

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.











Inhalt

BACnet MS/TP Anzeigesystem	3
Allgemeines	3
Funktionen und Bedienung	4
Beschreibung der Bedien- und Anzeige-Module	4
Modultypen	4
Module mit ausschließlich Grundfunktionen	5
Module mit spezieller Applikation für BSK und Rauchmelder	5
Montage	7
Elektrischer Anschluss	8
Adressen	9
Überwachung Modulstatus	10
BACnet Objekte	12
Allgemeine BACnet Objekte	12
Multi State Value Objekt #65280: Baudrate for MS/TP Bus	12
Binary Value Objekt #65280: Lampentest	12
Modulbeschreibung RDC712	13
Konfiguration des RDC712	
Device Objekt	
Binary-Value Objekt	
Structured-View Objekt	
Multistate-Value Objekt	
Modulbeschreibung RDC741	
Konfiguration des RDC741	20
Device Objekt	20
Binary-Input Objekt	
Binary-Value Objekt	
Structured-View Objekt	
Binary-Value Objekt	
Konfiguration der Status LED (Multistate-Value Objekt)	
Structured-View Objekt	
Modulbeschreibung RDC742	
Konfiguration des RDC742	
Device Objekt	
Binary-Input Objekt	
Structured-View Objekt	31
Test Status Multistate-Value Objekt	
BSK Status Structured-View Objekt	33
Klappe Multi-State-Input Objekt	35
Rauchschalter Status Structured-View Objekt	36
Rauchschalter Status BitStringValue Objekt	
Modulbeschreibung RDC743 und RDC744	
Konfiguration des RDC743 und RDC744	
Analog-Input Objekte Klappen-Fahrzeiten	
Multistate Value Objekt Klappen-Status	
Structured-View Objekt: Rückmeldungen und Abschaltungen	
Modulbeschreibung RDC745	
Konfiguration des RDC745	
Analog Input Objekt "Value Age"	42
Binary-Input Objekt	
Binary-Value Objekt	44
Structured-View Objekt Taster	
Rauchschalter Status Structured-View Objekt	46
Rauchschalter Status BitStringValue Objekt	
Technische Daten	
romutec vorhandene BACnet Objekte und Property	



BACnet MS/TP Anzeigesystem

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

Allgemeines

Die Module der RDC700-Serie dienen zur Anzeige von Betriebszuständen bzw. Klappenstellungen, und zur Anforderung von Selbsttestroutinen über mehrfarbige LEDs 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 Industrie-Anwendungen 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 mit jedem handelsüblichen BACnet Explorer möglich, der die verwendeten BACnet Objekte unterstützt.

Funktionen und Bedienung

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



Beschreibung der Bedien- und Anzeige-Module

Die Geräte RDC700 besitzen keine physikalischen Datenpunkte. Die vorhandenen Taster und LEDs 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 erfolgt über das Objekt "Structured-View". Hierüber wird eine Verknüpfung zum physikalischen Datenpunkt oder einem Datenpunkt im Fremdgerät hergestellt. Durch die Zuordnung von "Device" und dem "Objekt" wird der "Present Value" abgefragt. Durch den aktivierten "CoV" findet eine kontinuierliche Überwachung statt.

Modultypen

Es stehen sowohl Modultypen zur Verfügung, die eine Applikation enthalten, welche Funktionen bereitstellt, die auf Systeme bestehend aus Brandschutzklappen und Rauchmeldern angepasst sind (wie z.B. die Laufzeitüberwachung der BSK) als auch Typen, die keine solche Applikation enthalten.

Werden die Module der RDC7xx-Serie in Verbindung mit RDC683 oder RDC883 und dem BSK-Connector eingesetzt, so befindet sich die "Intelligenz" mit den Funktionen zur Überwachung des BSK-Systems in den RDC683/883. Dadurch können alle Anzeige- und Bedienfunktionen mit den beiden Typen RDC712 und RDC741 realisiert werden. Sollen jedoch an Stelle der RDC683/883 auch Fremdgeräte eingebunden werden, so muss bei der RDC7xx-Serie auf die Typen RDC742 … RDC745 zurückgegriffen werden.



• Module mit ausschließlich Grundfunktionen

Anzeigemodul (12 LEDs): 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

• Module mit spezieller Applikation für BSK und Rauchmelder

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 und Rauchmelder

LED - Anzeige des RDC742 :

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

Bedien-u. Anzeigemodul: RDC743

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

Anzeigemodul:

RDC744

Visualisierung von Digital-Eingängen als LED mit Auswahl der anzuzeigenden Farbe grün/gelb/rot mit und ohne blinken Applikation für Brandschutzklappen mit Laufzeitüberwachung

Anzeigemodul:

RDC745

Visualisierung von Digital-Eingängen als LED mit Auswahl der anzuzeigenden Farbe grün/gelb/rot mit und ohne blinken Applikation für Rauchmelder



Taster 0 für Testfunktionen Anlage AUS Reset je nach Applikation Testlauf Störung Klappe oder Rauchmelder LEDs mehrfarbig Brandschutzklappen NOT Hand "AUF" Auf Zustandsanzeige Anlage AUS grün/gelb/rot Reset Taster Testlauf blinkend Störung für Testfunktionen oder Dauerlicht Brandschutzklappen Zu je nach Applikation: NOT Hand "ZU" Anlage Aus zentrale Steuerung aller Reset Testlauf Klappen Störung über einen Taster BSK Anlage AUS Zu **Testfunktion Rauchschalter** Ver-schmutz Reset Testlauf Störung Rauchmelder 2 BSK Anlage 2 Zu Status RX/TX RDC741/742/743 LEDs mehrfarbig BSK Anlage 1 - 101 O ZU Zustandsanzeige Brandschutzklappe Ogefallen grün/gelb/rot blinkend Brandschutzklappe Sefallen oder Dauerlicht Brandschutzklappe Ogefallen Brandschutzklappe 🔵 grün = Brandschutzklappe prot/grün Brandschutzklappe Zu Brandschutzklappe Zu Brandschutzklappe Zu Brandschutzklappe Zu Brandschutzklappe Zu Brandschutzklappe 🕥 Zu Status O RX/TX RDC712/744/745



Montage

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



Abb. Einbau im Trägerrahmen Breite 50TE

Elektrischer Anschluss



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 (8)

Unterstützte Baudraten:

1 = 9.600 bps; 4 = 57.600 bps; 7 = 230.400 bps;	2 = 19.200 bps; 5 = 76.800 bps; 8 = AUTO	3 = 38.400 bps; 6 = 115.200 bps;
Format	8, N, 1; MS/TP RS485	EIA-485



Adressen

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



Wert	HB Einstell	Einstell	LB Wert				
links	Wert	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	А	А	10				
176	В	В	11				
192	С	С	12				
208	D	D	13				
224	Е	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 entschieden, ob das Gerät als Master oder Slave betrieben wird. Im Slave-Betrieb ist ein BACnet-Master notwendig, um die Daten des RDCs 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.



Vorrausetzung 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 gedrückt ist.

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 und ein gültiger Wert gesendet wurde. Beim RDC 742 "gelb/rot" blinkend. Dann wird der eingestellte Wert angezeigt.

Default ist GRÜN. Es kann einige Sekunden dauern, bis der Prozess gestartet ist. Ändert sich allerdings der "Present-Value" des Datenpunktes, wird ein CoV gesendet und der Wert sofort übernommen.









BACnet Objekte

Allgemeine BACnet Objekte

Multi State Value Objekt #65280: Baudrate for MS/TP Bus

Mit diesem Objekt wird die Baudrate eingestellt und ausgewählt werden.

1 = 9.600 bps;	2 = 19.200 bps;	3 = 38.400 bps;
4 = 57.600 bps;	5 = 76.800 bps;	6 = 115.200 bps;
7 = 230.400 bps;	8 = AUTO	-

Defaulteinstellung ist 8 = AUTO, d.h. das RDC synchronisiert sich an der am BUS verfügbaren Geschwindigkeit auf.

Soll das System ausfallsicher sein, so ist die Baudrate über den Present-Value vorzugeben. Beim Neustart wird dann bei festeingestellter Baudrate der Token generiert, der die Kommunikation am BUS startet.

ACHTUNG: Wird die Baudrate fest eingestellt, ist das RDC nur noch mit dieser zu erreichen.

Binary Value Objekt #65280: Lampentest

Mit diesem Objekt kann ein Lampentest der LEDs am Gerät ausgelöst werden. Die LEDs leuchten so lange wie der Present-Value aktiv ist.

Es kann hierfür das RDC741 mit seinen Taster als Auslöseeinheit verwendet werden.

¢	A http://192.168.3.29/#exp,d17,bv65280,prop		- c × 企会等 (9
R			Version 1.2.1 (Build 160721 RO)
	🕽 🕞 👻 🚯 🕕 🚺 Sie arbeiten mit einer	30-Tage-Testlizenz. <u>mehr</u> 🕕 Ihr Browser hat schlechte Performance! <u>mehr</u> 🕺 Administrat	or ⊘ 13:34:56 Kontakt Hilfe English
Navigation	Konfiguration Coporer # Coporer # Second State (Sauch Technich (Sauch)	♦ Lobate B&Onet MS/TE-Netzwerk #5	- ×
22			
	8 Geräte, 394 Objekte (aus einfacher Suche)	Eigenschaft Datentyp Wert	
	Binary-Input #0 "BinaryInput 0" / "Taster 0"	Identifikation	
	Binary-Input #1 "BinaryInput 1" / "laster 1" Binary-Input #2 "BinaryInput, 2" / "Taster 2"	Object-Identifier Object-Identifier Image: Binary-Value (5) # 65280	
	Binary-Input #3 "BinaryInput 3" / "Taster 3"	Binary-Value (ID:5)	
	Binary-Value #65280 "LAMP TEST" / "Enable LAMP TEST N	Object-Name Character-String IAMP TEST	
	Multi-State-Value #0 "Flap 0" / "Description" Multi-State-Value #1 "Flap 1" / "Description"		
	Multi-State-Value #2 "Flap 2" / "Description"	Enable LAMP TEST Mode	
	Multi-State-Value #3 "Flap 3" / "Description"	Status	
	Multi-State-Value #4 "Flap 4" / "Description"	🖩 🔳 Status-Flags Status-Flags 🔆 🍟 🕧	
	Wulti-State-Value #5 "Flap 5" / "Description"	Reliability Reliability No-Fault-Detected (ID:0)	
	Multi-State-Value #7 "Flap 7" / "Description"	Binary-Value-spezifische Eigenschaften	
	Multi-State-Value #8 "Flap 8" / "Description"	Present-Value Binary-P-V X Inactive (0) V	
	Multi-State-Value #9 "Flap 9" / "Description"	Inactive-Text Character-String Inactive	
	Multi-State-Value #10 "Flap 10" / "Description"		
		Active-Text Character-Soling Active	
	2 Offnen in der Konfiguration		
	III EDE-Export		
	🛞 Objekt löschen (BACnet Delete-Object)		
	Binary-Value #65280 in die Datenpunkt-Registratur aufnehmen	Binary-Value #65280 LAMP	



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 "Structured-View" Objekt abonniert. Die Statusänderung (COV = Change of Value) und Zustände am physikalischen Eingang werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann als "Confirmed_COV" oder "Unconfirmed_COV" übermittelt werden.

Konfiguration des RDC712

Device Objekt

Properti	BACnet Network #14 Image: Stever-u. Regelsysteme GmbH (ID:290) Model: RDC712 BACnet Network #14, Address 6 via Local BACnet IP Network #1, Address 192.168.1.14:47808 Properties Image: Present Values Image: Configuration RDC712 Image: Configuration RDC712 Image: Configuration RDC712							
_	Eigenschaft	Datentyp	Wert					
Identifi	cation							
Ξ	Object-Identifier	Object-Identifier	🖅 Device (8) 💙 # 6 🔗					
	Object-Type	Object-Type	Device (ID:8)					
Ξ	Object-Name	Character-String	RDC712 (SN: 4273864704)					
Ξ	Description	Character-String	Testdescription					
Ξ	Location	Character-String	Testlocation					
Vendor,	Model and Version							
Ξ	Vendor-Name	Character-String	romutec Steuer u Regelsysteme					
Ξ	Vendor-Identifier	Unsigned-16	290					
Ξ	Model-Name	Character-String	RDC712					
Ξ	Firmware-Revision	Character-String	0.11.006					
Ξ	Database-Revision	Unsigned-Integer	0					
	Application-Software-Version	Character-String	0.01					
Status								
	System-Status	Device-Status	Operational (ID:0)					
Protoco								
Ξ	Protocol-Version	Unsigned-Integer	1					
Ξ	Protocol-Revision	Unsigned-Integer	14					
Ŧ	Protocol-Services-Supported	Services-Supported	,,,, Subscribe-Cov,,,,, Read-Property,, Read-Property-Multiple,					
•	Protocol-Object-Types-Supporter	d Object-Types-Supported	,,,, Binary-Value (ID:5),,, Device (ID:8),,,,,,, Mul					
<	d Commetation							

Abb. Device Objekt

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

R						
	○ • 4					
In Navigation	Configuration C Explorer Configuration C Explorer C Configuration System view Technical (tree) Technical (list)	* BACnet	Network #14 🖤 Dev Binary-Value #0 BinaryValue 0 ES 🛄 Calendar	ice #6 "RDC712 (SN: 4273)) "TestDescription" 🛞 Supply General	36470	14)" / "Testdescription".
	B Anown Search Search					
	7 Geräte, 189 Objekte (aus einfacher Suche)	_	Eigenschaft	Datentyp		Wert
	Explorer	Identific	cation	_		
	BAChet Natwork #14 /via Local PAChet ID Natwork #1 Address 107 169		Object-Identifier	Object-Identifier		Image: Image
	Device #6 "RDC712 (SN: 4273864704)" / "Testdescript	F	Object Type	Object Tune		
	Binary-Value #0 "BinaryValue 0" / "TestDescription"		g Object-Type	Object-Type		
	Binary-Value #1 "BinaryValue 1" / "TestDescription"		Object-Name	Character-String		BinaryValue 0
	Binary-Value #2 "BinaryValue 2" / "TestDescription"		Description	Character-String		TestDescription
	Binary-Value #3 "BinaryValue 3" / "TestDescription"					
	Binary-Value #4 "BinaryValue 4" / "TestDescription"		Profile-Name	Character-String		290-0x0401
	Binary-Value #5 "BinaryValue 5" / "TestDescription"	Status				
	Binary-Value #6 "BinaryValue 6" / "TestDescription"	• E	Status-Flags	Status-Flags	V00	
	M Binary-Value #7 "BinaryValue 7" / "TestDescription"		Event-State	Event-State		Normal (ID:0)
	Binary-Value #8 "BinaryValue 8" / "TestDescription"			Event State		
	Binary-Value #9 Binary-Value 9 / TestDescription		Reliability	Reliability		No-Fault-Detected (ID:0)
	Binary-Value #10 BinaryValue 10 / TestDescriptic	Binary-V	/alue Specific Pro	perties		
	Multi-State-Value #0 "BinaryValue 0 - LED Confi		Present-Value	Binary-P-V	vov	Inactive (0) V
	Wulti-State-Value #1 "BinaryValue 1 - LED Confi					
	Multi-State-Value #2 "BinaryValue 2 - LED Confi		Inactive-Text	Character-String		Inactive
	🐻 Multi-State-Value #3 "BinaryValue 3 - LED Confi		Active-Text	Character-String		Active
	🐻 Multi-State-Value #4 "BinaryValue 4 - LED Confi		_			

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.

