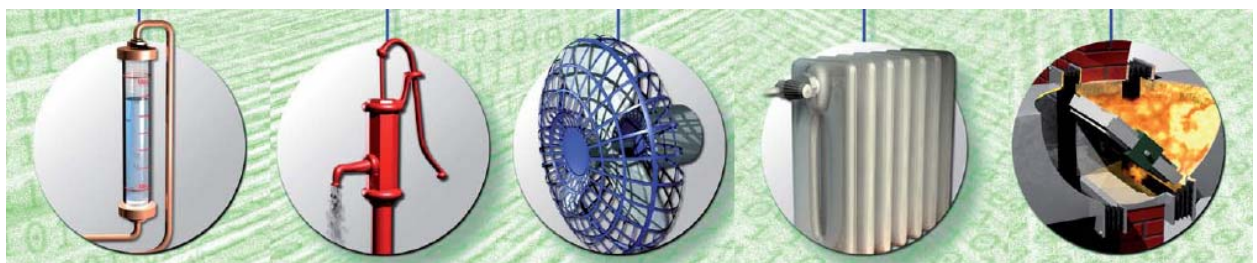


# Lokale Vorrangbedien- und Anzeigeeinrichtung (LVB) für physikalische und virtuelle Datenpunkte im BACnet MS/TP Netzwerk

für Rauchschalter (RM) und Brandschutzklappen (BSK)  
und sonstige virtuelle Datenpunkte die im gleichen BACnet  
MS/TP Bus Segment erreichbar sind.



## BACnet MS/TP Anzeigesystem

**für Rauchschalter (RM) und Brandschutzklappen (BSK) und sonstige virtuelle Datenpunkte die am gleichen BACnet MS/TP Bus erreichbar sind.**

### Allgemein

Die Module der RDC700-Serie dienen zur Anzeige von Betriebszuständen bzw. Klappenstellungen, und zur Anforderung von Selbsttestroutinen über mehrfarbige LED's und Taster. Die Bedienung der Datenpunkte erfolgt über I/O- Module verschiedener Hersteller. Als Kommunikationsmedium wird das BACnet MS/TP Protokoll im RS485 Netzwerk verwendet.

Die Module können im BACnet-Netzwerk mit anderen Geräten zur Erstellung von kostengünstigen, interoperablen Kontrollsystemen für Gebäude und Industrieanwendungen genutzt werden.

### Merkmale

- Anzeige des Klappenstatus
- Anzeige von Stör- und Warnmeldungen
- Anforderung von Selbsttestfunktion über Taster
- Montage in 19“-Einbaurahmen, 3 HE
- Einfache Installation durch ein steckbares Bussystem
- Beschriftung individuell möglich
- LED-Farbe parametrierbar über Software
- BACnet Objekte nach ANSI/ASHRAE Standard 135-2012
- Versorgungsspannung 24VAC/DC
- Autarkes System, Funktion auch ohne BACnet B-BC Controller

Konfiguration der Geräte erfolgt üblicherweise über das romutec Tool ROBA-CONNECT.



Es ist aber mit jedem handelsüblichen BACnet Explorer möglich, der die verwendeten BACnet Objekte unterstützt.

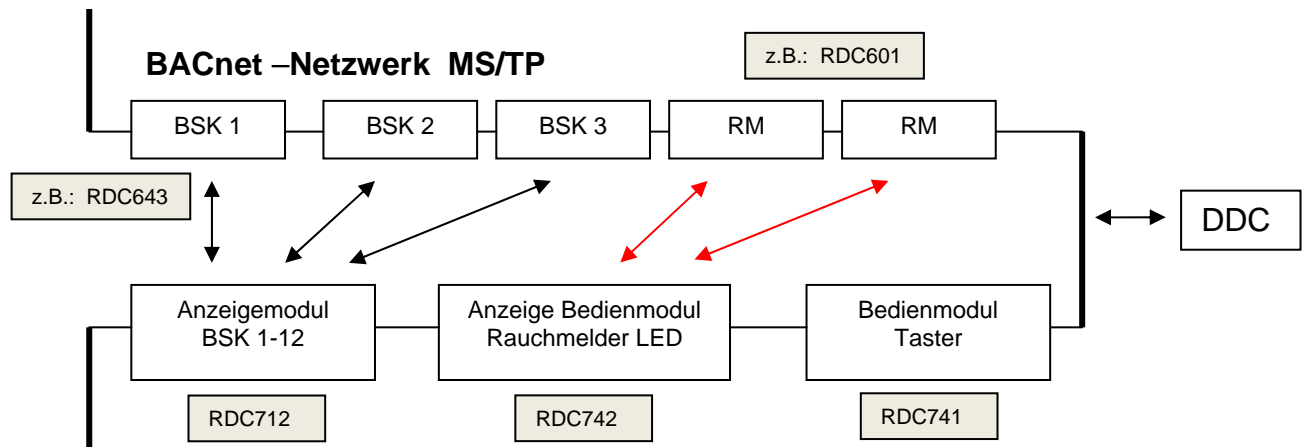
#### Copyright

Copyright© 2015 romutec® Steuer- und Regelsysteme GmbH. Alle Rechte vorbehalten. Ohne die ausdrückliche schriftliche Genehmigung darf diese Anleitung weder als Ganzes noch in Teilen reproduziert, übertragen, umgeschrieben, in Datenerfassungssystemen gespeichert oder in andere Landes- bzw. Computersprachen übersetzt werden. Dies gilt für jede Form und jedes Mittel, sei es elektronisch, mechanisch, magnetisch, optisch, manuell oder auf andere Art und Weise.

Stand 03.03.2015

## Funktionen und Bedienung

Die folgende Abbildung zeigt die Konfiguration von RDC700-Modulen, die zum Einsatz mit diversen Feldgeräten und Datensammlern Verwendung finden.



## Beschreibung der Bedien- und Anzeige-Module:

Die Geräte RDC700 besitzen keine physikalischen Datenpunkte. Die vorhandenen Taster und LED's werden über BACnet Objekte dem Netzwerk zur Verfügung gestellt. Über externe Controller (Geräte) werden diese dann für ihre verwendete Funktion weiterverarbeitet. Dort werden auch Funktionen programmiert, die Anlagenspezifisch gefordert sind. Die Auswertung der Datenpunkte im Netzwerk erfolgen über das Objekt „Structured-View“. Hierüber wird eine Verknüpfung zum physikalischen Datenpunkt, oder einem Datenpunkt im Fremdgerät hergestellt. Durch die Zuordnung des „Device“ und dem „Objekt“ wird der „Present Value“ abgefragt. Durch den aktivierten „CoV“ findet eine kontinuierliche Überwachung statt.

### Grundfunktionen:

#### Anzeigemodul (12 LED's): RDC 712

Visualisierung von Digitaleingängen als LED mit Auswahl der anzuzeigenden Farbe grün/gelb/rot mit und ohne blinken

#### Bedien-u. Anzeigemodul: RDC741

Visualisierung von Digital-Eingängen als LED mit Auswahl der anzuzeigenden Farbe grün/gelb/rot mit und ohne blinken  
Bedienung und Steuerung von Antrieben mit Taster  
und als Sammelmeldemodul bis max. 32 Devices

#### Bedien-u. Anzeigemodul: RDC742

Visualisierung von Digital-Eingängen als LED mit Auswahl der anzuzeigenden Farbe grün/gelb/rot mit und ohne blinken  
Bedienung und Steuerung von Antrieben mit Taster  
Applikation für Brandschutzklappen oder Rauchmelder

# Systembeschreibung RDC 700 Serie

LED - Anzeige des RDC742 :

		Funktion	Farbe	Beschriftung
<b>LEDx Oben</b>	<b>Status LED Verschmutzt</b>	OK	Aus	
		Taster gedrückt	Gelb	RM verschmutzt
		Test ausgelöst	Grün	
		MAC fehlt	rot blinken	
<b>LEDx Mitte</b>	<b>Rauchmelder LED Störung</b>	OK	Aus	
		Verschmutzt	Gelb	
		Störung	gelb blinken	RM gestört
		Ausgelöst	rot blinken	RM ausgelöst
<b>LEDx Unten</b>	<b>Klappe LED ZU</b>	„Offen“	Grün	BSK Offen
		„Zu“ bei Anlage Aus	Aus	BSK geschlossen
		Schmelz Lot	rot	BSK gefallen
		„Unterwegs“ (Fährt)	gelb blinken	BSK in Zwischenstellung
		Störung	gelb-rot blinken	

## Beschreibung Datensammler (optional) : Datensammler für Feldmontage sind in Vorbereitung !

Das Gerät besitzt physikalische Datenpunkte.

Als Grundlage dient hier die RDC600 Serie. Dort sind alle nötigen Datenpunkte vorhanden. Dieses sind universelle Eingänge, digitale Ein- und Ausgänge sowie Analogausgänge. Dort ist für jeden Datenpunkt eine LVB mit Status Überwachung und Bedienung vorzusehen.

Die Zuordnung zu den Datenpunkten erfolgt über die BACnet Objekte und Property.

Optional sollte eine Applikation parametrierbar oder programmiert werden können.

Überwachung der Klappenlaufzeit durch Auswertung der Endlagenschalter

Parametrierbar Soll-Zeitvorgabe (0-999 sec.)

Auswertung bei gefallener Klappe, ob diese über Schmelzlot oder Rauchmelder ausgelöst wurde.

Testlauf für alle Klappen über einen zentralen Taster.

Speicherung der Daten des letzten Testlaufes.

Testlauf und Reset des Rauchmelders über einen zentralen Taster.

# Systembeschreibung RDC 700 Serie

Taster  
für Testfunktionen  
je nach Applikation  
Klappe oder Rauchmelder

Taster  
für Testfunktionen  
je nach Applikation:

zentrale Steuerung aller  
Klappen  
über einen Taster

Testfunktion Rauchschalter



LED's mehrfarbig

Zustandsanzeige  
grün/gelb/rot  
blinkend  
oder dauerlicht



LED's mehrfarbig

Zustandsanzeige  
grün/gelb/rot  
blinkend  
oder dauerlicht

# Systembeschreibung RDC 700 Serie

## Montage:

Die Montage der im 19 Zoll Format erhältlichen Geräte erfolgt im Trägerrahmen, die in verschiedenen Abmessungen erhältlich sind. Es wird ein Platz von 4TE und 8 HE je Gerätetyp benötigt.



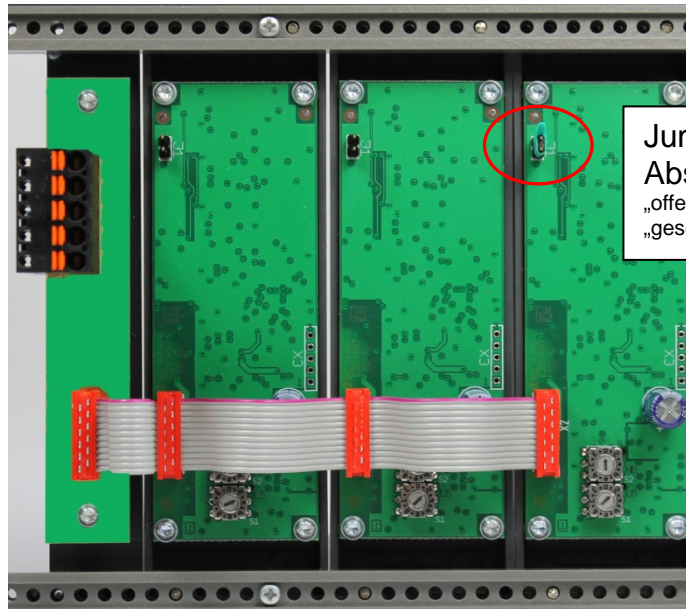
Abb. Einbau im Trägerrahmen Breite 50TE

## Elektrischer Anschluss:

### Anschlußklemmen:

1 = +24V DC  
2 = GND  
3 = B NET +  
4 = A NET -  
5 = NET GND

Verbindung Modul zu Modul:  
Mit einem Systemkabel werden die Geräte mit Spannung und dem BACnet MS/TP-BUS verbunden.



Jumper für BUS-Abschluss  
„offen“ = ohne  
„geschlossen“ = 120 Ohm

Abb. Rückansicht BUS Verbindung

### Busabschluss :

Terminierung der RS 485 Schnittstelle erforderlich (nach EIA 485) aktiv oder passiv !

BIAS Widerstände 47k vorhanden ! Kann auch mit 120 Ohm (A-B) abgeschlossen werden. Ausführung siehe BACnet – Dokumentation ANSI/ASHRAE Standard 135-2008 Network Seite 75

### Baudrate

Die RDC Module arbeiten mit AUTOBAUD. D.h. es muss keine Einstellung vorgenommen werden. Default = AUTOBAUD

### Unterstützte Modi :

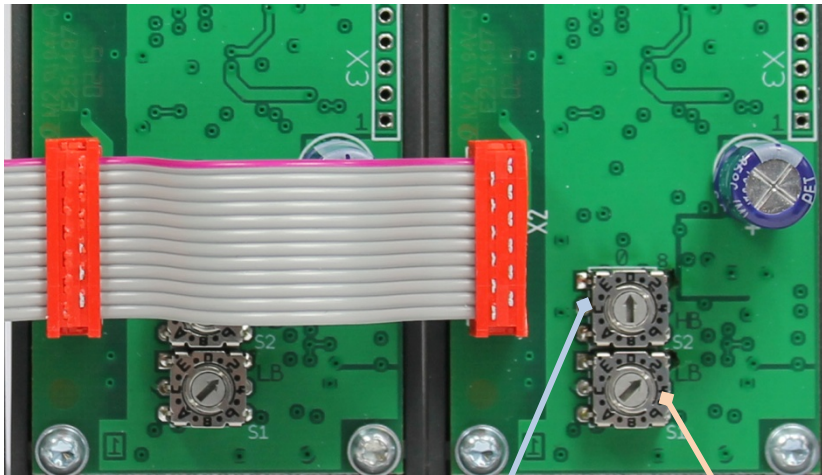
1 = 9.600 bps; 2 = 19.200 bps; 3 = 38.400 bps;  
4 = 76.800 bps; 5 = 115.000 bps

Format MS/TP RS485 EIA-485; 8,N,1

# Systembeschreibung RDC 700 Serie

## Adressen:

Die Geräteadresse MAC-Adresse wird an den Drehcodierschaltern auf der Rückseite unten am RDC Modul eingestellt. Diese sind in Hexadezimal Codiert.



