Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
1		**system data**	bTriggerScada2PLC	Heart beat from BMS to the local PLC; auto returns to FALSE	W			bool	TRUE = on	32768	0	(0x03; 0x06)
2		**system data**	bResetErrors	reset all errors	W			bool	TRUE = reset	32768	1	(0x03; 0x06)
3		**system data**	bLight	switch on internal maintenance lighting	W			bool	TRUE = on	32768	2	(0x03; 0x06)
4		**system data**	bTestFireDampers	starts test of all fire dampers	W			bool	TRUE = start	32768	3	(0x03; 0x06)
5	**D**	**system data**	eSystemMode	AHU operation mode	W			uint	0 = off; 1 = manual mode; 2 = auto mode	32769		(0x03; 0x06)
6	**S**	**system data**	fTempOutdoor	send present value outdoor temperature to local PLC	W	°C	10	int		32770		(0x03; 0x06)
7	**D**	**system data**	nResetErrors	reset all errors, auto returns to 0	W			int	1 = reset	32771		(0x03; 0x06)
8		**setpoints**	bStandbySetpoint	force AHU to standby mode	W			bool	TRUE = standby	32791	0	(0x03; 0x06)
9	**D**	**setpoints**	fFanSUPSetpoint	setpoint of the supply air fan, unit depends on control strategy	W	Pa, m³/h, ppm		uint		32792		(0x03; 0x06)
10	**D**	**setpoints**	fFanETASetpoint	setpoint of the extract air fan, unit depends on control strategy	W	Pa, m³/h, ppm		uint		32793		(0x03; 0x06)
11	**D**	**setpoints**	fTempMinSetpoint	setpoint of the min. air temperature	W	°C	10	int		32794		(0x03; 0x06)
12	**D**	**setpoints**	fTempMaxSetpoint	setpoint of the max. air temperature	W	°C	10	int		32795		(0x03; 0x06)
13	**S**	**setpoints**	fHumMinSetpoint	setpoint of the min. air humidity	W	%	10	uint		32796		(0x03; 0x06)
14	**S**	**setpoints**	fHumMaxSetpoint	setpoint of the max. air humidity	W	%	10	uint		32797		(0x03; 0x06)
15		**setpoints**	rInputPowerDemandIC	power demand 0…100% of the integrated refrigerating (standalone only)	W	%		uint		32798		(0x03; 0x06)
16		**setpoints**	rInputPowerDemandRac	power demand 0…100% of the run around coil system (standalone only)	W	%		uint		32799		(0x03; 0x06)
17		**setpoints**	rSupplyAirFlowRac	supply air volume flow to the run around coil system (standalone only)	W	m³/h		uint		32800		(0x03; 0x06)
18	**S**	**settings**	fSetTempSUPMin	setpoint of the min. supply air temperature	W	°C	10	int		32808		(0x03; 0x06)
19	**S**	**settings**	fSetTempSUPMax	setpoint of the max. supply air temperature	W	°C	10	int		32809		(0x03; 0x06)
20	**S**	**settings**	fSetHumSUPMin	setpoint of the min. supply air humidity	W	%rH	10	int		32810		(0x03; 0x06)
21	**S**	**settings**	fSetHumSUPMax	setpoint of the max. supply air humidity	W	%rH	10	int		32811		(0x03; 0x06)
22		**system data**	bTriggerPLC2Scada	heart beat from local PLC to the BMS	R			bool	TRUE = on	32768	0	(0x04)
23		**system data**	bLight	internal mainenance lighting	R			bool	TRUE = on	32768	1	(0x04)
24		**system data**	bExternalLock	ext. lock of the AHU	R			bool	TRUE = ok	32768	2	(0x04)
25	**S**	**system data**	bVoltageError	voltage error	R			bool	TRUE = ok	32768	3	(0x04)
26	**S**	**system data**	bMainFuse	main fuse error	R			bool	TRUE = ok	32768	4	(0x04)
27	**S**	**system data**	bFireAlarm	error triggered fire alarm	R			bool	TRUE = ok	32768	5	(0x04)
28		**system data**	bModbuslineError	error modbus line	R			bool	TRUE = ok	32768	6	(0x04)
29	**S**	**system data**	bFrostProctection	error triggered frost protection	R			bool	TRUE = ok	32768	7	(0x04)
30	**D**	**system data**	eEventNotification	notification of alarm class	R			uint	0 = no alarm; 1 = warning (B-alarm); 2 = critical (A-alarm)	32769		(0x04)
31	**S**	**system data**	fTempOutdoor	present value outdoor air temperature	R	°C	0.1	int		32770		(0x04)
32		**system data**	bSmokeDetector1	triggered smoke detector 1	R			bool	TRUE = ok	32771	0	(0x04)
33		**system data**	bSmokeDetector2	triggered smoke detector 2	R			bool	TRUE = ok	32771	1	(0x04)
34		**system data**	bSmokeDetector3	triggered smoke detector 3	R			bool	TRUE = ok	32771	2	(0x04)