Der Status, der LED wird im "Present-Value" als Inactive (0) oder Active (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

R										
0	• • 4	-				_	_			
In Navigation	Configuration C Explorer * C System view Technical (tree) Technical (ist)	BACnet Network #14 Device #6 "RDC712 (SN: 4273864704)" / "Testdescription" Structured-View #768 BinaryValue 0 sensor configuration Properties								
	🕼 Known 🛷 New 💿 🕤 📮 🗜 🤤									
	7 Geräte, 189 Objekte (aus einfacher Suche)	_	Eigenschaft	Datentyp	Wert					
	Binary-Value #9 "BinaryValue 9" / "TestDescription"	Identif	ication							
	Binary-Value #10 "BinaryValue 10" / "TestDescriptic"	[Object-Identifier	Object-Identifier	Structured-View (29)	~	# 768	00		
	Binary-Value #11 "BinaryValue 11" / "TestDescriptic			object Mendner			# 100	6.6		
	Multi-State-Value #0 Binaryvalue 0 - LED Conri		Object-Type	Object-Type	Structured-View (ID:29)	~				
	Multi-State-Value #2 "BinaryValue 2 - LED Confi	l i	Object-Name	Character-String	BinaryValue 0 sensor confi	iguration				
	Multi-State-Value #3 "BinaryValue 3 - LED Confi	6				-				
	Multi-State-Value #4 "BinaryValue 4 - LED Confi		Description	Character-String						
	8 Multi-State-Value #5 "BinaryValue 5 - LED Confi	Structu	red-View Specific Prop	erties		_	_			
	😻 Multi-State-Value #6 "BinaryValue 6 - LED Confi	= [Subordinate-List	Device-Object-Reference[1]	[[Device-Identifier=Device	#9, Object-	Identifier=Bina	ry-Value #0]]		
	😻 Multi-State-Value #7 "BinaryValue 7 - LED Confi		Device-Object	Reference [Device Identifier-D	avisa #0. Object Identifier-Pi	nany Value	#0]	-		
	😻 Multi-State-Value #8 "BinaryValue 8 - LED Confi	ľ		Therefore [Device-Identifier=D	evice #9, Object-Identifier=bi	nary-value	#0]			
	😻 Multi-State-Value #9 "BinaryValue 9 - LED Confi		Device-Ider	ntifier Object-Identifier 🖅 De	vice (8) 🗸 🗸	# 9	68			
	8 Multi-State-Value #10 "BinaryValue 10 - LED Cor.		Object-Ider	ntifier Object-Identifier 🔟 Bin	nary-Value (5) 🗸 🗸	# 0	00			
	😻 Multi-State-Value #11 "BinaryValue 11 - LED Cor		, — 							
	Structured-View #768 "BinaryValue 0 sensor cor		.							
	Structured-View #769 "BinaryValue 1 sensor cor	· · ·	Subordinate-Annotation	S Character-String[1]	["remote object"]					
	Structured-View #770 "BinaryValue 2 sensor cor				[]ace]					

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 (physikalischer oder virtueller) der Status LED zugeordnet. Diese zeigt dann den aktuellen Zustand im "Binary-Value" (Present-Value als Inactive oder Active) und in der eingestellten Farbe an.

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

Einzutragen sind

Device-Identifier:	Device(8)	DeviceID (hier die 9)
Objekt-Identifier:	Objekt-Type (hier Binary-Value)	Objekt-Number (hier die 0)

Die Statusänderung (COV = Change 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 Objekt

Die Funktion des Structured-View kann deaktiviert werden.

Dies ist nötig, wenn eine oder alle der 12 Status LEDs über den vorhandenen "Multistate-Value" (#0 bis #11) direkt beschrieben werden sollen.

Strue	ture	d-Vie	w-sp	ezifische Eigens	chaften									
	Subordinate-List Device-Object-Reference[1] ^{POII} [[Device-Identifier=Device #4194303, Object-Identifier=Binar										nary-Value #419	94303]]		
	[1] Device-Object-Reference [Device-Identifier=Device #4194303, Object-Identifier=Binary-Value #4194303]													
			≡	Device-Identifier	Object-Identifier	🕎 Dev	ice (8)	~	# 4	194303	00			
			≡	Object-Identifier	Object-Identifier	I Bina	ary-Value (5)	~	# 4	194303	00			
		÷												
	Ξ	Subo	rdinat	te-Annotations Cha	racter-String[1]	POLI 74	["remote object"]							
		≡	[1]	Character-String re	emote object									

Abb. Structured-View Deaktivierung

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

Dort in der Zuordnung

Device-Identifier:	Device(8)	DeviceID	4194303
Objekt-Identifier:	Objekt-Type (hier Binary-Value)	Objekt-Number	4194303

Siehe obige Abb. Structured-View Deaktivierung.



Multistate-Value Objekt

() () √ 4	
5 Configuration (Explorer * A	* BACnet Network #14 ኛ Device #5 "RDC712. (SN: 4273854704)" / "Testdescription".
	Multi-State-Value #0
	BinaryValue 0 - LED Configuration "LED Configuration for active state"
System view Technical (tree) Technical (list)	Properties 📠 Calendar 🚱 Supply General
🚯 Known 🛷 New 🔞 🕞 📮 🖗	
7 Geräte, 189 Objekte (aus einfacher Suche)	Eigenschaft Datentyp Wert
Explorer	Identification
Local BACnet IP Network #1	Object-Identifier Object-Identifier Multi-State-Value (19) # 0
BACnet Network #14 (via Local BACnet IP Network #1, Address 192.168	
Rippe//oluo #0 "R/commutative 0" / "TestDescription"	Object-Type Object-Type Multi-State-Value (ID:19)
Binary-Value #1 "BinaryValue 1" / "TestDescription"	Object-Name Character-String BinaryValue 0 - LED Configuration
Binary-Value #2 "BinaryValue 2" / "TestDescription"	Department of the IPD Configuration from the text
Binary-Value #3 "BinaryValue 3" / "TestDescription"	E Description Character String EED Coninguration for active state
Binary-Value #4 "BinaryValue 4" / "TestDescription"	Status
Binary-Value #5 "BinaryValue 5" / "TestDescription"	🗉 🧮 Status-Flags Status-Flags 😤 😭 🚺
Binary-Value #6 "BinaryValue 6" / "TestDescription"	Multi-State-Value Specific Properties
Binary-Value #7 "BinaryValue 7" / "TestDescription"	Present-Value Unsigned-Integer CV red (2)
Binary-Value #8 "BinaryValue 8" / "TestDescription"	
Binary-Value #9 "BinaryValue 9" / "TestDescription"	Number-Of-States Unsigned-Integer 10
Binary-Value #10 "BinaryValue 10" / "TestDescriptic	🗉 🗐 State-Text Character-String[10] ["off", "red", "grn", "yel", "blnk red", "blnk grn", "blnk yel", "blnk yel", "blnk red grn", "blnk red yel", "blnk grn yel"]
Binary-Value #11 "BinaryValue 11" / "TestDescriptic	
Multi-State-Value #0 "BinaryValue 0 - LED Confi	
Multi-State-Value #1 BinaryValue 1 - LED Confi	E [2] Character-String red
Multi-State-Value #2 BinaryValue 2 - LED Conri	[3] Character-String grn
Multi-State-Value #4 "BinaryValue 4 - LED Confi	E [4] Character-String vel
Multi-State-Value #5 "BinaryValue 5 - LED Confi	
Multi-State-Value #6 "BinaryValue 6 - LED Confi	E [5] Unaracter-string Dink red
Multi-State-Value #7 "BinaryValue 7 - LED Confi	E [6] Character-String blnk gm
Multi-State-Value #8 "BinaryValue 8 - LED Confi	[7] Character-String blnk yel
	[8] Character-String [blnk red gm
A Darris Carlingting	[10] Character String Male and and
Open in Configuration	Dirik red yer
EDE Export	[10] Character-String [blnk grn yel
🔯 Refresh (Multi-State-Value #0 only)	

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	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	IK_RED_GREEN Rot Grün Blinken		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 ACHTUNG : State-Text -1 eintragen!

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



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 geschriebendann wird die LED ausgeschaltet (off)der Wert 2 geschriebendann leuchtet die LED in der Farbe rot (red)der Wert 8 geschriebendann blinkt die LED in den Farben rot (red) – grün (green)



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



Modulbeschreibung RDC741

Schalten von bis zu 32 Digitalausgängen über jeden der 4 Taster und Visualisierung von bis zu 12 Digitaleingängen als LED Anzeige mit Auswahl der anzuzeigenden Farbe grün/gelb/rot, mit und ohne blinken.

Die zu schaltenden Datenpunkte und anzuzeigenden Statusmeldungen werden von den jeweiligen physikalischen oder virtuellen Datenpunkten mittels BACnet Objekte "Structured-View" Objekt abonniert. Die Statusänderung (COV = Change of Value) und Zustände am physikalischen Ein- und Ausgang werden durch den Service "Subscribe_COV" übermittelt. Es kann dieser Zustand als "Confirmed_COV" oder "Unconfirmed_COV" übermittelt werden.

Konfiguration des RDC741

Device Objekt

ang series and series	Device #7 Connucce RDC741 (SN: 4273864704) " Vendor: Romutec Steuer-u. Regelsysteme (C Testdescription" SmbH (ID:290) Model: RDC741	
Properti	es 🔛 Present Values 🚺 Confi	guration RDC741	5.1.14:47806
	Eigenschaft	Datentyp	Wert
Identifi	cation		
	Object-Identifier	Object-Identifier	🛫 Device (8) 💙 # 7 🔗
Ξ	Object-Type	Object-Type	Device (ID:8)
	Object-Name	Character-String	RDC741 (SN: 4273864704)
	Description	Character-String	Testdescription
Ξ	Location	Character-String	Testlocation
Vendor,	Model and Version		
Ξ	Vendor-Name	Character-String	romutec Steuer u Regelsysteme
	Vendor-Identifier	Unsigned-16	290
Ξ	Model-Name	Character-String	RDC741
	Firmware-Revision	Character-String	0.11.006
Ξ	Database-Revision	Unsigned-Integer	0
	Application-Software-Version	Character-String	0.01
Status			
Ξ	System-Status	Device-Status	Operational (ID:0)
Protoco			
	Protocol-Version	Unsigned-Integer	1
	Protocol-Revision	Unsigned-Integer	14
•	Protocol-Services-Supported	Services-Supported	,,,, Subscribe-Cov,,,,, Read-Property,, Read-Property-I
•	Protocol-Object-Types-Supporter	Object-Types-Supported	,, Binary-Input (ID:3),, Binary-Value (ID:5),,, Device (ID:8),,,,
<	d Commetation		

Abb. Device Objekt

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

R												
	• • £	_	_	_	_	_						
🚮 Navigation	Configuration C Explorer # C	* B	BAChet Network #14 97 Device #7 "RDC741 (SN: 4273864704)" / "Textdescription" Binary-Input #0 BinaryInput 0 "Taster 0" Yoperties									
	Known Wew Devices Wearch Search Search 7 Ceräte. 189 Objekte (aus einfacher Suche)			Cincer shaft	Debashar		w.t					
	Government G	Ide	ntifica	ation	Datentyp		wer					
	ACnet Network #14 (via Local BACnet IP Network #1, Address 192.168 Device #6 "RDC712 (SN: 4273864704)" / "Testdescript	E		Object-Identifier	Object-Identifier		Binary-Input (3) # 0 0 Binary-Input (10:3) V					
	Device #7 "RDC741 (SN: 4273864704)" / "Testdescript Binary-Input #0 "BinaryInput 0" / "Taster 0"		Ξ	Object-Name	Character-String		BinaryInput 0					
	10 Binary-Input #1 "BinaryInput 1" / "Taster 1" 10 Binary-Input #2 "BinaryInput 2" / "Taster 2"		Ξ	Description	Character-String		Taster 0					
	1 Binary-Input #3 "BinaryInput 3" / "Taster 3" Binary-Value #0 "BinaryValue 0" / "TestDescription"	Stat E	tus	Status-Flags	Status-Flags	COV						
	[1] Binary-Value #1 "BinaryValue 1" / "TestDescription" [1] Binary-Value #2 "BinaryValue 2" / "TestDescription"		Ξ	Event-State	Event-State		Normal (ID:0)					
	Binary-Value #3 "BinaryValue 3" / "TestDescription" Binary-Value #4 "BinaryValue 4" / "TestDescription"	Bina	E II-vre	Reliability	Reliability		No-Fault-Detected (ID:0)					
	Binary-Value #5 "BinaryValue 5" / "TestDescription" Binary-Value #6 "BinaryValue 6" / "TestDescription"		Ξ	Present-Value	Binary-P-V	00V	Inactive (0) V					
	Binary-Value #7 "BinaryValue 7" / "TestDescription" Binary-Value #8 "BinaryValue 8" / "TestDescription"		Ξ	Polarity	Polarity		Normal (ID:0) V					
	Binary-Value #9 "BinaryValue 9" / "TestDescription" Binary-Value #10 "BinaryValue 10" / "TestDescription"			Inactive-Text	Character-String		Inactive Active					
	Binary value F10 BinaryValue 10 / TestDescriptic Multi-State-Value #0 "BinaryValue 0 - LED Confi				9							

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 Active (1) angezeigt. Über die "Polarity" kann der Status invertiert werden.