Wert links	HB Einstell Wert	Einstell Wert	LB Wert rechts
0	0	0	0
16	1	1	1
32	2	2	2
48	3	3	3
64	4	4	4
80	5	5	5
96	6	6	6
112	7	7	7
128	8	8	8
144	9	9	9
160	A	A	10
176	B	B	11
192	C	C	12
208	D	D	13
224	E	E	14
240	F	F	15

Adresse eingestellt:            6 - 6 =102  
Berechnet : Wert links + Wert rechts = Adresse

Für BACnet MS/TP-Master Geräte ist der Adressbereich 1-127 zu verwenden. Slave Geräte sind im Adressbereich 128-250 zu verwenden.

Über die Festlegung bzw. Programmierung der **MAC Adresse** wird das Gerät im Betriebsmodus Master oder Slave betrieben. Im Slave-Betrieb ist ein BACnet-Master notwendig, um die Daten des RDC's zu erreichen.  
Dies kann auch ein Router mit der Funktion des Slave Proxy Mode sein.



## Überwachung Modulstatus:

Die Kommunikation kann an den beiden unteren LEDs der Module überwacht werden.

### LED RX/TX (unten rechts):

Werden Daten auf eine Antwort von BACnet MS/TP Netzwerk gesendet, blinkt die RX/TX LED grün. Der Token bleibt unberücksichtigt.



### LED Status Taster (unten rechts):

Vorraussetzung ist die Funktion des „Structured-View“ ist aktiviert. Wird noch auf den folgenden Seiten beschrieben.

Farbe „rot Dauerlicht“ und Taster gedrückt :  
Es ist keine Adresse im Objekt „Structured-View“ eingetragen  
oder  
es ist keine gültige Adresse im Objekt „Structured-View“ eingetragen

Taster wurde gedrückt, bis der „Present-Value“ gesendet ist, wird die LED aktiv mit der Farbe „orange“.  
Bleibt der Taster gedrückt, wechselt die LED in „rot“ Dauerlicht bis der Taster nicht mehr aktiv.

### Für die Typen RDC712 und 742 gilt.

Ist eine gültige Adresse im Objekt „Structured-View“ eingetragen blinkt die zugehörige LED solange „rot“ bis der abonnierte Datenpunkt erreicht wurde und ein gültiger Wert gesendet wurde. Beim RDC 742 „gelb/rot“ blinkend.  
Dann wird er eingestellte Wert angezeigt.  
Default ist GRÜN. Es kann einige Sekunden dauern, bis der Prozess gestartet ist.  
Ändert sich allerdings der „Present-Value“ des Datenpunkt wird ein CoV gesendet und der Wert sofort übernommen.



# Systembeschreibung RDC 700 Serie

## Modulbeschreibung RDC712 :

Visualisierung von bis zu 12 Digitaleingängen als LED Anzeige mit Auswahl der anzuzeigenden Farbe grün/gelb/rot, mit und ohne blinken.

Die anzuzeigenden Statusmeldungen werden von den jeweiligen physikalischen oder virtuellen Datenpunkten mittels BACnet Objekte „Structured-View“ Objekt abonniert. Die Statusänderung (COV = Cange of Value) und Zustände am physikalischen Eingang werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed\_COV“ oder „Unconfirmed\_COV“ übermittelt werden.

## Konfiguration des RDC712:

### Device Objekte:

Eigenschaft	Datentyp	Wert
<b>Identification</b>		
Object-Identifier	Object-Identifier	Device (8) # 6
Object-Type	Object-Type	Device (ID:8)
Object-Name	Character-String	RDC712 (SN: 4273864704)
Description	Character-String	Testdescription
Location	Character-String	Testlocation
<b>Vendor, Model and Version</b>		
Vendor-Name	Character-String	romutec Steuer u Regelsysteme
Vendor-Identifier	Unsigned-16	290
Model-Name	Character-String	RDC712
Firmware-Revision	Character-String	0.11.006
Database-Revision	Unsigned-Integer	0
Application-Software-Version	Character-String	0.01
<b>Status</b>		
System-Status	Device-Status	Operational (ID:0)
<b>Protocol</b>		
Protocol-Version	Unsigned-Integer	1
Protocol-Revision	Unsigned-Integer	14
Protocol-Services-Supported	Services-Supported	Subscribe-Cov, Read-Property, Read-Property-Multiple,
Protocol-Object-Types-Supported	Object-Types-Supported	Binary-Value (ID:5), Device (ID:8), Mul

Abb. Devce Objekte

Ab Werk ist die Instance Nummer (InstanceNumber) gleich der Seriennummer. Diese kann jedoch geändert werden.

Für die Bezeichnung (Name) und Ort (Location) können Texte bis zu 63 Zeichen hinterlegt werden.

## Binary-Value Objekt:

# Systembeschreibung RDC 700 Serie

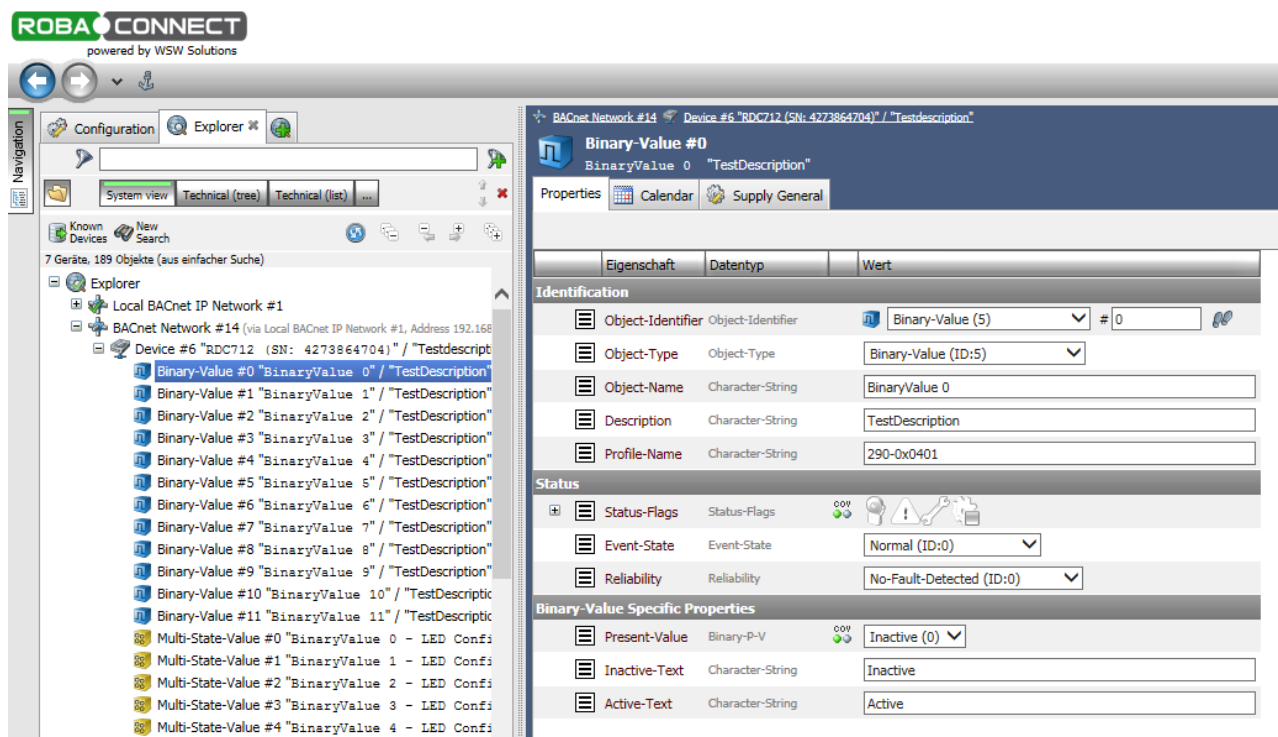


Abb. Binary-Value

Für jede der 12 Status LEDs gibt es einen „Binary-Value“ (#0-#11).

SubScribe\_COV ist verfügbar.

Für die Datenpunkte (Objekt-Name) und die Beschreibung (Description) können Texte bis zu 63 Zeichen hinterlegt werden.

Im Profil-Name ist die Farbe bzw. Eigenschaft der Status-LED hinterlegt. Dies kann über den zugehörigen Multi-State-Value (#x) verändert werden.

Den Status der LED wird im „Present-Value“ als Inactive (0) oder Aktiv (1) angezeigt.

Der „Binary-Value“ zeigt den zugeordneten, aktuellen Zustand eines Remote Datenpunktes an. Dieser wird über das zugehörige „Structured-View“ Objekt ausgewählt.

## Structured-View Objekt:

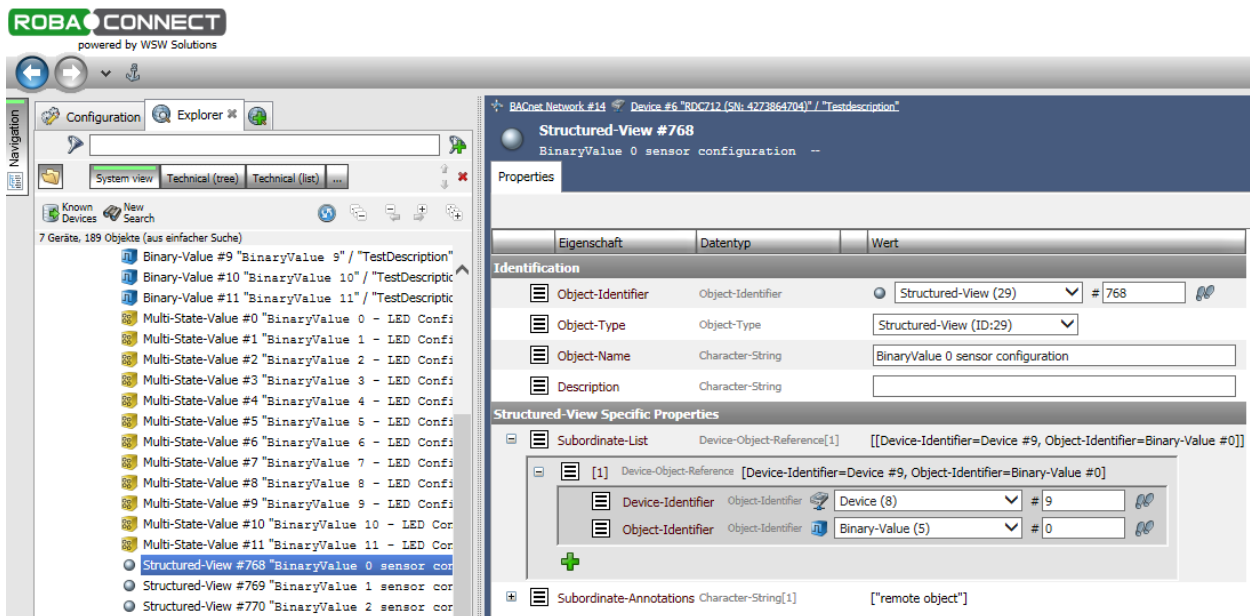


Abb. Structured-View

Für jede der 12 Status LEDs gibt es ein „Structured-View“ Sensor Configuration Objekt (#768 bis #779)

In diesem Objekt wird der Zustand eines Datenpunktes (physikalisch oder virtueller) der Status LED zugeordnet. Diese zeigt dann den aktuellen Zustand im „Binary-Value“ (Present-Value als Inactive oder Aktiv) und in der eingestellten Farbe an.

**Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein!**

Einzutragen sind

<b>Device-Identifizier:</b>	<b>Device(8)</b>	<b>DeviceID (hier die 9)</b>
<b>Objekt-Identifizier:</b>	<b>Objekt-Type (hier Binary-Value)</b>	<b>Objekt-Number (hier die 0)</b>

Die Statusänderung (COV = Cange of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed\_COV“ oder „Unconfirmed\_COV“ übermittelt werden.

Ist die abonnierte Datenverbindung gestört oder fehlerhaft blinkt die entsprechende Statusanzeige LED gelb/rot.

## Deaktivierung der Überwachungsfunktion des Structured-View Objekts:

Die Funktion des Structured-View kann deaktiviert werden. Dies ist nötig wenn ein oder alle der 12 Status LEDs über den vorhanden „Multistate-Value“ (#0 bis #11) direkt beschrieben werden sollen.

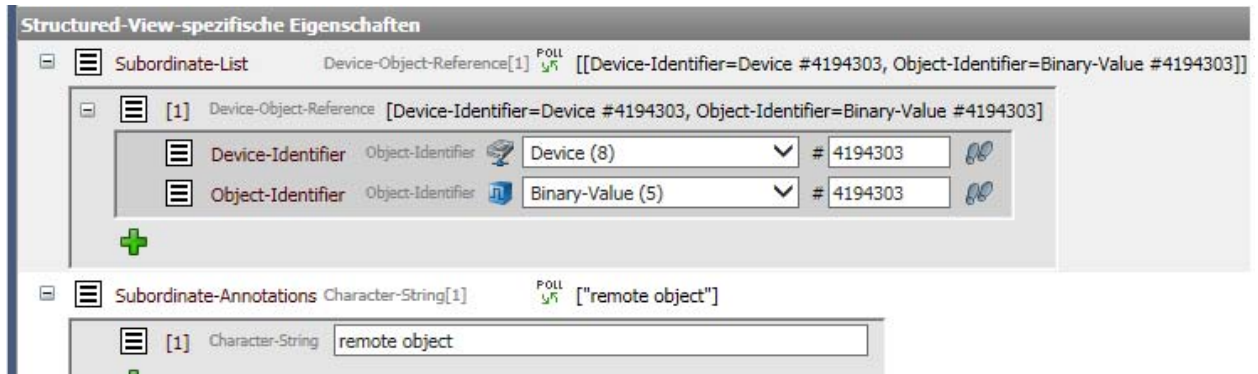


Abb. Structured-View Deaktivierung

Die Funktion wird deaktiviert, durch das Eintragen der Zahl "4194303" in die Device-Object-Reference.