35		**system data**	bSmokeDetector4	triggered smoke detector 4	R			bool	TRUE = ok	32771	3	(0x04)
36		**system data**	bSmokeDetector5	triggered smoke detector 5	R			bool	TRUE = ok	32771	4	(0x04)
37		**system data**	bSmokeDetectorDirty1	smoke detector 1 is dirty	R			bool	TRUE = ok	32771	5	(0x04)
38		**system data**	bSmokeDetectorDirty2	smoke detector 2 is dirty	R			bool	TRUE = ok	32771	6	(0x04)
39		**system data**	bSmokeDetectorDirty3	smoke detector 3 is dirty	R			bool	TRUE = ok	32771	7	(0x04)
40		**system data**	bSmokeDetectorDirty4	smoke detector 4 is dirty	R			bool	TRUE = ok	32771	8	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
41		**system data**	bSmokeDetectorDirty5	smoke detector 5 is dirty	R			bool	TRUE = ok	32771	9	(0x04)
42		**system data**	extAlarm1	State of the ext. alarm number 1, polarity depends on configuration	R			bool		32783	0	(0x04)
43		**system data**	extAlarm2	State of the ext. alarm number 2, polarity depends on configuration	R			bool		32783	1	(0x04)
44		**system data**	extAlarm3	State of the ext. alarm number 3, polarity depends on configuration	R			bool		32783	2	(0x04)
45		**system data**	extAlarm4	State of the ext. alarm number 4, polarity depends on configuration	R			bool		32783	3	(0x04)
46		**system data**	extAlarm5	State of the ext. alarm number 5, polarity depends on configuration	R			bool		32783	4	(0x04)
47		**system data**	extAlarm6	State of the ext. alarm number 6, polarity depends on configuration	R			bool		32783	5	(0x04)
48		**system data**	extAlarm7	State of the ext. alarm number 7, polarity depends on configuration	R			bool		32783	6	(0x04)
49		**system data**	extAlarm8	State of the ext. alarm number 8, polarity depends on configuration	R			bool		32783	7	(0x04)
50		**system data**	extAlarm9	State of the ext. alarm number 9, polarity depends on configuration	R			bool		32783	8	(0x04)
51		**system data**	extAlarm10	State of the ext. alarm number 10, polarity depends on configuration	R			bool		32783	9	(0x04)
52		**measurement data**	fHumRoom1	present value room humditiy 1	R	%rH	0.1	int		32785		(0x04)
53		**measurement data**	fHumRoom2	present value room humditiy 2	R	%rH	0.1	int		32786		(0x04)
54		**measurement data**	fHumRoom3	present value room humditiy 3	R	%rH	0.1	int		32787		(0x04)
55		**measurement data**	fHumRoom4	present value room humditiy 4	R	%rH	0.1	int		32788		(0x04)
56		**measurement data**	fHumRoom5	present value room humditiy 5	R	%rH	0.1	int		32789		(0x04)
57		**measurement data**	fHumRoomMean	mean value of all room humidities	R	%rH	0.1	int		32790		(0x04)
58	**S**	**measurement data**	fTempODA	present value outdoor air temperature	R	°C	0.1	int		32791		(0x04)
59	**S**	**measurement data**	fTempSUP	present value supply air temperature	R	°C	0.1	int		32792		(0x04)
60	**S**	**measurement data**	fTempETA	present value extracted air temperature	R	°C	0.1	int		32793		(0x04)
61	**S**	**measurement data**	fTempEHA	present value exhauts air temperature	R	°C	0.1	int		32794		(0x04)
62	**S**	**measurement data**	fHumODA	present value outdoor air humidity	R	%rH	0.1	uint		32795		(0x04)
63	**S**	**measurement data**	fHumSUP	present value supply air humidity	R	%rH	0.1	uint		32796		(0x04)
64	**S**	**measurement data**	fHumETA	present value extracted air humidity	R	%rH	0.1	uint		32797		(0x04)
65	**S**	**measurement data**	fHumEHA	present value exhauts air humidity	R	%rH	0.1	uint		32798		(0x04)
66	**S**	**measurement data**	fPressureSUP	present value supply duct pressure	R	Pa		uint		32799		(0x04)
67	**S**	**measurement data**	fPressureETA	present value exhaust duct pressure	R	Pa		uint		32800		(0x04)
68	**S**	**measurement data**	fVOC	present value voc concentration	R	ppm		uint		32801		(0x04)
69	**S**	**measurement data**	fCO2	present value CO2 concentration	R	ppm		uint		32802		(0x04)
70		**measurement data**	fTempRoom1	present value room temperature 1	R	°C	0.1	int		32803		(0x04)