Funktion:

Mit dem "Device Type" kann die Funktion der Taster ausgewählt werden.

z.B.: MODE : 0 =

beim 1. Drücken "Present-Value" wechselt von 0 nach 1 Sendet eine "1" an den Datenpunkt

beim 2. Drücken "Present-Value" wechselt von 0 nach 1 Sendet eine "0" an den Datenpunkt



Abb. Mode Funktion Taster



Beschreibung Taster-Funktion (Device Type):

Mit dem Property "Device Type" kann die Funktion der Taster ausgewählt werden. Folgende Funktionen stehen zur Auswahl:

Mode: 0 = rise aktive inaktive; Anstieg aktive inaktive Mode: 1 = rise aktive null; Anstieg aktive null Mode: 2 = rise null inaktive; Anstieg null inaktive

Mode: 3 = edge aktive inaktive; Toggle aktive inaktive Mode: 4 = edge aktive null; Flanke aktive null Mode: 5 = edge null inaktive; Flanke null inaktive Mode: 6 = edge aktive nil; Flanke aktive nil Mode: 7 = edge nil inaktive; Flanke inaktive nil Mode: 8 = edge null nil; Flanke null null EIN/AUS-Tastend ohne NULL EIN-Tastend/AUS-Tastend mit NULL AUS-Tastend mit NULL

Tastend EIN/AUS ohne NULL Tastend EIN mit NULL Tastend AUS mit NULL Tastend EIN ohne NULL Tastend AUS ohne NULL Tastend immer NULL

- Die rise Sequenzen lösen abwechselnd beim Drücken des Knopfes den entsprechenden Wert aus.

Mittels Polarität kann das auf die fallende Flanke geändert werden.

- Die edge Sequenzen lösen bei einer rising edge (steigende Flanke "Taster drücken") den ersten Wert und bei einer falling edge (fallende Flanke "Taster loslassen") den zweiten Wert aus.

Mittels Polarität kann die Funktion invertiert werden.

Beim Drücken oder Loslassen des Tasters wird der Status in Abhängigkeit der eingestellten Priorität und Tasterfunktion an den Datenpunkt gesendet.

Mit welcher Priorität der Datenpunkt beschrieben wird, kann im Objekt "Structured-View" und dort im Device-Type eingestellt werden.

Funktionsablauf (Beispiel):

Beim "EIN"-Tasten wird in das jeweilige "Priority-Array" eine "1" geschrieben. Beim "AUS"-Tasten wird die Priorität im "Priority-Array" auf "0" gesetzt, oder beim "AUS"-Schalten wird das Priority-Array" auf "NULL" gesetzt! Gewünschte Einstellungen müssen hierfür im Property "Device-Type" erfolgen.



Binary-Value Objekt

R										
		-			-					
📶 Navigation	Konfiguration Image: Constraint of the second sec	 ★ BACAst.Netzwerk #14								
	Becante Veue Suche Suche Soche		Ciacula 0	Detroiter		w-+				
	Binary-Value #0 [*] BinaryValue 0 [*] / "lestDescription" Binary-Value #1 [*] BinaryValue 1 [*] / "TestDescription"	Iden	tifikation	Datentyp		wert				
	Binary-Value #1 BinaryValue 1 / restDescription	1001	Object-Identifie	r Obiect-Identifier	POLL	■ Binary-Value (5)				
	Binary-Value #3 "BinaryValue 3" / "TestDescription" Binary-Value #4 "BinaryValue 4" / "TestDescription"		Object-Type	Object-Type	POLL S25	Binary-Value (ID:5)				
	Binary-Value #5 "BinaryValue 5" / "TestDescription"		Object-Name	Character-String	Poll 5	mirror for BI00				
	Binary-Value #6 "BinaryValue 6" / "TestDescription" Discription		Description	Character-String	POLL	mirror for BI00				
	Binary-Value #7 BinaryValue 7 / TestDescription Binary-Value #8 "BinaryValue 8" / "TestDescription"		Drafila Nama	Character String	POLL					
	Binary-Value #9 "BinaryValue 9" / "TestDescription"	C 1-1-	Prome-warne	Character-String	90					
	[1] Binary-Value #10 "BinaryValue 10" / "TestDescription" [1] Binary-Value #11 "BinaryValue 11" / "TestDescription"	E	Status-Flags	Status-Flags	POLL ይሻ					
	Binary-Value #256 "mirror for BI00" / "mirror for BI00" Binary-Value #257 " in a second of the BI01"		Event-State	Event-State	POLL 545	Normal (ID:0)				
	Binary-Value #257 mirror for BIO1 / mirror for BIO2		Reliability	Reliability	Poll 52	No-Fault-Detected (ID:0)				
	Binary-Value #259 "mirror for BI03" / "mirror for BI03"	Bina	ry-Value-spezifisch	e Eigenschaften						
	Multi-State-Value #0 "BinaryValue 0 - LED Configuration In Multi-State-Value #10"		Present-Value	Binary-P-V	POLL 52	Inactive (0) V				
	Multi-State-Value #1 "BinaryValue 1 - LED Configuration Multi-State-Value #2 "BinaryValue 2 - LED Configuration		Inactive-Text	Character-String	POLL	Inactive				
	Wulti-State-Value #3 "BinaryValue 3 - LED Configuration			Character-String	POLL	Active				
	80 Multi-State-Value #4 "BinaryValue 4 - LED Configuration		Active-rext	unaraccer punity	80	Incore				
	😻 Multi-State-Value #5 "BinaryValue 5 - LED Configuration									

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

powered by WSW Solutions											
5 Configuration (Explorer * (A)	* I	ACnet N	etwork #	#14 🛒 Device #7 "R	DC741 (SN: 427386470)4)" / "Test	description"				
		S	ructu	red-View #0							
		Bi	naryl	Input 0 trigge	r configuratio	n					
System view Technical (tree) Technical (list)	Pro	operties									
🕼 Known 🛷 New 🚳 🕞 📮 🖗											
7 Geräte, 189 Objekte (aus einfacher Suche)		_	Eigen	schaft	Datentyp	_	Wert	_			
Binary-Value #7 "BinaryValue 7" / "TestDescription"	Ide	ntifica	tion								
Binary-Value #8 "BinaryValue 8" / "TestDescription"			Object	t-Identifier	Object-Identifier	_	Structured-V	iew (29)	~	# 0	00
Binary-Value #9 "BinaryValue 9" / "TestDescription"			objec	c-toenuner	object Identifier		• Sudcarda •	GH (25)	-	* 0	60
Binary-Value #10 "BinaryValue 10" / "TestDescriptic			Object	t-Type	Object-Type		Structured-View	(ID:29)	~		
Binary-Value #11 "BinaryValue 11" / "TestDescriptic		Ξ	Object	t-Name	Character-String		BinaryInput 0 tric	aer confi	duration		
Multi-State-Value #0 "BinaryValue 0 - LED Confi Multi-State-Value #0 "BinaryValue 0 - LED Confi Multi-State-Value #0 "BinaryValue 0 - LED Confi											
Multi-State-Value #1 BinaryValue 1 - LED Confi			Descri	iption	Character-String						
Multi-State-Value #2 BinaryValue 2 - LED Confi	Str	ucture	d-Viev	w Specific Prope	rties	_		_	_		
Multi-State-Value #4 "BinaryValue 4 - LED Confi	E		Subor	dinate-List	Device-Object-Refe	rence[32]	[[Device-Identifier	=Device	#10, Object	t-Identifier=Bi	nary-Outpu
8 Multi-State-Value #5 "BinaryValue 5 - LED Confi				[1] Device-Object	Reference [Device	Idontifior	-Davica #10. Object Id	ontifior_F	linner Outer		
Multi-State-Value #6 "BinaryValue 6 - LED Confi			<u> </u>		IDevice	Identifier	=Device #10, Object-to	enuner=c	onary-Outpu	ut #0j	
8 Multi-State-Value #7 "BinaryValue 7 - LED Confi				Device-Ident	tifier Object-Identifi	* 🛫 🗖	Device (8)	~	# 10	@0	
😻 Multi-State-Value #8 "BinaryValue 8 - LED Confi				Object-Ident	ifier Object-Identifi	er 🗊 🖪	Binary-Output (4)	~	# 0	00	
😻 Multi-State-Value #9 "BinaryValue 9 - LED Confi				[2] Davies Object	Poforonco Device		Device #10 Object Id				
😻 Multi-State-Value #10 "BinaryValue 10 - LED Cor			<u> </u>	[2] Device-object	Hereferice [Device-	Identifier	=Device #10, Object-to	enuner=c	sinary-Outpu	ut #5]	
🐯 Multi-State-Value #11 "BinaryValue 11 - LED Cor				Device-Ident	tifier Object-Identifi	* 🛫 🗖	Device (8)	\sim	# 10	00	
Structured-View #0 "BinaryInput 0 trigger cont				Object-Ident	ifier Object-Identifi	er 🗊 🖪	Binary-Output (4)	~	# 5	00	
Structured-View #1 "BinaryInput 1 trigger cont				Fall Device Object	D-6		D : #4404000 ol:				40001
Structured-View #2 "BinaryInput 2 trigger coni			쁘	[3] Device-Object	-Kererence [Device-	Identifier	=Device #4194303, Obj	ect-Ident	ifier=Binary	-Output #419	4303]
Structured-View #3 "BinaryInput 3 trigger coni Constant View #360 "Free Provide Strigger Coni				Device-Ident	tifier Object-Identifi	r 🛫 🖸	Device (8)	\sim	# 4194303	; 00	
Structured-View #768 BinaryValue 0 sensor cor Gruntured View #760 "Discounting has a sensor cor				Object-Ident	ifier Object-Identifi	er 🗊 🖪	Binary-Output (4)	~	# 4194303	00	
Structured-View #700 Binaryvalue 1 sensor cor Structured-View #770 "PinaryValue 2 sensor cor											
Structured-View #771 "BinaryValue 3 sensor cor				[4] Device-Object	-Reference [Device-	Identifier	=Device #4194303, Obj	ect-Ident	ifier=Binary	-Output #419	4303]
Statutes for #//1 binaryvarde 5 sensor con				Device-Ident	t ifier Object-Identifi	* 🏆 🛛	Device (8)	~	# 4194303	00	
				Object-Ident	ifier Object-Identifi	er 🗊 🖪	Binary-Output (4)	~	# 4194303	00	
Open in Configuration				object four		-	,				
		I III		[5] Device-Object	-Reference [Device-	Identifier	=Device #4194303_Ohi	ect-Ident	ifier=Binary	/-Outnut #419	43031

Abb. Structured-View Objekt

Für jeden der 4 Taster gibt es ein "Structured-View" (trigger configuration) Objekt (#0 bis #3). Es können für jeden Taster 32 Ziele (Einträge) definiert werden. In diesem Objekt werden das Device und die Datenpunkte (physikalische oder virtuelle) des zu übersteuernden Gerätes zugeordnet. Es können Master oder Slave Devices gesteuert werden. Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein.

Einzutragen sind bei Master-Device:

Device-Identifier:	Device (8)
Objekt-Identifier:	Objekt-Type (hier Binary-Output)
	Binary-Output oder Binary-Value

DeviceID (hier die 10) Objekt-Number (hier die 0)

bei Slave-Device:

Device-Identifier: Characterstring-Value (40) Objekt-Identifier: Objekt-Type (hier Binary-Output) Binary-Output oder Binary-Value MAC-Adresse (hier die 10) Objekt-Number (hier die 1)

-	Ξ	Subor	dinat	e-List Dev	ice-Object-Refere	nce[3	2] [[Device-Identifier=D	evice)	#9,	, Object-Identifi	er=Bina	ary-(
	[1] Device-Object-Reference [Device-Identifier=					er=Device #9, Object-Identif	r=Device #9, Object-Identifier=Binary-Output #0]					
			Ξ	Device-Identifier	Object-Identifier	Ţ	Device (8)	~	#	9	Ø	
			Ξ	Object-Identifier	Object-Identifier	1	Binary-Output (4)	~	#	0	œ	
		∎	[2]	Device-Object-Refer	ence [Device-Id	entifi	er=Characterstring-Value #1	0, Ob	ject	-Identifier=Bina	ry-Outp	put #
			Ξ	Device-Identifier	Object-Identifier	0	Characterstring-Value (40)	~	#	10	Ø	
			Ξ	Object-Identifier	Object-Identifier	1	Binary-Output (4)	~	#	1	Ø	
	_		F =1	n i oli in f								

Abb. Structured-View Objekt Master/Slave





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 das Device nicht übergeben werden kann, wird dieses an der Statusanzeige unter dem 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

Device-Identifier:	Device(8)	DeviceID	4194303
Objekt-Identifier:	Objekt-Type (hier Binary-Value)	Objekt-Number	4194303

Siehe Abb. Structured-View Deaktivierung unter Structured View Objekt, Seite 16.

