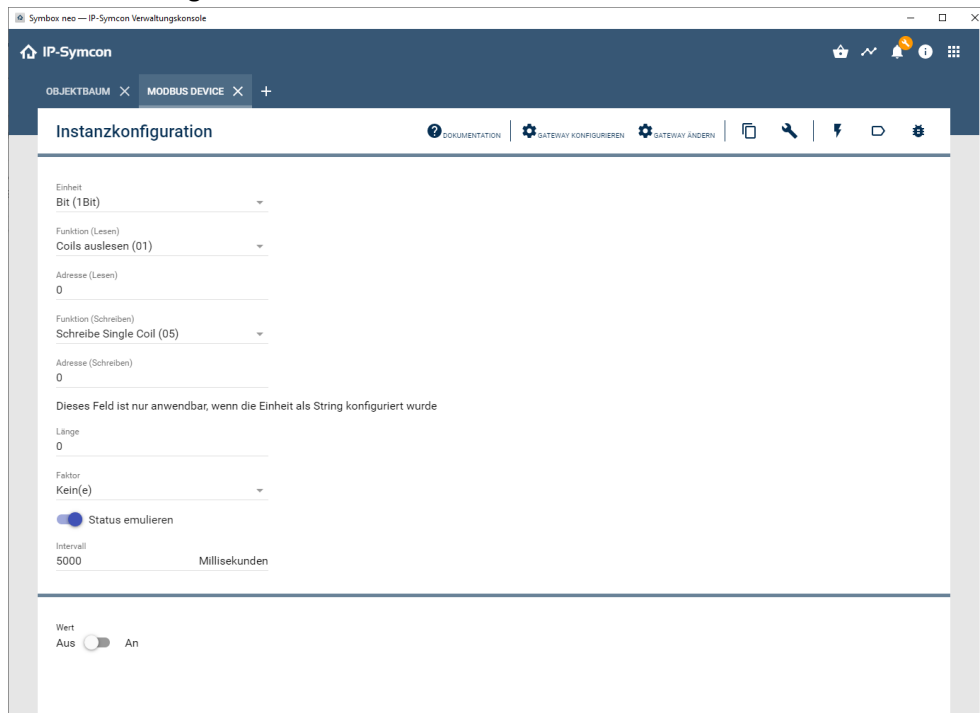


Die Konfiguration lässt lediglich „[ModbusDevice](#)“ zu. Es erfolgt eine Standard-Konfig, die entsprechend der SOLVIS-Schnittstelle geändert werden muss.

MODBUS-INSTANZ

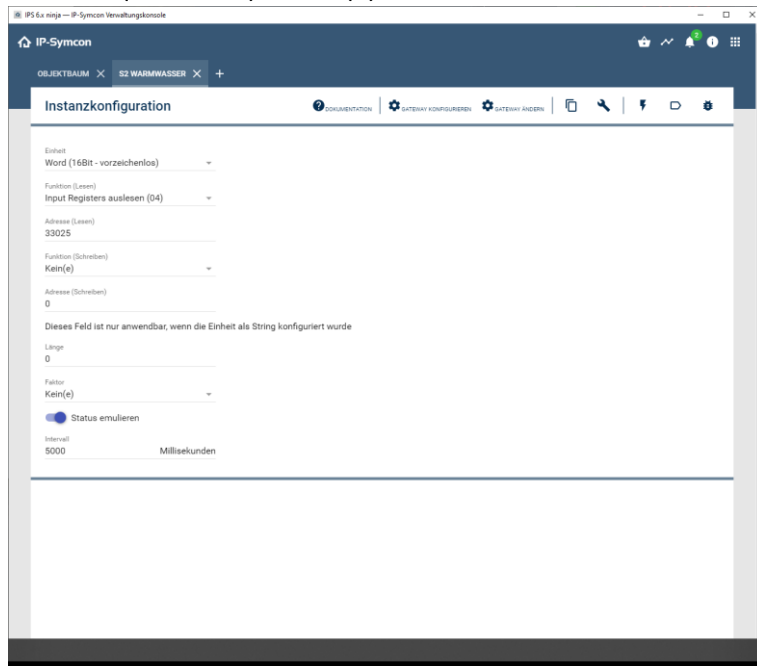
Die im Konfigurator erzeugte Modbus-Instanz wird im Objektbaum in der obersten Kategorie abgelegt. Diese Instanz muss allerdings noch nachbearbeitet werden, damit die Parameter stimmen.