71		**measurement data**	fTempRoom2	present value room temperature 2	R	°C	0.1	int		32804		(0x04)
72		**measurement data**	fTempRoom3	present value room temperature 3	R	°C	0.1	int		32805		(0x04)
73		**measurement data**	fTempRoom4	present value room temperature 4	R	°C	0.1	int		32806		(0x04)
74		**measurement data**	fTempRoom5	present value room temperature 5	R	°C	0.1	int		32807		(0x04)
75		**measurement data**	fTempRoom6	present value room temperature 6	R	°C	0.1	int		32808		(0x04)
76		**measurement data**	fTempRoom7	present value room temperature 7	R	°C	0.1	int		32809		(0x04)
77		**measurement data**	fTempRoom8	present value room temperature 8	R	°C	0.1	int		32810		(0x04)
78		**measurement data**	fTempRoomMean	mean value of all room temperatures	R	°C	0.1	int		32812		(0x04)
79	**S**	**cooler**	bCoolStateMotorProtection	error motor protection cooler pump	R			bool	TRUE = ok	32813	0	(0x04)
80		**cooler**	bCoolStateErrModbusValve	modbus comm. error with the cooler valve	R			bool	TRUE = ok	32813	1	(0x04)
81	**S**	**cooler**	bCoolCtrlPump	controlled value to switch on/off the cooler pump	R			bool	TRUE = on	32813	2	(0x04)
82	**S**	**cooler**	fCoolStateValve	current position of the cooler valve	R	%		uint		32814		(0x04)
83	**S**	**cooler**	fCoolMeaInletTemp	present value of the cooler inlet temperature	R	°C	0.1	int		32815		(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
84		**cooler**	fCoolCtrlValve	controlled value of the cooler valve	R	%		uint		32816		(0x04)
85	**S**	**preheater**	bPreHeatStateMotorProtection	error motor protection preheater pump	R			bool	TRUE = ok	32822	0	(0x04)
86		**preheater**	bPreHeatStateErrModbusValve	modbus comm. error with the preheater valve	R			bool	TRUE = ok	32822	1	(0x04)
87	**S**	**preheater**	bPreHeatCtrlPump	controlled value to switch on/off the preheater pump	R			bool	TRUE = on	32822	2	(0x04)
88	**S**	**preheater**	fPreHeatStateValve	current position of the preheater valve	R	%		uint		32823		(0x04)
89	**S**	**preheater**	fPreHeatMeaReturnTemp	present value of the preheater outlet temperature	R	°C	0.1	int		32824		(0x04)
90		**preheater**	fPreHeatCtrlValve	controlled value of the preheater valve	R	%		uint		32825		(0x04)
91	**S**	**reheater**	bReHeatStateMotorProtection	error motor protection reheater pump	R			bool	TRUE = ok	32831	0	(0x04)
92		**reheater**	bReHeatStateErrModbusValve	modbus comm. error with the reheater valve	R			bool	TRUE = ok	32831	1	(0x04)
93	**S**	**reheater**	bReHeatCtrlPump	controlled value to switch on/off the reheater pump	R			bool	TRUE = on	32831	2	(0x04)
94	**S**	**reheater**	fReHeatStateValve	current position of the reheater valve	R	%		uint		32832		(0x04)
95	**S**	**reheater**	fReHeatMeaReturnTemp	present value of the reheater outlet temperature	R	°C	0.1	int		32833		(0x04)
96		**reheater**	fReHeatCtrlValve	controlled value of the reheater valve	R	%		uint		32834		(0x04)
97		**damper**	bDamperStateErrModbusODA	modbus comm. error with the outdoor air damper	R			bool	TRUE = ok	32840	0	(0x04)
98		**damper**	bDamperStateErrModbusSUP	modbus comm. error with the supply air damper	R			bool	TRUE = ok	32840	1	(0x04)
99		**damper**	bDamperStateErrModbusETA	modbus comm. error with the extracted air damper	R			bool	TRUE = ok	32840	2	(0x04)
100		**damper**	bDamperStateErrModbusEHA	modbus comm. error with the exhaust air damper	R			bool	TRUE = ok	32840	3	(0x04)
101		**damper**	bDamperStateErrModbusRCA	modbus comm. error with the recovery air damper	R			bool	TRUE = ok	32840	4	(0x04)
102		**damper**	bDamperStateErrModbusODA2	modbus comm. error with the second outdoor air damper	R			bool	TRUE = ok	32840	5	(0x04)
103		**damper**	bDamperStateErrModbusSUP2	modbus comm. error with the second supply air damper	R			bool	TRUE = ok	32840	6	(0x04)