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).

R													
🔝 Navigation	Konfiguration C Explorer * C Systemsicht Technisch (Baum) Technisch (Lista) Sessere Systemsicht Gestrete Sessere Sesseree Sesseree	* B C Eig	BAChet-Netzwerk #14 77 Device #7 "RDC741 (SN: 4273864704)" / "Tendescription" Structured-View #0 BinaryInput 0 trigger configuration Eigenschaften										
	7 Geräte, 193 Objekte (aus einfacher Suche)		_	Eigenschaft	Datentyp		Wert	_			_		
	😻 Multi-State-Value #3 "BinaryValue 3 - LED Confi	Ide	ntifik	ation	1	_							
	😻 Multi-State-Value #4 "BinaryValue 4 - LED Confi			Obiest Identifier	Object Identifier		Structured View	(20)	¥ #0	00			
	😻 Multi-State-Value #5 "BinaryValue 5 - LED Confi			Object-Identifier	r Object-Identifier		Structured-view	(29)	• #U	68			
	Multi-State-Value #6 "BinaryValue 6 - LED Confi Design and the second		Ξ	Object-Type	Object-Type		Structured-View (ID:	29)	~				
	Multi-State-Value #7 "BinaryValue 7 - LED Confi Multi-State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value #8 "BinaryValue 7 - LED Confi Destruction of the State Value 7 - LED Confi Destruction of the State 7 - LED Confi Destruction o		Ξ	Object-Name	Character-String		BinaryInput 0 trigger	configuratio	on]		
	Multi-State-Value #0 BinaryValue 8 - LED Confi			-	Chanadas China			-			1		
	Multi-State-Value #10 "BinaryValue 10 - LED Cor			Description	Character-String								
	8 Multi-State-Value #11 "BinaryValue 11 - LED Cor		Ξ	Device-Type	Character-String		prio: 8						
	Structured-View #0 "BinaryInput 0 trigger cont	Stru	icture	d-View-spezifis	sche Eigenschaften								
	Structured-View #1 "BinaryInput 1 trigger cont		Ξ	Subordinate-List	t Device-Object-Referer	nce[32]	[[Device-Identifier=De	evice #9, Ol	piect-Identifier=E	linary-Output #	#01, [Dev		
	Structured-View #2 "BinaryInput 2 trigger cont								-				
	Structured-View #3 "BinaryInput 3 trigger cont				nce-Object-Reference [Device-Ide	entifier=De	vice #9, Object-Identifi	er=Binary-C	utput #0]	-			
	Structured-View #768 "BinaryValue 0 sensor cor			🔳 Devi	ice-Identifier Object-Identifier	Devic	e (8)	✓ # 9	00				
	Structured-View #769 "BinaryValue 1 sensor cor Chrystered View #770 "Discussion of the sensor cor			🗐 Obje	ect-Identifier Object-Identifier	Dinan Binan	-Output (4)	✓ # 0	00				
	Structured-View #771 "BinaryValue 2 sensor cor Structured-View #771 "BinaryValue 2 sensor cor				ico Obiost Roformora ID- : •		and the state of the	Object 71		4-14			
	Structured-View #772 "BinaryValue 4 sensor cor				Intervolution and a CDevice-106	entiner=Cha	aracterstring-Value #10	, Object-1de	nuner=Binary-O	utput #1]			
	Structured-View #773 "BinaryValue 5 sensor cor			🔳 Devi	ice-Identifier Object-Identifier	Chara	cterstring-Value (40)	✓ # 10	68				
	Structured-View #774 "BinaryValue 6 sensor cor			🔳 Obje	ect-Identifier Object-Identifier	I Binar	-Output (4)	✓ # 1	00				
A 1-1	- Ctru ature d Mieur Obielst Drierität												

Abb. Structured-View Objekt Priorität

Ansteuerung der LEDs 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 zugehörigen "Present-Value" (siehe hierzu auch Abb. Multistate-Value zur LED-Konfiguration).

Die LEDs leuchten dann nach den Eigenschaften, wie sie im "State Text" definiert sind. Wird zum Beispiel

der Wert 1 geschrieben	dann wird 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 Seiten 15 und 16, <u>Structured View Objekt</u> beim RDC712. Die vorzunehmenden Einstellungen sind identisch wie diesem Modul, Abb. Structured-View Deaktivierung.



Binary-Value Objekt

R	OBA ¢CONNECT											
	powered by WSW Solutions											
	• \$	_	_	_	-							
5	Configuration 🔕 Explorer 🕷 🚱	* BACnet N	letwork #14 🛒 Dev	rice #7 "RDC741 (SN: 42	738647	04)" / "Testdescription"						
avigati	*	Bi Bi	Binary-Value #U BinaryValue 0 "TestDescription"									
¥	System view Technical (tree) Technical (list)	Properties	S Calendar	🍪 Supply General								
	🚯 Known 🛷 New 🚳 🕞 🚽 🖗											
	7 Geräte, 189 Objekte (aus einfacher Suche)		Eigenschaft	Datentyp		Wert						
	🗆 🥘 Explorer	Identific	ation									
	🗄 🍁 Local BACnet IP Network #1											
	🖃 🌳 BACnet Network #14 (via Local BACnet IP Network #1, Address 192.168		Object-Identifier	r Object-Identitier		III Binary-Value (5)						
	H Interpretation of the second secon		Object-Type	Object-Type		Binary-Value (ID:5)						
	🖃 🛫 Device #7 "RDC741 (SN: 4273864704)" / "Testdescript			d t di								
	🗊 Binary-Input #0 "BinaryInput 0" / "Taster O"		Object-Name	Character-String		BinaryValue 0						
	Binary-Input #1 "BinaryInput 1" / "Taster 1"		Description	Character-String		TestDescription						
	Binary-Input #2 "BinaryInput 2" / "Taster 2"		Deefile Name	Character Chrise		200.0-0401						
	Binary-Input #3 "BinaryInput 3" / "Taster 3"		Profile-Iname	Character-String		290-0x0401						
	Binary-Value #0 "BinaryValue 0" / "TestDescription"	Status										
	Binary-Value #1 "BinaryValue 1" / "TestDescription" Binary-Value #2 "BinaryValue 2" / "TestDescription"	• =	Status-Flags	Status-Flags	00							
	Binary-Value #3 "BinaryValue 3" / "TestDescription"		Event-State	Event-State		Normal (ID:0)						
	Binary-Value #4 "BinaryValue 4" / "TestDescription"		Reliability	Reliability		No-Sensor (ID:1)						
	Binary-Value #5 "BinaryValue 5" / "TestDescription"	Dinamy V	Juo Enocific Dr	apartias								
	Binary-Value #6 "BinaryValue 6" / "TestDescription"	billary-va	alue Specific Pr	operues	0.07							
	Binary-Value #7 "BinaryValue 7" / "TestDescription"		Present-Value	Binary-P-V	00	Inactive (0) V						
	Binary-Value #8 "BinaryValue 8" / "TestDescription"		Inactive-Text	Character-String		Inactive						
	Binary-Value #9 "BinaryValue 9" / "TestDescription"			-								
	III Binary-Value #10 "BinaryValue 10" / "TestDescriptic		Active-Text	Character-String		Active						

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.

Der Status der LED wird im "Present-Value" als Inactive (0) oder Active (1) angezeigt.

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

Konfiguration der Status LED (Multistate-Value Objekt)

Siehe hierzu auch LED Konfiguration RDC712 (Multistate-Value Objekt).

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.

Structured-View Objekt

R									
0	• 4								_
In Navigation	Configuration Explorer *	* BACn	et Ne Sti Bin ties	twork #14 🚿 Device #6"F ructured-View #768 naryValue 0 sensor	NDC712(SN:4273864704)"/"Testde g configuration	scription".			
	Sevices V Search Search				1 1	-			
	/ Gerate, 189 Objekte (aus einfacher Suche)		-	Eigenschaft	Datentyp	Wert	_		
	Binary-Value #10 "Pinary-Value 5 / TestDescription	Identi	fica	tion					
	Binary-Value #11 "BinaryValue 10 / TestDescriptic	F	≣	Object-Identifier	Object-Identifier	Structured-View (29)	~	# 768	00
	Multi-State-Value #0 "BinaryValue 0 - LED Confi		_	-	als an				
	Multi-State-Value #1 "BinaryValue 1 - LED Confi	L L	=	Object-Type	Object-Type	Structured-View (ID:29)	~		
	Multi-State-Value #2 "BinarvValue 2 - LED Confi		Ξ	Object-Name	Character-String	BinaryValue 0 sensor confi	iguration		
	Multi-State-Value #3 "BinaryValue 3 - LED Confi	L I	=	Description	Character-String				
	Multi-State-Value #4 "BinaryValue 4 - LED Confi		=	Description	Character-Scring				
	8 Multi-State-Value #5 "BinaryValue 5 - LED Confi	Struct	ure	d-View Specific Prope	rties				
	😺 Multi-State-Value #6 "BinaryValue 6 - LED Confi		Ξ	Subordinate-List	Device-Object-Reference[1]	[[Device-Identifier=Device	#9, Object	-Identifier=Binar	y-Value #0]]
	😻 Multi-State-Value #7 "BinaryValue 7 - LED Confi	L F	_	E 11 Davies Object	Pafaranza (Deuise Identifica D	wine #0. Object Televices Pi	- Nelve		
	😻 Multi-State-Value #8 "BinaryValue 8 - LED Confi				Therefore [Device-Identifiel=D	evice #9, Object-Identifier=bi	hary-value	#0]	
	😻 Multi-State-Value #9 "BinaryValue 9 - LED Confi			Device-Iden	tifier Object-Identifier <table-cell> De</table-cell>	vice (8) V	# 9	66	
	😺 Multi-State-Value #10 "BinaryValue 10 - LED Cor			Object-Iden	tifier Object-Identifier 🔟 Bin	ary-Value (5)	# 0	60	
	😺 Multi-State-Value #11 "BinaryValue 11 - LED Cor								
	Structured-View #768 "BinaryValue 0 sensor cor			*					
	Structured-View #769 "BinaryValue 1 sensor cor	I III	=1	Subordinate-Annotation	s Character-String[1]	["remote object"]			
	Structured-View #770 "BinaryValue 2 sensor cor		_	Sabaranata Annotation		[ramote cojuce]			
Ahh	Structured-View								

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 Inactive oder Aktiv) und in der eingestellten Farbe an.

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

Einzutragen sind

Device-Identifier:	Device(8)	DeviceID (hier die 9)
Objekt-Identifier:	Objekt-Type (hier Binary-Value)	Objekt-Number (hier die 0)

Die Statusänderung (COV = Change of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann 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 "Structured-View" Objekt abonniert. Die Statusänderung (COV = Change 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 Objekt

	Device #8 Communication 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 192,158.1.14:47808									
Prop	erties	🔛 Present Values 🚺 Config	guration RDC742 Calendar	r						
_	_	Eigenschaft	Datentyp	Wert						
Iden	tifica	tion								
	Ξ	Object-Identifier	Object-Identifier	Device (8) # 8						
	Ξ	Object-Type	Object-Type	Device (ID:8)						
	≡	Object-Name	Character-String	RDC742 (5N: 4273864704)						
	Ξ	Description	Character-String	Testdescription						
	≡	Location	Character-String	Testlocation						
Vend	or, N	1odel and Version								
	Ξ	Vendor-Name	Character-String	romutec Steuer u Regelsysteme						
	Ξ	Vendor-Identifier	Unsigned-16	290						
	Ξ	Model-Name	Character-String	RDC742						
	Ξ	Firmware-Revision	Character-String	0.11.008						
	Ξ	Database-Revision	Unsigned-Integer	0						
	≡	Application-Software-Version	Character-String	0.01						
Statu	S									
	Ξ	System-Status	Device-Status	Operational (ID:0)						
Proto	col									
	Ξ	Protocol-Version	Unsigned-Integer	1						
	≡	Protocol-Revision	Unsigned-Integer	14						
Đ	≡	Protocol-Services-Supported	Services-Supported	,,,, Subscribe-Cov,,,,, Read-Property,, Read-Property-Multiple, Write-Property,,,						
±	Ξ	Protocol-Object-Types-Supported	Object-Types-Supported	,,, Binary-Input (ID:3),,,, Device (ID:8),,,, Multi-State-Input (ID:13),,,, Multi-State-Input (ID:13),,,,, Multi-State-Input (ID:13),,,,, Multi-State-Input (ID:13),,,,, Multi-State-Input (ID:13),,,,, Multi-State-Input (ID:13),,,,, Multi-State-Input (ID:13),,,,, Multi-State-Input (ID:13),,,,,, Multi-State-Input (ID:13),,,,,, Multi-State-Input (ID:13),,,,,,, Multi-State-Input (ID:13),,,,,,,,,, -						
A hh	De	Vice Objekt								