Dort in der Zuordnung

<b>Device-Identifer:</b>	<b>Device(8)</b>	<b>DeviceID</b>	<b>4194303</b>
<b>Objekt-Identifer:</b>	<b>Objekt-Type (hier Binary-Value)</b>	<b>Objekt-Number</b>	<b>4194303</b>

Siehe Abb. Structured-View Deaktivierung oben

## Multistate-Value Objekt:

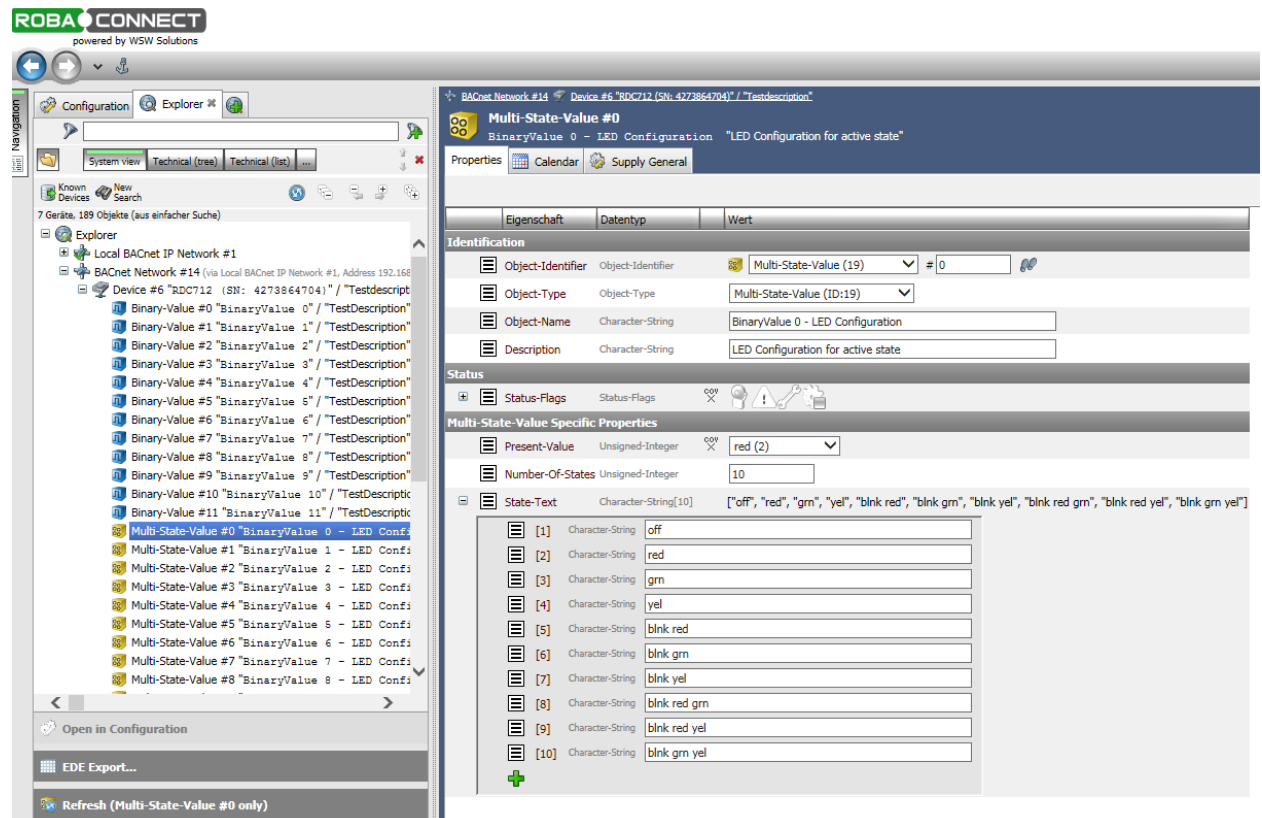


Abb. Multistate-Value zur LED Konfiguration

Für jede der 12 Status LEDs gibt es einen „Multistate-Value“ (#0 bis #11).

Im Profil-Name des „Binary-Value“ (#0-#11) ist die Farbe bzw. Eigenschaft der Status-LED hinterlegt. Die Einstellung kann direkt im Profil-Name (siehe Tabelle) oder über den zugehörigen „Multi-State-Value“ (#0-#11) hier mit dem „Present-Value“ (siehe „State Text“) verändert werden.

State-Text	Status	Eigenschaft		Prefix	nibble 1	nibble 2	nibble 3	nibble 4
				Profil-Name	NU	Fehler	Aus	An
1	OFF	Aus	0x0	290/0x	0	4		
2	RED	Rot	0x1	290/0x				
3	GREEN	Grün	0x2	290/0x				
4	YELLOW	Gelb	0x3	290/0x				
5	BLINK_RED	Rot Blinken	0x4	290/0x				
6	BLINK_GREEN	Grün Blinken	0x5	290/0x				
7	BLINK_YELLOW	Gelb Blinken	0x6	290/0x				
8	BLINK_RED_GREEN	Rot Grün Blinken	0x7	290/0x				
9	BLINK_RED_YELLOW	Rot Gelb Blinken	0x8	290/0x				
10	BLINK_GREEN_YELLOW	Grün Gelb Blinken	0x9	290/0x				

Abb. Tabelle LED Farben

## Konfiguration der Status LED (mit Structured-View Funktion)

Die gewünschte Farbe der Status LED wird über den State-Text ausgewählt. Die dort getroffene Auswahl wird auch im Profil-Name des „Binary-Value“ gespeichert.

Im Profil-Name steht z.B.: 290/0412 :

Farbe bei Fehler: orange      bei AUS/OFF: LED aus/off      bei EIN/ON: LED rot/red

# Systembeschreibung RDC 700 Serie

## Ansteuerung der LED's ohne Structured-View

Wird die Funktion des Structured-View wie beschrieben deaktiviert, kann jede der 12 Status LEDs über den vorhanden „Multistate-Value“ (#0 bis #11) direkt beschrieben werden.

Dies erfolgt mit dem zugehörigem „Present-Value“ (siehe hierzu auch Abb. Multistate-Value zur LED Konfiguration).

Die LEDs leuchten dann nach den Eigenschaften wie im „State Text“ definiert sind.

Wird zum Beispiel

der Wert 1 geschrieben

dann ist die LED ausgeschaltet (off)

der Wert 2 geschrieben

dann leuchtet die LED in der Farbe rot (red)

der Wert 8 geschrieben

dann blinkt die LED in den Farben rot (red) – grün (green)

The screenshot shows a configuration window for LED states. At the top, there is a field for 'Number-Of-States' (Unsigned-Integer) set to 10. Below it is the 'State-Text' field (Character-String[10]) with a preview of the state strings: ["off", "red", "grn", "yel", "blnk red", "blnk grn", "blnk yel", "blnk red grn", "blnk red yel", "blnk grn yel"]. The main area contains a list of 10 states, each with a state number in brackets, a 'Character-String' type, and a text input field containing the state string. A green plus sign is visible at the bottom left of the list.

State	Type	Value
[1]	Character-String	off
[2]	Character-String	red
[3]	Character-String	grn
[4]	Character-String	yel
[5]	Character-String	blnk red
[6]	Character-String	blnk grn
[7]	Character-String	blnk yel
[8]	Character-String	blnk red grn
[9]	Character-String	blnk red yel
[10]	Character-String	blnk grn yel

Abb. State-Text des Multistate-Value zur LED Konfiguration







# Systembeschreibung RDC 700 Serie

## Binary-Input Objekt:

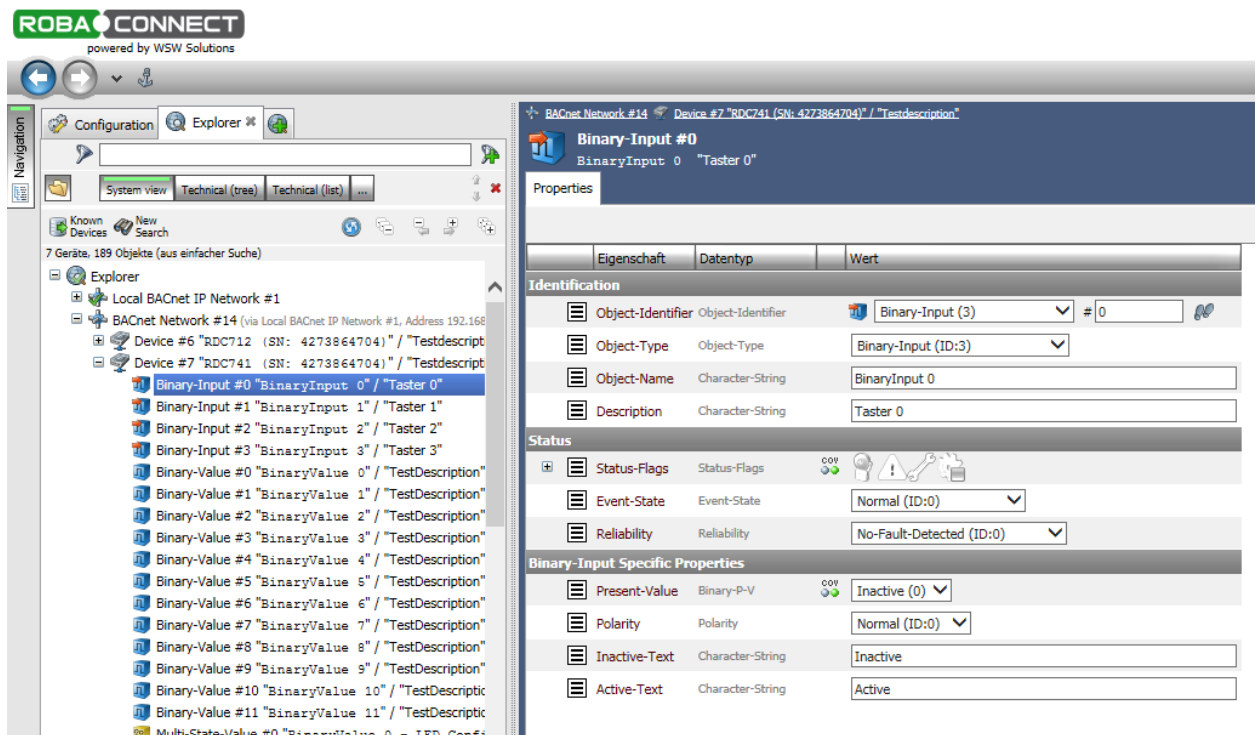


Abb. Binary-Input

Für jeden der 4 Taster gibt es einen „Binary-Input“ (#0-#3).

Jeder einzelne Taster kann bis zu 32 BACnet Objekte (Binary Output oder Binary Value) beschreiben. Diese werden im „Structured-View“ Objekt (#0-#3) eingetragen. SubScribe\_COV ist verfügbar.

Für die Datenpunkte (Objekt-Name) und die Beschreibung (Description) können Texte bis zu 63 Zeichen hinterlegt werden.

Im Profil-Name ist die Farbe bzw. Eigenschaft der Status-LED hinterlegt. Dies kann über den zugehörigen Multi-State-Value (#x) verändert werden.

Der Status des Tasters wird im „Present-Value“ als Inactive (0) oder Aktiv (1) angezeigt.

Über die „Polarity“ kann der Status invertiert werden.

### Funktion:

Mit dem „Device Type“ kann eine Toggle-Funktion eingestellt werden.

Toggle: 0 =  
„Present-Value“ wechselt von 0 nach 1

Toggle: 1 =  
„Present-Value“ wechselt von 1 nach 0

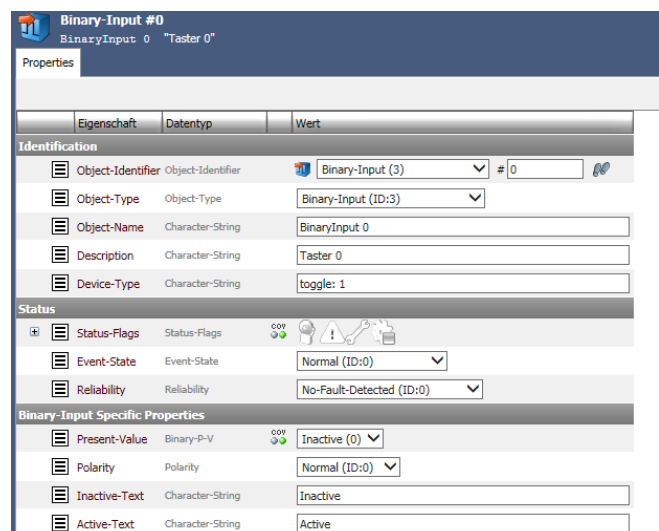


Abb. Toggle

## Binary-Value Objekt:

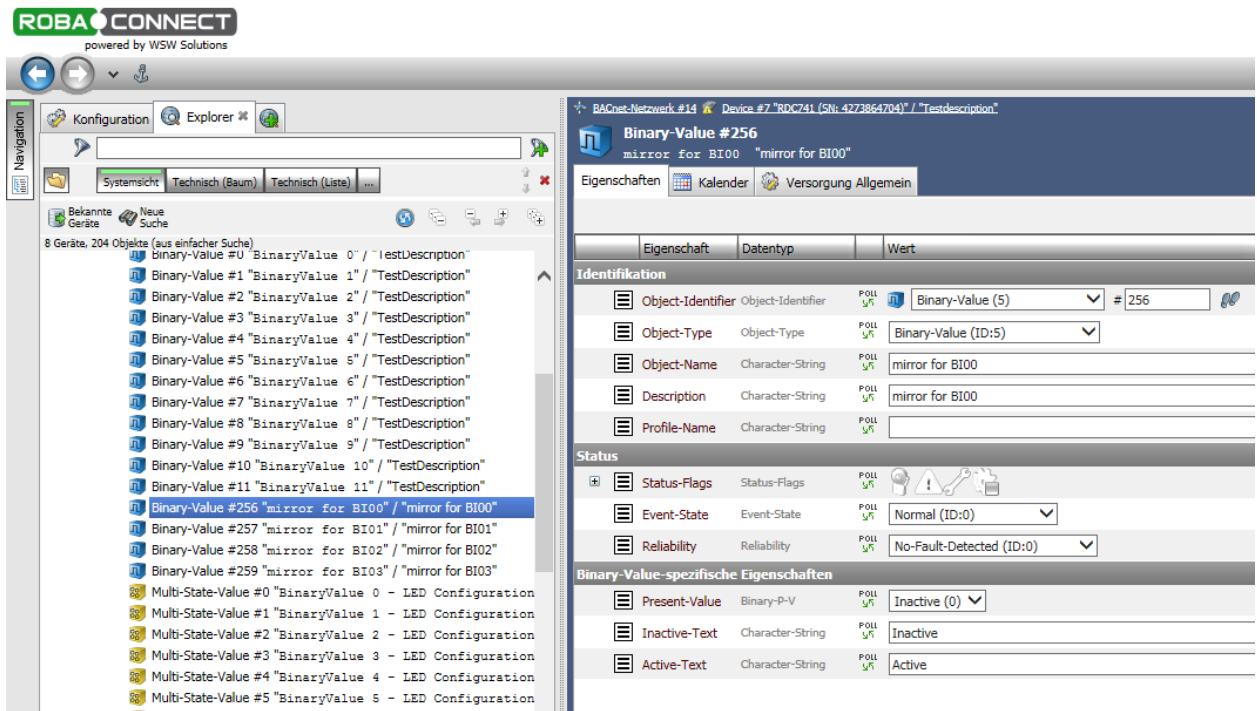


Abb. Binary-Value

Für jeden der 4 Taster gibt es einen „Binary-Value“ (#256-#259).

