

Aufbau eines IEC61131-3-basierten Energiemanagementsystems für Wohngebäude unter
Einbindung von Smart Metering

-Bachelor Thesis-

Prüfer: Prof. Dr. Aschendorf

Zweitprüfer: Dipl. -Ing. Schroeder

Tobias Redeker

Matrikel-Nr.: 7067839

Abgabetermin: 30.04.2010

Eidesstattliche Erklärung:

Hiermit erkläre ich, dass die Bachelor-Thesis von mir selbstständig verfasst wurde und nur die angegebenen Quellen und Hilfsmittel benutzt wurden.

Werne, den 30.04.2010

(Tobias Redeker)

Inhaltsverzeichnis

1	Einleitung.....	6
2	Smart Metering.....	7
3	Aufgabenstellung.....	8
4	Vorgehensweise.....	10
5	Speicherprogrammierbare Steuerung (SPS).....	11
6	Die Wago SPS.....	12
6.1	Das Wago-I/O-System 750.....	12
6.2	Wago Komponenten.....	13
6.2.1	3.2.1 Primär getaktete Spannungsversorgung 757-622.....	13
6.2.2	KNX IP Controller 750-849.....	14
6.2.3	8-Kanal Digital Ausgangsklemme 750-530.....	15
6.2.4	8-Kanal Digital Eingangsklemme 750-430.....	16
6.2.4	2-Kanal Relaisausgangsklemme 750-517.....	17
6.2.5	2-Kanal Analog Ausgangsklemme 0-20 mA 750-552.....	18
6.2.6	2-Kanal Analog Eingangsklemme für Thermoelemente 750-469.....	20
6.2.7	2 DI DC 24 V, Einbruchsmeldung 750-424.....	22
6.2.8	Bus-Endklemme 750-600.....	23
6.3	Wago Ethernet Settings.....	24
6.3.1	Eine IP-Adresse zuweisen.....	25
6.4	Die CoDeSys Software.....	27
6.4.1	Die Programmierung der WAGO-SPS mit der CoDeSys.....	27
6.4.1.1	Ein Zielsystem auswählen.....	27
6.4.1.2	Ein Programm erstellen.....	29
6.4.1.3	Die Programmiersprachen von CoDeSys.....	29
6.4.1.3.1	Anweisungsliste (AWL).....	30
6.4.1.3.2	Kontaktplan (KOP).....	31
6.4.1.3.3	Funktionsplan (FUP).....	32

	6.4.1.3.4	Ablaufsprache (AS).....	33
	6.4.1.3.5	Strukturierter Text (ST).....	34
	6.4.1.3.6	Frei grafischer Funktionsplan-Editor (CFC).....	35
	6.4.1.4	Programmierung einer Funktion.....	36
	6.4.1.5	Programmieren eines Funktionsblockes.....	38
	6.4.1.6	Die Bibliotheken.....	45
	6.4.1.6.1	Der Bibliotheksverwalter.....	46
	6.4.1.6.2	Arbeiten mit Bibliotheken.....	48
	6.4.1.7	CoDeSys Visualisierung.....	50
	6.4.1.7.1	Eine Visualisierung erstellen.....	54
7		Das Puppenhaus.....	74
	7.1	Bauteile im Haus.....	75
	7.2	Verdrahtung am Haus.....	79
	7.3	Funktionen Haus.....	85
	7.3.1	Haus Untergeschoss.....	85
	7.3.2	Kinderzimmer.....	86
	7.3.3	Schlafzimmer.....	87
	7.3.4	Küche.....	88
	7.3.5	Badezimmer.....	90
	7.3.6	Wohnzimmer.....	90
	7.4	Funktionsbeschreibung.....	91
	7.5	Der Funktionsblock „umschalten“.....	93
	7.6	Tastenauswertungen.....	95
	7.7	Der Funktionsblock HeizungFB.....	96
	7.8	Die „Lichtorgel“.....	98
8		Enocean.....	100
	8.1	Funkreceiver Enocean (750-642).....	100
	8.2	Die Implementierung und Programmierung der Enocean-Klemme	101
	8.3	Enocean Komponenten.....	102
	8.3.1	4-fach-Taster.....	102
	8.3.2	Thermokon Temperatursensor SR04.....	103

9	RTC-Modul (750-640).....	104
	9.1 Die Programmierung des RTC-Moduls.....	105
10	Smart Metering.....	114
	10.1 3-Phasen-Leistungsmessklemme (750-493).....	114
	10.2 Die Programmierung der 3-Phasen-Leistungsmessklemme.....	122
	10.3 Der Zählerbaustein.....	123
	10.4 Energetische Rechnung.....	124
11	Fazit.....	129
12	Zusammenfassung/Abstract.....	130
13	Abbildungsverzeichnis.....	131
14	Literaturverzeichnis.....	137
13	Anhang: Quelltexte.....	138
	13.1 Funktionen.....	138
	13.2 Funktionsblöcke.....	140
	13.3 Programme.....	145
	13.4 Globale Variablen.....	201
	13.5 Belegung der Klemmen.....	215

1 Einleitung

Das Thema Energieversorgung ist in den letzten Jahren mehr und mehr in das Interesse der Allgemeinheit gerückt. Steigende Energiekosten sowie die Tatsache dass fossile Energieträger wie Kohle oder Erdgas nur noch eine begrenzte Zeit zur Verfügung stehen, sind unter anderem Auslöser dafür, dass wir unsere Energiesituation überdenken sollten. Ein weiterer wichtiger Aspekt ist die weltweite Klimasituation. Die Energiegewinnung durch fossile Energieträger führt zu einer erhöhten Schadstoffemission sowie der Freisetzung von Treibhausgasen und ist deshalb ein Hauptgrund für die globale Erderwärmung. Auch die Atomkraft erhält im Bezug auf die Sicherheit einen zweifelhaften Ruf bei der Herstellung von Energie, daher rückt der Trend immer mehr in Richtung regenerative Energiequellen.

Neben der Energieerzeugung spielt die Energienutzung auch eine wichtige Rolle in der derzeitigen Energiesituation. Zur Zeit beschäftigen sich politische Debatten eher damit, wie Energie möglichst umweltschonend und effizient hergestellt oder die Schadstoffemission gesenkt werden kann. Themen wie Energieeinsparung oder Energiekostensenkung werden weniger stark diskutiert, dabei existieren gerade im Wohn-oder Gewerbebereich viele Energieeinsparpotentiale. Problem hierbei ist jedoch, dass gerade im Bereich von Wohn- und Nutzflächen der Energieverbrauch für den Benutzer schlecht oder gar nicht dargestellt wird. In jedem Haushalt befinden sich zwar Stromzähler, diese liefern jedoch keine detaillierten Angaben über mögliche „Stromfresser“ innerhalb des gesamten Energiehaushaltes.

Für das Energiezählen sollen in naher Zukunft sogenannte Smart Meter (intelligente Stromzähler) in jedem Haushalt eingesetzt werden. Diese Smart Meter sollen den alten Stromzähler komplett ablösen.

2 Smart Metering

In jedem Haushalt befinden sich Stromzähler, welche die elektrische Nutzenergie erfassen. Diese Geräte messen üblicherweise den Wechselstrom und die anliegende Wechselspannung, welche dann multipliziert und über die Zeit integriert werden. Diese Rechnung liefert als Ergebnis die genutzte elektrische Wirkenergie in Kilowattstunden.

Die Grundlage für Smart Metering ist der sogenannte Intelligente Stromzähler. Anders als die alten Zähler, welche hauptsächlich noch mechanisch arbeiten, bestehen die Smart Meter nur noch aus elektronischen Zählelementen. Zudem verfügen die Intelligen Stromzähler über eine Kommunikationsschnittstelle und können somit von der Ferne ausgelesen werden.

Diese Smart Meter sollen in Zukunft die alten Stromzähler komplett ersetzen. Die modernen Zähler bieten sowohl auf der Energieversorgerseite als auch beim Endkunden erhebliche Vorteile. Versorgungsunternehmen können zukünftig den Strom- bzw. Energieverbrauch von der Ferne auslesen und erhalten somit eine bessere Übersicht über die Gesamtnachfrage im ganzen Stromnetz. Die Kunden haben die Möglichkeit ihren Energieverbrauch durch Speicherung der Verbrauchsdaten einzusehen. Somit erhalten sie einen besseren Überblick über die Energiekosten und können mögliche „Stromfresser“ aufspüren.

Durch die Fernauslese entstehen jedoch gewisse Sicherheitsrisiken. Werden die Daten über ein Kommunikationsmedium, wie zum Beispiel dem Internet oder dem Mobilfunknetz ausgelesen, besteht das Risiko von Hacker-Angriffen. Die Verbrauchsdaten der Kunden stünden dann sozusagen auf dem Präsentierteller und könnten von unbefugten Personen eingesehen werden. Durch den Energieverbrauch im Haushalt können bestimmte Lebensgewohnheiten der dort lebenden Personen studiert und für kriminelle Zwecke genutzt werden.

3 Aufgabenstellung

Derzeit existieren eine Vielzahl von Gebäudebussystemen wie zum Beispiel der Europäische Installationsbus (EIB), PEHA House Control (PHC) oder Local Control Network (LCN). Die meisten dieser Bussysteme besitzen eine Firmeneigene Bearbeitungssoftware, über welche das jeweilige Bussystem parametrierbar und programmiert werden kann. Diese Programme haben in der Regel bereits vorprogrammierte Funktionen, wie zum Beispiel die klassische Taster-umschalten-Funktion, bei welcher das Betätigen eines Tasters einen Zustandswechsel bei dem jeweiligen Aktor hervorruft.

Nach dem derzeitigen Stand ist die Anschaffung eines Gebäudebussystems sehr teuer im Vergleich zu einer konventionellen elektrischen Installation und lohnt sich daher erst bei größeren Gebäuden mit einem hohen Ausstattungsgrad. Die meisten dieser Gebäudebussysteme werden daher in größeren Bürogebäuden oder Villen eingesetzt und sind in gewöhnlichen Privathaushalten kaum verbreitet.

Die meisten Hersteller von Gebäudebussystemen motivieren ihre potentiellen Kunden größtenteils mit Argumenten wie Komfort oder Sicherheit zum Kauf ihrer Produkte. Themen wie Energiemanagement oder Energieeinsparungen mit Hilfe von einer Gebäudeautomation sind daher selten im Umlauf.

Eine günstige und effektive Alternative zu den Gebäudebussystemen ist ein Speicherprogrammierbare Steuerung (SPS). Firmen wie Wago oder Beckhoff stellen seit Jahren Hard- und Software her, um eine Gebäudeautomation mit Hilfe einer SPS zu realisieren. Mit Hilfe spezieller Klemmen und Bibliotheken lassen sich Funktionen programmieren, welche energetische Berechnungen durchführen und deren Ergebnisse in einer Visualisierung darstellen.

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter
Einbindung von Smart Metering

Um den Energieverbrauch mit Hilfe Speicherprogrammierbaren Steuerung in einem Wohnhaus darzustellen, ergibt sich folgende Aufgabenstellung:

**Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter
Einbindung von Smart Metering**

4 Vorgehensweise

Zur Simulations- und Testzwecken wurde zunächst ein Puppenhaus mit verschiedenen Sensoren und Aktoren aufgebaut. Diese Sensoren und Aktoren wurden mit einer Wago SPS verbunden, welche danach über die Software CoDeSys parametriert und programmiert wurde.

Die Funktionen der Gebäudeautomationen im Puppenhaus konzentrieren auf ein passives Energiemanagement. Das bedeutet, das System reagiert automatisch um gewisse Energieeinsparungen zu gewährleisten. Wird beispielsweise ein Fenster geöffnet während ein Heizkörper noch aktiv, schaltet die SPS diesen Heizkörper automatisch ab.

Um ein Smart Metering zu simulieren, wurden im späteren Verlauf bestimmte Zählerbausteine programmiert. Diese Zählerbausteine zählen zunächst die Betriebsdauer aller im Puppenhaus befindlichen elektrischen Verbraucher. Die theoretischen Leitungsaufnahmen der Geräte sowie die Stromtarife werden über Software Manuell eingestellt.

Um ein tatsächliches Smart Metering darzustellen, wurden in dem Haus noch zusätzlich 230 V Lampen installiert. Deren Leistungen werden über spezielle Leistungsmessklemmen gemessen und mit in die Energetischen Berechnungen der Software integriert.

5 Speicherprogrammierbare Steuerung (SPS)

Eine Speicherprogrammierbare Steuerung oder kurz SPS besteht in der Regel aus Ein- und Ausgangsmodulen für Sensorik und Aktorik. Diese Ein- und Ausgangsmodule sind über ein Steuermodul miteinander gekoppelt. Das Steuermodul besitzt einen sogenannten Programmspeicher, auf welchen ein Steuerungsprogramm geladen werden kann. Mit diesem Programm wird bestimmt, wie die jeweiligen Ein- und Ausgänge miteinander verknüpft sind.

Die Speicherprogrammierbare Steuerung wurde zunächst entwickelt, um Maschinen oder Anlagen zu steuern. Firmen wie Wago oder Beckhoff arbeiten jedoch seit mehreren Jahren daran, eine Speicherprogrammierbare Steuerung innerhalb einer Gebäudeautomation einzusetzen. Dafür stellen sie spezielle Klemmen und Software-Bibliotheken her.

SPS-Systeme werden über eine spezielle Software programmiert. Diese Software arbeitet mit Programmiersprachen gemäß der IEC-61131-3. Diese Norm definiert fünf verschiedene Programmiersprachen.

AWL: Anweisungsliste

KOP: Kontaktplan

FBS: Funktionsbausteinsprache

AS: Ablaufsprache

ST: Strukturierter Text

6 Die Wago SPS

Die Firma Wago stellt seit einigen Jahren Komponenten und Softwarelösungen für speicherprogrammierbare Steuerungen her. Dabei spezialisiert Wago sich unter anderem auf den Bereich Gebäudesystemtechnik und stellt spezielle Klemmen und Bibliotheken her, welche zur Realisierung von gebäudetechnischen Funktionen sehr hilfreich sind.

6.1 Das Wago-I/O-System 750

Das Wago-I/O-System ist ein E/A-System, welches als sogenannter Feldbusknoten auf eine Tragschiene montiert wird. Dieser Feldbusknoten beginnt mit dem Feldbuskoppler/-Controller. Je nach Koppler können verschiedene Feldbussysteme wie zum Beispiel PROFIBUS, INTERBUS, ETHERNET TCP/IP, CAN, MODBUS, oder LON mit der SPS gekoppelt werden. Außerdem dient der Feldbuscontroller als Stromversorgung für die angereichten Busklemmen. Der Feldbusknoten wird mit der Endklemme abgeschlossen.

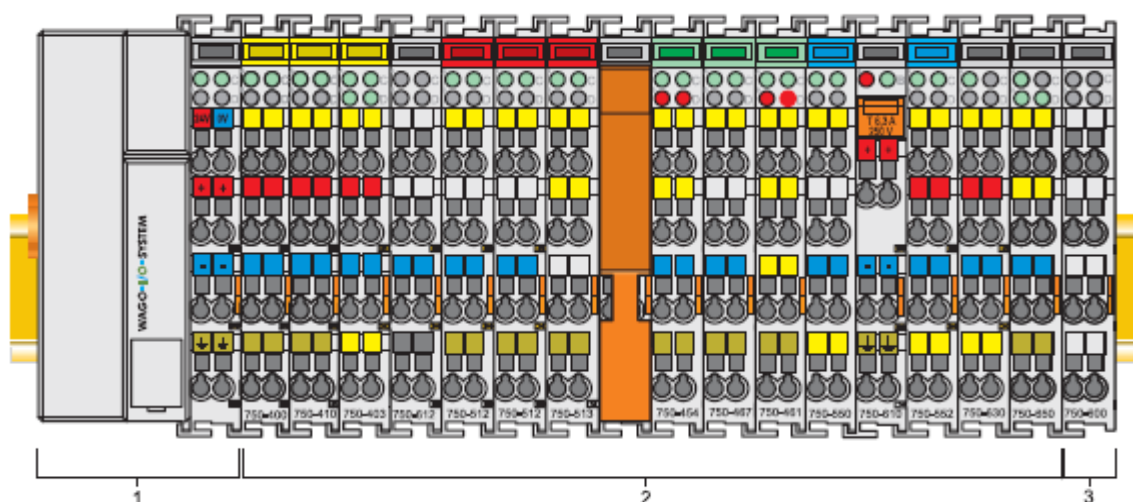


Abbildung 1: Feldbusknoten

6.2 Wago Komponenten

6.2.1 Primär getaktete Spannungsversorgung 757-622



Abbildung 2.1: Primär getaktete Spannungsversorgung (757-622)

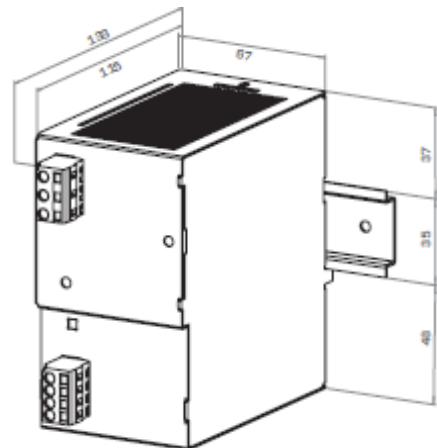


Abbildung 2.2: Primär getaktet Spannungsversorgung (Montage)

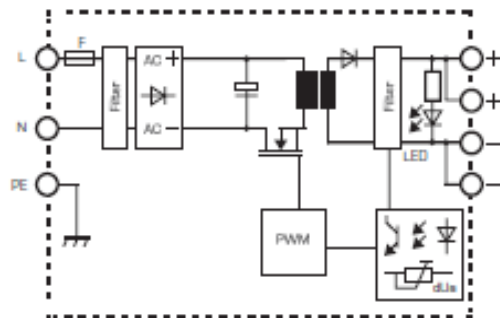


Abbildung 2.2: Primär getaktet Spannungsversorgung (Kontaktplan)

Die Wago Spannungsversorgung 757-622 versorgt den Klemmenbus bzw. den Feldbuskoppler mit der 24 V Versorgungsspannung. Sie hat Eingangsseitig L, N, PE Anschlüsse für 230 V. Ausgangsseitig besitzt die Spannungsversorgung jeweils zwei 24 V und zwei Masse Anschlüsse.

6.2.2 KNX IP Controller 750-849



Abbildung 3.1: KNX-IP Controller (750-849)

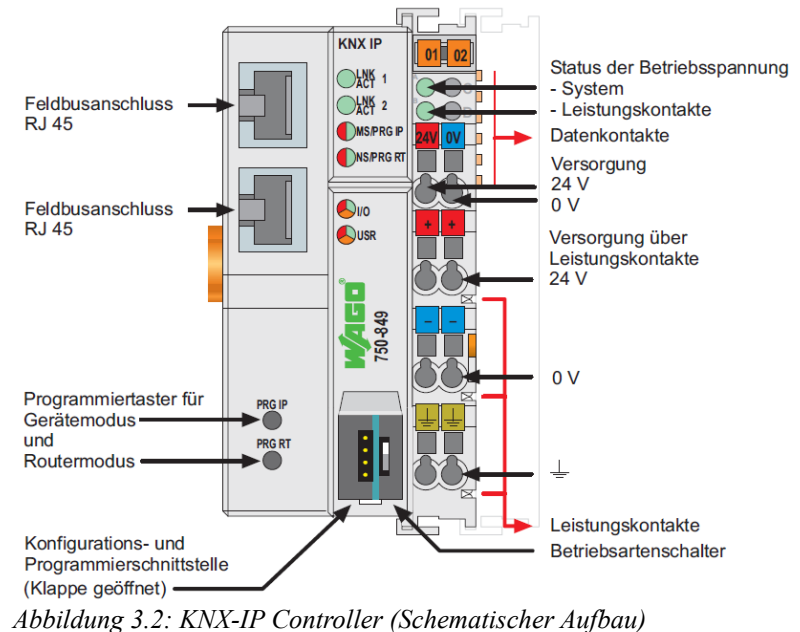


Abbildung 3.2: KNX-IP Controller (Schematischer Aufbau)

Der KNX IP Controller ist ein programmierbarer Feldbuscontroller. Als 2-Port Ethernet Switch verfügt er über zwei RJ45-Buchsen und kann direkt mit einem IP-Netzwerk verbunden werden.

Programmiert wird der Controller mit der Wago Software „WAGO-I/O-PRO CAA“ welche auf der IEC 61131-3 basiert. Hierfür stehen dem Controller 512 KB Programmspeicher, 256 KB Datenspeicher und 24 KB Retain-Speicher zu Verfügung. Die Konfigurationsschnittstelle befindet sich hinter einer Abdeckklappe. Über diese Konfigurationsschnittstelle kann der Controller, mit einem entsprechendem Kabel, direkt über eine serielle Schnittstelle (z.B. USB, RS-232) mit einem Computer verbunden werden.

6.2.3 8-Kanal Digital Ausgangsklemme 750-530



Abbildung 4.1: 8-Kanal Digital Ausgangsklemme (750-530)

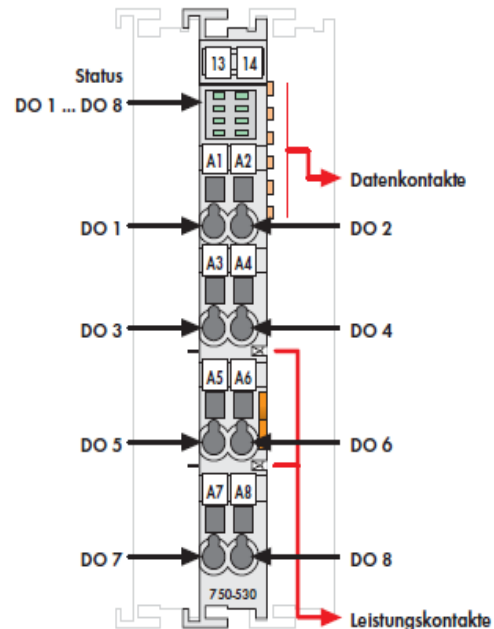


Abbildung 4.2: 8-Kanal Digital Ausgangsklemme (Schematischer Aufbau)

Die 8-Kanal Digital Ausgangsklemme überträgt digitale Ausgangssignale von dem Steuergerät direkt an den angeschlossenen Aktor. Die Ausgänge der Klemme sind High Aktiv was bedeutet: Sendet das Steuergerät eine logische „EINS“ (High Pegel), dann liegt an dem jeweiligen Ausgang ein Potential von 24 V. Dieses Potential wird intern über die Leistungskontakte einer vorgeschalteten Einspeiseklemme gespeist. Die Status LED am oberen Ende der Klemme gibt an, welche Ausgänge sich gegenwärtig auf einem High Potential befinden.

6.2.4 8-Kanal Digital Eingangsklemme 750-430



Abbildung 5.1: 8-Kanal Digital Eingangsklemme (750-430)

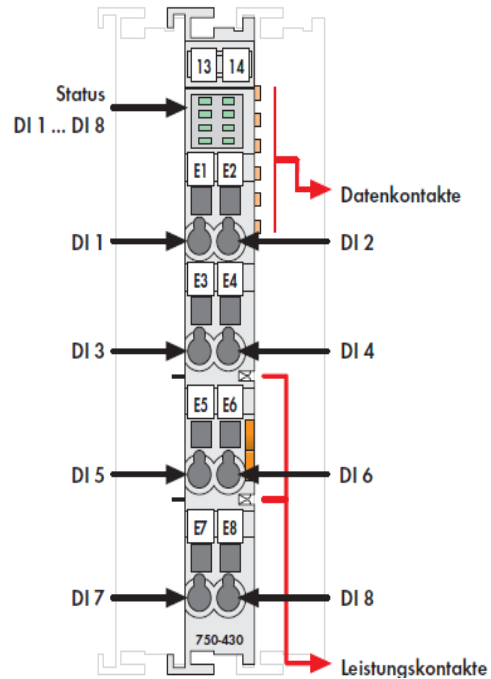


Abbildung 5.2: 8-Kanal Digital Eingangsklemme

Die 8-Kanal Digital Eingangsklemme nimmt Spannungspotentiale von Sensoren oder Tastern am jeweiligen Eingang auf und gibt diese als binäre Steuersignale an Bus weiter. Der Eingangspegel der Signalspannung muss dabei -3 V bis +5 V DC für eine logische „NULL“ und 15 V bis 30 V DC für eine logische „EINS“ betragen. Zur Störsicherheit ist jedem der acht Eingänge ein zusätzlicher RC-Filter mit einer Zeitkonstante von 3 ms vorgeschaltet. Die Status-LED am oberen Ende der Klemme gibt an, welchen Zustand der jeweilige Eingang gegenwärtig besitzt.

6.2.4 2-Kanal Relaisausgangsklemme 750-517



Abbildung 6.1: 2-Kanal
Relaisausgangsklemme (750-517)

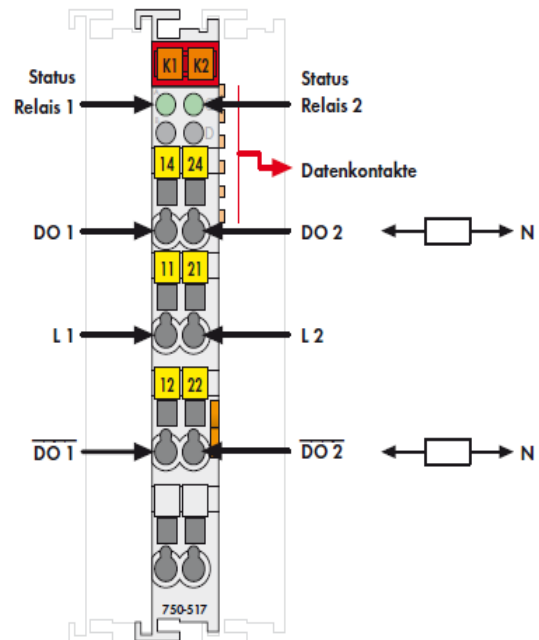


Abbildung 6.2: 2-Kanal Relaisausgangsklemme
(Schematischer Aufbau)

Die 2-Kanal Relaisausgangsklemme verfügt über zwei separate Relais, welche über die Steuersignale des Automatisierungsgerätes angesteuert werden. An den beiden Relaiskontakten L1 und L2 kann jeweils ein externes Potential angeschlossen werden. Die beiden Ausgangskanäle sowie die Steuersignale sind galvanisch voneinander getrennt, Somit können an den beiden Wechslern unterschiedliche Potentiale angeschlossen werden. Befindet sich das Relais und Ruheschaltung, so sind die Kontakte L1 und L2 auf die beiden unteren Ausgangskanäle $\overline{D0}$ und $\overline{D1}$ geschaltet. Sind die Kontakte L1 und L2 auf D0 und D1 geschaltet, so leuchtet die Status-LED auf.

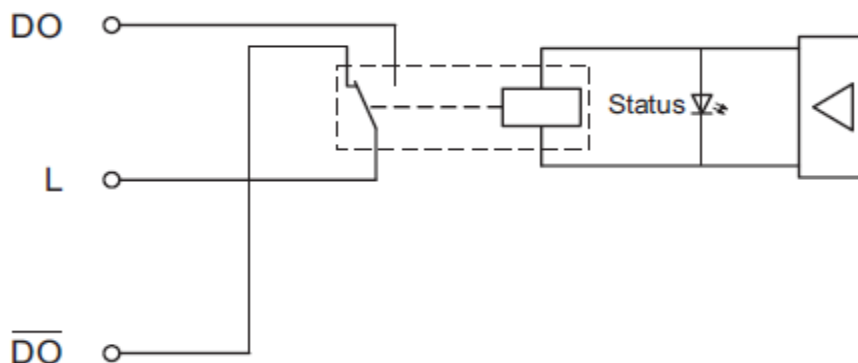


Abbildung 6.3: 2-Kanal Relaisausgangsklemme (Kontaktplan)

6.2.5 2-Kanal Analog Ausgangsklemme 0-20 mA 750-552



Abbildung 7.1: 2-Kanal Analog Ausgangsklemme (750-552)

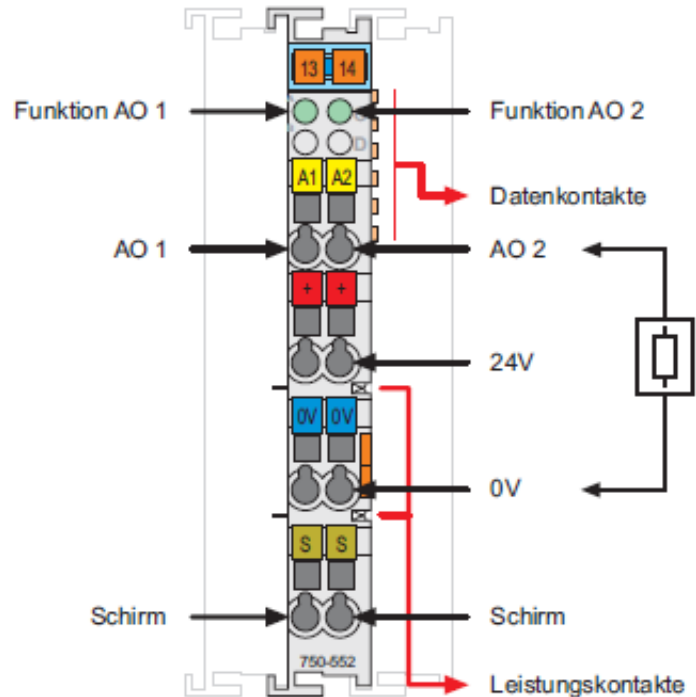


Abbildung 7.2: 2-Kanal Analog Ausgangsklemme (Schematischer Aufbau)

Die 2-Kanal Analog Ausgangsklemme erzeugt über einen A/D-Wandler Signale von 0-20 mA mit einer Auflösung von 12 Bit. Beider Ausgangskanäle sowie die Internen Steuersignale sind galvanisch voneinander getrennt. Die Klemme erhält ihre Spannungsversorgung von 24 V über die vorgeschaltete Busklemme oder über eine Einspeiseklemme.

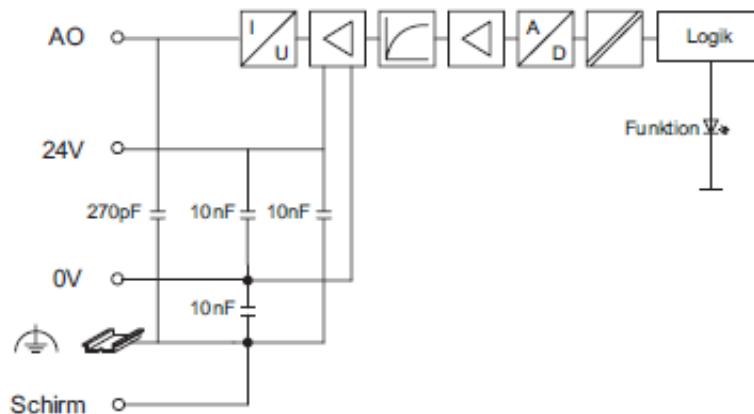


Abbildung 7.3: 2-Kanal Analog Ausgangsklemme (Kontaktplan)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Insgesamt liefert die Analog Ausgangsklemme je Kanal 16 Daten- und 8 Steuerbits. Der digitale Ausgabewert ist insgesamt 12 Bit groß und wird auf den Bits B3 bis B14 abgebildet. Die niederwertigen Bits B0 bis B2 werden nicht ausgewertet.

Prozesswerte der Klemme 750-552				
Ausgangsstrom 0 mA - 20 mA	Zahlenwert			Status- byte Hex.
	Binär Ausgabewert	Hex.	Dez.	
0,0	'0000.0000.0000.0000'	0x0000	0	0x00
2,5	'0001.0000.0000.0000'	0x1000	4096	0x00
5,0	'0010.0000.0000.0000'	0x2000	8192	0x00
7,5	'0011.0000.0000.0000'	0x3000	12288	0x00
10,0	'0100.0000.0000.0000'	0x4000	16384	0x00
12,5	'0101.0000.0000.0000'	0x5000	20480	0x00
15,0	'0110.0000.0000.0000'	0x6000	24576	0x00
17,5	'0111.0000.0000.0000'	0x7000	28672	0x00
20,0	'0111.1111.1111.1111'	0x7FFF	32767	0x00

Abbildung 7.4: 2-Kanal Analog Ausgangsklemme (Prozessabbild)

6.2.6 2-Kanal Analog Eingangsklemme für Thermoelemente 750-469



Abbildung 8.1: 2-Kanal Analog Eingangsklemme für Thermoelemente (750-469)

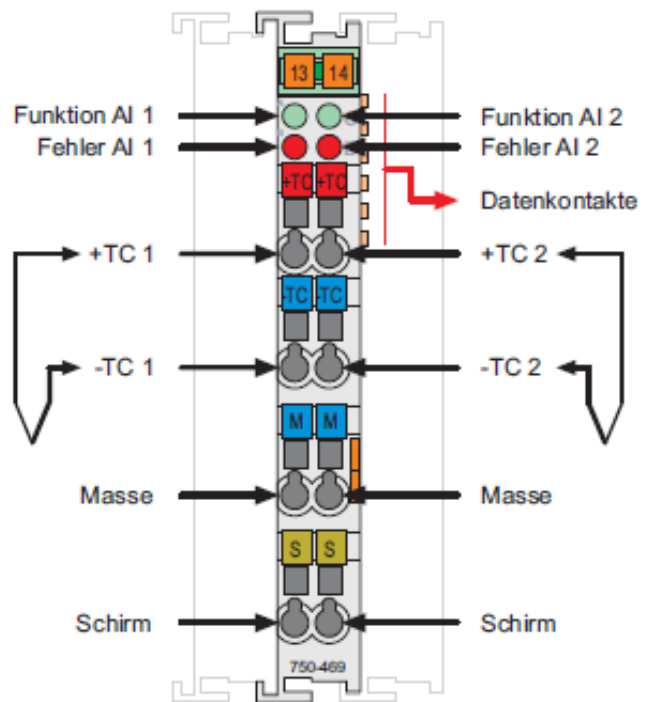


Abbildung 8.2: 2-Kanal Analog Eingangsklemme für Thermoelemente (Schematischer Aufbau)

Die Analog Eingangsklemme für Thermoelemente misst Spannungswerte und wandelt diese in Temperaturwerte um. Zur Umrechnung und Linearisierung der Spannungswerte verfügt die Klemme über einen eigenen Mikroprozessor. Die Klemme verfügt über zwei Eingangskanäle und kann daher die Daten von zwei Thermoelemente separat aufnehmen. Die Feld- und Systemebene sind galvanisch voneinander getrennt. Zudem verfügt die Klemme pro Kanal über zwei Status-LEDs. Die grüne LED signalisiert die Betriebsbereitschaft sowie die störungsfreie Kommunikation der Kanäle. Die rote LED signalisiert eine Unterbrechung der Sensorleitung oder die Überschreitung des zulässigen Messbereiches.

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Die Temperaturwerte werden in einem Datenwort (16 Bit) ausgewertet. Ein Digit entspricht dabei 0,1 °C. Bei einer Temperatur von 0°C entspricht demnach der Zahlenwert 0x0000 (dez. 0) und eine Temperatur von 50°C 0x0032 (dez. 500). Um die korrekte Temperatur später im Programm zu erhalten, muss dieser Wert mit 0,1 multipliziert werden.

750-469, Typ K, Drahtbruchdiagnose					
Temperatur	Zahlenwert ¹⁾			Status- byte	LED Fehler
°C	Binär	Hex.	Dez.	Hex.	AI 1, 2
<-100,0	'1111.1100.0001.1000'	0xFC18	-1000	0x41	ein
-100,0	'1111.1100.0001.1000'	0xFC18	-1000	0x00	aus
0,0	'0000.0000.0000.0000'	0x0000	0	0x00	aus
100,0	'0000.0011.1110.1000'	0x03E8	1000	0x00	aus
200,0	'0000.0111.1101.0000'	0x07D0	2000	0x00	aus
300,0	'0000.1011.1011.1000'	0x0BB8	3000	0x00	aus
400,0	'0000.1111.1010.0000'	0x0FA0	4000	0x00	aus
500,0	'0001.0011.1000.1000'	0x1388	5000	0x00	aus
1000,0	'0010.0111.0001.0000'	0x2710	10000	0x00	aus
1200,0	'0010.1110.1110.0000'	0x2EE0	12000	0x00	aus
1300,0	'0011.0010.1100.1000'	0x32C8	13000	0x00	aus
1370,0	'0011.0101.1000.0100'	0x3584	13700	0x00	aus
>ca.1370,0	'0011.0101.1000.0100'	>0x3584	>13700	0x42	ein
Drahtbruch	'0111.1111.1111.1111'	0x7FFF	32767	0x42	ein

Abbildung 8.3: 2-Kanal Analog Eingangsklemme für Thermoelemente (Prozessabbild)

6.2.7 2 DI DC 24 V, Einbruchsmeldung 750-424



Abbildung 9.1: 2 DI DC 24 V,
Einbruchsmeldung (750-424)

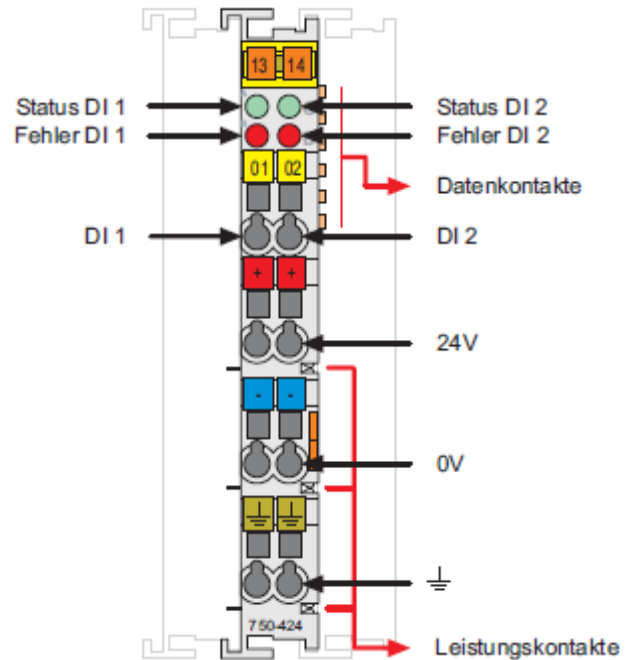


Abbildung 9.2: 2 DI DC 24 V, Einbruchsmeldung
(Schematischer Aufbau)

Die Einbruchsmeldung verfügt über zwei digitale Eingangskanäle und kann Meldekontakte wie z.B. Fensterkontakte überwachen. Es können zwei Meldekontakte in 2-Leiter Technik an die Klemme angeschlossen werden. Die grüne Status-LED signalisiert den Schaltzustand der Meldekontakte. Die zusätzliche rote Status-LED meldet einen Kurzschluss oder einen Drahtbruch.

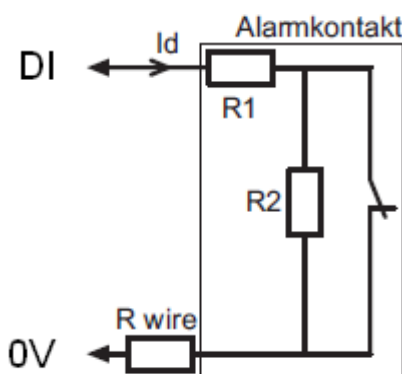


Abbildung 9.3: 2 DI DC 24 V,
Einbruchsmeldung (Alarmkontakt)

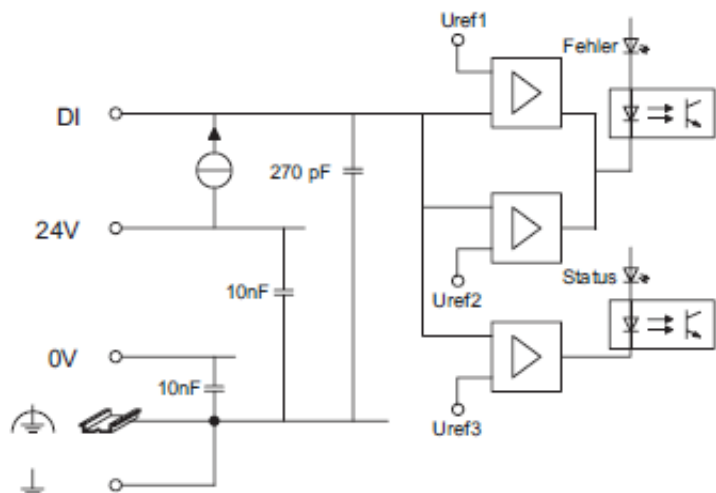


Abbildung 9.4: 2 DI DC 24 V, Einbruchsmeldung (Kontaktplan)

6.2.8 Bus-Endklemme 750-600



Abbildung 10.1: Bus-Endklemme
(750-600)

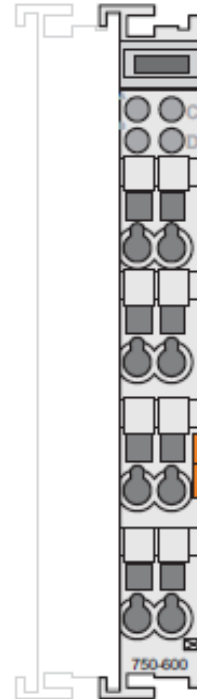


Abbildung 10.2: Bus-
Endklemme (Schematischer
Aufbau)

Die Bus-Endklemme sitzt am Ende des Feldbusknotens und dient dazu diesen ordnungsgemäß abzuschließen.

6.3 Wago Ethernet Settings

Damit das Programm später über die Ethernet -Schnittstelle auf den Wago KNX-IP Controller geladen werden kann, muss dieser zunächst über die Kommunikationsschnittstelle parametrisiert werden. Mit Hilfe des Softwaretools „WAGO Ethernet Settings“ kann dem Feldbuscontroller eine IP-Adresse zugewiesen werden.

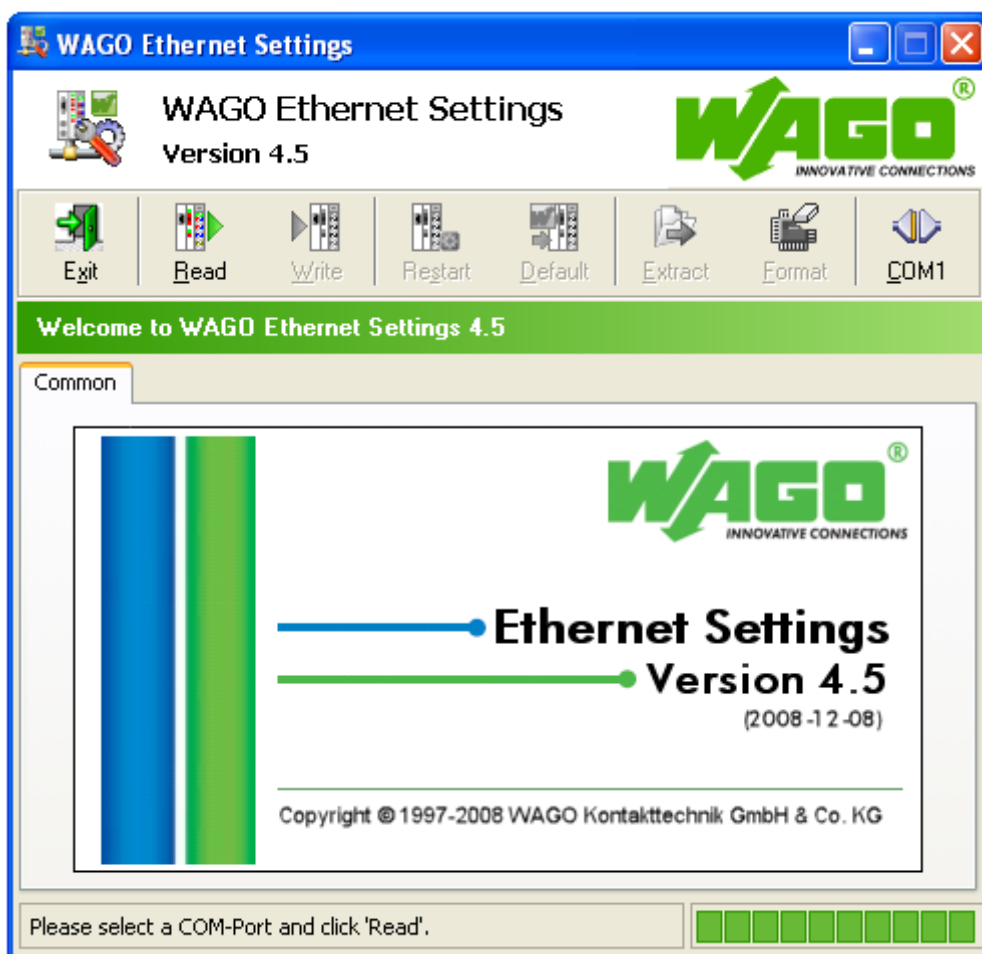


Abbildung 11.1: Ethernet Settings

6.3.1 Eine IP-Adresse zuweisen

Nach dem Öffnen des Programms muss zunächst bestimmt werden, über welche serielle Schnittstelle der Feldbuscontroller parametrisiert werden soll.

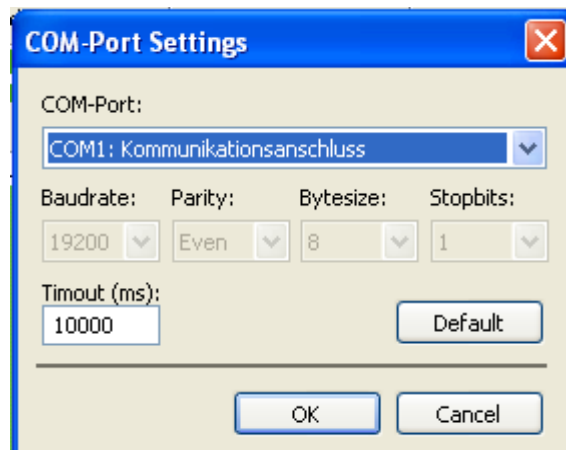


Abbildung 11.2: Ethernet Settings (Com Port Einstellungen)

Sobald die Verbindung aufgebaut ist, werden die aktuellen Daten des wie zum Beispiel Softwareversion, Mac-Adresse oder Seriennummer des Feldbuscontrollers angezeigt.

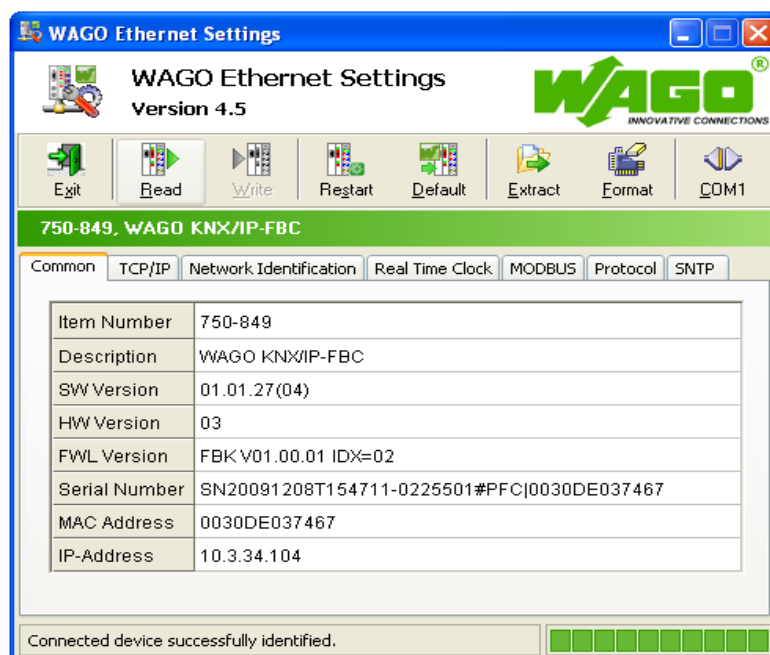


Abbildung 11.3: Ethernet Settings (Verbindungsdaten)

Über die Karteikarte „TCP/IP“ gelangt man zu dem Fenster für die IP-Adressen Einstellungen.



Abbildung 11.4: Ethernet Settings (TCP/IP Einstellungen)

Hier kann nun unter anderem auch die IP-Adresse vergeben werden. Die „Subnet Mask“ bestimmt, wie Netzwerk und Geräteteil der IP-Adresse voneinander getrennt werden. Wird der Feldbuscontroller später über einen Router oder einen Switch mit dem IP-Netzwerk verbunden, muss die Adresse des Routers bzw. des Switches im Feld „Gateway“ eingetragen werden. Für den Fall, dass das Gateway über einen DNS Server verfügt, kann zusätzlich noch die Adresse dieses Servers im Feld „Preferred DNS-Server“ angegeben werden.

6.4 Die CoDeSys Software

Die CoDeSys (Controller Development System) ist eine Entwicklungsumgebung für Speicherprogrammierbare Steuerungen und wurde von dem 1994 gegründeten Softwarehersteller „3S-Smart Software Solutions“ aus Kempten entwickelt und vermarktet. Gegenwärtig benutzen mehr als 200 Hersteller das CoDeSys Programmiersystem für ihre Hardware-Komponenten und Automatisierungsgeräte. Für die Programmierung stehen in der CoDeSys alle fünf von der IEC 61131-3 spezifizierten Sprachen zu Verfügung:

AWL: Anweisungsliste

KOP: Kontaktplan

FUP: Funktionsplan

AS: Ablaufsprache

ST: Strukturierter Text

CFC: Continuous Function Chart

6.4.1 Die Programmierung der WAGO-SPS mit der CoDeSys

6.4.1.1 Ein Zielsystem auswählen

Beim erstellen eines neuen Programms, muss zuerst das Zielsystem (750-849) ausgewählt werden, auf dem später das Programm geladen werden soll.

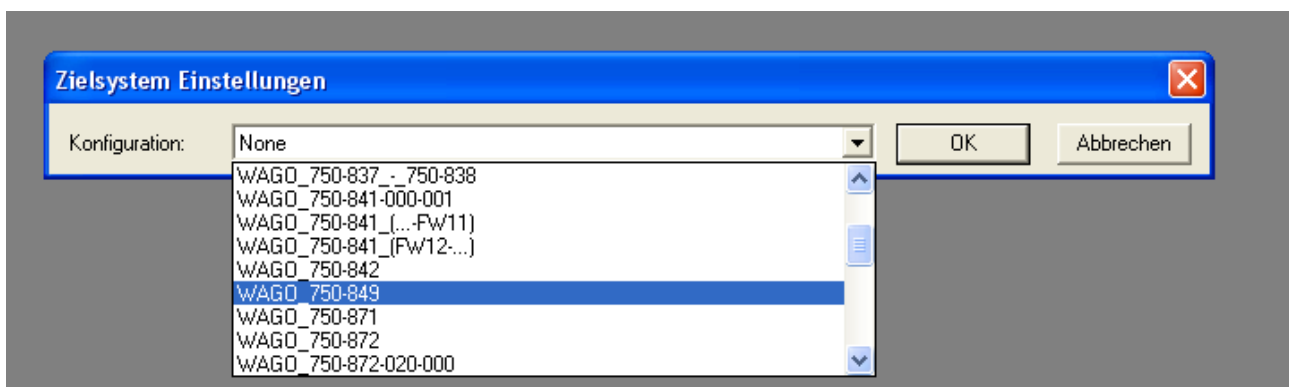


Abbildung 12.1: Codesys (Zielsystem auswählen)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Wurde ein Zielsystem gewählt, öffnet sich ein Fenster, in dem das Zielsystem zunächst Konfiguriert werden kann.

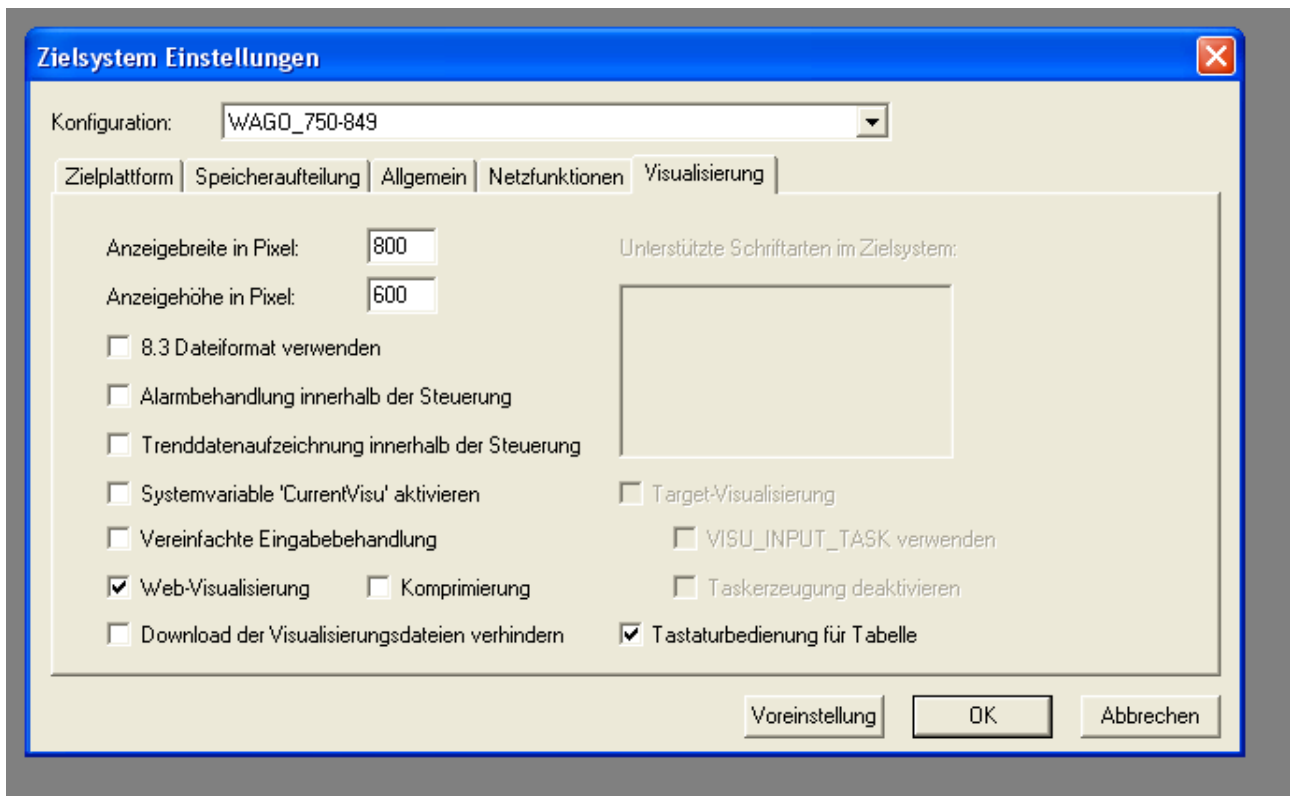


Abbildung 12.2: CoDeSys (Zielsystem Einstellungen)

6.4.1.2 Ein Programm erstellen

Nachdem das Zielsystem ausgewählt und parametriert wurde, öffnet sich das Fenster zum erstellen eines Programms. Auf der Linken Seite des Fensters kann der Benutzer wählen, ob er ein Programm, ein Funktionsblock oder eine Funktion programmieren möchte. Auf der rechten Seite des Fensters wird der Name des Programms und die Programmiersprache ausgewählt.

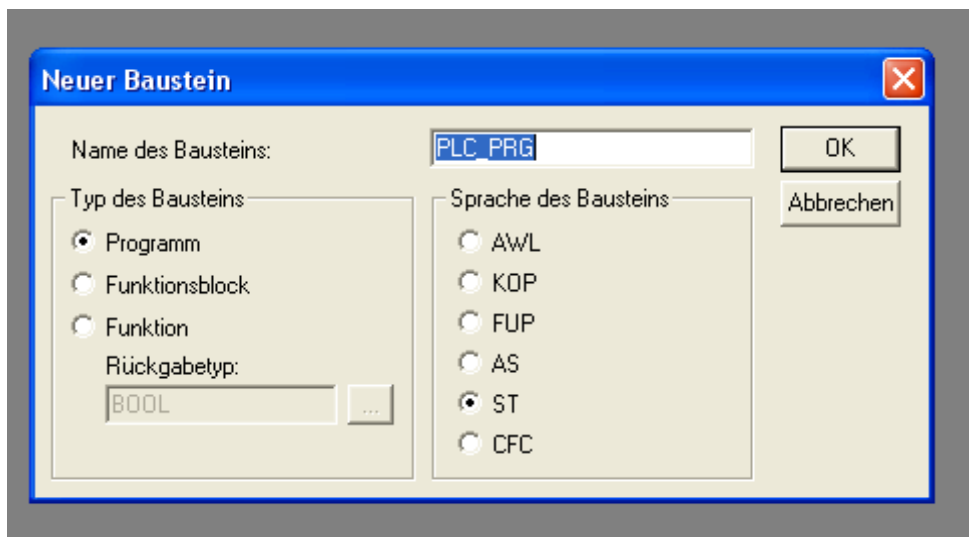


Abbildung 12.3: CoDeSys (Programm erstellen)

6.4.1.2 Die Programmiersprachen von CoDeSys

Im nächsten Schritt öffnet sich das Fenster zum erstellen eines Programms. Auf der Linken Seite des Fensters kann der Benutzer wählen, ob er ein Programm, ein Funktionsblock oder eine Funktion programmieren möchte. Auf der rechten Seite des Fensters wird der Name des Programms und die Programmiersprache ausgewählt.

6.4.1.3.1 Anweisungsliste (AWL)

Die Anweisungsliste ist eine Programmiersprache, die der Assemblersprache sehr ähnlich ist. Ein Programm, welches mit AWL geschrieben wird, beginnt in der Regel immer damit, dass ein Operand in den Akkumulator (Zwischenspeicher) geladen wird. Dies geschieht mit der Operation „LD“. Der Wert, der sich nun im Akkumulator befindet, dient nun als Parameter für die nächste Operation. Zu Schluss kann der Wert mit dem Befehl „ST“ in eine Variable gespeichert werden.

Beispiel: Zwei Zahlen (3 und 4) sollen miteinander multipliziert werden.

0001	PROGRAM	beispiel	
0002	VAR		
0003		Ergebnis: USINT;	
0004	END VAR		
			
0001	LD	3	(*Der Wert 3 wird in den Akkumulator geladen*)
0002	MUL	4	(*Der aktuelle Wert im Akkumulator (3) wird mit dem Wert 4 multipliziert*)
0003	ST	Ergebnis	(*Das Ergebnis (12) wird in die Variable "Ergebnis" gespeichert*)

Abbildung 12.4: CoDeSys (Anweisungsliste)

6.4.1.3.2 Kontaktplan (KOP)

Der Kontaktplan ist eine grafische Programmiersprache die einer Zeichnung von einem elektrischen Stromkreis sehr ähnlich ist. Über parallele oder serielle Kontakte können einfache Verknüpfungen wie „UND“ und „ODER“ realisiert werden. Auch Negationen sind mit dem Kontaktplan möglich.

Beispiel: Eine Lampe soll eingeschaltet werden, sobald „Taster1“ oder „Taster2“ betätigt werden.

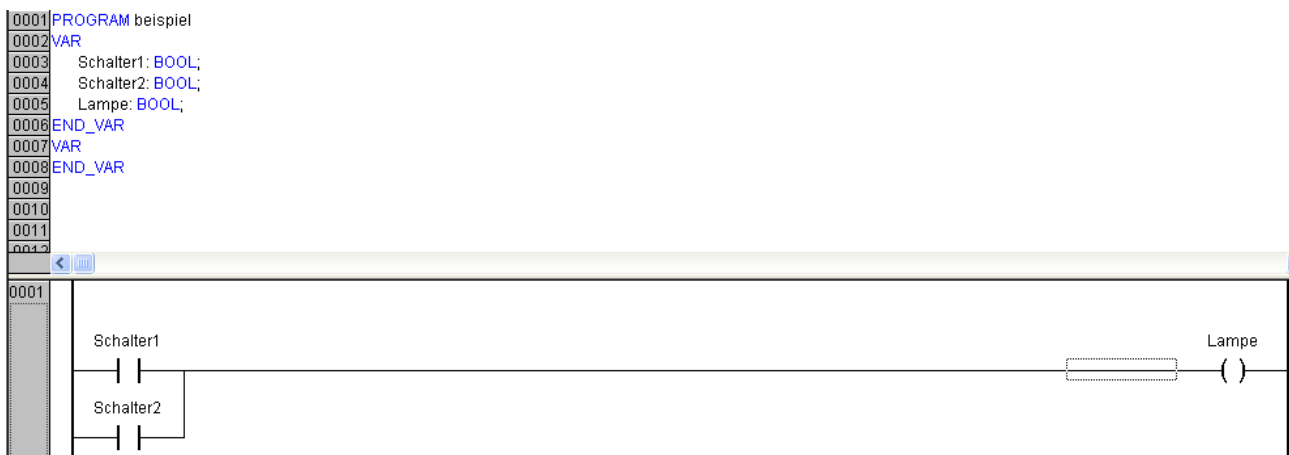


Abbildung 12.5: CoDeSys (Kontaktplan)

6.4.1.3.3 Funktionsplan (FUP)

Genau wie beim Kontaktplan handelt es sich bei dem Funktionsplan ebenfalls um eine grafische Programmiersprache. Mit dem Funktionsplan kann der Benutzer recht zügig programmieren, weil die Verbindungen automatisch gezeichnet werden. Die Verbindungen werden beim Einfügen der Bausteine automatisch gezeichnet. Die Ein- und Ausgänge der Bausteine und Funktionsblöcke können mit booleschen und analogen Werten oder Variablen belegt werden.

Beispiel: Eine Lampe soll eingeschaltet werden, sobald „Taster1“ oder „Taster2“ betätigt werden.

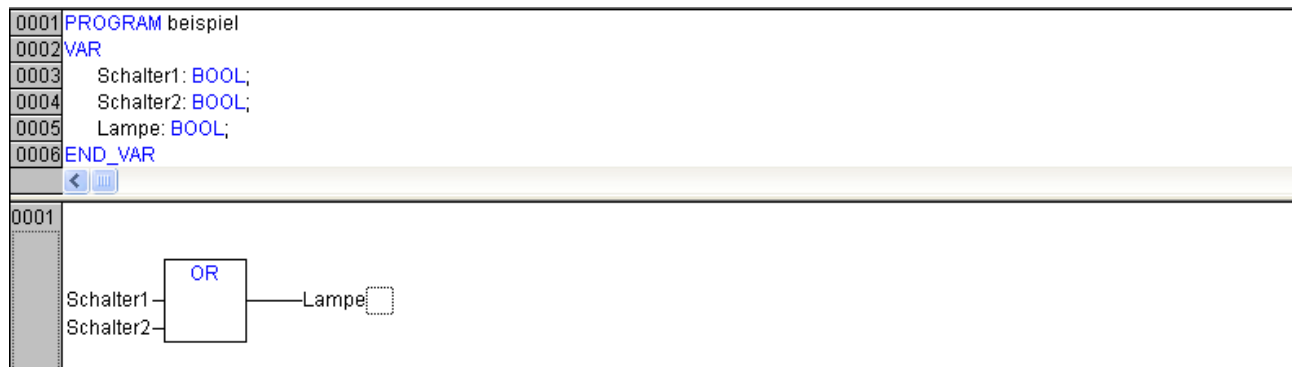


Abbildung 12.6: CoDeSys (Funktionsplan)

6.4.1.3.4 Ablaufsprache (AS)

Die Ablaufsprache dient in erster Linie zur Strukturierung eines Programms. Hierbei können Programme in sogenannten Steps (Schritten) abgelegt werden. Die einzelnen Schritte werden durch Bedingungen (Transitionen) aktiviert.

Beispiel: Durch die Betätigung eines Tasters sollen drei Lampen nacheinander in zeitlichen Abständen von 10 Sekunden geschaltet werden.

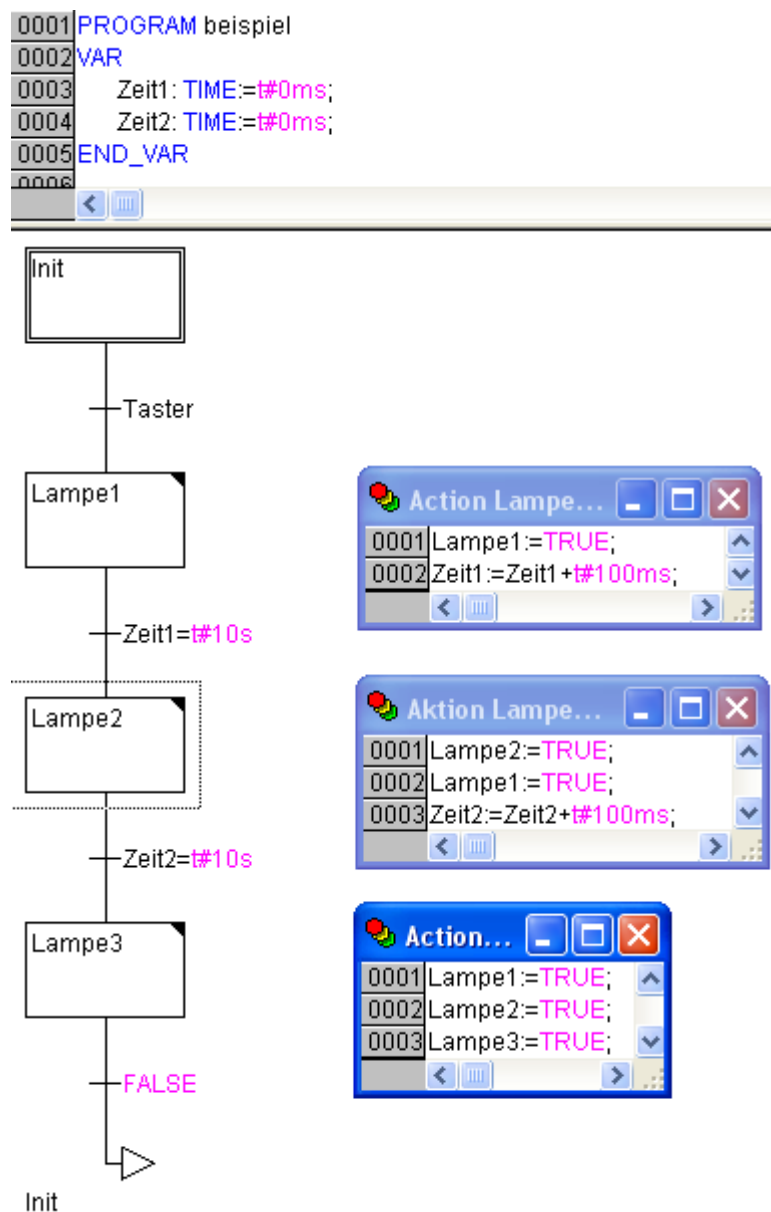


Abbildung 12.7: CoDeSys (Ablaufsprache)

Die Starttransition ist der Taster. Sobald er Betätigt wird und den Wert „TRUE“ erhält, startet das Programm. Im ersten Schritt (Lampe1) wird die erste Lampe eingeschaltet und die Zeit immer um 100 ms erhöht (wichtig: Bei dieser Funktion muss die Zykluszeit des Programms berücksichtigt werden). Sind 10 Sekunden vergangen ist die Bedingungen für die zweite Transition (Zeit1) erfüllt und der zweite Schritt (Lampe2) wird gestartet. In diesem Schritt muss Lampe2 eingeschaltet werden und Lampe1 soll weiterhin leuchten. Sobald wieder 10 Sekunden vergangen sind, ist die nächste Transition erfüllt (Zeit2). Im letzten Schritt (Lampe3), wird die dritte Lampe eingeschaltet und die anderen beiden Lampen sollen weiterhin leuchten. Die letzte Transition muss den Wert „FALSE“ haben, damit das Programm stoppt und nicht wieder von vorne beginnt.