104		**damper**	bDamperStateErrModbusETA2	modbus comm. error with the second extracted air damper	R			bool	TRUE = ok	32840	7	(0x04)
105		**damper**	bDamperStateErrModbusEHA2	modbus comm. error with the second exhaust air damper	R			bool	TRUE = ok	32840	8	(0x04)
106		**damper**	bDamperStateErrModbusRCA2	modbus comm. error with the second recovery air dampe	R			bool	TRUE = ok	32840	9	(0x04)
107		**damper**	bDamperStateErrModbusFanSUP	modbus comm. error with the supply air fan damper	R			bool	TRUE = ok	32840	10	(0x04)
108		**damper**	bDamperStateErrModbusFanETA	modbus comm. error with the extracted air fan damper	R			bool	TRUE = ok	32840	11	(0x04)
109		**damper**	bDamperStateErrModbusFanSUP2	modbus comm. error with the second supply air fan damper	R			bool	TRUE = ok	32840	12	(0x04)
110		**damper**	bDamperStateErrModbusFanETA2	modbus comm. error with the second extracted air fan damper	R			bool	TRUE = ok	32840	13	(0x04)
111	**S**	**damper**	fDamperStateODA	current position of the outdoor air damper	R	%		uint		32842		(0x04)
112	**S**	**damper**	fDamperStateSUP	current position of the supply air damper	R	%		uint		32843		(0x04)
113	**S**	**damper**	fDamperStateETA	current position of the extract air damper	R	%		uint		32844		(0x04)
114	**S**	**damper**	fDamperStateEHA	current position of the exhaust air damper	R	%		uint		32845		(0x04)
115	**S**	**damper**	fDamperStateRCA	current position of the recovery air damper	R	%		uint		32846		(0x04)
116		**damper**	fDamperStateODA2	current position of the second outdoor air damper	R	%		uint		32847		(0x04)
117		**damper**	fDamperStateSUP2	current position of the second supply air damper	R	%		uint		32848		(0x04)
118		**damper**	fDamperStateETA2	current position of the second extract air damper	R	%		uint		32849		(0x04)
119		**damper**	fDamperStateEHA2	current position of the second exhaust air damper	R	%		uint		32850		(0x04)
120		**damper**	fDamperStateRCA2	current position of the second recovery air damper	R	%		uint		32851		(0x04)
121		**damper**	fDamperStateFanSUP	current position of the supply air fan damper	R	%		uint		32852		(0x04)
122		**damper**	fDamperStateFanETA	current position of the extract air fan damper	R	%		uint		32853		(0x04)
123		**damper**	fDamperStateFanSUP2	current position of the second supply air fan damper	R	%		uint		32854		(0x04)
124		**damper**	fDamperStateFanETA2	current position of the second extract air fan damper	R	%		uint		32855		(0x04)
125	**S**	**damper**	fDamperCtrlODA	controlled value of the outdoor air damper position	R	%		uint		32856		(0x04)
126	**S**	**damper**	fDamperCtrlSUP	controlled value of the supply air damper position	R	%		uint		32857		(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
127	S	damper	fDamperCtrlETA	controlled value of the extract air damper position	R	%		uint		32858		(0x04)
128	S	damper	fDamperCtrlEHA	controlled value of the exhaust air damper position	R	%		uint		32859		(0x04)
129	S	damper	fDamperCtrlRCA	controlled value of the recovery air damper position	R	%		uint		32860		(0x04)
130		damper	fDamperCtrlODA2	controlled value of the second outdoor air damper position	R	%		uint		32861		(0x04)
131		damper	fDamperCtrlSUP2	controlled value of the second supply air damper position	R	%		uint		32862		(0x04)
132		damper	fDamperCtrlETA2	controlled value of the second extract air damper position	R	%		uint		32863		(0x04)
133		damper	fDamperCtrlEHA2	controlled value of the second exhaust air damper position	R	%		uint		32864		(0x04)
134		damper	fDamperCtrlRCA2	controlled value of the second recovery air damper position	R	%		uint		32865		(0x04)
135		damper	fDamperCtrlFanSUP	controlled value of the supply air fan damper position	R	%		uint		32866		(0x04)
136		damper	fDamperCtrlFanETA	controlled value of the extract air fan damper position	R	%		uint		32867		(0x04)
137		damper	fDamperCtrlFanSUP2	controlled value of the second supply air fan damper position	R	%		uint		32868		(0x04)