Abb. Device Objekt

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

R						
(• • £		_	_		
avigation	Configuration @ Explorer * @	* BAChe T	st Network #14 <i>%</i> Der Binary-Input #1 BinaryInput 0 ties	vice #8 "RDC742 (SN: 4 0 "Taster 0"	2738647	'04)" / "Testdescription"
	B Known 💞 Search 💿 🕞 🚽 🚱 7 Geräte, 189 Objekte (aus einfacher Suche)		Eigenschaft	Datentyp		Wert
	70 Binary-Input #0 "BinaryInput 0" / "Taster 0" 10 Binary-Input #1 "BinaryInput 1" / "Taster 1" 11 Binary-Input #1 "BinaryInput 1" / "Taster 1"	Identif	ication	r Object-Identifier		1 Binary-Input (3) 🗸 # 0 🔊
	 Binary-Input #2 SinaryInput 2" / Taster 3" Bitstring-Value #0 "BitstringValue 0" / "Rauchmek 		Object-Type	Object-Type		Binary-Input (ID:3)
	 Bitstring-Value #1 "BitStringValue 1" / "Rauchmek Bitstring-Value #2 "BitStringValue 2" / "Rauchmek 		Object-Name Description	Character-String Character-String		BinaryInput 0 Taster 0
	Bitstring-Value #3 "BitstringValue 3' ("Rauchmek Multi-State-Input #0 "MultiStateInput 0' ("Klappe Multi-State-Input #1 "Multi StateInput 1' ("Klappe Multi-State-Input #1 "Multi StateInput 1' ("Klappe	Status	Device-Type	Character-String		toggle: 1
	Multi-State-Input #2 "MultiStateInput 2" / "Klappe Multi-State-Input #3 "MultiStateInput 3" / "Klappe	• [Status-Flags	Status-Flags	00V @@	
	Multi-State-Value #0 "Test Status 0" / "Description" Multi-State-Value #1 "Test Status 1" / "Description"		Event-State Reliability	Event-State Reliability		Normal (ID:0)
	Wulti-State-Value #2 "Test Status 2" / "Description" Multi-State-Value #3 "Test Status 3" / "Description" Statutured-View #0 "Einspruch of trigger cont Statutured-View #0 "Einspruch of trigger cont	Binary-	Input Specific Pr	operties Binary-P-V	COV	
	 Structured-View #1 "BinaryInput 1 trigger coni Structured-View #2 "BinaryInput 2 trigger coni 		Polarity	Polarity		Normal (ID:0)
	 Structured-View #3 "BinaryInput 3 trigger cons Structured-View #256 "BitStringValue 0 sensor Structured-View #257 "BitStringValue 1 sensor 		Active-Text	Character-String Character-String		Inactive Active

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. Diese kann über den zugehörigen Multi-State-Value (#x) verändert werden.

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

Mit dem "Device Type" kann die Funktion der Taster ausgewählt werden.

z.B.: MODE : 1 =

beim 1. Drücken "Present-Value" wechselt von 0 nach 1 Sendet eine "1" an den Datenpunkt

beim 2. Drücken "Present-Value" wechselt von 0 nach 1 Sendet eine "0" an den Datenpunkt

Beim "AUS"-Schalten wird die Priorität im "Priority-Array" auf "NULL" gesetzt!

Binary-Input #0 BinaryInput 0 "Taster 0" Properties									
_	Eigenschaft	Datentyp		Wert					
Identi	fication	_							
[Object-Identifie	r Object-Identifier		1 Binary-Input (3) 🗸 # 0 🕅					
[Object-Type	Object-Type		Binary-Input (ID:3)					
[Object-Name	Character-String		BinaryInput 0					
[Description	Character-String		Taster 0					
[Device-Type	Character-String		toggle: 1					
Status									
•	Status-Flags	Status-Flags	00V						
[Event-State	Event-State		Normal (ID:0)					
[Reliability	Reliability		No-Fault-Detected (ID:0)					
Bina r y	-Input Specific Pr	operties							
[Present-Value	Binary-P-V	00V	Inactive (0) V					
[Polarity	Polarity		Normal (ID:0) V					
[Inactive-Text	Character-String		Inactive					
[Active-Text	Character-String		Active					
L	- riedene value	Contrary of the	~~	TUBERAE (A) +					
[Polarity	Polarity		Normal (ID:0) 🗸					
[Inactive-Text	Character-String		Inactive					
[Active-Text	Character-String		Active					

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 werden das Device und die Datenpunkte (physikalische oder virtuelle) des zu übersteuernden Gerätes zugeordnet.

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

Einzutragen sind

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

Die Statusänderung (COV = Change of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann 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.



Test Status Multistate-Value Objekt

R						
	○ • ₺		_	_		
而 Navigation	Configuration @ Explorer * @ System view Technical (tree) Technical (list) 2 *	Prop	ACnet Network #14 7 Device Multi-State-Value Test Status 0 "I perties Calendar	: #8 "RDC742 (SN: 42738 : #0 Description" Supply General	164704))" / "Testdescription"
	Image: Search Image: Search 7 Geräte, 189 Objekte (aus einfacher Suche) Image: BitStringValue 0" / "Rauchmek		Eigenschaft	Datentyp		Wert
	 Bitstring-Value #1 "BitStringValue 1" / "Rauchmek Bitstring-Value #2 "BitStringValue 2" / "Rauchmek Bitstring-Value #3 "BitStringValue 3" / "Rauchmek 	Iden	Diject-Identifier	Object-Identifier		8 Multi-State-Value (19)
	Multi-State-Input #0 "MultiStateInput 0" / "Klappe Multi-State-Input #1 "MultiStateInput 1" / "Klappe "Nulti-State-Input #1 "MultiStateInput 1" / "Klappe"		 Object-Type Object-Name 	Object-Type Character-String		Multi-State-Value (ID:19) Test Status 0
	Multi-State-Input #2 MultiStateInput 2 / Klappe Multi-State-Input #3 "MultiStateInput 3" / "Klappe Multi-State-Value #0 "Test Status 0" / "Description"	State	Description	Character-String		Description
	Multi-State-Value #1 "Test Status 1" / "Description" Multi-State-Value #2 "Test Status 2" / "Description" Multi-State-Value #3 "Test Status 3" / "Description"		Status-Flags ti-State-Value Specific	Status-Flags Properties	×	
	 Structured-View #0 "BinaryInput 0 trigger con: Structured-View #1 "BinaryInput 1 trigger con: 		Present-Value Number-Of-States	Unsigned-Integer Unsigned-Integer	×	off (1) on (2) 2
	 Structured-View #2 "BinaryInput 2 trigger coni Structured-View #3 "BinaryInput 3 trigger coni Structured-View #256 "BitStringValue 0 sensor 	٠	State-Text	Character-String[2]		["off", "on"]

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	Klappen Status	Klappen Status		
BI Flap closed	BI Flap open	MSV	MSI Flap State	MSI Input Status Flags		
			closed-movig-open- fusible_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	fusible_link	In_Alarm		
0	1	on / EIN	open			
1	0	on / EIN	fusible_link	In_Alarm		
1	1	on / EIN	fault	In_Alarm		



BSK Status Structured-View Objekt

	St	ructured-View #	#512	ion					
Prop	erties	;	o sensor conrigurad	1011					
_	_	Eigenschaft	Datentyp		Wert	_	_		
Iden	tifica	ition		_		_	_	_	
	≣	Object-Identifier	Object-Identifier		Structured-View ((29)	~	# 512	60
	≣	Object-Type	Object-Type		Structured-View (ID:	29)	~		
	∎	Object-Name	Character-String		MultiStateInput 0 sen	sor conf	iguratio	n	
	Ξ	Description	Character-String						
Struc	ture	d-View Specific P	roperties						
	≣	Subordinate-List	Device-Object-Refere	ence[2]	[[Device-Identifier=De	vice #1	0, Objec	t-Identifier=Bina	ry-Input #4
		[1] Device-0	bject-Reference [Device-Ide	entifier=Dev	ice #10, Object-Identif	ìer=Bina	ary-Inpu	t #4]	
		Device-	Identifier Object-Identifier	🕎 Devic	te (8)	> #	10	60	
		Object-	Identifier Object-Identifier	Binar	y-Input (3)	× #	4	00	
		[2] Device-0	bject-Reference [Device-Ide	entifier=Dev	rice #10, Object-Identif	ier=Bina	ary-Inpu	t #/]	
		Device-	Identifier Object-Identifier	🕎 Devic	te (8)	> #	10	60	
		Object-	Identifier Object-Identifier	🚺 Binar	y-Input (3)	× #	7	00	
		+							
	Ξ	Subordinate-Annota	ations Character-String[2]		["flap closed", "flap op	en"]			
		[1] Character	r-String flap closed						
		[2] Character	r-String Nap open						
		+							
	1								

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 (physikalische oder virtuelle) des zu überwachenden Gerätes, hier eine BSK, zugeordnet. Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein.

Einzutragen sind

Klap	pen-Meldungen Endlage	nschalter "ZU"(flap closed)	
1	Device-Identifier:	Device(8)	DeviceID (hier die 10)
1	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die 4)
Klap	pen- Meldungen Endlage	enschalter "AUF" (flap open)	
2	Device-Identifier:	Device(8)	DeviceID (hier die 10)
2	Objekt-Identifier [.]	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die 7)

Die Statusänderung (COV = Change of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann als "Confirmed_COV" oder "Unconfirmed_COV" übermittelt werden.



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 dies an der Statusanzeige unter dem Taster durch eine LED rot signalisiert.

<u>Gruppen</u>	<u>Bezeichnung</u>	<u>Status</u>	LED-Status	Funktion
LED	Rauchmelder	OK	Aus	
Oben	LED verschmutzt	verschmutzt	Gelb	RM verschmutzt
	Development	OK	Aus	
LED	Rauchmeider LED Störung	Störung elektrisch	gelb blinken	RM gestört
wille		Ausgelöst Rauch	rot blinken	RM ausgelöst
		_		
		Offen	Grün	BSK Offen
		Zu	Aus	BSK geschlossen
LED		Schmelz-Lot	Rot	BSK gefallen
Onten		Unterwegs (Fährt)	gelb blinken	BSK in Zwischenstellung
		Störung	gelb-rot blinken	

Diese Zustände werden an den jeweiligen LEDs zur Anzeige gebracht:

Abb.: LEDs je Gruppe



Klappe Multi-State-Input Objekt BOBACONNECT

IK.								
0		_		_				
5		* BACne	t Network #14 🖤 Devic	æ #8 "RDC742 (SN: 4273	864704	4)" / "Testdescription"		
lavigatio		8	Multi-State-Input #0 MultiStateInput 0 "Klappe 0"					
ŧ.	System view Technical (tree) Technical (list)	Properti	ies					
	🐻 Known 🛷 New 👩 🔁 📮 🖗							
	7 Geräte, 189 Objekte (aus einfacher Suche)	_	Eigenschaft	Datentyp		Wert		
	🗾 Binary-Input #0 "BinaryInput 0" / "Taster 0"	Identifi	cation					
	🗾 Binary-Input #1 "BinaryInput 1" / "Taster 1" 🦳	TUG						
	🔟 Binary-Input #2 "BinaryInput 2" / "Taster 2"		Object-Identifier	Object-Identifier		Multi-State-Input (13)	66	
	🗓 Binary-Input #3 "BinaryInput 3" / "Taster 3"		Object-Type	Object-Type		Multi-State-Input (ID:13)		
	Bitstring-Value #0 "BitStringValue 0" / "Rauchmek							
	Bitstring-Value #1 "BitStringValue 1" / "Rauchmek		Object-Name	Character-String		MultiStateInput 0		
	Bitstring-Value #2 "BitStringValue 2" / "Rauchmek		Description	Character-String		Klappe 0		
	Bitstring-Value #3 "BitStringValue 3" / "Rauchmek	Status	_					
	😻 Multi-State-Input #0 "MultiStateInput 0" / "Klappe				COV	A B		
	😻 Multi-State-Input #1 "MultiStateInput 1" / "Klappe		Status-Flags	Status-Flags	00			
	1 Multi-State-Input #2 "MultiStateInput 2" / "Klappe		Reliability	Reliability		No-Fault-Detected (ID:0)		
	Multi-State-Input #3 "MultiStateInput 3" / "Klappe	Multi-C	tate-Input Specific	Properties				
	Multi-State-Value #0 "Test Status 0" / "Description"	india s	tate-input specific	. Properties	COV			
	80 Multi-State-Value #1 "Test Status 1" / "Description"		Present-Value	Unsigned-Integer	00	moving (2)		
	80 Multi-State-Value #2 "Test Status 2" / "Description"		Number-Of-States	Unsigned-Integer		5		
	8 Multi-State-Value #3 "Test Status 3" / "Description"		-	-				
	Structured-View #0 "BinaryInput 0 trigger cont	- E	State-Text	Character-String[5]		["closed", "moving", "open", "tusible link", "fault"]		
A = =	Structured-View #1 "BinaryInput 1 trigger cont							

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, bzw. wertet diese entsprechend aus.