6.4.1.3.5 Strukturierte Text (ST)

Ähnlich wie die Anweisungsliste ist ST eine Text basierte Programmiersprache die sich aber von ihrer Syntax und Semantik sehr an PASCAL oder C orientiert. In ST kann der Benutzer Schleifen wie IF, WHILE, CASE, oder FOR programmieren.

Beispiel: Eine Lampe soll eingeschaltet werden, sobald „Taster1“ oder „Taster2“ betätigt werden.

```
0001 PROGRAM beispiel
0002 VAR
0003     Taster1:BOOL;
0004     Taster2:BOOL;
0005     Lampe:BOOL;
0006 END_VAR
0007
0001 IF Taster1=TRUE OR Taster2=TRUE THEN
0002     Lampe:=TRUE;
0003 ELSE
0004     Lampe:=FALSE;
0005 END_IF
```

Abbildung 12.8: CoDeSys (Strukturierter Text)

6.4.1.3.6 Frei grafischer Funktionsplan-Editor (Continuous Function Chart CFC)

Der Frei grafische Funktionsplan-Editor funktioniert ähnlich wie der Funktionsplan. Es werden Funktionsblöcke oder Bausteine eingefügt und bearbeitet. Der Unterschied zum FUP ist jedoch, dass die Verbindungen zwischen den Ein- und Ausgängen nicht automatisch gesetzt werden, sondern vom Benutzer manuell gezogen werden. Das ist zwar zeitaufwändiger, jedoch ist die Programmierung wesentlich flexibler.

Beispiel: Eine Lampe soll eingeschaltet werden, sobald „Taster1“ oder „Taster2“ betätigt werden.

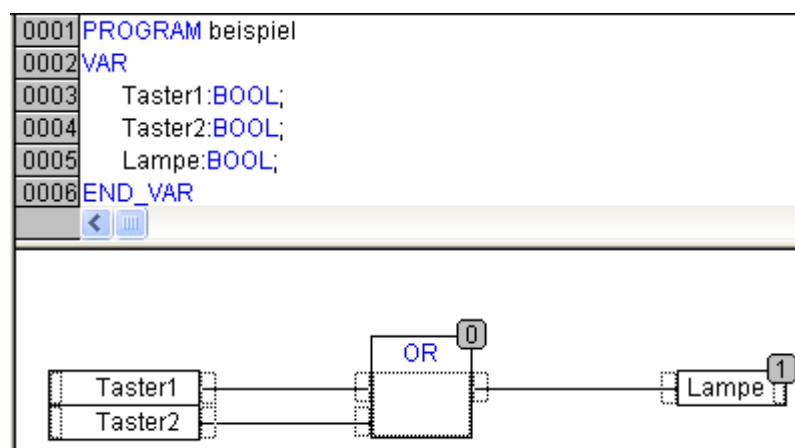


Abbildung 12.9: CoDeSys (Freigrafischer Funktionsplan Editor)

6.4.1.4 Programmierung einer Funktion

Kommen Ereignisse wie zum Beispiel eine bestimmte Berechnung mehrmals innerhalb eines Programms vor, dann ist es sinnvoll diese in eine Funktion zu programmieren. Eine Funktion liefert genau einen Rückgabewert in Abhängigkeit der Eingabewerte.

Beispiel: Drei Zahlen sollen miteinander addiert werden.

Funktion erstellen:

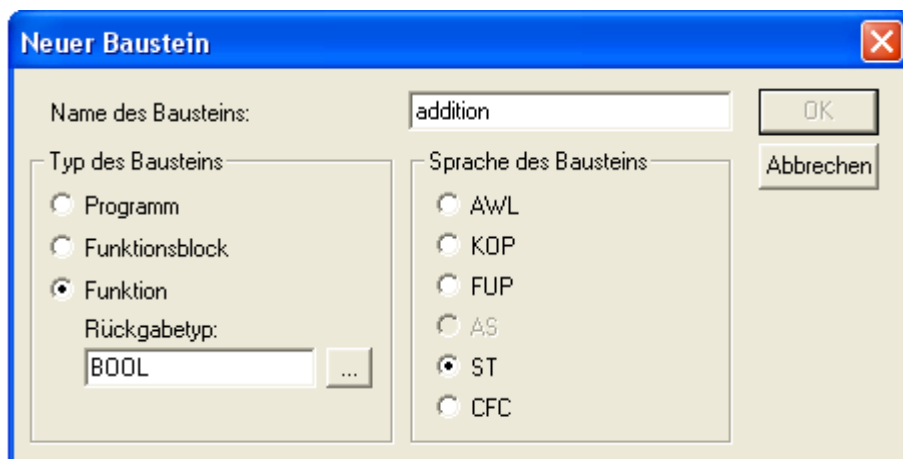


Abbildung 12.10: CoDeSys (Funktion erstellen)

Variablen deklarieren:

```
0001 FUNCTION addition : INT
0002 VAR_INPUT
0003     Zahl1:INT;
0004     Zahl2:INT;
0005     Zahl3:INT;
0006 END_VAR
```

Abbildung 12.11: CoDeSys
(Funktion Variablen)

Beim deklarieren der Variablen ist darauf zu achten, dass die Funktion einen Variablentyp haben muss. Das bedeutet, hinter dem Funktionsnamen muss zusätzlich noch der Variablentyp des

Rückgabewertes angegeben werden.

Programmierung:

```
0001 addition:=Zahl1+Zahl2+Zahl3;
```

Abbildung 12.11: CoDeSys
(Funktion Quelltext)

Funktionsaufruf:

Sollen nun die Zahlen „1, 6 und 8“ addiert werden muss die Funktion über ein Programm aufgerufen werden. Dies geschieht je nach Programmiersprache unterschiedlich.

Strukturierter Text:

```
0001 Ergebnis:=addition(1, 6, 8);
```

Abbildung 12.12: CoDeSys
(Funktionsaufruf in ST)

Freigrafischer Funktionsplan-Editor:

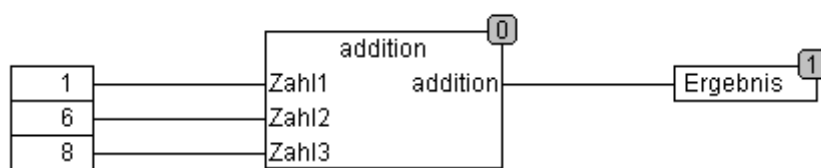


Abbildung 12.13: CoDeSys (Funktionsaufruf CFC)

Anweisungsliste:

```
0001 LD 1
0002 addition 6, 8
0003 ST Ergebnis
```

Abbildung 12.14: CoDeSys
(Funktionsaufruf AWL)

6.4.1.5 Programmieren eines Funktionsblockes

Im Gegensatz zur einer Funktion, liefert der Funktionsblock keinen Rückgabewert sondern besteht aus mehrern Ein- und Ausgängen (Input-Variablen, Output-Variablen). Die CoDeSys stellt mehrere Bibliotheken zur Verfügung, wo bereits vorprogrammierte Funktionsblöcke enthalten sind, allerdings können auch eigene Funktionsblöcke programmiert werden. Dies kann in größeren Projekten sehr vorteilhaft sein, da bestimmte Funktionen unter Umständen öfters vorkommen.

Beispiel: In einem Bürogebäude befinden sich auf einer Etage 25 Doppelbüros. Diese Büros haben jeweils eine Trennwand, die für Konferenzzwecke mit einem speziellen Schalter geöffnet werden kann. Jedes Einzelbüro hat eine türseitige und eine fensterseitige Lampe sowie zwei Taster an der Eingangstür, welche die beiden Lampen umschalten.

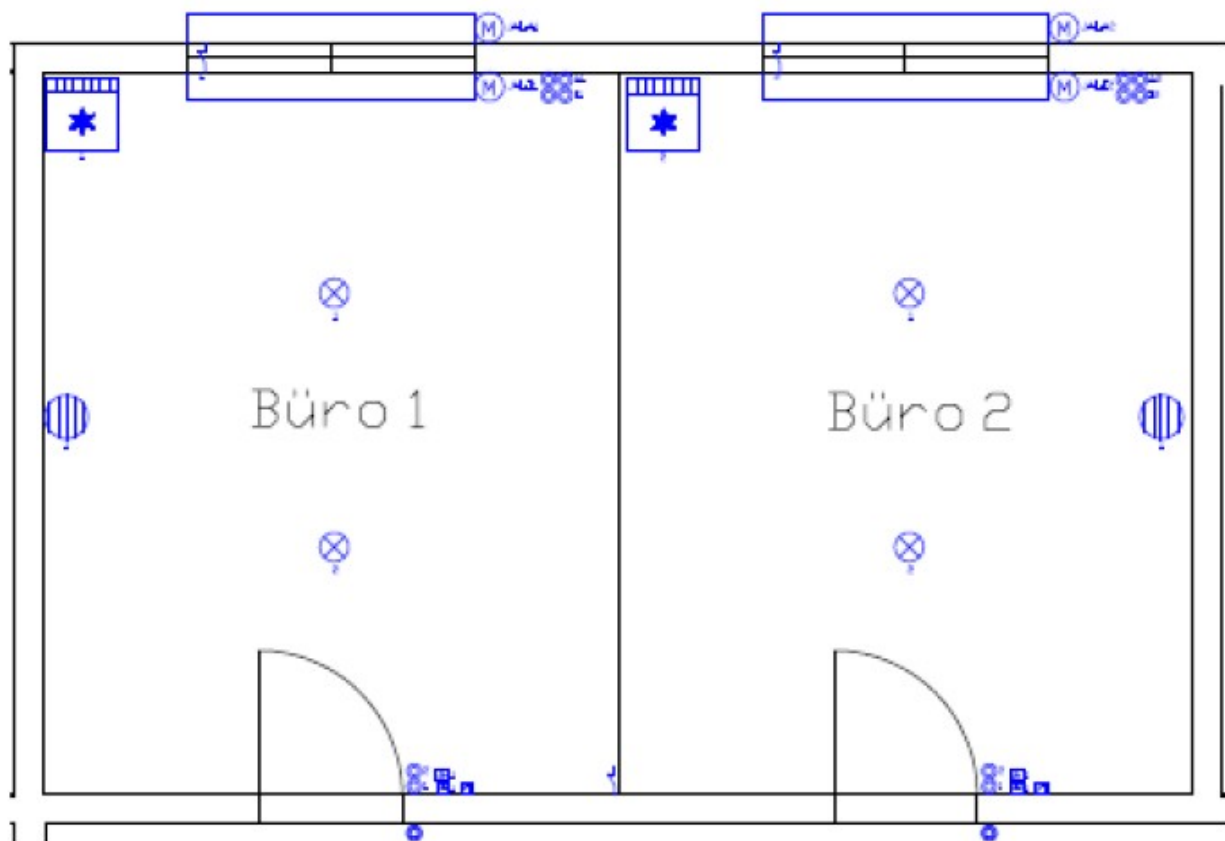


Abbildung 12.16: CoDeSys (Funktionsblock Büro)

Wird die Trennwand geöffnet, befinden sich insgesamt vier Lampen im Büro (zwei türseitige und zwei fensterseitige Lampen). Der Taster an der Eingangstür von Büro 1, welcher vorher die türseitige Lampe umgeschaltet hat, soll nun auch die türseitige Lampe von Büro 2 umschalten können. Dasselbe gilt für den Taster, welcher die vorher fensterseitige Lampe umgeschaltet hat. Dieser soll nun auch die fensterseitige Lampe von Büro 2 umschalten. Die zwei Taster an der Eingangstür von Büro 2 sollen ebenfalls die tür- und fensterseitigen Lampen beider Einzelbüros umschalten.

Nun befinden sich 25 dieser Doppelbüros auf einer Etage. Um nicht jedes einzelne Büro von neuem mit dieser Funktion zu programmieren, ist es wesentlich komfortabler die Funktion einmal als ein Funktionsblock zu schreiben.

Anhand dieses Beispiels soll nun ein Funktionsblock programmiert werden.

Zunächst wird ein neuer Baustein angelegt.

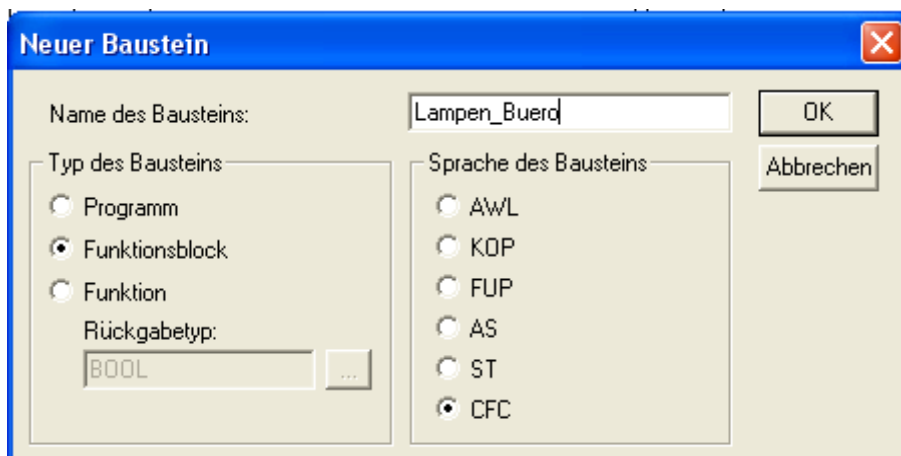


Abbildung 12.17: CoDeSys (Funktionsblock erstellen)

Auf der linken Seite wird bestimmt, dass es sich bei dem Baustein um einen Funktionsblock handelt. Auf der Rechten Seite wird der Name des Funktionsblocks bestimmt und die Sprache in der dieser programmiert werden soll. Am einfachsten ist diese Funktion in der CFC-Sprache zu programmieren.

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Nun werden die In- und Output Variablen und die allgemeinen Variablen deklariert.

```
0001 FUNCTION_BLOCK Lampen_Buero
0002 VAR_INPUT
0003     TasterRechtsLampeTuerseitig:BOOL;
0004     TasterLinksLampeTuerseitig:BOOL;
0005     TasterRechtsLampeFensterseitig:BOOL;
0006     TasterLinksLampeFensterseitig:BOOL;
0007     Trennwandschalter:BOOL;
0008 END_VAR
0009 VAR_OUTPUT
0010     LampeRechtsTuerseitig:BOOL;
0011     LampeRechtsFensterseitig:BOOL;
0012     LampeLinksTuerseitig:BOOL;
0013     LampeLinksFensterseitig:BOOL;
0014 END_VAR
0015 VAR
0016     ToggelnLampeRechtsTuerseitig:RS;
0017     ToggelnLampeRechtsFensterseitig:RS;
0018     ToggelnLampeLinksTuerseitig:RS;
0019     ToggelnLampeLinksFensterseitig:RS;
0020     TasterRechtsLampeTuerseitigTRIG:R_TRIG;
0021     TasterLinksLampeTuerseitigTRIG:R_TRIG;
0022     TasterRechtsLampeFensterseitigTRIG:R_TRIG;
0023     TasterLinksLampeFensterseitigTRIG:R_TRIG;
0024 END_VAR
```

Abbildung 12.18: CoDeSys (Funktionsblock Variablen deklarieren)

Danach wird der Funktionsblock programmiert.

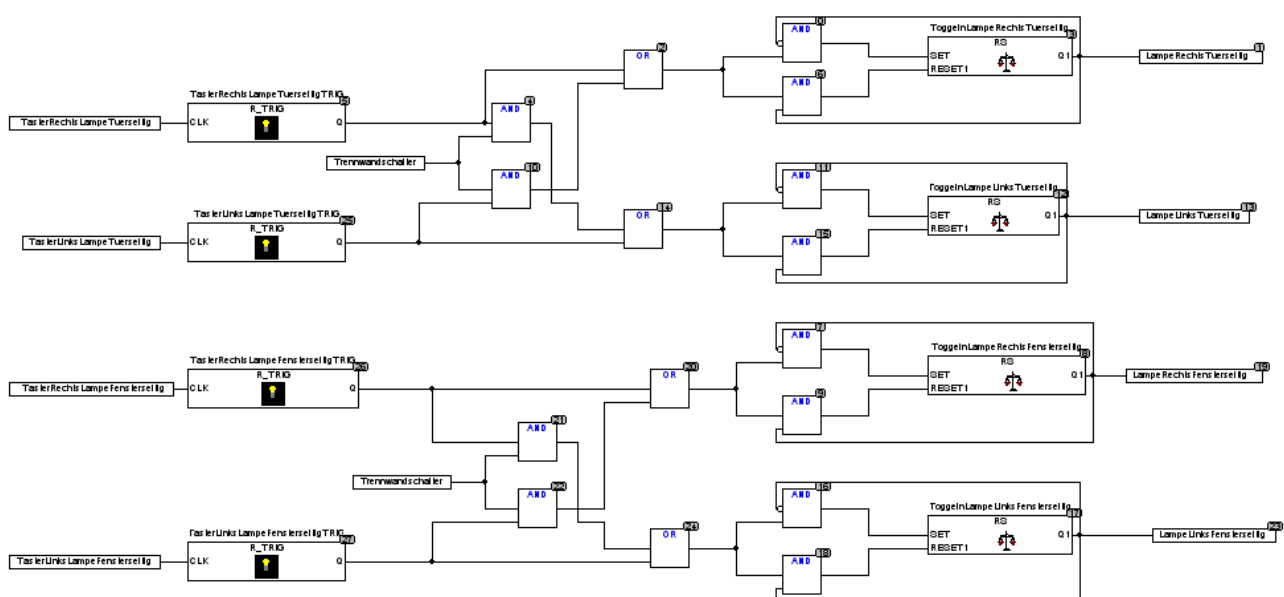


Abbildung 12.19: CoDeSys (Funktionsblock Quelltext)

Die Programmierung selber funktioniert im Prinzip genau wie bei einem normalen Programm.

Ist der Funktionsblock vollständig programmiert, kann nun das erste Büro programmiert werden.

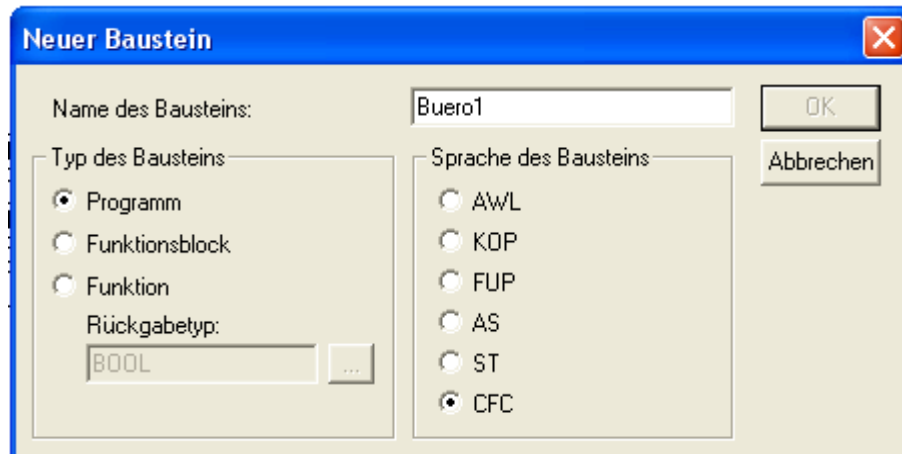


Abbildung 12.20: CoDeSys (Funktionsblock Büro erstellen)

Zunächst werden wieder die Variablen deklariert.

```
0001 PROGRAM Buero1
0002 VAR
0003     Taster_Buero1_Rechts1: BOOL;
0004     Taster_Buero1_Rechts2: BOOL;
0005     Taster_Buero1_Links1: BOOL;
0006     Taster_Buero1_Links2: BOOL;
0007     Trennwandschalter_Buero1: BOOL;
0008     Lampe_Buero1_Rechts1: BOOL;
0009     Lampe_Buero1_Rechts2: BOOL;
0010     Lampe_Buero1_Links1: BOOL;
0011     Lampe_Buero1_Links2: BOOL;
0012     Lampen_Buero1: Lampen_Buero;
0013 END_VAR
```

Abbildung 12.21: CoDeSys (Funktionsblock Büro Variablen)

Der Funktionsblock muss ebenfalls in den Variablenlisten aufgeführt werden.

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Im Programm wird nun ein neuer Funktionsbaustein eingefügt.

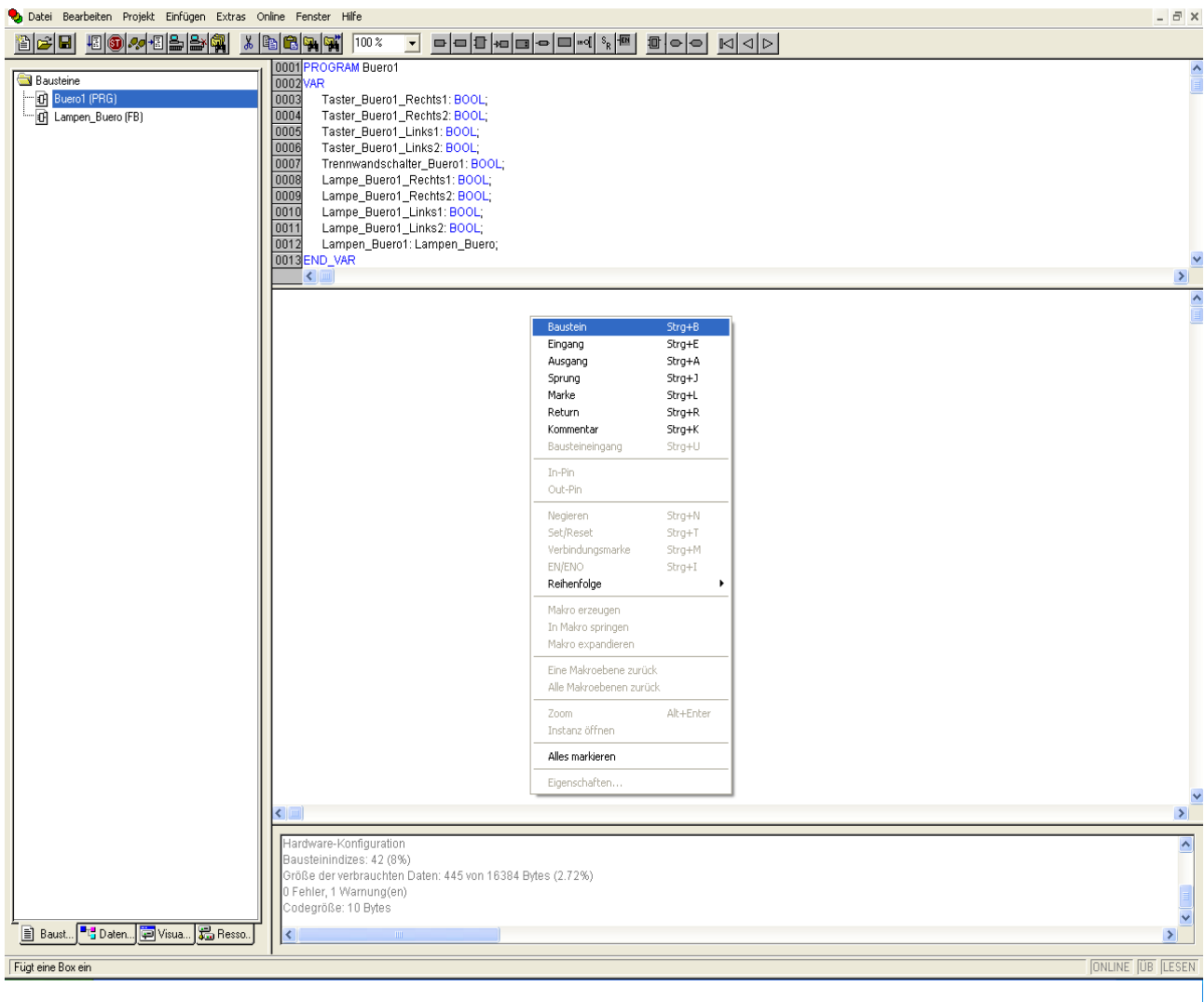


Abbildung 12.22: CoDeSys (Funktionsblock Funktionsbaustein einfügen)

Wählt man nun den Baustein aus und drückt die Taster „F2“ öffnet sich die Eingabehilfe. Der programmierte Funktionsblock befindet sich unter „definierte Funktionsblöcke“.

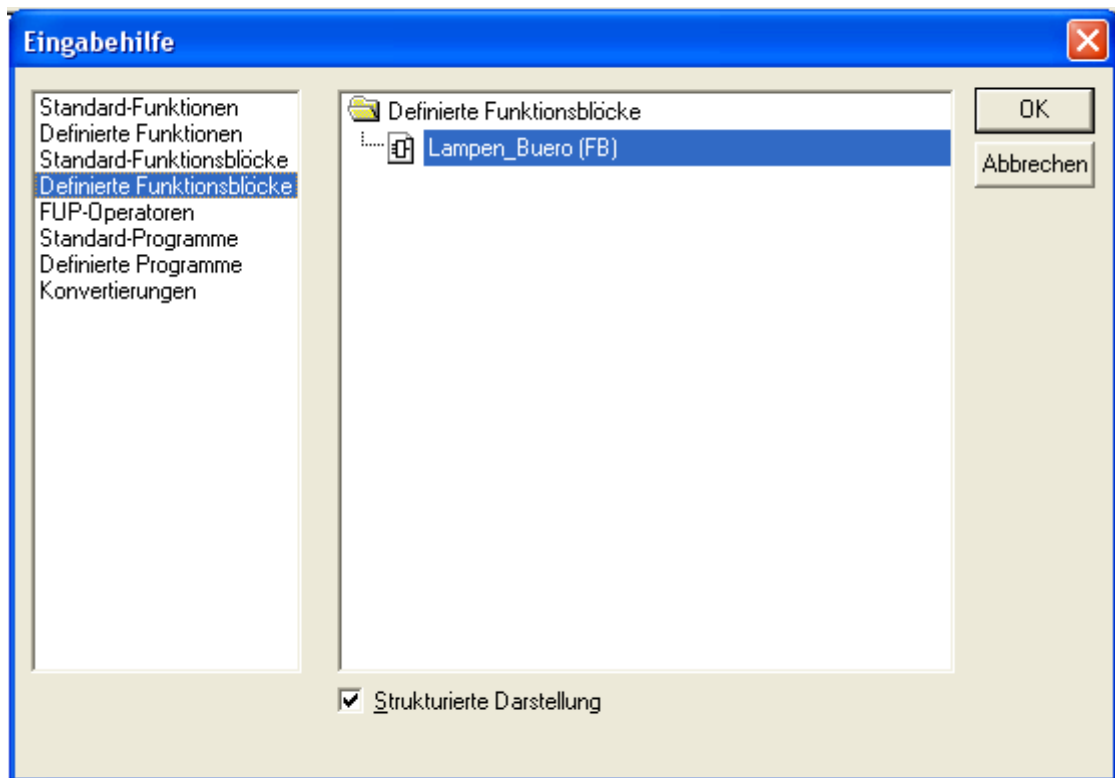


Abbildung 12.23: CoDeSys (Funktionsblock Eingabehilfe)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Nun erscheint der Funktionsblock als Baustein. Auf der rechten Seite befinden die Eingänge, die als Input-Variablen definiert wurden. Oben in der Mitte steht der Name des Funktionsblocks und rechts befinden sich die Ausgänge, die als Output-Variablen definiert wurden. Über dem Baustein wird der Name eingetragen, der für dieses Programm benutzt wird. Dieser muss zusätzlich bei den Variablen eingetragen werden.

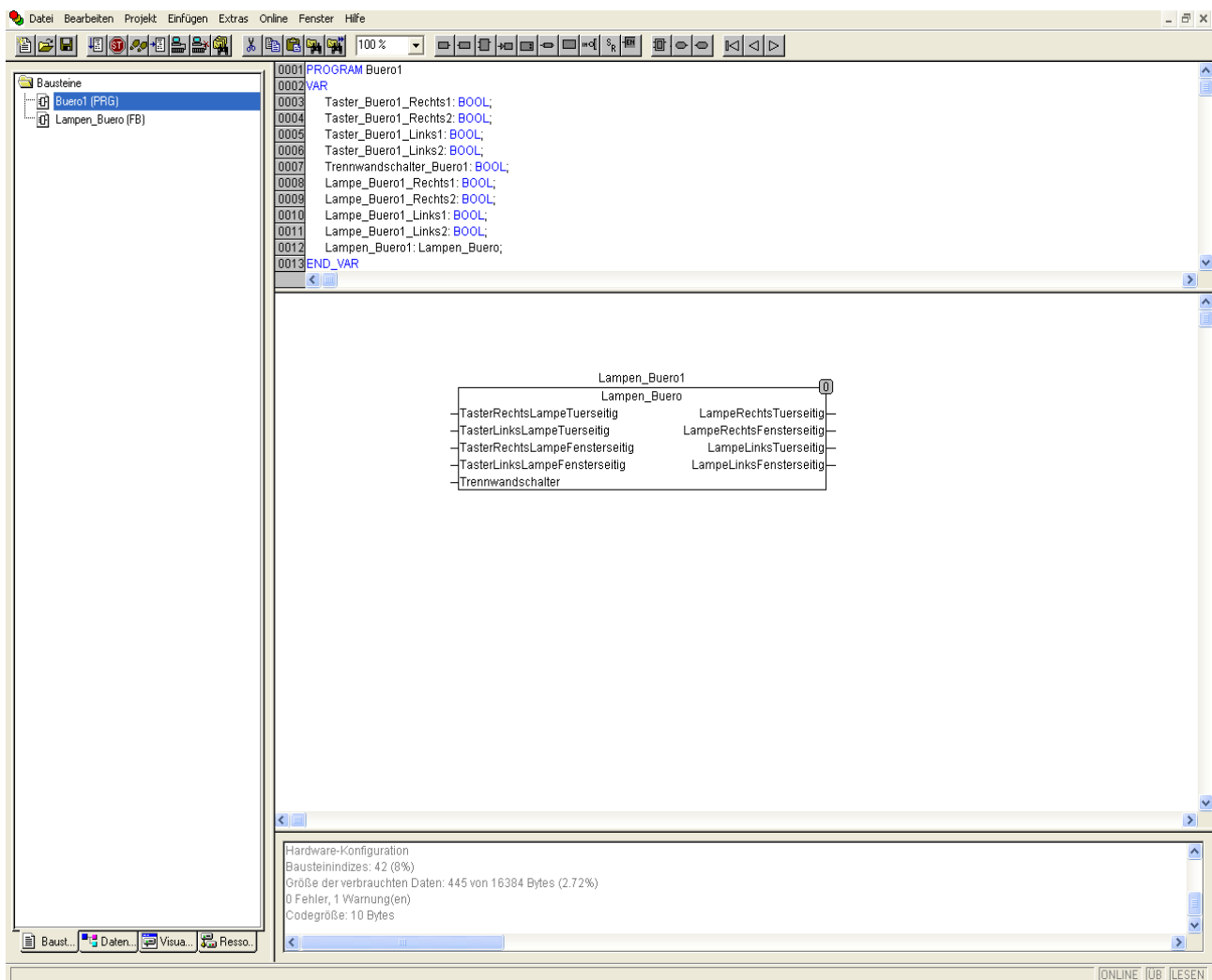


Abbildung 12.24: CoDeSys (Funktionsblock als CFC Baustein)

Nun können die Ein- und Ausgänge belegt werden.

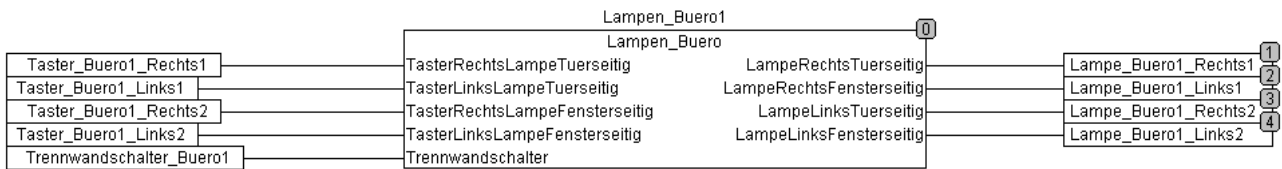


Abbildung 12.25: CoDeSys (Funktionsblock Ein-und Ausgänge)

Die restlichen 24 Büros können nun genau so programmiert werden. Dass erspart eine Menge Zeit und Arbeit.

6.4.1.6 Die Bibliotheken

Im vorherigen Kapitel wurde gezeigt, dass es bei größeren Projekten durchaus sinnvoll ist, für Aktionen die eine komplizierte Programmierung erfordern (wie zum Beispiel das Doppelbüro mit dem Trennwandschalter), Funktionsblöcke oder Funktionen anzulegen. Die Firma Wago stellt dafür komplette Bibliotheken zur Verfügung, welche bereits über vorprogrammierte Funktionsblöcke, globale Variablen und sogar Visualisierungselemente verfügen. Auch für spezielle Klemmen wie zum Beispiel der KNX-TP1-Klemme oder dem EnOcean Funkempfänger existieren Bibliotheken mit Funktionsblöcken die speziell auf die Klemmen zugeschnitten sind.

6.4.1.6.1 Der Bibliotheksverwalter

Die Bibliotheken in CoDeSys werden über einen sogenannten „Bibliotheksverwalter“ verwaltet, welcher sich unter dem Reiter „Ressourcen“ im Verwaltungsfenster befindet.

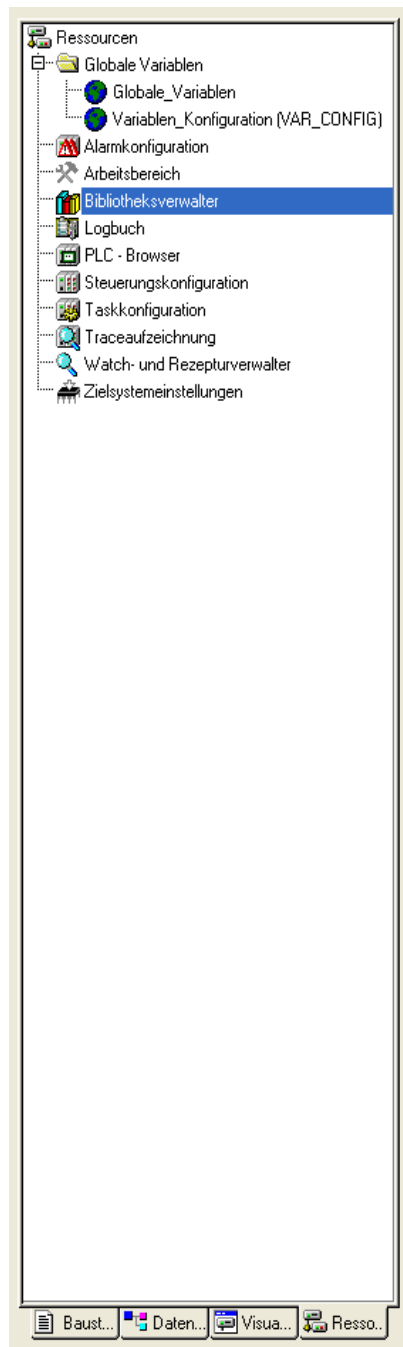


Abbildung 12.26: CoDeSys (
Bibliotheksverwalter auswählen)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Mit einem Rechtsklick auf das Feld oben Links im Bibliotheksverwalter, öffnet sich ein Fenster zum Einfügen einer weiteren Bibliothek.

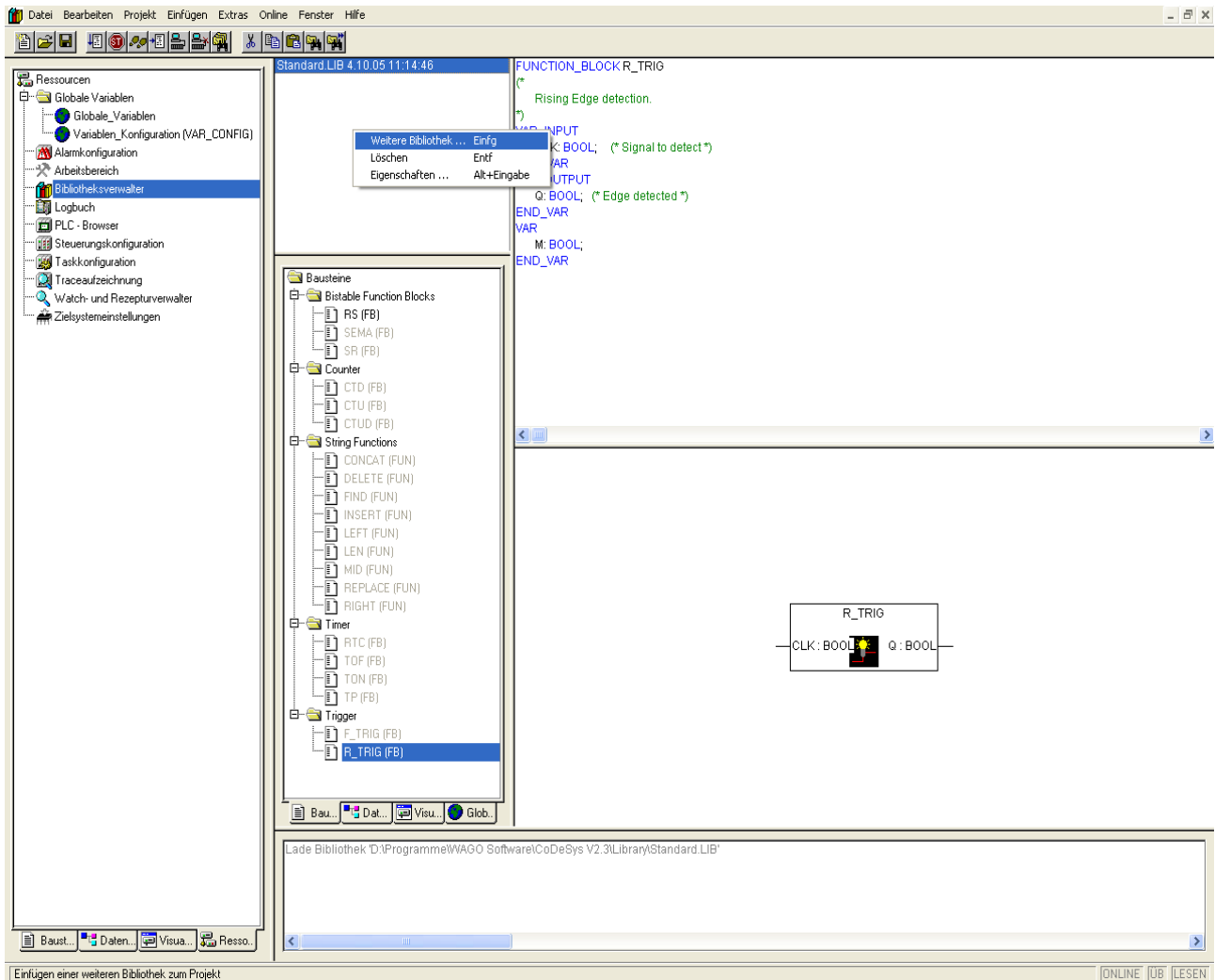


Abbildung 12.27: CoDeSys (Bibliotheksverwalter Bibliothek einfügen)

Die Wago Bibliotheken befinden sich standardmäßig im Verzeichnis „/Targets/Wago/Libraries“ und ihre Dateierweiterung ist „.LIB“.

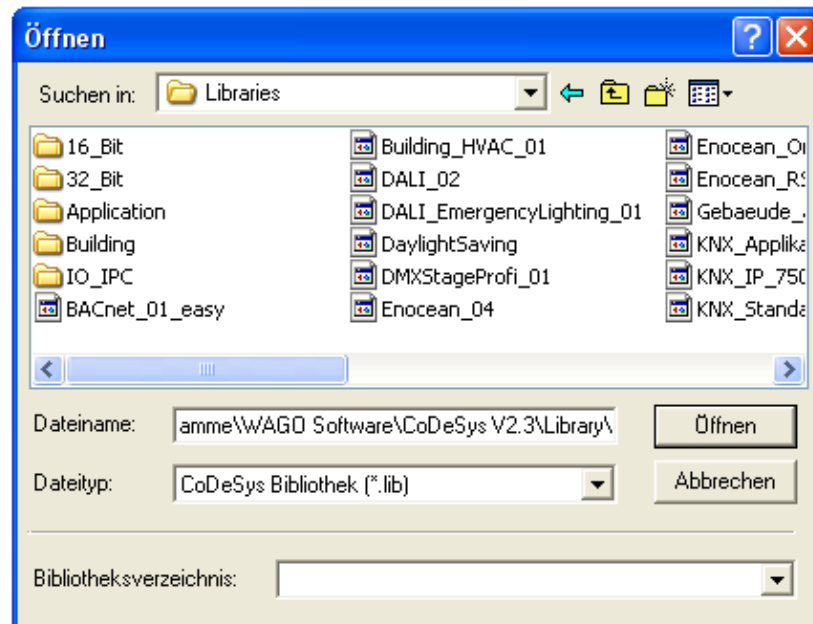


Abbildung 12.28: CoDeSys (Bibliotheksverwalter Bibliothek auswählen)

6.4.1.6.2 Arbeiten mit Bibliotheken

Als Beispiel soll nun eine einfache „Lampe_umschalten“- Funktion programmiert werden. Die Funktion wird mit Hilfe von zwei verschiedenen Bibliotheken realisiert.

1. Standard : Die Bibliothek „Standard“ wird standardmäßig von der Firma „Smart Software Solutions“ für die CoDeSys zur Verfügung gestellt. Sie beinhaltet sequentielle Binäre Schaltungen wie z.B.: RS-Flip-Flops, Auf- und Abwärtszähler oder Taktflankengesteuerte Trigger.
2. Gebäude_allgemein: „Gebäude_allgemein“ ist eine Bibliothek der Firma Wago. Sie beinhaltet vorprogrammierte Funktionen und Funktionsblöcke wie z.B.: Stromstoss, Taster kurz und lang oder Jalousiefunktionen, die innerhalb einer Gebäudeautomatisierung sehr nützlich sein können.

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

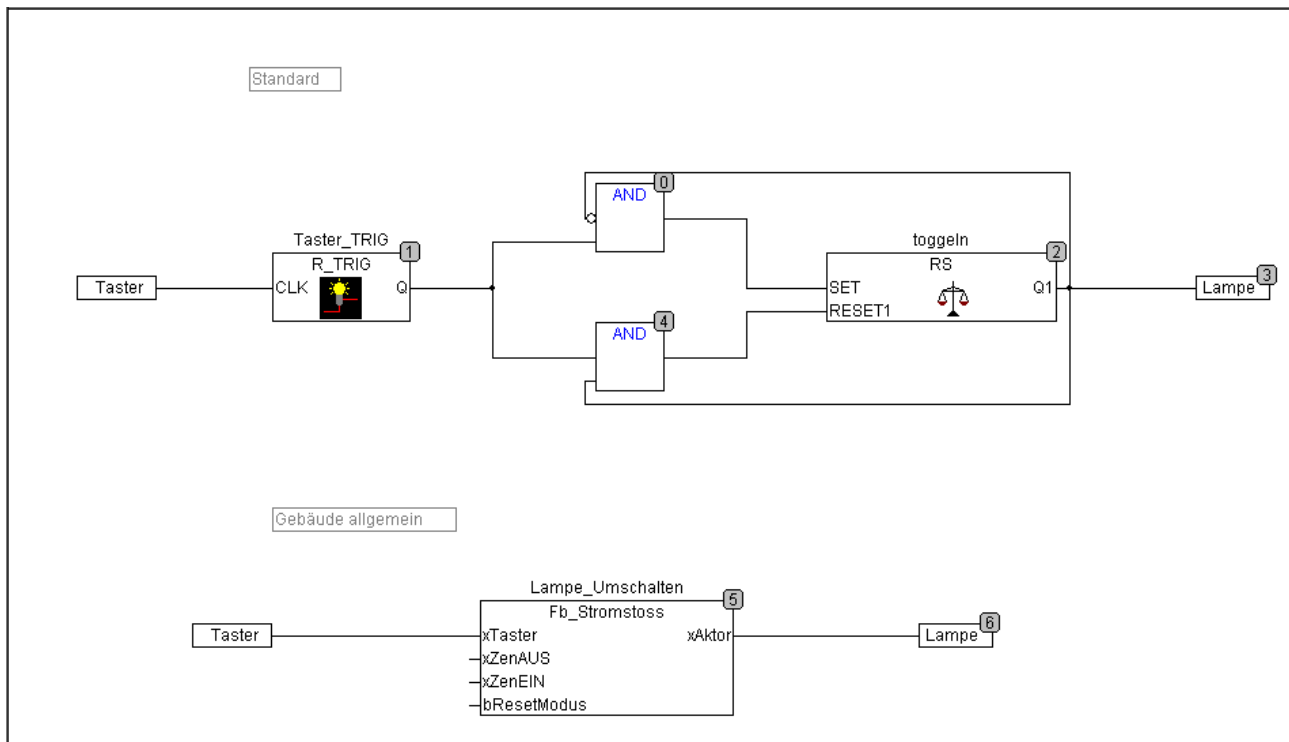


Abbildung 12.29: CoDeSys (Bibliotheken Vergleich)

Das Bild zeigt nun wie die Funktion „Lampe_umschalten“ mit beiden Bibliotheken realisiert wird. Hierbei ist deutlich zu erkennen, dass die Realisierung mit dem Funktionsblock „Stromstoss“ der Bibliothek „Gebäude_allgemein“ wesentlich komfortabler ist. Zudem besitzt der Funktionsblock „Stromstoss“ noch zusätzliche Eingänge wie z.B.: „xZenEIN“ oder „xZenAUS“ mit dem die Aktoren noch über zentrale Funktionen ein oder aus geschaltet werden können.

6.4.1.7 CoDeSys Visualisierung

Die CoDeSys verfügt über einen eigenen Visualisierungs-Editor, welcher grafische Elemente bereitstellt, die mit den Projektvariablen verknüpft werden können.

Zum Visualisierungs-Editor gelangt man über den Reiter „Visualisierung“.

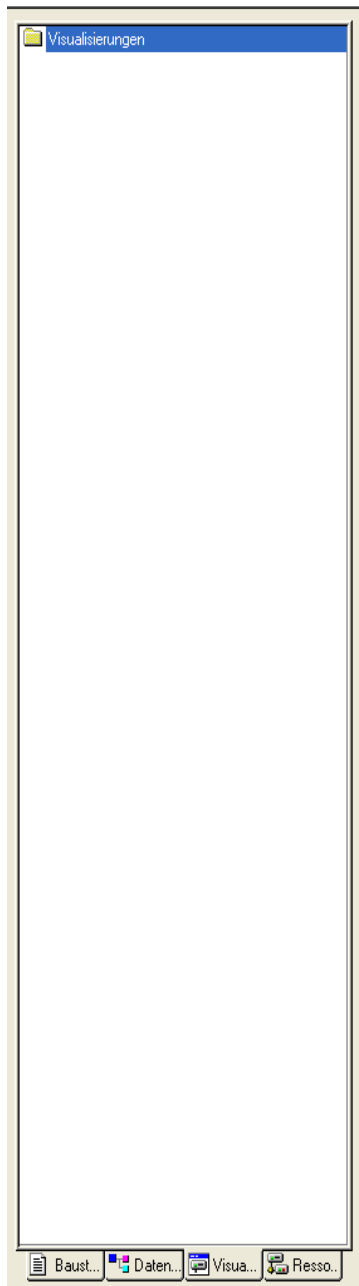


Abbildung 12.30: CoDeSys
(Visualisierungseditor aufrufen)

Mit einem Rechtsklick öffnet sich das Kontextmenü. Unter „Objekt einfügen, wird eine neues Visualisierungsobjekt angelegt.

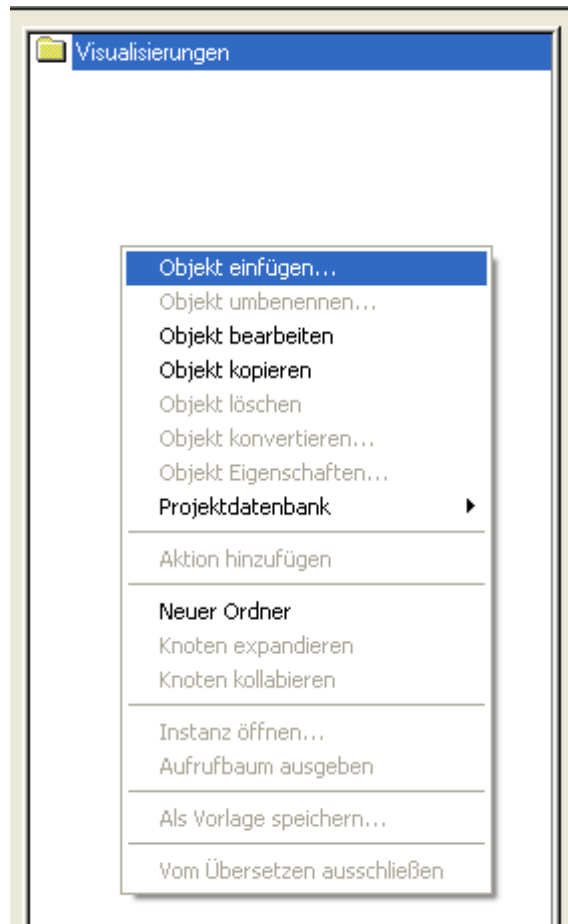


Abbildung 12.30: CoDeSys (Visualisierungsobjekt einfügen)

Die Visualisierungsoberfläche funktioniert ähnlich wie ein CAD oder Zeichenprogramm. Neben den Standardmäßigen Zeichenelementen können auch Bitmaps, ActiveX Elemente oder Windows Meta Files eingefügt werden.

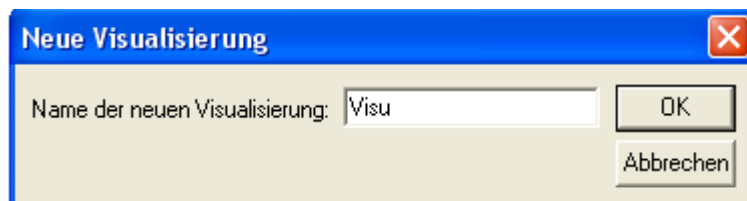


Abbildung 12.31: CoDeSys (Visualisierungsobjekt benennen)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

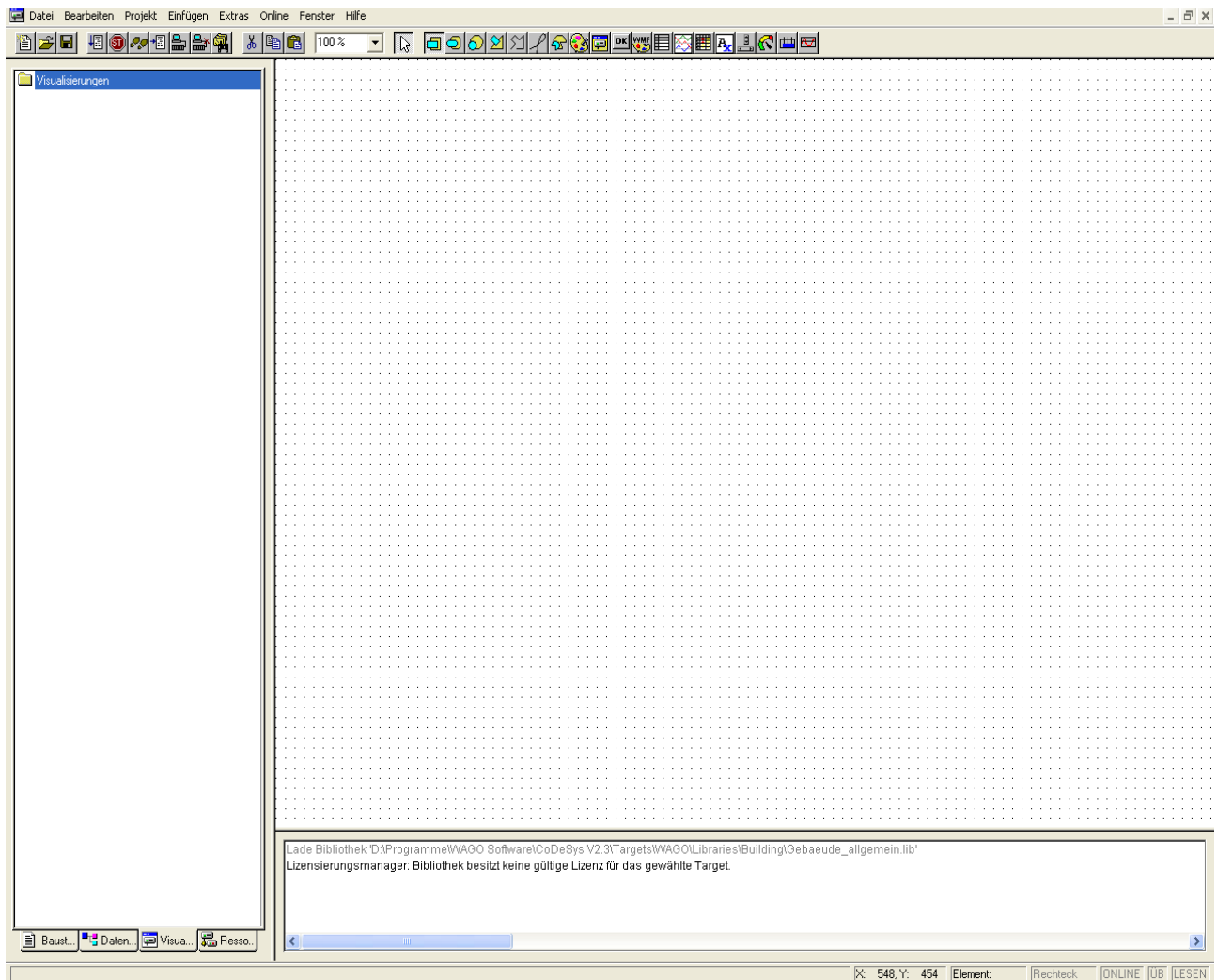


Abbildung 12.32: CoDeSys (Visualisierung Oberfläche)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Für die Visualisierung stehen folgende Objekte zur Verfügung:



Rechteck



Schaltfläche



Abgerundetes Rechteck



ActiveX-Element



Ellipse



Scrollleiste



Polygon



Tabelle



Linienzug



Zeigerinstrument



Kurve



Balkenanzeige



Kreissektor



Histogramm



Bitmap



Alarmtabelle



Visualisierung



Trend



Windows Metafile

6.4.1.7.1 Eine Visualisierung erstellen

Beispiel: Eine Schaltfläche soll eine Lampe Schalten.

1. Schaltfläche einfügen

Aus der Symbolleiste wird das Element „Schaltfläche“ ausgewählt“.



Abbildung 12.33: CoDeSys (Visualisierung Schaltfläche einfügen)

Danach wird die Schaltfläche gezeichnet.

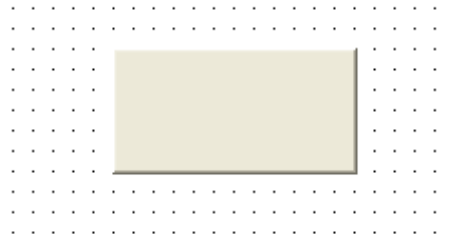


Abbildung 12.34: CoDeSys (Visualisierung Schaltfläche)

Wird nun mit der rechten Maustaste auf die Schaltfläche geklickt, öffnet sich ein Popup-Fenster. Hier wählt man nun den Befehl „Konfigurieren“ aus.

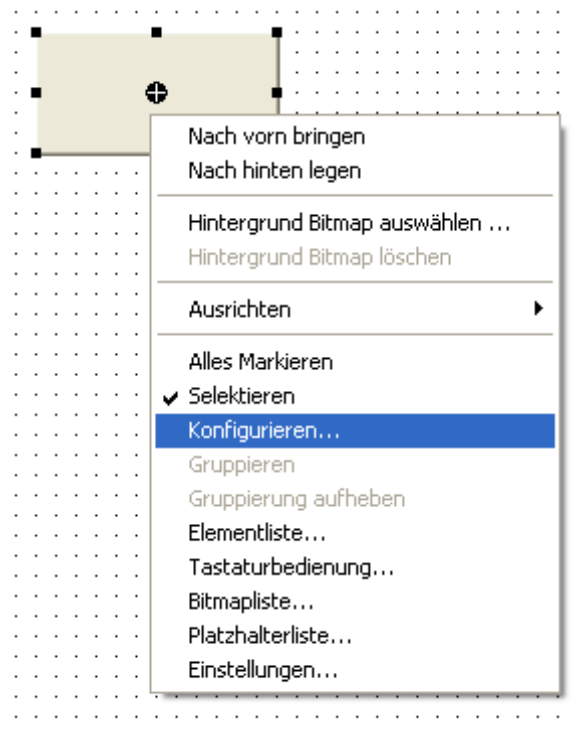


Abbildung 12.35: CoDeSys (Visualisierung Schaltfläche Konfigurieren)

Nun öffnet sich eine Fenster zu Konfigurieren des Visualisierungselementes.

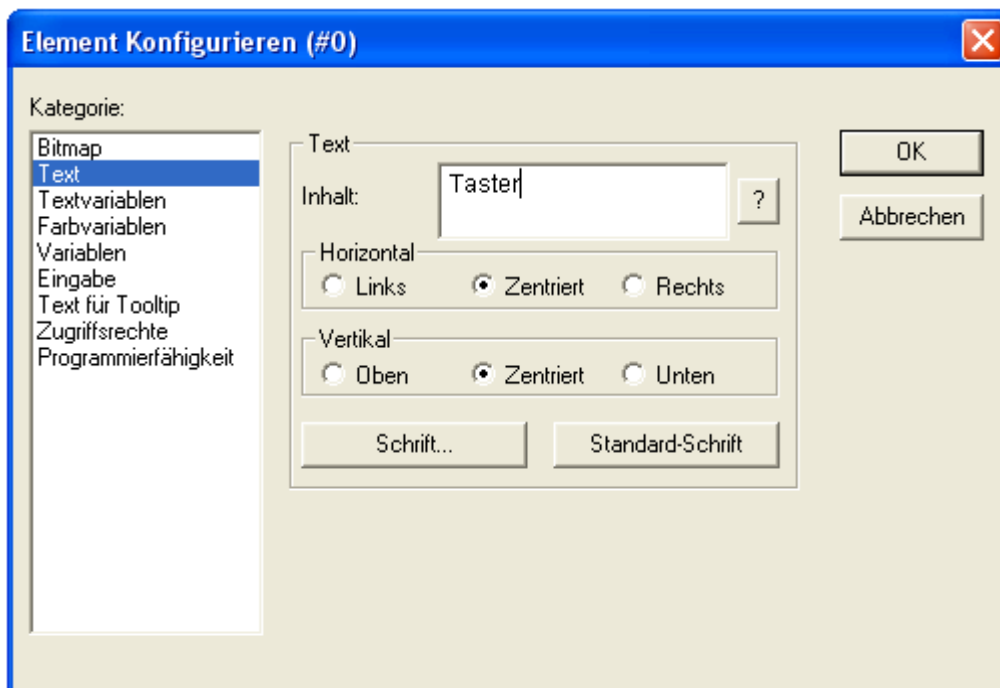


Abbildung 12.36: CoDeSys (Visualisierung Schaltfläche Text)

Hier können nun verschiedene Einstellungen vernommen werden. Zunächst wird ein Text eingefügt, welcher später an dem Element zu sehen ist dazu wählt man den Befehl „Text“ auf der Linken Seite des Fensters. Nun kann in dem Eingabefeld ein Text eingegeben werden. Zusätzlich kann die horizontale und vertikale Ausrichtung des Textes sowie Schriftgröße und Schrifttyp bestimmt werden.

Im nächsten Schritt wird der Befehl „Eingabe“ ausgewählt. Hier kann nun bestimmt werden, was passiert, wenn mit der Maus auf das Element geklickt wird.

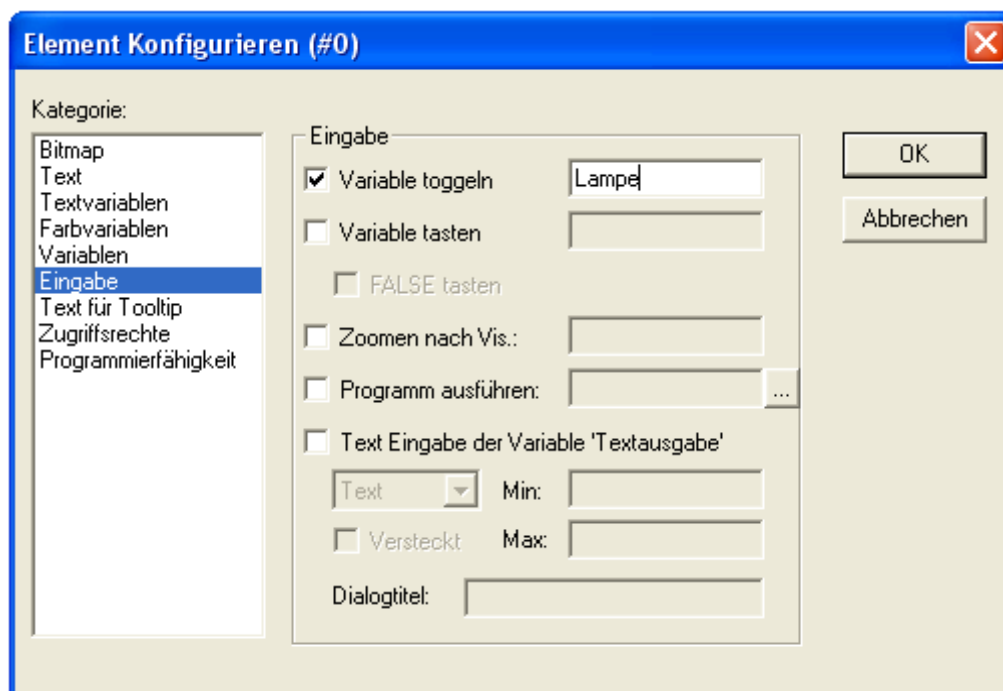


Abbildung 12.37: CoDeSys (Visualisierung Schaltfläche Eingabe)

Hier können nun verschiedene Aktionen gewählt werden. Setzt man nun einen Haken vor die gewünschte Aktion, wird das Eingabefeld aktiviert. Hier kann nun Die Variable (hier Lampe) eingefügt werden, deren Zustand beim „Klick“ auf das Element geändert werden soll (Hierzu kann auch die Eingabehilfe verwendet werden). Mit dem Befehl „Variable Toggeln“ wird der Zustand Variable gewechselt (die Aktion funktioniert nur bei Booleschen Variablen). Der Befehl Variable Tasten funktioniert ähnlich, hierbei wird die Variable ähnlich wie bei einem Taster auf den Wert „TRUE“ oder „FALSE“ (falls ein Haken an das Feld „FALSE tasten“ gesetzt wird) gesetzt. Ähnlich wie bei einer HTML-Seite kann eine Schaltfläche auch die Funktion eines „Links“ übernehmen, welcher, sobald man mit der Maus drauf klickt, ein anderes Visualisierungsfenster öffnet. Hierzu wird das gewünschte Fenster im Feld „Zoomen nach Vis“ eingefügt. Im Feld „Programm ausführen“ kann ein Programm ausgewählt werden, welches beim „Klick“ auf die Schaltfläche ausgeführt wird. Auf die Aktion „Text Eingabe der Variable 'Textausgabe'“ wird später noch einmal genauer eingegangen.

2. Farbwechsel bei einem Visualisierungselement

Eine Zustandsänderung einer booleschen Variable soll mit Hilfe eines Visualisierungselementes mit einem Farbwechsel signalisiert werden. Hierzu wird zunächst wieder ein gewünschtes Element (hier Ellipse) eingefügt.



Abbildung 12.38: CoDeSys (Visualisierung Ellipse einfügen)

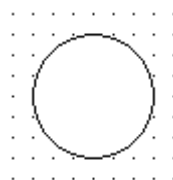


Abbildung
12.39:
CoDeSys
(Visualisierung
Ellipse)

Auch hier wird mit nun wieder das Konfigurationsfenster geöffnet.

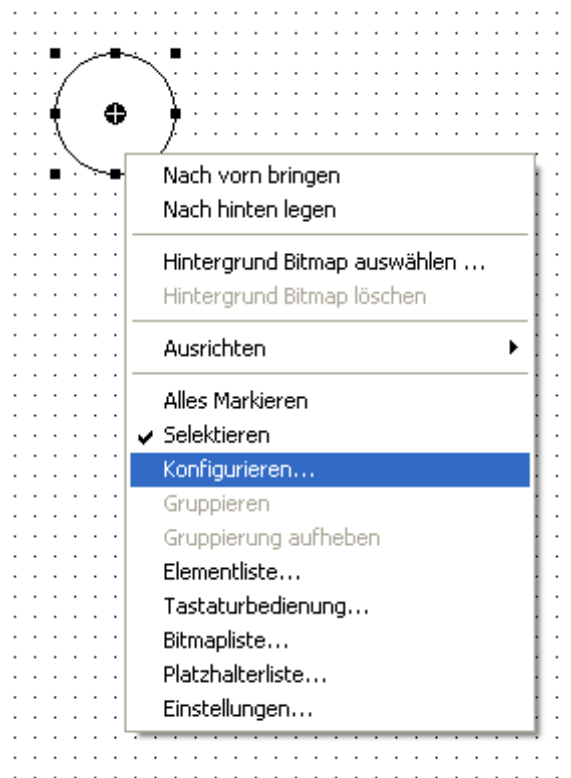


Abbildung 12.40: CoDeSys (Visualisierung Ellipse konfigurieren)

Auf der Linken Seite des Fenster wird nun der Befehl „Farbvariablen“ gewählt.