Standard-Konfig

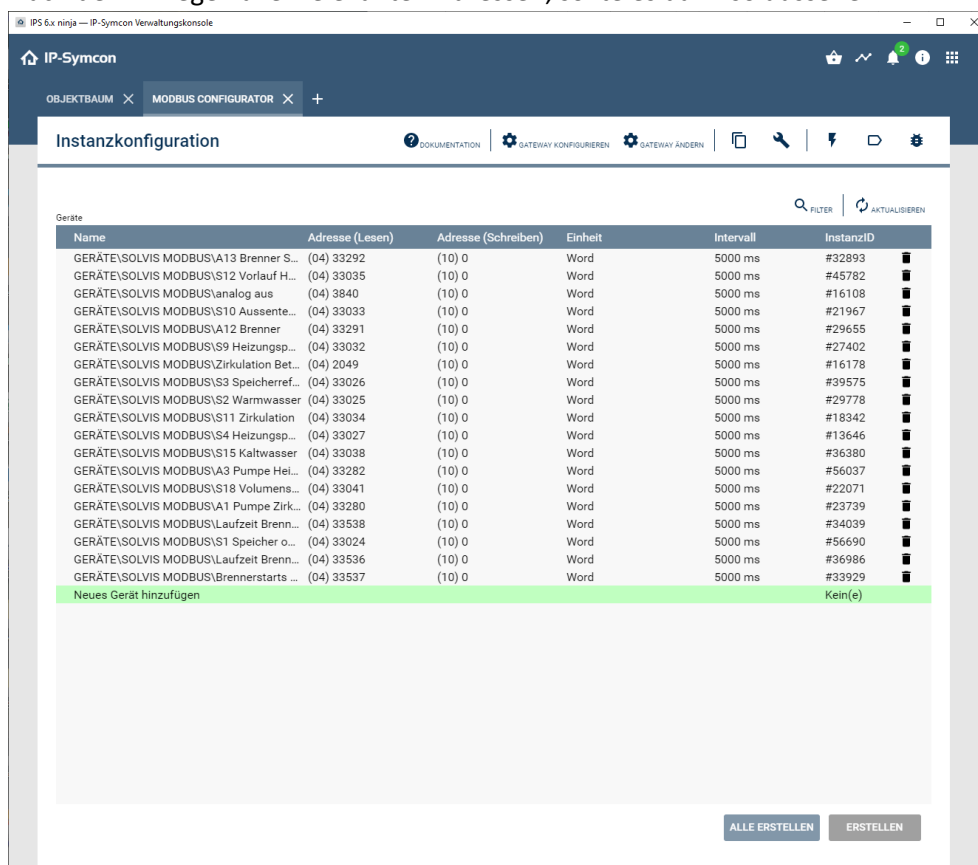


Änderungen:

Einheit: Word(16Bit-vorzeichenlos) ändern
Funktion (Lesen): Input Registers auslesen (04)
Adresse (Lesen): 33025 (hier WW, die benötigten Adressen kommen aus der SOLVIS-Liste)
Funktion (Schreiben): Kein(e)



Nach dem Anlegen aller relevanten Adressen, sollte es dann so aussehen.



Da es sehr mühsam ist, über den Konfigurator mehrere Instanzen anzulegen, wird nur eine angelegt und diese später im Verzeichnisbaum kopiert und nur die benötigte Adresse geändert.

Dies geschieht dann hier über Strg-C / Strg-V (Copy/Paste) oder über „Objekt duplizieren“¹

Symbox neo — IP-Symcon Verwaltungskontrolle

IP-Symcon

OBJEKTBAUM

Objektbaum

ID	Name	Typ	Wert	Aktualisiert
	Symbol neo			
41042	Media			
45367	Pioneer			
38552	ModBus Device	ModBus Address		
10906	Wert	Boolean	Aus	Nie
	I/O Instanzen			
	Splitter Instanzen			
	Kern Instanzen			
	Konfigurator Instanzen			
43720	ModBus Configurator	ModBus Configurator		
19841	Pioneer Configurator	OnlyoConfigurator		
	Discovery Instanzen			
	Visualisierung Instanzen			