Dieser spiegelt den aktuell gespeicherten Wert (Aktiv oder Inaktiv) des Taster wieder. Nach dem externen lesen des Wertes wird dieser wieder zurück gesetzt . Wurde der Taster betätigt, wechselt der „Present-Value“ auf „Aktiv“. Wird nun der „Present-Value“ von extern gelesen, so ändert sich der Zustand von „Aktiv“ auf „Inaktiv“. Mit dieser Funktion wird das Betätigen des Tasters gespeichert, bis der Zustand gelesen und sicher weiterverarbeitet ist.

Hierbei ist zu beachten, dass nur das Device den „Binary-Value“ liest, welches auch die Funktion weiterverarbeiten soll.

Die im Objekt „Binary-Input“ (#0-#3) enthaltenen Property wie „Polarity“ (Normal oder Reverse) sowie die Einstellungen im Device Type (Toggle 0 oder 1) haben keine Auswirkungen auf die Logik. Er wechselt von „Inaktiv“ (Taster nicht gedrückt) nach „Aktiv“ (Taster wurde betätigt).

## Structured-View Objekt:

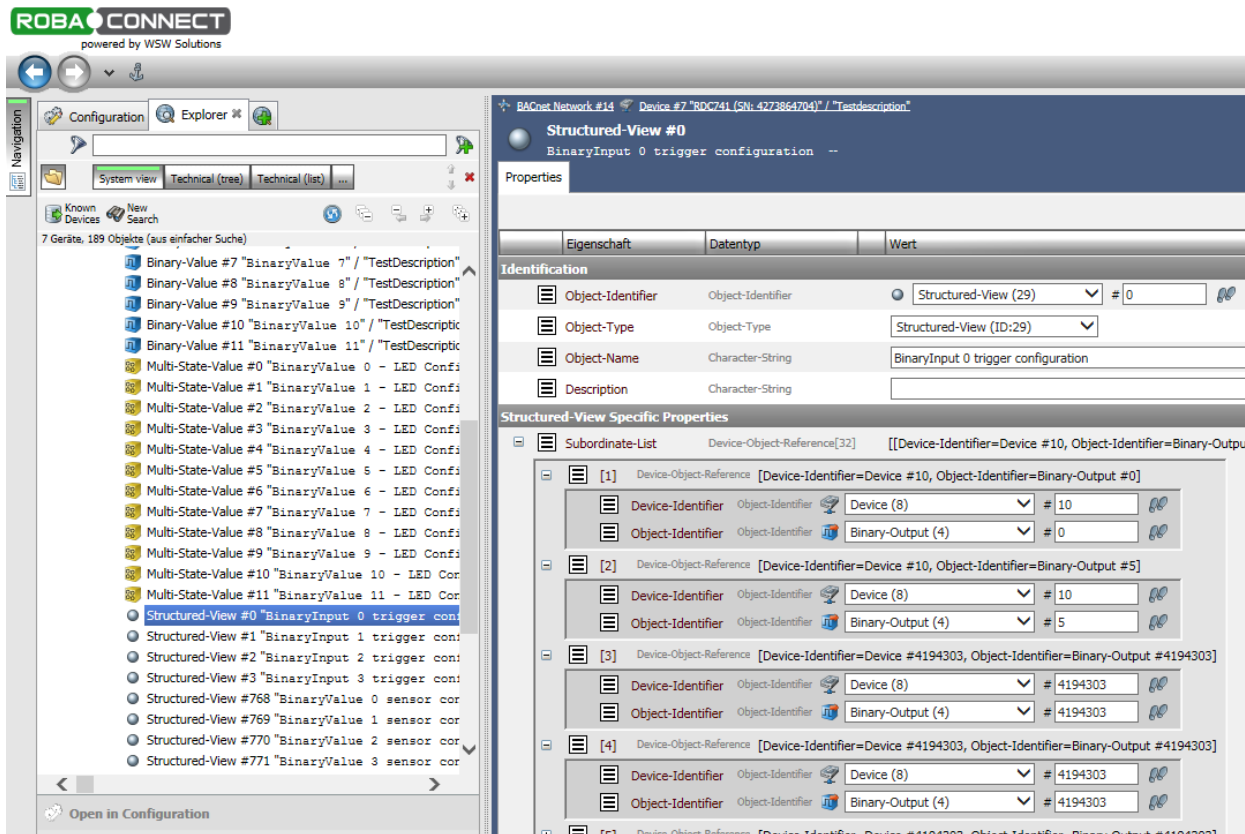


Abb. Structured-View Objekt

Für jeden der 4 Taster gibt es ein „Structured-View“ (trigger configuration) Objekt (#0 bis #3)

Es könne für jeden Taster 32 Ziele (Einträge) definiert werden.

In diesem Objekt wird das Device und die Datenpunkte (physikalisch oder virtueller) des zu Übersteuernden Gerätes zugeordnet. Es könne Master oder Slave Devices gesteuert werden.

**Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein !**

Einzutragen sind  
bei Master-Device:

<b>Device-Identifizier:</b>	<b>Device (8)</b>	<b>DeviceID (hier die 10)</b>
<b>Objekt-Identifizier:</b>	<b>Objekt-Type (hier Binary-Output) Binary-Output oder Binary-Value</b>	<b>Objekt-Number (hier die 0)</b>

bei Slave-Device:

<b>Device-Identifizier:</b>	<b>Characterstring-Value (40)</b>	<b>MAC-Adresse (hier die 10)</b>
<b>Objekt-Identifizier:</b>	<b>Objekt-Type (hier Binary-Output) Binary-Output oder Binary-Value</b>	<b>Objekt-Number (hier die 1)</b>

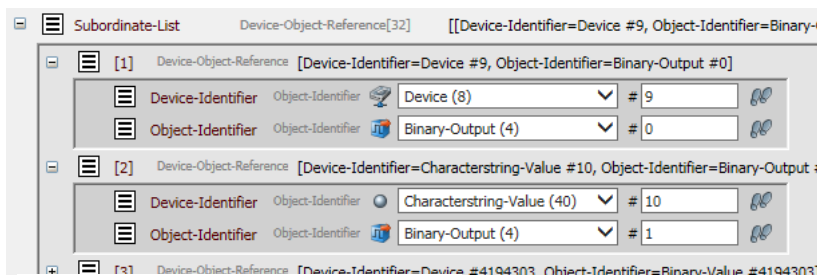


Abb. Structured-View Objekt Master/Slave

# Systembeschreibung RDC 700 Serie

Die Statusänderung (COV = Cange of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed\_COV“ oder „Unconfirmed\_COV“ übermittelt werden.

Ist die abonnierte Datenverbindung gestört oder fehlerhaft blinkt die entsprechende Statusanzeige gelb/rot.

Wenn eine Aktion an die Device nicht übergeben werden kann, wird dieses an der Statusanzeige unter den Taster durch eine LED rot signalisiert.

Die Funktion wird deaktiviert, durch das Eintragen der Zahl “4194303“ in die Device-Object-Reference.

Dort in der Zuordnung

<b>Device-Identifizier:</b>	<b>Device(8)</b>	<b>DeviceID</b>	<b>4194303</b>
<b>Objekt-Identifizier:</b>	<b>Objekt-Type (hier Binary-Value)</b>	<b>Objekt-Number</b>	<b>4194303</b>

Siehe Abb. Structured-View Deaktivierung Seite

## Ansteuerung der Binary-Output im Structured-View (Priorität)

Mit welcher Priorität der Datenpunkt beschrieben wird, kann im Objekt „Structured-View“ und dort im Device-Type eingestellt werden. Am Beispiel unten ist es die Priorität 8 (prio: 8).

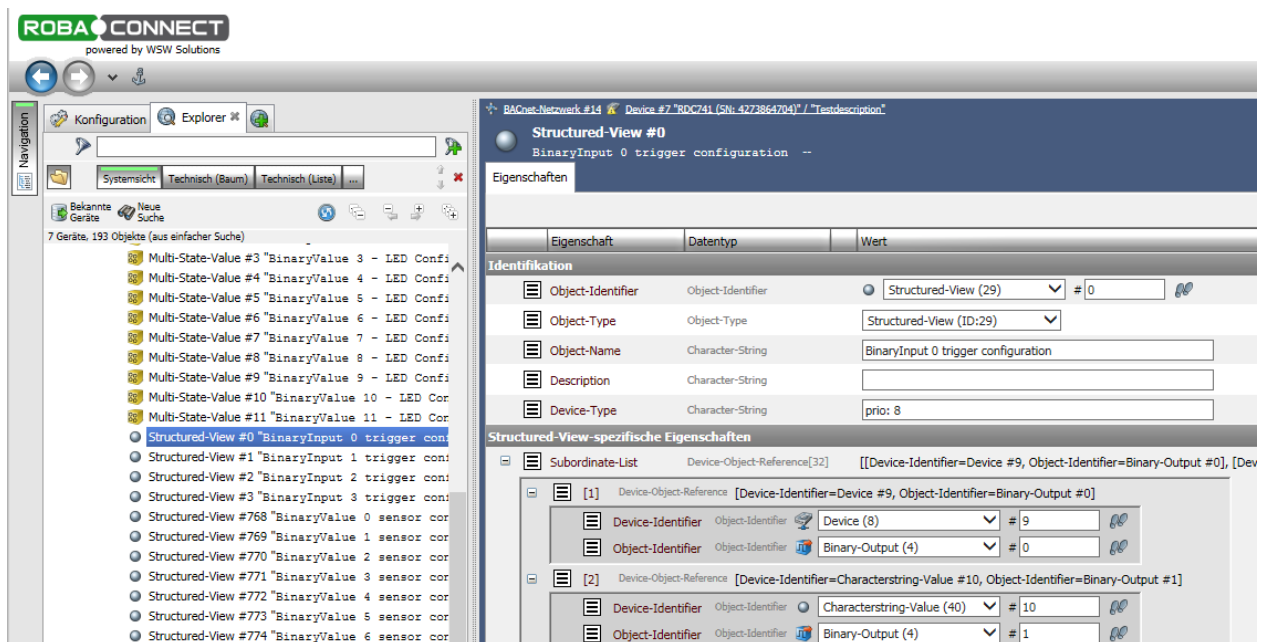


Abb. Structured-View Objekt Priorität

## Ansteuerung der LED´s ohne Structured-View

Wird die Funktion des Structured-View wie beschrieben deaktiviert, kann jede der 12 Status LEDs über den vorhanden „Multistate-Value“ (#0 bis #11) direkt beschrieben werden sollen.

Dies erfolgt mit dem zugehörigem „Present-Value“ (siehe hierzu auch Abb. Multistate-Value zur LED Konfiguration).

Die LEDs leuchten dann nach den Eigenschaften wie im „State Text“ definiert sind.

Wird zum Beispiel

der Wert 1 geschrieben

dann ist die LED ausgeschaltet (off)

der Wert 2 geschrieben

dann leuchtet die LED in der Farbe rot (red)

der Wert 8 geschrieben

dann blinkt die LED in den Farben rot (red) – grün (green)

Weitere Details siehe Seite 8 und 9. Die Einstellung sind gleich dem Modul RDC712.

Siehe Abb. Structured-View Deaktivierung.

## Binary-Value Objekt:

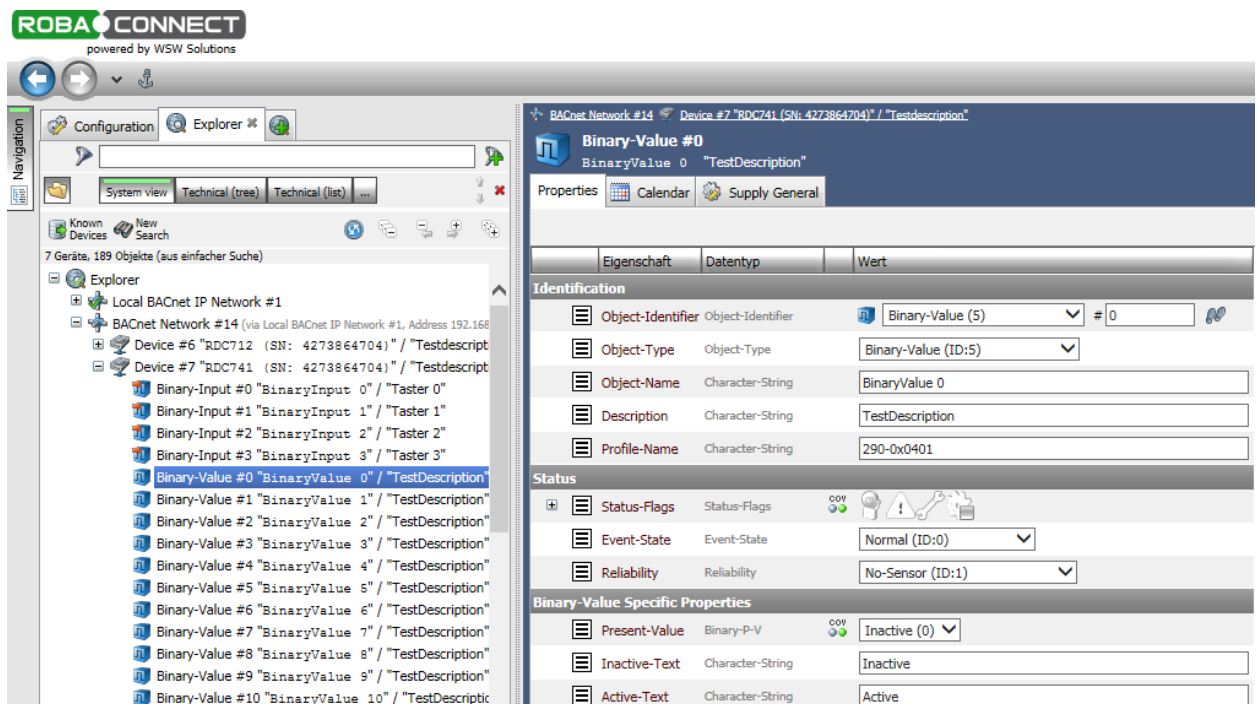


Abb. Binary-Value

Für jede der 12 Status LED's gibt es einen „Binary-Value“ (#0-#11).