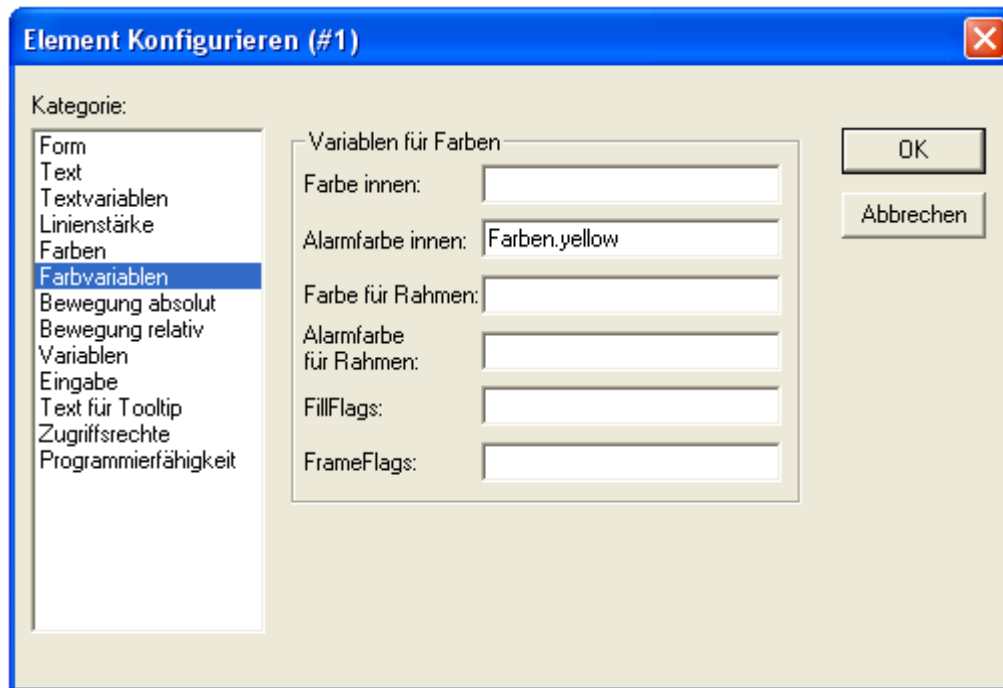


Abbildung 12.41: CoDeSys (Visualisierung Ellipse Farbvariablen)

Hier können nun Farben für die bestimmten Bereiche des Elementes ausgewählt werden.

In den beiden Felder „Alarmfarbe innen“ und „Alarmfarbe für Rahmen“ werden die Farben eingetragen, die bei einer Zustandsänderung der entsprechenden Variable im Element erscheinen. Die Farben können nicht direkt in die Felder eingegeben werden, sondern müssen in einem bestimmten Programm als Konstante aufgeführt werden. Die Farben werden als RGB Farben in einem DWORD als Hexadezimalwert gespeichert.

```
0001 yellow:=16#00FFFF;  
0002 red:=16#0000FF;  
0003 blue:=16#FF0000;
```

Abbildung 12.42: CoDeSys
(Visualisierung
Farbvariablen)

Unter dem Befehl „Variablen“ im Linken Teil des Fensters, kann die Variable ausgewählt werden, deren Zustandsänderung einen Farbwechsel hervorrufen soll.

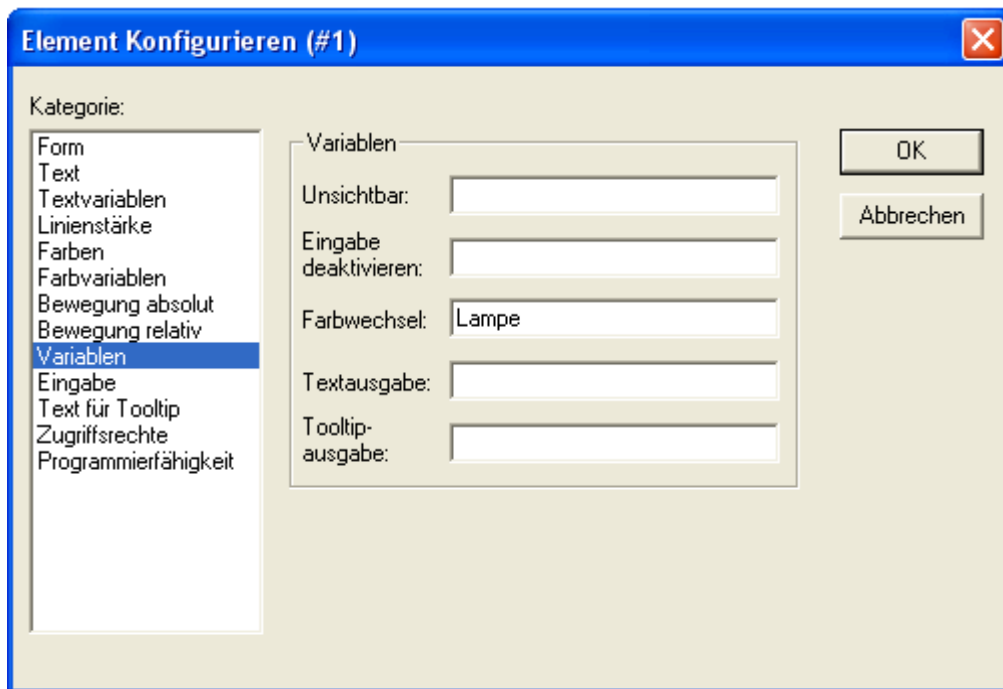


Abbildung 12.43: CoDeSys (Visualisierung Variablen)

Hier wird nun wieder die Variable „Lampe“ ausgewählt, welche ja auch über die Schaltfläche „getoggelt“ wird. Wird in dem Feld „Unsichtbar“ eine Variable eingetragen, ist das Element nur sichtbar, wenn die Variable den Wert „FALSE“ hat. Mit dem Feld „Eingabe deaktivieren“ werden alle Einstellungen in der Kategorie „Eingabe“ nicht berücksichtigt, wenn die Variable, die in dem Feld eingetragen ist, den Wert „TRUE“ hat. Auf das Feld „Textausgabe“ wird später nochmal genauer eingegangen.

Darstellung im Online Modus:



Abbildung 12.44: CoDeSys (Visualisierung Lampe aus)



Abbildung 12.45: CoDeSys (Visualisierung Lampe ein)

Scrollleiste :

Eine Scrollleiste ist ein grafisches Visualisierungselement, welches den Wert einer Variable ändern kann.

1. Scrollleiste einfügen und Konfigurieren



Abbildung 12.46: CoDeSys (Visualisierung Scrollleiste einfügen)

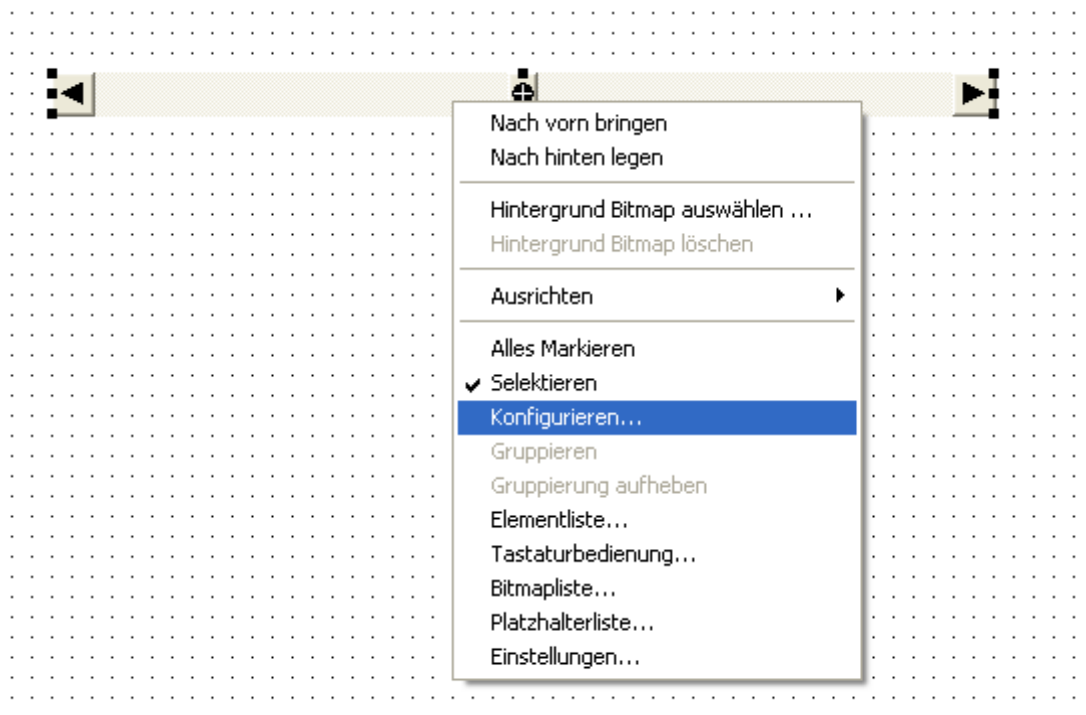


Abbildung 12.47: CoDeSys (Visualisierung Scrollleiste konfigurieren)

Im Konfigurationsfenster wird nun bestimmt, von welcher Variable der Wert geändert wird.

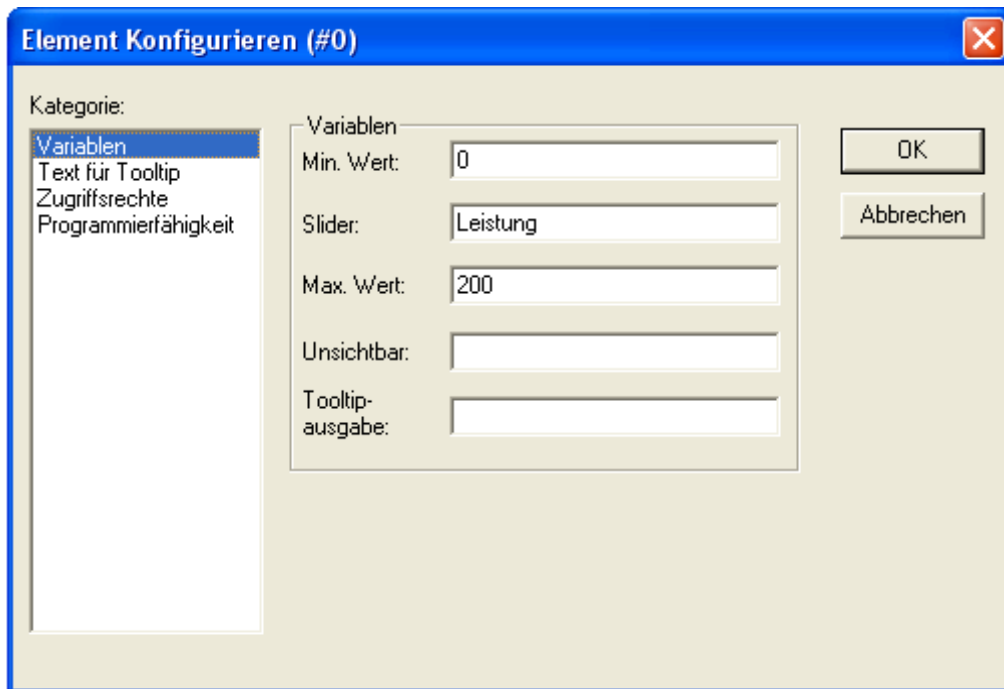


Abbildung 12.48: CoDeSys (Visualisierung Scrollleiste Variablen)

Im Feld „Min. Wert“ und „Max. Wert“ kann der Wertebereich der Variable eingestellt werden (hier 0 - 200). Im Feld „Slider“ wird die Variable eingetragen, welche durch den Schiebebalken der Scrollleiste im Online Modus verändert wird.

Zeigerinstrument:

Über ein Zeigerinstrument kann der Wert einer Variable im Online Modus angezeigt werden.



Abbildung 12.49: CoDeSys (Visualisierung Zeigerinstrument einfügen)

Sobald das Zeigerinstrument gezeichnet wurde, öffnet sich ein Fenster zum parametrieren des Instrumentes. Hier können diverse Einstellungen wie Pfeilart, Start- und Endwinkel, Zeigerfarbe, Position der Beschriftung oder Farbbereiche auf der Skala vernommen werden.

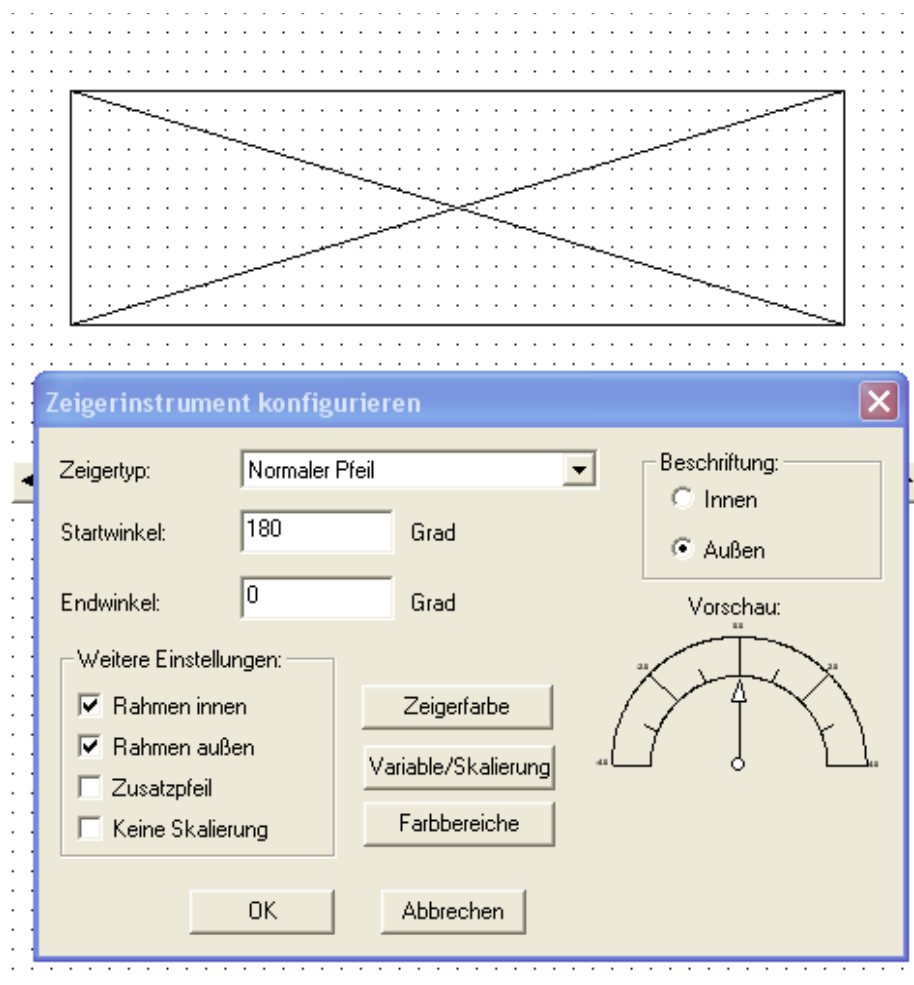
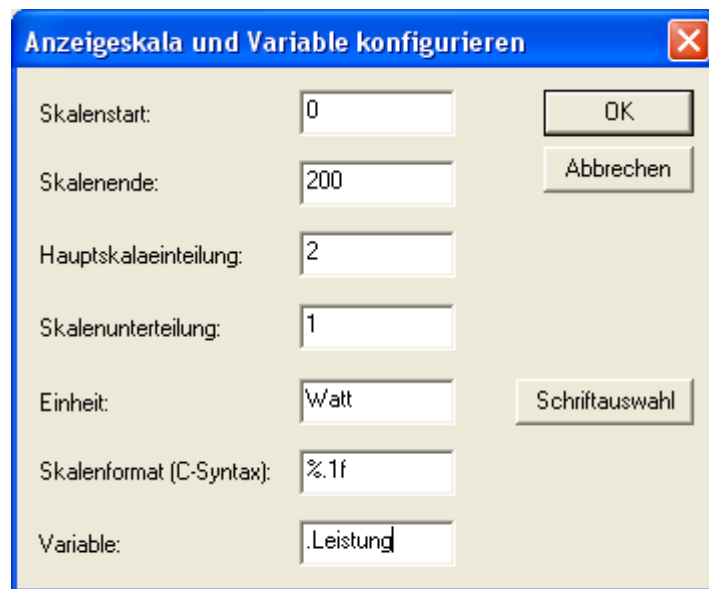


Abbildung 12.50: CoDeSys (Visualisierung Zeigerinstrument konfigurieren)

Über den Button „Variable/Skalierung“ gelangt man zum Konfigurationsfenster der Anzeigeskala und der Variable.



The screenshot shows a Windows-style dialog box titled "Anzeigeskala und Variable konfigurieren". It has a blue title bar with a red 'X' close button. The dialog contains the following fields and controls:

- Skalenstart:** Input field with value "0".
- Skalenende:** Input field with value "200".
- Hauptskala-einteilung:** Input field with value "2".
- Skalenunterteilung:** Input field with value "1".
- Einheit:** Input field with value "Watt".
- Skalenformat (C-Syntax):** Input field with value "%.1f".
- Variable:** Input field with value ".Leistung".
- Buttons:** "OK", "Abbrechen", and "Schriftauswahl".

Abbildung 12.51: CoDeSys (Visualisierung Zeigerinstrument
Anzeigeskala Variable konfigurieren)

Hier kann nun der zunächst Minimale und der Maximale Wert der Skala eingestellt werden (zur Erinnerung: Die Scrollleiste verändert die Variable „Leistung“ in einem Wertebereich von 0 bis 200 Watt). Die Hauptskaleneinteilung gibt an, in welchen Schritten die Skalenwerte angezeigt werden (hier jeder zweite ganzzahlige Wert). Eine zusätzliche „Skalenunterteilung“ zeigt noch einmal kurze unbeschriftete Striche an. Im Feld „Einheit“ kann die Einheit der angezeigten Variable (hier Watt) eingetragen werden. Diese wird am Zeigerursprung angezeigt. Im Feld „Skalenformat“ wird eingestellt, in welchem Format die Zahlen ausgegeben werden (hier %.1f: Die Zahlen werden ganzzahlig und ohne Nachkommastelle ausgegeben). Die Variable, die das Zeigerinstrument anzeigen soll, wird im Feld „Variable“ eingetragen.

Darstellung im Online Modus:

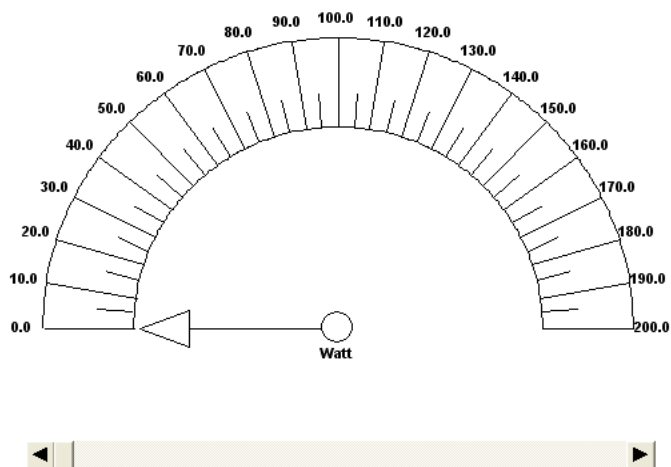


Abbildung 12.52: CoDeSys (Visualisierung Zeigerinstrument Wert=0)

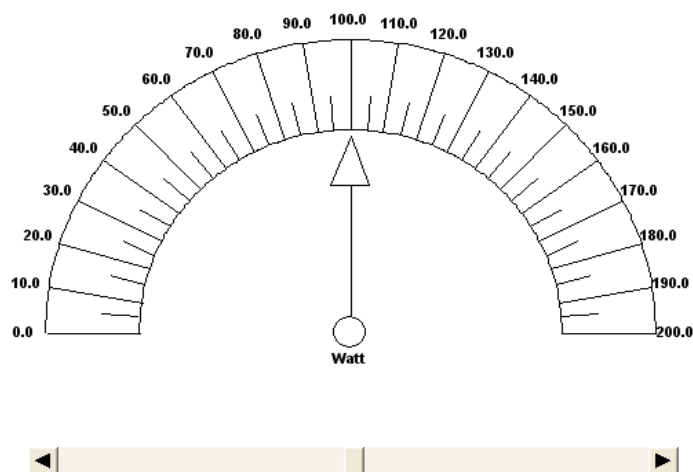


Abbildung 12.53: CoDeSys (Visualisierung Zeigerinstrument Wert=100)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

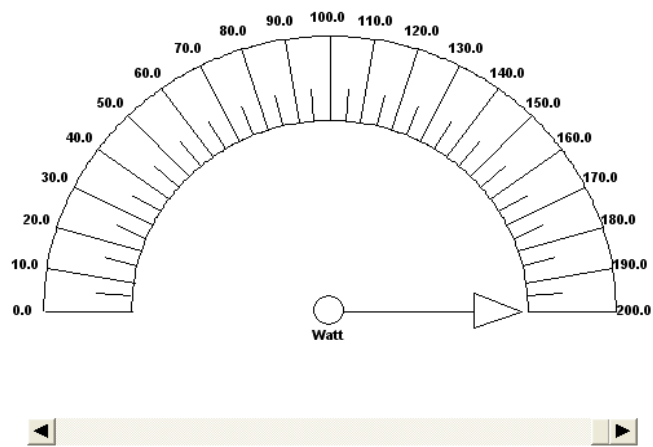


Abbildung 12.54: CoDeSys (Visualisierung Zeigerinstrument
Wert=200)

Ein- und Ausgabe über ein „Eingabefeld“:

Variablen können auch über ein Eingabefeld ein- oder ausgegeben werden. Hierzu müssen bestimmte Einstellungen im Konfigurationsfenster gemacht werden.

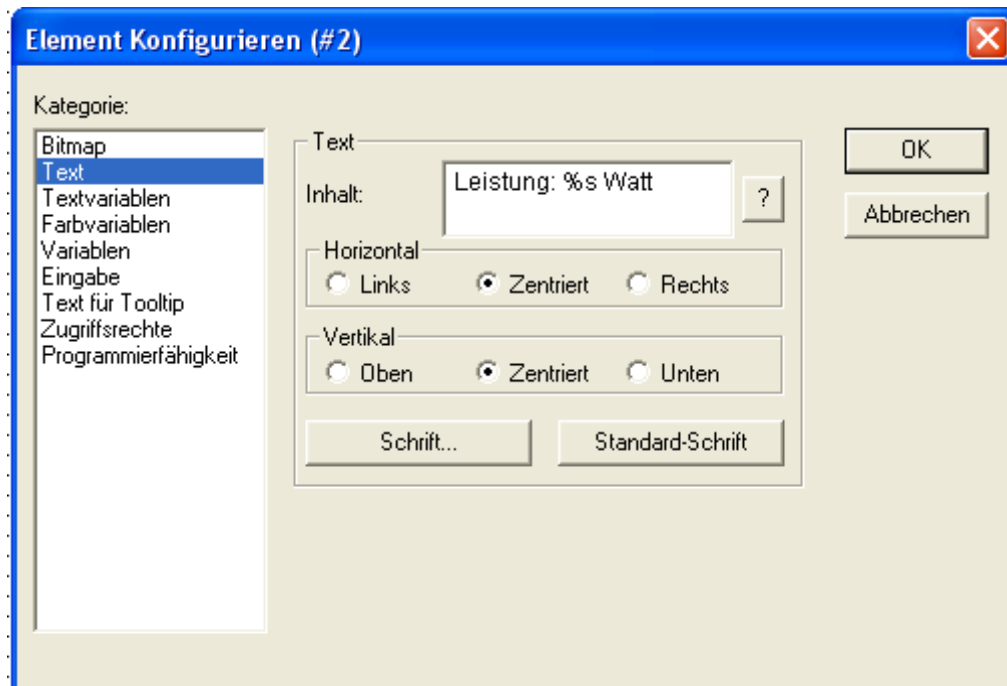


Abbildung 12.55: CoDeSys (Visualisierung Eingabefeld Text)

Zunächst wird der Text eingegeben, der später auf dem Eingabefeld erscheinen soll. „%s“ ist ein Platzhalter. An dieser Stelle erscheint später der Wert der Variable.

Unter dem Befehl „Variablen“ wird im Feld „Textausgabe“ die Variable eingetragen, deren Wert in

der Textausgabe bei dem Platzhalter „%s“ erscheint.

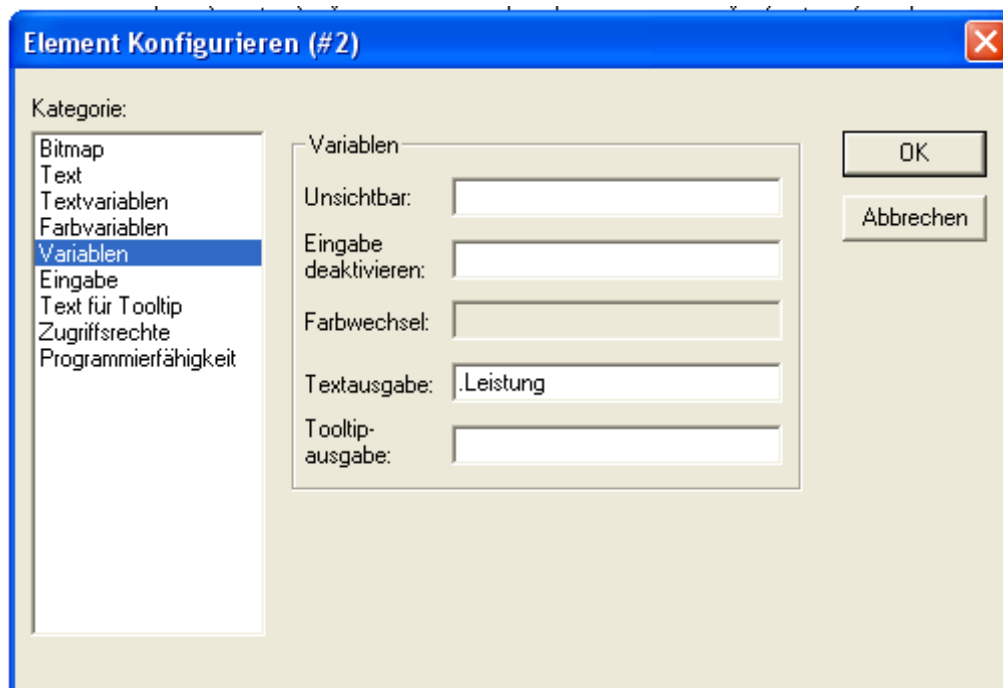


Abbildung 12.56: CoDeSys (Visualisierung Eingabefeld Variablen)

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Im Online Modus wird nun der Wert der Variable auf dem Eingabefeld angezeigt.

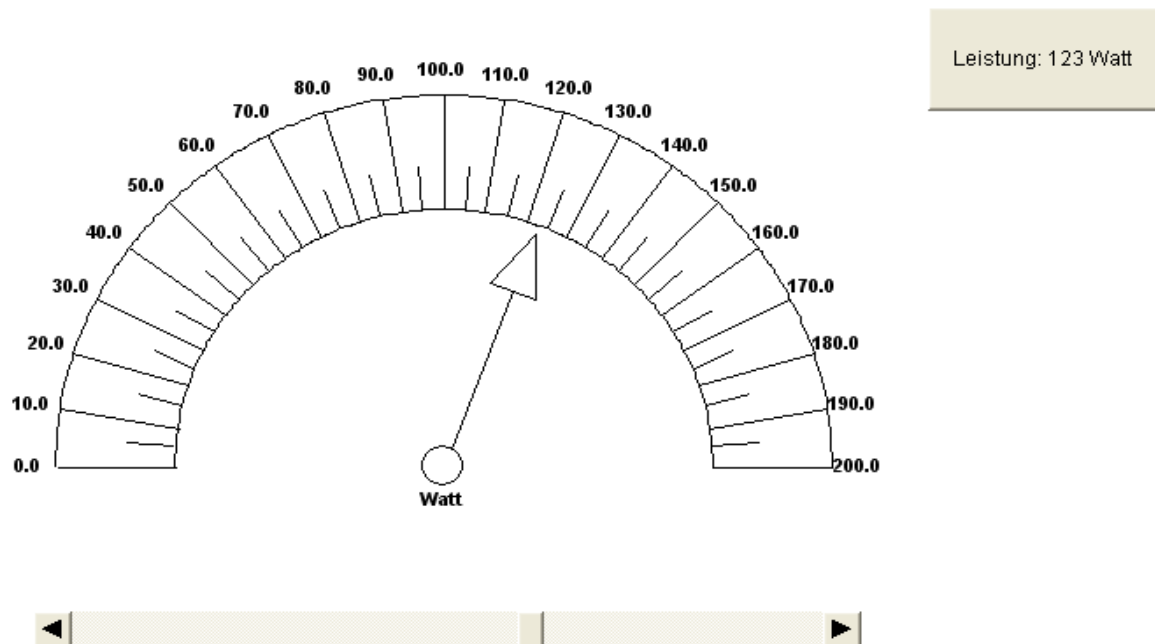


Abbildung 12.57: CoDeSys (Visualisierung Eingabefeld Wert=123)

Soll nun der Wert der Variable auch über dem Eingabefeld verändert werden muss zusätzlich noch im Befehl „Eingabe“ ein Haken am Feld „Text Eingabe der Variable 'Texteingabe'“ gemacht werden. Nun kann auch hier nochmal der Wertebereich eingestellt werden.

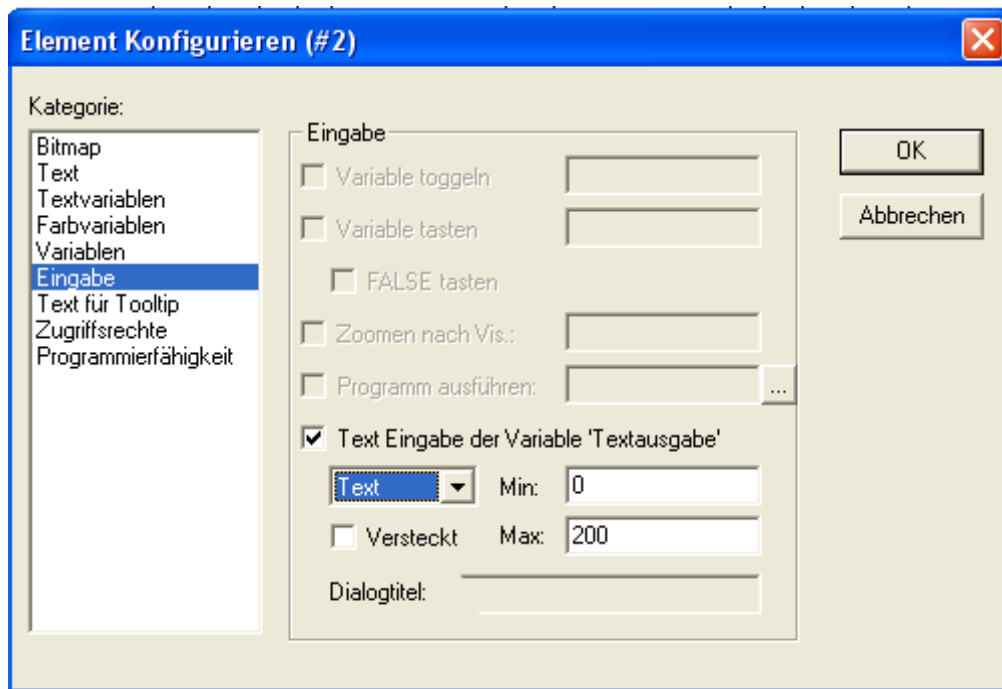


Abbildung 12.58: CoDeSys (Visualisierung Eingabefeld Eingabe)

Zusätzlich kann in dem Pull-Down Menu die Art der Eingabe gewählt werden. Hier kann zum Beispiel ein Keypad und Numpad ausgewählt werden. Geschieht dies, so wird im Online Modus die Nachbildung eines alphabetischen bzw. numerischen Tastaturfeldes geöffnet. Wird ein Haken an dem Feld „Versteckt“ gesetzt, erscheint anstatt dem Wert der Variable im Online Modus nur „*****“ an der entsprechenden Stelle.

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Darstellung im Online Modus:

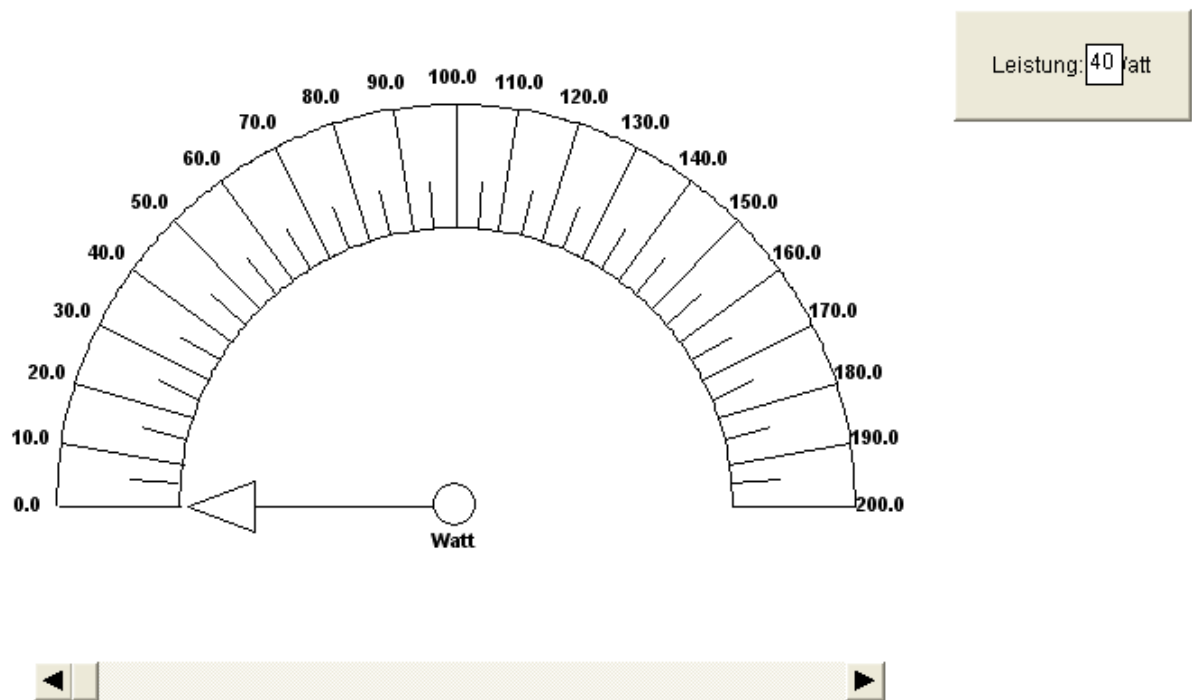


Abbildung 12.59: CoDeSys (Visualisierung Eingabefeld Werteingabe)

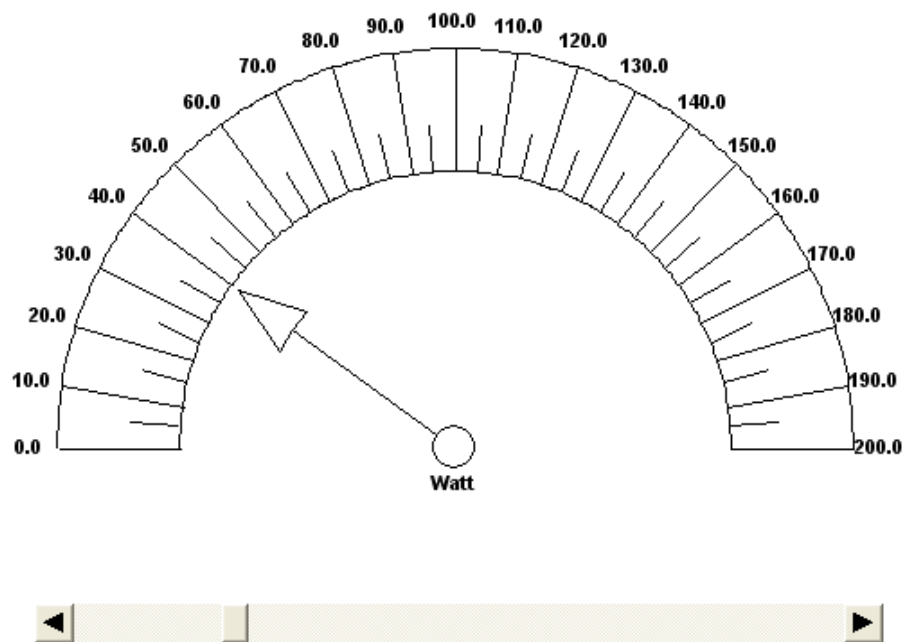


Abbildung 12.60: CoDeSys (Visualisierung Eingabefeld Werteingabe)

7 Das Puppenhaus

Zur Demonstrations- und Testzwecken wurde ein Puppenhaus verwendet. In diesem Puppenhaus wurden eine Vielzahl von Tastern, Lampen, LEDs und Thermoelementen installiert.



Abbildung 13.1: Puppenhaus

7.1 Bauteile im Haus

Taster:

Anzahl: 16



*Abbildung
13.2:
Puppenhaus
Taster*

Die Taster sind auf die digitalen Eingangsmodule der SPS geschaltet. Damit die Eingangsmodule beim Betätigen des Taster eine logische „EINS“ erkennen, muss jeder Taster mit der 24 V Spannungsversorgung verbunden werden.

Glühlampen 24V:

Anzahl: 11

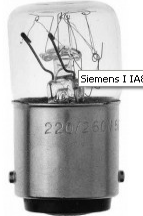


*Abbildung 13.3:
Puppenhaus 24 V Lampe*

Die 24 V Glühlampen sind auf die digitalen Ausgangsmodule der SPS geschaltet. Sobald die jeweiligen Ausgänge auf der SPS-Klemme einen High-Pegel haben, befinden diese sich auf einem Spannungspotential von 24 V. Die Lampen müssen alle mit der Masse verbunden werden.

Glühlampen 230 V:

Anzahl:5



*Abbildung
13.3:
Puppenhaus
230 V Lampe*

Damit über die 3-Phasen-Leistungsmessklemmen eine richtige Leistung gemessen werden kann, wurden noch zusätzlich 230 V Glühlampen in das Haus eingebaut. Diese Glühlampen werden über die Relais-Ausgangsklemmen (750-517) geschaltet und mit Strom versorgt.

Temperaturfühler: 6

Zur Messung der Temperatur wurden in jedem Zimmer sowie außen am Haus Temperatursensoren eingebaut. Diese Temperatursensoren sind mit den Thermokopplern (750-469) auf der SPS verbunden.

LEDs:

Anzahl Heizungen: 5

Anzahl LED „Lichtorgel“: 10

Anzahl Herd: 1

Anzahl Waschmaschine: 1

Anzahl Dunstabzugshaube: 1

Anzahl LED Lampen Badezimmer: 3

Bestimmte Geräte im Haus wie zum Beispiel Heizungen, Waschmaschine, Herd, Staubsauger und die Dunstabzugshaube besitzen LEDs die bei eingeschalteten Zustand leuchten. Außerdem befinden sich im Badezimmer LED Lampen und im Schlafzimmer wurde eine Art „Lichtorgel“ mit verschiedenen LED Farben eingebaut. Diese LEDs wurden auf die analogen Ausgangsklemmen (750-552) geschaltet.

Kontakttaster:

Anzahl 4:

An den Fenstern sowie an der Eingangstür vom Haus befinden sich Kontaktschalter, welche signalisieren, ob ein Fenster oder eine Tür geöffnet ist. Die Taster sind auf die digital Eingangsklemmen (750-430) geschaltet und werden zusätzlich für den High-Pegel mit der 24 V Spannungsversorgung verbunden.

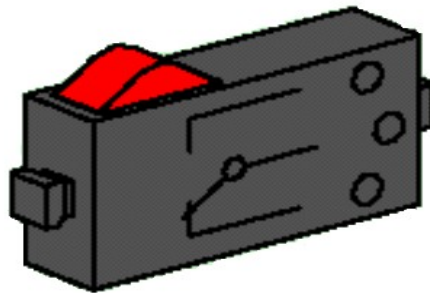


Abbildung 13.4: Puppenhaus Kontakttaster

Jalousie-Motoren:

Anzahl: 2

Die Jalousie-Motoren können sich in zwei Richtungen drehen. Sie sind auf die Relais-Ausgangsklemmen (750-469) über eine speziellen Brückenschalter geschaltet.

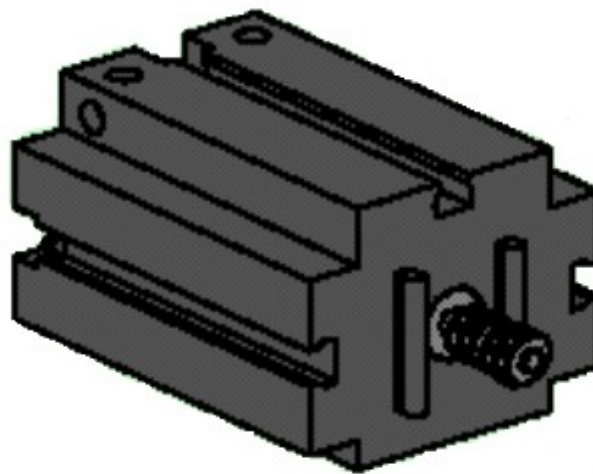


Abbildung 13.5: Puppenhaus Jalousiemotor

7.2 Verdrahtung am Haus:

Ein Großteil Der Lampen und Taster im Puppenhaus sind mit den drei Klemmenblöcken auf der Rückseite des Hauses verdrahtet.

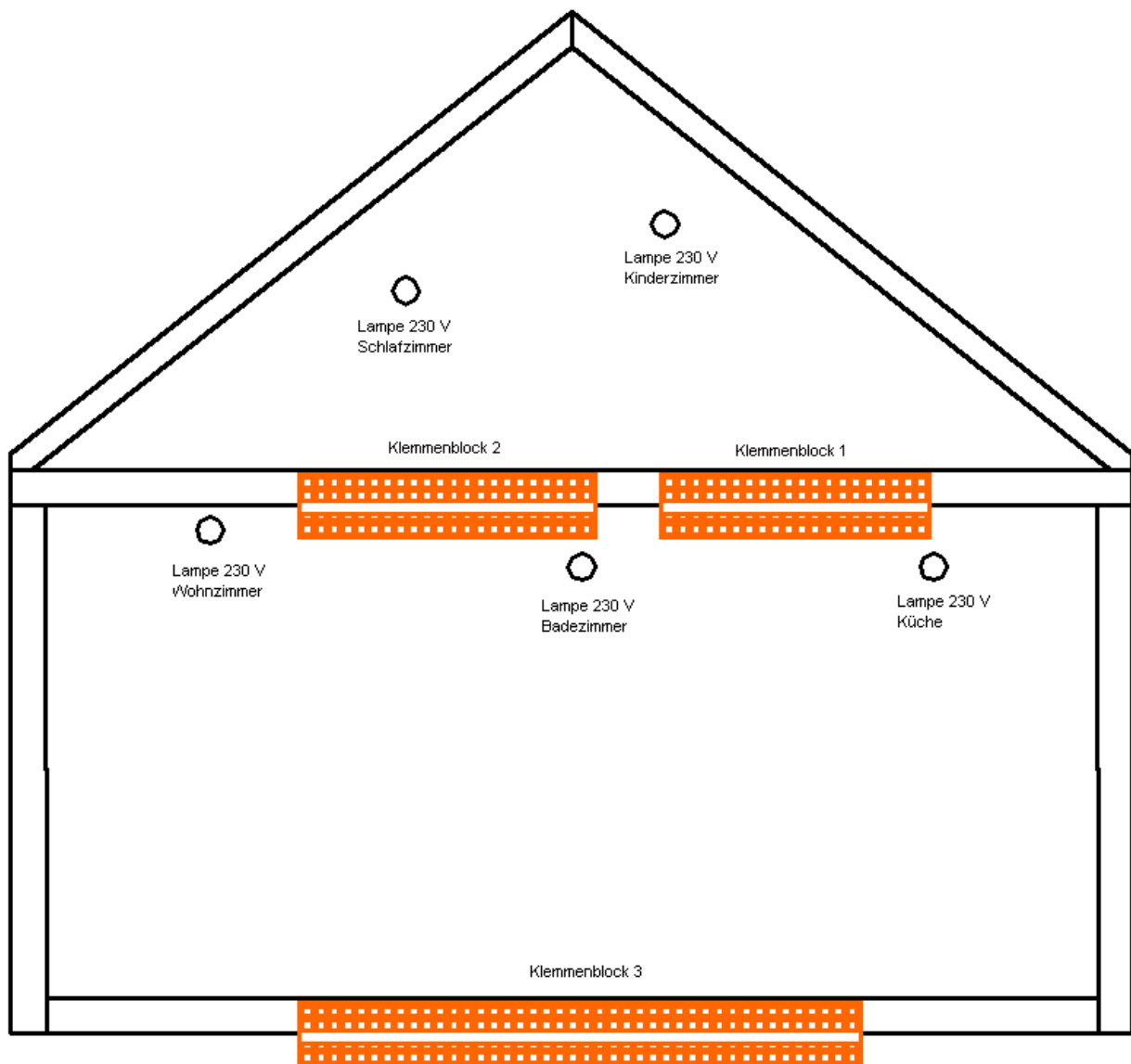


Abbildung 13.6: Puppenhaus Rückseite Klemmenblöcke

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Die elektrischen Bauteile sind auf die oberen Anschlüsse der Klemmen geschaltet. Die unteren Anschlüsse der Klemmen sind je nach Bauteil entweder auf Masse oder 24 V geschaltet. Für die Verbindung der Bauteile zur SPS wurden spezielle farbige Drähte verwendet. Diese Drähte wurden auf einen Centronics Stecker gelötet. Das Gegenstück des Steckers (Buchse) ist mit der SPS verbunden. Auf die Buchse wurden die selben Drähte mit der gleichen Farbkombination wie die vom Stecker gelötet.

Belegung Klemmenblock 1:

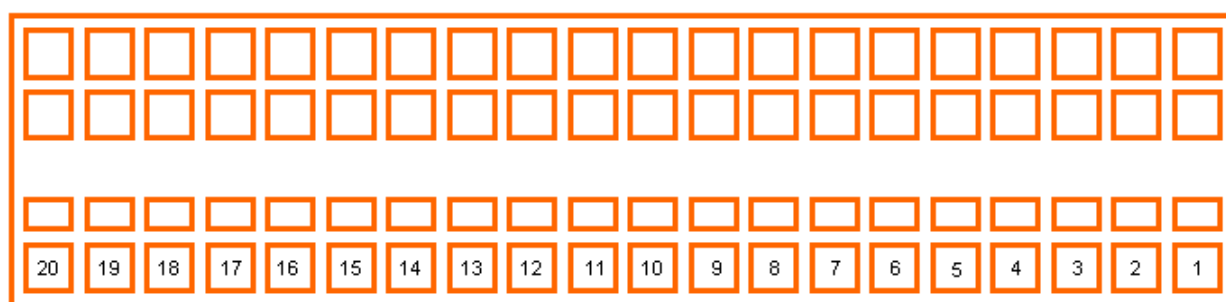


Abbildung 13.7: Puppenhaus Klemmenblock 1

Klemmenblock 1	Klemme	Farbe (Kabel)
Kinderzimmer Steckdose	1	Braun/Grün
Kinderzimmer Steckdose (Masse)	2	Rot
Kinderzimmer Links Taster (24 V)	3	Blau
Kinderzimmer Links Taster	4	Braun/Rot
Kinderzimmer Bett oben Licht	5	Gelb/Schwarz
Kinderzimmer Bett oben Licht (Masse)	6	Rot
Kinderzimmer Bett oben Taster (24 V)	7	Blau
Kinderzimmer Bett oben Taster	8	Gelb/Blau
Kinderzimmer Bett unten Licht	9	Grün/Braun
Kinderzimmer Bett unten Licht (Masse)	10	Rot
Kinderzimmer Bett unten Taster	11	Weiß/Rosa
Kinderzimmer Bett unten Taster (24 V)	12	Blau
Kinderzimmer Licht	13	Braun/Schwarz
Kinderzimmer Licht (Masse)	14	Rot
Kinderzimmer rechts Taster	15	Gelb/Braun
Kinderzimmer rechts Taster (24 V)	16	Blau
Kinderzimmer Heizung	17	Rosa/Braun
Kinderzimmer Heizung (Masse)	18	Rot
Jalousie Links auf	19	Braun/Blau
Jalousie Links ab	20	Braun/Blau

Tabelle 1: Belegung Klemmenblock 1

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Belegung Klemmenblock 2:

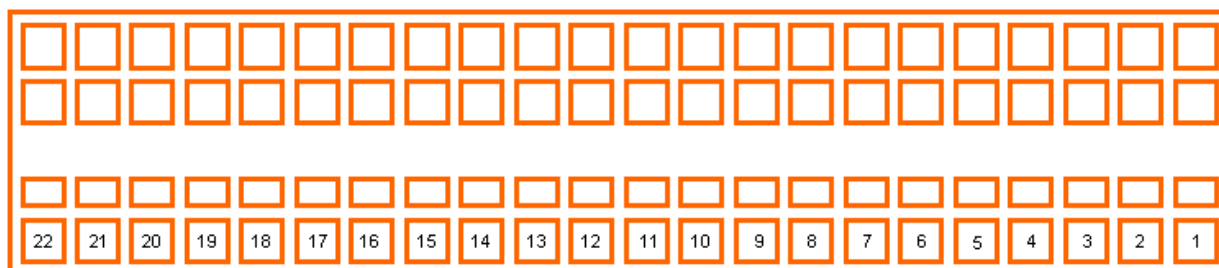


Abbildung 13.7: Puppenhaus Klemmenblock 2

Klemmenblock 2	Klemme	Farbe (Kabel)
Schlafzimmer vorne Licht	1	Gelb/Rosa
Schlafzimmer vorne Licht (Masse)	2	Rot
Schlafzimmer hinten Licht	3	Grün/Blau
Schlafzimmer hinten Licht (Masse)	4	Rot
Schlafzimmer links Taster (24 V)	5	Blau
Schlafzimmer links Taster	6	Weiß/Grün
Schlafzimmer Bett Licht	7	Weiß/Rot
Schlafzimmer Bett Licht (Masse)	8	Rot und Braun
Schlafzimmer Bett rechts Taster (24 V)	9	Blau
Schlafzimmer Bett rechts Taster	10	Grau/Braun
Schlafzimmer Bett links Taster (24 V)	11	Blau und Rot
Schlafzimmer Bett links Taster	12	Rot/Blau
LEDs Masse	13	Blau
LED Blau	14	Weiß/Blau
LED Gelb	15	Grün/Schwarz
LED Rot	16	Weiß/Grau
LED Grün	17	Weiß/Schwarz
LED Weiß	18	Gelb/Grau
Schlafzimmer Heizung	19	Grau/Grün
Schlafzimmer Heizung (Masse)	20	Blau und Braun
Jalousie rechts auf	21	Weiß/Gelb
Jalousie rechts ab	22	Weiß/Gelb

Tabelle 2: Belegung Klemmenblock 2

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Belegung Klemmenblock 3



Abbildung 13.8: Puppenhaus Klemmenblock 3

Klemmenblock 3	Klemme	Farbe (Kabel)
Küche Herd	1	Braun/Grün
Küche Herd (Masse)	2	Rot
Küche Dunstabzugshaube	3	Gelb/Schwarz
Küche Dunstabzugshaube (Masse)	4	Rot
Küche Licht	5	Grün/Braun
Küche Licht (Masse)	6	Rot
Küche links Taster	7	Weiß/Rosa
Küche links Taster (24 V)	8	Blau
Küche Waschmaschine	9	Rosa/Braun
Küche Waschmaschine (Masse)	10	Rot
Küche rechts Taster	11	Gelb/Braun
Küche rechts Taster (24 V)	12	Blau
Bad links Taster	13	Weiß/Grün
Bad links Taster (24 V)	14	Blau
Bad Links vorne Licht	15	Gelb/Rosa
Bad Links vorne Licht (Masse)	16	Rot
Bad links hinten Licht	17	Grün/Blau
Bad links hinten Licht (Masse)	18	Rot
Küche Heizung	19	Weiß/Rot
Küche Heizung (Masse)	20	Rot
Bad Heizung	21	Grau/Grün
Bad Heizung (Masse)	22	Rot
Bad rechts Taster	23	Grau/Braun
Bad rechts Taster (24 V)	24	Blau
Bad rechts hinten Licht	25	Weiß/Blau
Bad rechts hinten Licht Masse	26	Rot
Bad rechts vorne Licht	27	Grün/Schwarz
Bad rechts vorne Licht Masse	28	Rot
Bad rechts mitte Licht	29	Weiß/Grau
Bad rechts mitte Licht (Masse)	30	Rot
Wohnzimmer hinten Licht	31	Weiß/Schwarz
Wohnzimmer hinten Licht (Masse)	32	Rot
Wohnzimmer vorne Licht	33	Gelb/Grau
Wohnzimmer vorne Licht (Masse)	34	Rot
Wohnzimmer Taster	35	Rot/Blau
Wohnzimmer Taster (24 V)	36	Blau
Wohnzimmer Stehlampe	37	Rosa/Grün
Wohnzimmer Stehlampe (24 V)	38	Blau
Wohnzimmer Heizung	39	Braun/Rot
Wohnzimmer Heizung (Masse)	40	Rot

Tabelle 3: Belegung Klemmenblock 3

Centronics Stecker und Buchse (50 Polig):

Stecker:



Abbildung 13.9: Puppenhaus Centronics Stecker

Buchse:



Abbildung 13.10: Puppenhaus Centronics Buchse

Pinbelegung auf der Rückseite des Steckers:

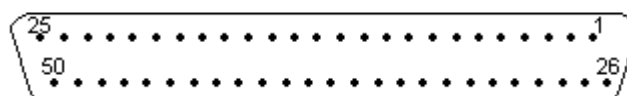


Abbildung 13.11: Puppenhaus Centronics Stecker Rückseite

Auf den Stecker wurden insgesamt 50 Drähte mit 25 verschiedenen Farben gelötet. Die Farbkombination wiederholt sich ab dem 26. Pin. An die Buchse auf der Gegenseite wurde exakt die gleiche Farbkombination gelötet. Die Kabel an der Buchse wurden direkt mit der SPS verbunden.

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Belegung Stecker:

Pin	Farbe	Komponente
1	Weiß/Gelb	Jalousie rechts auf
2	Braun/Blau	Jalousie Links auf
3	Gelb/Rot	nicht belegt
4	Rot/Blau	Schlafzimmer Bett links Taster
5	Grau/Braun	Schlafzimmer Bett rechts Taster
6	Weiß/Grün	Schlafzimmer links Taster
7	Gelb/Braun	Kinderzimmer rechts Taster
8	Weiß/Rosa	Kinderzimmer Bett unten Taster
9	Gelb/Blau	Kinderzimmer Bett oben Taster
10	Braun/Rot	Kinderzimmer Links Taster
11	Rosa/Grün	nicht belegt
12	Gelb/Grau	LED Weiß
13	Weiß/Schwarz	LED Grün
14	Weiß/Grau	LED Rot
15	Grün/Schwarz	LED Gelb
16	Weiß/Blau	LED Blau
17	Grau/Grün	Schlafzimmer Heizung
18	Weiß/Rot	Schlafzimmer Bett Licht
19	Grün/Blau	Schlafzimmer hinten Licht
20	Gelb/Rosa	Schlafzimmer vorne Licht
21	Rosa/Braun	Kinderzimmer Heizung
22	Braun/Schwarz	Kinderzimmer Licht
23	Grün/Braun	Kinderzimmer Bett unten Licht
24	Gelb/Schwarz	Kinderzimmer Bett oben Licht
25	Grün/Braun	Kinderzimmer Bett unten Licht
26	Weiß/Gelb	Jalousie rechts ab
27	Braun/Blau	Jalousie rechts auf
28	Gelb/Rot	nicht belegt
29	Rot/Blau	Wohnzimmer Taster
30	Grau/Braun	Bad rechts Taster
31	Weiß/Grün	Bad links Taster
32	Gelb/Braun	Küche rechts Taster
33	Weiß/Rosa	Küche links Taster
34	Gelb/Blau	nicht belegt
35	Braun/Rot	Wohnzimmer Heizung
36	Rosa/Grün	Wohnzimmer Stehlampe
37	Gelb/Grau	Wohnzimmer vorne Licht
38	Weiß/Schwarz	Wohnzimmer hinten Licht
39	Weiß/Grau	Bad rechts mitte Licht
40	Grün/Schwarz	Bad rechts vorne Licht
41	Weiß/Blau	Bad rechts hinten Licht
42	Grau/Grün	Bad rechts Taster
43	Weiß/Rot	Küche Heizung
44	Grün/Blau	Bad links hinten Licht
45	Gelb/Rosa	Bad Links vorne Licht
46	Rosa/Braun	Küche Waschmaschine
47	Braun/Schwarz	nicht belegt
48	Grün/Braun	Küche Licht
49	Gelb/Schwarz	Küche Dunstabzugshaube
50	Grün/Braun	Küche Licht

Tabelle 4: Belegung Stecker Rückseite

7.3 Funktionen Haus

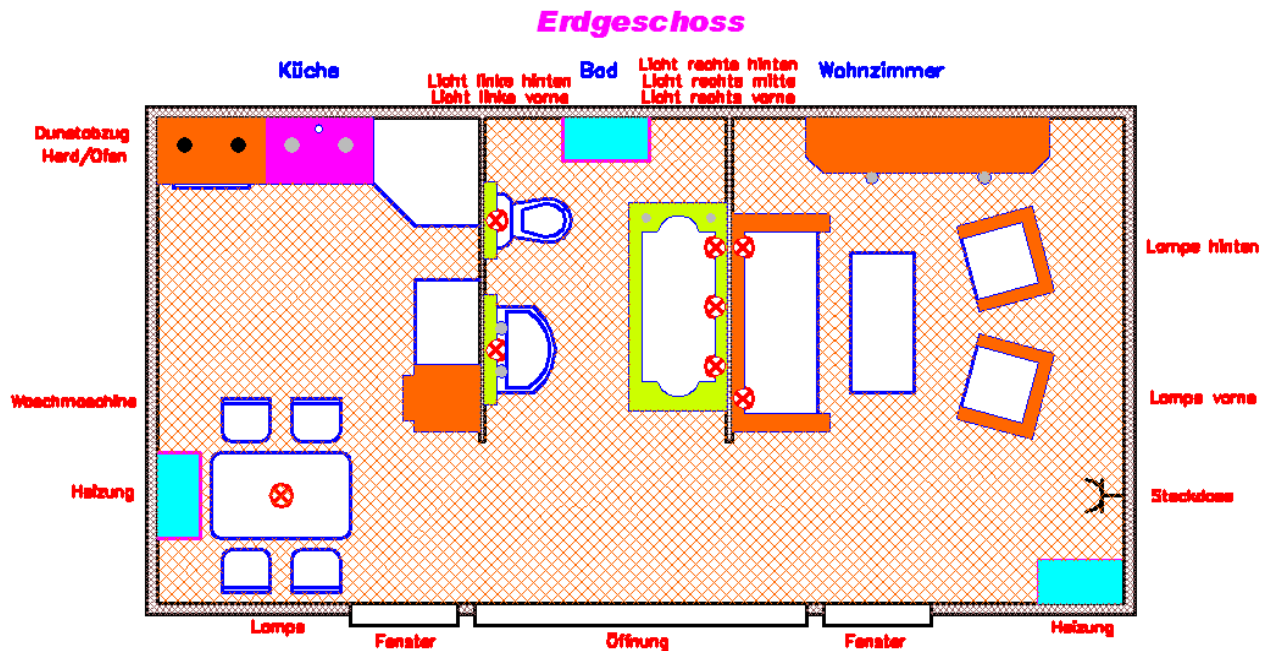


Abbildung 13.12: Puppenhaus Erdgeschoss

7.3.1 Haus Untergeschoss

Sensoren:

Taster Jalousie links auf
Taster Jalousie links ab
Taster Jalousie rechts auf
Taster Jalousie rechts ab
Fensterkontakt Fenster Links
Fensterkontakt Fenster rechts
Türkontakt links
Türkontakt rechst
Temperatursensor (aussen)

Aktoren:

Jalousie Links
Jalousie rechts

Lokale Funktionen:

Jalousie auffahren
Jalousie abfahren

Globale Funktionen:

Beschattungsposition
anfahren
Sicherheit
Alarm

7.3.2 Kinderzimmer

Das Kinderzimmer befindet sich im Obergeschoss auf der Linken Seite.

Geräte Kinderzimmer:

Sensoren:

Taster rechts

Taster Bett oben

Taster Bett unten

Taster Links

Temperatursensor

Aktoren:

Lampe

Lampe Bett oben

Lampe Bett unten

Lampe 230 V

Aktive Steckdose

Heizung

Funktionen Kinderzimmer:

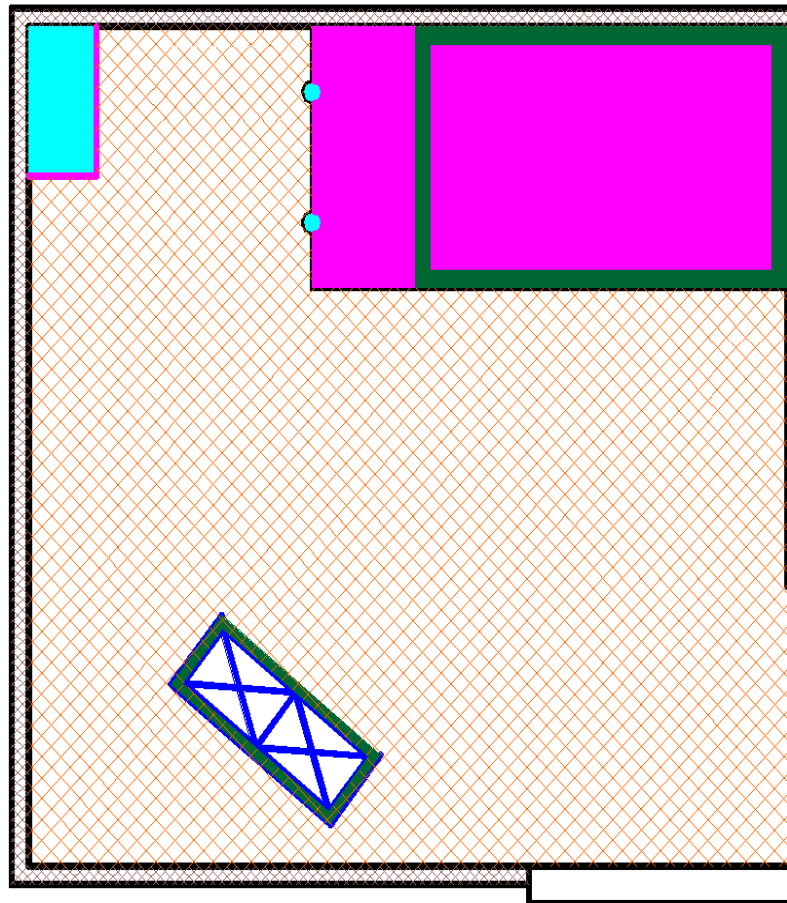


Abbildung 13.13: Puppenhaus Kinderzimmer

Lokale Funktionen:

Lampe umschalten

Lampe Bett oben umschalten
umschalten

Aktive Steckdose ein/aus

Lampe 230 V umschalten

Globale Funktionen:

Heizung ein/aus

Haus verlassen (Alle Aktoren Lampe Bett unten
ausschalten)

Kind 1 oder Kind 2 anwesend

(Lampe und Lampe 230 V einschalten)

Timer Geräte ein/aus

7.3.3 Schlafzimmer

Das Schlafzimmer befindet sich im Obergeschoss auf der rechten Seite.