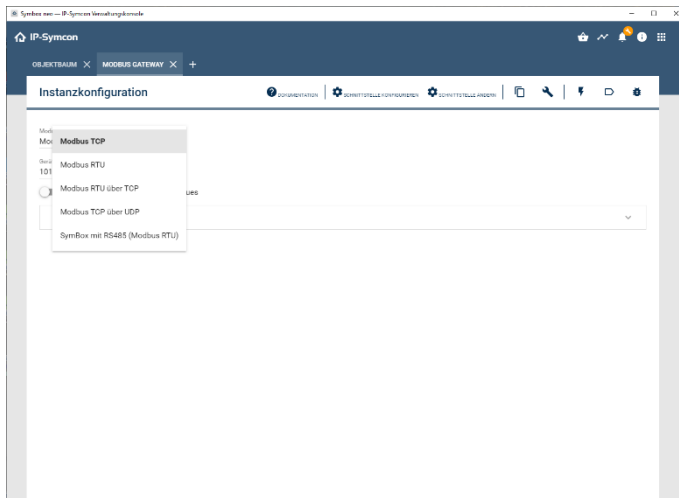
Mit der Anlage einer Modbus-Instanz wird gleichzeitig auch ein Modbus Gateway und ein [Modbus-Client Socket](#) erzeugt. Diese lassen sich aber auch manuell anlegen.

GATEWAY

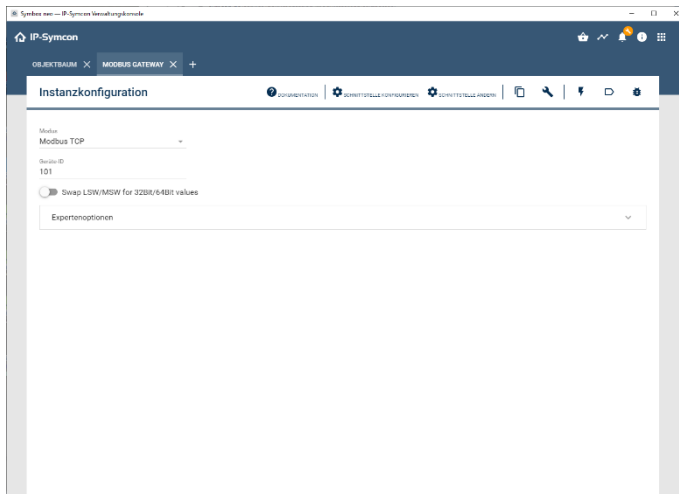
Das [Gateway](#) wird im Bereich „Splitter“ installiert

Ausgewählt werden muss das Protokoll: Modbus TCP

Über die Debug-Funktion kann man hier jetzt schon die Kommunikation zwischen Symcon und SOLVIS SC-03 sehen



Die Geräte-ID aus der Remote-Konsole, Standard ist 101, wird im Gateway eingetragen