SubScribe\_COV ist verfügbar.

Für die Datenpunkte (Objekt-Name) und die Beschreibung (Description) können Texte bis zu 63 Zeichen hinterlegt werden.

Im Profil-Name ist die Farbe bzw. Eigenschaft der Status-LED hinterlegt. Dies kann über den zugehörigen Multi-State-Value (#x) verändert werden.

Den Status der LED wird im „Present-Value“ als Inactive (0) oder Aktiv (1) angezeigt.

Der „Binary-Value“ zeigt den zugeordneten, aktuellen Zustand eines Remote Datenpunktes an. Dieser wird über das zugehörigen „Structured-View“ Objekt ausgewählt.

## Konfiguration der Status LED

### Multistate-Value Objekt:

Siehe hierzu auch LED Konfiguration RDC712

Für jede der 12 Status LED gibt es einen „Multistate-Value“ (#0 bis #11).

Im Profil-Name des „Binary-Value“ (#0-#11) ist die Farbe bzw. Eigenschaft der Status-LED hinterlegt. Die Einstellung kann direkt im Profil-Name (siehe Tabelle) oder über den zugehörigen „Multi-State-Value“ (#0-#11) hier mit dem „Present-Value“ (siehe „State Text“) verändert werden.

## Structured-View Objekt:

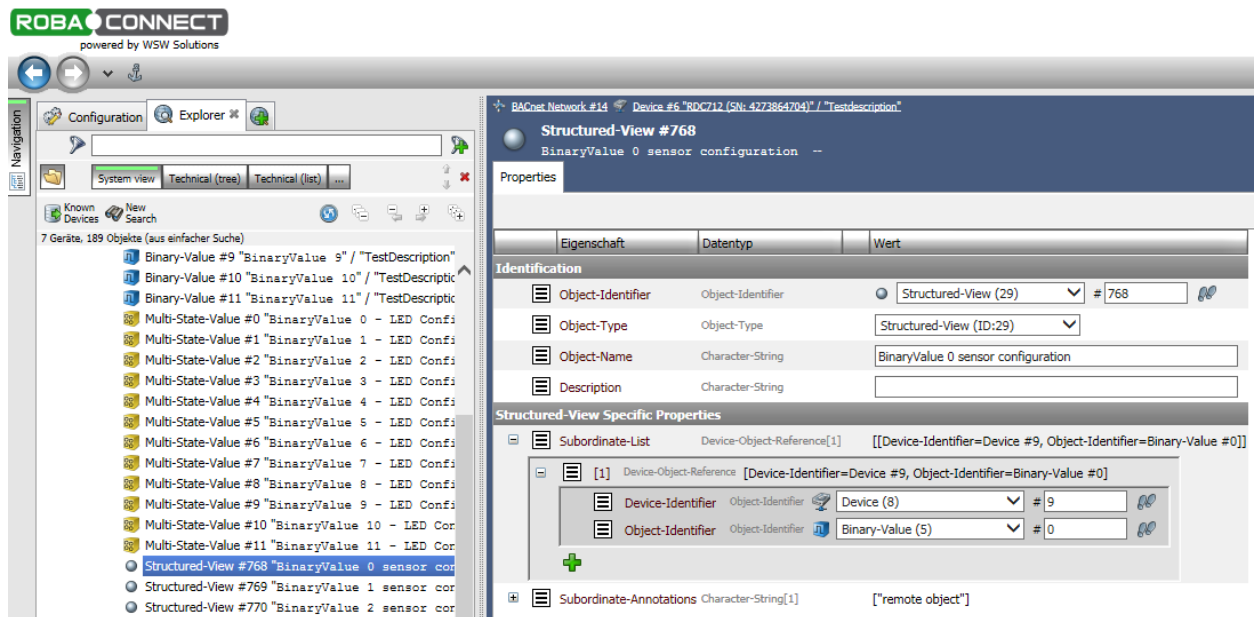


Abb. Structured-View

Siehe hierzu auch LED Konfiguration RDC712

Für jede der 12 Status LEDs gibt es ein „Structured-View“ Sensor Configuration Objekt (#768 bis #779)

In diesem Objekt wird der Zustand eines Datenpunktes (physikalisch oder virtueller) der Status LED zugeordnet. Diese zeigt dann den aktuellen Zustand im „Binary-Value“ (Present-Value als Inaktiv oder Aktiv) und in der eingestellten Farbe an.

**Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein !**

Einzutragen sind

<b>Device-Identifizier:</b>	<b>Device(8)</b>	<b>DeviceID (hier die 9)</b>
<b>Objekt-Identifizier:</b>	<b>Objekt-Type (hier Binary-Value)</b>	<b>Objekt-Number (hier die 0)</b>

Die Statusänderung (COV = Cange of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed\_COV“ oder „Unconfirmed\_COV“ übermittelt werden.

Ist die abonnierte Datenverbindung gestört oder fehlerhaft blinkt die entsprechende Statusanzeige gelb/rot.





## Modulbeschreibung RDC742 : Applikation Rauchschalter und Brandschutzklappe

Visualisierung von bis zu 12 Digitaleingängen als LED Anzeige mit Auswahl der anzuzeigenden Farbe grün/gelb/rot mit und ohne blinken.

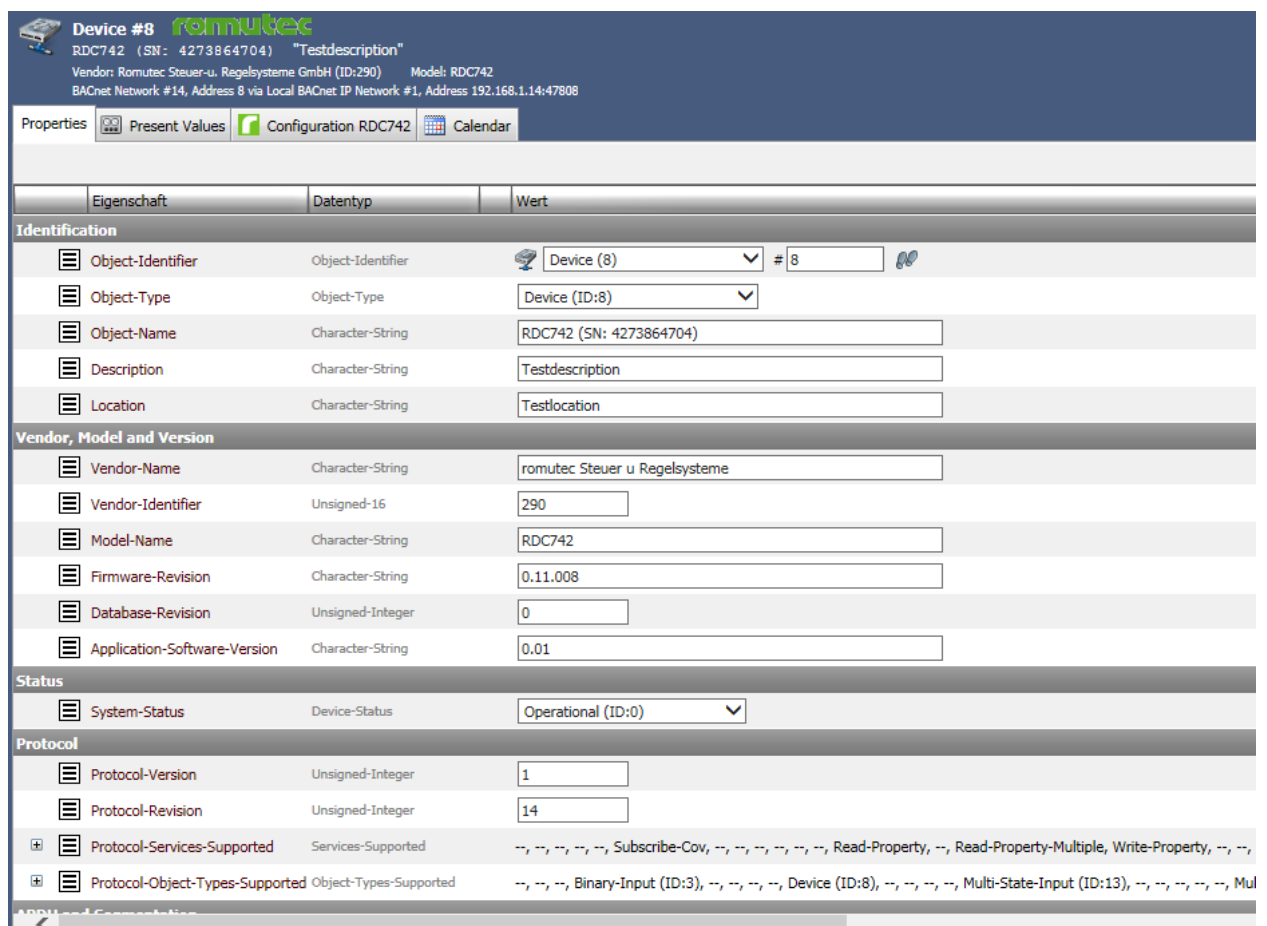
Das Modul besteht aus 4 Gruppen mit je einem Taster und drei mehrfarbigen LEDs.

Die Applikation bildet die Funktion eines Rauchmelders(RM) mit Brandschutzklappe (BSK) ab. Es werden die Meldungen des Rauchmelders sowie die Meldungen der BSK dargestellt. Mit dem Taster kann ein Testlauf ausgelöst werden.

Die anzuzeigenden Statusmeldungen werden von den jeweiligen physikalischen oder virtuellen Datenpunkten mittels BACnet Objekte „Structured-View“ Objekt abonniert. Die Statusänderung (COV = Cange of Value) und Zustände am physikalischen Eingang werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed\_COV“ oder „Unconfirmed\_COV“ übermittelt werden.

## Konfiguration des RDC742:

### Device Objekte:



**Device #8 romutec**  
 RDC742 (SN: 4273864704) "Testdescription"  
 Vendor: Romutec Steuer-u. Regelsysteme GmbH (ID:290) Model: RDC742  
 BACnet Network #14, Address 8 via Local BACnet IP Network #1, Address 192.168.1.14:47808

Properties Present Values Configuration RDC742 Calendar

Eigenschaft	Datentyp	Wert
<b>Identification</b>		
Object-Identifier	Object-Identifier	Device (8) # 8
Object-Type	Object-Type	Device (ID:8)
Object-Name	Character-String	RDC742 (SN: 4273864704)
Description	Character-String	Testdescription
Location	Character-String	Testlocation
<b>Vendor, Model and Version</b>		
Vendor-Name	Character-String	romutec Steuer u Regelsysteme
Vendor-Identifier	Unsigned-16	290
Model-Name	Character-String	RDC742
Firmware-Revision	Character-String	0.11.008
Database-Revision	Unsigned-Integer	0
Application-Software-Version	Character-String	0.01
<b>Status</b>		
System-Status	Device-Status	Operational (ID:0)
<b>Protocol</b>		
Protocol-Version	Unsigned-Integer	1
Protocol-Revision	Unsigned-Integer	14
Protocol-Services-Supported	Services-Supported	Subscribe-Cov, Read-Property, Read-Property-Multiple, Write-Property, ...
Protocol-Object-Types-Supported	Object-Types-Supported	Binary-Input (ID:3), Device (ID:8), Multi-State-Input (ID:13), ...

Abb. Decive Objekte

Ab Werk ist die Instance Nummer (InstanceNumber) gleich der Seriennummer. Diese kann jedoch geändert werden.

Für die Bezeichnung (Name) und Ort (Location) können Texte bis zu 63 Zeichen hinterlegt werden.

## Binary-Input Objekt:

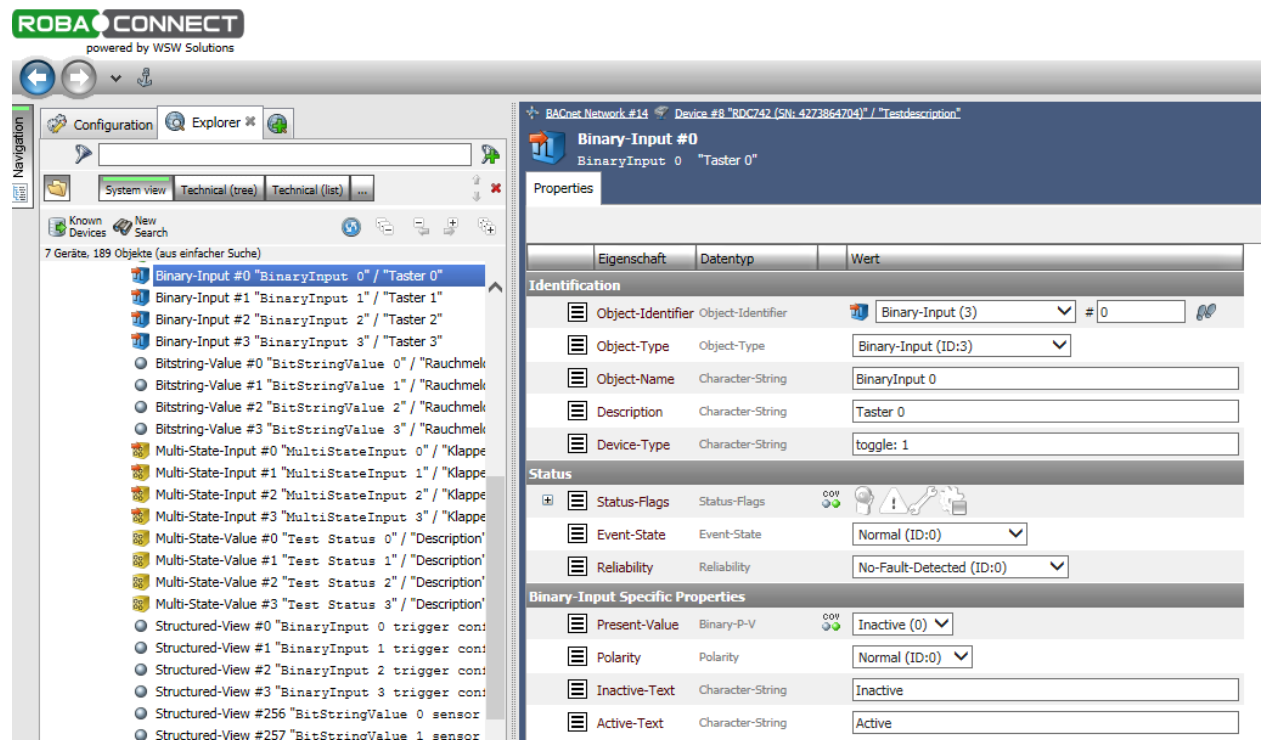


Abb. Binary-Input

Für jeden der 4 Taster gibt es einen „Binary-Input“ (#0-#3).