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

State Texte: 1 = closed2 = moving4 = fusible link 3 = open5 = fault

<u>Auswirkungen :</u>

Überwachung der BSK findet bei Anlage EIN (ON) laut Tabelle statt. Fahrbetrieb und Schmelzlot werden 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	Klappen Status	Klappen Status	
BI Flap closed	BI Flap open	MSV	MSI Flap State	MSI Input Status Flags	
			closed-movig-open- fusible_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	fusible_link	In_Alarm	
0	1	on / EIN	open		
1	0	on / EIN	fusible_link	In_Alarm	
1	1	on / EIN	fault	In_Alarm	

Rauchschalter Status Structured-View Objekt

-	powered by WSW Solutions							
(
📶 Navigation 📙	Configuration Explorer * System view Technical (tree) Technical (list)	Prope	Cnet Ne Sti Bi erties	etwork #14 🚿 Device #8 "Rf ructured-View #256 tStringValue 0 sens)C742 (SN: 4273864704)"/"T sor configuration	estdescription".		
	Known WNew Search Sarch							
	7 Geräte, 189 Objekte (aus einfacher Suche) Bitstring-Value #0 "BitStringValue 0" / "Rauchmek Bitstring-Value #1 "BitStringValue 1" / "Rauchmek Bitstring-Value #2 "BitStringValue 2" / "Rauchmek	Iden	tifica	Eigenschaft tion Object-Identifier	Datentyp Object-Identifier	Wert Structured-View (29)	✔ # 25	6
	 Bitstring-Value #3 "BitStringValue 3" / "Rauchmek Multi-State-Input #0 "MultiStateInput 0" / "Klappe Multi-State Input #1 "MultiStateInput 0" / "Klappe 			Object-Type	Object-Type	Structured-View (ID:29)		
	<pre>Wold-State-Input = 1 MultistateInput 1 / Nappe Multi-State-Input #2 "MultiStateInput 2" / "Klappe Multi-State-Input #3 "MultiStateInput 3" / "Klappe</pre>			Description	Character-String			
	88 Multi-State-Value #0 "Test Status 0" / "Description" 88 Multi-State-Value #1 "Test Status 1" / "Description" 88 Multi-State-Value #2 "Test Status 2" / "Description"			Subordinate-List	Device-Object-Reference[:	[[Device-Identifier=Device	#10, Object-Ident	tifier=Bina
	 Multi-State-Value #3 "Test Status 3" / "Description" Structured-View #0 "BinaryInput 0 trigger cons Structured-View #1 "BinaryInput 1 trigger cons 			Device-Identi Object-Identi	ifier Object-Identifier 🚀	Device (8)	# 10 # 1] 60] 60
	 Structured-View #2 "BinaryInput 2 trigger cons Structured-View #3 "BinaryInput 3 trigger cons Structured-View #256 "BitStringValue 0 sensor 		•	[2] Device-Object-R Device-Identi	eference [Device-Identifie ifier Object-Identifier 🏈	r=Device #10, Object-Identifier=1	Binary-Input #2]	
	 Structured-View #257 "BitStringValue 1 sensor Structured-View #258 "BitStringValue 2 sensor Structured-View #259 "BitStringValue 3 sensor 			Object-Identi I3 Device-Object-R	fier Object-Identifier 🧃	Binary-Input (3)	# 2	00
	 Structured-View #512 "MultiStateInput 0 sensor Structured-View #513 "MultiStateInput 1 sensor Structured-View #514 "MultiStateInput 2 sensor 			Device-Identi Object-Identi	ifier Object-Identifier 🚀	Device (8)	# 10 # 0] 60] 60
	● Structured-View #515 "MultiStateInput 3 sensor			Subordinate-Annotations	Character-String[3]	["polluted", "error", "smoke	="]	
	Open in Configuration EDE Export			[1] Character-String [2] Character-String [3] Character-String	polluted error smoke			

Abb. Structured-View Objekt Rauchschalter Status

Für jede der 4 Gruppen gibt es ein "Structured-View" (BitStringValue #x sensor configuration) Objekt (#256 bis #259). Mit diesem Objekt werden das Device und die Datenpunkte (physikalische oder virtuelle) des zu überwachenden Gerätes, hier ein Rauschschalter, zugeordnet. Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein.

Einzutragen sind

Rauchschalter-Meldungen "Verschmutzt" (polluted)

1 1000	shoonallor molaangon "r	credininater (penatea)								
1	Device-Identifier:	Device(8)	DeviceID (hier die 10)							
1	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die 1)							
Rauc	Rauchschalter-Meldungen "Störung" (error)									
2	Device-Identifier:	Device(8)	DeviceID (hier die 10)							
2	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die 2)							
Rauc	Rauchschalter-Meldungen "Alarm" (smoke)									
3	Device-Identifier:	Device(8)	DeviceID (hier die 10)							
3	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die 0)							

Die Statusänderung (COV = Change of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann 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 dies an der Statusanzeige unter dem Taster durch eine LED rot signalisiert.



Rauchschalter Status BitStringValue Objekt

ROBACCONNECT ered 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 entsprechend aus.

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

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

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

<u>Gruppen</u>	<u>Bezeichnung</u>	<u>Status</u>	LED-Status	Funktion	
LED	Rauchmelder	ОК	Aus		
Oben	LED verschmutzt	verschmutzt	gelb	RM verschmutzt	
1.50	Developmentation	ОК	Aus		
LED	Rauchmelder	Störung elektrisch	gelb blinken	RM gestört	
witte	LED Storung	Ausgelöst Rauch	rot blinken	RM ausgelöst	
		Offen	grün	BSK Offen	
	Klappe	Zu	Aus	BSK geschlossen	
LED		Schmelz Lot	rot	BSK gefallen	
onten	LED ZO	Unterwegs (Fährt)	gelb blinken	BSK in Zwischenstellung	
		Störung	gelb-rot blinken		

Diese Zustände werden an den jeweiligen LEDs zur Anzeige gebracht:

Abb. LEDs je Gruppe



Modulbeschreibung RDC743 und RDC744 Applikation Brandschutzklappe

Diese Typen sind zusätzlich mit einer Laufzeitüberwachung der Brandschutzklappen ausgestattet. Die Grundfunktionen sind wie bei den Modulen RDC741 (=RDC743) und RDC712 (=RDC744). Grundfunktion ist die Visualisierung von bis zu 12 Digitaleingängen als LED-Anzeige mit festen definierten Farben grün/gelb/rot, mit und ohne blinken (siehe Tabelle). Die anzuzeigenden Statusmeldungen werden von den jeweiligen physikalischen oder virtuellen Datenpunkten mittels BACnet "Structured-View" Objekt abonniert. Die Statusänderung (COV = Change of Value) und Zustände am physikalischen Eingang werden durch den Service "Subscribe_COV"

übermittelt. Dieser Zustand kann als "Confirmed_COV" oder "Unconfirmed_COV" übermittelt werden.

Konfiguration des RDC743 und RDC744

6									
	C v &	mit eine	r 30-Ta	ge-Testlizenz. <u>m</u>	<u>ehr</u> 🕕 Ihr Bro	wser h	at schlechte Performance! <u>mehr</u>		
See Navigation	Konfiguration @ Explorer # Systemskt Technisch (Liste)	<	*- Lokales BACnet-MS/TE-Netzwerk #5 Coexics #17 "BDC743 (SN: 17)" / "Testdescription" Analog-Input #0 Flap 0 open time "N/A" Eigenschaften						
	Bekannte 🛷 Neue 💿 😰 🕞 📮 🖗								
	8 Geräte, 394 Objekte (aus einfacher Suche)			Eigenschaft	Datentyp		Wert		
	Verice #10 "DEV-KRM-BAC" / "KRM Device Object"	Ide	ntifika	tion					
	E 2 Device #15 RDC 601 SN: 00010880 / BSK inte DeviceDesc E 2 Device #17 "RDC743 (SN: 17)" / "Testdescription"		Ξ	Object-Identifier	Object-Identifier		😼 Analog-Input (0) 💙 # 0 🖉		
	👧 Analog-Input #0 "Flap 0 open time" / "N/A"	-		Object-Type	Object-Type		Analog-Input (ID:0)		
	😼 Analog-Input #1 "Flap 1 open time" / "N/A"	1							
	3 Analog-Input #2 "Flap 2 open time" / "N/A"	-	=	Object-Name	Character-String		Flap 0 open time		
	Manalog-Input #3 "Flap 3 open time" / "N/A"		\equiv	Description	Character-String		N/A		
	Manalog-Input #4 "Flap 4 open time" / "N/A"	Stat	tus						
	M Analog-Input #5 Flap 5 open time / N/A			Status-Flans	Status-Flans	007	◎ <u>∧</u> <u>/</u> 212		
	Analog-Input #7 "Flap 7 open time" / "N/A"			Status Hags	Subus Hugs	~			
	Analog-Input #8 "Flap 8 open time" / "N/A"	1	≡	Event-State	Event-State		Normal (ID:0)		
	🐻 Analog-Input #9 "Flap 9 open time" / "N/A"	1	≡	Reliability	Reliability		No-Fault-Detected (ID:0)		
	🔣 Analog-Input #10 "Flap 10 open time" / "N/A"	Ana	loa-In	put-spezifisch	e Eigenschaften				
	🐯 Analog-Input #11 "Flap 11 open time" / "N/A"			Dracant Value	Real	007	0.0		
	🐻 Analog-Input #256 "Flap 0 close time" / "N/A"	-		Present-value	Neal		0,0		
	Manalog-Input #257 "Flap 1 close time" / "N/A"	1	≡	Min-Pres-Value	Real		0,0		
	3 Analog-Input #258 "Flap 2 close time" / "N/A"	1	≡	Max-Pres-Value	Real		60.0		
	M Analog-Input #259 "Flap 3 close time" / "N/A"	1							
	Analog-input #260 flap 4 close time / N/A								
	Malandog-input #261 riap 5 close time / N/A	1							
	Analog-hipdt #202 filep 6 close bime / N/A	-							

Analog-Input Objekte Klappen-Fahrzeiten

Für die Konfiguration gibt es je Datenpunkt ein Analog-Input Objekt:

open time = "AUF" Fahrtzeit der Klappe. Hier werden in den **Min-Pres-Value** und **Max-Pres-Value** die Werte der Klappe eingetragen.

close time = "ZU" Fahrtzeit der Klappe. Hier werden in den **Min-Pres-Value** und **Max-Pres-Value** die Werte der Klappe eingetragen.

latency time = Verzögerungszeit bis der jeweilige Endlagenschalter reagiert. Hier werden in den **Min-Pres-Value** und **Max-Pres-Value** die Werte der Klappe eingetragen.

UNKNOWN time = Messwert Zeit (intern)

Bei Unter- oder Überschreitung der Grenzwerte wird das Status-Flag gesetzt und zusätzlich ein Multistate-State-Value (#0 bis #11) "Flap" (flap 0-11) je nach Status gesetzt. Dieser steuert dann auch die LEDs an der Frontseite des Moduls an.

Multistate Value Objekt Klappen-Status

R			
	🕐 🗸 🚯 Dies ist eine	Demov	oversion. <u>mehr</u> Ihre Sitzung wird in 240 Min. beendet. <u>mehr</u>
福丽 Navigation	Image: System schr Technisch (Baum) Technisch (Liste) Image: System schr Image: System schr Technisch (Baum) Technisch (Liste) Image: System schr Image: System schr Technisch (Baum) Technisch (Liste) Image: System schr Image: System schr Technisch (Baum) Technisch (Liste) Image: System schr Image: System schr Image: State schr Susche Image: Susche schr Image: Susche schr	<u>لما</u> دور	skales BAChet MS/TP-Netzwerk #5 😻 Device #2 "RDC/44 (SN: 32)" / "Testdescription" Multi-State-Value #0 Flap @ "Description" enschaften 🗰 Kalender 🍪 Versorgung Allgemein
	15 Geräte, 132 Objekte (bekannte Geräte +1)		Eigenschaft Datentyp Wert
	Binary-Input #2 "BinaryInput 2" / "Taster 2"	Ŧ	🔳 Status-Flags 😅 🎬 🐴 🖉
	Binary-Input #3 "BinaryInput 3" / "Taster 3" Binary Value #366 "minare Gas accord" / "minare for BIG		Reliability Reliability No-Sensor (ID:1)
	Binary-Value #256 mirror for B100 / mirror for B1 Binary-Value #257 "mirror for B101" / "mirror for B1	Mult	ti-State-Value-spezifische Eigenschaften
	Binary-Value #258 "mirror for BI02" / "mirror for BIC		Present-Value Unsigned-Integer
	Binary-Value #259 "mirror for BI03" / "mirror for BIC		Number-Of-States Unsigned-Integer 9
	Binary-Value #65280 / Multi-State-Value #0 "Flap 0" / "Description"		State-Text Character-String[9] ["closed", "moving", "open", "fusible link", "alarm",
	8 Multi-State-Value #1 "Flap 1" / "Description"	L	[1] Character-String closed
	8 Multi-State-Value #2 "Flap 2" / "Description"	L	
	Multi-State-Value #3 "F1ap 3" / "Description"		
	Multi-State-Value #4 "Flap 4" / "Description"		[3] Character-String Open
	Multi-State-Value #5 Flap 5 / Description		[4] Character-String fusible link
	8 Multi-State-Value #7 "Flap 7" / "Description"	L	5 Character-String alarm
	×		[6] Character-String Device alarm
	🔗 Öffnen in der Konfiguration		Time fault close
			[8] Character-String Time fault movin
	EDE-Export		[9] Character-String Time fault open
	🛞 Objekt löschen (BACnet Delete-Object)		
	🍀 Multi-State-Value #0 in die Datenpunkt-Registratur aufnehmer		Multi-State-Value #0 Flap

Multi-State-Value Klappen Status

MULTI-STATE-VALUE Klappe:

2 = moving fährt 3 = open offen	
4 = fusible linkSchmelzlot ausgelöst5 = alarmStörung6 = Device alarmGeräte Störung7 = Time fault closedLaufzeitfehler geschlossen (Klappe fährt AUF nac8 = Time fault movingLaufzeitfehler (Grenzwert allgemein AUF oder ZU9 = Time fault openLaufzeitfehler geschlossen (Klappe fährt ZU nach	h ZU)) n AUF)