138		damper	fDamperCtrlFanETA2	controlled value of the second extract air fan damper position	R	%		uint		32869		(0x04)
139		supply air fan	bFanStateMotorProtectionSUP	motor protection of the supply air fan is inactive	R			bool	TRUE = ok	32871	0	(0x04)
140		supply air fan	bFanStateRepairSwitchSUP	repair switch of the supply air fan is active	R			bool	TRUE = ok	32871	1	(0x04)
141	S	supply air fan	bFanStateErrorSUP	internal error of the supply air fan	R			bool	TRUE = ok	32871	2	(0x04)
142		supply air fan	bFanStateErrModbusFuSUP	modbus comm. error with the supply air fan	R			bool	TRUE = ok	32871	3	(0x04)
143		supply air fan	bFanStateErrModbusDpSUP	modbus comm. error with the pressure transmitter of the supply air fan	R			bool	TRUE = ok	32871	4	(0x04)
144	S	supply air fan	bFanCtrlOperationSUP	controlled value to switch on/off the supply air fan	R			bool	TRUE = on	32871	5	(0x04)
145		supply air fan	bFanStateMotorProtectionSUP2	motor protection of the second supply air fan is inactive	R			bool	TRUE = ok	32871	6	(0x04)
146		supply air fan	bFanStateRepairSwitchSUP2	repair switch of the second supply air fan is active	R			bool	TRUE = ok	32871	7	(0x04)
147		supply air fan	bFanStateErrorSUP2	internal error of the second supply air fan	R			bool	TRUE = ok	32871	8	(0x04)
148		supply air fan	bFanStateErrModbusFuSUP2	modbus comm. error with the second supply air fan	R			bool	TRUE = ok	32871	9	(0x04)
149		supply air fan	bFanStateErrModbusDpSUP2	modbus comm. error with the pressure transmitter of the second supply air fan	R			bool	TRUE = ok	32871	10	(0x04)
150		supply air fan	bFanCtrlOperationSUP2	controlled value to switch on/off the second supply air fan	R			bool	TRUE = on	32871	11	(0x04)
151	S	supply air fan	fFanCtrlSpeedSUP	controlled value of the supply air fan speed [0..100%]	R	%		uint		32873		(0x04)
152	S	supply air fan	fFanMeaDpSUP	present value supply air fan diff. pressure	R	Pa		uint		32874		(0x04)
153	S	supply air fan	fFanMeaAirFlowSUP	present value supply airflow	R	m³/h		uint		32875		(0x04)
154		supply air fan	fFanCtrlSpeedSUP2	controlled value of the second supply air fan speed [0..100%]	R	%		uint		32876		(0x04)
155		supply air fan	fFanMeaDpSUP2	present value second supply air fan diff. pressure	R	Pa		uint		32877		(0x04)
156		supply air fan	fFanMeaAirFlowSUP2	present value second supply airflow	R	m³/h		uint		32878		(0x04)
157		extract air fan	bFanStateMotorProtectionETA	motor protection of the extract air fan is inactive	R			bool	TRUE = ok	32881	0	(0x04)
158		extract air fan	bFanStateRepairSwitchETA	repair switch of the extract air fan is active	R			bool	TRUE = ok	32881	1	(0x04)
159	S	extract air fan	bFanStateErrorETA	internal error of the extract air fan	R			bool	TRUE = ok	32881	2	(0x04)
160		extract air fan	bFanStateErrModbusFuETA	modbus comm. error with the extract air fan	R			bool	TRUE = ok	32881	3	(0x04)
161		extract air fan	bFanStateErrModbusDpETA	modbus comm. error with the pressure transmitter of the extract air fan	R			bool	TRUE = ok	32881	4	(0x04)
162	S	extract air fan	bFanCtrlOperationETA	controlled value to switch on/off the extract air fan	R			bool	TRUE = on	32881	5	(0x04)
163		extract air fan	bFanStateMotorProtectionETA2	motor protection of the second extract air fan is inactive	R			bool	TRUE = ok	32881	6	(0x04)
164		extract air fan	bFanStateRepairSwitchETA2	repair switch of the second extract air fan is active	R			bool	TRUE = ok	32881	7	(0x04)
165		extract air fan	bFanStateErrorETA2	internal error of the second extract air fan	R			bool	TRUE = ok	32881	8	(0x04)
166		extract air fan	bFanStateErrModbusFuETA2	modbus comm. error with the second extract air fan	R			bool	TRUE = ok	32881	9	(0x04)
167		extract air fan	bFanStateErrModbusDpETA2	modbus comm. error with the pressure transmitter of the second extract air fan	R			bool	TRUE = ok	32881	10	(0x04)