Geräte Schlafzimmer

Sensoren:

Taster links

Taster Bett rechts

Taster Bett links

Temperatursensor

Aktoren:

Lampe rechts vorne

Lampe rechts hinten

Lampe Bett

LED-Lichter

Lampe 230 V

Heizung

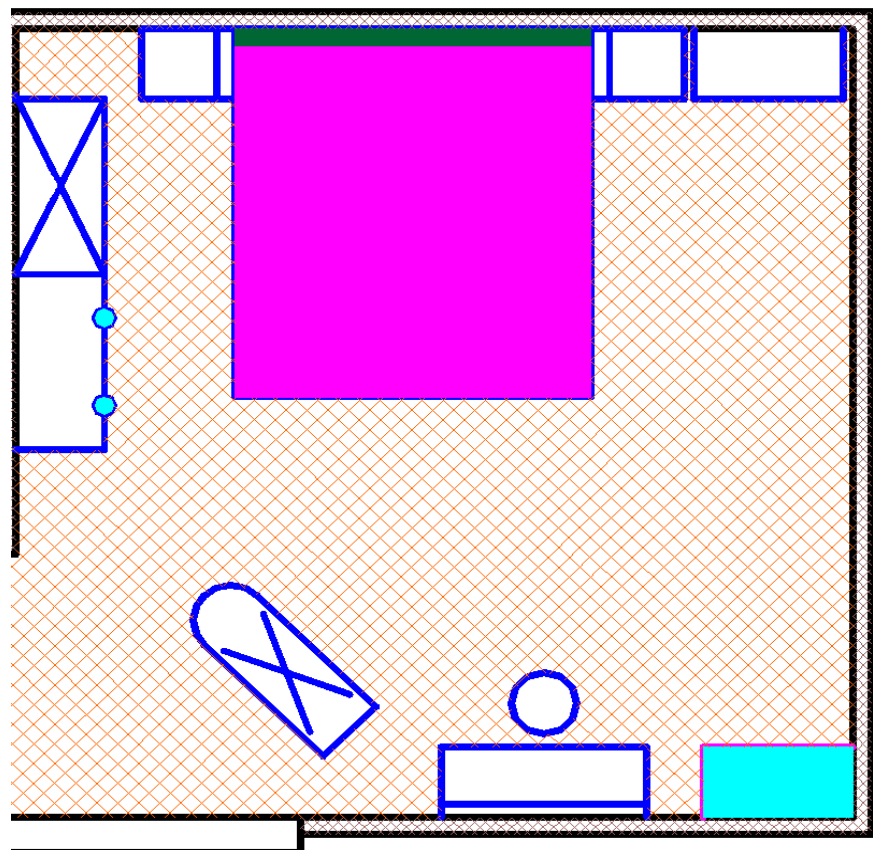


Abbildung 13.14: Puppenhaus Schlafzimmer

Funktionen Schlafzimmer:

Lokale Funktionen:

Lampe rechts vorne umschalten

Lampe rechts hinten umschalten

Lampe Bett umschalten

LED Lichter ein/aus

Lampe 230 V umschalten

Globale Funktionen:

Heizung ein/aus

Haus verlassen

Mutter und Vater anwesend

Timer Geräte ein/aus

7.3.4 Küche

Die Küche befindet sich im Erdgeschoss auf der linken Seite

Geräte Küche:

Sensoren:

Taster rechts

Taster links

Temperatursensor

Aktoren:

Lampe

Waschmaschine

Herd

Dunstabzugshaube

Heizung

Lampe 230 V

Funktionen Küche:

Lokale Funktionen:

Lampe umschalten

Waschmaschine ein/aus

Herd ein/aus

Dunstabzugshaube ein/aus

Lampe 230 V umschalten

Globale Funktionen:

Heizung ein/aus

Fenster oder Tür offen

Haus verlassen

Mutter anwesend

Timer Geräte ein/aus

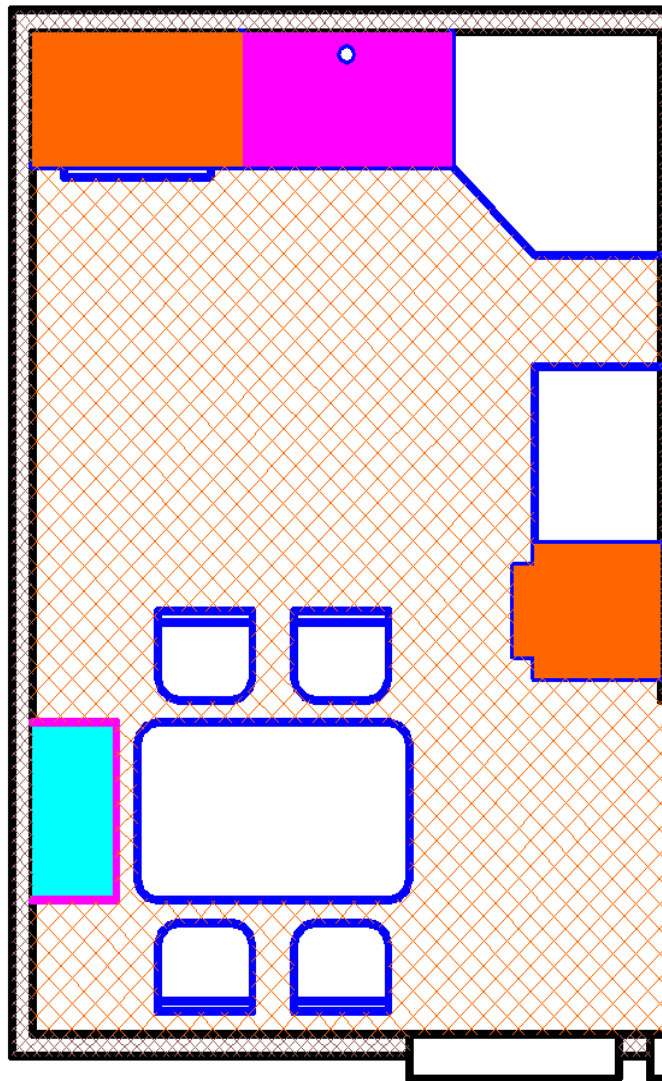


Abbildung 13.15: Puppenhaus Küche

7.3.5 Badezimmer

Das Badezimmer befindet sich im Erdgeschoss in der Mitte.

Geräte Badezimmer:

Sensoren:

Taster links

Taster rechts

Temperatursensor

Aktoren:

Lampe rechts vorne

Lampe rechts hinten

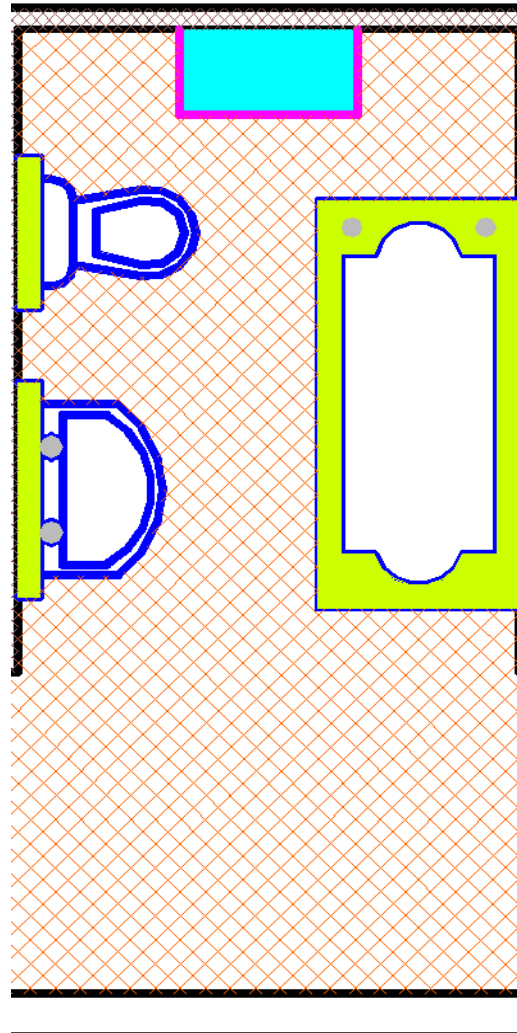
LED links vorne

LED links hinten

LED links mitte

Heizung

Lampe 230 V



Funktionen Badezimmer:

Lokale Funktionen:

Lampe rechts vorne umschalten

Lampe rechts hinten umschalten

LED links vorne umschalten

LED links hinten umschalten

LED mitte umschalten

Lampe 230 V umschalten

Globale Funktionen:

Heizung ein/aus

Fenster oder Tür offen

Haus verlassen

Timer Geräte ein/aus

Abbildung 13.16: Puppenhaus Badezimmer

7.3.6 Wohnzimmer

Das Wohnzimmer befindet sich im Erdgeschoss auf der linken Seite.

Geräte Wohnzimmer:

Sensoren:

Taster Links

Temperatursensor

Aktoren:

Lampe links vorne

Lampe links hinten

Heizung

Aktive Steckdose (Stehlampe)

Lampe 230 V

Funktionen:

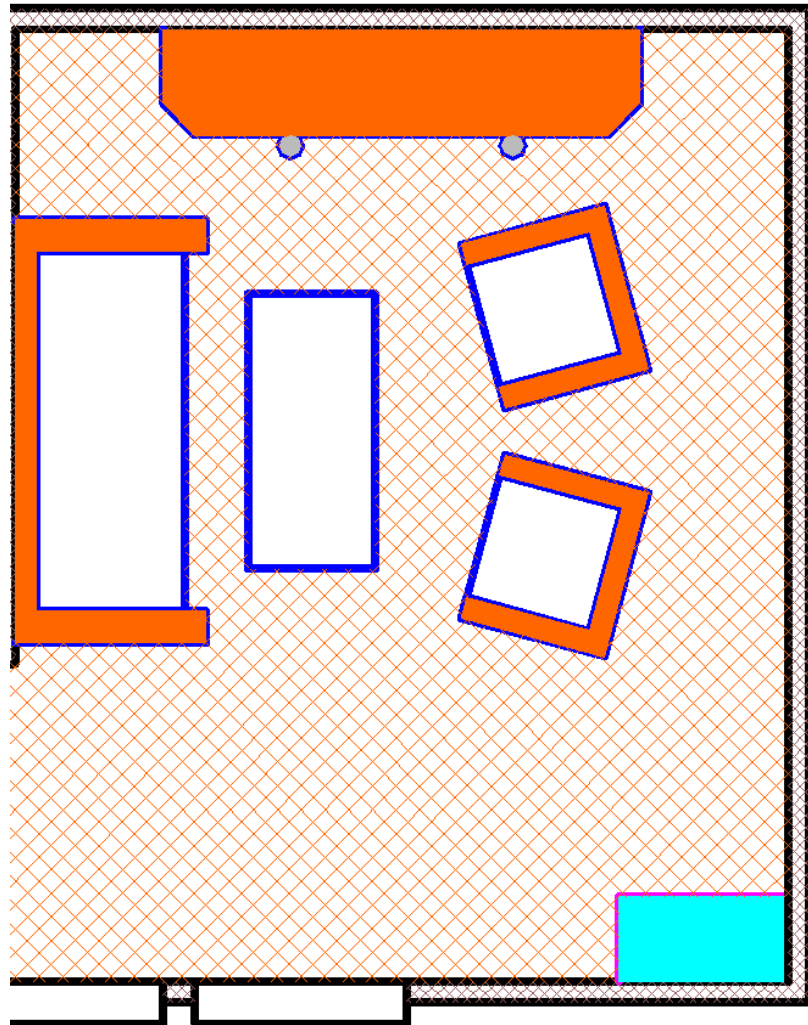


Abbildung 13.17: Puppenhaus Wohnzimmer

Lokale Funktionen:

Lampe links vorne umschalten

Lampe links hinten umschalten

Stehlampe umschalten

Lampe 230 V umschalten

Globale Funktionen:

Heizung ein/aus

Fenster oder Tür offen

Haus verlassen

Vater anwesend

Timer Geräte ein/aus

7.4 Funktionsbeschreibung

7.4.1 Lokale Funktionen

Umschalten:

Bei der Funktion „umschalten“ soll beim Drücken eines Tasters der entsprechende Aktor seinen Zustand ändern.

Temperatur:

Jedes Zimmer im Puppenhaus besitzt einen Temperatursensor. Die Temperaturwerte dieser Sensoren müssen später durch bestimmte Rechnungen linearisiert werden.

7.4.2 Globale Funktionen

Heizung:

Mit den gemessenen Temperaturwerten und der Solltemperatur wird über eine Differenz die Heizung ein oder ausgeschaltet.

Anwesenheit:

Das Puppenhaus hat eine Anwesenheitskontrolle. Hierbei wird festgestellt, welcher Bewohner sich gegenwärtig im Haus befindet. Sobald sich ein Bewohner anmeldet, werden in bestimmten Zimmer die Lichter eingeschaltet.

Haus verlassen:

Sobald alle Bewohner im Haus abgemeldet sind, werden sämtliche Geräte automatisch abgeschaltet. Lediglich die Heizkörper werden wieder eingeschaltet, sobald die Temperatur unter einem bestimmten Wert fällt.

Alarm:

Ist das Haus verlassen und ein Fenster oder eine Tür wird geöffnet, geht der Alarm los.

Fenster oder Tür offen:

Um Energie zu sparen, werden die Heizkörper beim Öffnen eines Fensters oder einer Tür automatisch abgeschaltet. Sie werden dann nur wieder eingeschaltet, wenn die Temperatur unter einen bestimmten Wert fällt.

Helligkeit:

In der Visualisierung ist ein Helligkeitssensor programmiert worden. Wenn die Helligkeit über einen bestimmten Wert steigt, dann werden alle Lampen im Haus ausgeschaltet und die Jalousien werden auf eine Beschattungsposition gefahren. Sinkt die Helligkeit unter einen bestimmten Wert, dann werden einige der Lampen im Haus eingeschaltet.

Timer Funktionen:

Der Benutzer hat über die Visualisierung die Möglichkeit, alle Geräte im Haus zu einer bestimmten Tageszeit an bestimmten Wochentagen automatisch ein- oder auszuschalten zu lassen.

7.5 Der Funktionsblock „umschalten“

Im Kapitel „Bibliotheksverwaltung“ wurde gezeigt, dass eine einfache Funktion wie „Taster-umschalten“ komfortable mit dem Funktionsblock „Stromstoss“ aus der Wago Bibliothek „Gebäude allgemein“ gelöst werden kann. Zudem hat der Funktionsblock zwei zusätzliche Eingänge „xZenEIN“ und „xZenAUS“ womit der Aktor über eine zentrale Funktion ein-oder ausgeschaltet werden kann. Nachteil der beiden zusätzlichen Eingänge ist allerdings, das der Handbetrieb nachrangig ist. Wird eine z.B. Lampe nun über eine zentrale Funktion eingeschaltet, kann dieser nicht mehr manuell über den Taster ausgeschaltet werden solange am Eingang „xZenEIN“ der boolische Wert „TRUE“ anliegt. Für solche Fälle wurde nun ein spezieller Funktionsblock programmiert .

Variablen:

```
0001 FUNCTION_BLOCK umschalten
0002 VAR_INPUT
0003     Taster:BOOL;
0004     EIN:BOOL;
0005     AUS:BOOL;
0006 END_VAR
0007 VAR_OUTPUT
0008     Aktor:BOOL;
0009 END_VAR
0010 VAR
0011     Taster_TRIG: R_TRIG;
0012     Aktor_umschalten: RS;
0013     Taster_EIN: RS;
0014     Taster_AUS: RS;
0015     Taster_umschalten: RS;
0016 END_VAR
```

Abbildung 14.1: Funktionsblock
umschalten Variablen

Funktionsblock:

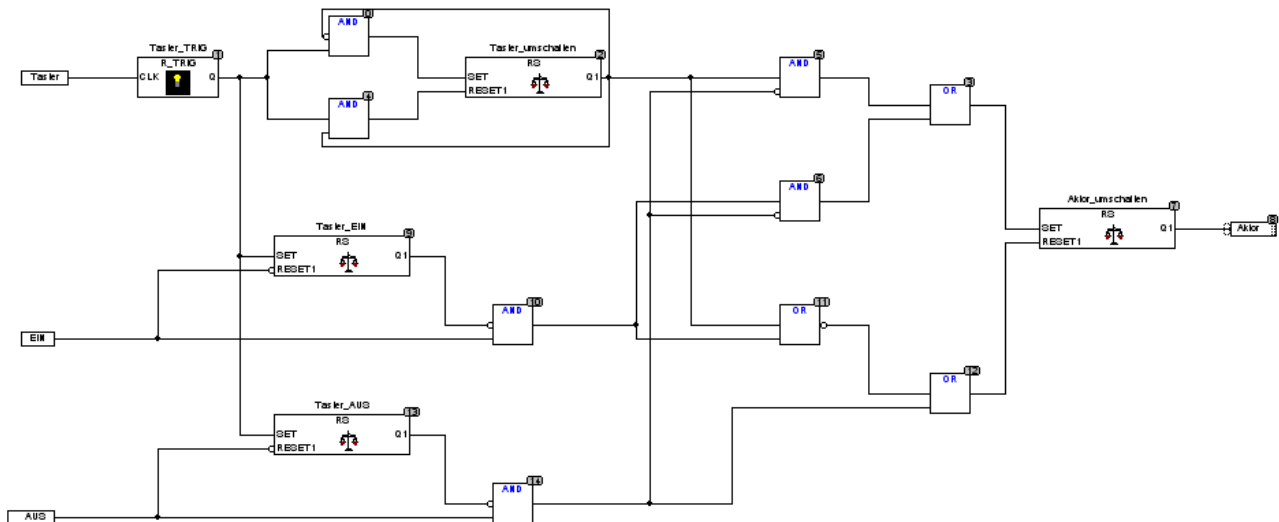


Abbildung 14.2: Funktionsblock umschalten Quellcode

Bei diesem speziellen Funktionsblock ist der Handbetrieb wieder vorrangig. Nachteil ist jedoch. Liegt an den Eingängen „EIN“ oder „AUS“ der boolische Wert „TRUE“ an, muss zunächst einmal der Taster betätigt werden, um die zentrale Funktion aufzuheben. Erst dann kann der Aktor wieder normal umgeschaltet werden.

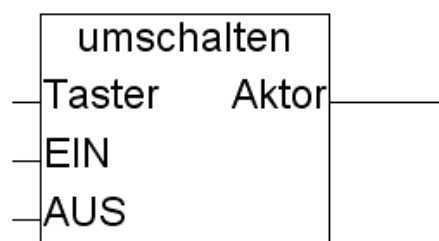


Abbildung 14.3: Funktionsblock umschalten

7.6 Tasterauswertungen:

Weil es in einigen Räumen im Haus mehr Aktoren als Taster gibt, müssen einige Taster doppelt belegt werden. Hierfür stellt die Bibliothek „Gebäude allgemein“ zwei Funktionsblöcke („Fb_Klick“ und „Fb_KurzLang“) zur Verfügung. Der Funktionsblock „Fb_Klick“ wertet aus, ob der Taster einfach oder doppelt betätigt wurde. Der Funktionsblock „Fb_KurzLang“ wertet aus, ob der Taster kurz und lange gedrückt wurde.

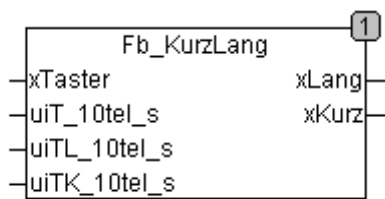


Abbildung 14.4: Funktionsblock
Fb_KurzLang

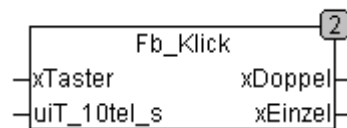


Abbildung 14.5: Funktionsblock
Fb_Klick

7.7 Der Funktionsblock HeizungFB

Für die Heizung wurde ebenfalls ein spezieller Funktionsblock programmiert.

Variablen:

```
0001 FUNCTION_BLOCK HeizungFB
0002 VAR_INPUT
0003     Frostschutz_Temperatur:REAL;
0004     Temperatur_IST:REAL;
0005     Temperatur_SOLL:REAL;
0006     xZenAus:BOOL;
0007 END_VAR
0008 VAR_OUTPUT
0009     EIN: BOOL;
0010 END_VAR
0011 VAR
0012 END_VAR
```

Abbildung 14.6: Funktionsblock HeizungFB
Variablen

Funktionsblock:

```
0001 IF Temperatur_IST>Frostschutz_Temperatur THEN
0002     IF xZenAUS=FALSE THEN
0003         IF Temperatur_SOLL-Temperatur_IST>0.5 THEN
0004             EIN:=TRUE;
0005         ELSE
0006             EIN:=FALSE;
0007         END_IF
0008     ELSE
0009         EIN:=FALSE;
0010     END_IF
0011 ELSE
0012     EIN:=TRUE;
0013 END_IF
```

Abbildung 14.7: Funktionsblock HeizungFB Quelltext

Die höchste Priorität bei der dem Funktionsblock „Heizung“ hat der Frostschutz. Sinkt die Temperatur unter den Wert der Frostschutztemperatur, muss die Heizung eingeschaltet werden, egal ob jemand im Haus ist oder nicht. Bei der Temperaturdifferenz ist eine Toleranz von 0,5 ° C erlaubt. Übersteigt die Differenz diesen Wert, schaltet sich die Heizung ein.

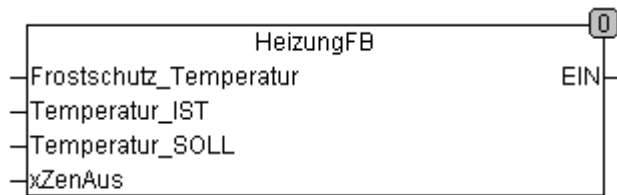


Abbildung 14.8: Funktionsblock HeizungFB

7.8 Die „Lichtorgel“

Die „Lichtorgel“ befindet sich im Schlafzimmer. Hierbei werden zehn verschieden farbige LEDs welche abwechselnd ein- und ausgedimmt werden.

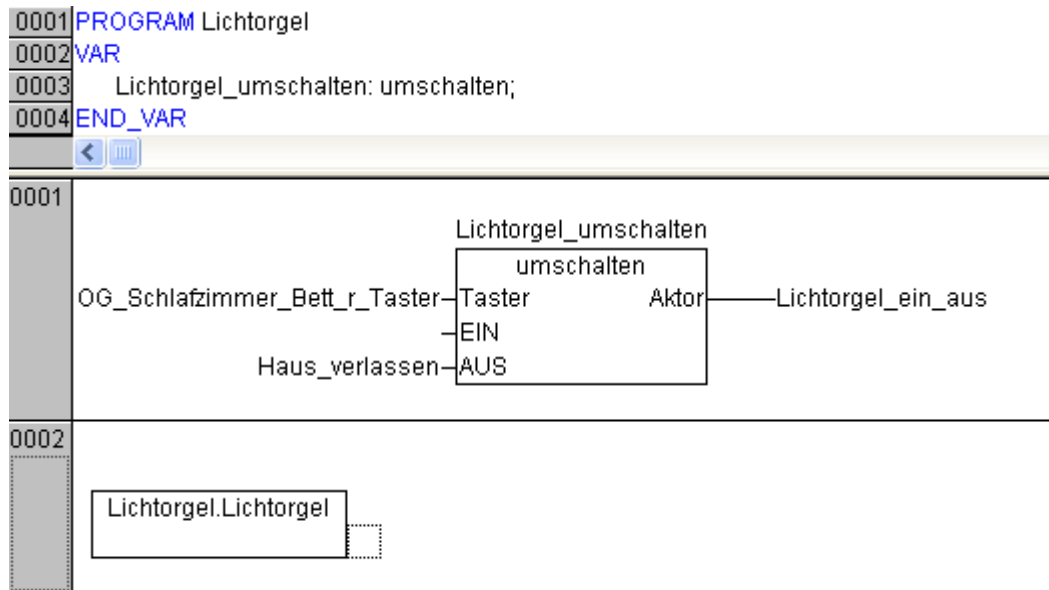


Abbildung 15.1: Lichtorgel umschalten

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Programm:

```
0001 IF Lichtorgel_ein_aus=TRUE THEN
0002   Light_Blue:=Light_Blue+15;
0003   IF Light_Blue > 65535 THEN
0004     Light_Blue:=45000;
0005   END_IF
0006   Light_Yellow:=Light_Yellow+25;
0007   IF Light_Yellow > 65535 THEN
0008     Light_Yellow:=45000;
0009   END_IF
0010   Light_Green:=Light_Green+30;
0011   IF Light_Green > 65535 THEN
0012     Light_Green:=45000;
0013   END_IF
0014   Light_Red:=Light_Red+40;
0015   IF Light_Red > 65535 THEN
0016     Light_Red:=45000;
0017   END_IF
0018   Light_White:=Light_White+45;
0019   IF Light_White > 65535 THEN
0020     Light_White:=45000;
0021   END_IF
0022 ELSE
0023   Light_Blue:=0;
0024   Light_Yellow:=0;
0025   Light_Green:=0;
0026   Light_Red:=0;
0027   Light_White:=0;
0028 END_IF
```

Abbildung 15.2: Lichtorgel Quelltext

8 Enocean

8.1 Funkreceiver Enocean (750-642)



Abbildung 16.1: Enocean Funkreceiver (750-642)

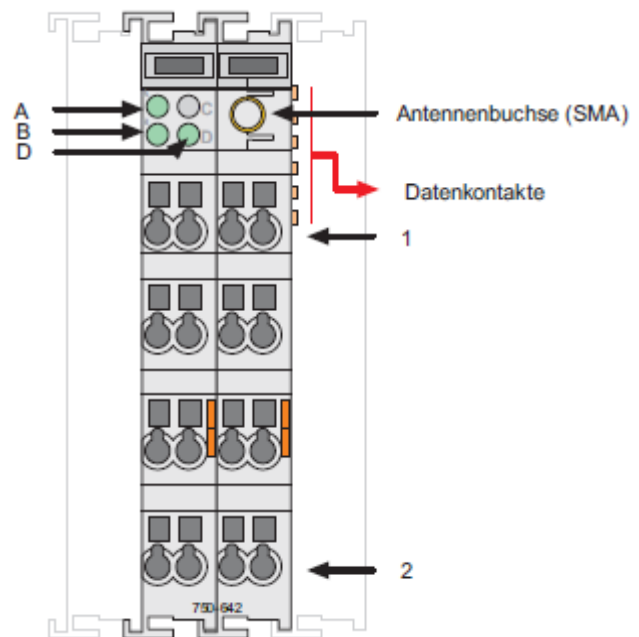


Abbildung 16.2: Enocean Funkreceiver (Schematischer Aufbau)

Der Enocean Funkreceiver nimmt Funksignale auf und gibt diese an den Klemmenbus weiter. Enocean stellt verschiedene Funksensoren her, welche größtenteils über eine Piezotechnologie verfügen. Die Sensoren arbeiten daher ohne zusätzliche Stromversorgung und brauchen keine Batterien oder ähnliches.

8.2 Die Implementierung und Programmierung der EnOcean-Klemme von Wago

Die Firma Wago stellt zur Implementierung der EnOcean-Klemme 750-642 die Bibliothek „EnOcean04_d_lib“ mit speziellen Funktionsblöcken zur Verfügung.

Der Baustein „FbEnOceanReceive“ verwaltet die Kommunikation zwischen dem EnOcean Funkreceiver und der Wago SPS.

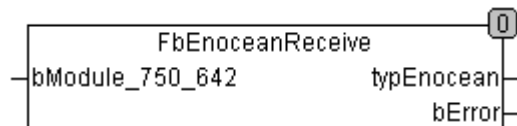


Abbildung 16.3: EnOcean FBReceive Baustein

An den Eingang „bModule_750_642“ wird der Klemmenindex des Funkreceivers angelegt. Jeder zusätzlich gesteckte Receiver bekommt seinen eigenen Klemmenindex.

8.3 EnOcean Komponenten

8.3.1 4-fach-Taster

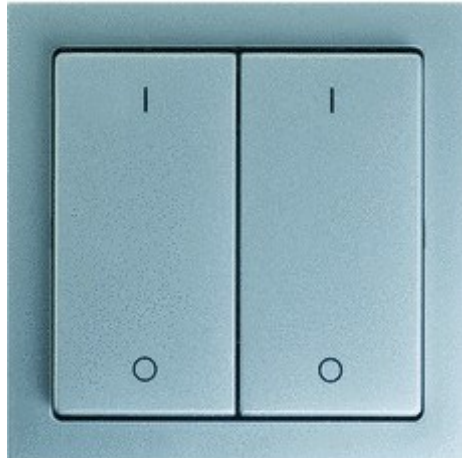


Abbildung 16.4: EnOcean 4-fach Taster

Implementierung 4-fach-Taster:

Zur Implementierung des 4-fach-Tasters gibt es einen speziellen Funktionsblock „FbButton_4_Channel“ in der Bibliothek „EnOcean04_d_lib“.

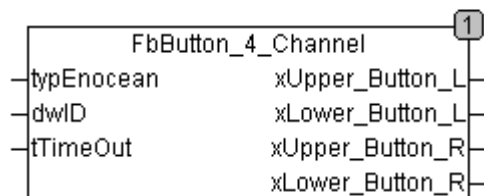


Abbildung 16.5: EnOcean FbButton_4_Channel

Der Eingang „typEnOcean“ des Tasters muss mit dem Ausgang des jeweiligen Funkreceivers verbunden werden. Jeder EnOcean Sensor bekommt eine eigene „ID“. Diese muss an den jeweiligen Eingängen angelegt werden.

8.3.2 Thermokon Temperatursensor SR04:



Abbildung 16.6: Thermokon Sensor

Der Temperatursensor SR04 nimmt Temperaturwerte auf und verfügt zusätzlich über einen Drehschalter, welcher Werte in einem definiertem Bereich einstellen kann.

Implementierung Temperatursensor SR04:

Für den Temperatursensor steht ebenfalls ein Funktionsblock in der Bibliothek „Enocean04_d_lib“ zur Verfügung.

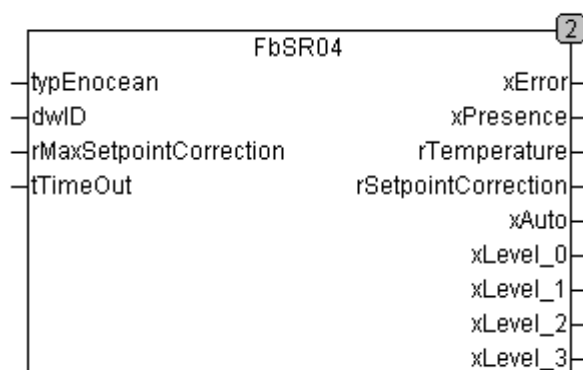


Abbildung 16.7: Enocean fbSR04

Auch dieser EnOcean-Sensor muss mit dem jeweiligen Funkreceiver am Eingang „typEnOcean“ verbunden werden und bekommt eine eigene ID. Am Eingang „rMaxSetpointCorrection“ wird der Wertebereich eingestellt, den man über den Drehschalter am Sensor verstellen kann.

9 RTC-Modul (750-640)



Abbildung 17.1: RTC-Modul (750-640)

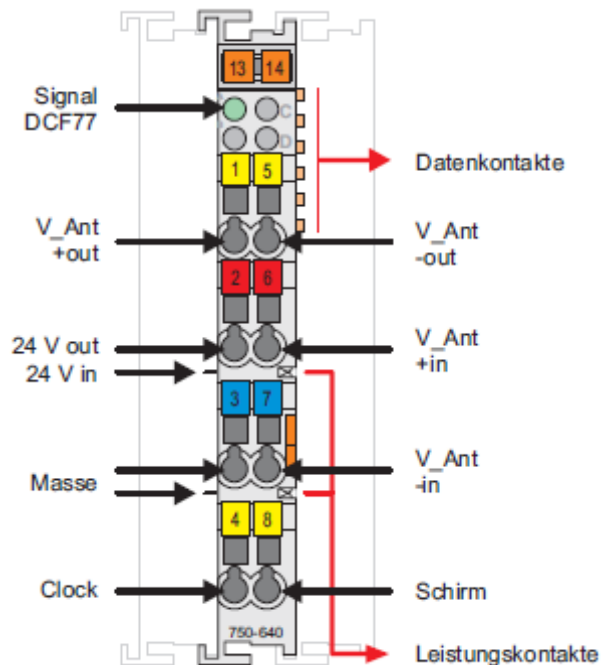


Abbildung 17.2: RTC-Modul (Schematischer Aufbau)

Das RTC-Modul stellt der Steuerung die aktuelle Uhrzeit zur Verfügung. Diese Uhrzeit kann mit Hilfe eines Stützkondensators selbst nach einem Spannungsausfall noch über eine gewisse Zeit gehalten werden.

9.1 Die Programmierung des RTC-Moduls

Weil für das RTC-Modul noch keine Bibliotheken verfügbar sind, war die Programmierung etwas umständlich.

Das RTC-Modul hat einen logischen Kanal worüber 6 Bytes im Ein- und Ausgangsprozessabbild zur Verfügung gestellt werden. Die Sende- und Empfangsdaten werden in 4 Ein- und Ausgangsbytes (D0 ... D3) abgelegt. 1 Steuerbyte (C) und 1 Statusbyte (S) dienen zur Kontrolle des Funktionsaufrufs. 1 Befehlsbyte (ID) dient zur Auswahl und Bestätigung der auszuführenden Funktion.

Eingangsdaten		Ausgangsdaten	
S	Statusbyte	C	Steuerbyte
ID	Funktion-Auswahl-ID	ID	Funktions-Auswahl-ID
D0	Datenbyte 0	D0	Datenbyte 0
D1	Datenbyte 1	D1	Datenbyte 1
D2	Datenbyte 2	D2	Datenbyte 2
D3	Datenbyte 3	D3	Datenbyte 3

Abbildung 17.3: RTC-Modul Ein- und Ausgangsdaten

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Übersicht Funktionen:

ID	Funktion	Beschreibung	Seite
0x00	IDLE	Kein Auftrag	34
Datum- und Uhrzeit-Funktionen			
0x01	GET_TOD100	Uhrzeit in 100µs seit 00:00 auslesen	35
0x09	GET_CURRENT_TOD100	Uhrzeit in 100µs seit 00:00 auslesen, (fortlaufend)	36
0x02	GET_TOD	Uhrzeit in Millisekunden seit 00:00 auslesen	37
0x0A	GET_CURRENT_TOD	Uhrzeit in Millisekunden seit 00:00 auslesen, (fortlaufend)	38
0x03	GET_DT	Uhrzeit in Sekunden seit 1.1.1970 00:00 auslesen	39
0x0B	GET_CURRENT_DT	Uhrzeit in Sekunden seit 1.1.1970 00:00 auslesen, (fortlaufend)	40
0x04	GET_UTC	Uhrzeit in Sekunden seit 1.1.1970 00:00 (UTC) auslesen	41
0x0C	GET_CURRENT_UTC	Uhrzeit in Sekunden seit 1.1.1970 00:00 (UTC) auslesen, (fortlaufend)	42
0x05	GET_DATE	Datum in Sekunden seit 1.1.1970 00:00 auslesen	43
0x0D	GET_CURRENT_DATE	Datum in Sekunden seit 1.1.1970 00:00 auslesen, (fortlaufend)	44
0x06	GET_MS	Millisekundenanteil der letzten GET_DT Abfrage auslesen	45
0x07	GET_WEEKDAY	Wochentag auslesen	46
0x0F	SET_DT	Uhrzeit in Sekunden seit 1.1.1970 00:00 setzen	47
Timer-Funktionen			
0x10	GET_CHANNEL	Schaltzustände der Schaltkanäle auslesen	48
0x11	SET_CHANNEL	Schaltzustände der Schaltkanäle setzen	49
0x12	CLEAR_CHANNEL	Schaltzustände der Schaltkanäle zurücksetzen	50
0x13	GET_AUTO	Automatikzustände der Schaltkanäle auslesen	51
0x14	SET_AUTO	Automatikzustände der Schaltkanäle setzen	52

Abbildung 17.4: RTC-Modul Funktionsübersicht Teil 1

ID	Funktion	Beschreibung	Seite
Adress-Funktionen			
0x20	GET_ADDR	Adress-Offset für Konfigurationsdaten oder Betriebsstundenzähler auslesen	53
0x21	SET_ADDR	Adress-Offset für Konfigurationsdaten oder Servicedaten setzen	54
Servicedaten-Funktionen			
0x30	GET_OPH	Servicedaten auslesen	55
0x31	GET_OPH_INC	Servicedaten auslesen, (autoinkrement)	56
0x32	SET_OPH	Servicedaten setzen	57
0x33	SET_OPH_INC	Servicedaten setzen, (autoinkrement)	58
0x34	GET_SERVICE_FLAGS	Wartungsanzeige der Betriebsstundenzähler auslesen	59
Konfigurationsdaten-Funktionen			
0x38	GET_CNF	Konfigurationsdaten auslesen	60
0x39	GET_CNF_INC	Konfigurationsdaten auslesen, (autoinkrement)	61
0x3A	SET_CNF	Konfigurationsdaten auslesen	62
0x3B	SET_CNF_INC	Konfigurationsdaten setzen, (autoinkrement)	63
0xFF	GET_DIAG	Selbsttest-Diagnose auslesen	64

Abbildung 17.5: RTC-Modul Funktionsübersicht Teil 2

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Beispiel: Uhrzeit und Datum einstellen:

Mit der Funktion SET-DT wird das Datum und die Uhrzeit eingestellt. Dabei werden die verstrichenen Sekunden seit dem 01.01.1970 00:00 eingegeben. Für den 13.04.2010 12.00 Uhr gilt dann:

13. April 2010 12:00:00 = 1271160000 (dez) = 4BC45CC0 (hex)

Ausgangsdaten: D0=C0 D1=5C D2=C4 D3=4B

Funktionsaufruf (Request)								
Byte	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
C	0	0	0	0	0	0	0	TR
ID	0x0F							
D0	Datum und Uhrzeit (Low-Byte)							
D1	---							
D2	---							
D3	Datum und Uhrzeit (High-Byte)							

Abbildung 17.6: RTC-Modul Funktionsaufruf

Funktionsantwort (Response)								
Byte	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
S	0	0	0	0	0	0	0	TA
ID	0x0F							
D0	0x00							
D1	0x00							
D2	0x00							
D3	0x00							

Abbildung 17.7: RTC-Modul Funktionsantwort

Die Funktion

Die Funktion GET_CURRENT_TOD von RTC-Modul liefert die aktuelle Uhrzeit die seit Beginn des Tages (00:00 Uhr) vergangen ist. Anders als bei der Funktion GET_TOD wird die Funktion GET_CURRENT_TOD mit jedem neuen Buszyklus aktualisiert, somit kann sie wie einige richtige Uhr mit Millisekundenzähler auf dem Bildschirm angezeigt werden.

Funktionsaufruf (Request)								
Byte	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
C	0	0	0	0	0	0	0	TR
ID	0x0A							
D0	0xXX							
D1	0xXX							
D2	0xXX							
D3	0xXX							

Abbildung 17.8: RTC-Modul Funktionsaufruf

Funktionsantwort (Response)								
Byte	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
S	0	0	0	0	0	0	0	TA
ID	0x0A							
D0	Uhrzeit (Low-Byte)							
D1	...							
D2	...							
D3	Uhrzeit (High-Byte)							

Abbildung 17.9: RTC-Modul Funktionsantwort

Die Funktions-ID lautet 0x0A. Dieser Wert muss am Ausgang des RTC-Moduls (RTC_AUSGANG_ID) anliegen. Die Werte restlichen Ausgänge (D0, D1, D2, D3) sind bei dieser Funktion irrelevant. Der Ausgangswert kann nun an den Datenbytes der Funktionsantwort (Response) abgefragt werden. Die Uhrzeit wird im Format TOD (Time Of Day) in 1ms Schritten angegeben und ist insgesamt 4 Byte lang. Das Format TOD ist im Prinzip nicht anderes als ein Double Word (DWORD) und setzt sich aus den einzelnen Datenbytes zusammen.

Beispiel: 08:45:35 Uhr

$$(8*60+45)*60+35)*1000 = 31535000 = 0x01E112F998$$

Eingangsdaten: D0 = 0x98, D1 = 0x2F, D2 = 0xE1, D3 = 0x01

Die vier Bytes müssen nun zusammen in ein DWORD-Format gespeichert werden. Dazu werden die einzelnen Bytes zunächst in einzelne DWORD-Variablen gespeichert. Die CoDeSys hat dafür eine spezielle Funktion (BYTE_TO_DWORD)

(Programmauszug in ST)

D0:=BYTE_TO_DWORD(RTC_Eingang_D0); (Datenbyte D0)

D1:=BYTE_TO_DWORD(RTC_Eingang_D1); (Datenbyte D1)

D2:=BYTE_TO_DWORD(RTC_Eingang_D2); (Datenbyte D2)

D3:=BYTE_TO_DWORD(RTC_Eingang_D3); (Datenbyte D3)

Datenbyte	BYTE	DWORD
D0	0x98=1001 1000	0000 0000 0000 0000 0000 0000 1001 1000
D1	0x2F=0010 1111	0000 0000 0000 0000 0000 0000 0010 1111
D2	0xE1=1110 0001	0000 0000 0000 0000 0000 0000 1110 0001
D3	0x01=0000 0001	0000 0000 0000 0000 0000 0000 0000 0001

Beim konvertieren von BYTE in DWORD, wird der Wert immer in das low-Byte gespeichert. Nun muss eine Funktion programmiert werden, die alle vier DWORD-Variablen (D0, D1, D2, D3) wieder in eine einzelne DWORD-Variable speichert. Das geschieht nun in AWL.

Mit der Funktion „LD“ werden die einzelnen Variablen in einen sogenannten Akkumulator (Zwischenspeicher oder Puffer-Speicher) gespeichert. Da der Wert D3 später an letzter Stelle im Double Word stehen muss, wird er als erstes geladen.

LD D3 (D3 wird in dem Akkumulator geladen)

Aktueller Wert im Akkumulator : 0000 0000 0000 0000 0000 0000 0000 0001

Nun muss das komplette Byte bitweise um jeweils 24 Stellen verschoben. Dies geschieht mit dem Befehl „ROL“ (Left Rotate (nach Links verschieben)). Dieser Befehl kann mit einem Schieberegister aus der Digitaltechnik verglichen werden.

ROL 24 (Nach Links verschieben um 24 Stellen)

Aktueller Wert im Akkumulator : 0000 0001 0000 0000 0000 0000 0000 0000

Der Wert des Bytes steht jetzt an der letzten Stelle im DWORD. Mit dem Befehl ST wird der Wert im Akkumulator nun in ein Zwischenregister (hier TEMP) gespeichert.

ST TEMP (Wert im Akkumulator in die Variable TEMP speichern)

TEMP = 0000 0001 0000 0000 0000 0000 0000 0000

Jetzt wird in den Akkumulator das nächste Datenbyte geladen.

LD D2 (D2 wird in den Akkumulator geladen)

Aktueller Wert im Akkumulator : 0000 0000 0000 0000 0000 0000 1110 0001

Dieses Byte wird um 16 Stellen verschoben.

ROL 16 (Nach Links verschieben um 16 Stellen)

Aktueller Wert im Akkumulator : 0000 0000 1110 0001 0000 0000 0000 0000

Mit einer Oder-Verknüpfung wird dieser Wert nun bitweise mit dem TEMP-Register verknüpft. Das Ergebnis wird wieder im Akkumulator gespeichert.

OR TEMP (Aktuellen Wert im Akkumulator mit Register TEMP Oder-Verknüpfen)

Akkumulator : 0000 0000 1110 0001 0000 0000 0000 0000

OR

TEMP : 0000 0001 0000 0000 0000 0000 0000 0000

Ergebnis : 0000 0001 1110 0001 0000 0000 0000 0000

Nun wird der Wert im Akkumulator wieder ins TEMP-Register geschrieben.

ST TEMP (Wert im Akkumulator in die Variable TEMP speichern)

TEMP = 0000 0001 1110 0001 0000 0000 0000 0000

Jetzt kann wieder das nächste Datenbyte in den Akkumulator geschrieben werden.

LD D1 (D1 wird in den Akkumulator geladen)

Aktueller Wert im Akkumulator : 0000 0000 0000 0000 0000 0000 0010 1111

Dieses Byte wird nun um 8 Stellen nach Links verschoben.

ROL 16 (Nach Links verschieben um 8 Stellen)

Aktueller Wert im Akkumulator : 0000 0000 0000 0000 0010 1111 0000 0000

Dieser Wert wird nun wieder mit dem TEMP-Register verknüpft.

OR TEMP (Aktuellen Wert im Akkumulator mit Register TEMP Oder-Verknüpfen)

Akkumulator : 0000 0000 0000 0000 0010 1111 0000 0000

OR

TEMP: : 0000 0001 1110 0001 0000 0000 0000 0000

Ergebnis : 0000 0001 1110 0001 0010 1110 0000 0000

Das Ergebnis wird in nochmal ins TEMP-Register gespeichert.

ST TEMP (Wert im Akkumulator in die Variable TEMP speichern)

TEMP = 0000 0001 1110 0001 0010 1110 0000 0000

Nun fehlt noch das letzte Datenbyte.

LD D0 (D0 wird in den Akkumulator geladen)

Aktueller Wert im Akkumulator : 0000 0000 0000 0000 0000 0000 1001 1000

Das Byte von D0 steht bereits an richtiger Stelle und muss nicht mehr verschoben werden.

OR TEMP (Aktuellen Wert im Akkumulator mit Register TEMP Oder-Verknüpfen)

Aktueller Wert im Akkumulator : 0000 0001 1110 0001 0010 1110 1001 1000

Der Wert kann nun an das entsprechende Programm zurück gegeben werden.

ST ROT (Gibt den aktuelle Wert im Akkumulator als Rückgabewert zurück)

0000 0001 1110 0001 0010 1110 1001 1000 = 0x01E112F998

Nun kann dieser Wert im Hauptprogramm geladen und konvertiert werden.

Programmauszug in ST:

Ergebnis:=ROT(D3, D2, D1, D0); (Funktion aufrufen mit den
Eingangsvariablen D0, D1, D2 und D3)

Time:=**DWORD_TO_TOD**(Ergebnis); (Variable von DWORD nach TOD
konvertieren)

10 Smart Metering

10.1 3-Phasen-Leistungsmessklemme (750-493)



Abbildung 18.1: 3-Phasen-Leistungsmessklemme (750-493)

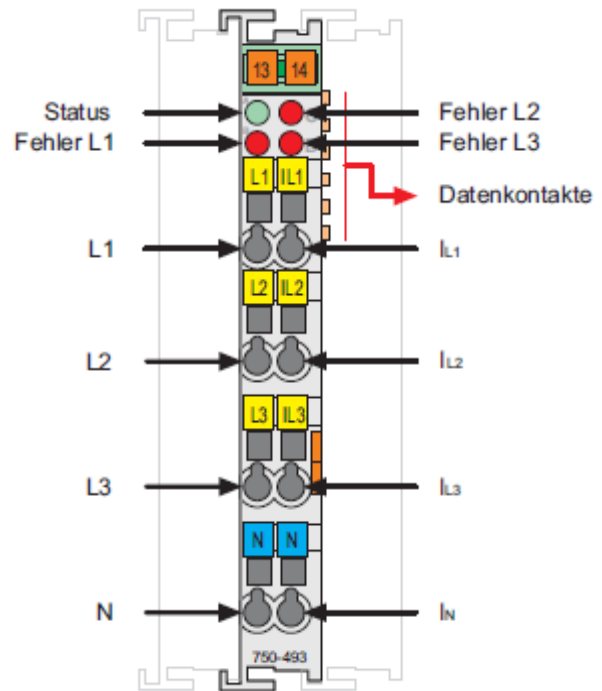
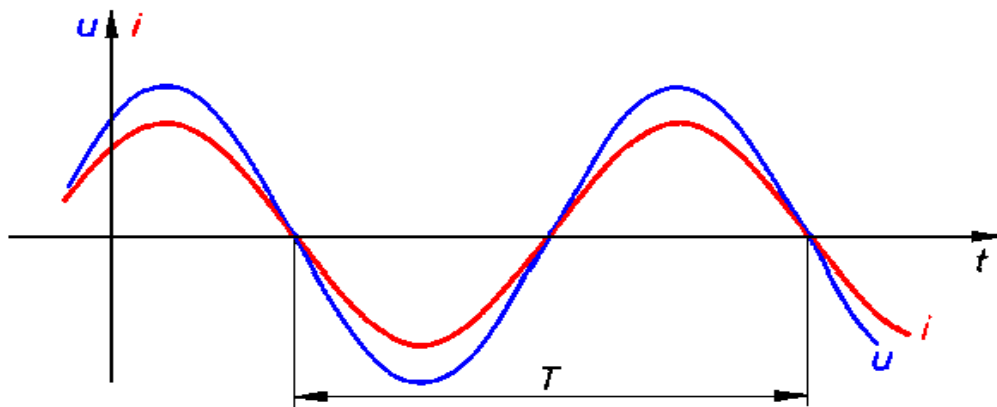


Abbildung 18.2: 3-Phasen-Leistungsmessklemme (Schematischer Aufbau)

Die 3-Phasen-Leistungsmessklemme misst sinus- und nichtsinusförmige Spannungs-, Strom- und Leistungswerte aus einem dreiphasigem Versorgungsnetz. Die Messklemme tastet dabei die Spannungs-, Strom- und Leistungsverläufe ab und bildet aus diesen gemessenen Werten den quadratischen Mittelwert (Effektivwert). Für die Messungen stehen der 3-Phasen-Leistungsmessklemme insgesamt sechs Analog-/Digitalwandler zur Verfügung, daher können die Berechnungen bereits innerhalb der Klemme gelöst werden, ohne viel Rechenleistung des Knotens in Anspruch zu nehmen.

Messprinzip

Die 3-Phasen-Leistungsmessklemme stellt für den Klemmenbus die Effektivwerte zur Verfügung. Um diese zu bestimmen, wird das Sinussignal der Spannung und des Stromes jeweils mit einer Abtastfrequenz von c.a. 16 μs abgetastet. Das bedeutet das für eine Wechselspannung mit einer Frequenz von c.a. 50 Hz und einer Periodendauer von 20ms werden etwa 1250 Werte gemessen. Alle drei Phasen werden Zeitsynchron gemessen.



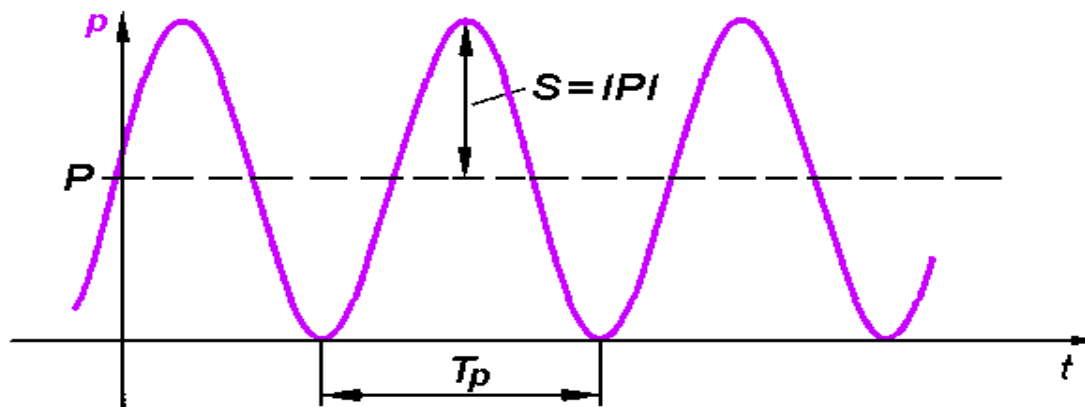
$$U = \sqrt{\frac{1}{n} \sum_{1}^n u_{(t)}^2}$$

$$I = \sqrt{\frac{1}{n} \sum_{1}^n i_{(t)}^2}$$

Zur Berechnung der Effektivwerte von Strom und Spannung werden die Momentanwerte der Messung quadriert, aufsummiert und zum Schluss wird die Wurzel gezogen.

Leistungen:

Zur Berechnung der Wirkleistung werden die Momentanwerte von Strom und Spannung miteinander multipliziert. Das Ergebnis dieser Multiplikation wird dann durch die Anzahl der Messungen dividiert.



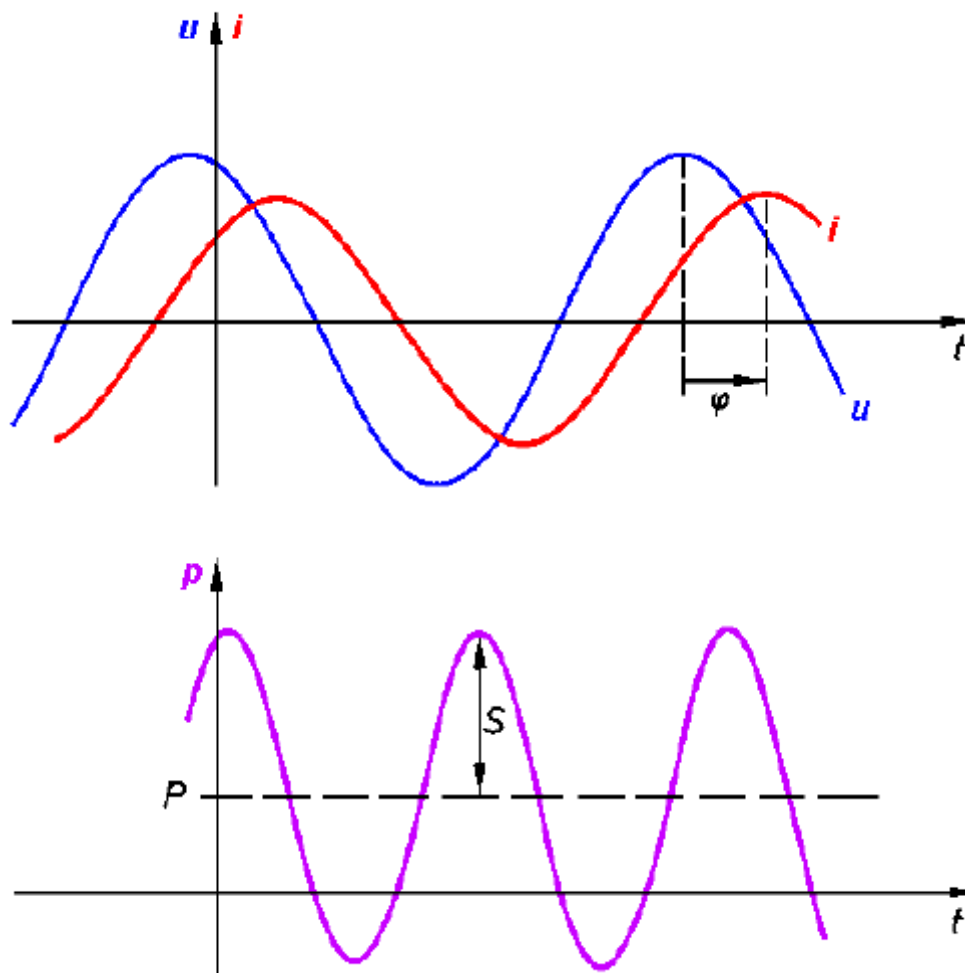
$$P = \frac{1}{n} \sum_{1}^n u_{(t)} \cdot i_{(t)}$$

$$s_{(t)} = u_{(t)} \cdot i_{(t)}$$

Die Leistung $s(t)$ ist die Momentanleistung, welche zu jedem Abtastschritt berechnet wird. Aus dieser Momentanleistung wird dann der Mittelwert gebildet.

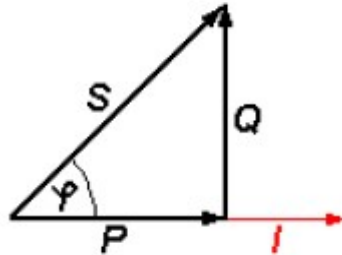
Berechnung der Scheinleistung:

Wenn ein Stromkreis außer den ohmschen Verbrauchern auch induktive oder kapazitive Verbraucher hat, kommt es zwischen den Strom und Spannungsverläufen zur einer Phasenverschiebungen. Die gemessenen Effektivwerte der Strom- und Spannungsverläufe werden dadurch nicht beeinflusst. Aus dem Produkt der Effektivwerte von Strom- und Spannung ergibt sich die Scheinleistung.



Kommt es bei einer Messung zu einer besagten Phasenverschiebung, dann muss außer der Wirkleistung zusätzlich noch die Blind- und Scheinleistung sowie der Leistungsfaktor $\cos\varphi$ (φ :Phasenverschiebungswinkel) bestimmt werden.

Für Wirk-, Blind und Scheinleistung und dem Phasenverschiebungswinkel gelten folgende Beziehung:



Die Scheinleistung berechnet sich aus den Produkten der Effektivwerte von Strom und Spannung.

$$S = U \cdot I$$

Die Blindleistung kann mit Hilfe vom Satz des Pythagoras berechnet werden. Dabei sind die Wirk- und die Blindleistung die beiden Katheten und die Scheinleistung die Hypotenuse des Rechtwinkligen Dreiecks.

$$Q = \sqrt{S^2 - P^2}$$

Das Verhältnis aus Wirk- und Scheinleistung ergibt dann den Leistungsfaktor $\cos \varphi$.

$$\cos \varphi = \frac{P}{S}$$

Der Phasenverschiebungswinkel kann später in CoDeSys mit Hilfe der Funktion „ACOS“ (Arkusfunktion vom Kosinus) berechnet werden.

Prozessabbild:

Die Daten der 3-Phasen-Leistungsmessklemme werden über 3 logische Kanäle in 9 Ein- und Ausgangsdatenbytes auf die SPS-Anlage übertragen. 3 Steuer- und Statusbytes dienen zur Auswahl der Prozessdaten und der Einstellung der Klemme. Die Sende- und Empfangsdaten werden in insgesamt 6 Datenbytes übertragen.

Eingangsdaten		Ausgangsdaten	
S0	Statusbyte 0	C0	Controlbyte 0
D0	Eingangsdatenwort 1 (LSB)	D0	Ausgangsdatenwort 1 (LSB)
D1	Eingangsdatenwort 1 (MSB)	D1	Ausgangsdatenwort 1 (MSB)
S1	Statusbyte 1	C1	Controlbyte 1
D2	Eingangsdatenwort 2 (LSB)	D2	Ausgangsdatenwort 2 (LSB)
D3	Eingangsdatenwort 2 (MSB)	D3	Ausgangsdatenwort 2 (MSB)
S2	Statusbyte 2	C2	Controlbyte 2
D4	Eingangsdatenwort 3 (LSB)	D4	Ausgangsdatenwort 3 (LSB)
D5	Eingangsdatenwort 3 (MSB)	D5	Ausgangsdatenwort 3 (MSB)

Abbildung18.3: 3-Phasen-Leistungsmessklemme Ein- und Ausgangsdaten

Prozesswerte:

Prozesswerte der Klemme 750-493 und 750-493/000-001			
Eingangs- spannung U_L 0 ... 500 V	Zahlenwert		
	Binär Ausgabewert	Hex.	Dez.
0,0	'0000.0000.0000.0000'	0x0000	0
50,0	'0000.0001.1111.0100'	0x01F4	500
100,0	'0000.0011.1110.1000'	0x03E8	1000
150,0	'0000.0101.1101.1100'	0x05DC	1500
200,0	'0000.0111.1101.0000'	0x07D0	2000
250,0	'0000.1001.1100.0100'	0x09C4	2500
300,0	'0000.1011.1011.1000'	0x0BB8	3000
350,0	'0000.1101.1010.1100'	0x0DAC	3500
400,0	'0000.1111.1010.0000'	0x0FA0	4000
450,0	'0001.0001.1001.0100'	0x1194	4500
500,0	'0001.0011.1000.1000'	0x1388	5000

Abbildung 18.4: 3-Phasen-Leistungsmessklemme Prozesswerte Spannung

Prozesswerte der Klemme 750-493			
Eingangs- strom I_L 0 ... 1 A	Zahlenwert		
	Binär Ausgabewert	Hex.	Dez.
0,000	'0000.0000.0000.0000'	0x0000	0
0,100	'0000.0000.0110.0100'	0x0064	100
0,200	'0000.0000.1100.1000'	0x00C8	200
0,300	'0000.0001.0010.1100'	0x012C	300
0,400	'0000.0001.1001.0000'	0x0190	400
0,500	'0000.0001.1111.0100'	0x01F4	500
0,600	'0000.0010.0101.1000'	0x0258	600
0,700	'0000.0010.1011.1100'	0x02BC	700
0,800	'0000.0011.0010.0000'	0x0320	800
0,900	'0000.0011.1000.0100'	0x0348	900
1,000	'0000.0011.1110.1000'	0x03E8	1000

Abbildung 18.5: 3-Phasen-Leistungsmessklemme Prozesswerte Strom

Prozesswerte der Klemme 750-493			
Wirkleistung P_L 0 ... 500 W	Zahlenwert		
	Binär Ausgabewert	Hex.	Dez.
0,0	'0000.0000.0000.0000'	0x0000	0
50,0	'0000.0001.1111.0100'	0x01F4	500
100,0	'0000.0011.1110.1000'	0x03E8	1000
150,0	'0000.0101.1101.1100'	0x05DC	1500
200,0	'0000.0111.1101.0000'	0x07D0	2000
250,0	'0000.1001.1100.0100'	0x09C4	2500
300,0	'0000.1011.1011.1000'	0x0BB8	3000
350,0	'0000.1101.1010.1100'	0x0DAC	3500
400,0	'0000.1111.1010.0000'	0x0FA0	4000
450,0	'0001.0001.1001.0100'	0x1194	4500
500,0	'0001.0011.1000.1000'	0x1388	5000

Abbildung 18.6: 3-Phasen-Leistungsmessklemme Prozesswerte Wirkleistung

10.2 Die Programmierung der 3-Phasen-Leistungsmessklemme

Zur Programmierung der 3-Phasen-Leistungsmessklemme stellt die Firma Wago die Bibliothek „PowerMeasurement_02.Lib“ zur Verfügung. In dieser Bibliothek gibt es den Baustein „Fb750_493_Master3Phase“, welchen die Klemme parametrisiert werden kann.

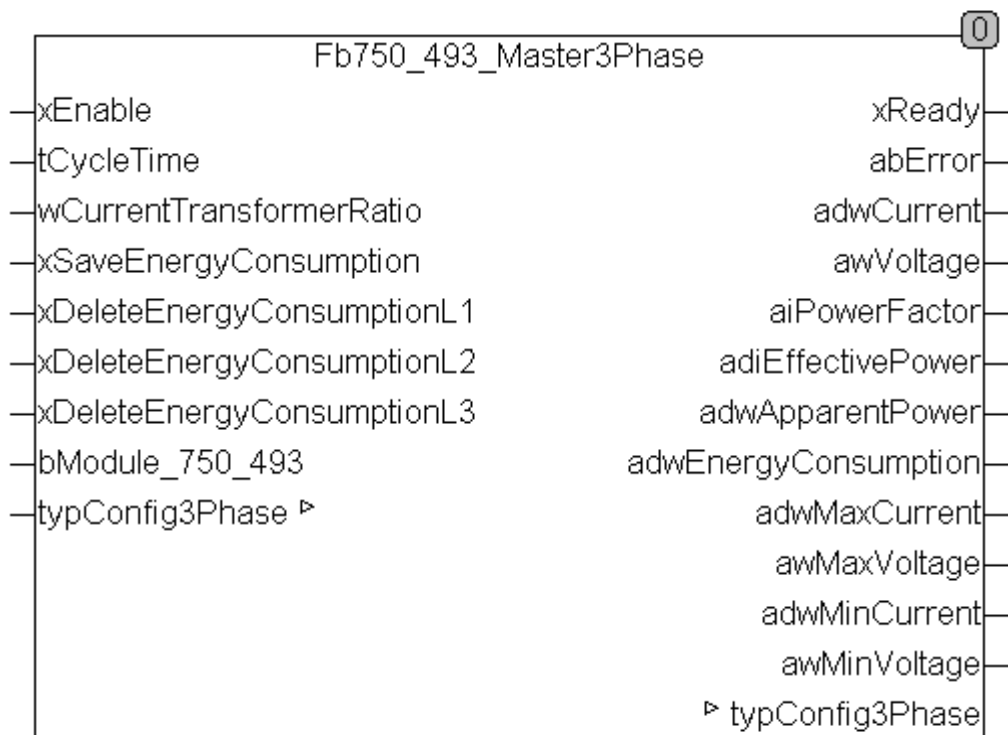


Abbildung 18.7: Funktionsbaustein Fb750_493_Master3Phase

Der Baustein liefert die Strom- und Spannungswerte sowie die Leistungen und den Leistungsfaktor. Diese Werte können dann in einer Energetischen Rechnung weiter verwendet werden.

10.3 Der Zählerbaustein

Für das Smart Metering musste ein spezieller Zählerbaustein programmiert werden. Dieser zählt die Betriebsdauer des Aktors sowie die aktuelle Arbeit.

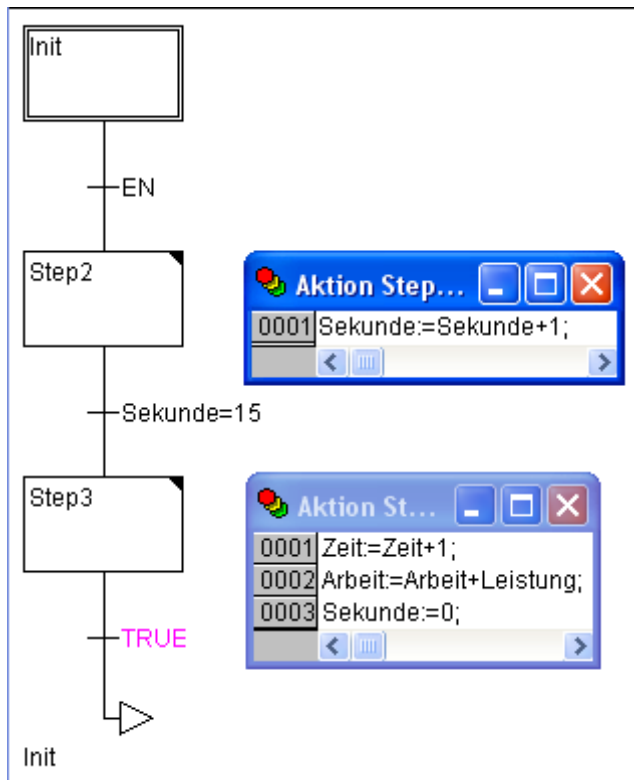


Abbildung 18.8: Funktionsblock Zaehlen

Der Baustein wurde mit der Ablaufsprache programmiert. Der Eingang „EN“ des Funktionsblockes ist die Starttransition. Sobald an dem Eingang der boolesche Wert „TRUE“ anliegt startet das Programm. Weil die Zykluszeit der Task deutlich unter einer Sekunde liegt muss eine Sekunde simuliert werden, dazu dient die Variable „Sekunde“. Der zweite Schritt wird 15 mal Wiederholt, erst dann wird eine Sekunde weiter gezählt.

10.4 Energetische Rechnung:

Mit Hilfe des Zählerbausteins kann ein Funktionsblock programmiert, welcher einige Energetische Berechnungen durchführt.

Variablen:

0001	FUNCTION_BLOCK Energetische_Rechnung
0002	VAR_INPUT
0003	Aktor:BOOL;
0004	Leistung_Aktor:INT;
0005	Kosten:REAL;
0006	END_VAR
0007	VAR_OUTPUT
0008	Laufzeit_Sek: DWORD;
0009	Laufzeit_Time: TIME;
0010	akt_Arbeit_Wh: REAL;
0011	akt_Arbeit_kWh: REAL;
0012	durchschn_Arbeit: REAL;
0013	akt_Kosten: REAL;
0014	kalk_Kosten: REAL;
0015	
0016	END_VAR
0017	VAR
0018	Zaehler: Zaehlen;
0019	END_VAR

Abbildung 18.9: Funktionsblock Energetische Rechnung Variablen

Quellcode:

0001	Zaehler(EN:=Aktor , Leistung:=INT_TO_DWORD(Leistung_Aktor));
0002	Laufzeit_Sek:=Zaehler.Zeit;
0003	akt_Arbeit_Wh:=DWORD_TO_REAL(Zaehler.Arbeit)/3600;
0004	Laufzeit_Time:=DWORD_TO_TIME(Laufzeit_Sek);
0005	akt_Arbeit_kWh:=akt_Arbeit_Wh/1000;
0006	akt_Kosten:=Kosten*akt_Arbeit_kWh;
0007	durchschn_Arbeit:=akt_Arbeit_kWh/INT_TO_REAL(DAY_OF_YEAR(akt_Date));
0008	kalk_Kosten:=durchschn_Arbeit*INT_TO_REAL(DAYS_IN_YEAR(akt_Date))*Kosten;

Abbildung 18.10: Funktionsblock Energetische Rechnung Quelltext

Der Zählerbaustein liefert die Betriebsdauer in Sekunden und die Arbeit in Wattsekunden. Mit einer Konvertierungsfunktion wird die Zeit für die Visualisierung von DWORD nach TIME konvertiert. Die aktuelle Arbeit wird von Wattsekunden in Wattstunden und in Kilowattstunden umgerechnet. Mit den Kilowattstunden werden die aktuellen Kosten berechnet.