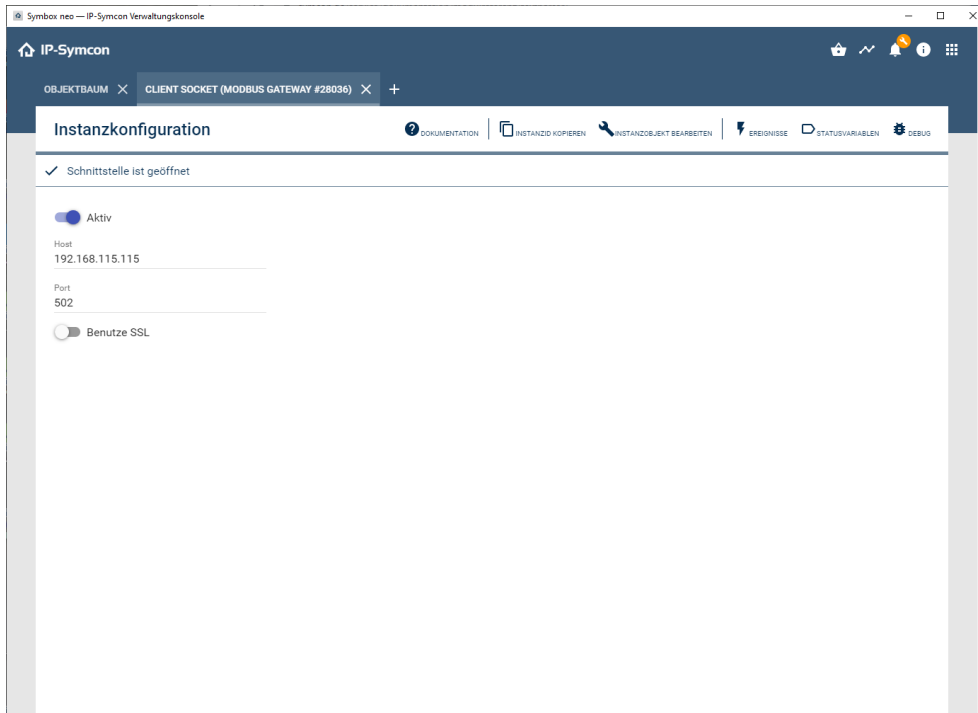


Zeit	Nachricht	Daten	Länge
10:11:04	WAITING	04 75 30 00 01	5
10:11:04	TRANSMIT	04 75 30 00 01	5
10:11:04	BUFFER IN	00 95 00 00 00 05 65 04 02 FF FF	11
10:11:04	RESULT	04 02 FF FF	4
10:11:05	WAITING	04 75 30 00 01	5
10:11:05	TRANSMIT	04 75 30 00 01	5
10:11:05	BUFFER IN	00 96 00 00 00 05 65 04 02 FF FF	11
10:11:05	RESULT	04 02 FF FF	4
10:11:06	WAITING	04 75 30 00 01	5
10:11:06	TRANSMIT	04 75 30 00 01	5
10:11:06	WAITING	04 08 01 00 01	5
10:11:06	BUFFER IN	00 97 00 00 00 05 65 04 02 FF FF	11
10:11:06	RESULT	04 02 FF FF	4
10:11:06	TRANSMIT	04 08 01 00 01	5
10:11:07	BUFFER IN	00 98 00 00 00 05 65 04 02 00 03	11
10:11:07	RESULT	04 02 00 03	4

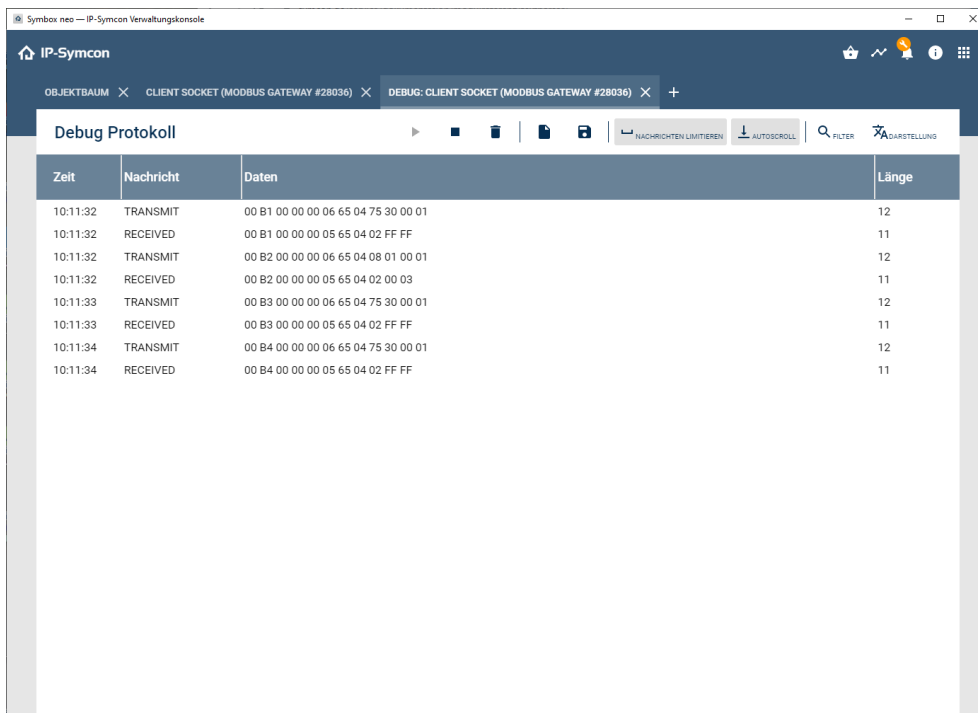
CLIENT SOCKET

Der (automatisch) in den „[I/O Instanzen](#)“ angelegte Modbus-Client Socket benötigt die IP-Adresse der SC-03 (hier 192.168.115.115) und deren Port 502 (ModbusTCP-Standard).