168		extract air fan	bFanCtrlOperationETA2	controlled value to switch on/off the second extract air fan	R			bool	TRUE = on	32881	11	(0x04)
169	S	extract air fan	fFanCtrlSpeedETA	controlled value of the extract air fan speed [0..100%]	R	%		uint		32883		(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
170	**S**	**extract air fan**	fFanMeaDpETA	present value extract air fan diff. pressure	R	Pa		uint		32884		(0x04)
171	**S**	**extract air fan**	fFanMeaAirFlowETA	present value extract airflow	R	m³/h		uint		32885		(0x04)
172		**extract air fan**	fFanCtrlSpeedETA2	controlled value of the second extract air fan speed [0..100%]	R	%		uint		32886		(0x04)
173		**extract air fan**	fFanMeaDpETA2	present value second extract air fan diff. pressure	R	Pa		uint		32887		(0x04)
174		**extract air fan**	fFanMeaAirFlowETA2	present value second extract airflow	R	m³/h		uint		32888		(0x04)
175	**S**	**filter**	bFilterStateWarningODA	error outdoor air filter change required	R			bool	TRUE = ok	32891	0	(0x04)
176	**S**	**filter**	bFilterStateWarningSUP	error supply air filter change required	R			bool	TRUE = ok	32891	1	(0x04)
177	**S**	**filter**	bFilterStateWarningETA	error extract air filter change required	R			bool	TRUE = ok	32891	2	(0x04)
178		**filter**	bFilterStateErrModbusDpODA	comm. error pressure sensor outdoor air filter	R			bool	TRUE = ok	32891	3	(0x04)
179		**filter**	bFilterStateErrModbusDpSUP	comm. error pressure sensor supply air filter	R			bool	TRUE = ok	32891	4	(0x04)
180		**filter**	bFilterStateErrModbusDpETA	comm. error pressure sensor extract air filter	R			bool	TRUE = ok	32891	5	(0x04)
181		**filter**	bFilterStateWarningODA2	error second outdoor air filter change required	R			bool	TRUE = ok	32891	6	(0x04)
182		**filter**	bFilterStateWarningSUP2	error second supply air filter change required	R			bool	TRUE = ok	32891	7	(0x04)
183		**filter**	bFilterStateWarningETA2	error second extract air filter change required	R			bool	TRUE = ok	32891	8	(0x04)
184		**filter**	bFilterStateErrModbusDpODA2	comm. error pressure sensor second outdoor air filter	R			bool	TRUE = ok	32891	9	(0x04)
185		**filter**	bFilterStateErrModbusDpSUP2	comm. error pressure sensor second supply air filter	R			bool	TRUE = ok	32891	10	(0x04)
186		**filter**	bFilterStateErrModbusDpETA2	comm. error pressure sensor second extract air filter	R			bool	TRUE = ok	32891	11	(0x04)
187	**S**	**filter**	uiFilterStateHoldingTimeODA	holding time outdoor air filter (in hour)	R	h		uint		32893		(0x04)
188	**S**	**filter**	uiFilterStateHoldingTimeSUP	holding time supply air filter (in hour)	R	h		uint		32894		(0x04)
189	**S**	**filter**	uiFilterStateHoldingTimeETA	holding time extract air filter (in hour)	R	h		uint		32895		(0x04)
190		**filter**	uiFilterStateOperatingTimeODA	operation time outdoor air filter (in hour)	R	h		uint		32896		(0x04)
191		**filter**	uiFilterStateOperatingTimeSUP	operation time supply air filter (in hour)	R	h		uint		32897		(0x04)
192		**filter**	uiFilterStateOperatingTimeETA	operation time extract air filter (in hour)	R	h		uint		32898		(0x04)
193	**S**	**filter**	fFilterMeaDpODA	present value outdoor air filter diff. pressure	R	Pa		uint		32899		(0x04)
194	**S**	**filter**	fFilterMeaDpSUP	present value supply air filter diff. pressure	R	Pa		uint		32900		(0x04)
195	**S**	**filter**	fFilterMeaDpETA	present value extract air filter diff. pressure	R	Pa		uint		32901		(0x04)
196		**filter**	uiFilterStateHoldingTimeODA2	holding time second outdoor air filter (in hour)	R	h		uint		32902		(0x04)
197		**filter**	uiFilterStateHoldingTimeSUP2	holding time second supply air filter (in hour)	R	h		uint		32903		(0x04)
198		**filter**	uiFilterStateHoldingTimeETA2	holding time second extract air filter (in hour)	R	h		uint		32904		(0x04)
199		**filter**	uiFilterStateOperatingTimeODA2	operation time second outdoor air filter (in hour)	R	h		uint		32905		(0x04)