Durchschnittliche Arbeit und Kalkulierte Kosten:

Für die durchschnittliche Arbeit wurden zwei spezielle Funktionen aus der Bibliothek OScat verwendet. Die Funktion „DAY_OF_YEAR“ liefert die Anzahl der Tage die bereits im aktuellen Jahr vergangen sind. Die Funktion „DAYS_IN_YEAR“ berechnet wie viele Tage das aktuelle Jahr insgesamt hat. Für beide Funktionen muss das aktuelle Datum vom RTC-Modul bereit gestellt werden.

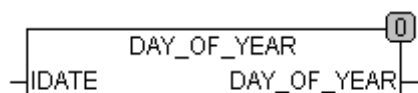


Abbildung 18.11: Funktionsblock Day_of_Year

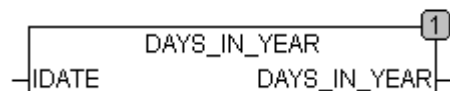


Abbildung 18.12: Funktionsblock Days_in_Year

Der Quotient aus den bereits aus den vergangenen Tagen und den gesamt Tagen bildet den Faktor für die Berechnung der Durchschnittlichen Arbeit. Die durchschnittliche Arbeit multipliziert mit dem Kosten liefert die kalkulierten Kosten für das aktuelle Jahr.

10.5 Die Visualisierung des Smart Meterings

Um die Ergebnisse der Energetischen Rechnungen in geeigneter Form darzustellen, wurde für jeden Raum eine Visualisierungsoberfläche angelegt.

Beispiel :Kinderzimmer

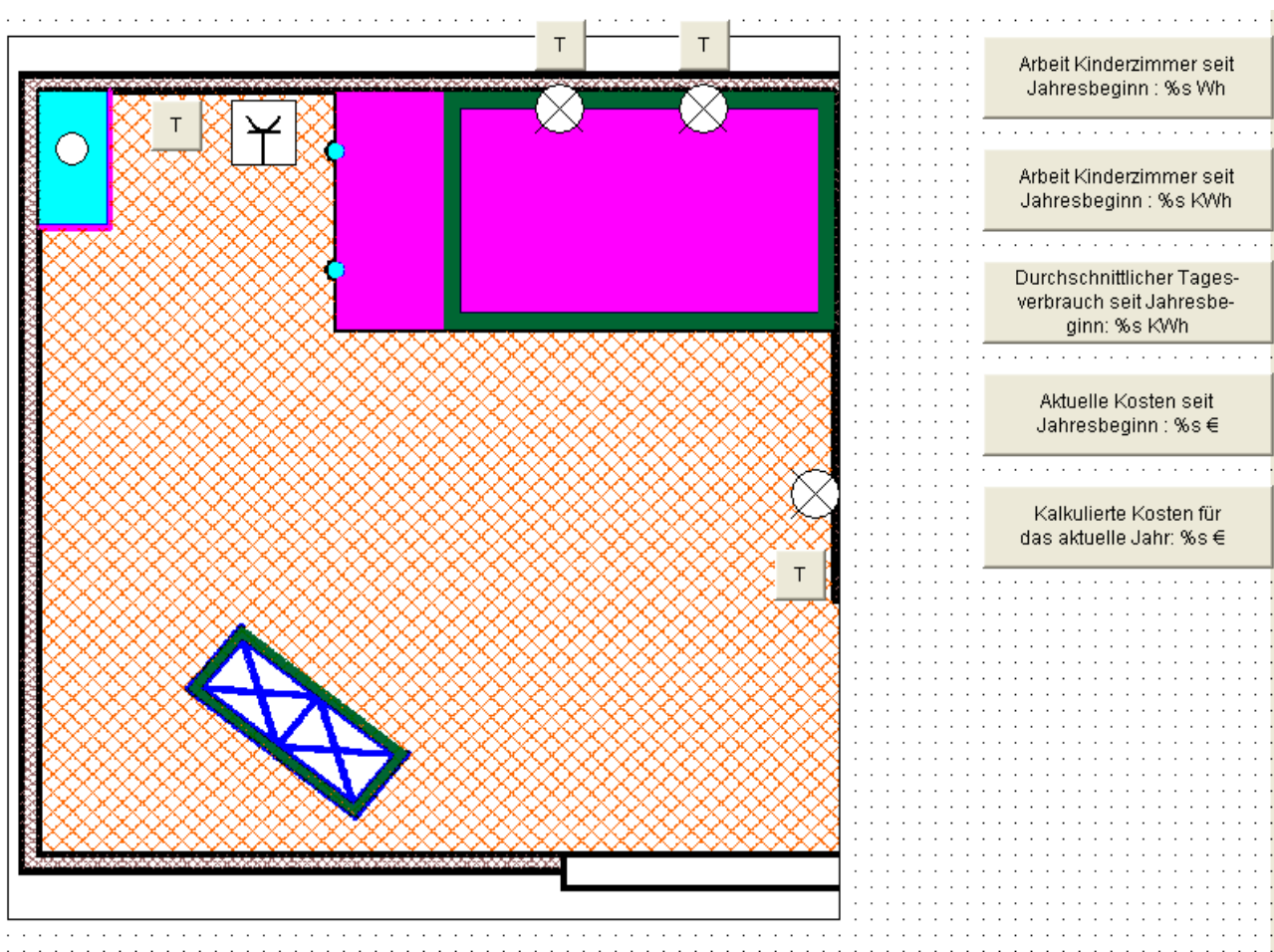


Abbildung 19.1: Visualisierung Kinderzimmer

Alle Geräte im Zimmer sind über Visualisierungselemente im Raum aufgeführt. Diese Visualisierungselemente sind mit den Systemvariablen verbunden und wechseln die Farbe sobald diese ihren Zustand ändern. Zudem können die Aktoren im Zimmer auch über die Visualisierung ein- oder ausgeschaltet werden. An der linken Seite sind die Ergebnisse der Energetischen Rechnungen für das gesamte Kinderzimmer aufgeführt. Um den Energieverbrauch der einzelnen Verbraucher einzusehen, wurde für diese noch einmal spezielle Visualisierungen erstellt. Mit einen Mausklick auf die Visualisierungselemente der Verbraucher im Online-Modus gelangt man zu den Ergebnissen der Energetischen Rechnungen der einzelnen Verbraucher.

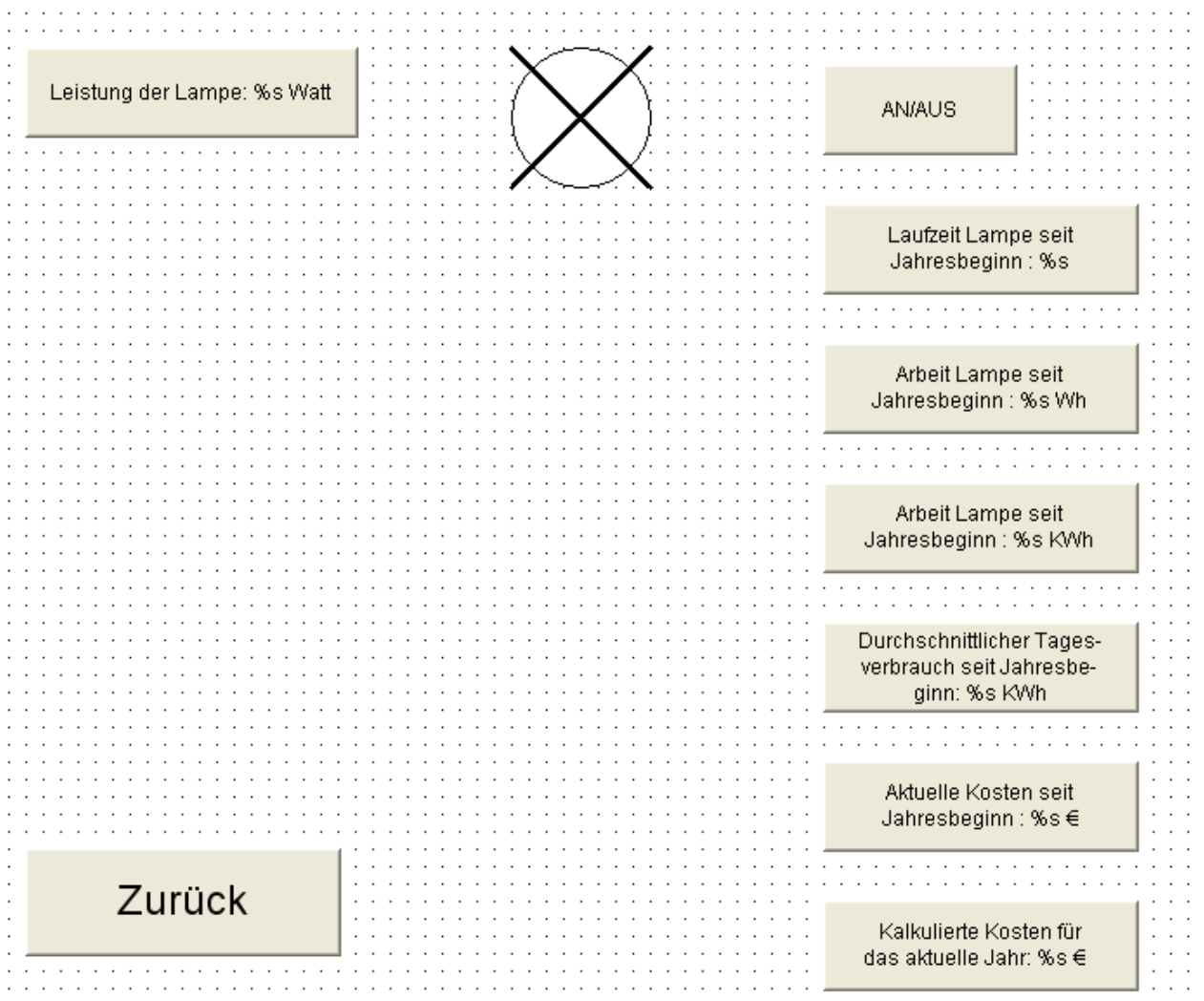


Abbildung 19.2: Visualisierung Lampe Kinderzimmer

In diesem Fenster sind nun die Laufzeiten sowie der aktuelle Energieverbrauch der seit Jahresbeginn und die Kosten der Lampe im Kinderzimmer aufgeführt. Da für die meisten Verbraucher keine Messklemmen zur Verfügungen standen oder die Leistungen aufgrund der kleinen Strom und Spannungswerte kaum messbar waren, muss eine fiktive Leistung auf der Schaltfläche links oben eingetragen werden. Die Stromtarife werden ebenfalls über eine Schaltfläche bzw. eine Scrollleiste in der Visualisierung eingetragen.

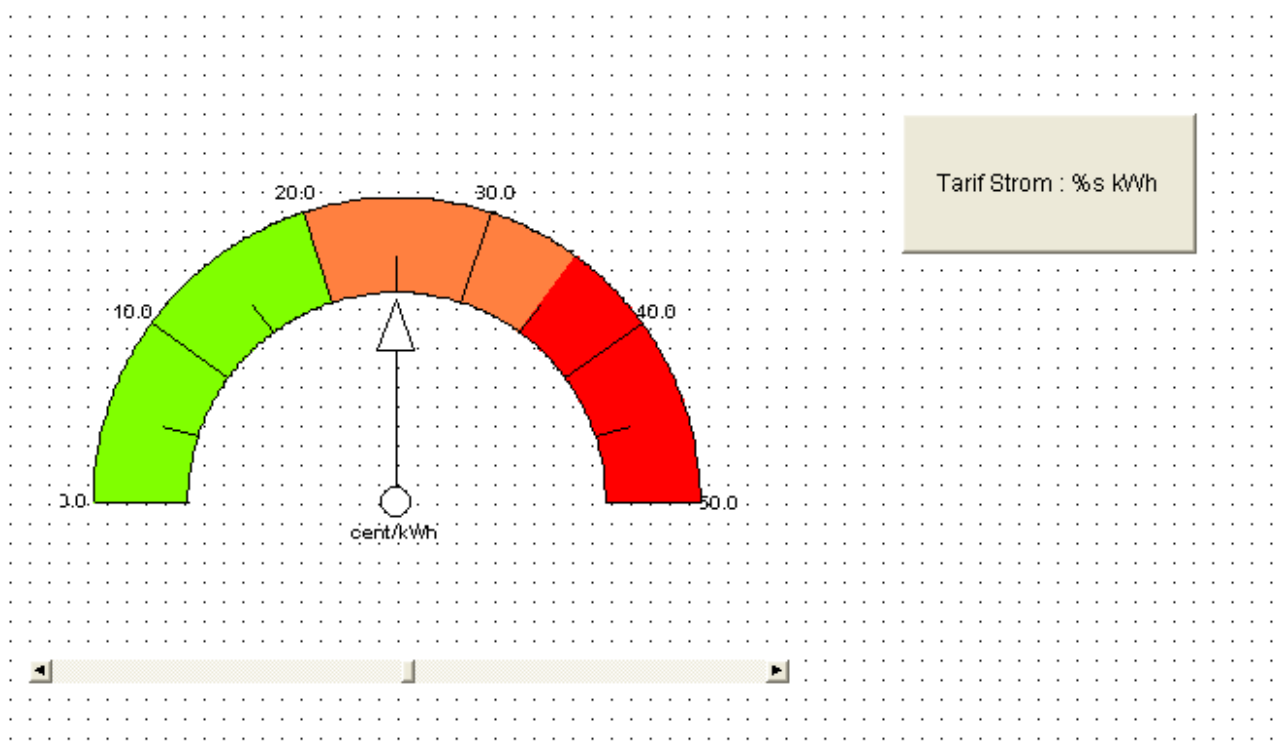


Abbildung 19.3: Visualisierung Tarif Strom

11 Fazit

Im Bereich Gebäudeautomation bietet die Wago SPS gegenüber anderen Gebäudebussystemen einige Vorteile. Zum einen ist sie um einiges preiswerter als die meisten Systeme und zum anderen lässt sie sich problemlos in andere Gebäudebussysteme integrieren. Ein weiterer wichtiger Vorteil ist, dass der Programmierer im Vergleich zur LCN oder KNX wesentlich mehr Freiheiten hat, um seine Funktionen zu realisieren.

Diese Arbeit hat jedoch gezeigt, dass bestimmte Funktionen recht umfangreich zu programmieren waren. Hier lag der Vorteil der IEC-61131 darin, dass insgesamt sechs verschiedene Programmiersprachen zur Verfügung standen. All diese Sprachen haben ihre eigene Vor- und Nachteile, daher sollte man sich beim Programmieren nicht nur auf eine einzige der Sprachen fokussieren. Auch einige Funktionen der Wago Bibliotheken, welche eigentlich ein komfortableres Programmieren des Systems gewährleisten sollten, waren eher mittelmäßige Hilfen für die Realisierung bestimmter Funktionen.

Für die Einbindung von einem Smart Metering in ein Gebäudebussystem, ist es allerdings wesentlich effizienter und auch komfortabler, wenn ein spezielles Modul vorhanden ist, welches dem System die Betriebsdauer sowie die energetischen Messungen und Berechnung der einzelnen Verbraucher abnimmt. Hier gerät selbst ein leistungsfähiges und flexibles Bussystem wie die Wago SPS an ihre Grenzen.

12 Zusammenfassung/Abstract

Aufgrund der fortwährend steigenden Energiepreise sowie der begrenzten Kapazität fossiler Energieträger rückt das Thema Energiemanagement immer mehr ins allgemeine Interesse. Diese Arbeit beschäftigt sich mit dem Thema Smart Metering, welches in ein auf der IEC-61131-3 basierendes Gebäudebussystem der Firma Wago eingebunden wurde. Hierbei wurde zunächst ein Puppenhaus aufgebaut durch welches später ein tatsächlicher Energiehaushalt simuliert werden sollte. Die Gebäudetechnischen Funktionen wurden mit der CoDeSys Software der Firma Smart Software Solutions gemäß der IEC-61131-3 programmiert und auf eine Speicherprogrammierbare Steuerung der Firma Wago aufgespielt. Zur Ermittlung des Energieverbrauchs wurden für einige Verbraucher spezielle Leistungsmessklemmen verwendet. Zudem wurden spezielle Zählerbausteine programmiert, welche die Betriebsdauer aller Geräte innerhalb des Puppenhauses angeben. Die speziellen energetischen Berechnungen, wie die aktuelle verbrauchte Arbeit seit Jahresbeginn, sowie die kalkulierte Arbeit bis zum Jahresende wurden ebenfalls über spezielle Programme in der CoDeSys realisiert. Um den Energieverbrauch sowie die Kosten für den Benutzer transparent zu machen, wurde eine zusätzliche Visualisierung ebenfalls mit der CoDeSys Software erstellt.

On the basis of the continually rising energy prices and the limited capacity of fossil fuels, the topic of energy management becomes more and more important for general interests. This thesis describes smart metering, which was involved into a Wago Programmable Logic Controller based on the IEC 61131 standard. For the simulation of a realistic energy consumption a dummyhouse was built at first. The Building engineering functions were programmed by the software package CoDeSys, which was developed by the smart Software Solution Enterprises according to the IEC 61131 programming languages. After that the created application was transferred to the Wago Programmable Logic Controller. To determine the energy consumption of several electrical consumers a power measurement modul was used. Further some special counting applications were written to determine the runtime of every device inside the dummyhouse. The energy consumption from the beginning of the year and the calculated energy consumption until the end of the year were also calculated by several CoDeSys applications. At last a special visualization tool - which is also a part of the CoDeSys Software package was used to create a visualization to show the user the results of the calculations of the energy consumption and the energy bill.

13 Abbildungsverzeichnis

Abbildung 1: Feldbusknoten.....	12
Abbildung 2.1: Primär getaktete Spannungsversorgung 757-622.....	13
Abbildung 2.2: Primär getaktet Spannungsversorgung (Montage).....	13
Abbildung 2.2: Primär getaktet Spannungsversorgung (Kontaktplan).....	13
Abbildung 3.1: KNX-IP Controller (750-849).....	14
Abbildung 3.2: KNX-IP Controller (Schematischer Aufbau).....	14
Abbildung 4.1: 8-Kanal Digital Ausgangsklemme (750-530).....	15
Abbildung 4.2: 8-Kanal Digital Ausgangsklemme (Schematischer Aufbau).....	15
Abbildung 5.1: 8-Kanal Digital Eingangsklemme (750-430).....	16
Abbildung 5.2: 8-Kanal Digital Eingangsklemme.....	16
Abbildung 6.1: 2-Kanal Relaisausgangsklemme (750-517).....	17
Abbildung 6.2: 2-Kanal Relaisausgangsklemme (Schematischer Aufbau).....	17
Abbildung 6.3: 2-Kanal Relaisausgangsklemme (Kontaktplan).....	17
Abbildung 7.1: 2-Kanal Analog Ausgangsklemme (750-552).....	18
Abbildung 7.2: 2-Kanal Analog Ausgangsklemme (Schematischer Aufbau).....	18
Abbildung 7.3: 2-Kanal Analog Ausgangsklemme (Kontaktplan).....	18
Abbildung 7.4: 2-Kanal Analog Ausgangsklemme (Prozessabbild).....	19
Abbildung 8.1: 2-Kanal Analog Eingangsklemme für Thermoelemente (750-469).....	20
Abbildung 8.2: 2-Kanal Analog Eingangsklemme für Thermoelemente (Schematischer Aufbau).....	20
Abbildung 8.3: 2-Kanal Analog Eingangsklemme für Thermoelemente (Prozessabbild).....	21
Abbildung 9.1: 2 DI DC 24 V, Einbruchsmeldung (750-424).....	22
Abbildung 9.2: 2 DI DC 24 V, Einbruchsmeldung (Schematischer Aufbau).....	22
Abbildung 9.3: 2 DI DC 24 V, Einbruchsmeldung (Alarmkontakt).....	22
Abbildung 9.4: 2 DI DC 24 V, Einbruchsmeldung (Kontaktplan).....	22
Abbildung 10.1: Bus-Endklemme (750-600).....	23
Abbildung 10.2: Bus-Endklemme (Schematischer Aufbau).....	23

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Abbildung 11.1: Ethernet Settings.....	24
Abbildung 11.2: Ethernet Settings (Com Port Einstellungen).....	25
Abbildung 11.3: Ethernet Settings (Verbindungsdaten).....	25
Abbildung 11.4: Ethernet Settings (TCP/IP Einstellungen).....	26
Abbildung 12.1: Codesys (Zielsystem auswählen).....	27
Abbildung 12.2: CoDeSys (Zielsystem Einstellungen).....	28
Abbildung 12.3: CoDeSys (Programm erstellen).....	29
Abbildung 12.4: CoDeSys (Anweisungsliste).....	30
Abbildung 12.5: CoDeSys (Kontaktplan).....	31
Abbildung 12.6: CoDeSys (Funktionsplan).....	32
Abbildung 12.7: CoDeSys (Ablaufsprache).....	33
Abbildung 12.8: CoDeSys (Strukturierter Text).....	34
Abbildung 12.9: CoDeSys (Freigrafischer Funktionsplan Editor).....	35
Abbildung 12.10: CoDeSys (Funktion erstellen).....	36
Abbildung 12.11: CoDeSys (Funktion Variablen).....	36
Abbildung 12.12: CoDeSys (Funktion Quelltext).....	37
Abbildung 12.13: CoDeSys (Funktionsaufruf in ST).....	37
Abbildung 12.14: CoDeSys (Funktionsaufruf CFC).....	37
Abbildung 12.15: CoDeSys (Funktionsaufruf AWL).....	37
Abbildung 12.16: CoDeSys (Funktionsblock Büro).....	38
Abbildung 12.17: CoDeSys (Funktionsblock erstellen).....	39
Abbildung 12.18: CoDeSys (Funktionsblock Variablen deklarieren).....	40
Abbildung 12.19: CoDeSys (Funktionsblock Quelltext).....	40

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Abbildung 12.20: CoDeSys (Funktionsblock Büro erstellen).....	41
Abbildung 12.21: CoDeSys (Funktionsblock Büro Variablen).....	41
Abbildung 12.22: CoDeSys (Funktionsblock Funktionsbaustein einfügen).....	42
Abbildung 12.23: CoDeSys (Funktionsblock Eingabehilfe).....	43
Abbildung 12.24: CoDeSys (Funktionsblock als CFC Baustein).....	44
Abbildung 12.25: CoDeSys (Funktionsblock Ein-und Ausgänge).....	45
Abbildung 12.26: CoDeSys (Bibliotheksverwalter auswählen).....	46
Abbildung 12.27: CoDeSys (Bibliotheksverwalter Bibliothek einfügen).....	47
Abbildung 12.28: CoDeSys (Bibliotheksverwalter Bibliothek auswählen).....	48
Abbildung 12.29: CoDeSys (Bibliotheken Vergleich).....	49
Abbildung 12.30: CoDeSys (Visualisierungseditor aufrufen).....	50
Abbildung 12.30: CoDeSys (Visualisierungsobjekt einfügen).....	51
Abbildung 12.31: CoDeSys (Visualisierungsobjekt benennen).....	51
Abbildung 12.32: CoDeSys (Visualisierung Oberfläche).....	52
Abbildung 12.33: CoDeSys (Visualisierung Schaltfläche einfügen).....	54
Abbildung 12.34: CoDeSys (Visualisierung Schaltfläche).....	54
Abbildung 12.35: CoDeSys (Visualisierung Schaltfläche Konfigurieren).....	55
Abbildung 12.36: CoDeSys (Visualisierung Schaltfläche Text).....	55
Abbildung 12.37: CoDeSys (Visualisierung Schaltfläche Eingabe).....	56
Abbildung 12.38: CoDeSys (Visualisierung Ellipse einfügen).....	57
Abbildung 12.39: CoDeSys (Visualisierung Ellipse).....	57
Abbildung 12.40: CoDeSys (Visualisierung Ellipse konfigurieren).....	57

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Abbildung 12.41: CoDeSys (Visualisierung Ellipse Farbvariablen).....	59
Abbildung 12.42: CoDeSys (Visualisierung Farbvariablen).....	59
Abbildung 12.43: CoDeSys (Visualisierung Variablen).....	60
Abbildung 12.44: CoDeSys (Visualisierung Lampe aus).....	61
Abbildung 12.45: CoDeSys (Visualisierung Lampe ein).....	61
Abbildung 12.46: CoDeSys (Visualisierung Scrollleiste einfügen).....	62
Abbildung 12.47: CoDeSys (Visualisierung Scrollleiste konfigurieren).....	62
Abbildung 12.48: CoDeSys (Visualisierung Scrollleiste Variablen).....	62
Abbildung 12.49: CoDeSys (Visualisierung Zeigerinstrument einfügen).....	64
Abbildung 12.50: CoDeSys (Visualisierung Zeigerinstrument konfigurieren).....	64
Abbildung 12.51: CoDeSys (Visualisierung Zeigerinstrument Anzeigeskala Variable konfigurieren).....	65
Abbildung 12.52: CoDeSys (Visualisierung Zeigerinstrument Wert=0).....	66
Abbildung 12.53: CoDeSys (Visualisierung Zeigerinstrument Wert=100).....	66
Abbildung 12.54: CoDeSys (Visualisierung Zeigerinstrument Wert=200).....	67
Abbildung 12.55: CoDeSys (Visualisierung Eingabefeld Text).....	68
Abbildung 12.56: CoDeSys (Visualisierung Eingabefeld Variablen).....	69
Abbildung 12.57: CoDeSys (Visualisierung Eingabefeld Wert=123).....	70
Abbildung 12.58: CoDeSys (Visualisierung Eingabefeld Eingabe).....	71
Abbildung 12.59: CoDeSys (Visualisierung Eingabefeld Werteingabe).....	72
Abbildung 12.60: CoDeSys (Visualisierung Eingabefeld Werteingabe).....	73
Abbildung 13.1: Puppenhaus.....	74
Abbildung 13.2: Puppenhaus Taster.....	75

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Abbildung 13.3: Puppenhaus 24 V Lampe.....	75
Abbildung 13.3: Puppenhaus 230 V Lampe.....	76
Abbildung 13.4: Puppenhaus Kontakttaster.....	77
Abbildung 13.5: Puppenhaus Jalousiemotor.....	78
Abbildung 13.6: Puppenhaus Rückseite Klemmenblöcke.....	79
Abbildung 13.7: Puppenhaus Klemmenblock 1.....	80
Abbildung 13.7: Puppenhaus Klemmenblock 2.....	81
Abbildung 13.8: Puppenhaus Klemmenblock 3.....	82
Abbildung 13.9: Puppenhaus Centronics Stecker.....	83
Abbildung 13.10: Puppenhaus Centronics Buchse.....	83
Abbildung 13.11: Puppenhaus Centronics Stecker Rückseite.....	83
Abbildung 13.12: Puppenhaus Erdgeschoss	85
Abbildung 13.13: Puppenhaus Kinderzimmer.....	86
Abbildung 13.14: Puppenhaus Schlafzimmer.....	87
Abbildung 13.15: Puppenhaus Küche.....	88
Abbildung 13.16: Puppenhaus Badezimmer.....	89
Abbildung 13.17: Puppenhaus Wohnzimmer.....	90
Abbildung 14.1: Funktionsblock umschalten Variablen.....	93
Abbildung 14.2: Funktionsblock umschalten Quellcode.....	94
Abbildung 14.3: Funktionsblock umschalten.....	94
Abbildung 14.4: Funktionsblock Fb_KurzLang.....	95
Abbildung 14.5: Funktionsblock Fb_Klick.....	95

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Abbildung 14.6: Funktionsblock HeizungFB Variablen.....	96
Abbildung 14.7: Funktionsblock HeizungFB Quelltext	96
Abbildung 14.8: Funktionsblock HeizungFB	97
Abbildung 15.1: Lichtorgel umschalten.....	98
Abbildung 15.2: Lichtorgel Quelltext.....	99
Abbildung 16.1: Enocean Funkreceiver (750-642).....	100
Abbildung 16.2: Enocean Funkreceiver (Schematischer Aufbau).....	100
Abbildung 16.3: Enocean FBReceive Baustein.....	102
Abbildung 16.4: Enocean 4-fach Taster.....	102
Abbildung 16.5: Enocean FbButton_4_Channel.....	102
Abbildung 16.6: Thermokon Sensor.....	103
Abbildung 16.7: Enocean fbSR04.....	103
Abbildung 17.1: RTC-Modul (750-640).....	104
Abbildung 17.2: RTC-Modul (Schematischer Aufbau).....	104
Abbildung 17.3: RTC-Modul Ein- und Ausgangsdaten.....	105
Abbildung 17.4: RTC-Modul Funktionsübersicht Teil 1.....	106
Abbildung 17.5: RTC-Modul Funktionsübersicht Teil 2.....	107
Abbildung 17.6: RTC-Modul Funktionsaufruf.....	108
Abbildung 17.7: RTC-Modul Funktionsantwort.....	108
Abbildung 17.8: RTC-Modul Funktionsaufruf.....	109
Abbildung 17.9: RTC-Modul Funktionsantwort.....	109
Abbildung 18.1: 3_Phasen-Leistungsmessklemme (750-493).....	114

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Abbildung 18.2: 3-Phasen-Leistungsmessklemme (Schematischer Aufbau).....	114
Abbildung 18.3: 3-Phasen-Leistungsmessklemme Ein- und Ausgangsdaten.....	119
Abbildung 18.4: 3-Phasen-Leistungsmessklemme Prozesswerte Spannung.....	120
Abbildung 18.5: 3-Phasen-Leistungsmessklemme Prozesswerte Strom.....	120
Abbildung 18.6: 3-Phasen-Leistungsmessklemme Prozesswerte Wirkleistung.....	121
Abbildung 18.7: Funktionsbaustein Fb750_493_Master3Phase.....	121
Abbildung 18.8: Funktionsblock Zählen.....	122
Abbildung 18.9: Funktionsblock Energetische Rechnung Variablen.....	124
Abbildung 18.10: Funktionsblock Energetische Rechnung Quelltext.....	124
Abbildung 18.11: Funktionsblock Day_of_Year.....	125
Abbildung 18.12: Funktionsblock Days_in_Year.....	125
Abbildung 19.1: Visualisierung Kinderzimmer.....	126
Abbildung 19.2: Visualisierung Lampe Kinderzimmer.....	126
Abbildung 19.3: Visualisierung Tarif Strom.....	128
Tabelle 5: Belegung Klemmenblock 1.....	80
Tabelle 6: Belegung Klemmenblock 2.....	81
Tabelle 7: Belegung Klemmenblock 3.....	82

14 Literaturverzeichnis

Aschendorf, P.D. (2010) Vorlesungsscript Elektrische Gebäudesystemtechnik

Gleißmann, C. (15.02.2008). Diplomarbeit. Vergleich der Implementierung der Sensorik und Aktorik des KNX/EIB-Gebäudebussystems in IEC 61131-basierende Gebäudeautomatisierungssysteme von Wago und Beckhoff

Klötter, M. (20.07.2009). Bachelor-Thesis. Homogene und heterogene Interaktion von Speicherprogrammierbaren Steuerungen im Bereich der Gebäudeautomation

Internetseite der WAGO Kontaktechnik GmbH & Co.KG, <http://www.wago.com/>, Stand: April 2010

Internetseite EnOcean GmbH, <http://www.enocean.com/de/>, Stand: April 2010

Internetseite <http://www.oscat.de>, Stand: April 2010

Internetseite Firma Smart Software Solutions <http://www.3s-software.com>, Stand April: 2010

15 Anhang: Quelltexte

15.1 Funktionen

Rotieren:

Variablen:

0001	FUNCTION Rotieren : DWORD
0002	VAR_INPUT
0003	A: DWORD;
0004	B: DWORD;
0005	C: DWORD;
0006	D: DWORD;
0007	END_VAR
0008	VAR
0009	TEMP: DWORD;
0010	END_VAR

Quelltext:

0001	LD	A
0002	ROL	24
0003	ST	TEMP
0004	LD	B
0005	ROL	16
0006	OR	TEMP
0007	ST	TEMP
0008	LD	C
0009	ROL	8
0010	OR	TEMP
0011	ST	TEMP
0012	LD	D
0013	OR	TEMP
0014	ST	Rotieren

RTC-Modul:

Variablen:

0001	FUNCTION RTC_Modul:DWORD
0002	VAR_INPUT
0003	E0: BYTE;
0004	E1: BYTE;
0005	E2: BYTE;
0006	E3: BYTE;
0007	ID: BYTE;
0008	END_VAR
0009	VAR
0010	D0: DWORD;
0011	D3: DWORD;
0012	D2: DWORD;
0013	D1: DWORD;
0014	END_VAR

Quelltext:

0001	RTC_Ausgang_D3:=E3;
0002	RTC_Ausgang_D2:=E2;
0003	RTC_Ausgang_D1:=E1;
0004	RTC_Ausgang_D0:=E0;
0005	RTC_Ausgang_ID:=ID;
0006	RTC_Ausgang_C:=NOT RTC_Ausgang_C;
0007	
0008	D0:=BYTE_TO_DWORD(RTC_Eingang_D0);
0009	D1:=BYTE_TO_DWORD(RTC_Eingang_D1);
0010	D2:=BYTE_TO_DWORD(RTC_Eingang_D2);
0011	D3:=BYTE_TO_DWORD(RTC_Eingang_D3);
0012	
0013	RTC_Modul:=Rotieren(D3, D2, D1, D0);
0014	

13. 2 Funktionsblöcke

Energetische Rechnung:

Variablen:

0001	FUNCTION_BLOCK Energetische_Rechnung
0002	VAR_INPUT
0003	Aktor:BOOL;
0004	Leistung_Aktor:INT;
0005	Kosten:REAL;
0006	END_VAR
0007	VAR_OUTPUT
0008	Laufzeit_Sek: DWORD;
0009	Laufzeit_Time: TIME;
0010	akt_Arbeit_Wh: REAL;
0011	akt_Arbeit_KWh: REAL;
0012	durchschn_Arbeit: REAL;
0013	akt_Kosten: REAL;
0014	kalk_Kosten: REAL;
0015	
0016	END_VAR
0017	VAR
0018	Zaehler: Zaehlen;
0019	akt_Date: DATE;
0020	Tage: INT;
0021	ges_Tage: INT;
0022	END_VAR

Quelltext:

0001	Zaehler(EN:=Aktor , Leistung:=INT_TO_DWORD(Leistung_Aktor));
0002	Laufzeit_Sek:=Zaehler.Zeit;
0003	akt_Arbeit_Wh:=DWORD_TO_REAL(Zaehler.Arbeit)/3600;
0004	Laufzeit_Time:=DWORD_TO_TIME(Laufzeit_Sek);
0005	akt_Arbeit_KWh:=akt_Arbeit_Wh/1000;
0006	akt_Kosten:=Kosten*akt_Arbeit_KWh;
0007	durchschn_Arbeit:=akt_Arbeit_KWh/INT_TO_REAL(DAY_OF_YEAR(akt_Date));
0008	kalk_Kosten:=durchschn_Arbeit*INT_TO_REAL(DAYS_IN_YEAR(akt_Date))*Kosten;

HeizungFB:

Variablen:

```
0001 FUNCTION_BLOCK HeizungFB
0002 VAR_INPUT
0003     Frostschutz_Temperatur:REAL;
0004     Temperatur_IST:REAL;
0005     Temperatur_SOLL:REAL;
0006     xZenAus:BOOL;
0007 END_VAR
0008 VAR_OUTPUT
0009     EIN: BOOL;
0010 END_VAR
0011 VAR
0012 END_VAR
```

Quelltext:

```
0001 IF Temperatur_IST>Frostschutz_Temperatur THEN
0002     IF xZenAUS=FALSE THEN
0003         IF Temperatur_SOLL-Temperatur_IST>0.5 THEN
0004             EIN:=TRUE;
0005         ELSE
0006             EIN:=FALSE;
0007         END_IF
0008     ELSE
0009         EIN:=FALSE;
0010     END_IF
0011 ELSE
0012     EIN:=TRUE;
0013 END_IF
```

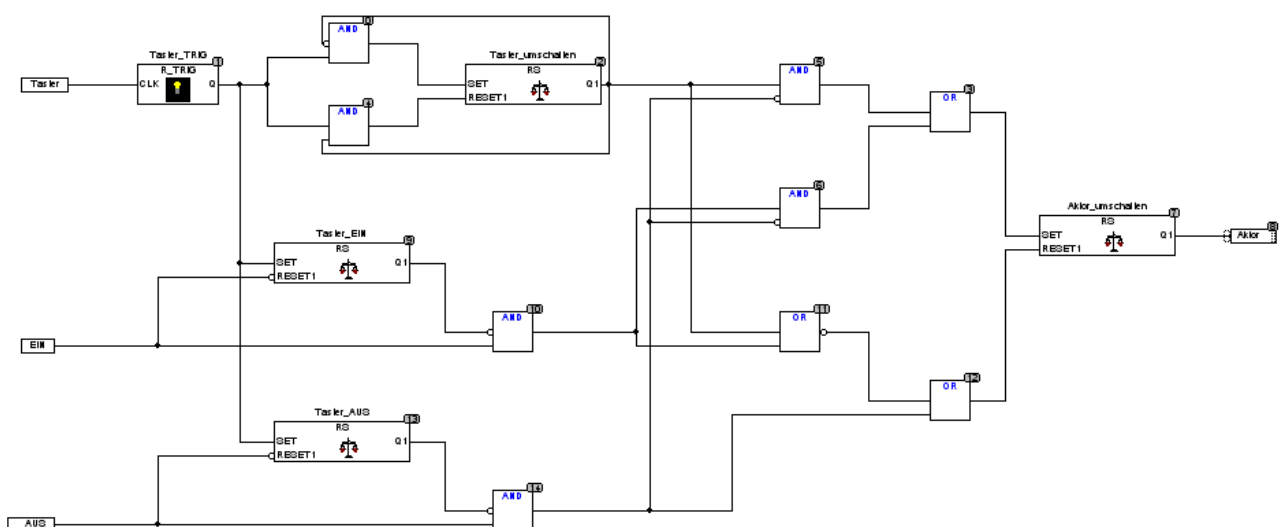
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Umschalten:

Variablen:

```
0001 FUNCTION_BLOCK umschalten
0002 VAR_INPUT
0003     Taster:BOOL;
0004     EIN:BOOL;
0005     AUS:BOOL;
0006 END_VAR
0007 VAR_OUTPUT
0008     Aktor:BOOL;
0009 END_VAR
0010 VAR
0011     Taster_TRIG: R_TRIG;
0012     Aktor_umschalten: RS;
0013     Taster_EIN: RS;
0014     Taster_AUS: RS;
0015     Taster_umschalten: RS;
0016 END_VAR
```

Quelltext:

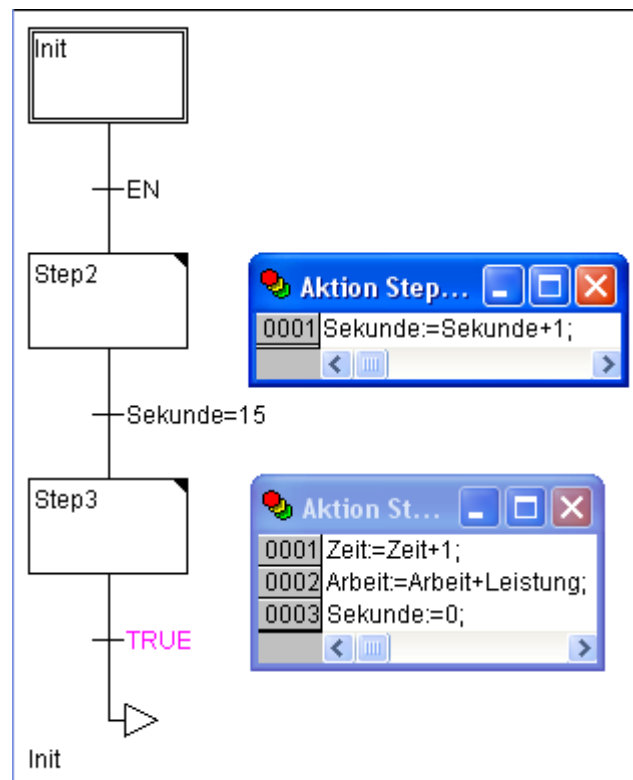


Zählen:

Variablen:

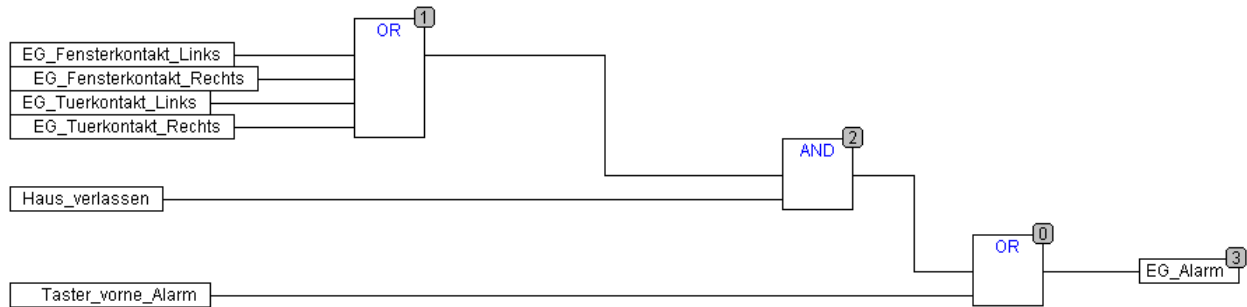
```
0001 FUNCTION_BLOCK Zaehlen
0002 VAR_INPUT
0003     EN:BOOL;
0004     Leistung:DWORD;
0005 END_VAR
0006 VAR_OUTPUT
0007     Zeit:DWORD;
0008     Arbeit:DWORD;
0009 END_VAR
0010 VAR
0011     Sekunde:INT;
0012 END_VAR
```

Quelltext:



13.3 Programme:

Alarm:



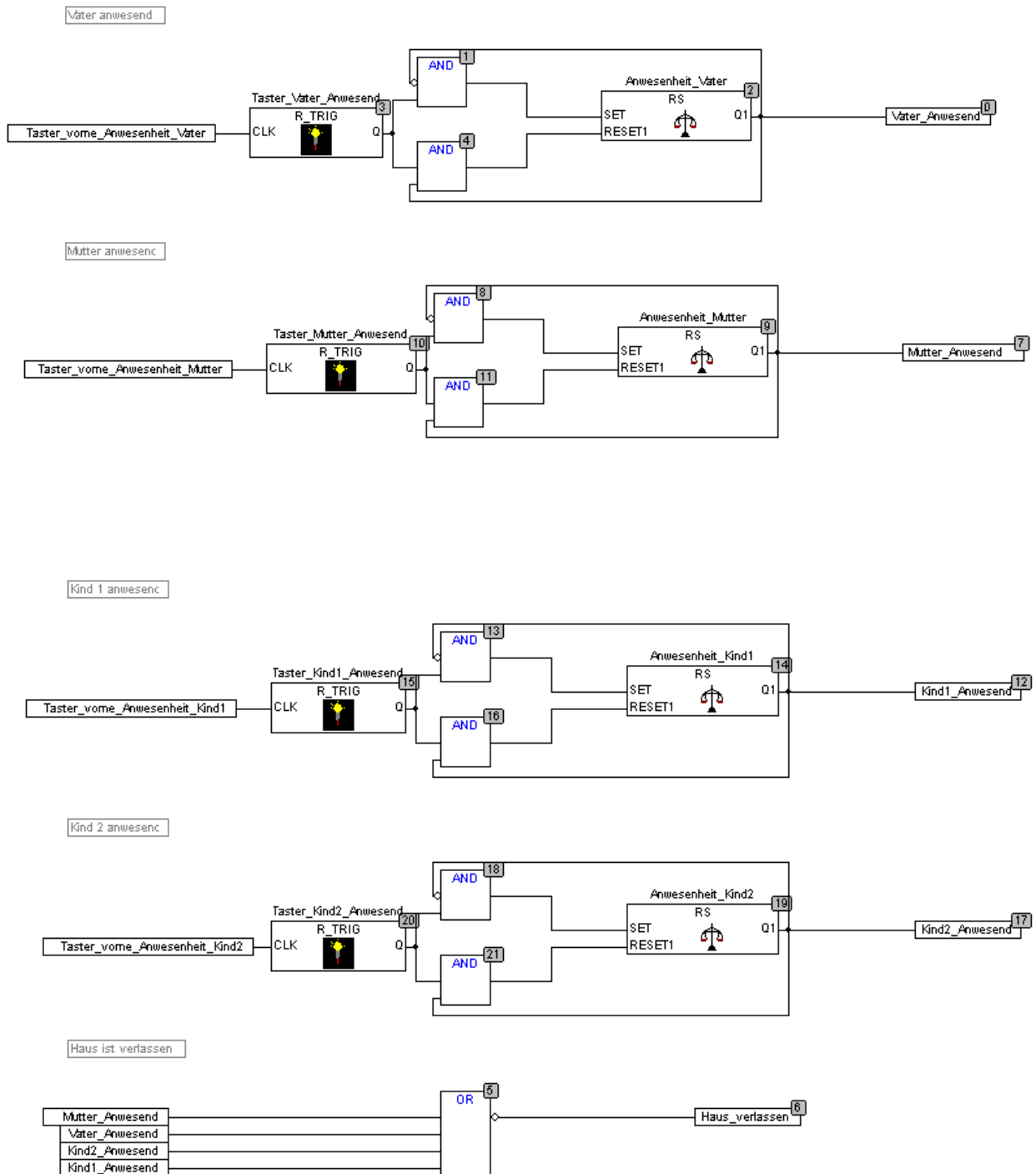
Anwesenheit:

Variablen:

0001	PROGRAM Anwesenheit
0002	VAR
0003	Anwesenheit_Vater: RS;
0004	Taster_Vater_Anwesend: R_TRIG;
0005	Anwesenheit_Mutter: RS;
0006	Taster_Mutter_Anwesend: R_TRIG;
0007	Taster_Kind1_Anwesend: R_TRIG;
0008	Anwesenheit_Kind1: RS;
0009	Taster_Kind2_Anwesend: R_TRIG;
0010	Anwesenheit_Kind2: RS;
0011	END_VAR

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:



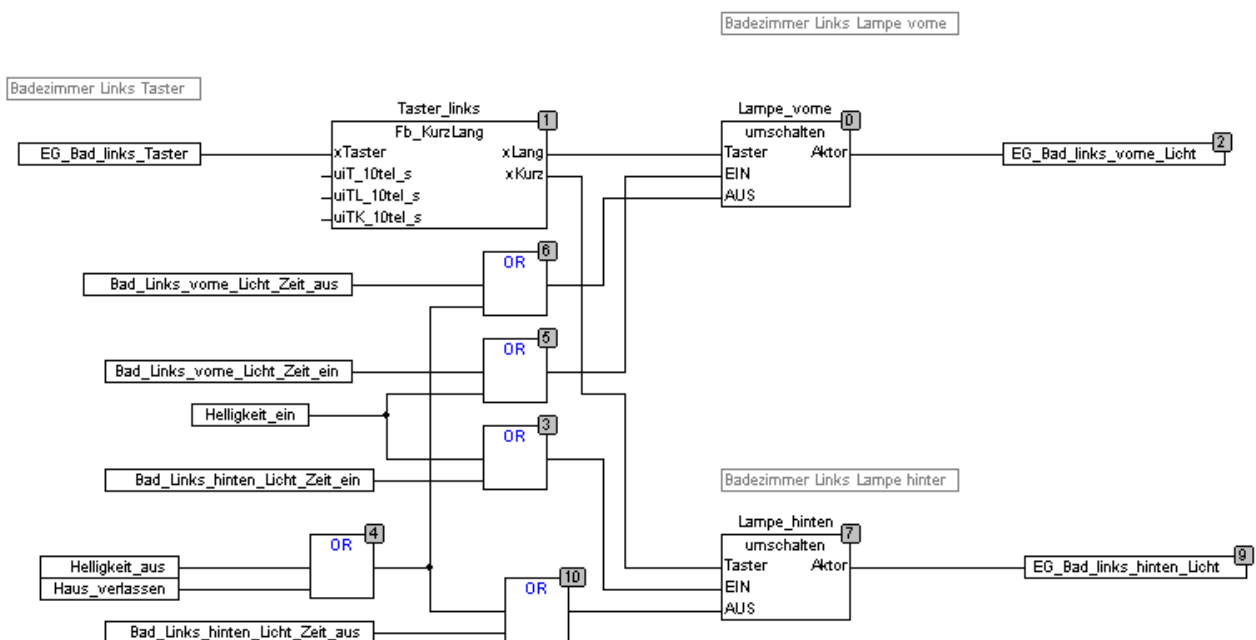
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Badezimmer:

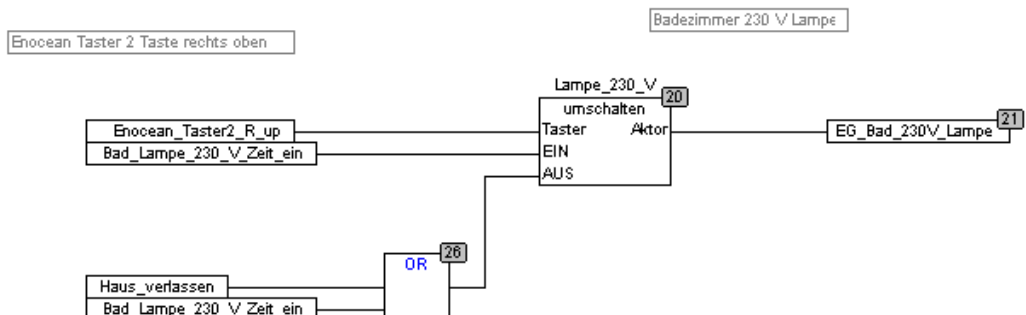
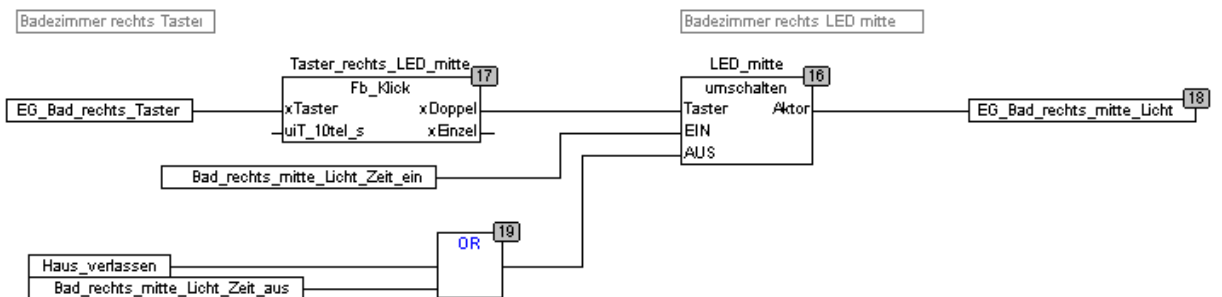
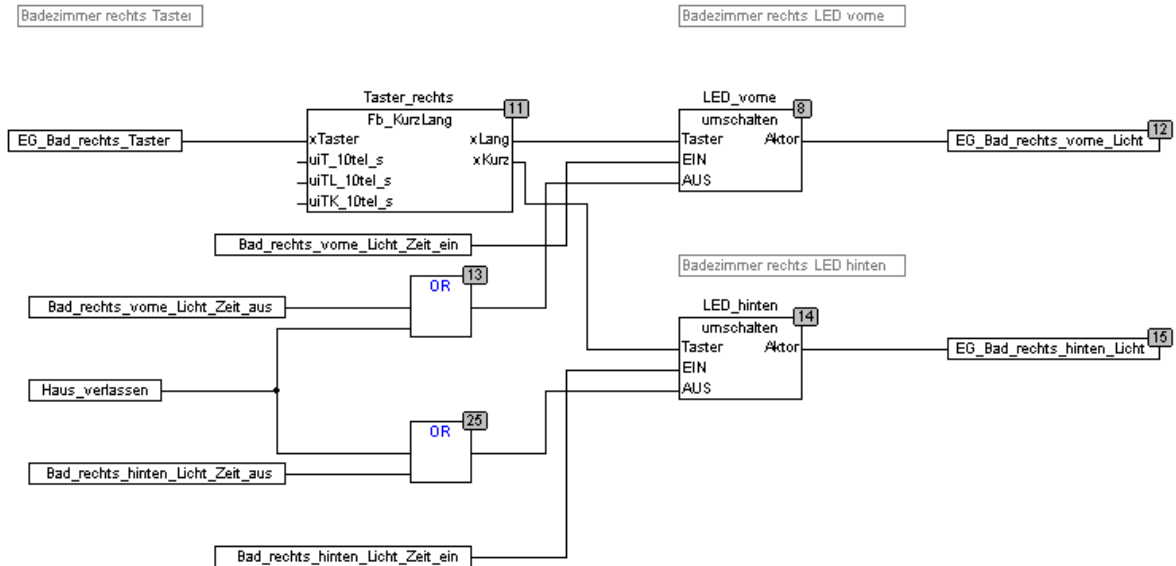
Variablen:

```
0001 PROGRAM Badezimmer
0002 VAR
0003     Lampe_vorne: umschalten;
0004     Lampe_hinten: umschalten;
0005     Taster_links: Fb_KurzLang;
0006     LED_vorne: umschalten;
0007     LED_hinten: umschalten;
0008     Taster_rechts: Fb_KurzLang;
0009     LED_mitte: umschalten;
0010     Taster_rechts_LED_mitte: Fb_Klick;
0011     Lampe_230_V: umschalten;
0012     Heizung: HeizungFB;
0013 END_VAR
```

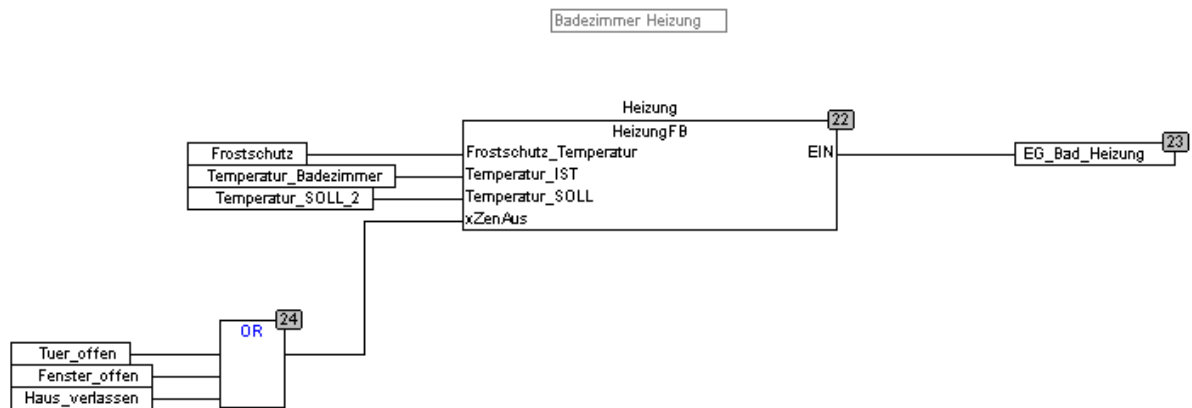
Quelltext:



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Farben:

Variablen:

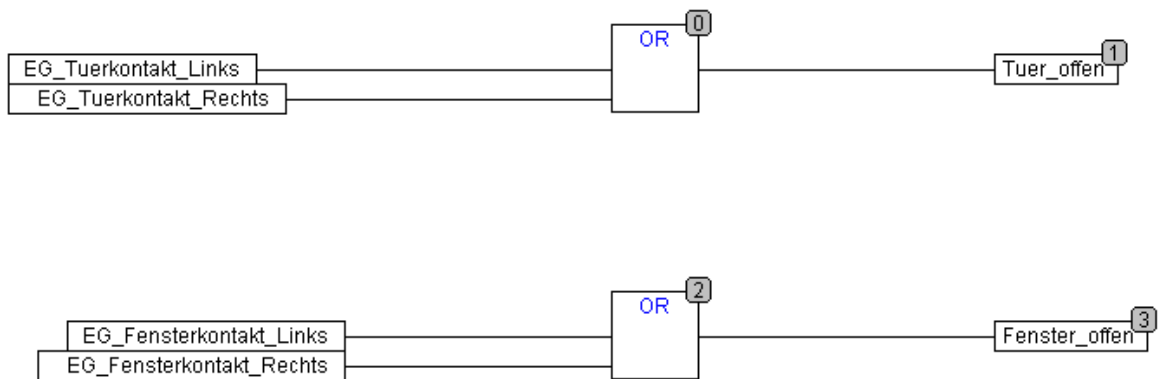
```
0001 PROGRAM Farben_PRG
0002 VAR
0003     var_fillcol: DWORD;
0004     var_fillcol_a_yellow: DWORD;
0005     var_framecol: DWORD;
0006     var_fillcol_a_red: DWORD;
0007     var_fillcol_a_blue: DWORD;
0008 END_VAR
```

Quelltext:

```
0001 var_fillcol_a_yellow:=16#00FFFF;
0002 var_fillcol_a_red:=16#0000FF;
0003 var_fillcol_a_blue:=16#FF0000;
0004 var_fillcol:=16#FFFFFF;
0005 var_framecol:=16#000000;
```

Fenster_Tuer_offen

Quelltext:

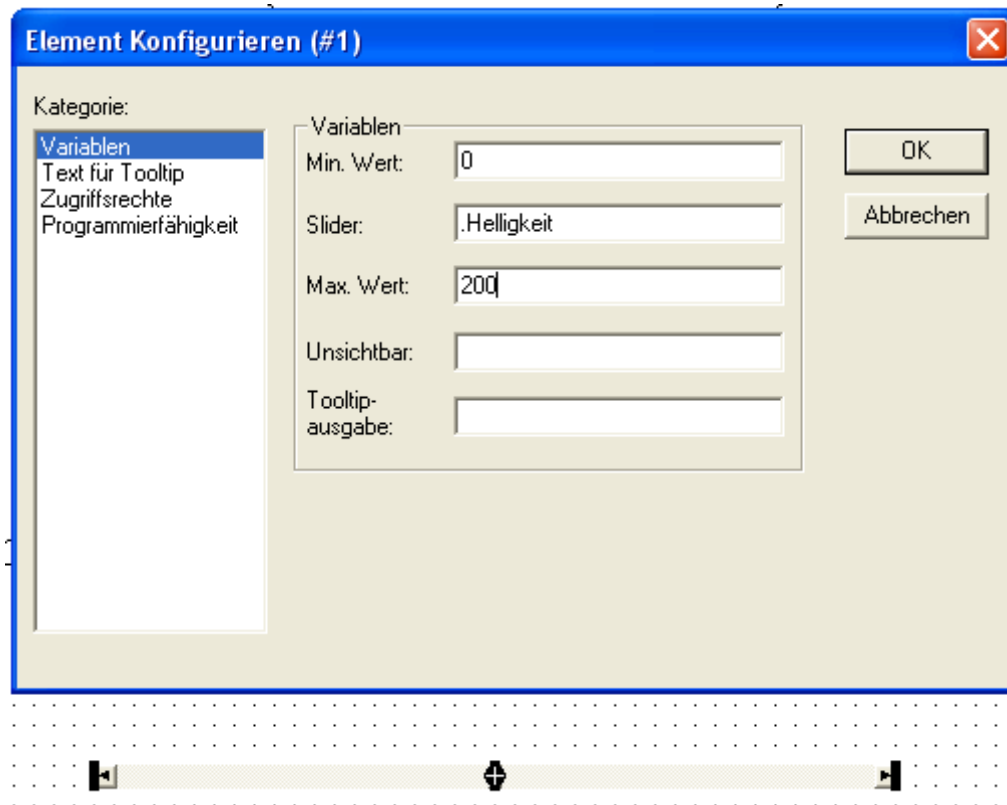


Helligkeit:

Quelltext:

```
0001 IF Helligkeit>160 THEN
0002 Helligkeit_aus:=TRUE;
0003 END_IF
0004 IF Helligkeit<60 THEN
0005 Helligkeit_ein:=TRUE;
0006 END_IF
```

Visualisierung:



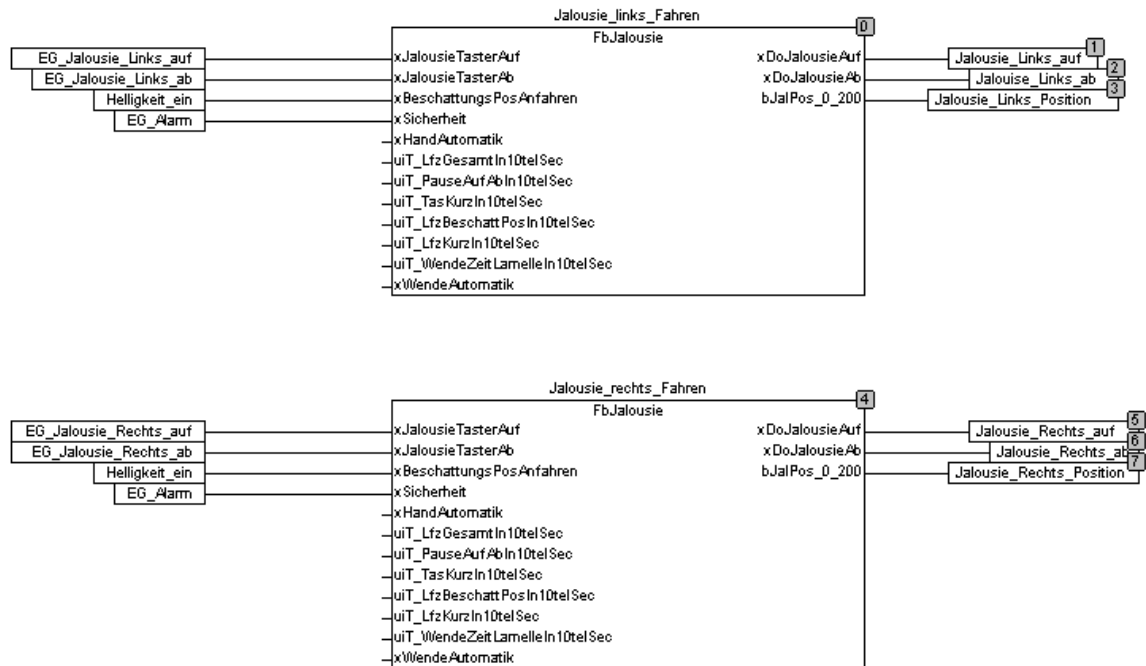
Jalousie:

Variablen:

```
0001 PROGRAM Jalousie
0002 VAR
0003     Jalousie_links_Fahren: FbJalousie;
0004     Jalousie_rechts_Fahren: FbJalousie;
0005 END_VAR
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:



Kinderzimmer:

Variablen:

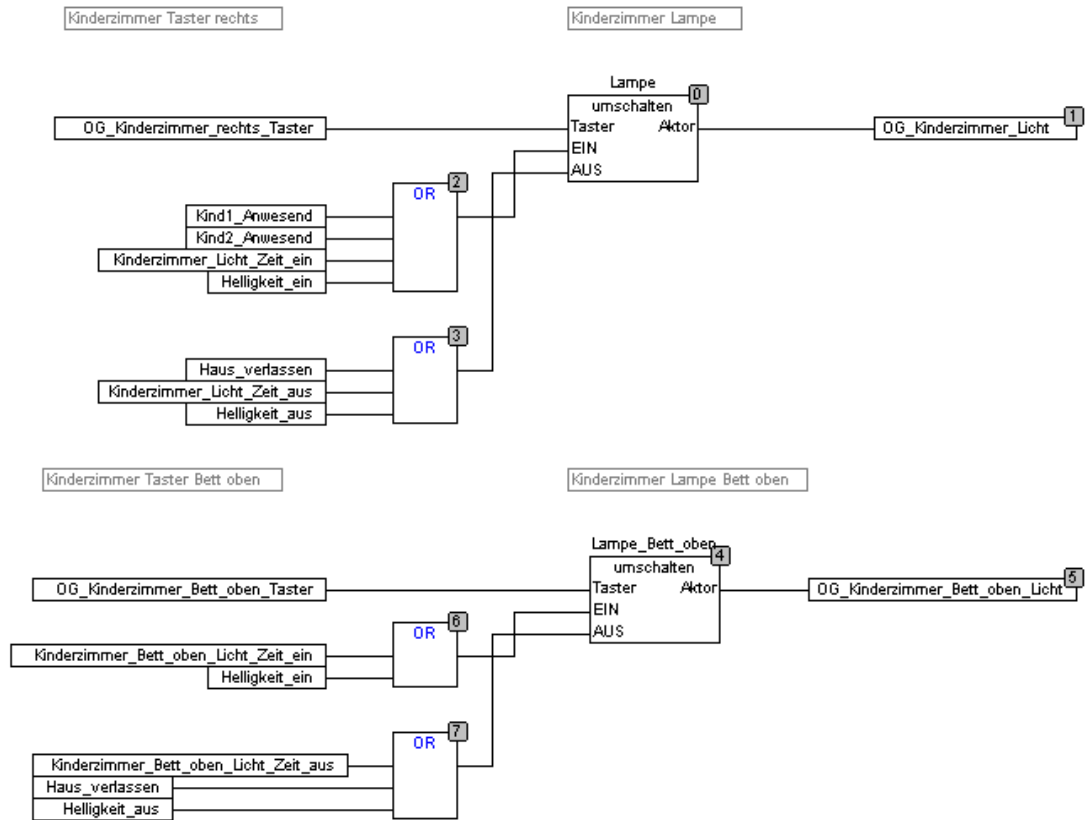
```

0001 PROGRAM Kinderzimmer
0002 VAR
0003     Lampe: umschalten;
0004     Lampe_Bett_oben: umschalten;
0005     Lampe_Bett_unten: umschalten;
0006     Aktive_Steckdose: umschalten;
0007     Lampe_230_V: umschalten;
0008     Heizung: HeizungFB;
0009 END_VAR