Structured-View Objekt: Rückmeldungen und Abschaltungen

R									
	• 4	i Dies ist eine	e Demov	ersior	n. <u><i>mehr</i></u>	\Lambda Ihre Si	tzung wird in 239 Min. been	det. <u>mehr</u>	
In Navigation	Konfiguration C Explorer # @ Systemsicht Technisch (Baum) Sestante Neue Geränte Suche Sone Sestante	לאי <u>Lok</u> Eiger	ales BA Str Fla	ACnet-MS/TP ructured ap 0 sen: ften	<u>Netzwerk #5</u> - View #5 : sor configu	② Device #2 "RDC744 (SN: 32)" 12 Jration —	/"Trestdescription"		
	15 Geräte, 132 Objekte (bekannte Geräte +1)			_	Eigenscha	aft	Datentyp	Wert	
	Multi-State-Value #65280 "Baudrate" / "Ba	audrate for M ▲	LUCID		Object Id	ontifior	Object Identifier	Etructured View (20)	¥ # 512
	Structured-View #0 BinaryInput 0 trig	ger config			Object-10	enumer	objectiventinen	tuccured view (25)	. # 012
	Structured-View #2 "BinaryInput 2 trig	ger config		Ξ	Object-Ty	pe	Object-Type	Structured-View (ID:29)	·
	Structured-View #3 "BinaryInput 3 trig	gger config		Ξ	Object-Na	ime	Character-String	Flap 0 sensor configuration	
	醒 Structured-View #512 "Flap Ø sensor co	onfiguratio		Ξ	Descriptio	n	Character-String		
	Structured-View #513 "Flap 1 sensor co	onfiguratic	Strue	ture	d-View-s	pezifische	Eigenschaften		
	E Structured-View #514 Flap 2 sensor co	onfiguratic	Ŧ		Subordina	te-List	Device-Object-Reference[7]	[[Device-Identifier=Device #419430	3, Object-Identifi
	Structured-View #516 "Flap 4 sensor co Structured-View #517 "Flap 5 sensor co Structured-View #517 "Flap 5 sensor co	onfiguratic			Subordina	ite-Annotatio	ons Character-String[7]		ommand", "ventila
	Structured-View #518 "Flap 6 sensor co	onfiguratic			[1]	Character-Stri	ing flap opened		
	Structured-View #519 "Flap 7 sensor compared by the sensor compar	onfiguratic			[2]	Character-Stri	ing flap closed		
	🕎 Structured-View #520 "Flap 8 sensor co	onfiguratic			= [3]	Character-Stri	ing flap command		7
	Structured-View #521 "Flap 9 sensor co	onfiguratic				Character Chri	ventilation off 0		-
	tem Structured-View #522 Flap 10 sensor o	configurati ♥				Character-Str	ventilation off 1		-
	C Officer in des Kasfingerhing				E [5]	Character-Str	ng ventriation off 1		-
	W onnen in der Könnguration				[6]	Character-Stri	ventilation off 2		-
	EDE-Export				[7]	Character-Stri	ing ventilation off 3		

Abb.: Structured-View je Klappe mit Abschaltung

Für jede der 12 Status LEDs oder der Gruppe gibt es ein "Structured-View" Sensor Configuration Objekt (#512 bis #523) und für jede der 4 Gruppen gibt es ein "Structured-View" (Multistate Input 0 sensor configuration) Objekt (#512 bis #515). Mit diesen Objekten werden das Device und die Datenpunkte (physikalische oder virtuelle) des zu überwachenden Gerätes, hier eine BSK, zugeordnet. Wichtig: Die Geräte bzw. Datenpunkte müssen im selben MS/TP-Netzwerk sein.

Einzutragen sind

Klappe	en-Meldungen Endlager	ischalter "ZU" (flap open)	
1	Device-Identifier:	Device(8)	DeviceID (hier die xx)
1	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die x)
Klappe	en- Meldungen Endlage	nschalter "AUF" (flap closed)	
2	Device-Identifier:	Device(8)	DeviceID (hier die xx)
2	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die x)
		[Binary-Output oder Binary-Value]	
Anlage	<u>en-Meldung "EIN" (flap c</u>	command)	
3	Device-Identifier:	Device(8)	DeviceID (hier die xx)
3	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die x)
		[Binary-Output oder Binary-Value]	
Anlage	<u>e-Ausschalten "AUS" (ve</u>	entilation off 0-3)	
4-7	Device-Identifier:	Device(8)	DeviceID (hier die xx)
4-7	Objekt-Identifier:	Objekt-Type (hier Binary-Output) [Binary-Output oder Binary-Value]	Objekt-Number (hier die x)

Die Statusänderung (COV = Change of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann als "Confirmed_COV" oder "Unconfirmed_COV" übermittelt werden.

LED - Anzeige des RDC743 und RDC744 :

		Funktion	Farbe	Beschriftung
LEDx Status		ОК	Aus	
Oben	LED Verschmutzt	Taster gedrückt	orange	RM verschmutzt
		Test ausgelöst	grün	
		MAC fehlt	rot blinken	
LEDx	Rauchmelder	OK	Aus	
Mitte	LED Störung	Verschmutzt	orange	
		Störung	orange blinken	RM gestört
		Ausgelöst	rot	RM ausgelöst
LEDx	Klappe	"Offen"	grün	BSK Offen
Unten	LED ZU	"Zu"	orange	BSK geschlossen
		Schmelz Lot	rot blinken	BSK gefallen
		"Unterwegs" (Fährt)	grün blinken	BSK in Zwischenstellung
		Laufzeitfehler	rot	Störung
		Laufzeitfehler "ZU"	blinken grün/orange	Störung ZU
		Laufzeitfehler "AUF"	Blinken rot/orange	Störung AUF



Modulbeschreibung RDC745 Applikation Rauchschalter

Das Modul besteht aus 4 Gruppen mit je einem Taster und drei mehrfarbigen LEDs. Die Applikation bildet die Funktion eines Rauchmelders (RM) ab. Über je drei LEDs werden die Meldungen des Rauchmelders dargestellt. Mit dem Taster kann ein Reset/Test des Rauchmelders ausgelöst werden.

Die anzuzeigenden Statusmeldungen werden von den jeweiligen physikalischen oder virtuellen Datenpunkten mittels BACnet "Structured-View" Objekt abonniert. Die Statusänderung (COV = Change 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 RDC745

Analog Input Objekt "Value Age"



Abb. Analog Input Objekt Smoke Value Age

Für jeden der 4 Rauchmelder gibt es ein Analog Input Objekt "Value Age". Dieses Objekt zeigt das Alter des aktuellen Messwertes an.

Unter Verwendung des Properties "Max-Present-Value" in diesem Objekt kann der Ablesezyklus des Moduls eingestellt werden.

Ein Wert von 10s bedeutet, dass ein kompletter Messzyklus dieses RDC innerhalb von zehn Sekunden erfolgen muss. Bei unterschiedlichen Einstellungen je Kanal gilt immer der kürzeste Wert.

Bei zeitlich unkritischen Messungen kann so der Datenverkehr auf dem Bus deutlich verringert werden.



Binary-Input Objekt

R												
	• • 4		_									
Navigation	Image: Systemschit Technisch (Saun) Technisch (Liste) Image: Systemschit Technisch (Liste)	♦ BAChetHetzwerk #2 Cente #37 EDC745 [SH: 37! / Trestdesoration" Binary-Input #0 "Taster 0" BinaryInput 0 "Taster 0"										
	Image: Strain and the strain and	Eigenschaft Datentyp Wert										
		Object-Identifier Object-Identifier Imary-Input (3) #0 Object-Type Object-Type Binary-Input (1D:3)										
		Dipect-Name Character-String BinaryInput 0 Description Character-String Taster 0 Device-Twee Character-String mode: 0										
	Device #35 "RDC743 (SN: 35)" / "Testdescription" Device #37 "RDC745 (SN: 37)" / "Testdescription" Analog-Input #66304 "Smoke 0 value age" / "N/A"	Status										
	33 Analog-Input #66305 "Smoke 1 value age"/"N/A" 33 Analog-Input #66306 "Smoke 2 value age"/"N/A" 33 Analog-Input #66307 "Smoke 3 value age"/"N/A"											
	<pre>10 Binary-Input #0 "BinaryInput 0" / "Taster 0" 10 Binary-Input #1 "BinaryInput 1" / "Taster 1" 10 Binary-Input #2 "BinaryInput 2" / "Taster 2"</pre>	Binary-Input-spezifische Eigenschaften Present-Value Binary-P-V 33 Inactive (0) ~										
	O Öffnen in der Konfiguration	Polarity Normal (ID:0) ~ Inactive-Text Character-String Inactive Inactive										
	IIII EDE-Export	Active lext character-string Active										

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 Active (1) angezeigt. Über die "Polarity" kann der Status invertiert werden.

Funktion:

Mit dem "Device Type" kann die Funktion Taster ausgewählt werden.

z.B.: MODE : 0 =

beim 1. Drücken "Present-Value" wechselt von 0 nach 1 Sendet eine "1" an den Datenpunkt

beim 2. Drücken "Present-Value" wechselt von 0 nach 1 Sendet eine "0" an den Datenpunkt



Abb. Mode Funktion Taster



Binary-Value Objekt

0	POBA CONNECT powered by WSW Solutions								
(O - 4	_	_			-			
See Navigation	Konfiguration Image: Constraint of the second sec	* BAC	BAChetNetzwerk #2 Device #37.TRDC745 (SN: 37)" / Trestdescription Binary-Value #256 mirror for BIO0 "mirror for BIO0" Eigenschaften Maender Salender Second Algemein						
	Bekannte 🛷 Neue 💿 📴 🕞 😓 🗜 🚱								
	13 Geräte, 557 Objekte (aus einfacher Suche)		Eig	enschaft	Datentyp		Wert		
	Opevice #15 "RDC 633 SN: 00011235" / "DeviceDesc"	Ident	tifikation	1					
	Device #32 "RDC742 (SN: 35)" / "Testdescription" Ø Device #33 "RDC743 (SN: 33)" / "Testdescription"		🔳 Obj	ject-Identifier	Object-Identifier		Image: Binary-Value (5) // # 256		
	■ 2 Device #35 "RDC743 (SN: 35)" / "Testdescription"		🔳 Obj	ject-Type	Object-Type		Binary-Value (ID:5)		
	E 💇 Device #37 "RDC745 (SN: 37)" / "Testdescription"		Obj	ject-Name	Character-String		mirror for BIO0		
	Mail Analog-Input #66304 "Smoke 0 value age" / "N/A"			corintian	Character String		mirror for PIOO		
	33 Analog-Input #66305 "Smoke 1 value age" / "N/A"	Chata		scripuori	character-sching	_			
	Manalog-Input #66306 "Smoke 2 value age" / "N/A"	Statu	15			_			
	Analog-Input #66307 Smoke 3 value age / 'N/A Bipap/Input #0 "BicareuTarut 0" / "Taster 0"		Out	t-Of-Service	Boolean		False V		
	Binary-Input #1 "BinaryInput 1" / "Taster 1"	÷	🔳 Sta	itus-Flags	Status-Flags	×			
	Binary-Input #2 "BinaryInput 2" / "Taster 2"		E Reli	iability	Reliability		No-Fault-Detected (ID:0)		
	Binary-Input #3 "BinaryInput 3" / "Taster 3"	Binar	y-Value	-spezifische	Eigenschaften				
	Binary-Value #256 "mirror for BI00" / "mirror for BI00"		E Pre	sent-Value	Binary-P-V	00V	Inactive (0) V		
	Binary-Value #257 "mirror for BI01" / "mirror for BI01"			ctive Text	Character String		Inactive		
	Binary-Value #258 "mirror for BI02" / "mirror for BI02"			CUVE-TEXT	character-scring		Inderve		
	Binary-Value #259 "mirror for BI03" / "mirror for BI03"		■ Act	ive-Text	Character-String		Active		
	Binary-Value #65280 "LAMP TEST" / "Enable LAMP TEST Mode"								

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 Taster

R										
0) 🔿 🗸 🕹									
Navigation	Konfiguration Q Explorer # Q	* B	* BACnetNetwerk #2 @ Denke #37 TBDC745 (SN: 37) / Testdescripton* Structured-View #0 BinaryInput 0 trigger configuration Egenschaften							
	Bekannte Whene Suche Geräte, 557 Objete (aus einfacher Suche) There highlight (aus einfacher Suche)			Eigenschaft	Datentyp	Wert				
	 Binary-Value #258 "mirror for B101" / "mirror for B101" Binary-Value #258 "mirror for B101" / "mirror for B101" Binary-Value #258 "mirror for B102" / "mirror for B102" 	Ide	ntifik	Object-Identifier	Object-Identifier	Structured-View (29) # 0				
	 Binary-Value #259 "mirror for BIO3" / "mirror for BIO3" Binary-Value #65280 "LAMP TEST" / "Enable LAMP TEST Mode" Bi-String-Value #65280 "LAMP TEST" / "Enable LAMP TEST Mode" 		Ξ	Object-Type Object-Name	Object-Type Character-String	Structured-View (ID:29) V BinaryInput 0 trigger configuration				
	 Bit-String-Value #1 "BitStringValue 1" / "Rauchmelder 1" Bit-String-Value #2 "BitStringValue 2" / "Rauchmelder 2" 		Ξ	Description Device-Type	Character-String Character-String	prio: 16				
	 Bit-String-Value #3 "BitStringValue 3" / "Rauchmelder 3" Multi-State-Value #65280 "Baudrate" / "Baudrate for MSTP bus" Structure/Vew #6 "State structure." 	Str		ed-View-spezifische Eig Subordinate-List	genschaften Device-Object-Reference[4]	[[Device-Identifier=Device #4194303, Object-Identifier=Binar				
	SUBCURED-VEW #J BinaryInput 0 trigger configuration Structured-Vew #J "BinaryInput 1 trigger configuration Structured-Vew #J "BinaryInput 2 trigger configuration Structured-Vew #J 256 "BitStringValue 0 sensor configure Structured-Vew #J 256 "BitStringValue 1 sensor configure Structured-Vew #J 8" BitStringValue 2 sensor configure Structured-Vew #J 8" BitStringValue 2 sensor configure Structured-Vew #J 8" BitStringValue 2 sensor configure			[1] Device-Object-F Device-Ident Object-Ident	Reference [Device-Identifier=Device-Identifier=Device-Identifier 🦅 Device	vice #4194303, Object-Identifier=Binary-Value #4194303] te (8)				
			•	[2] Device-Object-F [3] Device-Object-F	Reference [Device-Identifier=D	vice #4194303, Object-Identifier=Binary-Value #4194303] vice #4194303, Object-Identifier=Binary-Value #4194303]				
	Structured-View #259 "BitStringValue 3 sensor configure v		æ	(4) Device-Object-F	Reference [Device-Identifier=Dev	vice #4194303, Object-Identifier=Binary-Value #4194303]				
	Öffnen in der Konfiguration Im EDE-Export	Œ		Subordinate-Annotation	ns Character-String[4]	["trigger 0", "trigger 1", "trigger 2", "trigger 3"]				

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 werden das Device und die Datenpunkte (physikalische oder virtuelle) des zu übersteuernden Gerätes zugeordnet.