200		**filter**	uiFilterStateOperatingTimeSUP2	operation time second supply air filter (in hour)	R	h		uint		32906		(0x04)
201		**filter**	uiFilterStateOperatingTimeETA2	operation time second extract air filter (in hour)	R	h		uint		32907		(0x04)
202		**filter**	fFilterMeaDpODA2	present value second outdoor air filter diff. pressure	R	Pa		uint		32908		(0x04)
203		**filter**	fFilterMeaDpSUP2	present value second supply air filter diff. pressure	R	Pa		uint		32909		(0x04)
204		**filter**	fFilterMeaDpETA2	present value second extract air filter diff. pressure	R	Pa		uint		32910		(0x04)
205		**plate heat exchanger**	bPlaHexStateErrModbusBypass	modbus comm. error with the plate heat exchanger dampe	R			bool	TRUE = ok	32911	0	(0x04)
206		**plate heat exchanger**	bPlaHexStateErrModbusBypass2	modbus comm. error with the second plate heat exchanger dampe	R			bool	TRUE = ok	32911	1	(0x04)
207		**plate heat exchanger**	bPlaHexStateErrModbusDp	modbus comm. error with the plate heat exchanger diff. pressure sensor	R			bool	TRUE = ok	32911	2	(0x04)
208	**S**	**plate heat exchanger**	fPlaHexStateBypass	current position bypass plate heat exchanger	R	%		uint		32912		(0x04)
209		**plate heat exchanger**	fPlaHexStateBypass2	current position second bypass plate heat exchanger	R	%		uint		32913		(0x04)
210		***plate heat exchanger***	*fPlaHexMeaTemp*	*present value recovery air temperature*	*R*	*°C*	*0.1*	*int*		32914		(0x04)
211	**S**	**plate heat exchanger**	fPlaHexMeaDp	present value plate heat exchanger diff. pressure	R	%		uint		32915		(0x04)
212	**S**	**plate heat exchanger**	fPlaHexCtrlBypass	controlled value bypass plate heat exchanger	R	%		uint		32916		(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
213		**plate heat exchanger**	fPlaHexCtrlBypass2	controlled value of the second bypass plate heat exchanger	R	%		uint		32917		(0x04)
214		**rotary heat exchanger**	bRotHexStateRotationAlarm	internal error of the heat recovery wheel	R			bool	TRUE = ok	32922	0	(0x04)
215		**rotary heat exchanger**	bRotHexStateErrModbus	modbus comm. error with the heat recovery wheel	R			bool	TRUE = ok	32922	1	(0x04)
216	**S**	**rotary heat exchanger**	bRotHexCtrlOperation	controlled value to release the heat recovery wheel	R			bool	TRUE = ok	32922	2	(0x04)
217	**S**	**rotary heat exchanger**	fRotHexCtrlRPM	controlled value heat recovery wheel speed	R	%		uint		32923		(0x04)
218		**modbus comm. error**	bModErrHumODA	modbus comm. error with the outdoor air humidity sensor	R			bool	TRUE = ok	32929	0	(0x04)
219		**modbus comm. error**	bModErrHumSUP	modbus comm. error with the supply air humidity sensor	R			bool	TRUE = ok	32929	1	(0x04)
220		**modbus comm. error**	bModErrHumETA	modbus comm. error with the extract air humidity sensor	R			bool	TRUE = ok	32929	2	(0x04)
221		**modbus comm. error**	bModErrHumEHA	modbus comm. error with the exhaust air humidity sensor	R			bool	TRUE = ok	32929	3	(0x04)
222		**modbus comm. error**	bModErrTempODA	modbus comm. error with the outdoor air temperature sensor	R			bool	TRUE = ok	32929	4	(0x04)
223		**modbus comm. error**	bModErrTempSUP	modbus comm. error with the supply air temperature sensor	R			bool	TRUE = ok	32929	5	(0x04)
224		**modbus comm. error**	bModErrTempETA	modbus comm. error with the extract air temperature sensor	R			bool	TRUE = ok	32929	6	(0x04)
225		**modbus comm. error**	bModErrTempEHA	modbus comm. error with the exhaust air temperature sensor	R			bool	TRUE = ok	32929	7	(0x04)
226		**modbus comm. error**	bModErrDpSUP	modbus comm. error with the supply air pressure sensor	R			bool	TRUE = ok	32929	8	(0x04)
227		**modbus comm. error**	bModErrDpETA	modbus comm. error with the extract air pressure sensor	R			bool	TRUE = ok	32929	9	(0x04)
228		**modbus comm. error**	bModErrCO2	modbus comm. error with the CO2 sensor	R			bool	TRUE = ok	32929	10	(0x04)