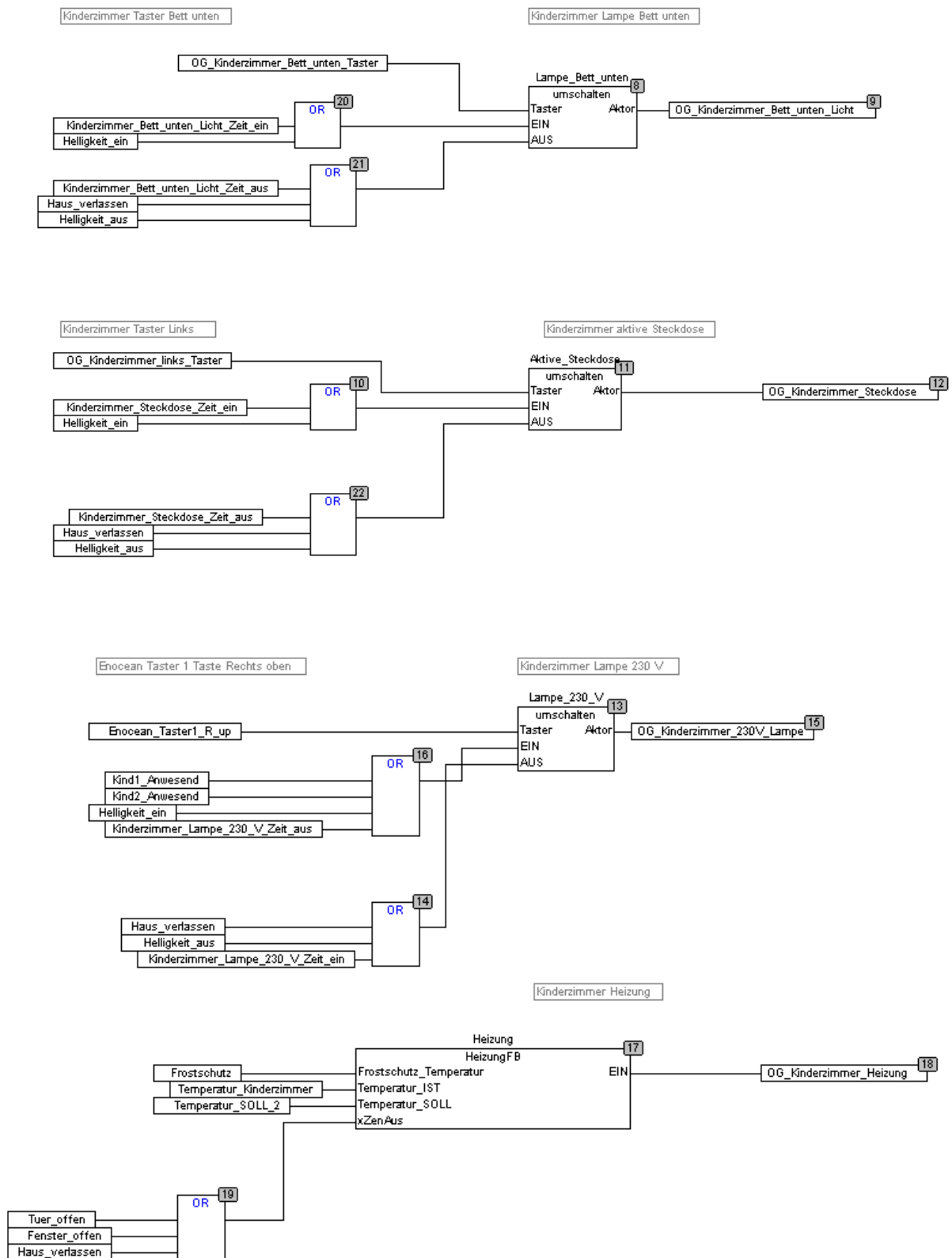
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



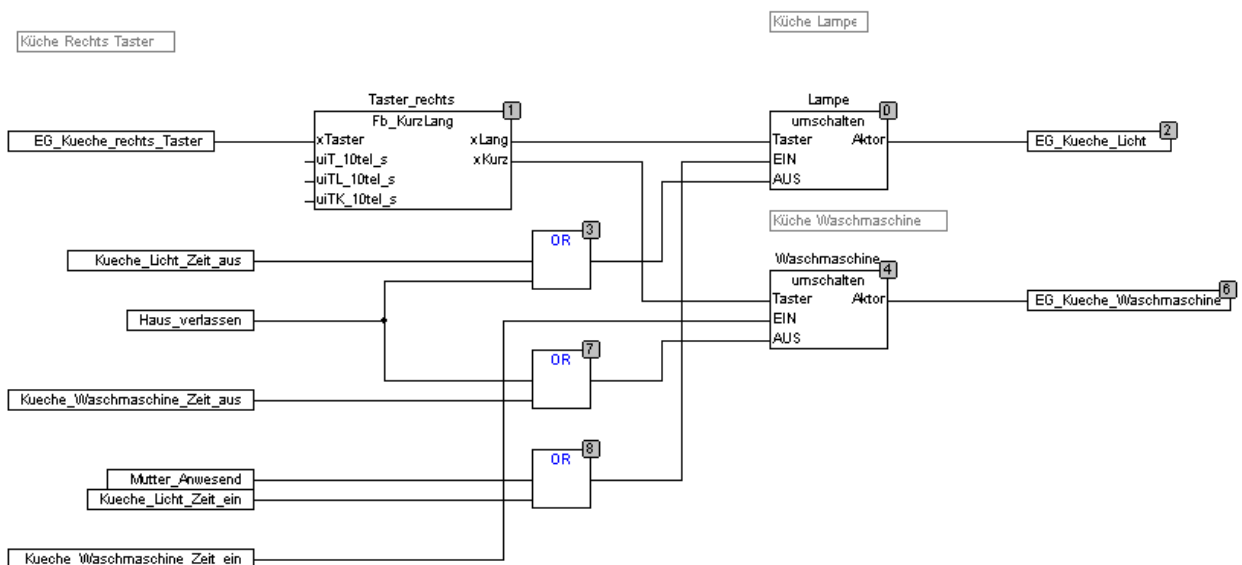
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Badezimmer:

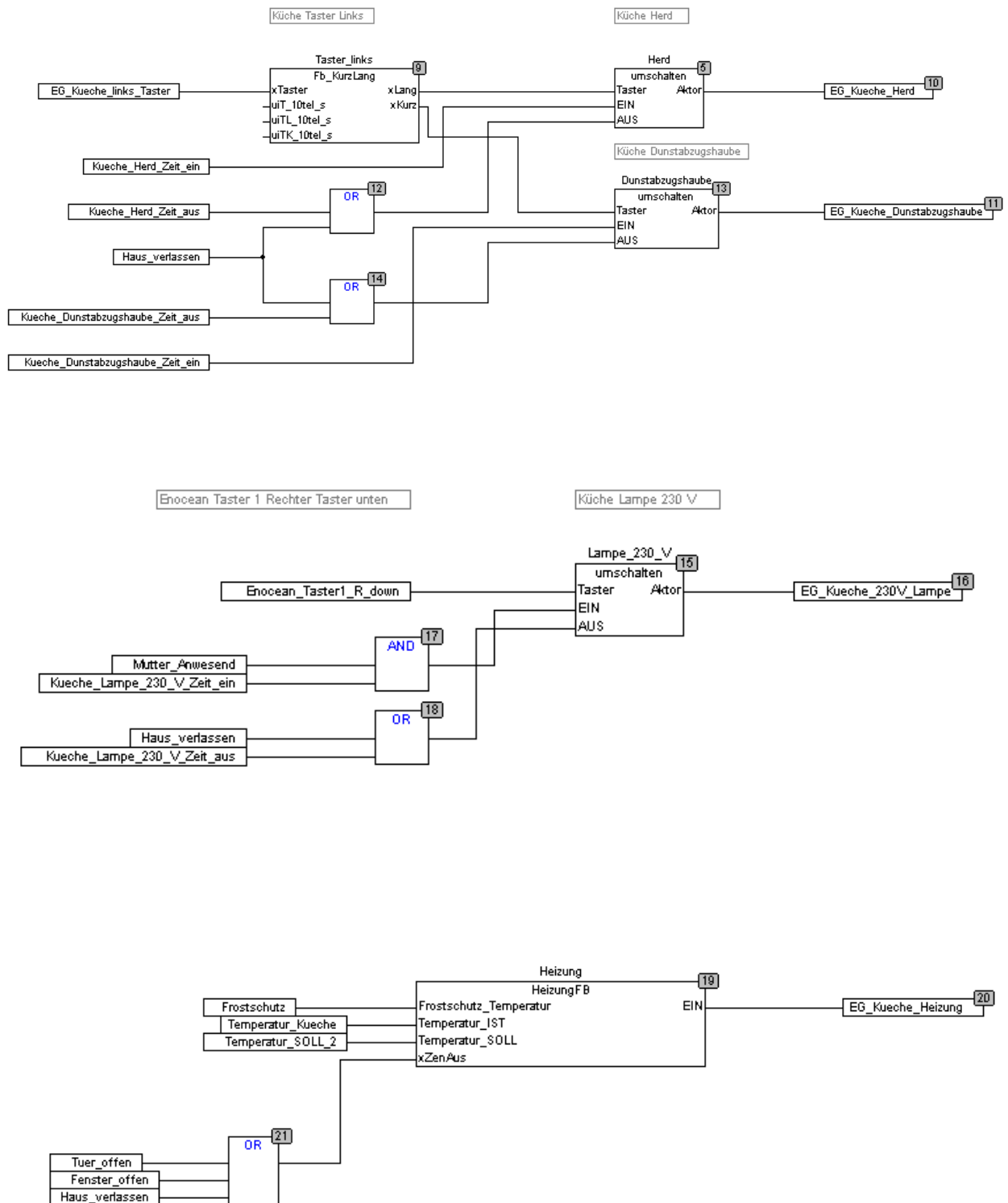
Variablen:

```
0001 PROGRAM Kueche
0002 VAR
0003     Taster_rechts: Fb_KurzLang;
0004     Taster_links: Fb_KurzLang;
0005     Lampe: umschalten;
0006     Waschmaschine: umschalten;
0007     Herd: umschalten;
0008     Dunstabzugshaube: umschalten;
0009     Lampe_230_V: umschalten;
0010     Heizung: HeizungFB;
0011 END_VAR
```

Quellcode:



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

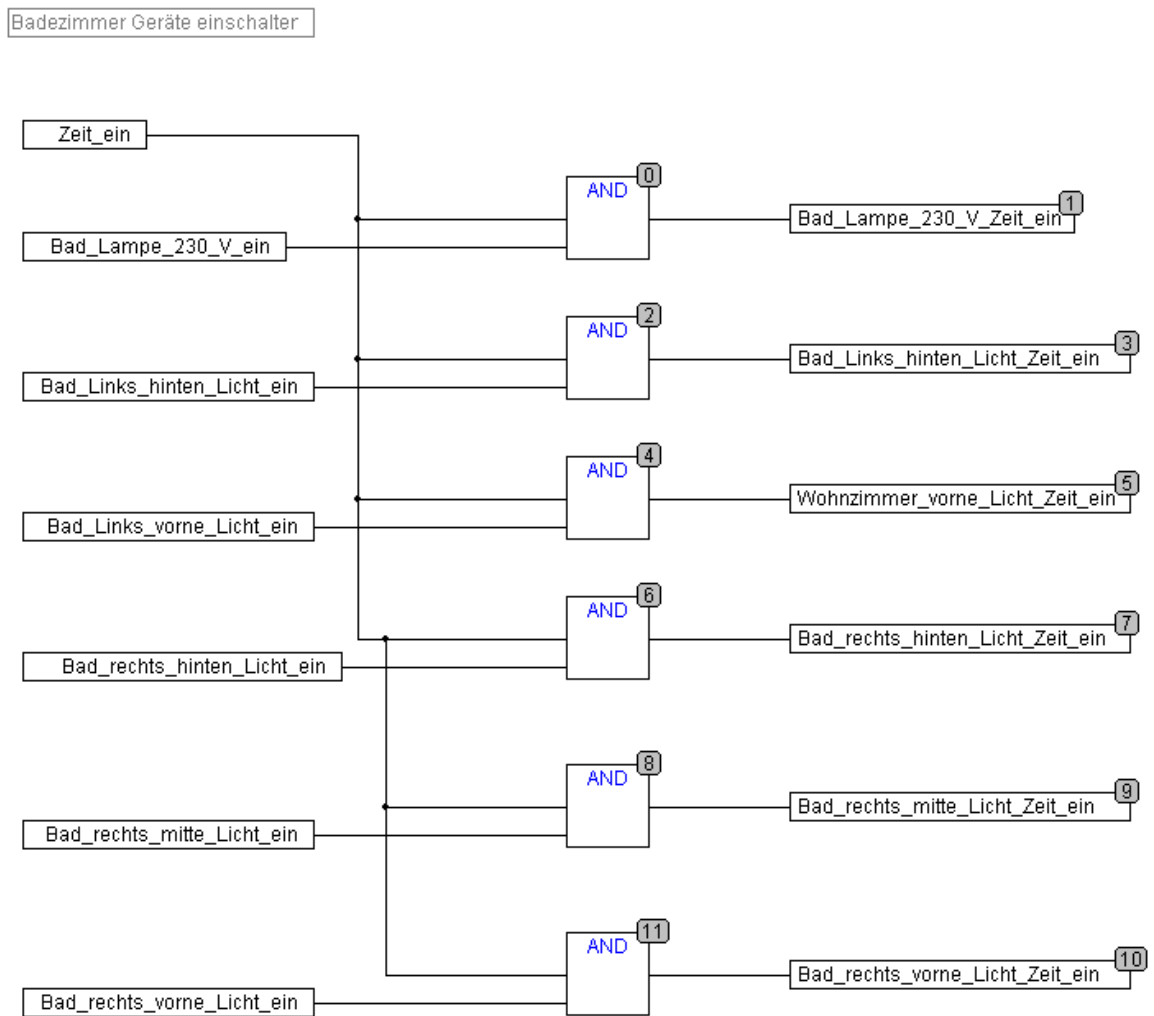
Timer:

Variablen:

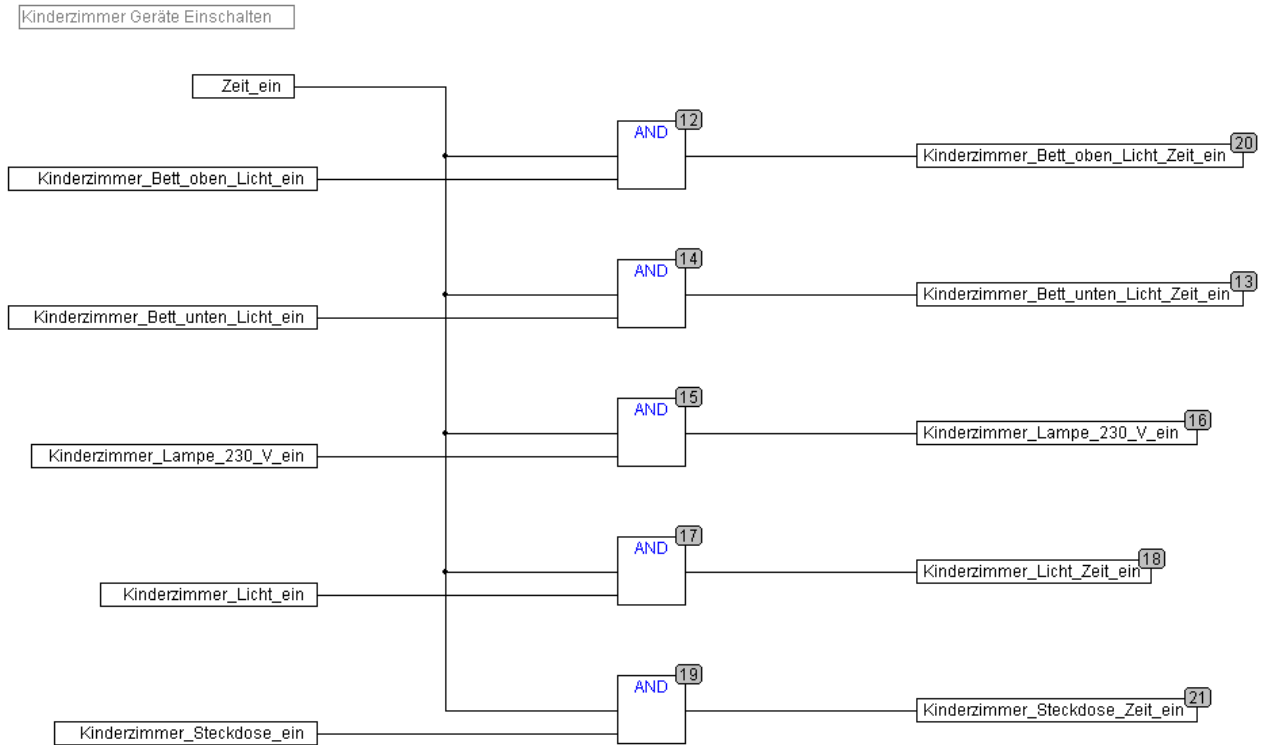
0001	PROGRAM PLC_PRG	0047	Kueche_Dunstabzugshaube_aus: BOOL ;
0002	VAR	0048	Kueche_Heizung_aus: BOOL ;
0003	Kinderzimmer_Steckdose_ein: BOOL ;	0049	Kueche_Lampe_230_V_aus: BOOL ;
0004	Kinderzimmer_Bett_oben_Licht_ein: BOOL ;	0050	Kueche_Licht_aus: BOOL ;
0005	Kinderzimmer_Bett_unten_Licht_ein: BOOL ;	0051	Kueche_Waschmaschine_aus: BOOL ;
0006	Kinderzimmer_Licht_ein: BOOL ;	0052	Schlafzimmer_Bett_Licht_aus: BOOL ;
0007	Kinderzimmer_Lampe_230_V_ein: BOOL ;	0053	Schlafzimmer_Heizung_aus: BOOL ;
0008	Kinderzimmer_Heizung_ein: BOOL ;	0054	Schlafzimmer_hinten_Licht_aus: BOOL ;
0009	Schlafzimmer_vorne_Licht_ein: BOOL ;	0055	Schlafzimmer_Lampe_230_V_aus: BOOL ;
0010	Schlafzimmer_hinten_Licht_ein: BOOL ;	0056	Schlafzimmer_vorne_Licht_aus: BOOL ;
0011	Schlafzimmer_Bett_Licht_ein: BOOL ;	0057	Wohnzimmer_Heizung_aus: BOOL ;
0012	Schlafzimmer_Lampe_230_V_ein: BOOL ;	0058	Wohnzimmer_hinten_Licht_aus: BOOL ;
0013	Schlafzimmer_Heizung_ein: BOOL ;	0059	Wohnzimmer_Lampe_230_V_aus: BOOL ;
0014	Kueche_Licht_ein: BOOL ;	0060	Wohnzimmer_Stehlampe_aus: BOOL ;
0015	Kueche_Waschmaschine_ein: BOOL ;	0061	Wohnzimmer_vorne_Licht_aus: BOOL ;
0016	Kueche_Herd_ein: BOOL ;	0062	Sekunde: INT ;
0017	Kueche_Dunstabzugshaube_ein: BOOL ;	0063	Minuten_ein: INT ;
0018	Kueche_Lampe_230_V_ein: BOOL ;	0064	Stunden_ein: INT ;
0019	Kueche_Heizung_ein: BOOL ;	0065	Wochentag_ein: INT ;
0020	Bad_Links_vorne_Licht_ein: BOOL ;	0066	Minuten_aus: INT ;
0021	Bad_Links_hinten_Licht_ein: BOOL ;	0067	Stunden_aus: INT ;
0022	Bad_rechts_vorne_Licht_ein: BOOL ;	0068	Wochentag_aus: INT ;
0023	Bad_rechts_hinten_Licht_ein: BOOL ;	0069	END_VAR
0024	Bad_rechts_mitte_Licht_ein: BOOL ;		
0025	Bad_Lampe_230_V_ein: BOOL ;		
0026	Bad_Heizung_ein: BOOL ;		
0027	Wohnzimmer_vorne_Licht_ein: BOOL ;		
0028	Wohnzimmer_hinten_Licht_ein: BOOL ;		
0029	Wohnzimmer_Stehlampe_ein: BOOL ;		
0030	Wohnzimmer_Lampe_230_V_ein: BOOL ;		
0031	Wohnzimmer_Heizung_ein: BOOL ;		
0032	Zeit_ein: BOOL ;		
0033	Bad_Heizung_aus: BOOL ;		
0034	Bad_Lampe_230_V_aus: BOOL ;		
0035	Bad_Links_hinten_Licht_aus: BOOL ;		
0036	Zeit_aus: BOOL ;		
0037	Bad_Links_vorne_Licht_aus: BOOL ;		
0038	Bad_rechts_hinten_Licht_aus: BOOL ;		
0039	Bad_rechts_mitte_Licht_aus: BOOL ;		
0040	Bad_rechts_vorne_Licht_aus: BOOL ;		
0041	Kinderzimmer_Bett_oben_Licht_aus: BOOL ;		
0042	Kinderzimmer_Bett_unten_Licht_aus: BOOL ;		
0043	Kinderzimmer_Heizung_aus: BOOL ;		
0044	Kinderzimmer_Licht_aus: BOOL ;		
0045	Kinderzimmer_Steckdose_aus: BOOL ;		
0046	Kueche_Herd_aus: BOOL ;		

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:

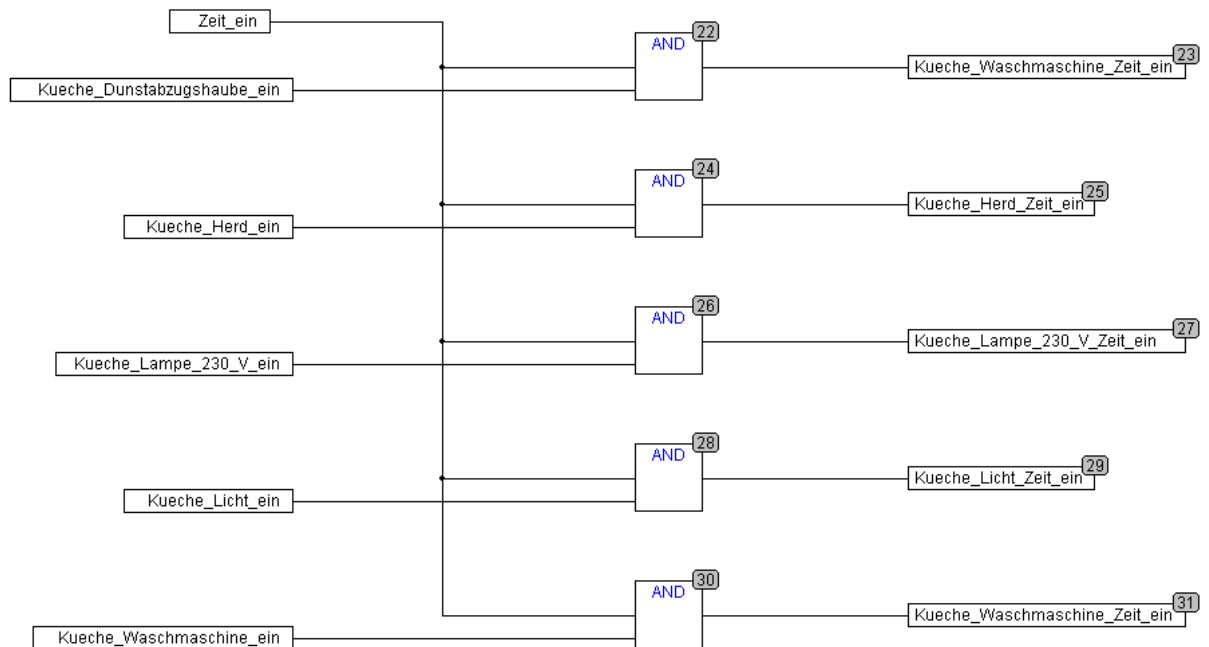


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

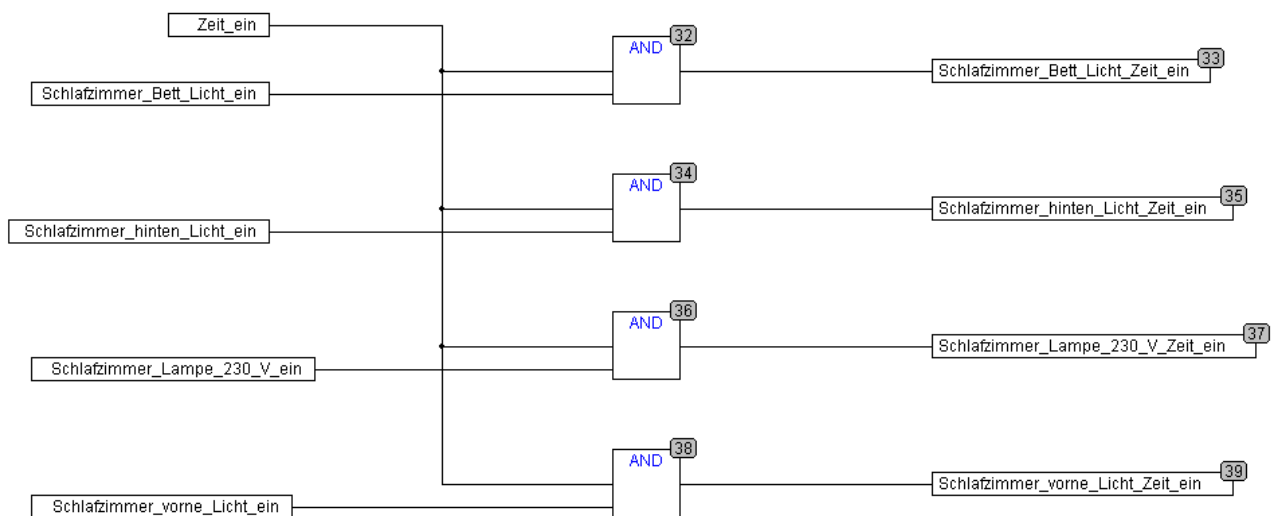


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

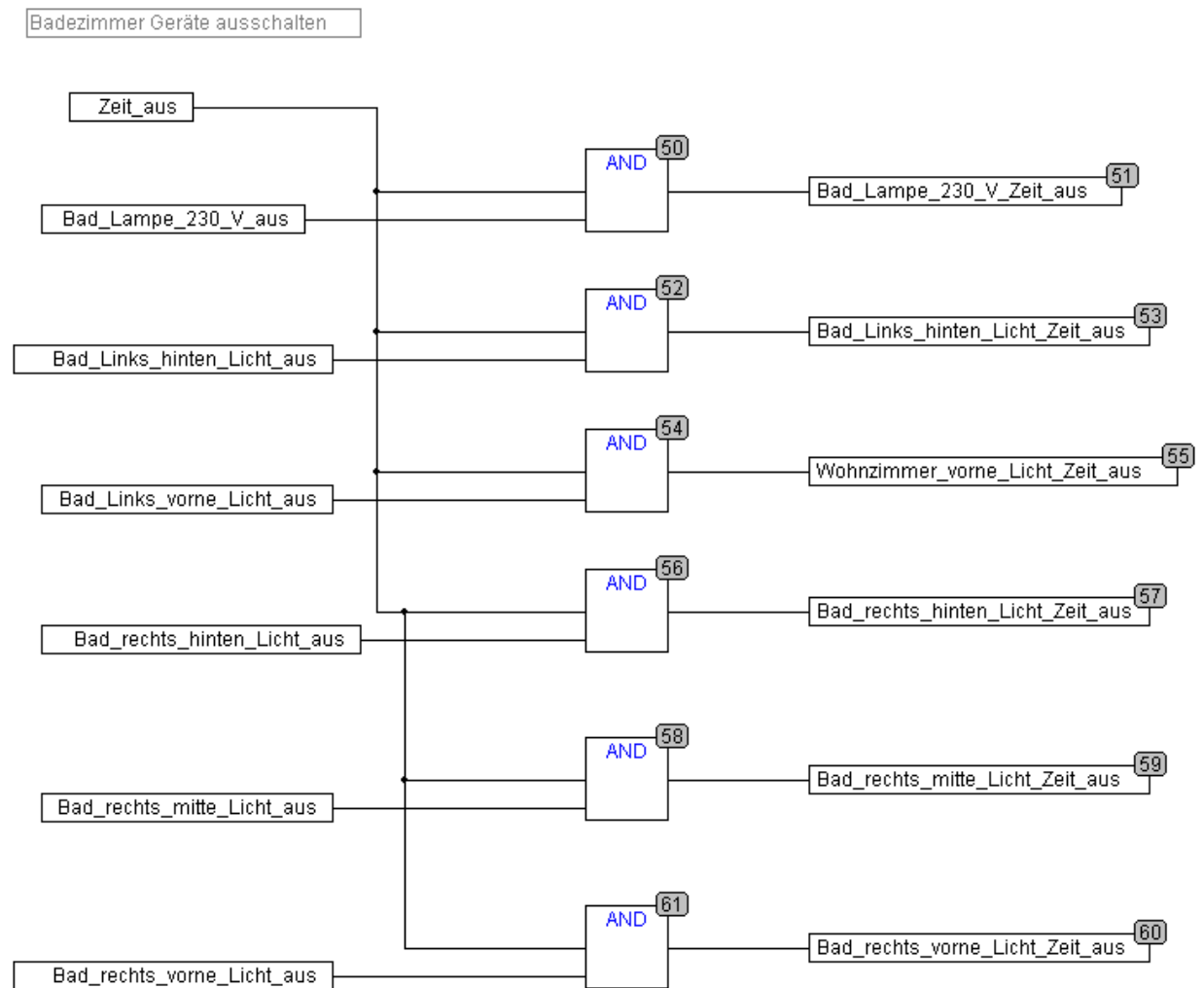
Küche Geräte einschalten



Schlafzimmer Geräte einschalten

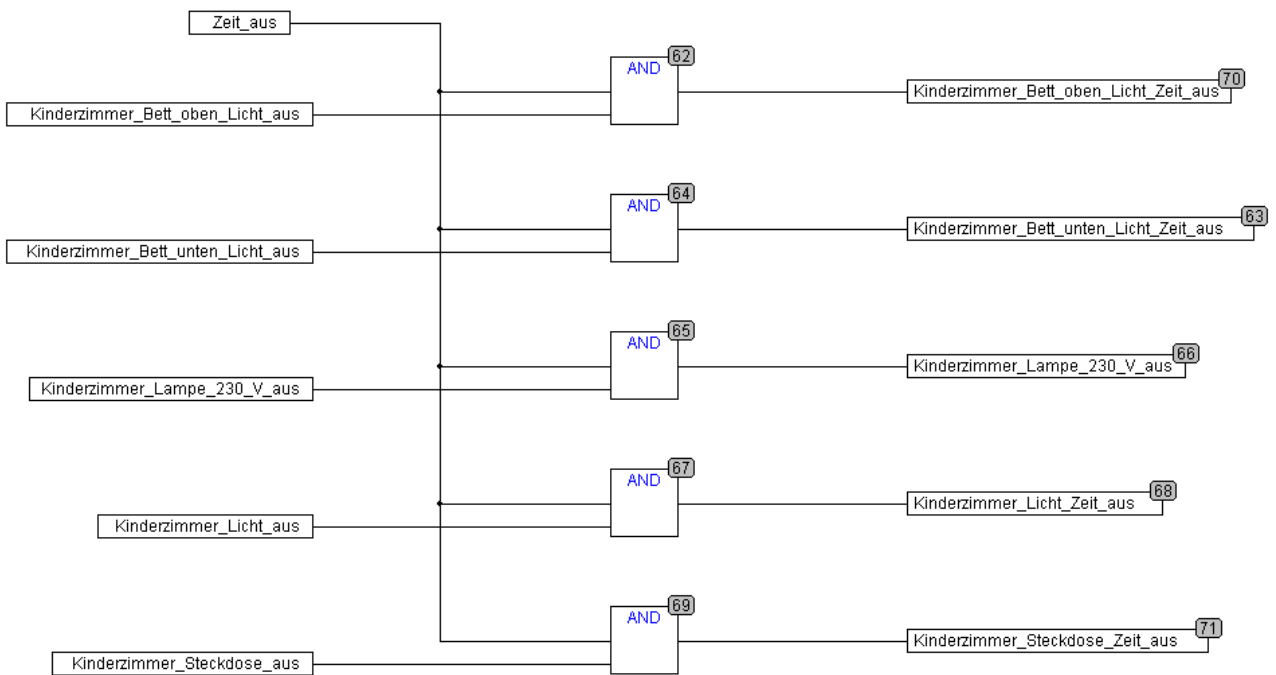


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

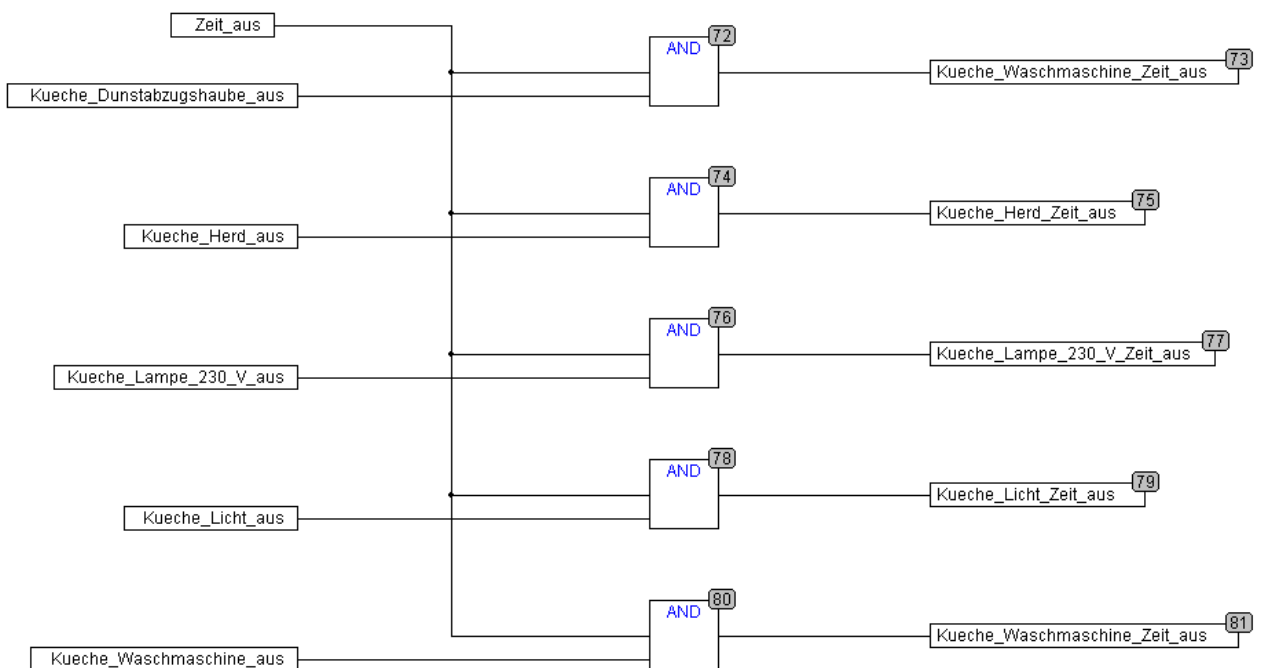


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Kinderzimmer Geräte ausschalten

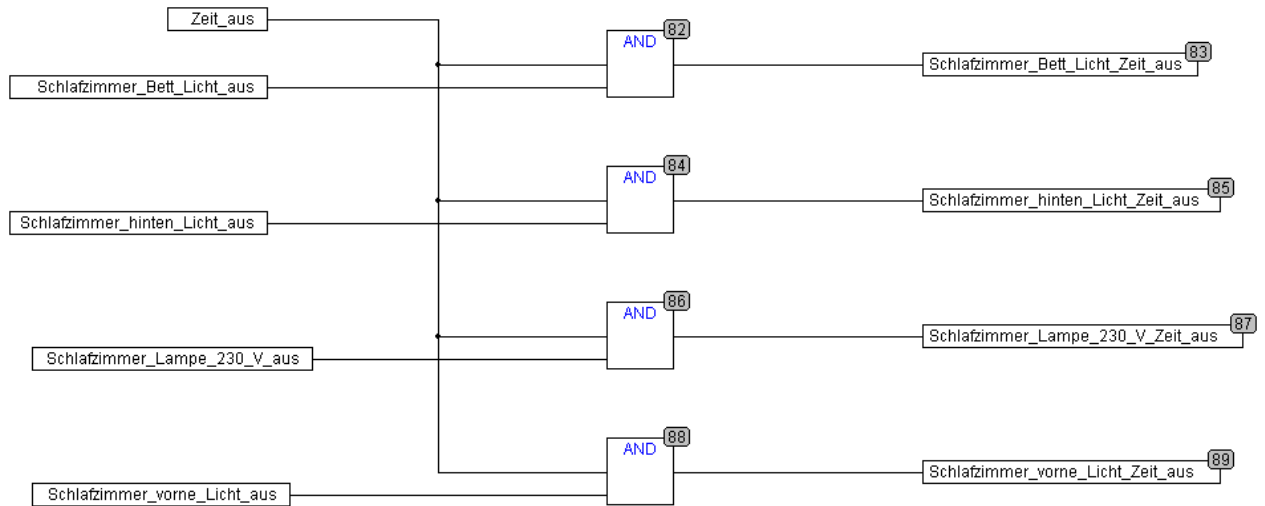


Küche Geräte ausschalten

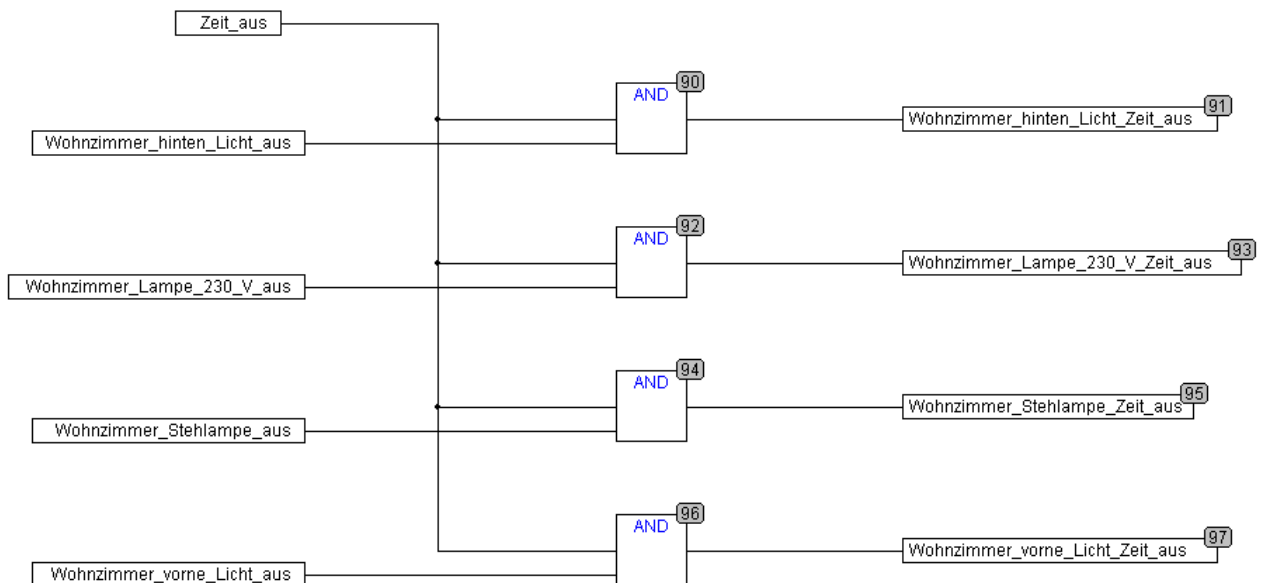


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Schlafzimmer Geräte ausschalten



Wohnzimmer Geräte ausschalten



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Zeitfunktion:

```
0001 IF Minuten_ein=aktMinute AND Stunden_ein=aktStunde AND Wochentag_ein=aktWeekday THEN
0002 Zeit_ein:=BOOL;
0003 END_IF
0004
0005 IF Minuten_aus=aktMinute AND Stunden_aus=aktStunde AND Wochentag_aus=aktWeekday THEN
0006 Zeit_aus:=BOOL;
0007 END_IF
```

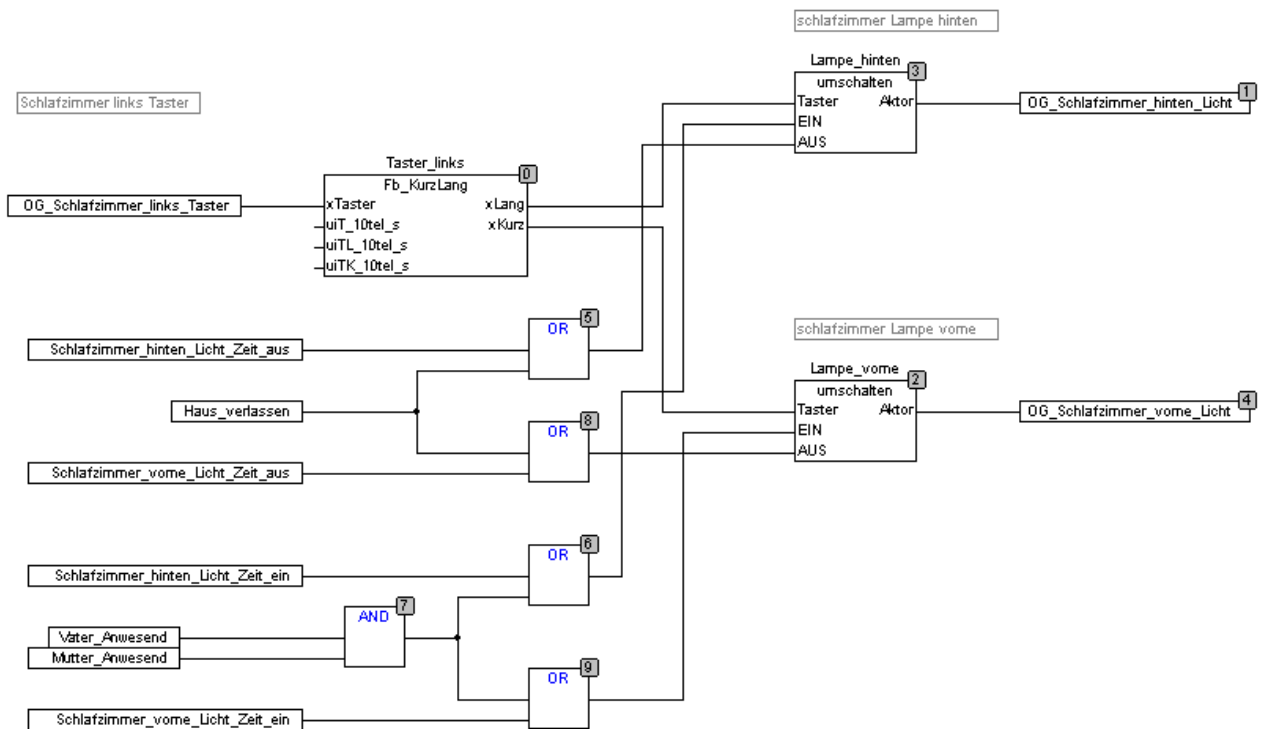
Schlafzimmer:

Variablen:

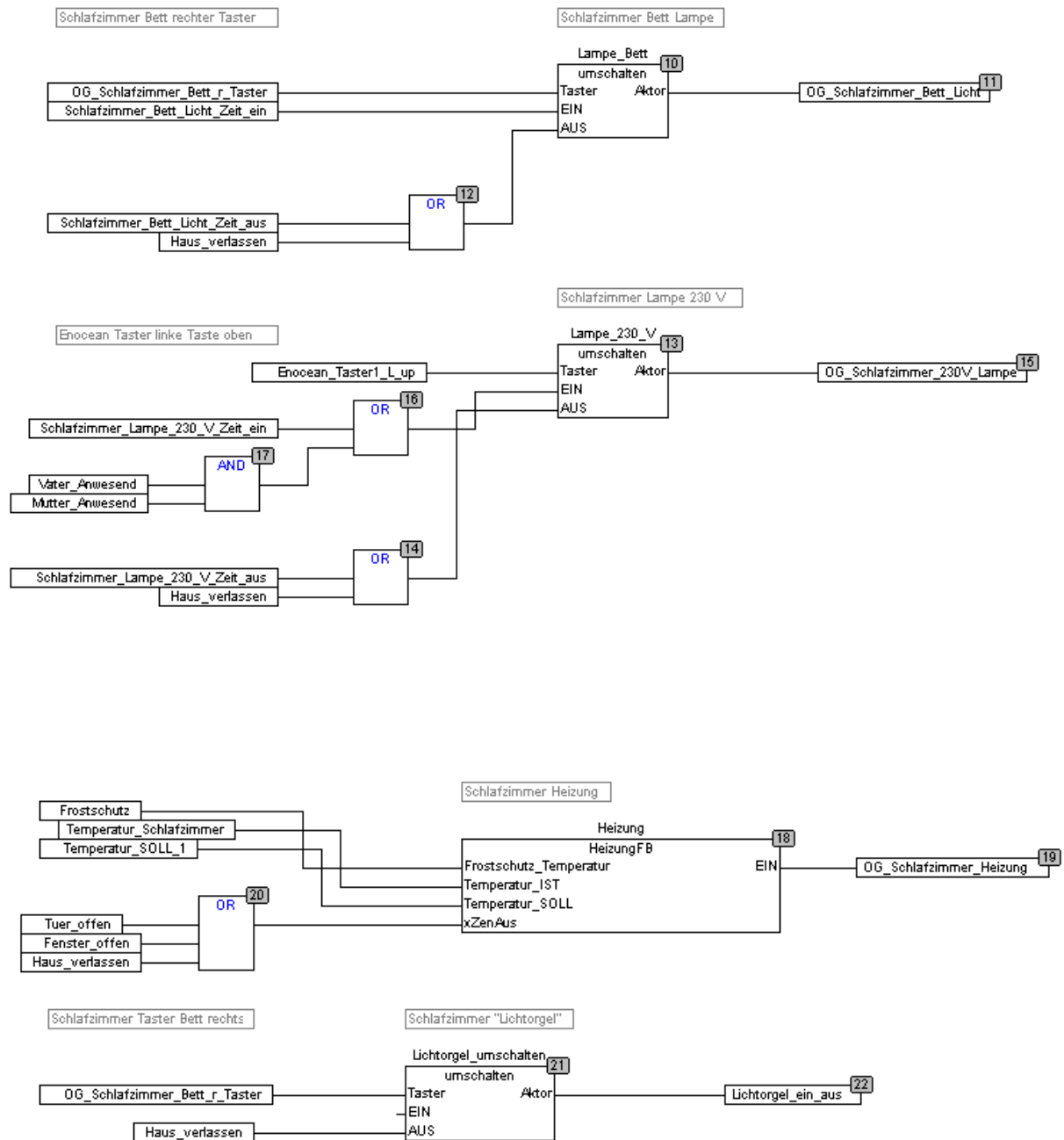
```
0001 PROGRAM Schlafzimmer
0002 VAR
0003     Taster_links: Fb_KurzLang;
0004     Lampe_Bett: umschalten;
0005     Lampe_230_V: umschalten;
0006     Heizung: HeizungFB;
0007     Lichtorgel_umschalten: umschalten;
0008     Lampe_hinten: umschalten;
0009     Lampe_vorne: umschalten;
0010 END_VAR
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quellcode:



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Lichtorgel:

```
0001 IF Lichtorgel_ein_aus=TRUE THEN
0002   Light_Blue:=Light_Blue+15;
0003   IF Light_Blue > 65535 THEN
0004     Light_Blue:=45000;
0005   END_IF
0006   Light_Yellow:=Light_Yellow+25;
0007   IF Light_Yellow > 65535 THEN
0008     Light_Yellow:=45000;
0009   END_IF
0010   Light_Green:=Light_Green+30;
0011   IF Light_Green > 65535 THEN
0012     Light_Green:=45000;
0013   END_IF
0014   Light_Red:=Light_Red+40;
0015   IF Light_Red > 65535 THEN
0016     Light_Red:=45000;
0017   END_IF
0018   Light_White:=Light_White+45;
0019   IF Light_White > 65535 THEN
0020     Light_White:=45000;
0021   END_IF
0022 ELSE
0023   Light_Blue:=0;
0024   Light_Yellow:=0;
0025   Light_Green:=0;
0026   Light_Red:=0;
0027   Light_White:=0;
0028 END_IF
```

Temperatur:

Variablen:

```
0001 PROGRAM Temperatur
0002 VAR
0003   Temperatur_Kinderzimmer_INT: INT;
0004   Temperatur_Schlafzimmer_INT: INT;
0005   Temperatur_Kueche_INT: INT;
0006   Temperatur_Badezimmer_INT: INT;
0007   Temperatur_Wohnzimmer_INT: INT;
0008   Temperatur_Aussen_INT: INT;
0009 END_VAR
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:

```
0001 Temperatur_Kinderzimmer_INT:=OG_Kinderzimmer_Temperatur;
0002 Temperatur_Schlafzimmer_INT:=OG_Schlafzimmer_Temperatur;
0003 Temperatur_Kueche_INT:=EG_Kueche_Temperatur;
0004 Temperatur_Badezimmer_INT:=EG_Bad_Temperatur;
0005 Temperatur_Wohnzimmer_INT:=EG_Wohnzimmer_Temperatur;
0006 Temperatur_Aussen_INT:=Haus_aussen_Temperatur;
0007
0008
0009 Temperatur_Kinderzimmer:=INT_TO_REAL(Temperatur_Kinderzimmer_INT)/10;
0010 Temperatur_Schlafzimmer:=INT_TO_REAL(Temperatur_Schlafzimmer_INT)/10;
0011 Temperatur_Kueche:=INT_TO_REAL(Temperatur_Kueche_INT)/10;
0012 Temperatur_Badezimmer:=INT_TO_REAL(Temperatur_Badezimmer_INT)/10;
0013 Temperatur_Wohnzimmer:=INT_TO_REAL(Temperatur_Wohnzimmer_INT)/10;
0014 Temperatur_Aussen:=INT_TO_REAL(Temperatur_Aussen_INT)/10;
0015
0016 Frostschutz:=15; (*Frostschutztemperatur*)
```

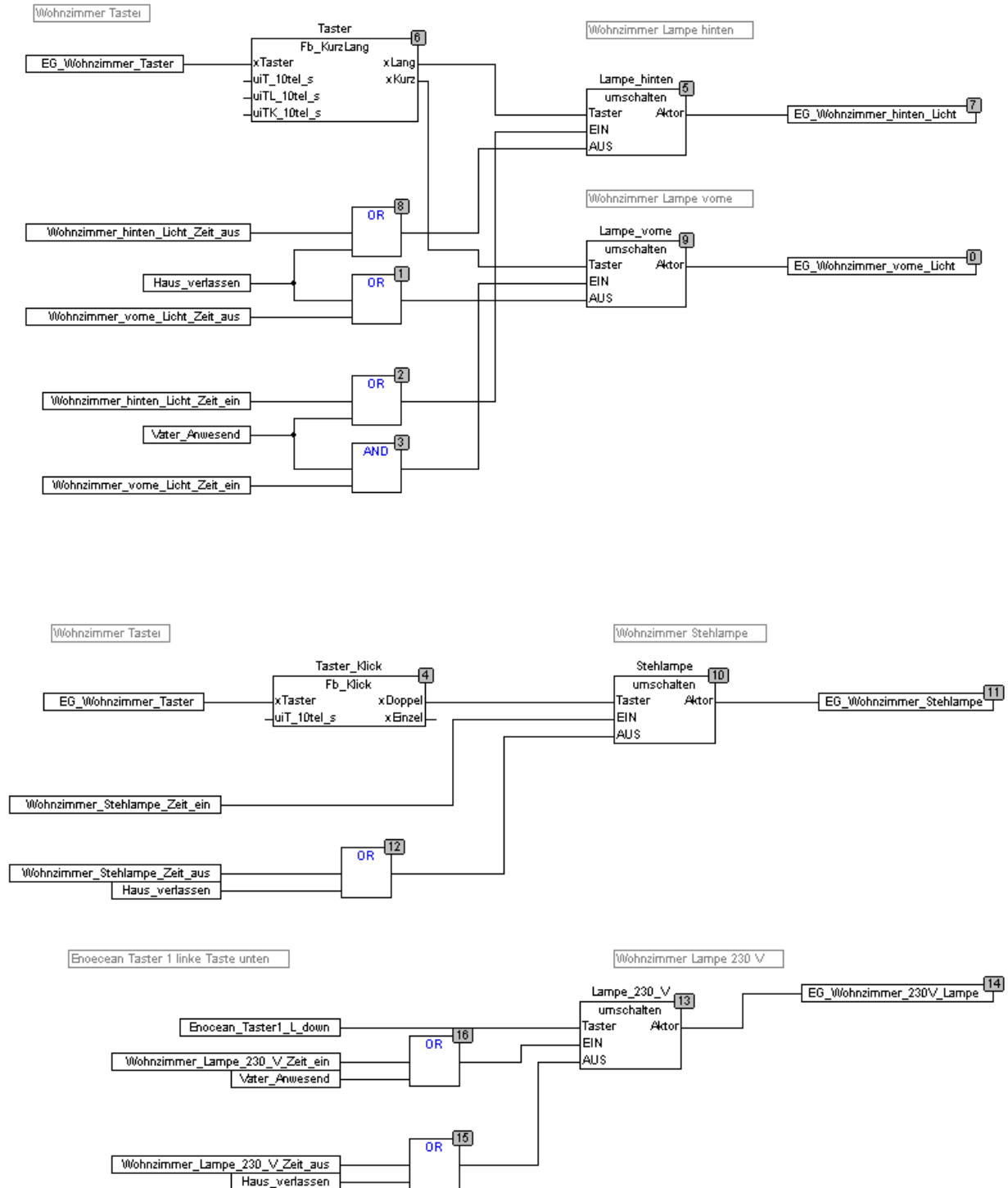
Wohnzimmer:

Variablen:

```
0001 PROGRAM Wohnzimmer
0002 VAR
0003     Taster: Fb_KurzLang;
0004     Taster_Klick: Fb_Klick;
0005     Lampe_hinten: umschalten;
0006     Lampe_vorne: umschalten;
0007     Stehlampe: umschalten;
0008     Lampe_230_V: umschalten;
0009     Heizung: HeizungFB;
0010 END_VAR
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:



Enocean:

Variablen:

0001	PROGRAM	Enocean
0002	VAR	
0003	Receiver	: FbEnoceanReceive;
0004	typEnocean	: typEnocean;
0005	bError	: BYTE;
0006	Search_ID	: FbShow_ID_By_Click;
0007	Received_ID	: DWORD;
0008	Set_ID	: BOOL;
0009	Enocean_Button	: FbButton_2_Channel;
0010	(*Variablen für Thermokon Sensor*)	
0011	Search_ID_2	: FbShow_ID_By_Click;
0012	Set_ID_2	: BOOL;
0013	Received_ID_2	: DWORD;
0014	Thermokon_Sensor1	: FbSR04;
0015	Thermokon_Sensor2	: FbSR04;
0016	Error1	: BOOL;
0017	Presence1	: BOOL;
0018	Temperature1	: REAL;
0019	Setp_Correction1	: REAL;
0020	Auto1	: BOOL;
0021	Level_1_0	: BOOL;
0022	Level_1_1	: BOOL;
0023	Level_1_2	: BOOL;
0024	Level_1_3	: BOOL;
0025	Error2	: BOOL;
0026	Presence2	: BOOL;
0027	Temperature2	: REAL;
0028	Setp_Correction2	: REAL;
0029	Auto2	: BOOL;
0030	Level_2_0	: BOOL;
0031	Level_2_1	: BOOL;
0032	Level_2_2	: BOOL;
0033	Level_2_3	: BOOL;
0034	(*-----*)	
0035	Drehung_1	: INT; (*nur für Visualisierung*)
0036	Drehung_2	: INT; (*nur für Visualisierung*)
0037	Taster1	: FbButton_4_Channel;
0038		
0039		
0040	Taster2	: FbButton_4_Channel;
0041		
0042	END_VAR	

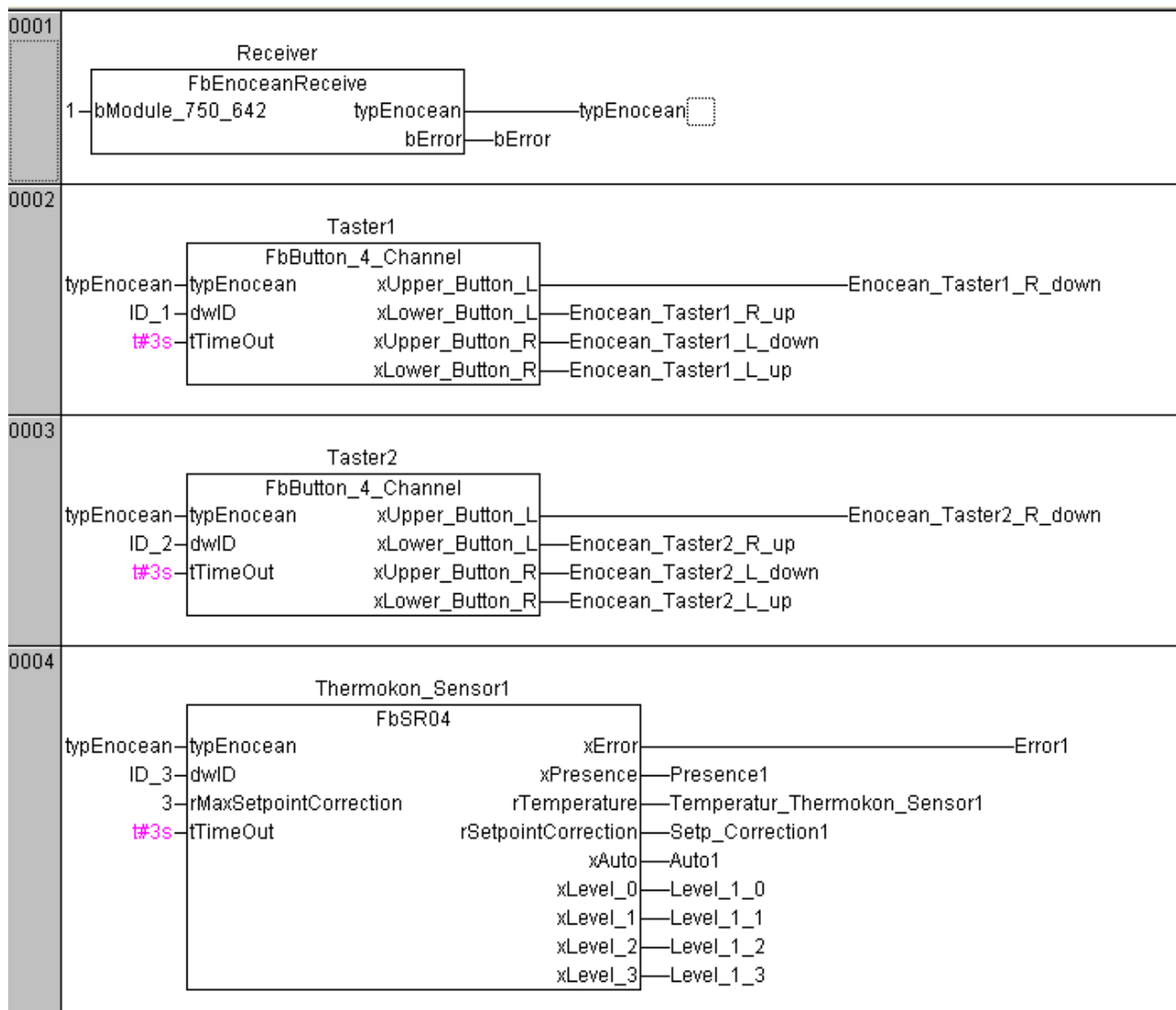
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

```

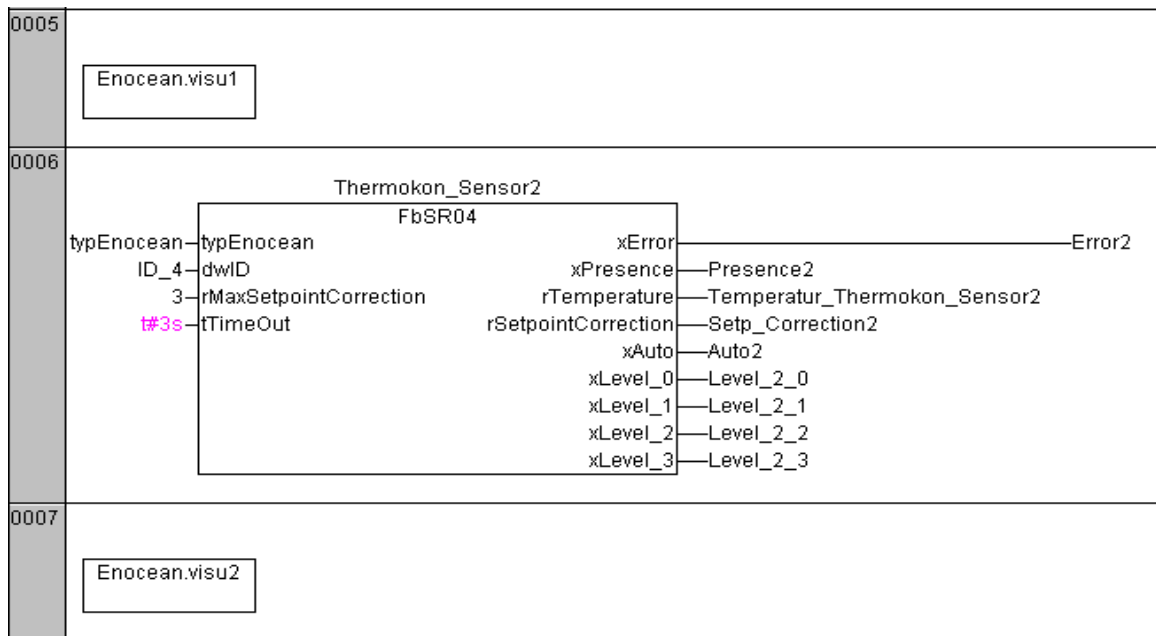
0044 VAR RETAIN
0045 ID_1          : DWORD := 1180628;  (*ID für Taster1*)
0046 ID_2          : DWORD := 1140678;  (*ID für Taster2*)
0047 ID_3          : DWORD := 62059;    (*ID für Temperaturregler1*)
0048 ID_4          : DWORD := 62052;    (*ID für Temperaturregler2*)
0049 END_VAR

```

Quelltext:



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Visu 1:

```
0001 IF Auto1 THEN
0002 Drehung_1:=0;
0003 ELSIF Level_1_0 THEN
0004 Drehung_1:=30;
0005 ELSIF Level_1_1 THEN
0006 Drehung_1:=60;
0007 ELSIF Level_1_2 THEN
0008 Drehung_1:=90;
0009 ELSIF Level_1_3 THEN
0010 Drehung_1:=120;
0011 ELSE
0012 Drehung_1:=0;
0013 END_IF
0014
0015 Drehung_2:= REAL_TO_INT(Setp_Correction1*400/10);
0016 Temperatur_SOLL_1:=Temperatur_Thermokon_Sensor1+Setp_Correction1;
```

Visu 2:

```
0001 IF Auto1 THEN
0002 Drehung_2:=0;
0003 ELSIF Level_2_0 THEN
0004 Drehung_2:=30;
0005 ELSIF Level_2_1 THEN
0006 Drehung_2:=60;
0007 ELSIF Level_2_2 THEN
0008 Drehung_2:=90;
0009 ELSIF Level_2_3 THEN
0010 Drehung_2:=120;
0011 ELSE
0012 Drehung_2:=0;
0013 END_IF
0014
0015 Drehung_2:= REAL_TO_INT(Setp_Correction1*400/10);
0016 Temperatur_SOLL_2:=Temperatur_Thermokon_Sensor2+Setp_Correction2;
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

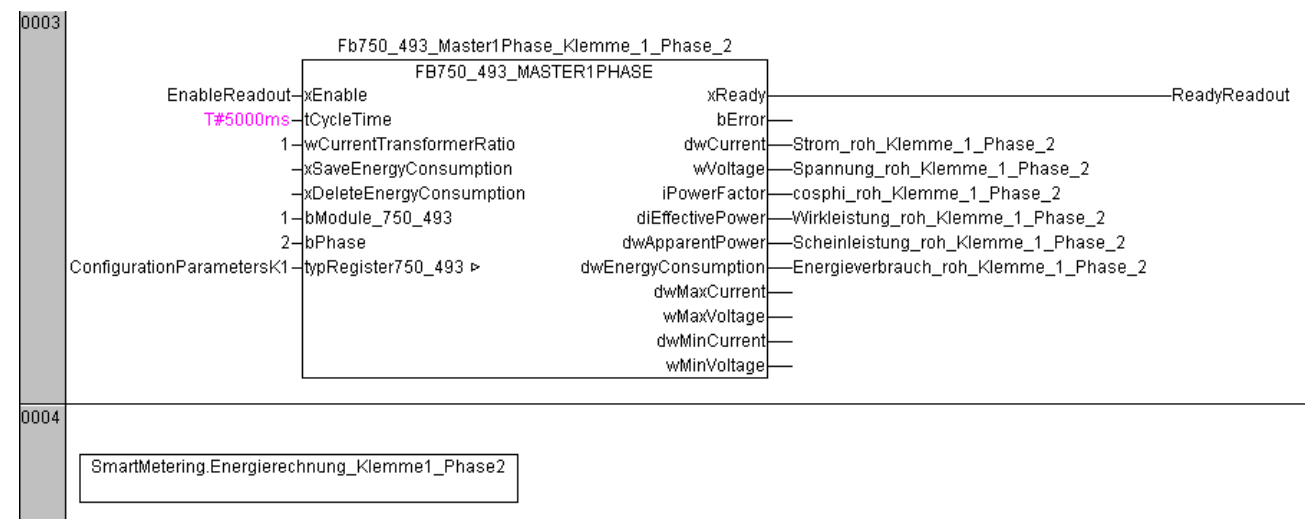
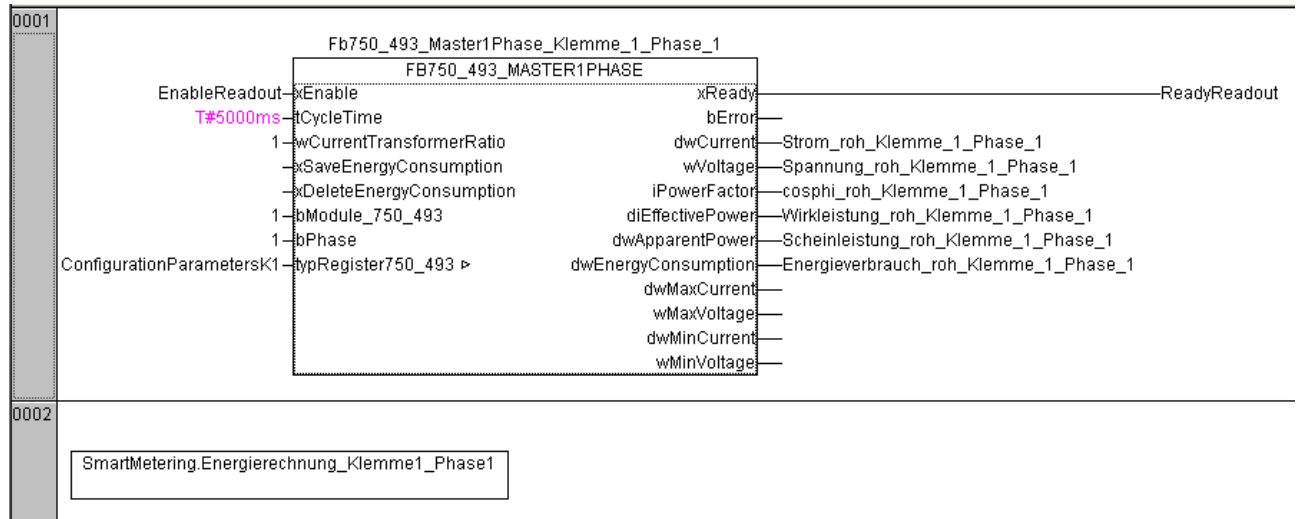
Smart Metering:

Variablen:

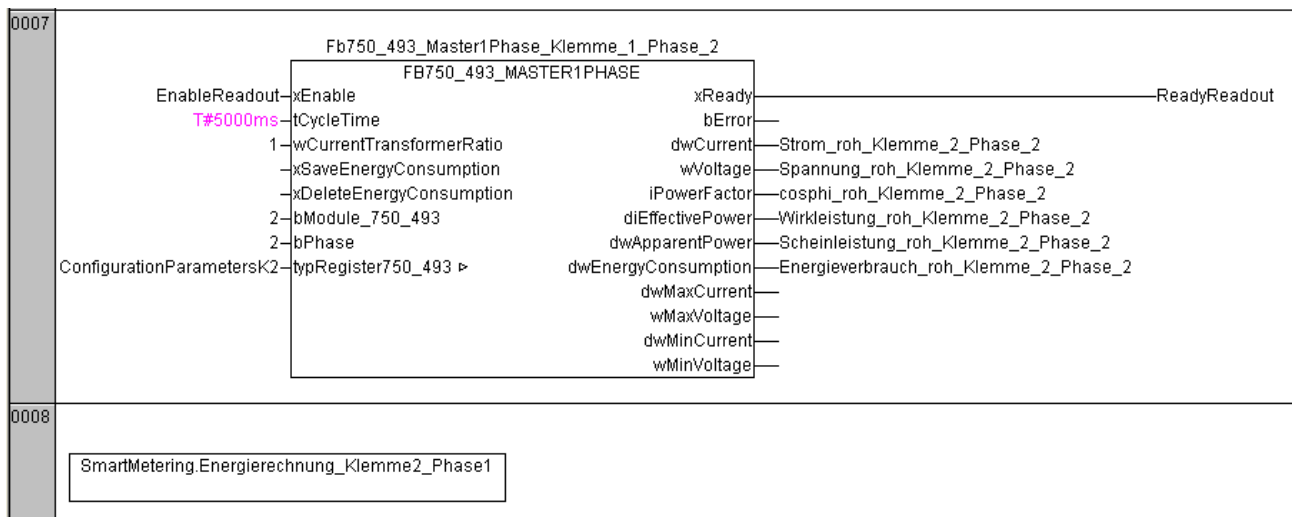
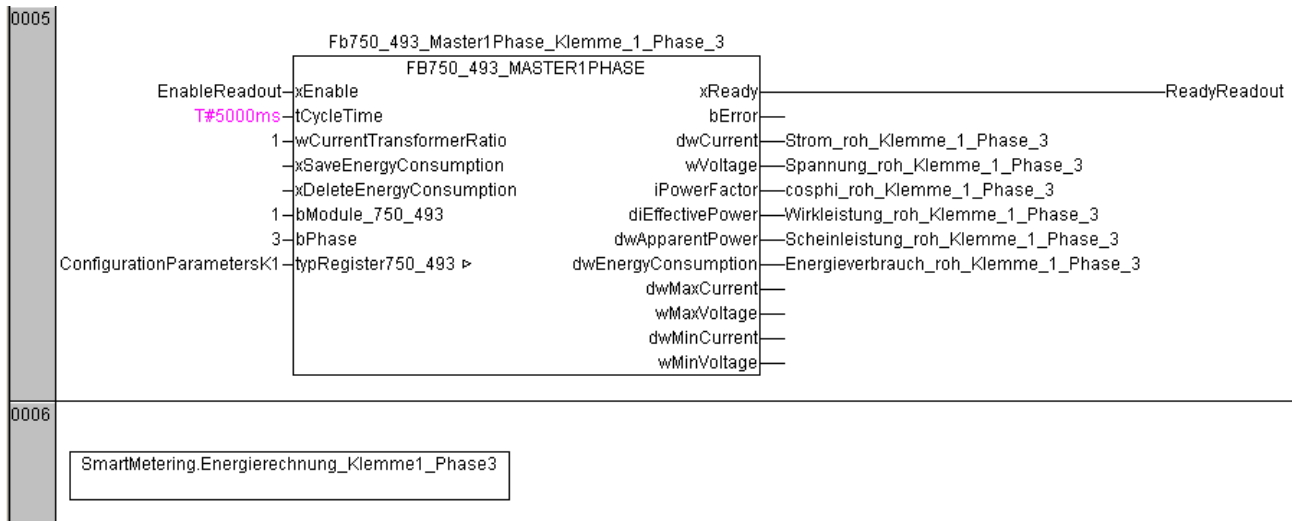
```
0001 PROGRAM SmartMetering
0002 VAR
0003   EnableReadout: BOOL := TRUE;
0004   Strom_roh_Klemme_1_Phase_1: DWORD;
0005   Spannung_roh_Klemme_1_Phase_1: WORD;
0006   cosphi_roh_Klemme_1_Phase_1: INT;
0007   Wirkleistung_roh_Klemme_1_Phase_1: DINT;
0008   Scheinleistung_roh_Klemme_1_Phase_1: DWORD;
0009   Energieverbrauch_roh_Klemme_1_Phase_1: DWORD;
0010   Strom_roh_Klemme_1_Phase_2: DWORD;
0011   Spannung_roh_Klemme_1_Phase_2: WORD;
0012   cosphi_roh_Klemme_1_Phase_2: INT;
0013   Wirkleistung_roh_Klemme_1_Phase_2: DINT;
0014   Scheinleistung_roh_Klemme_1_Phase_2: DWORD;
0015   Energieverbrauch_roh_Klemme_1_Phase_2: DWORD;
0016   Strom_roh_Klemme_1_Phase_3: DWORD;
0017   Spannung_roh_Klemme_1_Phase_3: WORD;
0018   cosphi_roh_Klemme_1_Phase_3: INT;
0019   Wirkleistung_roh_Klemme_1_Phase_3: DINT;
0020   Scheinleistung_roh_Klemme_1_Phase_3: DWORD;
0021   Energieverbrauch_roh_Klemme_1_Phase_3: DWORD;
0022   Strom_roh_Klemme_2_Phase_1: DWORD;
0023   Spannung_roh_Klemme_2_Phase_1: WORD;
0024   cosphi_roh_Klemme_2_Phase_1: INT;
0025   Wirkleistung_roh_Klemme_2_Phase_1: DINT;
0026   Scheinleistung_roh_Klemme_2_Phase_1: DWORD;
0027   Energieverbrauch_roh_Klemme_2_Phase_1: DWORD;
0028   Strom_roh_Klemme_2_Phase_2: DWORD;
0029   Spannung_roh_Klemme_2_Phase_2: WORD;
0030   cosphi_roh_Klemme_2_Phase_2: INT;
0031   Wirkleistung_roh_Klemme_2_Phase_2: DINT;
0032   Scheinleistung_roh_Klemme_2_Phase_2: DWORD;
0033   Energieverbrauch_roh_Klemme_2_Phase_2: DWORD;
0034   Strom_roh_Klemme_2_Phase_3: DWORD;
0035   Spannung_roh_Klemme_2_Phase_3: WORD;
0036   cosphi_roh_Klemme_2_Phase_3: INT;
0037   Wirkleistung_roh_Klemme_2_Phase_3: DINT;
0038   Scheinleistung_roh_Klemme_2_Phase_3: DWORD;
0039   Energieverbrauch_roh_Klemme_2_Phase_3: DWORD;
0040   ReadyReadout: BOOL;
0041   Fb750_493_Master1Phase_Klemme_1_Phase_1: Fb750_493_Master1Phase;
0042   Fb750_493_Master1Phase_Klemme_1_Phase_2: Fb750_493_Master1Phase;
0043   Fb750_493_Master1Phase_Klemme_1_Phase_3: Fb750_493_Master1Phase;
0044   Fb750_493_Master1Phase_Klemme_2_Phase_1: Fb750_493_Master1Phase;
0045   Fb750_493_Master1Phase_Klemme_2_Phase_2: Fb750_493_Master1Phase;
0046   Fb750_493_Master1Phase_Klemme_2_Phase_3: Fb750_493_Master1Phase;
0047   MUL_Wirkleistung: REAL := 0.1;
0048   MUL_Energie: REAL := 0.001;
0049   MUL_Scheinleistung: REAL := 0.1;
0050   MUL_Spannung: REAL := 0.1;
0051   MUL_Strom: REAL := 0.001;
0052 END_VAR
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

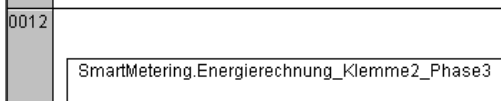
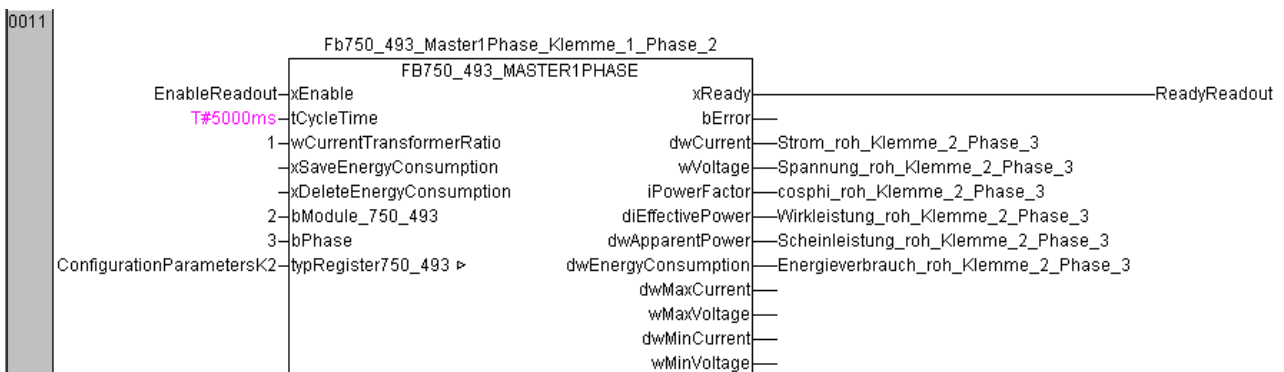
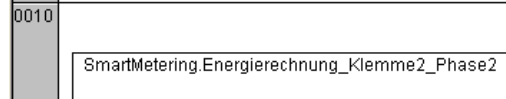
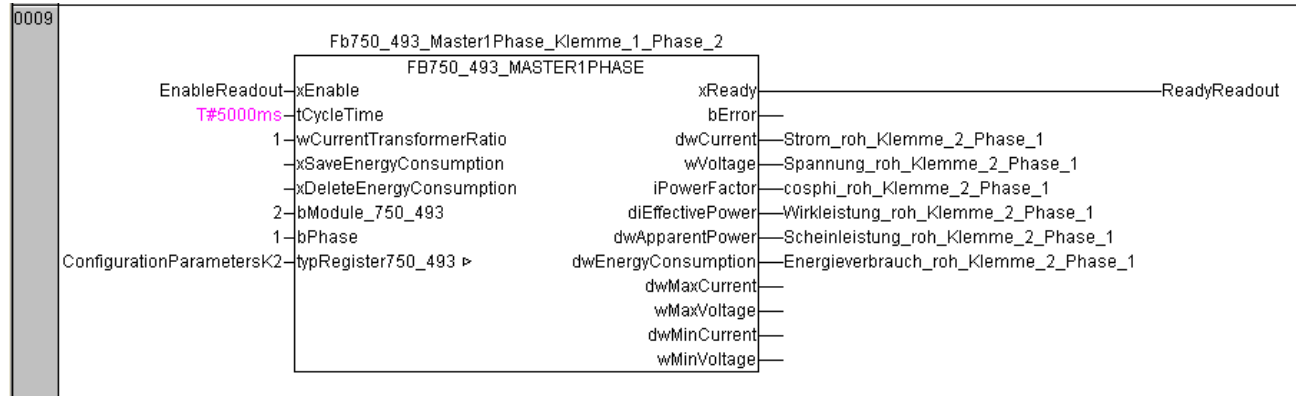
Quelltext:



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Energierechnung Klemme 1 Phase 1:

Quelltext:

0001	cosphi_Klemme_1_Phase_1:=INT_TO_REAL(cosphi_roh_Klemme_1_Phase_1)/100;
0002	Wert_Wirkleistung_Klemme_1_Phase_1:=DINT_TO_REAL(Wirkleistung_roh_Klemme_1_Phase_1)*MUL_Wirkleistung;
0003	Wert_Scheinleistung_Klemme_1_Phase_1:=DWORD_TO_REAL(Scheinleistung_roh_Klemme_1_Phase_1)*MUL_Scheinleistung;
0004	Wert_Spannung_Klemme_1_Phase_1:=DWORD_TO_REAL(Spannung_roh_Klemme_1_Phase_1)*MUL_Spannung;
0005	Wert_Strom_Klemme_1_Phase_1:=DWORD_TO_REAL(Strom_roh_Klemme_1_Phase_1)*MUL_Strom;
0006	Wert_Energieverbrauch_Klemme_1_Phase_1:=DWORD_TO_REAL(Energieverbrauch_roh_Klemme_1_Phase_1)*MUL_Energie;

Energierechnung Klemme 1 Phase 2:

Quelltext:

0001	cosphi_Klemme_1_Phase_2:=INT_TO_REAL(cosphi_roh_Klemme_1_Phase_2)/100;
0002	Wert_Wirkleistung_Klemme_1_Phase_2:=DINT_TO_REAL(Wirkleistung_roh_Klemme_1_Phase_2)*MUL_Wirkleistung;
0003	Wert_Scheinleistung_Klemme_1_Phase_2:=DWORD_TO_REAL(Scheinleistung_roh_Klemme_1_Phase_2)*MUL_Scheinleistung;
0004	Wert_Spannung_Klemme_1_Phase_2:=DWORD_TO_REAL(Spannung_roh_Klemme_1_Phase_2)*MUL_Spannung;
0005	Wert_Strom_Klemme_1_Phase_2:=DWORD_TO_REAL(Strom_roh_Klemme_1_Phase_2)*MUL_Strom;
0006	Wert_Energieverbrauch_Klemme_1_Phase_2:=DWORD_TO_REAL(Energieverbrauch_roh_Klemme_1_Phase_2)*MUL_Energie;

Energierechnung Klemme 1 Phase 3:

Quelltext:

0001	cosphi_Klemme_1_Phase_3:=INT_TO_REAL(cosphi_roh_Klemme_1_Phase_3)/100;
0002	Wert_Wirkleistung_Klemme_1_Phase_3:=DINT_TO_REAL(Wirkleistung_roh_Klemme_1_Phase_3)*MUL_Wirkleistung;
0003	Wert_Scheinleistung_Klemme_1_Phase_3:=DWORD_TO_REAL(Scheinleistung_roh_Klemme_1_Phase_3)*MUL_Scheinleistung;
0004	Wert_Spannung_Klemme_1_Phase_3:=DWORD_TO_REAL(Spannung_roh_Klemme_1_Phase_3)*MUL_Spannung;
0005	Wert_Strom_Klemme_1_Phase_3:=DWORD_TO_REAL(Strom_roh_Klemme_1_Phase_3)*MUL_Strom;
0006	Wert_Energieverbrauch_Klemme_1_Phase_3:=DWORD_TO_REAL(Energieverbrauch_roh_Klemme_1_Phase_3)*MUL_Energie;

Energierechnung Klemme 2 Phase 1:

Quelltext:

```
0001 cosphi_Klemme_2_Phase_1:=INT_TO_REAL(cosphi_roh_Klemme_2_Phase_1)/100;  
0002 Wert_Wirkleistung_Klemme_2_Phase_1:=DINT_TO_REAL(Wirkleistung_roh_Klemme_2_Phase_1)*MUL_Wirkleistung;  
0003 Wert_Scheinleistung_Klemme_2_Phase_1:=DWORD_TO_REAL(Scheinleistung_roh_Klemme_2_Phase_1)*MUL_Scheinleistung;  
0004 Wert_Spannung_Klemme_2_Phase_1:=DWORD_TO_REAL(Spannung_roh_Klemme_2_Phase_1)*MUL_Spannung;  
0005 Wert_Strom_Klemme_2_Phase_1:=DWORD_TO_REAL(Strom_roh_Klemme_2_Phase_1)*MUL_Strom;  
0006 Wert_Energieverbrauch_Klemme_2_Phase_1:=DWORD_TO_REAL(Energieverbrauch_roh_Klemme_2_Phase_1)*MUL_Energie;
```

Energierechnung Klemme 2 Phase 2:

Quelltext:

```
0001 cosphi_Klemme_2_Phase_2:=INT_TO_REAL(cosphi_roh_Klemme_2_Phase_2)/100;  
0002 Wert_Wirkleistung_Klemme_2_Phase_2:=DINT_TO_REAL(Wirkleistung_roh_Klemme_2_Phase_2)*MUL_Wirkleistung;  
0003 Wert_Scheinleistung_Klemme_2_Phase_2:=DWORD_TO_REAL(Scheinleistung_roh_Klemme_2_Phase_2)*MUL_Scheinleistung;  
0004 Wert_Spannung_Klemme_2_Phase_2:=DWORD_TO_REAL(Spannung_roh_Klemme_2_Phase_2)*MUL_Spannung;  
0005 Wert_Strom_Klemme_2_Phase_2:=DWORD_TO_REAL(Strom_roh_Klemme_2_Phase_2)*MUL_Strom;  
0006 Wert_Energieverbrauch_Klemme_2_Phase_2:=DWORD_TO_REAL(Energieverbrauch_roh_Klemme_2_Phase_2)*MUL_Energie;
```

Energierechnung Klemme 2 Phase 3:

Quelltext:

```
0001 cosphi_Klemme_2_Phase_3:=INT_TO_REAL(cosphi_roh_Klemme_2_Phase_3)/100;  
0002 Wert_Wirkleistung_Klemme_2_Phase_3:=DINT_TO_REAL(Wirkleistung_roh_Klemme_2_Phase_3)*MUL_Wirkleistung;  
0003 Wert_Scheinleistung_Klemme_2_Phase_3:=DWORD_TO_REAL(Scheinleistung_roh_Klemme_2_Phase_3)*MUL_Scheinleistung;  
0004 Wert_Spannung_Klemme_2_Phase_3:=DWORD_TO_REAL(Spannung_roh_Klemme_2_Phase_3)*MUL_Spannung;  
0005 Wert_Strom_Klemme_2_Phase_3:=DWORD_TO_REAL(Strom_roh_Klemme_2_Phase_3)*MUL_Strom;  
0006 Wert_Energieverbrauch_Klemme_2_Phase_3:=DWORD_TO_REAL(Energieverbrauch_roh_Klemme_2_Phase_3)*MUL_Energie;
```

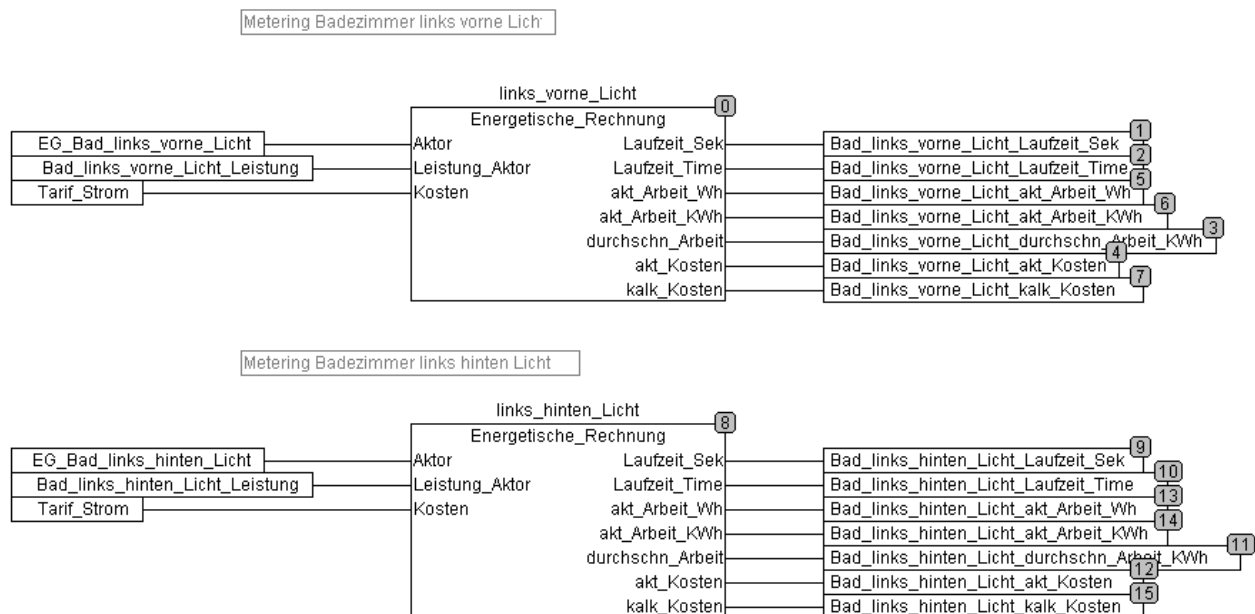
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Badezimmer Metering

Variablen:

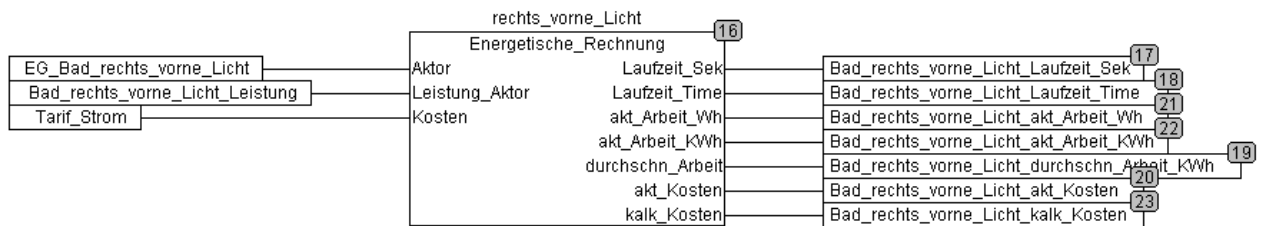
```
0001 PROGRAM Badezimmer_Metering
0002 VAR
0003     links_vorne_Licht: Energetische_Rechnung;
0004     links_hinten_Licht: Energetische_Rechnung;
0005     rechts_vorne_Licht: Energetische_Rechnung;
0006     rechts_hinten_Licht: Energetische_Rechnung;
0007     rechts_mitte_Licht: Energetische_Rechnung;
0008     Heizung: Energetische_Rechnung;
0009     Lampe_230_V: Energetische_Rechnung;
0010 END_VAR
```

Quelltext:

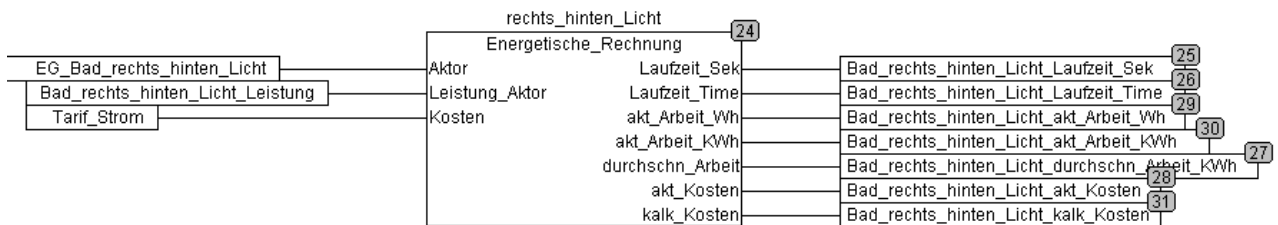


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Metering Badezimmer rechts vorne Licht



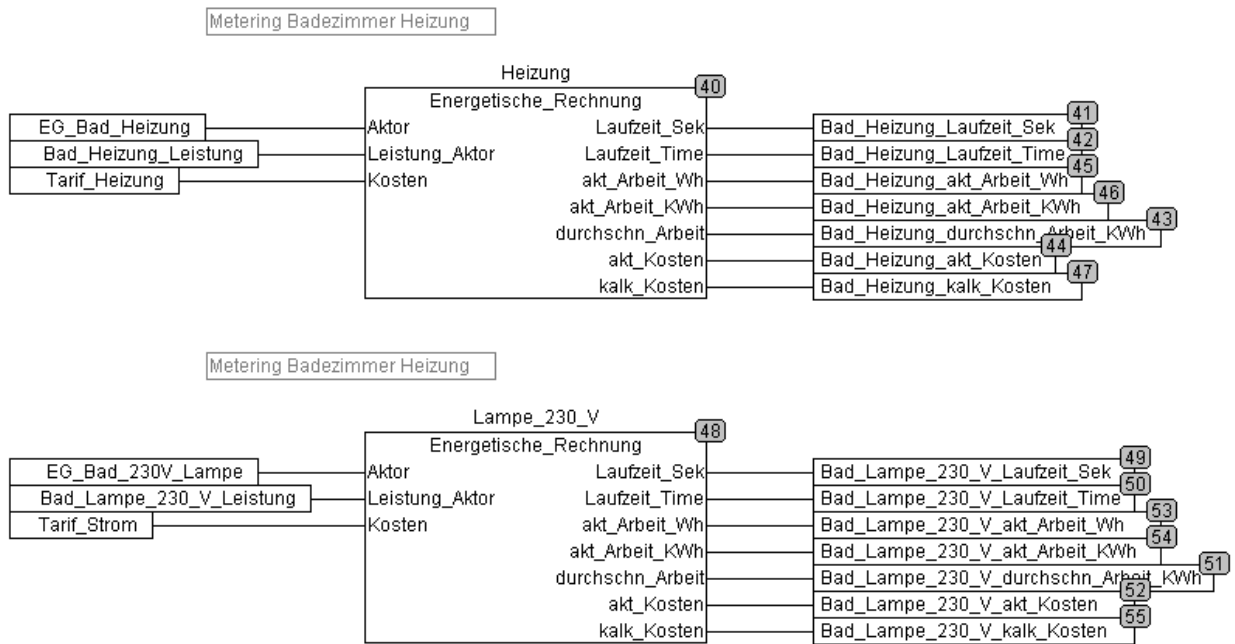
Metering Badezimmer rechts hinten Licht



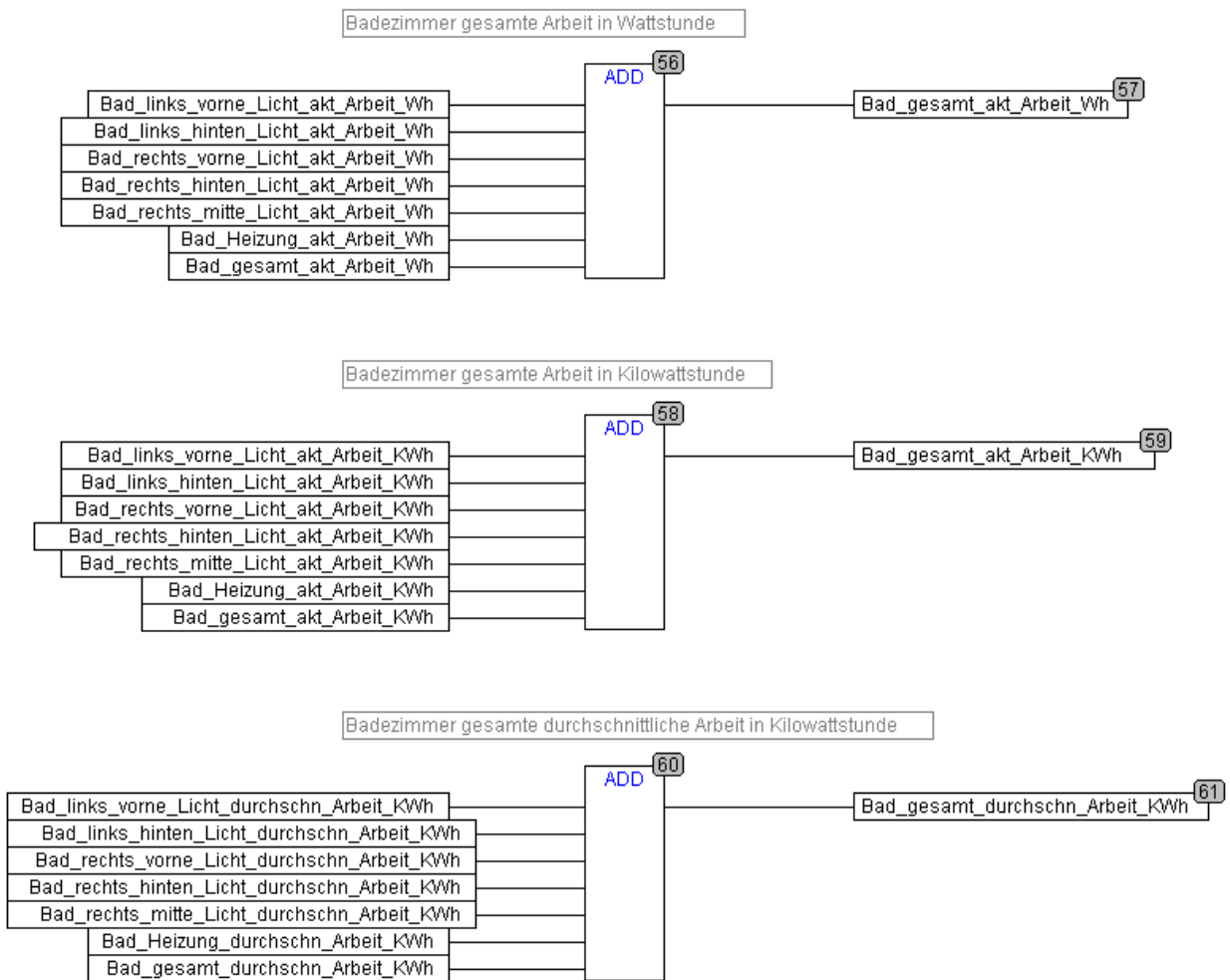
Metering Badezimmer rechts mitte Licht



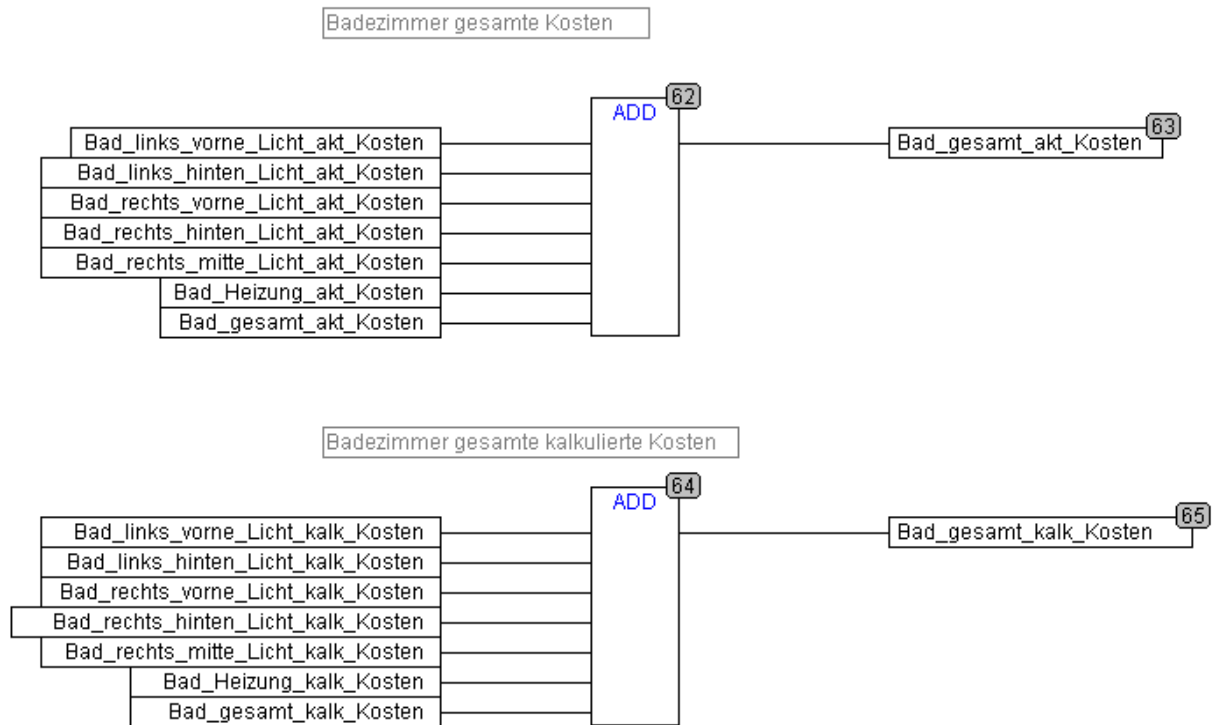
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



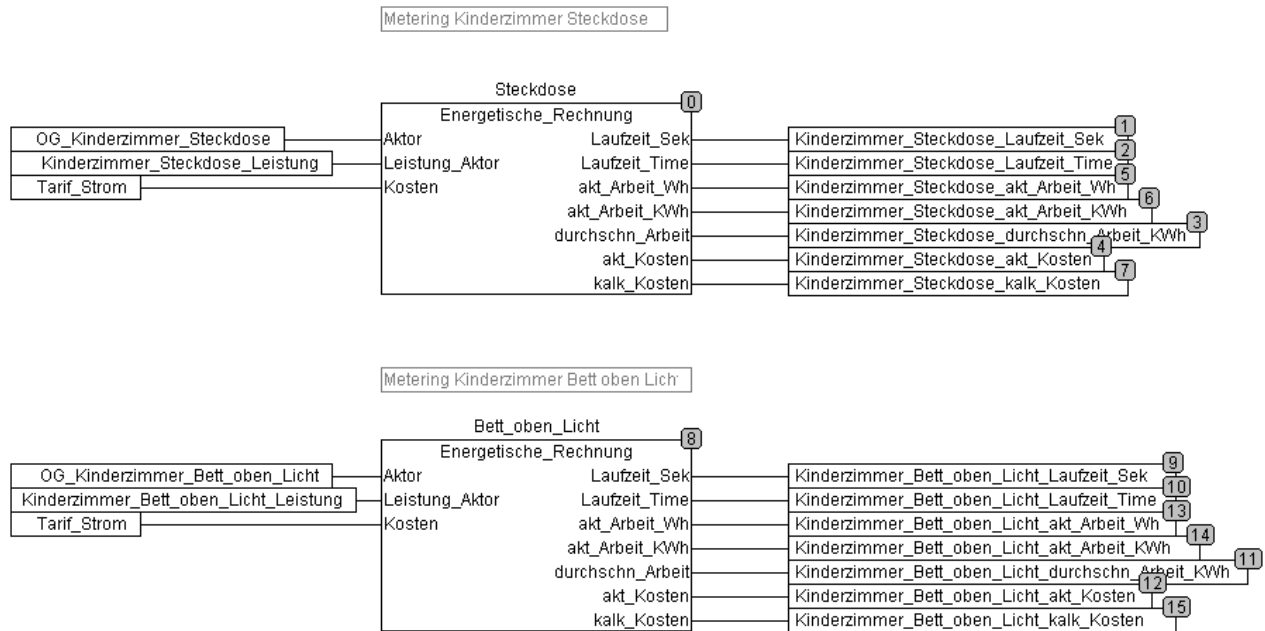
Metering Kinderzimmer:

Variablen:

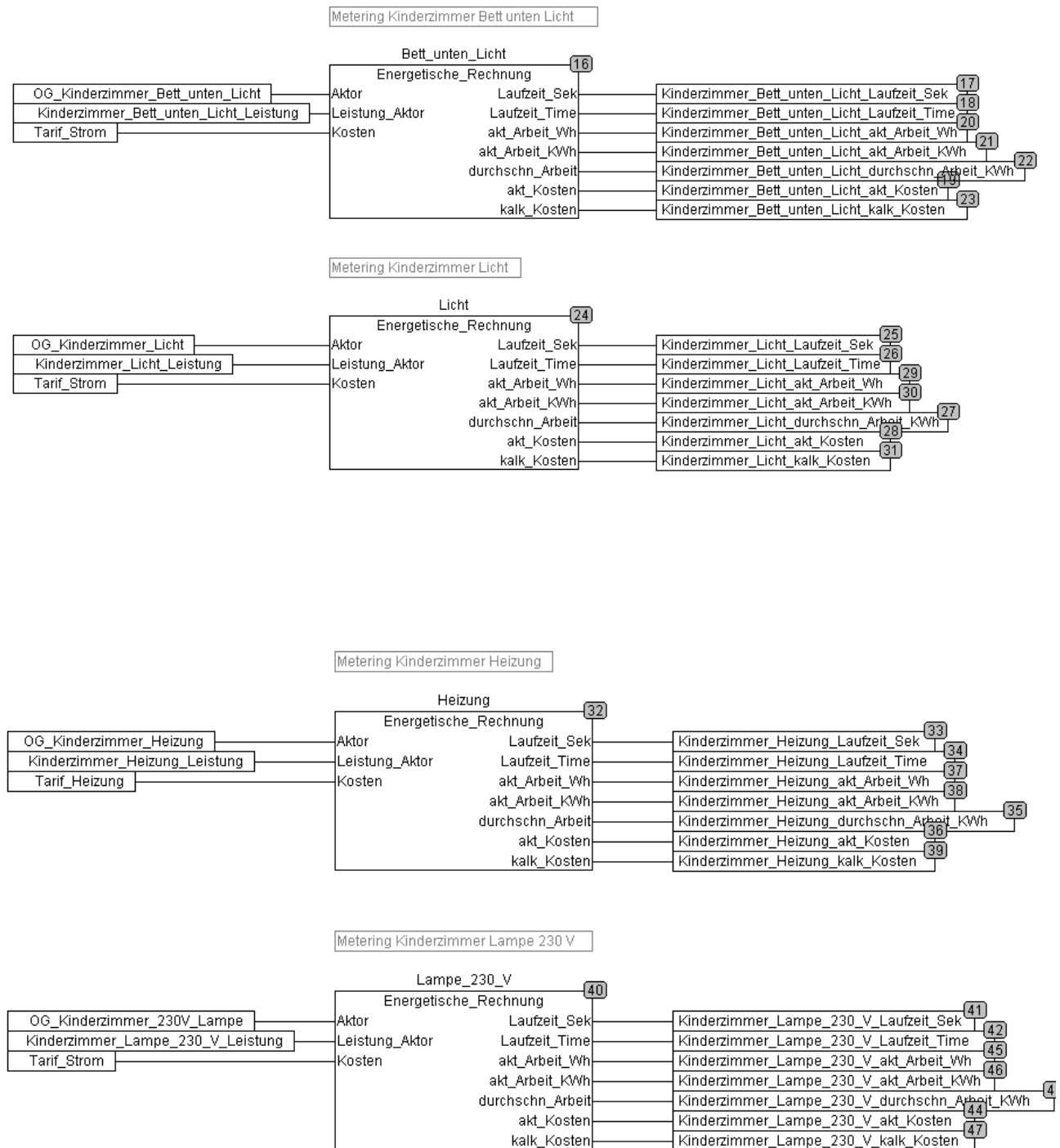
```
0001 PROGRAM Kinderzimmer_Metering
0002 VAR
0003   Steckdose: Energetische_Rechnung;
0004   Bett_oben_Licht: Energetische_Rechnung;
0005   Bett_unten_Licht: Energetische_Rechnung;
0006   Licht: Energetische_Rechnung;
0007   Heizung: Energetische_Rechnung;
0008   Lampe_230_V: Energetische_Rechnung;
0009 END_VAR
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

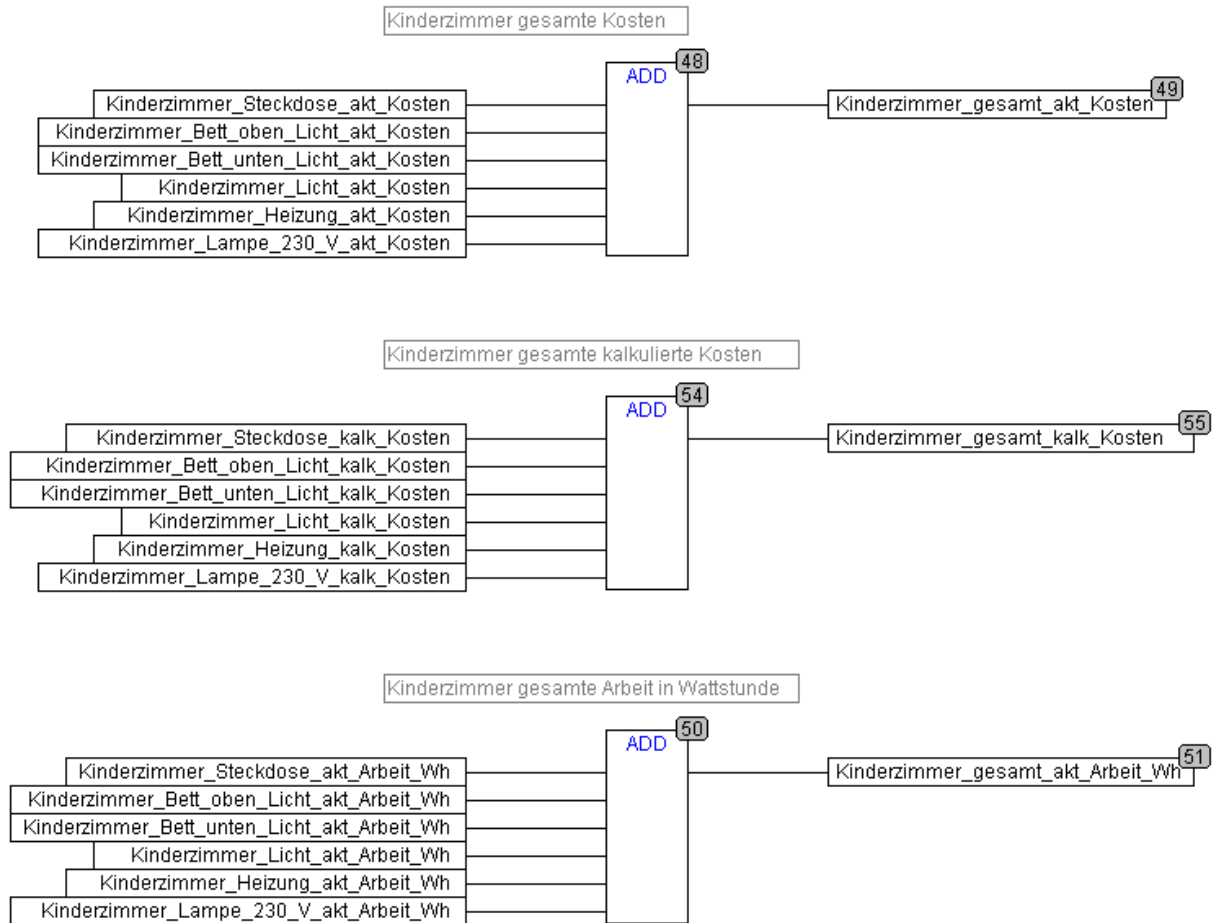
Quelltext:



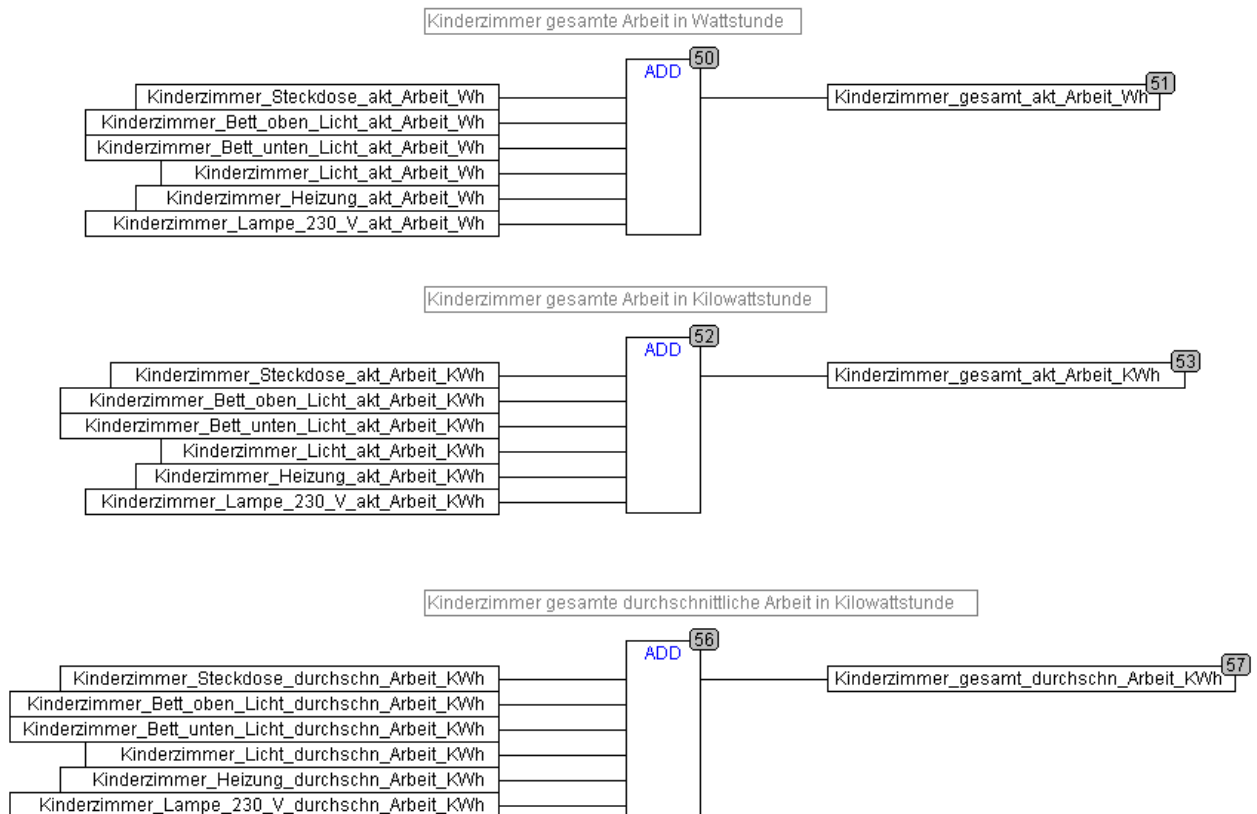
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Metering Küche:

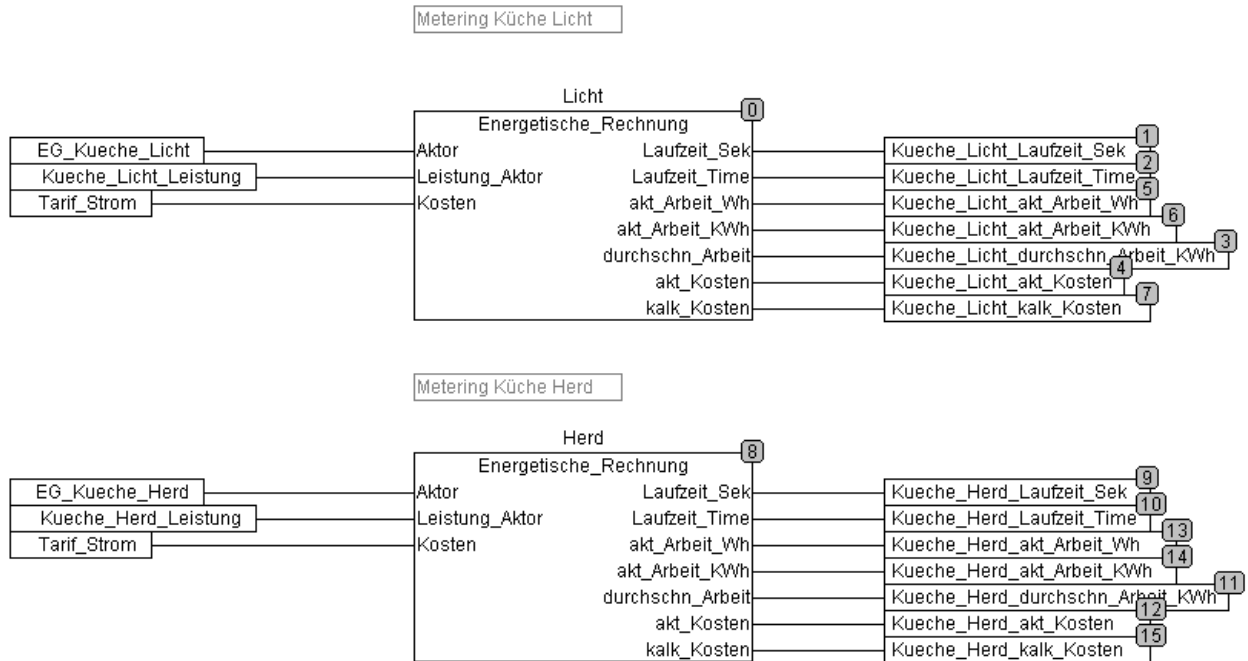
Variablen:

```

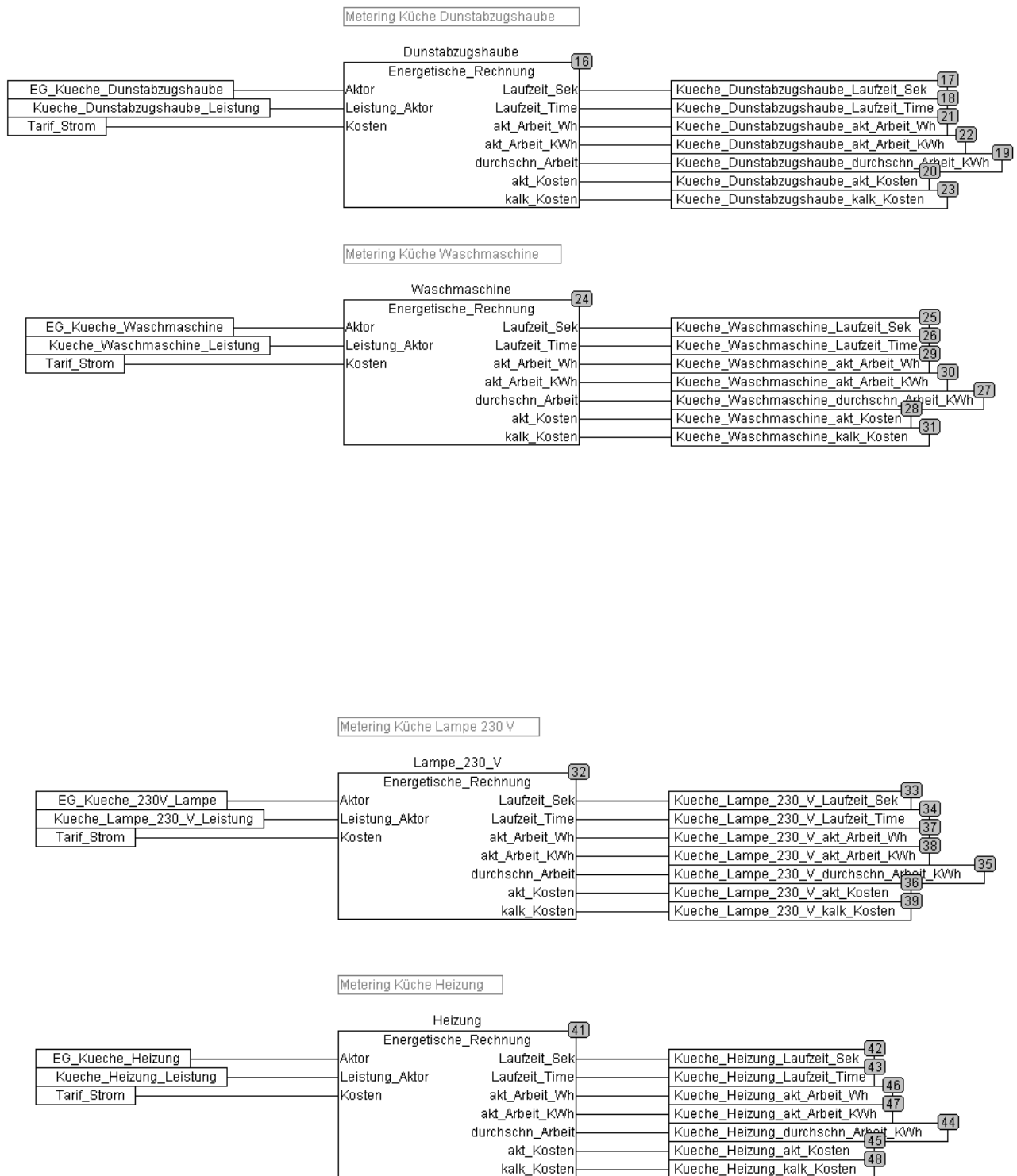
0001 PROGRAM Kueche_Metering
0002 VAR
0003     Licht: Energetische_Rechnung;
0004     Herd: Energetische_Rechnung;
0005     Dunstabzugshaube: Energetische_Rechnung;
0006     Waschmaschine: Energetische_Rechnung;
0007     Lampe_230_V: Energetische_Rechnung;
0008     Heizung: Energetische_Rechnung;
0009 END_VAR
    
```

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

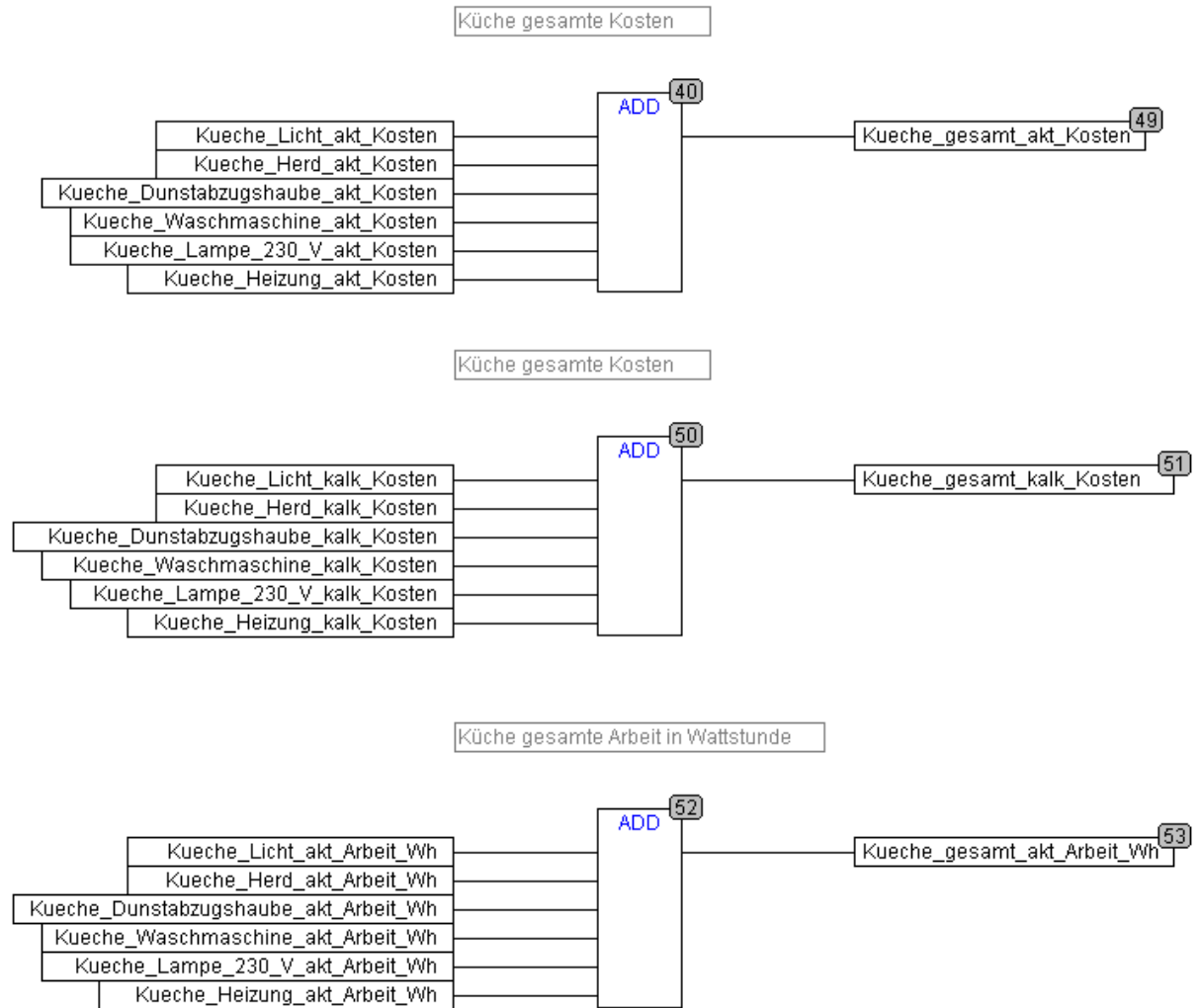
Quelltext:



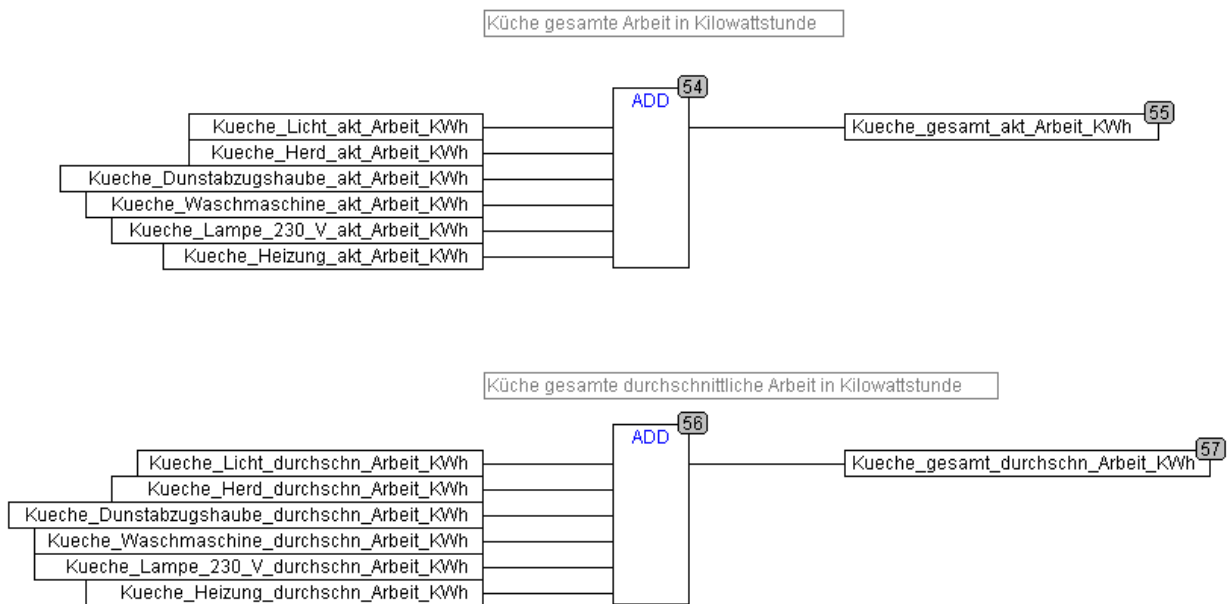
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



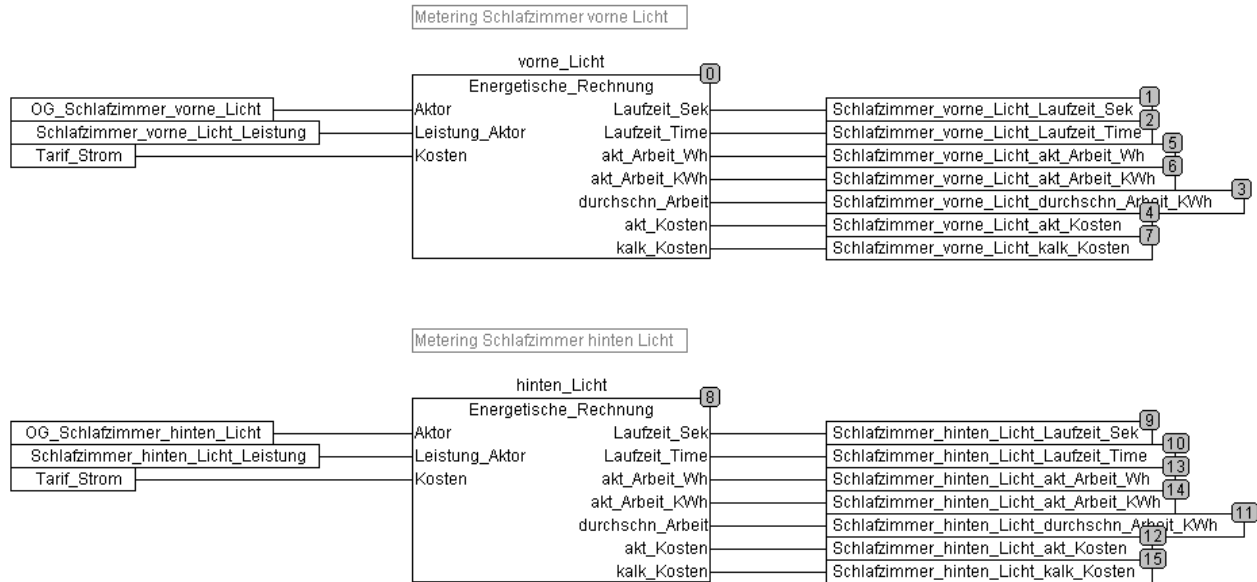
Metering Wohnzimmer:

Variablen:

```
0001 PROGRAM Schlafzimmer_Metering
0002 VAR
0003     vorne_Licht: Energetische_Rechnung;
0004     hinten_Licht: Energetische_Rechnung;
0005     Bett_Licht: Energetische_Rechnung;
0006     Heizung: Energetische_Rechnung;
0007     Lampe_230_V: Energetische_Rechnung;
0008 END_VAR
```

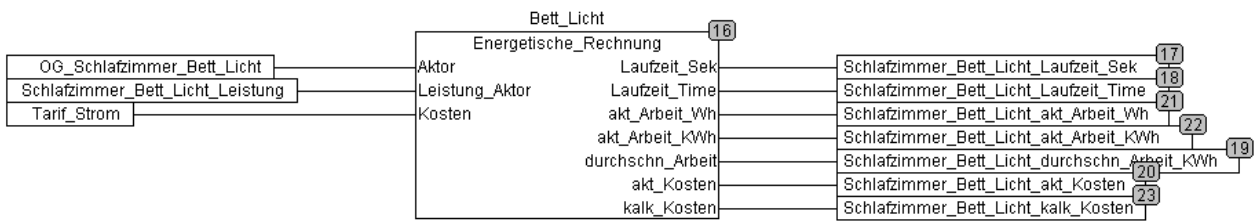
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:

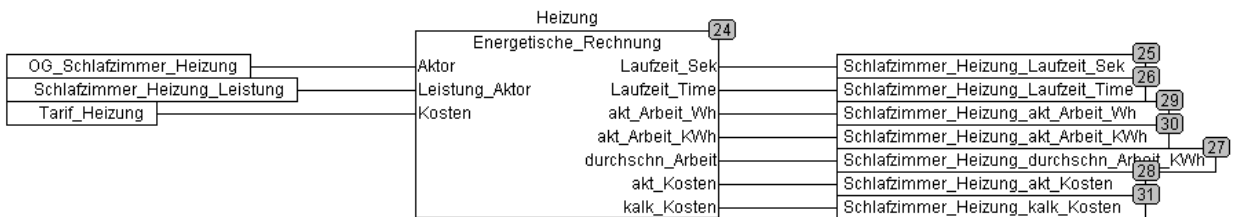


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

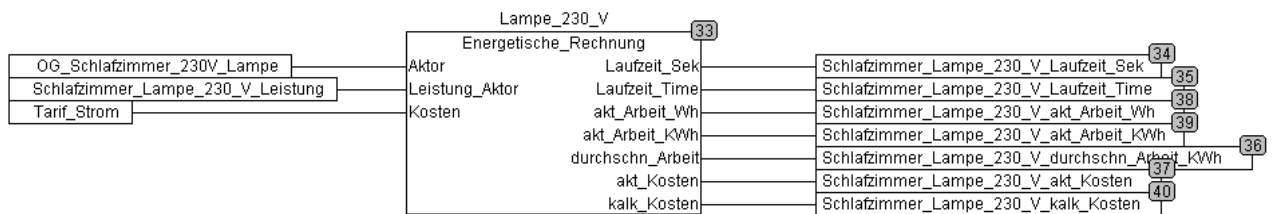
Metering Schlafzimmer Bett Licht



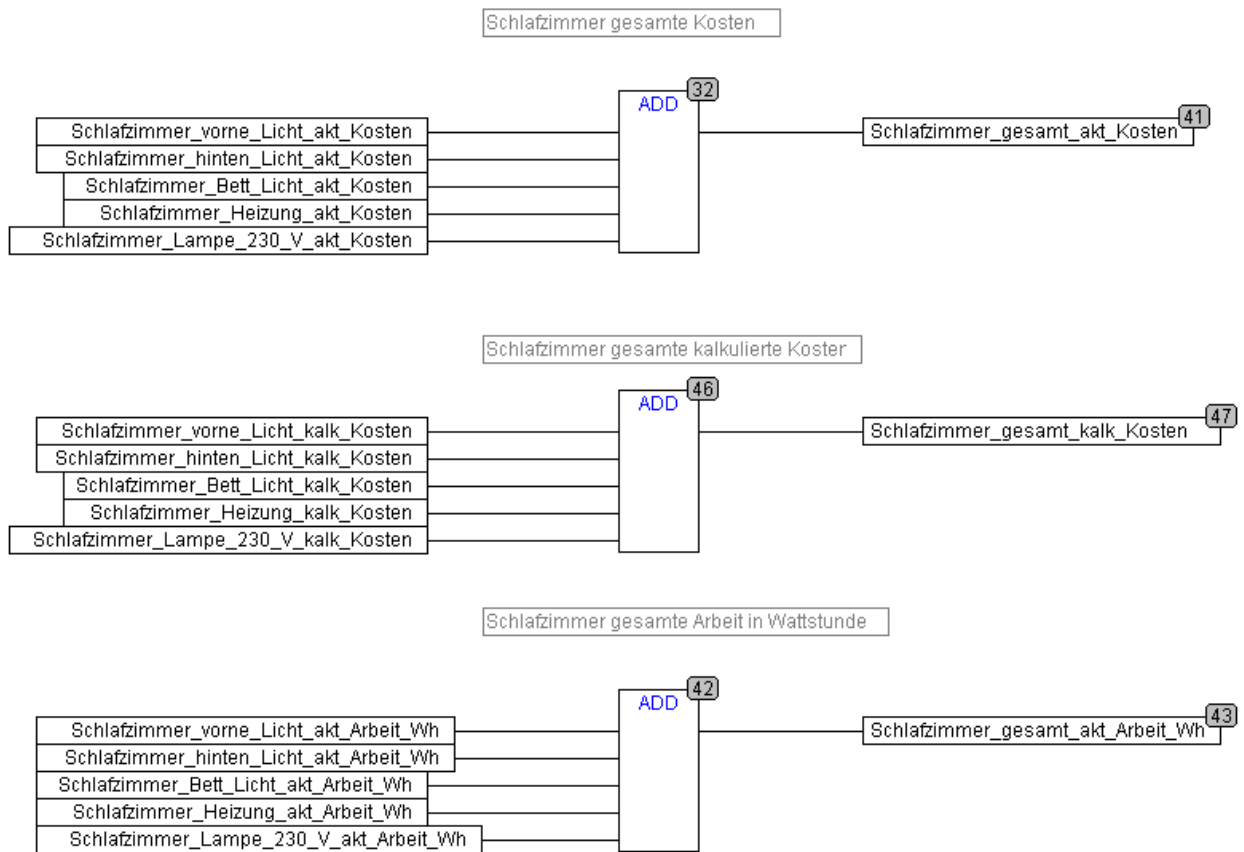
Metering Schlafzimmer Heizung



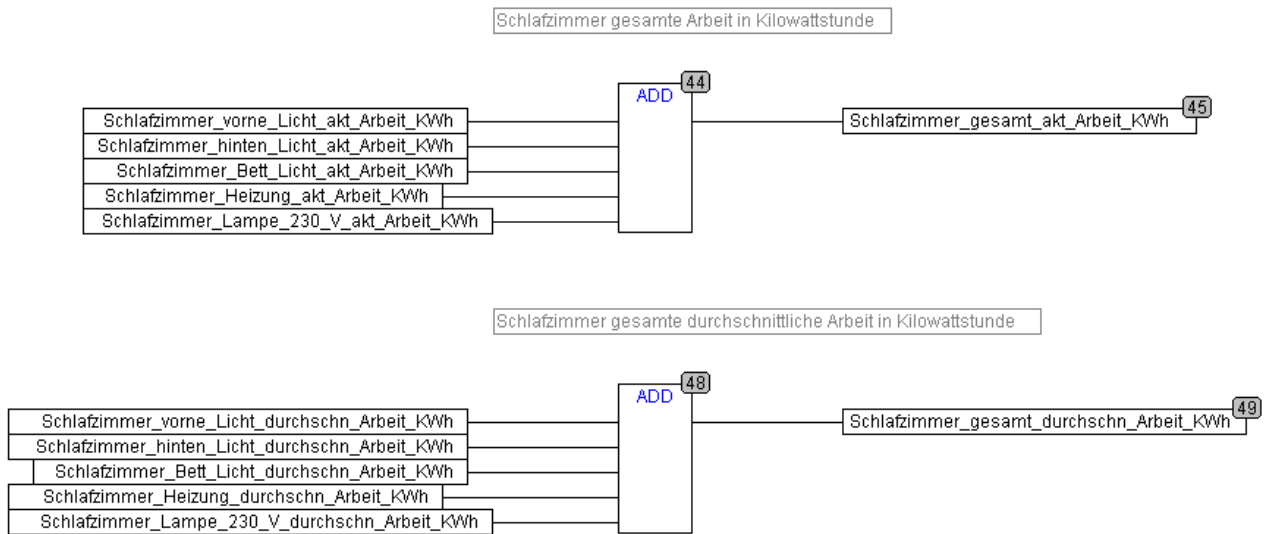
Metering Schlafzimmer Lampe 230 V



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



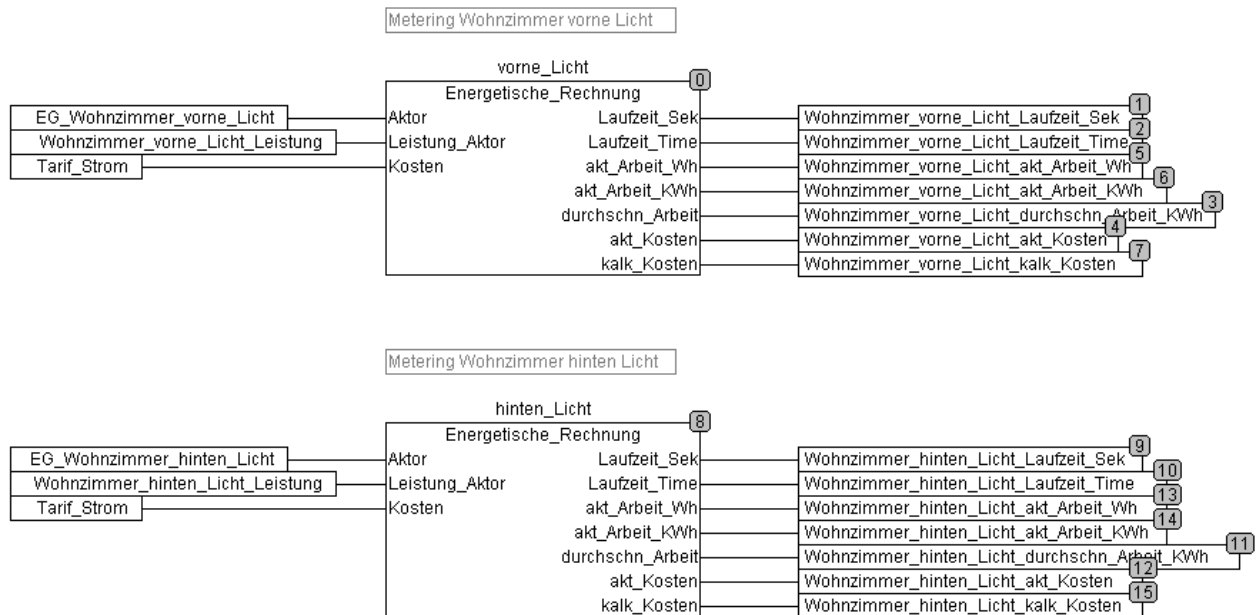
Metering Wohnzimmer:

Variablen:

```
0001 PROGRAM Wohnzimmer_Metering
0002 VAR
0003     vorne_Licht: Energetische_Rechnung;
0004     hinten_Licht: Energetische_Rechnung;
0005     Stehlampe: Energetische_Rechnung;
0006     Heizung: Energetische_Rechnung;
0007     Lampe_230_V: Energetische_Rechnung;
0008 END_VAR
```

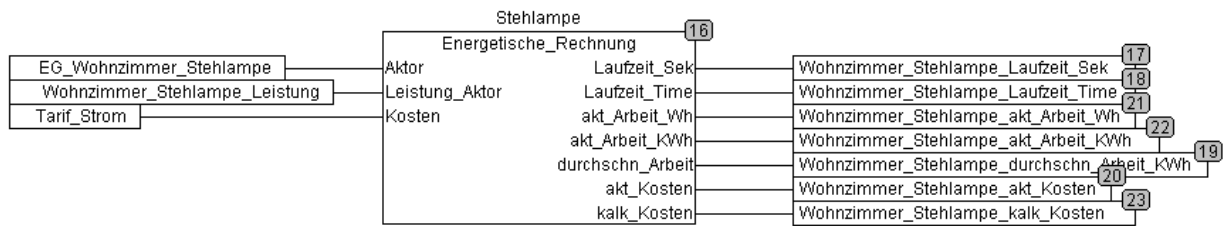
Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Quelltext:

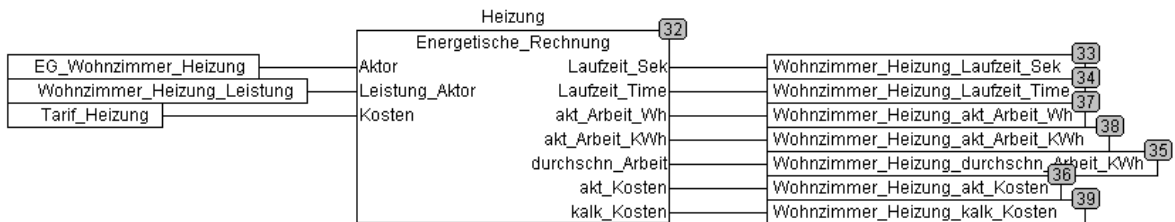


Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

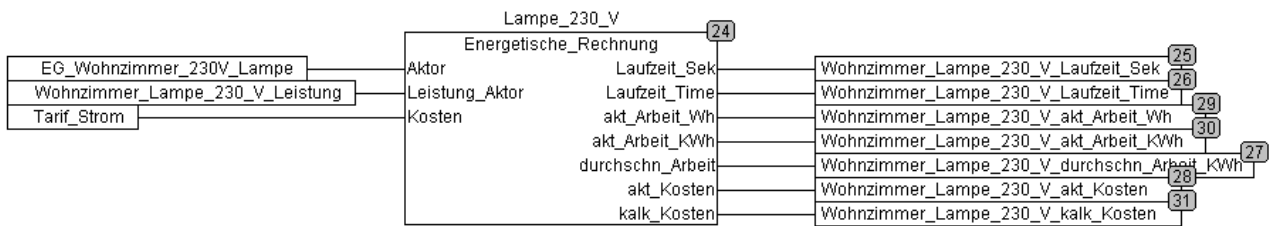
Metering Wohnzimmer Stehlampe



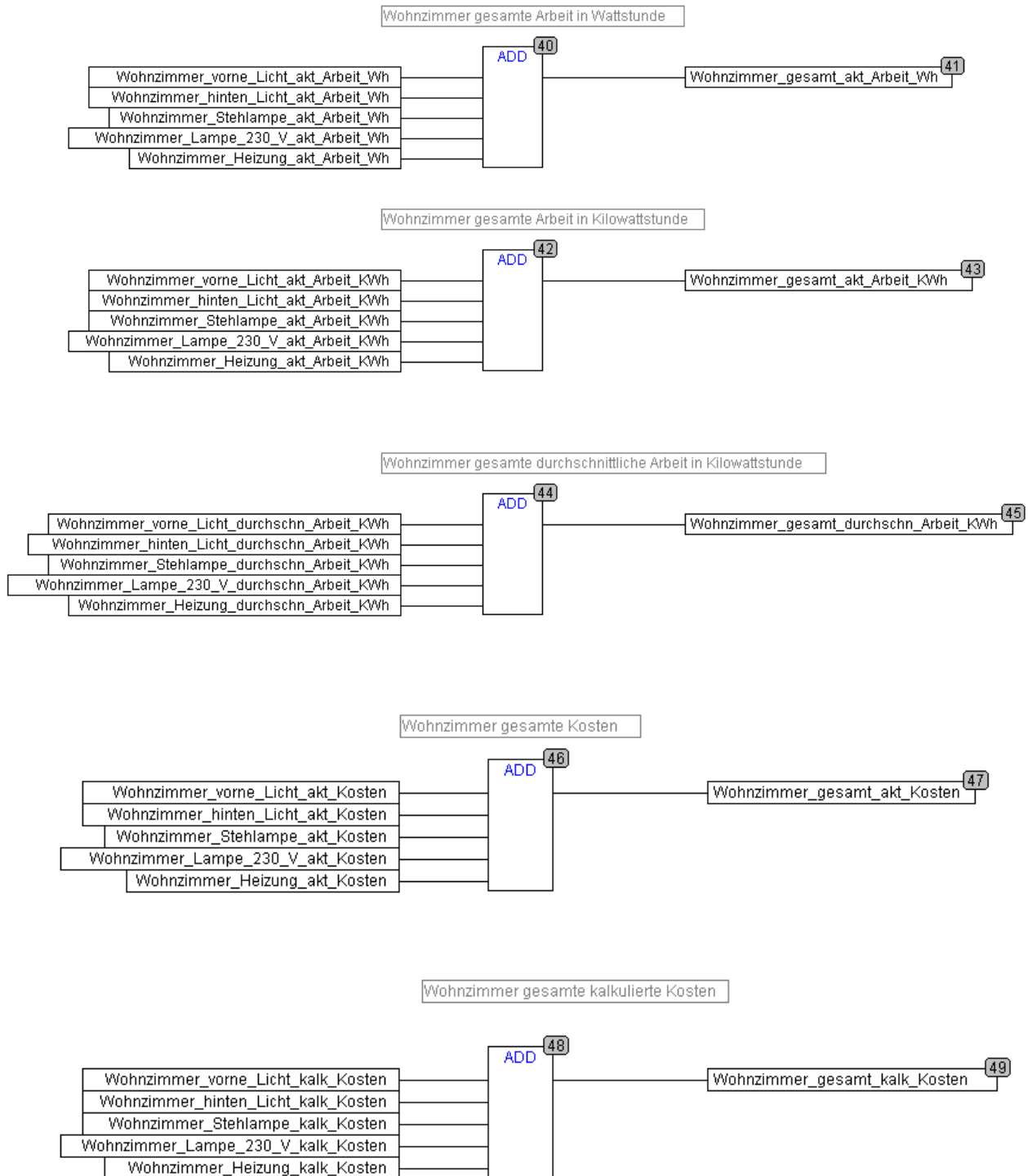
Metering Wohnzimmer Heizung



Metering Wohnzimmer Lampe 230 V



Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering



VAR_GLOBAL

Lichttorgel_ein_aus: BOOL;

(*Anwesenheit*)

Kind1_Anwesend:BOOL;

Kind2_Anwesend:BOOL;

Mutter_Anwesend:BOOL;

Vater_Anwesend:BOOL;

Haus_verlassen:BOOL;

(*Handbetrieb wiederaufnehmen*)

Hand:BOOL;

Haus_verlassen_Hand:BOOL;

(*Helligkeit*)

Helligkeit:USINT;

Helligkeit_aus: BOOL;

Helligkeit_ein: BOOL;

(*Jalousie*)

Jalousie_Links_Position:USINT;

Jalousie_Rechts_Position:USINT;

(*Enocean*)

(*Taster 1*)

Enocean_Taster1_R_up:BOOL;

Enocean_Taster1_R_down:BOOL;

Enocean_Taster1_L_up:BOOL;

Enocean_Taster1_L_down:BOOL;

(*Taster 2*)

Enocean_Taster2_R_up:BOOL;

Enocean_Taster2_R_down:BOOL;

Enocean_Taster2_L_up:BOOL;

Enocean_Taster2_L_down:BOOL;

(*Thermokon Sensor 1*)

Temperatur_Thermokon_Sensor1:REAL;

```
Temperatur_SOLL_1:REAL;
(*Thermokon Sensor 2*)
Temperatur_Thermokon_Sensor2:REAL;
Temperatur_SOLL_2:REAL;
(*Zeit Variablen*)
(*aktuelle Uhrzeit*)
aktZeit:DT;
(*automatisches Einschalten bei Einschaltzeit*)
Einschaltzeit:DWORD;
Kinderzimmer_Steckdose_Zeit_ein:BOOL;
Kinderzimmer_Bett_oben_Licht_Zeit_ein:BOOL;
Kinderzimmer_Bett_unten_Licht_Zeit_ein:BOOL;
Kinderzimmer_Licht_Zeit_ein:BOOL;
Kinderzimmer_Lampe_230_V_ein:BOOL;
Kinderzimmer_Heizung_Zeit_ein:BOOL;
Schlafzimmer_vorne_Licht_Zeit_ein:BOOL;
Schlafzimmer_hinten_Licht_Zeit_ein:BOOL;
Schlafzimmer_Bett_Licht_Zeit_ein:BOOL;
Schlafzimmer_Lampe_230_V_Zeit_ein:BOOL;
Schlafzimmer_Heizung_Zeit_ein:BOOL;
Kueche_Licht_Zeit_ein:BOOL;
Kueche_Waschmaschine_Zeit_ein:BOOL;
Kueche_Herd_Zeit_ein:BOOL;
Kueche_Dunstabzugshaube_Zeit_ein:BOOL;
Kueche_Lampe_230_V_Zeit_ein:BOOL;
Kueche_Heizung_Zeit_ein:BOOL;
Bad_Links_vorne_Licht_Zeit_ein:BOOL;
Bad_Links_hinten_Licht_Zeit_ein:BOOL;
Bad_rechts_vorne_Licht_Zeit_ein:BOOL;
Bad_rechts_hinten_Licht_Zeit_ein:BOOL;
Bad_rechts_mitte_Licht_Zeit_ein:BOOL;
```

Bad_Lampe_230_V_Zeit_ein:BOOL;
Bad_Heizung_Zeit_ein:BOOL;
Wohnzimmer_vorne_Licht_Zeit_ein:BOOL;
Wohnzimmer_hinten_Licht_Zeit_ein:BOOL;
Wohnzimmer_Stehlampe_Zeit_ein:BOOL;
Wohnzimmer_Lampe_230_V_Zeit_ein:BOOL;
Wohnzimmer_Heizung_Zeit_ein:BOOL;
(*automatisches Ausschalten bei Ausschaltzeit*)
Ausschaltzeit:DWORD;
Kinderzimmer_Steckdose_Zeit_aus:BOOL;
Kinderzimmer_Bett_oben_Licht_Zeit_aus:BOOL;
Kinderzimmer_Bett_unten_Licht_Zeit_aus:BOOL;
Kinderzimmer_Licht_Zeit_aus:BOOL;
Kinderzimmer_Lampe_230_V_aus:BOOL;
Kinderzimmer_Heizung_Zeit_aus:BOOL;
Schlafzimmer_vorne_Licht_Zeit_aus:BOOL;
Schlafzimmer_hinten_Licht_Zeit_aus:BOOL;
Schlafzimmer_Bett_Licht_Zeit_aus:BOOL;
Schlafzimmer_Lampe_230_V_Zeit_aus:BOOL;
Schlafzimmer_Heizung_Zeit_aus:BOOL;
Kueche_Licht_Zeit_aus:BOOL;
Kueche_Waschmaschine_Zeit_aus:BOOL;
Kueche_Herd_Zeit_aus:BOOL;
Kueche_Dunstabzugshaube_Zeit_aus:BOOL;
Kueche_Lampe_230_V_Zeit_aus:BOOL;
Kueche_Heizung_Zeit_aus:BOOL;
Bad_Links_vorne_Licht_Zeit_aus:BOOL;
Bad_Links_hinten_Licht_Zeit_aus:BOOL;
Bad_rechts_vorne_Licht_Zeit_aus:BOOL;
Bad_rechts_hinten_Licht_Zeit_aus:BOOL;
Bad_rechts_mitte_Licht_Zeit_aus:BOOL;

Bad_Lampe_230_V_Zeit_aus:BOOL;
Bad_Heizung_Zeit_aus:BOOL;
Wohnzimmer_vorne_Licht_Zeit_aus:BOOL;
Wohnzimmer_hinten_Licht_Zeit_aus:BOOL;
Wohnzimmer_Stehlampe_Zeit_aus:BOOL;
Wohnzimmer_Lampe_230_V_Zeit_aus:BOOL;
Wohnzimmer_Heizung_Zeit_aus:BOOL;

(*Temperatur Variablen*)

Temperatur_Kinderzimmer:REAL;
Temperatur_Schlafzimmer:REAL;
Temperatur_Kueche:REAL;
Temperatur_Badezimmer:REAL;
Temperatur_Wohnzimmer:REAL;
Temperatur_Aussen:REAL;
Frostschutz:REAL;

(*Sicherheit*)

Fenster_offen:BOOL;
Tuer_offen:BOOL;

(*Variablen für Power Measurement*)

Wert_Wirkleistung_Klemme_1_Phase_1: REAL;
Wert_Scheinleistung_Klemme_1_Phase_1: REAL;
Wert_Energieverbrauch_Klemme_1_Phase_1: REAL;
Wert_Spannung_Klemme_1_Phase_1: REAL;
Wert_Strom_Klemme_1_Phase_1: REAL;
cosphi_Klemme_1_Phase_1: REAL;
Wert_Wirkleistung_Klemme_1_Phase_2: REAL;
Wert_Scheinleistung_Klemme_1_Phase_2: REAL;
Wert_Energieverbrauch_Klemme_1_Phase_2: REAL;
Wert_Spannung_Klemme_1_Phase_2: REAL;
Wert_Strom_Klemme_1_Phase_2: REAL;
cosphi_Klemme_1_Phase_2: REAL;

Wert_Wirkleistung_Klemme_1_Phase_3: REAL;
Wert_Scheinleistung_Klemme_1_Phase_3: REAL;
Wert_Energieverbrauch_Klemme_1_Phase_3: REAL;
Wert_Spannung_Klemme_1_Phase_3: REAL;
Wert_Strom_Klemme_1_Phase_3: REAL;
cosphi_Klemme_1_Phase_3: REAL;
Wert_Wirkleistung_Klemme_2_Phase_1: REAL;
Wert_Scheinleistung_Klemme_2_Phase_1: REAL;
Wert_Energieverbrauch_Klemme_2_Phase_1: REAL;
Wert_Spannung_Klemme_2_Phase_1: REAL;
Wert_Strom_Klemme_2_Phase_1: REAL;
cosphi_Klemme_2_Phase_1: REAL;
Wert_Wirkleistung_Klemme_2_Phase_2: REAL;
Wert_Scheinleistung_Klemme_2_Phase_2: REAL;
Wert_Energieverbrauch_Klemme_2_Phase_2: REAL;
Wert_Spannung_Klemme_2_Phase_2: REAL;
Wert_Strom_Klemme_2_Phase_2: REAL;
cosphi_Klemme_2_Phase_2: REAL;
Wert_Wirkleistung_Klemme_2_Phase_3: REAL;
Wert_Scheinleistung_Klemme_2_Phase_3: REAL;
Wert_Energieverbrauch_Klemme_2_Phase_3: REAL;
Wert_Spannung_Klemme_2_Phase_3: REAL;
Wert_Strom_Klemme_2_Phase_3: REAL;
cosphi_Klemme_2_Phase_3: REAL;

(*Lauf- und Taskzeiten*)

Taskzeit_Energie_Rechnung_real:REAL;
Taskzeit_Energie_Rechnung: TIME;
Laufzeit:TIME;
Laufzeit_Tag_real:REAL;

(*Tarife*)

Tarif_Strom:REAL;

Tarif_Heizung:REAL;

END_VAR

VAR_GLOBAL RETAIN

Kinderzimmer_Steckdose_Laufzeit_Sek:DWORD;
Kinderzimmer_Steckdose_Laufzeit_Time:TIME;
Kinderzimmer_Bett_oben_Licht_Laufzeit_Sek:DWORD;
Kinderzimmer_Bett_oben_Licht_Laufzeit_Time:TIME;
Kinderzimmer_Bett_unten_Licht_Laufzeit_Sek:DWORD;
Kinderzimmer_Bett_unten_Licht_Laufzeit_Time:TIME;
Kinderzimmer_Licht_Laufzeit_Sek:DWORD;
Kinderzimmer_Licht_Laufzeit_Time:TIME;
Kinderzimmer_Heizung_Laufzeit_Sek:DWORD;
Kinderzimmer_Heizung_Laufzeit_Time:TIME;
Kinderzimmer_Lampe_230_V_Laufzeit_Sek:DWORD;
Kinderzimmer_Lampe_230_V_Laufzeit_Time:TIME;

Schlafzimmer_vorne_Licht_Laufzeit_Sek:DWORD;
Schlafzimmer_vorne_Licht_Laufzeit_Time:TIME;
Schlafzimmer_hinten_Licht_Laufzeit_Sek:DWORD;
Schlafzimmer_hinten_Licht_Laufzeit_Time:TIME;
Schlafzimmer_Bett_Licht_Laufzeit_Sek:DWORD;
Schlafzimmer_Bett_Licht_Laufzeit_Time:TIME;
Schlafzimmer_Heizung_Laufzeit_Sek:DWORD;
Schlafzimmer_Heizung_Laufzeit_Time:TIME;
Schlafzimmer_Lampe_230_V_Laufzeit_Sek:DWORD;
Schlafzimmer_Lampe_230_V_Laufzeit_Time:TIME;

Kueche_Licht_Laufzeit_Sek: DWORD;
Kueche_Licht_Laufzeit_Time: TIME;

Kueche_Herd_Laufzeit_Sek: DWORD;
Kueche_Herd_Laufzeit_Time: TIME;
Kueche_Dunstabzugshaube_Laufzeit_Sek: DWORD;
Kueche_Dunstabzugshaube_Laufzeit_Time: TIME;
Kueche_Waschmaschine_Laufzeit_Sek: DWORD;
Kueche_Waschmaschine_Laufzeit_Time: TIME;
Kueche_Lampe_230_V_Laufzeit_Sek: DWORD;
Kueche_Lampe_230_V_Laufzeit_Time: TIME;
Kueche_Heizung_Laufzeit_Sek: DWORD;
Kueche_Heizung_Laufzeit_Time: TIME;

Bad_links_vorne_Licht_Laufzeit_Sek: DWORD;
Bad_links_vorne_Licht_Laufzeit_Time: TIME;
Bad_links_hinten_Licht_Laufzeit_Sek: DWORD;
Bad_links_hinten_Licht_Laufzeit_Time: TIME;
Bad_rechts_vorne_Licht_Laufzeit_Sek: DWORD;
Bad_rechts_vorne_Licht_Laufzeit_Time: TIME;
Bad_rechts_hinten_Licht_Laufzeit_Sek: DWORD;
Bad_rechts_hinten_Licht_Laufzeit_Time: TIME;
Bad_rechts_mitte_Licht_Laufzeit_Sek: DWORD;
Bad_rechts_mitte_Licht_Laufzeit_Time: TIME;
Bad_Heizung_Laufzeit_Sek: DWORD;
Bad_Heizung_Laufzeit_Time: TIME;
Bad_Lampe_230_V_Laufzeit_Sek: DWORD;
Bad_Lampe_230_V_Laufzeit_Time: TIME;

Wohnzimmer_vorne_Licht_Laufzeit_Sek: DWORD;
Wohnzimmer_vorne_Licht_Laufzeit_Time: TIME;
Wohnzimmer_hinten_Licht_Laufzeit_Sek: DWORD;
Wohnzimmer_hinten_Licht_Laufzeit_Time: TIME;
Wohnzimmer_Stehlampe_Laufzeit_Sek: DWORD;

Wohnzimmer_Stehlampe_Laufzeit_Time: TIME;
Wohnzimmer_Lampe_230_V_Laufzeit_Sek: DWORD;
Wohnzimmer_Lampe_230_V_Laufzeit_Time: TIME;
Wohnzimmer_Heizung_Laufzeit_Sek: DWORD;
Wohnzimmer_Heizung_Laufzeit_Time: TIME;

Kinderzimmer_Steckdose_akt_Arbeit_Wh:REAL;
Kinderzimmer_Steckdose_akt_Arbeit_KWh:REAL;
Kinderzimmer_Steckdose_durchschn_Arbeit_KWh:REAL;
Kinderzimmer_Bett_oben_Licht_akt_Arbeit_Wh:REAL;
Kinderzimmer_Bett_oben_Licht_akt_Arbeit_KWh:REAL;
Kinderzimmer_Bett_oben_Licht_durchschn_Arbeit_KWh:REAL;
Kinderzimmer_Bett_unten_Licht_akt_Arbeit_Wh:REAL;
Kinderzimmer_Bett_unten_Licht_akt_Arbeit_KWh:REAL;
Kinderzimmer_Bett_unten_Licht_durchschn_Arbeit_KWh:REAL;
Kinderzimmer_Licht_akt_Arbeit_Wh:REAL;
Kinderzimmer_Licht_akt_Arbeit_KWh:REAL;
Kinderzimmer_Licht_durchschn_Arbeit_KWh:REAL;
Kinderzimmer_Heizung_akt_Arbeit_Wh:REAL;
Kinderzimmer_Heizung_akt_Arbeit_KWh:REAL;
Kinderzimmer_Heizung_durchschn_Arbeit_KWh:REAL;
Kinderzimmer_Lampe_230_V_akt_Arbeit_Wh:REAL;
Kinderzimmer_Lampe_230_V_akt_Arbeit_KWh:REAL;
Kinderzimmer_Lampe_230_V_durchschn_Arbeit_KWh:REAL;
Kinderzimmer_gesamt_akt_Arbeit_Wh:REAL;
Kinderzimmer_gesamt_akt_Arbeit_KWh:REAL;
Kinderzimmer_gesamt_durchschn_Arbeit_KWh:REAL;

Schlafzimmer_vorne_Licht_akt_Arbeit_Wh:REAL;

Schlafzimmer_vorne_Licht_akt_Arbeit_KWh:REAL;
Schlafzimmer_vorne_Licht_durchschn_Arbeit_KWh:REAL;
Schlafzimmer_hinten_Licht_akt_Arbeit_Wh:REAL;
Schlafzimmer_hinten_Licht_akt_Arbeit_KWh:REAL;
Schlafzimmer_hinten_Licht_durchschn_Arbeit_KWh:REAL;
Schlafzimmer_Bett_Licht_akt_Arbeit_Wh:REAL;
Schlafzimmer_Bett_Licht_akt_Arbeit_KWh:REAL;
Schlafzimmer_Bett_Licht_durchschn_Arbeit_KWh:REAL;
Schlafzimmer_Heizung_akt_Arbeit_Wh:REAL;
Schlafzimmer_Heizung_akt_Arbeit_KWh:REAL;
Schlafzimmer_Heizung_durchschn_Arbeit_KWh:REAL;
Schlafzimmer_Lampe_230_V_akt_Arbeit_Wh:REAL;
Schlafzimmer_Lampe_230_V_akt_Arbeit_KWh:REAL;
Schlafzimmer_Lampe_230_V_durchschn_Arbeit_KWh:REAL;
Schlafzimmer_gesamt_akt_Arbeit_Wh:REAL;
Schlafzimmer_gesamt_akt_Arbeit_KWh:REAL;
Schlafzimmer_gesamt_durchschn_Arbeit_KWh:REAL;

Kueche_Licht_akt_Arbeit_Wh: REAL;
Kueche_Licht_akt_Arbeit_KWh: REAL;
Kueche_Licht_durchschn_Arbeit_KWh:REAL;
Kueche_Herd_akt_Arbeit_Wh: REAL;
Kueche_Herd_akt_Arbeit_KWh: REAL;
Kueche_Herd_durchschn_Arbeit_KWh:REAL;
Kueche_Dunstabzugshaube_akt_Arbeit_Wh: REAL;
Kueche_Dunstabzugshaube_akt_Arbeit_KWh: REAL;
Kueche_Dunstabzugshaube_durchschn_Arbeit_KWh:REAL;
Kueche_Waschmaschine_akt_Arbeit_Wh: REAL;
Kueche_Waschmaschine_akt_Arbeit_KWh: REAL;
Kueche_Waschmaschine_durchschn_Arbeit_KWh:REAL;
Kueche_Lampe_230_V_akt_Arbeit_Wh: REAL;

Kueche_Lampe_230_V_akt_Arbeit_KWh: REAL;
Kueche_Lampe_230_V_durchschn_Arbeit_KWh:REAL;
Kueche_Heizung_akt_Arbeit_Wh: REAL;
Kueche_Heizung_akt_Arbeit_KWh: REAL;
Kueche_Heizung_durchschn_Arbeit_KWh:REAL;
Kueche_gesamt_akt_Arbeit_Wh: REAL;
Kueche_gesamt_akt_Arbeit_KWh: REAL;
Kueche_gesamt_durchschn_Arbeit_KWh:REAL;

Bad_links_vorne_Licht_akt_Arbeit_Wh: REAL;
Bad_links_vorne_Licht_akt_Arbeit_KWh: REAL;
Bad_links_vorne_Licht_durchschn_Arbeit_KWh:REAL;
Bad_links_hinten_Licht_akt_Arbeit_Wh: REAL;
Bad_links_hinten_Licht_akt_Arbeit_KWh: REAL;
Bad_links_hinten_Licht_durchschn_Arbeit_KWh:REAL;
Bad_rechts_vorne_Licht_akt_Arbeit_Wh: REAL;
Bad_rechts_vorne_Licht_akt_Arbeit_KWh: REAL;
Bad_rechts_vorne_Licht_durchschn_Arbeit_KWh:REAL;
Bad_rechts_hinten_Licht_akt_Arbeit_Wh: REAL;
Bad_rechts_hinten_Licht_akt_Arbeit_KWh: REAL;
Bad_rechts_hinten_Licht_durchschn_Arbeit_KWh:REAL;
Bad_rechts_mitte_Licht_akt_Arbeit_Wh: REAL;
Bad_rechts_mitte_Licht_akt_Arbeit_KWh:REAL;
Bad_rechts_mitte_Licht_durchschn_Arbeit_KWh:REAL;
Bad_Heizung_akt_Arbeit_Wh: REAL;
Bad_Heizung_akt_Arbeit_KWh:REAL;
Bad_Heizung_durchschn_Arbeit_KWh:REAL;
Bad_Lampe_230_V_akt_Arbeit_Wh: REAL;
Bad_Lampe_230_V_akt_Arbeit_KWh: REAL;
Bad_Lampe_230_V_durchschn_Arbeit_KWh:REAL;
Bad_gesamt_akt_Arbeit_Wh: REAL;

Bad_gesamt_akt_Arbeit_KWh: REAL;

Bad_gesamt_durchschn_Arbeit_KWh:REAL;

Wohnzimmer_vorne_Licht_akt_Arbeit_Wh: REAL;

Wohnzimmer_vorne_Licht_akt_Arbeit_KWh: REAL;

Wohnzimmer_vorne_Licht_durchschn_Arbeit_KWh:REAL;

Wohnzimmer_hinten_Licht_akt_Arbeit_Wh: REAL;

Wohnzimmer_hinten_Licht_akt_Arbeit_KWh: REAL;

Wohnzimmer_hinten_Licht_durchschn_Arbeit_KWh:REAL;

Wohnzimmer_Stehlampe_akt_Arbeit_Wh: REAL;

Wohnzimmer_Stehlampe_akt_Arbeit_KWh: REAL;

Wohnzimmer_Stehlampe_durchschn_Arbeit_KWh:REAL;

Wohnzimmer_Lampe_230_V_akt_Arbeit_Wh: REAL;

Wohnzimmer_Lampe_230_V_akt_Arbeit_KWh: REAL;

Wohnzimmer_Lampe_230_V_durchschn_Arbeit_KWh:REAL;

Wohnzimmer_Heizung_akt_Arbeit_Wh: REAL;

Wohnzimmer_Heizung_akt_Arbeit_KWh: REAL;

Wohnzimmer_Heizung_durchschn_Arbeit_KWh:REAL;

Wohnzimmer_gesamt_akt_Arbeit_Wh: REAL;

Wohnzimmer_gesamt_akt_Arbeit_KWh: REAL;

Wohnzimmer_gesamt_durchschn_Arbeit_KWh:REAL;

Kinderzimmer_Steckdose_akt_Kosten:REAL;

Kinderzimmer_Steckdose_kalk_Kosten:REAL;

Kinderzimmer_Bett_oben_Licht_akt_Kosten:REAL;

Kinderzimmer_Bett_oben_Licht_kalk_Kosten:REAL;

Kinderzimmer_Bett_unten_Licht_akt_Kosten:REAL;

Kinderzimmer_Bett_unten_Licht_kalk_Kosten:REAL;

Kinderzimmer_Licht_akt_Kosten:REAL;

Kinderzimmer_Licht_kalk_Kosten:REAL;
Kinderzimmer_Heizung_akt_Kosten:REAL;
Kinderzimmer_Heizung_kalk_Kosten:REAL;
Kinderzimmer_Lampe_230_V_akt_Kosten:REAL;
Kinderzimmer_Lampe_230_V_kalk_Kosten:REAL;
Kinderzimmer_gesamt_akt_Kosten:REAL;
Kinderzimmer_gesamt_kalk_Kosten:REAL;

Schlafzimmer_vorne_Licht_akt_Kosten:REAL;
Schlafzimmer_vorne_Licht_kalk_Kosten:REAL;
Schlafzimmer_hinten_Licht_akt_Kosten:REAL;
Schlafzimmer_hinten_Licht_kalk_Kosten:REAL;
Schlafzimmer_Bett_Licht_akt_Kosten:REAL;
Schlafzimmer_Bett_Licht_kalk_Kosten:REAL;
Schlafzimmer_Heizung_akt_Kosten:REAL;
Schlafzimmer_Heizung_kalk_Kosten:REAL;
Schlafzimmer_Lampe_230_V_akt_Kosten:REAL;
Schlafzimmer_Lampe_230_V_kalk_Kosten:REAL;
Schlafzimmer_gesamt_akt_Kosten:REAL;
Schlafzimmer_gesamt_kalk_Kosten:REAL;

Kueche_Licht_akt_Kosten: REAL;
Kueche_Licht_kalk_Kosten: REAL;
Kueche_Herd_akt_Kosten: REAL;
Kueche_Herd_kalk_Kosten: REAL;
Kueche_Dunstabzugshaube_akt_Kosten: REAL;
Kueche_Dunstabzugshaube_kalk_Kosten: REAL;
Kueche_Waschmaschine_akt_Kosten: REAL;
Kueche_Waschmaschine_kalk_Kosten: REAL;
Kueche_Lampe_230_V_akt_Kosten: REAL;
Kueche_Lampe_230_V_kalk_Kosten: REAL;

Kueche_Heizung_akt_Kosten: REAL;
Kueche_Heizung_kalk_Kosten: REAL;
Kueche_gesamt_akt_Kosten: REAL;
Kueche_gesamt_kalk_Kosten: REAL;

Bad_links_vorne_Licht_akt_Kosten: REAL;
Bad_links_vorne_Licht_kalk_Kosten: REAL;
Bad_links_hinten_Licht_akt_Kosten: REAL;
Bad_links_hinten_Licht_kalk_Kosten: REAL;
Bad_rechts_vorne_Licht_akt_Kosten: REAL;
Bad_rechts_vorne_Licht_kalk_Kosten: REAL;
Bad_rechts_hinten_Licht_akt_Kosten: REAL;
Bad_rechts_hinten_Licht_kalk_Kosten: REAL;
Bad_rechts_mitte_Licht_akt_Kosten: REAL;
Bad_rechts_mitte_Licht_kalk_Kosten: REAL;
Bad_Heizung_akt_Kosten: REAL;
Bad_Heizung_kalk_Kosten: REAL;
Bad_Lampe_230_V_akt_Kosten: REAL;
Bad_Lampe_230_V_kalk_Kosten: REAL;
Bad_gesamt_akt_Kosten: REAL;
Bad_gesamt_kalk_Kosten: REAL;

Wohnzimmer_vorne_Licht_akt_Kosten: REAL;
Wohnzimmer_vorne_Licht_kalk_Kosten: REAL;
Wohnzimmer_hinten_Licht_akt_Kosten: REAL;
Wohnzimmer_hinten_Licht_kalk_Kosten: REAL;
Wohnzimmer_Stehlampe_akt_Kosten: REAL;
Wohnzimmer_Stehlampe_kalk_Kosten: REAL;
Wohnzimmer_Lampe_230_V_akt_Kosten: REAL;
Wohnzimmer_Lampe_230_V_kalk_Kosten: REAL;
Wohnzimmer_Heizung_akt_Kosten: REAL;

Wohnzimmer_Heizung_kalk_Kosten: REAL;

Wohnzimmer_gesamt_akt_Kosten: REAL;

Wohnzimmer_gesamt_kalk_Kosten: REAL;

Kinderzimmer_Steckdose_Leistung:INT;

Kinderzimmer_Bett_oben_Licht_Leistung:INT;

Kinderzimmer_Bett_unten_Licht_Leistung:INT;

Kinderzimmer_Licht_Leistung:INT;

Kinderzimmer_Heizung_Leistung:INT;

Kinderzimmer_Lampe_230_V_Leistung:INT;

Schlafzimmer_vorne_Licht_Leistung:INT;

Schlafzimmer_hinten_Licht_Leistung:INT;

Schlafzimmer_Bett_Licht_Leistung:INT;

Schlafzimmer_Heizung_Leistung:INT;

Schlafzimmer_Lampe_230_V_Leistung:INT;

Kueche_Licht_Leistung:INT;

Kueche_Herd_Leistung: INT;

Kueche_Dunstabzugshaube_Leistung: INT;

Kueche_Waschmaschine_Leistung: INT;

Kueche_Lampe_230_V_Leistung: INT;

Kueche_Heizung_Leistung: INT;

Bad_links_vorne_Licht_Leistung: INT;

Bad_links_hinten_Licht_Leistung: INT;

Bad_rechts_hinten_Licht_Leistung: INT;

Bad_rechts_vorne_Licht_Leistung: INT;

Bad_rechts_mitte_Licht_Leistung: INT;

Bad_Heizung_Leistung:INT;

Bad_Lampe_230_V_Leistung: INT;

Wohnzimmer_vorne_Licht_Leistung: INT;

Wohnzimmer_hinten_Licht_Leistung: INT;

Wohnzimmer_Stehlampe_Leistung: INT;

Wohnzimmer_Lampe_230_V_Leistung: INT;

Wohnzimmer_Heizung_Leistung: INT;

END_VAR

13.5 Belegung der Klemmen:

*Hardware configuration (Id.: 2010000001)

Knotennummer: -1
Eingabeadresse: %IB0
Ausgabeadresse: %QB0
Diagnoseadresse: %MB0
Download: 1
AutoAdr: 1

*K-Bus[FIX] (Id.: 11994)

Knotennummer: 0
Eingabeadresse: %IB0
Ausgabeadresse: %QB0
Diagnoseadresse: %MB0
Download: 1
AutoAdr: 1

*0750-0424 2 DI 24V DC Intruder Detection[VAR] (Id.: 2000025002)

Knotennummer: 0
Eingabeadresse: %IB0
Ausgabeadresse: %QB0
Diagnoseadresse: %MB0
Download: 1
AutoAdr: 1
Parameter:
PAAssignment: PLC

Kanäle:

AT %IX35.0: BOOL; (* Ch_1 Digital Input *) [CHANNEL (I)]
AT %IX35.1: BOOL; (* Ch_1 diagnostic *) [CHANNEL (I)]
AT %IX35.2: BOOL; (* Ch_2 Digital Input *) [CHANNEL (I)]
AT %IX35.3: BOOL; (* Ch_2 diagnostic *) [CHANNEL (I)]
*0750-0640 RTC Modul[VAR] (Id.: 2000043001)

Knotennummer: 1
Eingabeadresse: %IB1
Ausgabeadresse: %QB1
Diagnoseadresse: %MB0
Download: 1
AutoAdr: 1
Parameter:
PAAssignment: PLC

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Kanäle:

RTC_Eingang_S AT %IX0.0: BOOL; (* Transmit-Acknowledge (Toggle Flag) *) [CHANNEL (I)]
RTC_IRQ AT %IX0.1: BOOL; (* IRQ *) [CHANNEL (I)]
RTC_Error AT %IX0.6: BOOL; (* Error *) [CHANNEL (I)]
RTC_Eingang_ID AT %IB1: BYTE; (* Response Opcode *) [CHANNEL (I)]
RTC_Eingang_D0 AT %IB2: BYTE; (* Input data byte 0 *) [CHANNEL (I)]
RTC_Eingang_D1 AT %IB3: BYTE; (* Input data byte 1 *) [CHANNEL (I)]
RTC_Eingang_D2 AT %IB4: BYTE; (* Input data byte 2 *) [CHANNEL (I)]
RTC_Eingang_D3 AT %IB5: BYTE; (* Input data byte 3 *) [CHANNEL (I)]
RTC_Ausgang_C AT %QX0.0: BOOL; (* Transmit-Request (Toggle Flag) *) [CHANNEL (Q)]
RTC_Ausgang_ID AT %QB1: BYTE; (* Request Opcode *) [CHANNEL (Q)]
RTC_Ausgang_D0 AT %QB2: BYTE; (* Output data byte 0 *) [CHANNEL (Q)]
RTC_Ausgang_D1 AT %QB3: BYTE; (* Output data byte 1 *) [CHANNEL (Q)]
RTC_Ausgang_D2 AT %QB4: BYTE; (* Output data byte 2 *) [CHANNEL (Q)]
RTC_Ausgang_D3 AT %QB5: BYTE; (* Output data byte 3 *) [CHANNEL (Q)]
*0750-0531 4 DO 24V DC 0.5A[VAR] (Id.: 2000010004)

Knotennummer: 2

Eingabeadresse: %IB2

Ausgabeadresse: %QB2

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

EG_Alarm AT %QX35.0: BOOL; (* Klingel (Erdgeschoss) *) [CHANNEL (Q)]
LED_vorne_Anwesenheit_Mutter AT %QX35.1: BOOL; (* Ch_2 Digital output *) [CHANNEL (Q)]
LED_vorne_Anwesenheit_Vater AT %QX35.2: BOOL; (* Ch_3 Digital output *) [CHANNEL (Q)]
LED_vorne_Rot AT %QX35.3: BOOL; (* Ch_4 Digital output *) [CHANNEL (Q)]
*0750-0530 8 DO 24 V DC 0.5A[VAR] (Id.: 2000010008)

Knotennummer: 3

Eingabeadresse: %IB3

Ausgabeadresse: %QB3

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

OG_Kinderzimmer_Steckdose AT %QX35.4: BOOL; (* Kinderzimmer Steckdose (Obergeschoss) *) [CHANNEL (Q)]
OG_Kinderzimmer_Bett_oben_Licht AT %QX35.5: BOOL; (* Kinderzimmer Bett oben Licht (Obergeschoss) *)
[CHANNEL (Q)]
OG_Kinderzimmer_Bett_unten_Licht AT %QX35.6: BOOL; (* Kinderzimmer Bett unten Licht (Obergeschoss) *)
[CHANNEL (Q)]
OG_Kinderzimmer_Licht AT %QX35.7: BOOL; (* Kinderzimmer Licht (Obergeschoss) *) [CHANNEL (Q)]
OG_Kinderzimmer_Heizung AT %QX35.8: BOOL; (* Kinderzimmer Heizung (Obergeschoss) *) [CHANNEL (Q)]
OG_Schlafzimmer_vorne_Licht AT %QX35.9: BOOL; (* Schlafzimmer vorne Licht (Obergeschoss) *) [CHANNEL (Q)]
OG_Schlafzimmer_hinten_Licht AT %QX35.10: BOOL; (* Schlafzimmer hinten Licht (Obergeschoss) *) [CHANNEL (Q)]
OG_Schlafzimmer_Bett_Licht AT %QX35.11: BOOL; (* Schlafzimmer Bett Licht (Obergeschoss) *) [CHANNEL (Q)]
*0750-0530 8 DO 24 V DC 0.5A[VAR] (Id.: 2000010008)

Knotennummer: 4

Eingabeadresse: %IB4

Ausgabeadresse: %QB4

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Parameter:

PAAssignment: PLC

Kanäle:

OG_Schlafzimmer_Heizung AT %QX35.12: BOOL; (* Schlafzimmer Heizung (Obergeschoss) *) [CHANNEL (Q)]

EG_Kueche_Herd AT %QX35.13: BOOL; (* Küche Herd (Erdgeschoss) *) [CHANNEL (Q)]

EG_Kueche_Dunstabzugshaube AT %QX35.14: BOOL; (* Küche Dunstabzugshaube (Erdgeschoss) *) [CHANNEL (Q)]

EG_Kueche_Licht AT %QX35.15: BOOL; (* Küche Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Kueche_Waschmaschine AT %QX36.0: BOOL; (* Küche Waschmaschine (Erdgeschoss) *) [CHANNEL (Q)]

EG_Bad_links_vorne_Licht AT %QX36.1: BOOL; (* Badezimmer links vorne Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Bad_links_hinten_Licht AT %QX36.2: BOOL; (* Badezimmer links hinten Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Kueche_Heizung AT %QX36.3: BOOL; (* Küche Heizung (Erdgeschoss) *) [CHANNEL (Q)]

*0750-0530 8 DO 24 V DC 0.5A[VAR] (Id.: 2000010008)

Knotennummer: 5

Eingabeadresse: %IB5

Ausgabeadresse: %QB5

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

EG_Bad_Heizung AT %QX36.4: BOOL; (* Badezimmer Heizung (Erdgeschoss) *) [CHANNEL (Q)]

EG_Bad_rechts_hinten_Licht AT %QX36.5: BOOL; (* Badezimmer rechts hinten Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Bad_rechts_vorne_Licht AT %QX36.6: BOOL; (* Badezimmer rechts vorne Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Bad_rechts_mitte_Licht AT %QX36.7: BOOL; (* Badezimmer rechts mitte Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Wohnzimmer_hinten_Licht AT %QX36.8: BOOL; (* Wohnzimmer hinten Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Wohnzimmer_vorne_Licht AT %QX36.9: BOOL; (* Wohnzimmer vorne Licht (Erdgeschoss) *) [CHANNEL (Q)]

EG_Wohnzimmer_Stehlampe AT %QX36.10: BOOL; (* Wohnzimmer Stehlampe (Erdgeschoss) *) [CHANNEL (Q)]

EG_Wohnzimmer_Heizung AT %QX36.11: BOOL; (* Wohnzimmer Heizung (Erdgeschoss) *) [CHANNEL (Q)]

*0750-0430 8 DI 24 V DC 3.0ms[VAR] (Id.: 2000001008)

Knotennummer: 6

Eingabeadresse: %IB6

Ausgabeadresse: %QB6

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

OG_Kinderzimmer_links_Taster AT %IX35.4: BOOL; (* Kinderzimmer Linker Taster (Obergeschoss) *) [CHANNEL (I)]

OG_Kinderzimmer_Bett_oben_Taster AT %IX35.5: BOOL; (* Kinderzimmer Bett oben Taster (Obergeschoss) *)

[CHANNEL (I)]

OG_Kinderzimmer_Bett_unten_Taster AT %IX35.6: BOOL; (* Kinderzimmer Bett unten Taster (Obergeschoss) *)

[CHANNEL (I)]

OG_Kinderzimmer_rechts_Taster AT %IX35.7: BOOL; (* Kinderzimmer rechter Taster (Obergeschoss) *) [CHANNEL (I)]

OG_Schlafzimmer_links_Taster AT %IX35.8: BOOL; (* Schlafzimmer linker Taster (Obergeschoss) *) [CHANNEL (I)]

OG_Schlafzimmer_Bett_r_Taster AT %IX35.9: BOOL; (* Schlafzimmer Bett Taster rechts (Obergeschoss) *) [CHANNEL

(I)]

OG_Schlafzimmer_Bett_l_Taster AT %IX35.10: BOOL; (* Schlafzimmer Bett Taster links (Obergeschoss) *) [CHANNEL

(I)]

EG_Kueche_links_Taster AT %IX35.11: BOOL; (* Küche linker Taster (Erdgeschoss) *) [CHANNEL (I)]

*0750-0430 8 DI 24 V DC 3.0ms[VAR] (Id.: 2000001008)

Knotennummer: 7

Eingabeadresse: %IB7

Ausgabeadresse: %QB7

Diagnoseadresse: %MB0

Download: 1

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

EG_Kueche_rechts_Taster AT %IX35.12: BOOL; (* Küche rechter Taster (Erdgeschoss) *) [CHANNEL (I)]

EG_Bad_links_Taster AT %IX35.13: BOOL; (* Badezimmer linker Taster (Erdgeschoss) *) [CHANNEL (I)]

EG_Bad_rechts_Taster AT %IX35.14: BOOL; (* Badezimmer rechter Taster (Erdgeschoss) *) [CHANNEL (I)]

EG_Wohnzimmer_Taster AT %IX35.15: BOOL; (* Taster Wohnzimmer (Erdgeschoss) *) [CHANNEL (I)]

EG_Jalousie_Links_auf AT %IX36.0: BOOL; (* Taster linke Jalousie hoch fahren (Erdgeschoss) *) [CHANNEL (I)]

EG_Jalousie_Links_ab AT %IX36.1: BOOL; (* Taster linke Jalousie runter fahren (Erdgeschoss) *) [CHANNEL (I)]

EG_Jalousie_rechts_auf AT %IX36.2: BOOL; (* Taster rechte Jalousie hoch fahren (Erdgeschoss) *) [CHANNEL (I)]

EG_Jalousie_rechts_ab AT %IX36.3: BOOL; (* Taster rechte Jalousie runter fahren (Erdgeschoss) *) [CHANNEL (I)]

*0750-0430 8 DI 24 V DC 3.0ms[VAR] (Id.: 2000001008)

Knotennummer: 8

Eingabeadresse: %IB8

Ausgabeadresse: %QB8

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

Taster_vorne_Freigabe_Kochfeld AT %IX36.4: BOOL; (* Ch_1 Digital input *) [CHANNEL (I)]

Taster_vorne_Freigabe_Herd AT %IX36.5: BOOL; (* Ch_2 Digital input *) [CHANNEL (I)]

Taster_vorne_Freigabe_Waschmaschine AT %IX36.6: BOOL; (* Ch_3 Digital input *) [CHANNEL (I)]

Taster_vorne_Anwesenheit_Vater AT %IX36.7: BOOL; (* Vater ist Anwesend *) [CHANNEL (I)]

Taster_vorne_Anwesenheit_Mutter AT %IX36.8: BOOL; (* Mutter ist Anwesend *) [CHANNEL (I)]

Taster_vorne_Anwesenheit_Kind1 AT %IX36.9: BOOL; (* Kind 1 ist Anwesend *) [CHANNEL (I)]

Taster_vorne_Anwesenheit_Kind2 AT %IX36.10: BOOL; (* Kind 2 ist Anwesend *) [CHANNEL (I)]

Taster_vorne_Licht_Kueche AT %IX36.11: BOOL; (* Ch_8 Digital input *) [CHANNEL (I)]

*0750-0430 8 DI 24 V DC 3.0ms[VAR] (Id.: 2000001008)

Knotennummer: 9

Eingabeadresse: %IB9

Ausgabeadresse: %QB9

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

Taster_vorne_Licht_Wohnzimmer AT %IX36.12: BOOL; (* Ch_1 Digital input *) [CHANNEL (I)]

EG_Fensterkontakt_Links AT %IX36.13: BOOL; (* Fensterkontakt Linkes Fenster (Erdgeschoss) *) [CHANNEL (I)]

EG_Fensterkontakt_Rechts AT %IX36.14: BOOL; (* Fensterkontakt Rechtes Fenster (Erdgeschoss) *) [CHANNEL (I)]

EG_Tuerkontakt_Links AT %IX36.15: BOOL; (* Linker Türkontakt (Erdgeschoss) *) [CHANNEL (I)]

EG_Tuerkontakt_Rechts AT %IX37.0: BOOL; (* Rechter Türkontakt (Erdgeschoss) *) [CHANNEL (I)]

Taster_vorne_Alarm AT %IX37.1: BOOL; (* Taster vorne Alarm *) [CHANNEL (I)]

Kippschalter_vorne_auf AT %IX37.2: BOOL; (* Ch_7 Digital input *) [CHANNEL (I)]

Kippschalter_vorne_ab AT %IX37.3: BOOL; (* Ch_8 Digital input *) [CHANNEL (I)]

*0750-0552 2 AO 0-20mA[VAR] (Id.: 2000014002)

Knotennummer: 10

Eingabeadresse: %IB10

Ausgabeadresse: %QB10

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Parameter:

PAAssignment: PLC

Kanäle:

Light_Blue AT %QW3: WORD; (* Ch_1 Analog output *) [CHANNEL (Q)]

Light_Yellow AT %QW4: WORD; (* Ch_2 Analog output *) [CHANNEL (Q)]

*0750-0552 2 AO 0-20mA[VAR] (Id.: 2000014002)

Knotennummer: 11

Eingabeadresse: %IB11

Ausgabeadresse: %QB11

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

Light_Red AT %QW5: WORD; (* Ch_1 Analog output *) [CHANNEL (Q)]

Light_Green AT %QW6: WORD; (* Ch_2 Analog output *) [CHANNEL (Q)]

*0750-0552 2 AO 0-20mA[VAR] (Id.: 2000014002)

Knotennummer: 12

Eingabeadresse: %IB12

Ausgabeadresse: %QB12

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

Light_White AT %QW7: WORD; (* Ch_1 Analog output *) [CHANNEL (Q)]

AT %QW8: WORD; (* Ch_2 Analog output *) [CHANNEL (Q)]

*0750-0469 2 AI Thermocouple / K / Diag.[VAR] (Id.: 2000044002)

Knotennummer: 13

Eingabeadresse: %IB13

Ausgabeadresse: %QB13

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

OG_Kinderzimmer_Temperatur AT %IW3: INT; (* Temperatur Kinderzimmer (Obergeschoss) *) [CHANNEL (I)]

OG_Schlafzimmer_Temperatur AT %IW4: INT; (* Temperatur Schlafzimmer (Obergeschoss) *) [CHANNEL (I)]

*0750-0469 2 AI Thermocouple / K / Diag.[VAR] (Id.: 2000044002)

Knotennummer: 14

Eingabeadresse: %IB14

Ausgabeadresse: %QB14

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Kanäle:

EG_Kueche_Temperatur AT %IW5: INT; (* Temperatur Küche (Erdgeschoss) *) [CHANNEL (I)]
EG_Bad_Temperatur AT %IW6: INT; (* Temperatur Badezimmer (Erdgeschoss) *) [CHANNEL (I)]
*0750-0469 2 AI Thermocouple / K / Diag.[VAR] (Id.: 2000044002)

Knotennummer: 15

Eingabeadresse: %IB15

Ausgabeadresse: %QB15

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

EG_Wohnzimmer_Temperatur AT %IW7: INT; (* Temperatur Wohnzimmer(Erdgeschoss) *) [CHANNEL (I)]
Haus_aussen_Temperatur AT %IW8: INT; (* Aussentemperatur *) [CHANNEL (I)]
*0750-0517 2 DO 230V AC 1.0A Rel 2CO (PotFree)[VAR] (Id.: 2000010002)

Knotennummer: 16

Eingabeadresse: %IB16

Ausgabeadresse: %QB16

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

Jalousie_Rechts_auf AT %QX36.12: BOOL; (* Rechte Jalousie rauffahren *) [CHANNEL (Q)]
Jalousie_Rechts_ab AT %QX36.13: BOOL; (* Rechte Jalousie runterfahren *) [CHANNEL (Q)]
*0750-0517 2 DO 230V AC 1.0A Rel 2CO (PotFree)[VAR] (Id.: 2000010002)

Knotennummer: 17

Eingabeadresse: %IB17

Ausgabeadresse: %QB17

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

Jalousie_Links_auf AT %QX36.14: BOOL; (* Linke Jalousie rauffahren *) [CHANNEL (Q)]
Jalousie_Links_ab AT %QX36.15: BOOL; (* Linke Jalousie runterfahren *) [CHANNEL (Q)]
*0750-0517 2 DO 230V AC 1.0A Rel 2CO (PotFree)[VAR] (Id.: 2000010002)

Knotennummer: 18

Eingabeadresse: %IB18

Ausgabeadresse: %QB18

Diagnoseadresse: %MB0

Download: 1

AutoAdr: 1

Parameter:

PAAssignment: PLC

Kanäle:

OG_Kinderzimmer_230V_Lampe AT %QX37.0: BOOL; (* Kinderzimmer 230 V Lampe (Obergeschoss) *) [CHANNEL (Q)]
OG_Schlafzimmer_230V_Lampe AT %QX37.1: BOOL; (* Schlafzimmer 230 V Lampe (Obergeschoss) *) [CHANNEL (Q)]
*0750-0517 2 DO 230V AC 1.0A Rel 2CO (PotFree)[VAR] (Id.: 2000010002)

Knotennummer: 19

Eingabeadresse: %IB19

Ausgabeadresse: %QB19

Diagnoseadresse: %MB0

Download: 1

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

AutoAdr: 1
Parameter:
PAAssignment: PLC

Kanäle:
EG_Kueche_230V_Lampe AT %QX37.2: BOOL; (* Küche 230 V Lampe (Erdgeschoss) *) [CHANNEL (Q)]
EG_Bad_230V_Lampe AT %QX37.3: BOOL; (* Badezimmer 230 V Lampe (Erdgeschoss) *) [CHANNEL (Q)]
*0750-0517 2 DO 230V AC 1.0A Rel 2CO (PotFree)[VAR] (Id.: 2000010002)
Knotennummer: 20
Eingabeadresse: %IB20
Ausgabeadresse: %QB20
Diagnoseadresse: %MB0
Download: 1
AutoAdr: 1
Parameter:
PAAssignment: PLC

Kanäle:
EG_Wohnzimmer_230V_Lampe AT %QX37.4: BOOL; (* Wohnzimmer 230 V Lampe (Erdgeschoss) *) [CHANNEL (Q)]
AT %QX37.5: BOOL; (* Ch_2 Digital output *) [CHANNEL (Q)]
*0750-0493 3-phase power measurement module[VAR] (Id.: 2000047003)
Knotennummer: 21
Eingabeadresse: %IB21
Ausgabeadresse: %QB21
Diagnoseadresse: %MB0
Download: 1
AutoAdr: 1
Parameter:
PAAssignment: PLC

Kanäle:
AT %IB18: BYTE; (* Ch_1 Status byte *) [CHANNEL (I)]
AT %IB20: BYTE; (* Ch_1 Input word Low Byte *) [CHANNEL (I)]
AT %IB21: BYTE; (* Ch_1 Input word High Byte *) [CHANNEL (I)]
AT %QB18: BYTE; (* Ch_1 Control byte *) [CHANNEL (Q)]
AT %QB20: BYTE; (* Ch_1 Output word Low Byte *) [CHANNEL (Q)]
AT %QB21: BYTE; (* Ch_1 Output word High Byte *) [CHANNEL (Q)]
AT %IB22: BYTE; (* Ch_2 Status byte *) [CHANNEL (I)]
AT %IB24: BYTE; (* Ch_2 Input word Low Byte *) [CHANNEL (I)]
AT %IB25: BYTE; (* Ch_2 Input word High Byte *) [CHANNEL (I)]
AT %QB22: BYTE; (* Ch_2 Control byte *) [CHANNEL (Q)]
AT %QB24: BYTE; (* Ch_2 Output word Low Byte *) [CHANNEL (Q)]
AT %QB25: BYTE; (* Ch_2 Output word High Byte *) [CHANNEL (Q)]
AT %IB26: BYTE; (* Ch_3 Status byte *) [CHANNEL (I)]
AT %IB28: BYTE; (* Ch_3 Input word Low Byte *) [CHANNEL (I)]
AT %IB29: BYTE; (* Ch_3 Input word High Byte *) [CHANNEL (I)]
AT %QB26: BYTE; (* Ch_3 Control byte *) [CHANNEL (Q)]
AT %QB28: BYTE; (* Ch_3 Output word Low Byte *) [CHANNEL (Q)]
AT %QB29: BYTE; (* Ch_3 Output word High Byte *) [CHANNEL (Q)]
*0750-0493 3-phase power measurement module[VAR] (Id.: 2000047003)
Knotennummer: 22
Eingabeadresse: %IB22
Ausgabeadresse: %QB22
Diagnoseadresse: %MB0

Aufbau eines IEC61131-3-basierenden Energiemanagementsystems für Wohngebäude unter Einbindung von Smart Metering

Download: 1
AutoAdr: 1
Parameter:
PAAssignment: PLC

Kanäle:

AT %IB30: BYTE; (* Ch_1 Status byte *) [CHANNEL (I)]
AT %IB32: BYTE; (* Ch_1 Input word Low Byte *) [CHANNEL (I)]
AT %IB33: BYTE; (* Ch_1 Input word High Byte *) [CHANNEL (I)]
AT %QB30: BYTE; (* Ch_1 Control byte *) [CHANNEL (Q)]
AT %QB32: BYTE; (* Ch_1 Output word Low Byte *) [CHANNEL (Q)]
AT %QB33: BYTE; (* Ch_1 Output word High Byte *) [CHANNEL (Q)]
AT %IB34: BYTE; (* Ch_2 Status byte *) [CHANNEL (I)]
AT %IB36: BYTE; (* Ch_2 Input word Low Byte *) [CHANNEL (I)]
AT %IB37: BYTE; (* Ch_2 Input word High Byte *) [CHANNEL (I)]
AT %QB34: BYTE; (* Ch_2 Control byte *) [CHANNEL (Q)]
AT %QB36: BYTE; (* Ch_2 Output word Low Byte *) [CHANNEL (Q)]
AT %QB37: BYTE; (* Ch_2 Output word High Byte *) [CHANNEL (Q)]
AT %IB38: BYTE; (* Ch_3 Status byte *) [CHANNEL (I)]
AT %IB40: BYTE; (* Ch_3 Input word Low Byte *) [CHANNEL (I)]
AT %IB41: BYTE; (* Ch_3 Input word High Byte *) [CHANNEL (I)]
AT %QB38: BYTE; (* Ch_3 Control byte *) [CHANNEL (Q)]
AT %QB40: BYTE; (* Ch_3 Output word Low Byte *) [CHANNEL (Q)]
AT %QB41: BYTE; (* Ch_3 Output word High Byte *) [CHANNEL (Q)]
*0750-0642 EnOcean/RF-Receiver[VAR] (Id.: 2000037001)

Knotennummer: 23
Eingabeadresse: %IB23
Ausgabeadresse: %QB23
Diagnoseadresse: %MB0
Download: 1
AutoAdr: 1
Parameter:
PAAssignment: PLC

Kanäle:

AT %IX21.0: BOOL; (* Transmission acknowledgement *) [CHANNEL (I)]
AT %IX21.1: BOOL; (* Reception request *) [CHANNEL (I)]
AT %IX21.2: BOOL; (* Initialization acknowledgement *) [CHANNEL (I)]
AT %IX21.3: BOOL; (* Number of received characters Bit 0 *) [CHANNEL (I)]
AT %IX21.4: BOOL; (* Number of received characters Bit 1 *) [CHANNEL (I)]
AT %IX21.5: BOOL; (* Number of received characters Bit 2 *) [CHANNEL (I)]
AT %IX21.6: BOOL; (* Input buffer is full *) [CHANNEL (I)]
AT %IB43: BYTE; (* Input data Byte 0 *) [CHANNEL (I)]
AT %IB44: BYTE; (* Input data Byte 1 *) [CHANNEL (I)]
AT %IB45: BYTE; (* Input data Byte 2 *) [CHANNEL (I)]
AT %QX21.0: BOOL; (* Transmission request *) [CHANNEL (Q)]
AT %QX21.1: BOOL; (* Reception acknowledgement *) [CHANNEL (Q)]
AT %QX21.2: BOOL; (* Initialization request *) [CHANNEL (Q)]