SSL nur bei installierten Zertifikaten

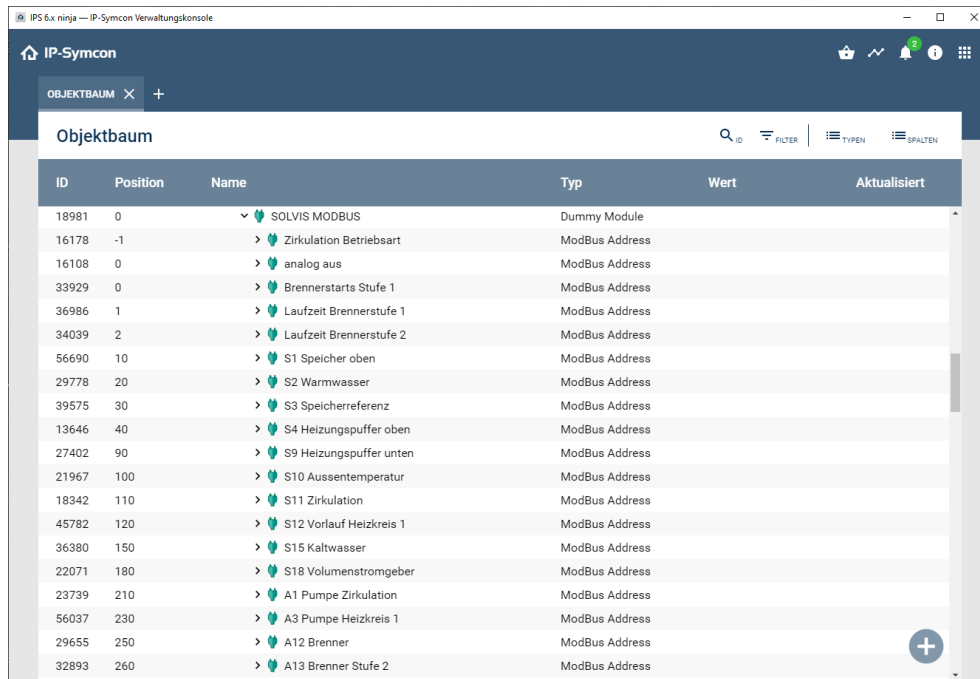


Auch hier ist über die Debugfunktion der (hex-)Datenfluss zu sehen.