Jeder Taster kann bis zu 32 BACnet Objekte (Binary Output oder Binary Value) beschreiben. Diese werden im „Structured-View“ Objekt (#0-#3) eingetragen. SubScribe\_COV ist verfügbar.

Für die Datenpunkte (Objekt-Name) und die Beschreibung (Description) können Texte bis zu 63 Zeichen hinterlegt werden.

Im Profil-Name ist die Farbe bzw. Eigenschaft der Status-LED hinterlegt. Dies kann über den zugehörigen Multi-State-Value (#x) verändert werden.

Der Status des Tasters wird im „Present-Value“ als Inactive (0) oder Aktiv (1) angezeigt. Über die „Polarity“ kann der Status invertiert werden.

Mit dem „Device Type“ kann eine Toggle-Funktion eingestellt werden.

Toggle: 0 =  
„Present-Value“ wechselt von 0 nach 1

Toggle: 1 =  
„Present-Value“ wechselt von 1 nach 0

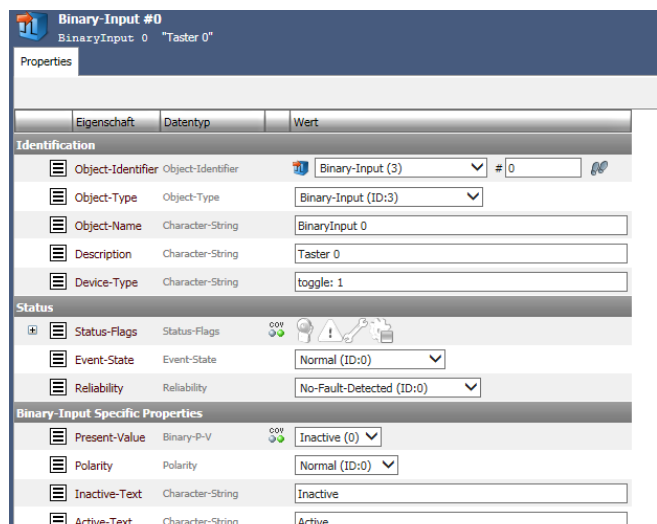


Abb. Toggle

## Structured-View Objekt:



powered by WSIW Solutions

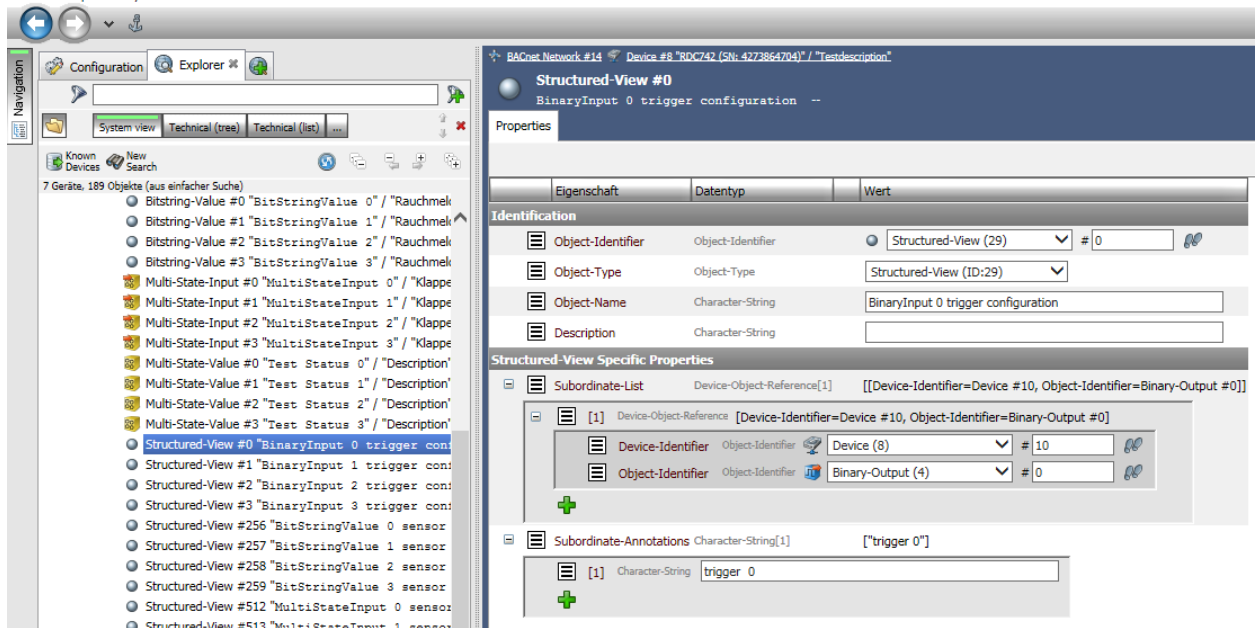


Abb. Structured-View Objekt

Für jeden der 4 Taster gibt es ein „Structured-View“ (trigger configuration) Objekt (#0 bis #3)

Es kann für jeden Taster ein Ziel (Einträge) definiert werden.

**Hierbei ist zu beachten, dass die Anlage vor Auslösung des Testbetriebes abgeschaltet wird!**

In diesem Objekt wird das Device und die Datenpunkt (physikalisch oder virtueller) des zu übersteuernden Gerätes zugeordnet.

**Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein !**

Einzutragen sind

<b>Device-Identifier:</b>	<b>Device(8)</b>	<b>DeviceID (hier die 10)</b>
<b>Objekt-Identifier:</b>	<b>Objekt-Type (hier Binary-Output)</b>	<b>Objekt-Number (hier die 0)</b>
	<b>[Binary-Output oder Binary-Value]</b>	

Die Statusänderung (COV = Cange of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed\_COV“ oder „Unconfirmed\_COV“ übermittelt werden.

Ist die Datenverbindung gestört oder fehlerhaft blinkt die entsprechende Statusanzeige gelb/rot. Wenn eine Aktion an die Device nicht übergeben werden kann, wird dieses an der Statusanzeige unter den Taster durch eine LED rot signalisiert.

# Systembeschreibung RDC 700 Serie

## Test Status Multistate-Value Objekt:

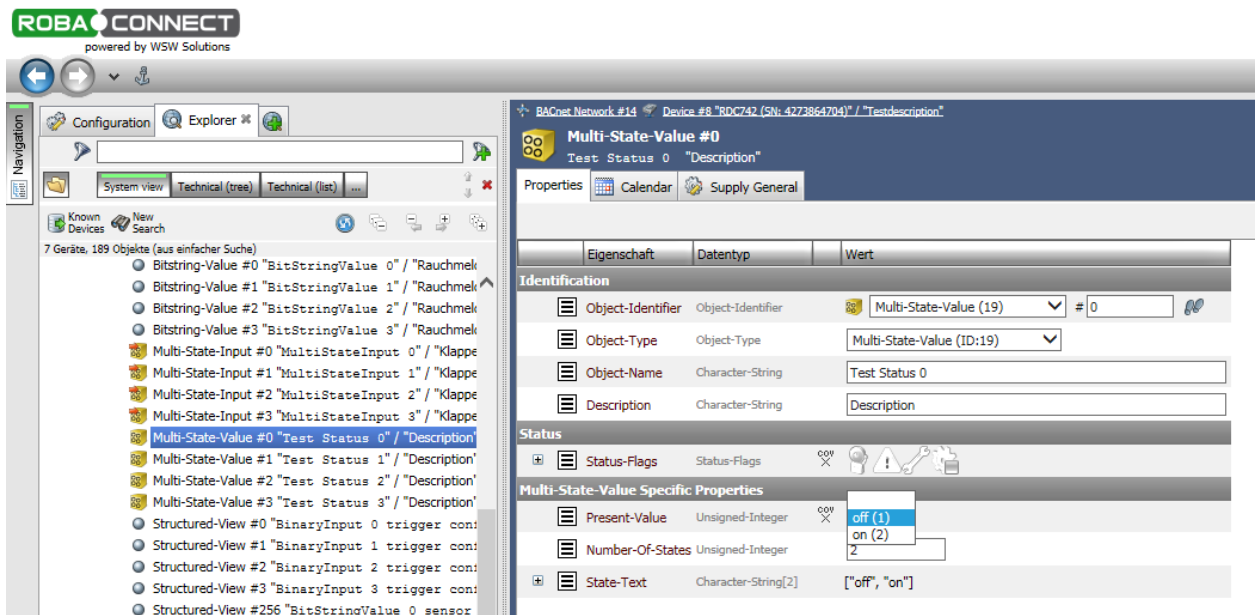


Abb. Multistate-Value zum Anlagen Status

Für jede der 4 Gruppen gibt es einen „Multi-State-Value“ Test Status (#0 bis #11).

Hier wird der Applikation vorgegeben, dass die Anlage „EIN (on)“ oder „AUS (off)“ ist.

Einstellungen im „Multi-State-Value“ (#0 bis -#3) hier mit dem „Present-Value“ off oder on „State Text“ :                    1 = off                    2 = on

Auswirkungen :

Überwachung der BSK findet bei Anlage EIN (on) laut Tabelle statt. Fahrbetrieb und Schmelzlot wird überwacht. Bei Fehler und Alarm wird das In\_Alarm Status Flag gesetzt.

Wird bei Anlage AUS (off) die Klappe gefahren, wird nur das Overridden Flag gesetzt.

Klappe ZU	Klappe AUF	Anlage System Status MSV	Klappen Status MSI Flap State	Klappen Status MSI Input Status Flags
BI Flap closed	BI Flap open		closed-movig-open- fusable_link	In_Alarm-Fault-Overridden-Out of Service
0	0	off / AUS	moving	Overridden
0	1	off / AUS	open	Overridden
1	0	off / AUS	closed	Overridden
1	1	off / AUS	fault	In_Alarm
0	0	on / EIN	fusable_link	In_Alarm
0	1	on / EIN	open	
1	0	on / EIN	fusable_link	In-Alarm
1	1	on / EIN	fault	In_Alarm

## BSK Status Structured-View Objekt:

**Structured-View #512**  
MultiStateInput 0 sensor configuration --

Properties

Eigenschaft	Datentyp	Wert
<b>Identification</b>		
Object-Identifier	Object-Identifier	Structured-View (29) # 512
Object-Type	Object-Type	Structured-View (ID:29)
Object-Name	Character-String	MultiStateInput 0 sensor configuration
Description	Character-String	
<b>Structured-View Specific Properties</b>		
Subordinate-List Device-Object-Reference[2] [[Device-Identifier=Device #10, Object-Identifier=Binary-Input #4]		
[1] Device-Object-Reference [Device-Identifier=Device #10, Object-Identifier=Binary-Input #4]		
Device-Identifier	Object-Identifier	Device (8) # 10
Object-Identifier	Object-Identifier	Binary-Input (3) # 4
[2] Device-Object-Reference [Device-Identifier=Device #10, Object-Identifier=Binary-Input #7]		
Device-Identifier	Object-Identifier	Device (8) # 10
Object-Identifier	Object-Identifier	Binary-Input (3) # 7
Subordinate-Annotations Character-String[2] ["flap closed", "flap open"]		
[1] Character-String flap closed		
[2] Character-String flap open		

Abb. Structured-View Objekt BSK Status

Für jede der 4 Gruppen gibt es ein „Structured-View“ (Multistate Input 0 sensor configuration) Objekt (#512 bis #515). Mit diesem Objekt wird das Device und die Datenpunkte (physikalisch oder virtueller) des zu überwachenden Gerätes, hier eine BSK zugeordnet.

**Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein!**

Einzutragen sind

Klappen-Meldungen Endlagenschalter „ZU“ (flap closed)

1	<b>Device-Identifier:</b>	<b>Device(8)</b>	<b>DeviceID (hier die 10)</b>
1	<b>Objekt-Identifier:</b>	<b>Objekt-Type (hier Binary-Input)</b>	<b>Objekt-Number (hier die 4)</b>

Klappen- Meldung Endlagenschalter „AUF“ (flap open)

2	<b>Device-Identifier:</b>	<b>Device(8)</b>	<b>DeviceID (hier die 10)</b>
2	<b>Objekt-Identifier:</b>	<b>Objekt-Type (hier Binary-Input)</b> [Binary-Output oder Binary-Value]	<b>Objekt-Number (hier die 7)</b>

Die Statusänderung (COV = Cange of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed\_COV“ oder „Unconfirmed\_COV“ übermittelt werden.

# Systembeschreibung RDC 700 Serie

## Statusmeldungen an den LED Gruppen

Ist die abonnierte Datenverbindung gestört oder fehlerhaft blinkt die entsprechende Statusanzeige gelb/rot.

Wenn eine Aktion an das Device nicht übergeben werden kann, wird dieses an der Statusanzeige unter den Taster durch eine LED rot signalisiert.