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

Einzutragen sind

Device-Identifier:	Device(8)	DeviceID (hier die xx)
Objekt-Identifier:	Objekt-Type (hier Binary-Value)	Objekt-Number (hier die x)
	[Binary-Output oder Binary-Value]	

Die Statusänderung (COV = Change of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann 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.



Structured-View Objekt Rauchschalter Status

R						
	○ ~ 4	-	-			
Systemscht Technisch (Liste)				werk #2 🐔 Device #37 Rd ictured-View #256 StringValue 0 sens en	<u>XC745(SN:37)"/"Testdescripton"</u> or configuration	
	Til Geräte Vuche V Suche V Suc		E	igenschaft	Datentyp	Wert
	Binary-Value #256 "mirror for BIO0" / "mirror for BIO0" ^	Iden	tifikatio	on	/F	
	Binary-Value #257 "mirror for BI01" / "mirror for BI01"			bject-Identifier	Object-Identifier	Structured-View (29)
	Binary-Value #258 "mirror for BI02" / "mirror for BI02" Binary-Value #259 "mirror for BI03" / "mirror for BI03"			bject-Type	Object-Type	Structured-View (ID:29)
	Binary-Value #65280 "LAMP TEST" / "Enable LAMP TEST Mode"			bject-Name	Character-String	BitStringValue 0 sensor configuration
	Bit-String-Value #0 "BitStringValue 0" / "Rauchmelder 0"		ΞD	escription	Character-String	
	Bit-String-Value #1 "BitStringValue 1" / "Rauchmelder 1" Bit-String-Value #2 "BitStringValue 2" / "Pauchmelder 2"	Stru	ctured-	-View-spezifische Eig	enschaften	
	 Bit-String-Value #3 "BitStringValue 3" / "Rauchmelder 3" 		E S	ubordinate-List	Device-Object-Reference[4]	[[Device-Identifier=Device #4194303, Object-Identifier=B
	😻 Multi-State-Value #65280 "Baudrate" / "Baudrate for MSTP bus"	L	Đ	[1] Device-Object-Re	eference [Device-Identifier=De	vice #4194303, Object-Identifier=Binary-Value #4194303]
	Structured-View #0 "BinaryInput 0 trigger configuration			[2] Device-Object-Re	eference [Device-Identifier=Dev	vice #4194303, Object-Identifier=Binary-Value #4194303]
	Structured-View #1 "BinaryInput 1 trigger configuration			[3] Device-Object-Re	ference [Device-Identifier=Dev	vice #4194303 Object-Identifier=Binary-Value #4194303]
	Structured-View #2 "BinaryInput 2 trigger configuration Structured-View #2 "BinaryInput 2 trigger configuration			[4] Device Object-Re	farance [Device-Identifier=Dev	vice #4194202 Object Identifier=Binany Value #4194202]
	Structured-View #256 "BitStringValue 0 sensor configuration				Terence [Device Identifier=Device	vice #4194909, Object Identifici = bindi y valde #4194909j
	Structured-View #257 "BitStringValue 1 sensor configura		1	r .		
	Structured-View #258 "BitStringValue 2 sensor configure		≣ s	ubordinate-Annotations	Character-String[4]	["polluted", "error", "smoke", "ventilation off"]
	Structured-View #259 "BitStringValue 3 sensor configure 🗸	L		[1] Character-String	polluted	
		L		[2] Character-String	error	
	Öffnen in der Konfiguration			[3] Character-String	smoke	
	EDE-Export			[4] Character-String	ventilation off	

Abb. Structured-View Objekt Rauchschalter Status

Für jede der 4 Gruppen gibt es ein Structured-View BitStringValue sensor configuration Objekt (#256 bis #259). Mit diesem Objekt werden das Device und die Datenpunkte (physikalische oder virtuelle) des zu überwachenden Gerätes, hier ein Rauschschalter, zugeordnet.

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

Einzutragen sind

Rauch	Rauchschalter-Meldungen "Verschmutzt" (polluted)						
1	Device-Identifier:	Device(8)	DeviceID (hier die xx)				
1	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die x)				
Rauch	<u>schalter-Meldungen "Stö</u>	<u>rung" (error)</u>					
2	Device-Identifier:	Device(8)	DeviceID (hier die xx)				
2	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die x)				
Rauch	schalter-Meldungen "Ala	rm <u>" (smoke)</u>					
3	Device-Identifier:	Device(8)	DeviceID (hier die xx)				
3	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die x)				
Rauch	Rauchschalter-Meldungen "Luftmangel" (ventilation off)						
4	Device-Identifier:	Device(8)	DeviceID (hier die xx)				
4	Objekt-Identifier:	Objekt-Type (hier Binary-Input)	Objekt-Number (hier die x)				

Die Statusänderung (COV = Change of Value) und Zustände an den abonnierten Datenpunkten werden durch den Service "Subscribe_COV" übermittelt. Dieser Zustand kann 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 dies an der Statusanzeige unter dem Taster durch eine LED rot signalisiert.



BitStringValue Objekt Rauchschalter Status

R	DBACCONNECT powered by WSW Solutions						
	• • 4						
Mavigation	Image: Systemath Technisch (Baun) Technisch (Liste) Image: Systemath	BACnet	* BACnet/tetzwerk #2 Device 537 "BDC745 (SN: 37)" / Testdescription" Bit-String-Value #0 BitStringValue 0 "Rauchmeder 0" Egenschaften machender				
	Serate Quelle Sude Sude Gerate S57 Objekte (aus einfacher Suche)	Identifi	Eigenschaft	Datentyp		Wert	
	Binary-Value #257 "mirror for BIO1" / "mirror for BIO1" Binary-Value #258 "mirror for BIO2" / "mirror for BIO2" Binary-Value #259 "mirror for BIO3" / "mirror for BIO3"		Object-Identifie	r Object-Identifier Object-Type		● Bit-String-Value (39)	
	Binary-Value #65280 "LAMP TEST" / "Enable LAMP TEST Mode" Bit-String-Value #0 "BitStringValue 0" / "Rauchmelder 0" Bit-String-Value #1 "BitStringValue 1" / "Rauchmelder 1"		Object-Name Description	Character-String Character-String		BitStringValue 0 Rauchmelder 0	
	 Bit-String-Value #2 "BitStringValue 2" / "Rauchmelder 2" Bit-String-Value #3 "BitStringValue 3" / "Rauchmelder 3" Multi-State-Value #65280 "Baudrate" / "Baudrate for MSTP bus" 	Status	Out-Of-Service Status-Flags	Boolean Status-Flags	007 20	False V	
	Structured-Vew #0 "BinaryInput 0 trigger configuration Structured-Vew #1 "BinaryInput 1 trigger configuration Structured-Vew #2 "BinaryInput 2 trigger configuration		Event-State Reliability	Event-State Reliability		Normal (ID:0) V No-Sensor (ID:1) V	
	Structured-Vew #3 "BinaryInput 3 trigger configuration Structured-Vew #256 "BitStringValue 0 sensor configure Structured-Vew #257 "BitStringValue 1 sensor configure	Bit-Strin	ng-Value-spezifi Present-Value	sche Eigenschaften Bit-String	007 007		
	Structured-Vew #258 "BitStringValue 2 sensor configure Structured-Vew #259 "BitStringValue 3 sensor configure <		Bit-Text	Character-String[4]		["polluted", "error", "smoke", "ventilation off"]	

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) eingetragenen Werte den LEDs zu, bzw. wertet diese entsprechend aus.

Anzeige Optionen im "BitStringValue" (#0 bis #3), hier mit dem "Present-Value" (1-4):

State Texte: 1 = "Verschmutzt" (polluted) 2 = "Störung" (error) 3 = "Alarm" (smoke) 4 = "Luftmangel" (ventilation off)

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

<u>Gruppen</u>	Bezeichnung	<u>Status</u>	LED-Status	Funktion
	Rauchmelder	OK	grün	
Oben	LED verschmutzt oder Alarm	verschmutzt	gelb	RM verschmutzt
		Rauch-Alarm	rot	RM Rauch-Alarm
LED	Rauchmelder	OK	Aus	ОК
Mitte	LED Störung	Störung elektrisch	gelb	RM gestört
LED Unten	Luftmangel	kein Luftmangel	grün	Luftstrom OK
	(ventilation off)	Luftmangel	Aus	Luftmangel

Diese Zustände werden an den jeweiligen LEDs zur Anzeige gebracht:

Abb. LEDs je Gruppe



Technische Daten

Versorgungsspannung	24 VDC/AC, ± 10%				
Stromaufnahme	max. 100	max. 100 mA			
Prozessor Typ Taktfrequenz	Microchip 32 MHz	Microchip PIC32 32 MHz			
Bus-Schnittstelle Anschluss Bus Abschluss	RS485 (El über 120 (A 485) Dhm Widerstand			
Speicher	Flash 256 k RAM 256 k				
Protokoll	BACnet M	S/TP			
Digitale Ein-/Ausgänge	extern übe	er BACnet MS/TP-BUS			
Umgebungsbedingungen Betriebstemperatur Transport- und	050°C				
Lagertemperatur Relative Feuchte	-35+70° 595%, n	C icht kondensierend			
Schutzart	IP 40				
Abmessungen 19"-Geräte	Breite 8 TI Gehäuse I	E, Höhe 3 HE, Einbautiefe < 40 mm 3 x H x T = 130 x 40 x 35 mm			
Anschlussadapter	Breite 4 TI Gehäuse I	E, Höhe 3 HE, Einbautiefe < 50 mm 3 x H x T = 130 x 20 x 45 mm			
CE-Konformität					
	EN 61000 ESD	-4-2 / IEC 801-2 Elektrostatische Entladung			
	Kontakten FN 61000	tladung 8 kV / Luftentladung 8 kV -4-5 / IEC 801-5 Surge-Prüfung			
	Versorgun	gspg. AC 4 kV, DC 0,5 kV			
	EN 61000	ngen ∠ к∨ -4-4 / IEC 801-4 Burst-Prüfung			



romutec vorhandene BACnet Objekte und Property

BACnet Objekte nach ANSI/ASHRAE Standard 135-2012

Gerät/Device

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

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

Analogeingang / Analog Input

Analogausgang / Analog Output

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

Binary Output	Object_Identifier	R	
Binary Output	Object_Name	R/W	
Binary Output	Object_Type	R	
Binary Output	Present_Value	R	
Binary Output	Description	W/R	
Binary Output	Status_Flags	R	
Binary Output	Event_State	R	
Binary Output	Reliability	R	
Binary Output	Out_Of_Service	R	
Binary Output	Polarity	R/W	
Binary Output	Inactive_Text	R/W	
Binary Output	Active_Text	R/W	
Binary Output	Priority_Array	R	
Binary Output	Relinguish Default	R	

Digitalausgang / Binary Input

Digitaleingang / Binary Input

		_	
Binary Input	Object_identifier	R	
Binary Input	Object_Name	R/W	
Binary Input	Object_Type	R	
Binary Input	Present_Value	R	
Binary Input	Description	W/R	64 Zeichen
Binary Input	Device_Type	R/W	Konfiguration
Binary Input	Status_Flags	R	
Binary Input	Event_State	R	
Binary Input	Reliability	R	
Binary Input	Out_Of_Service	W	
Binary Input	Polarity	R/W	
Binary Input	Inactive_Text	W/R	
Binary Input	Active_Text	W/R	



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

Digital Wert / Binary Value

Mehrstufiger Wert - Multistate Value

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

R = read; W = write

Structured View - strukturierte Sichten auf Objekte zu definieren

Structured View	Object_Identifier	R	
Structured View	Object_Name	R	
Structured View	Object_Type	R	
Structured View	Description	0	
Structured View	Node_Type	R	
Structured View	Node_Subtype	0	
Structured View	Subordinate_List	R/W	
Structured View	Subordinate_Annotations	0	
Structured View	Profile_Name	0	



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 aufgeschalten 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)