VARIABLEN

Nachdem alle ¹ benötigten Modbus-Adressen angelegt und benamst sind,

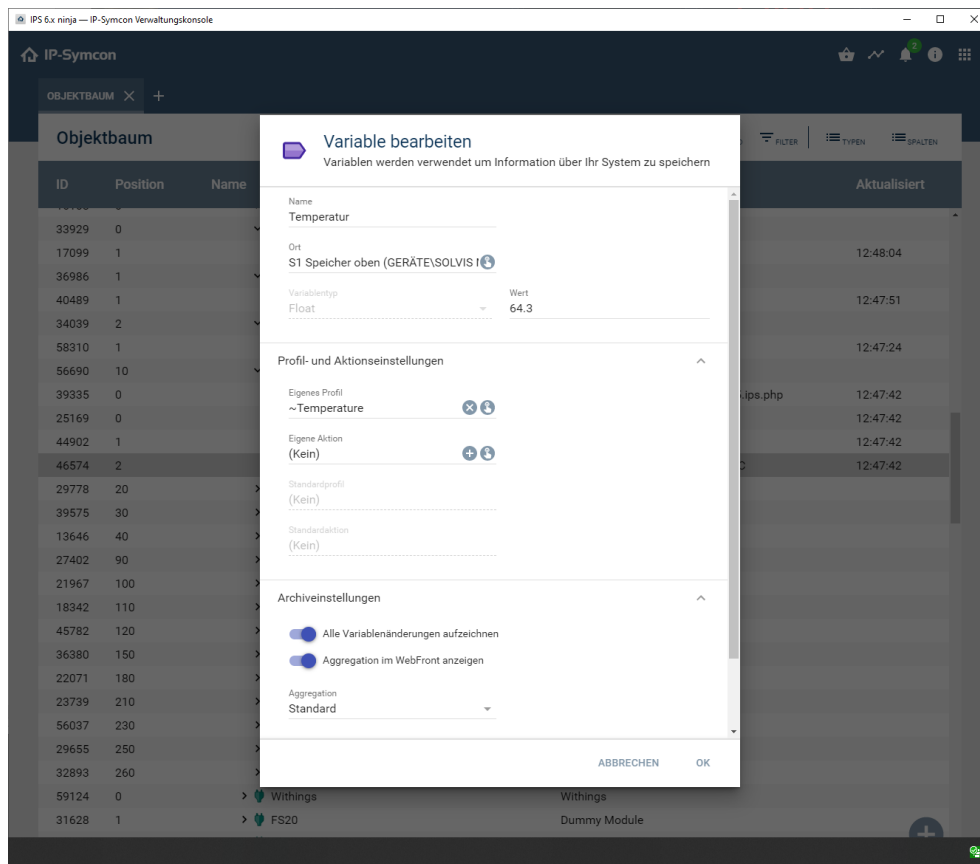


ID	Position	Name	Typ	Wert	Aktualisiert
18981	0	▼ SOLVIS MODBUS	Dummy Module		
16178	-1	> Zirkulation Betriebsart	ModBus Address		
16108	0	> analog aus	ModBus Address		
33929	0	> Brennerstarts Stufe 1	ModBus Address		
36986	1	> Laufzeit Brennerstufe 1	ModBus Address		
34039	2	> Laufzeit Brennerstufe 2	ModBus Address		
56690	10	> S1 Speicher oben	ModBus Address		
29778	20	> S2 Warmwasser	ModBus Address		
39575	30	> S3 Speicherreferenz	ModBus Address		
13646	40	> S4 Heizungspuffer oben	ModBus Address		
27402	90	> S9 Heizungspuffer unten	ModBus Address		
21967	100	> S10 Aussentemperatur	ModBus Address		
18342	110	> S11 Zirkulation	ModBus Address		
45782	120	> S12 Vorlauf Heizkreis 1	ModBus Address		
36380	150	> S15 Kaltwasser	ModBus Address		
22071	180	> S18 Volumenstromgeber	ModBus Address		
23739	210	> A1 Pumpe Zirkulation	ModBus Address		
56037	230	> A3 Pumpe Heizkreis 1	ModBus Address		
29655	250	> A12 Brenner	ModBus Address		
32893	260	> A13 Brenner Stufe 2	ModBus Address		

kann man die untergeordneten Werte der einzelnen Adressen weiterverarbeiten.

Für spätere Auswertungen oder Diagramme müssen die Werte in der Datenbank geloggt werden.

Dies wird in der Variablenbearbeitung in den Archiveinstellungen „Änderungen aufzeichnen“ hinterlegt.



Variable bearbeiten
Variablen werden verwendet um Information über Ihr System zu speichern

Name: Temperatur

Ort: S1 Speicher oben (GERÄTE\SOLVIS I)

Variablentyp: Float Wert: 64.3

Profil- und Aktionseinstellungen

Eigenes Profil: ~Temperature

Eigene Aktion: (Kein)

Standardprofil: (Kein)

Standardaktion: (Kein)