Diese Zustände werden an den jeweiligen LEDs zur Anzeige gebracht:

<u>Gruppen</u>	<u>Bezeichnung</u>	<u>Status</u>	<u>LED-Status</u>	<u>Funktion</u>
<i>LED Oben</i>	<i>Rauchmelder LED verschmutzt</i>	<i>OK</i>	<i>Aus</i>	
		<i>verschmutzt</i>	<i>gelb</i>	<i>RM verschmutzt</i>
<i>LED Mitte</i>	<i>Rauchmelder LED Störung</i>	<i>OK</i>	<i>Aus</i>	
		<i>Störung elektrisch</i>	<i>gelb blinken</i>	<i>RM gestört</i>
		<i>Ausgelöst Rauch</i>	<i>rot blinken</i>	<i>RM ausgelöst</i>
<b>LED Unten</b>	<b>Klappe LED ZU</b>	Offen	grün	BSK Offen
		Zu	Aus	BSK geschlossen
		Schmelz Lot	rot	BSK gefallen
		Unterwegs (Fährt)	gelb blinken	BSK in Zwischenstellung
		Störung	gelb-rot blinken	

Abb. LEDs je Gruppe

## Klappe Multi-State-Input Objekt:

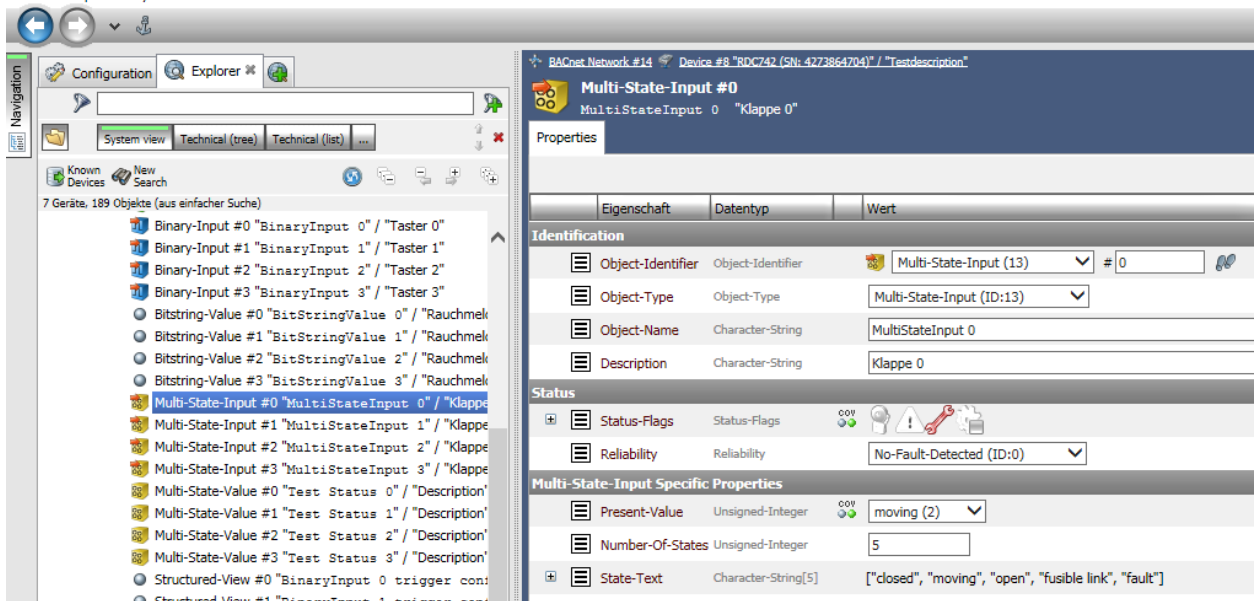


Abb. Multi-State-Value Objekt

Für jede der 4 Gruppen gibt es einen „Multi-State-Input“ Klappen Status (#0 bis #3).

Hier wird der Status der Klappen angezeigt. Die Applikation vergleicht dort die im Structured View (#512 bis #515) eingestellten Werte an, bzw. Wertet diese entsprechen aus.

Anzeige Optionen im „Multi-State-Input“ (#0 bis -#3) hier mit dem „Present-Value“ (1-5):

State Texte:     1 = closed     2 = moving     3 = open     4 = fusible link     5 = fault

### Auswirkungen :

Überwachung der BSK findet bei Anlage EIN (on) laut Tabelle statt. Fahrbetrieb und Schmelzlot wird überwacht. Bei Fehler und Alarm wird das In\_Alarm Status Flag gesetzt.

Wird bei Anlage AUS (off) die Klappe gefahren, wird nur das Overridden Flag gesetzt.

Klappe ZU	Klappe AUF	Anlage System Status MSV	Klappen Status MSI Flap State	Klappen Status MSI Input Status Flags
BI Flap closed	BI Flap open		closed-movig-open- fusable_link	In_Alarm-Fault-Overridden-Out of Service
0	0	off / AUS	moving	Overridden
0	1	off / AUS	open	Overridden
1	0	off / AUS	closed	Overridden
1	1	off / AUS	fault	In_Alarm
0	0	on / EIN	fusable_link	In_Alarm
0	1	on / EIN	open	
1	0	on / EIN	fusable_link	In-Alarm
1	1	on / EIN	fault	In_Alarm



## Rauchscharter Status Structured-View Objekt:

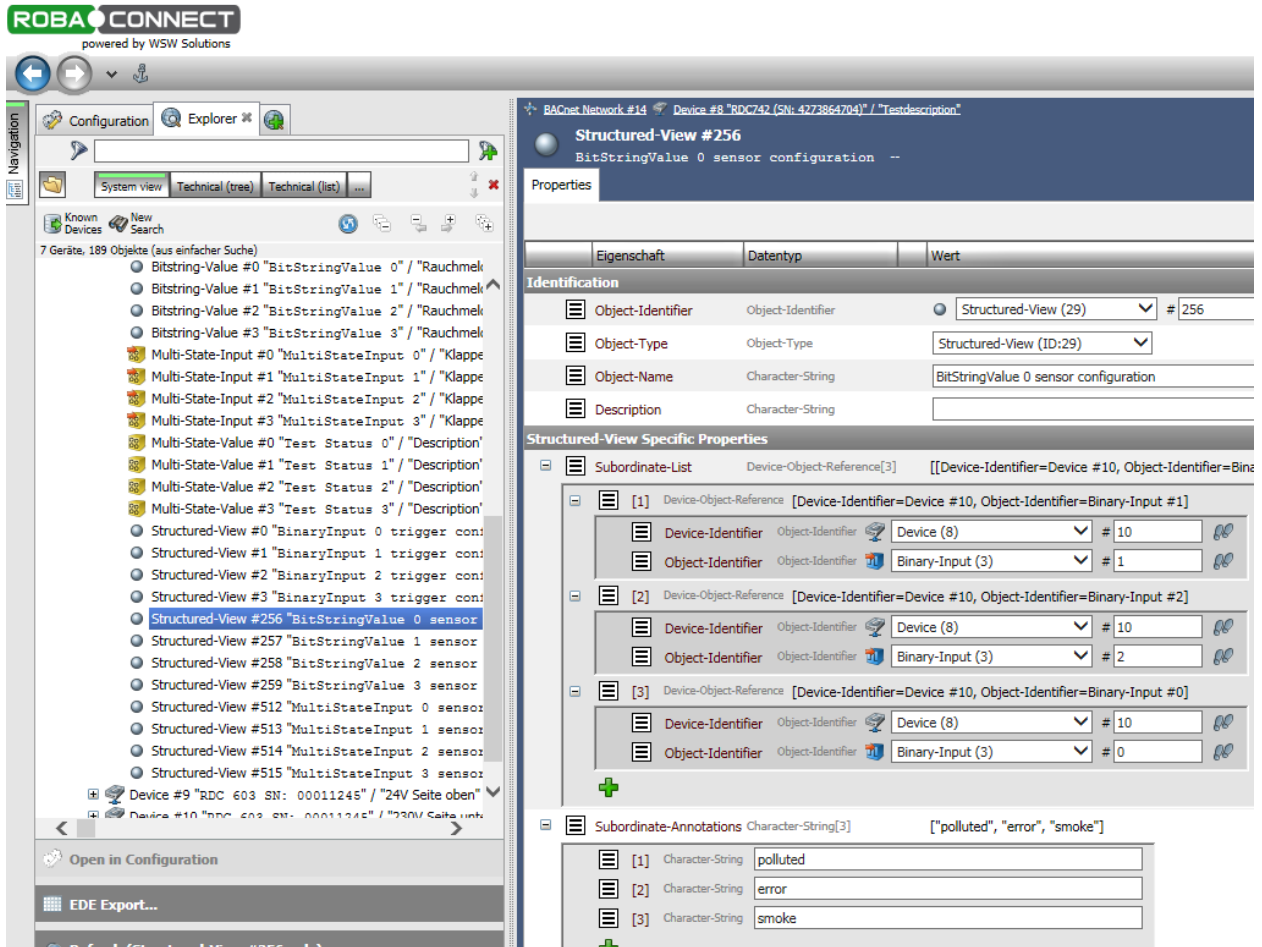


Abb. Structured-View Objekt Rauchscharter Status

Für jede der 4 Gruppen gibt es ein „Structured-View“ (BitStringValue 0 sensor configuration) Objekt (#256 bis #259). Mit diesem Objekt wird das Device und die Datenpunkte (physikalisch oder virtueller) des zu überwachenden Gerätes, hier ein Rauchscharter zugeordnet.

**Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein !**

Einzutragen sind

Rauchscharter-Meldungen „Verschmutzt“ (polluted)

- |   |                      |  |                                   |
|---|----------------------|--|-----------------------------------|
| 1 | Device-Identifizier: | <b>Device(8)</b>                       | <b>DeviceID (hier die 10)</b>     |
| 1 | Objekt-Identifizier: | <b>Objekt-Type (hier Binary-Input)</b> | <b>Objekt-Number (hier die 1)</b> |

Rauchscharter-Meldungen „Störung“ (error)

- |   |                      |  |                                   |
|---|----------------------|--|-----------------------------------|
| 2 | Device-Identifizier: | <b>Device(8)</b>                       | <b>DeviceID (hier die 10)</b>     |
| 2 | Objekt-Identifizier: | <b>Objekt-Type (hier Binary-Input)</b> | <b>Objekt-Number (hier die 2)</b> |

Rauchscharter-Meldungen „Alarm“ (smoke)

- |   |                      |  |                                   |
|---|----------------------|--|-----------------------------------|
| 3 | Device-Identifizier: | <b>Device(8)</b>                       | <b>DeviceID (hier die 10)</b>     |
| 3 | Objekt-Identifizier: | <b>Objekt-Type (hier Binary-Input)</b> | <b>Objekt-Number (hier die 0)</b> |

Die Statusänderung (COV = Cange of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service „Subscribe\_COV“ übermittelt. Es kann dieser Zustand als „Confirmed COV“ oder „Unconfirmed COV“ übermittelt werden.

Ist die Datenverbindung gestört oder fehlerhaft blinkt die entsprechende Statusanzeige gelb/rot. Wenn eine Aktion an das Device nicht übergeben werden kann, wird dieses an der Statusanzeige unter den Taster durch eine LED rot signalisiert.

## Rauchschalter Status BitStringValue Objekt:



powered by WSW Solutions

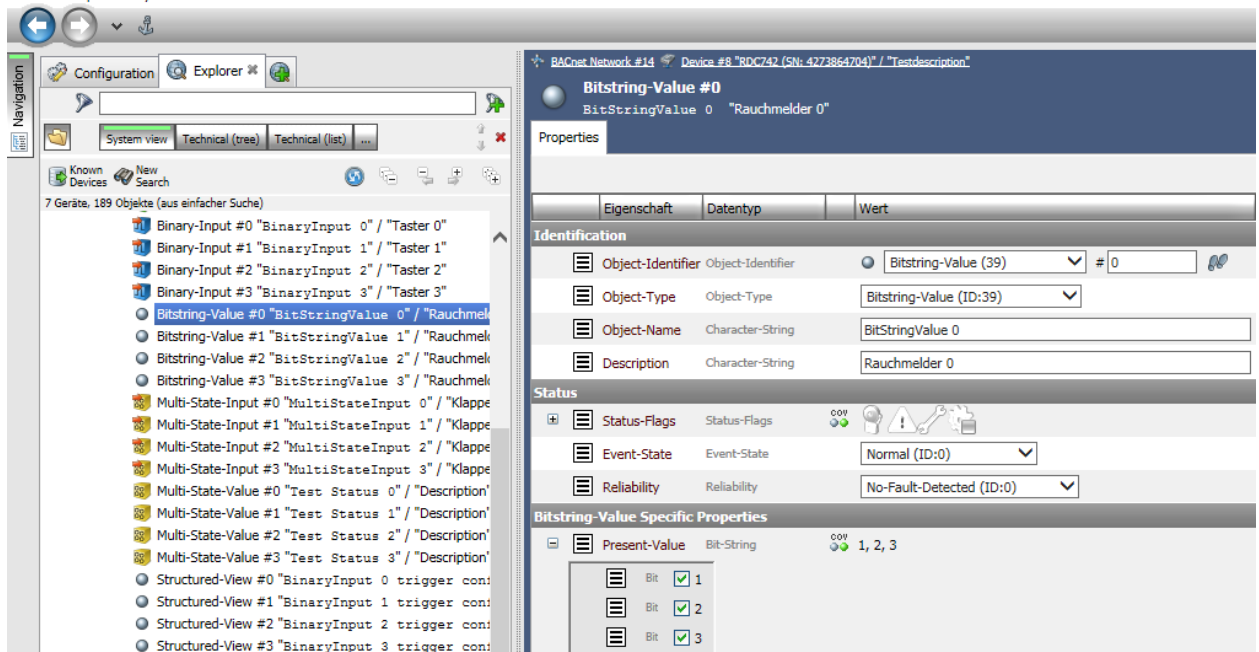


Abb. BitStringValue Objekt Rauchschalter Status

Für jede der 4 Gruppen gibt es ein „BitString-Value Objekt“: (BitStringValue 0/Rauchmelder) #0 bis #3).

Hier wird der Status des Rauchschalters angezeigt. Die Applikation ordnet die im Structured View (#256 bis #259) eingestellten Werte an, bzw. Wertet diese entsprechen aus.