Archiveinstellungen

☒ Alle Variablenänderungen aufzeichnen

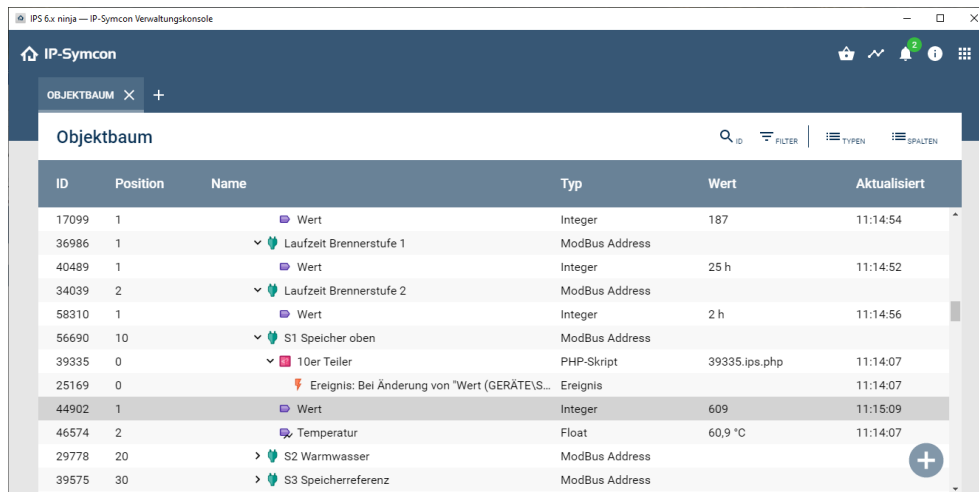
☒ Aggregation im WebFront anzeigen

Aggregation: Standard

ABBRECHEN OK

WERTE

SOLVIS liefert über die Schnittstelle einen Integer-Wert und daher sind die Temperaturwerte ohne Dezimalzeichen, hier im Beispiel werden 609 Int geliefert, dies muss nun noch in einen Dezimalwert „umgerechnet / -gewandelt“ werden. Einfachheit halber bietet sich eine 10er Teilung an:



ID	Position	Name	Typ	Wert	Aktualisiert
17099	1	Wert	Integer	187	11:14:54
36986	1	Laufzeit Brennerstufe 1	ModBus Address		
40489	1	Wert	Integer	25 h	11:14:52
34039	2	Laufzeit Brennerstufe 2	ModBus Address		
58310	1	Wert	Integer	2 h	11:14:56
56690	10	S1 Speicher oben	ModBus Address		
39335	0	10er Teiler	PHP-Skript	39335.ips.php	11:14:07
25169	0	Ereignis: Bei Änderung von "Wert (GERÄTEIS...	Ereignis		11:14:07
44902	1	Wert	Integer	609	11:15:09
46574	2	Temperatur	Float	60,9 °C	11:14:07
29778	20	S2 Warmwasser	ModBus Address		
39575	30	S3 Speicherreferenz	ModBus Address		

Zehner-Teiler Skript Beispiel:

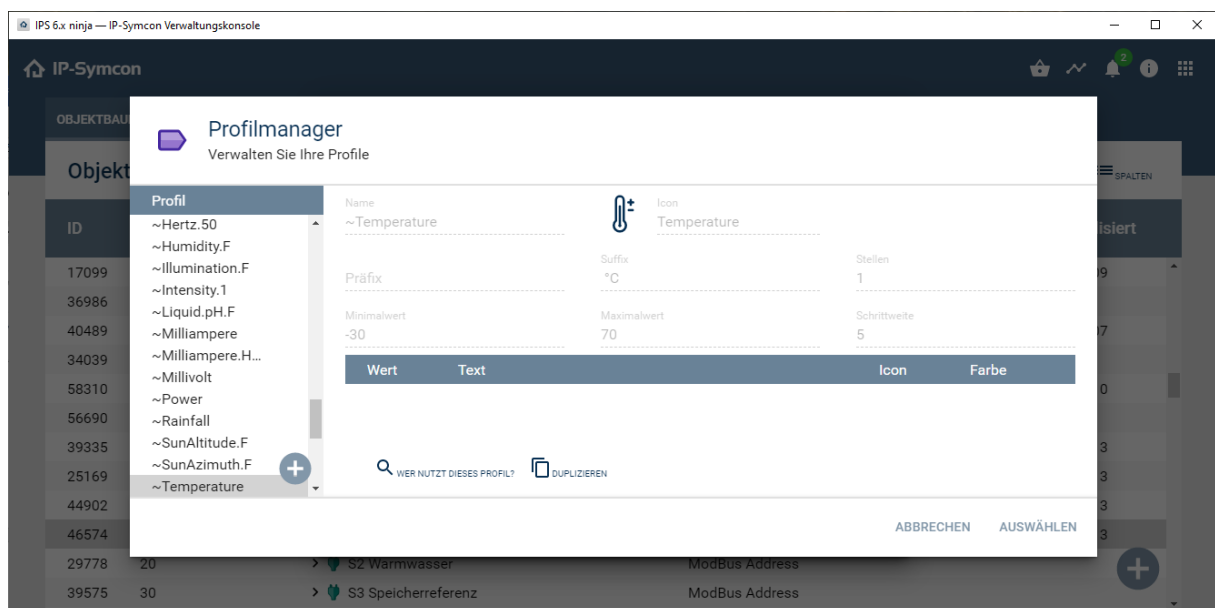
```
<?php
```

```
$wert = GetValue(44902); // Integerwert der ID, hier 609
$tempID = 46574; // Zielvariable für das Ergebnis, hier 60,9
$berechnung = ($wert/10); // Berechnung 609/10 = 60,9
SetValue($tempID,$berechnung); // Zuweisung des Ergebnisses an die ZielVari.
```