Anzeige Optionen im „BitStringValue“ (#0 bis #-3) hier mit dem „Present-Value“ (1-3):

State Texte: 1 = „Verschmutzt“ (polluted) 2 = „Störung“ (error) 3 = „Alarm“ (smoke)

**Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein !**

Diese Zustände werden an den jeweiligen LEDs zur Anzeige gebracht:

<u>Gruppen</u>	<u>Bezeichnung</u>	<u>Status</u>	<u>LED-Status</u>	<u>Funktion</u>
<b>LED Oben</b>	<b>Rauchmelder LED verschmutzt</b>	OK	Aus	
		verschmutzt	gelb	RM verschmutzt
<b>LED Mitte</b>	<b>Rauchmelder LED Störung</b>	OK	Aus	
		Störung elektrisch	gelb blinken	RM gestört
		Ausgelöst Rauch	rot blinken	RM ausgelöst
<i>LED Unten</i>	<i>Klappe LED ZU</i>	<i>Offen</i>	<i>grün</i>	<i>BSK Offen</i>
		<i>Zu</i>	<i>Aus</i>	<i>BSK geschlossen</i>
		<i>Schmelz Lot</i>	<i>rot</i>	<i>BSK gefallen</i>
		<i>Unterwegs (Fahrt)</i>	<i>gelb blinken</i>	<i>BSK in Zwischenstellung</i>
		<i>Störung</i>	<i>gelb-rot blinken</i>	

Abb. LEDs je Gruppe

# Systembeschreibung RDC 700 Serie

---

## Technische Daten (Planung)

<b>Versorgungsspannung</b>	24 VDC/AC, $\pm 10\%$
<b>Stromaufnahme</b>	max. 100 mA
<b>Prozessor</b>	
Typ	Microchip PIC32
Taktfrequenz	32 MHz
<b>Bus-Schnittstelle</b>	
Anschluß	RS485 (EIA 485)
Bus Abschluß :	über 120 Ohm Widerstand
<b>Speicher</b>	
Flash	256 k
RAM	256 k
<b>Protokoll</b>	BACnet MS/TP
<b>Digitale Ein-/Ausgänge</b>	extern über BACnet MS/TP-BUS
<b>Umgebungsbedingungen</b>	
Betriebstemperatur	0...50°C
Transport- und Lager- temperatur	-35...+70°C
Relative Feuchte	5...95%, nicht kondensierend
<b>Schutzart</b>	IP 40
<b>Abmessungen</b>	
19"-Geräte	Breite 8 TE, Höhe 3 HE, Einbautiefe < 40 mm Gehäuse B x H x T = 130 x 40 x 35 mm
<b>Anschlussadapter</b>	Breite 4 TE, Höhe 3 HE, Einbautiefe < 50 mm Gehäuse B x H x T = 130 x 20 x 45 mm
<b>CE-Konformität</b>	EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD Kontaktentladung 8 kV / Luftentladung 8 kV EN 61000-4-5 / IEC 801-5 Surge-Prüfung Versorgungspg. AC 4 kV, DC 0,5 kV Signalleitungen 2 kV EN 61000-4-4 / IEC 801-4 Burst-Prüfung

## romutec verhandene BACnet Objekte und Property:

### BACnet Objekte nach ANSI/ASHRAE Standard 135-2012

#### Gerät/Device

DEVICE	<b>Object_Identifier</b>	R	
DEVICE	<b>Object_Name</b>	R/W	
DEVICE	<b>Object_Type</b>	R	
DEVICE	<b>System_Status</b>	R	
DEVICE	<b>Vendor_Name</b>	R	
DEVICE	<b>Vendor_Identifier</b>	R	
DEVICE	<b>Model_Name</b>	R	
DEVICE	<b>Firmware_Revision</b>	R	
DEVICE	<b>Application_Software_Version</b>	R	
DEVICE	<b>Location</b>	R/W	
DEVICE	<b>Description</b>	W	
DEVICE	<b>Protocol_Version</b>	R	
DEVICE	<b>Protocol_Revision</b>	R	
DEVICE	<b>Protocol_Services_Supported</b>	R	
DEVICE	<b>Protocol_Object_Types_Supported</b>	R	
DEVICE	<b>Object_List</b>	R	
DEVICE	<b>Max_APDU_Length_Accepted</b>	R	480 Zeichen
DEVICE	<b>Segmentation_Supported</b>	R	No Segmentation
DEVICE	<b>APDU_Segment_Timeout</b>	R/W	
DEVICE	<b>APDU_Timeout</b>	R/W	
DEVICE	<b>Number_Of_APDU_Retries</b>	R	
DEVICE	<b>Time_Synchronization_Recipients</b>		
DEVICE	<b>Max_Master</b>	R/W	
DEVICE	<b>Max_Info_Frames</b>	R/W	
DEVICE	<b>Database_Revision</b>	R	

## Analogeingang / Analog Input

Analog Input	<b>Object_Identifier</b>	R	
Analog Input	<b>Object_Name</b>	R/W	
Analog Input	<b>Object_Type</b>	R	
Analog Input	<b>Present_Value</b>	R <sup>1</sup>	
Analog Input	<b>Description</b>	W	
Analog Input	<b>Device_Type</b>	R/W	Konfiguration AI
Analog Input	<b>Status_Flags</b>	R	
Analog Input	<b>Event_State</b>	R	
Analog Input	<b>Reliability</b>	R	
Analog Input	<b>Out_Of_Service</b>	W/R	
Analog Input	<b>Min_Pres_Value</b>	R/W	
Analog Input	<b>Max_Pres_Value</b>	R/W	
Analog Input	<b>COV_Increment</b>	W/R	

## Analogausgang / Analog Output

Analog Output	<b>Object_Identifier</b>	R	
Analog Output	<b>Object_Name</b>	R/W	
Analog Output	<b>Object_Type</b>	R	
Analog Output	<b>Present_Value</b>	W	
Analog Output	<b>Description</b>	W/R	
Analog Output	<b>Device_Type</b>	W/R	
Analog Output	<b>Status_Flags</b>	R	
Analog Output	<b>Reliability</b>	R	
Analog Output	<b>Out_Of_Service</b>	R	
Analog Output	<b>Units</b>	R/W	
Analog Output	<b>Min_Pres_Value</b>	R/W	
Analog Output	<b>Max_Pres_Value</b>	R/W	
Analog Output	<b>Priority_Array</b>	R	
Analog Output	<b>Relinquish_Default</b>	R	
Analog Output	<b>COV_Increment</b>	W	

## Digitalausgang / Binary Input

Binary Output	<b>Object_Identifier</b>	R	
Binary Output	<b>Object_Name</b>	R/W	
Binary Output	<b>Object_Type</b>	R	
Binary Output	<b>Present_Value</b>	R	
Binary Output	<b>Description</b>	W/R	
Binary Output	<b>Status_Flags</b>	R	
Binary Output	<b>Event_State</b>	R	
Binary Output	<b>Reliability</b>	R	
Binary Output	<b>Out_Of_Service</b>	R	
Binary Output	<b>Polarity</b>	R/W	
Binary Output	<b>Inactive_Text</b>	R/W	
Binary Output	<b>Active_Text</b>	R/W	
Binary Output	<b>Priority_Array</b>	R	
Binary Output	<b>Relinquish_Default</b>	R	

## Digitaleingang / Binary Input

Binary Input	<b>Object_Identifier</b>	R	
Binary Input	<b>Object_Name</b>	R/W	
Binary Input	<b>Object_Type</b>	R	
Binary Input	<b>Present_Value</b>	R	
Binary Input	<b>Description</b>	W/R	64 Zeichen
Binary Input	<b>Device_Type</b>	R/W	Konfiguration
Binary Input	<b>Status_Flags</b>	R	
Binary Input	<b>Event_State</b>	R	
Binary Input	<b>Reliability</b>	R	
Binary Input	<b>Out_Of_Service</b>	W	
Binary Input	<b>Polarity</b>	R/W	
Binary Input	<b>Inactive_Text</b>	W/R	
Binary Input	<b>Active_Text</b>	W/R	

# Systembeschreibung RDC 700 Serie

## Digital Wert / Binary Value

Binary Value	<b>Object_Identifier</b>	R	
Binary Value	<b>Object_Name</b>	R/W	
Binary Value	<b>Object_Type</b>	R	
Binary Value	<b>Present_Value</b>	R	
Binary Value	<b>Description</b>	W/R	
Binary Value	<b>Status_Flags</b>	R	
Binary Value	<b>Reliability</b>	R	
Binary Value	<b>Out_Of_Service</b>	W	
Binary Value	<b>Inactive_Text</b>	R/W	
Binary Value	<b>Active_Text</b>	R/W	
Binary Value	<b>Profile_Name</b>	R/W	Konfiguration Farbe LED

## Mehrstufiger Wert - Multistate Value

Multistate Value	<b>Object_Identifier</b>	R	
Multistate Value	<b>Object_Name</b>	R/W	
Multistate Value	<b>Object_Type</b>	R	
Multistate Value	<b>Present_Value</b>	R	
Multistate Value	<b>Description</b>	W/R	
Multistate Value	<b>Status_Flags</b>	R	
Multistate Value	<b>Out_Of_Service</b>	W/R	
Multistate Value	<b>Number_Of_States</b>	R	
Multistate Value	<b>State_Text</b>	R	

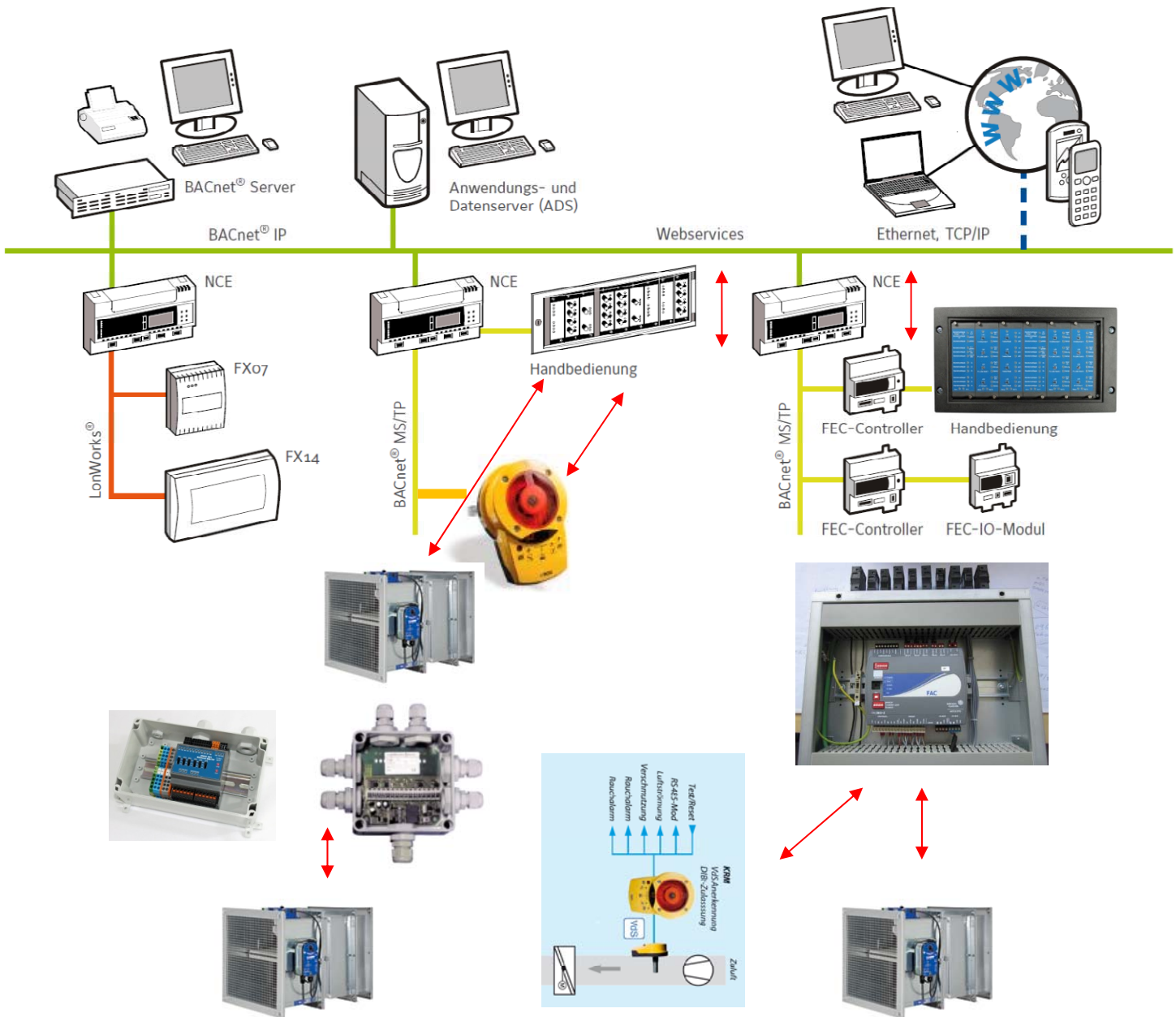
R = read; W=write

## Structured View - strukturierte Sichten auf Objekte zu definieren

Structured View	<b>Object_Identifier</b>	R	
Structured View	<b>Object_Name</b>	R	
Structured View	<b>Object_Type</b>	R	
Structured View	<b>Description</b>	O	
Structured View	<b>Node_Type</b>	R	
Structured View	<b>Node_Subtype</b>	O	
Structured View	<b>Subordinate_List</b>	R/W	
Structured View	<b>Subordinate_Annotations</b>	O	
Structured View	<b>Profile_Name</b>	O	

# Systembeschreibung RDC 700 Serie

## Als Beispiel mit Geräten der Firma Johnson Controls Situationsbeschreibung:



Die Funktionen im BACnet Netzwerk werden über den NCE realisiert. D.h. alle Teilnehmer und Datenpunkte am MS/TP-Bus werden über eine Zuordnung im NCE verbunden. Eine Weiterleitung der Daten erfolgt über den NCE an die Leitstation. Selbst wenn wie im rechten Teil der Rauchmelder und die BSK's auf den FEC aufgeschaltet werden, ist nur eine Verbindung zur Leittechnik über den NCE möglich. Um auch eine Funktion zu gewährleisten, wenn der DDC-Controller NCE ausfällt, wird eine Zuordnung im Anzeige und Bedienmodul erfolgen. Funktion LVB DIN ISO16484 und VDI3814. Eine Alarm- und Ereignismeldung ist nicht nötig, diese wird im NCE oder FEC aufbereitet. Optional ist eine Variante verfügbar, dort werden die Datenpunkte über ein RDC-Gerät erfasst, und über den BACnet MS/TP-Bus zum Anzeigemodul gesendet.

(Quelle: Johnson Controls, Trox, Oppermann)