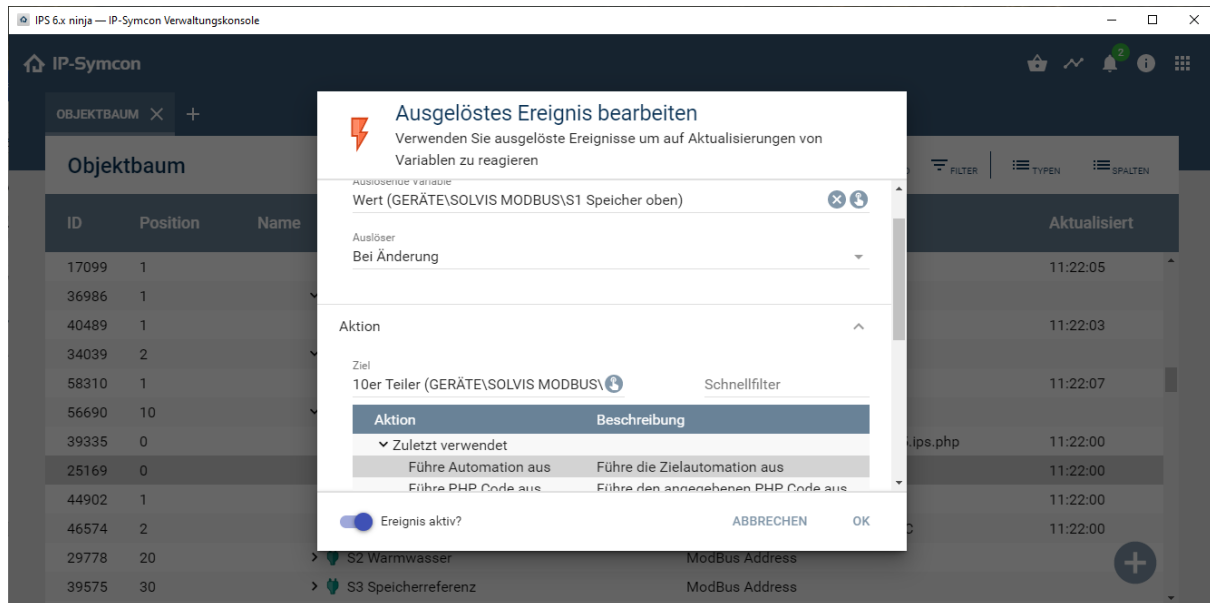
Variable Temperatur, wird geloggt.

Temperatur ist die Arbeitsvariable während Wert die Basis darstellt.

Als Profil wird das Standard-Temperatur Profil gewählt. Es kann auch ein eigenes über den Profilmanager erstellt werden.



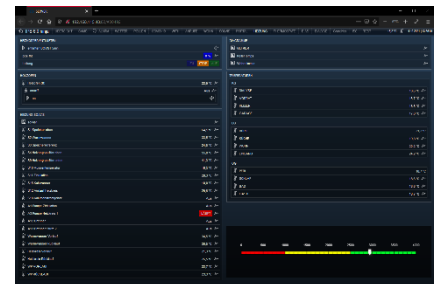
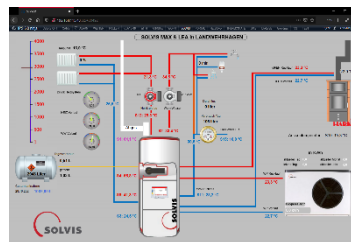
Das erstellte Event triggert das Berechnungsskript jedes Mal, wenn der Wert sich ändert.



Die (geloggten) Werte der verschiedenen Modbus-Adressen können nun

1. im [Webfront](#) mittels Verlinkungen (reine Zahlenwerte)
2. in IPS-View entsprechend grafisch aufbereitet

dargestellt werden.



Eine grafische Darstellung erfordert allerdings einen höheren Zeitaufwand

IPS-View ([IPSSTUDIO](#)) ist eine hervorragende (kostenpflichtige) Zusatzsoftware eines Drittanbieters. Grafische Symbole / Icons usw. sind nicht Bestandteil und müssen daher z.B. über das Internet oder entspr. Software (zB [Colobrico](#)) erstellt/geladen/gekauft werden.