229		**modbus comm. error**	bModErrVOC	modbus comm. error with the VOC sensor	R			bool	TRUE = ok	32929	11	(0x04)
230		***modbus comm. error***	*bModErrTempRoom1*	*modbus comm. error with the room temperature sensor 1*	*R*			*bool*	*TRUE = ok*	32930	0	(0x04)
231		***modbus comm. error***	*bModErrTempRoom2*	*modbus comm. error with the room temperature sensor 2*	*R*			*bool*	*TRUE = ok*	32930	1	(0x04)
232		***modbus comm. error***	*bModErrTempRoom3*	*modbus comm. error with the room temperature sensor 3*	*R*			*bool*	*TRUE = ok*	32930	2	(0x04)
233		***modbus comm. error***	*bModErrTempRoom4*	*modbus comm. error with the room temperature sensor 4*	*R*			*bool*	*TRUE = ok*	32930	3	(0x04)
234		***modbus comm. error***	*bModErrTempRoom5*	*modbus comm. error with the room temperature sensor 5*	*R*			*bool*	*TRUE = ok*	32930	4	(0x04)
235		***modbus comm. error***	*bModErrHumRoom1*	*modbus comm. error with the room humidity sensor 1*	*R*			*bool*	*TRUE = ok*	32930	5	(0x04)
236		***modbus comm. error***	*bModErrHumRoom2*	*modbus comm. error with the room humidity sensor 2*	*R*			*bool*	*TRUE = ok*	32930	6	(0x04)
237		***modbus comm. error***	*bModErrHumRoom3*	*modbus comm. error with the room humidity sensor 3*	*R*			*bool*	*TRUE = ok*	32930	7	(0x04)
238		***modbus comm. error***	*bModErrHumRoom4*	*modbus comm. error with the room humidity sensor 4*	*R*			*bool*	*TRUE = ok*	32930	8	(0x04)
239		***modbus comm. error***	*bModErrHumRoom5*	*modbus comm. error with the room humidity sensor 5*	*R*			*bool*	*TRUE = ok*	32930	9	(0x04)
240		**current operation mode**	fSetpointTempSUP	current set point of the supply air temperature	R	°C	0.1	int		32940		(0x04)
241		**current operation mode**	fSetpointTempETA	current set point of the extract or room air temperature	R	°C	0.1	int		32941		(0x04)
242	**S**	**current operation mode**	eOperationMode	current operation mode of the air handling unit.	R			uint	0 = off; 1 = standby; 2 = control; 3 = freeze protection; 4 = deicing; 5 = startup; 6 = shutdown; 7 = manual; 8 = nightpurge; 9 = intermittent Operation; 10 = cooling protection; 11 = fire protection	32942		(0x04)
243		***current operation mode***	*fFanPowerSUP1*	*electrical power supply fan 1*	*R*	*kW*	*0.01*	*uint*		32943		(0x04)
244		***current operation mode***	*fFanPowerSUP2*	*electrical power supply fan 2*	*R*	*kW*	*0.01*	*uint*		32944		(0x04)
245		***current operation mode***	*fFanPowerETA1*	*electrical power extract fan 1*	*R*	*kW*	*0.01*	*uint*		32945		(0x04)
246		***current operation mode***	*fFanPowerETA2*	*electrical power extract fan 2*	*R*	*kW*	*0.01*	*uint*		32946		(0x04)
247		***current operation mode***	*fFanSfpSUP1*	*specific fan power supply fan 1*	*R*	*Ws/m³*		*uint*		32947		(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
248		***current operation mode***	*fFanSfpETA1*	*specific fan power extract fan 1*	*R*	*Ws/m³*		*uint*		32949		(0x04)
249		**rac**	bReleasePump	pump release	R			bool	TRUE = on	32960	0	(0x04)
250		**rac**	bPumpError	pump error	R			bool	TRUE = alarm	32960	1	(0x04)
251		**rac**	bBrinePressure1	pressure step 1 triggered	R			bool	TRUE = alarm	32960	2	(0x04)
252		**rac**	bBrinePressure2	pressure step 2 triggered	R			bool	TRUE = alarm	32960	3	(0x04)
253		**rac**	bMsgMinTempInletETA	exhaust air heat exchanger is frosting (prio=2)	R			bool	TRUE = alarm	32960	4	(0x04)
254		***rac***	*bMsgPumpError*	*pump error (prio=3)*	*R*			*bool*	*TRUE = alarm*	32960	5	(0x04)
255		**rac**	bMsgFrostFeedCoil	alarm feed coil frosting (prio=3)	R			bool	TRUE = alarm	32960	6	(0x04)
256		**rac**	bMsgNoRecovery	heat recovery is currently not possible (prio=2)	R			bool	TRUE = alarm	32960	7	(0x04)
257		**rac**	bMsgNoFeed	feed doesnt have cooling or heating (prio=2)	R			bool	TRUE = alarm	32960	8	(0x04)
258		***rac***	*bMsgPumpMinVolumeFlow*	*min. pump volume flow (prio=3)*	*R*			*bool*	*TRUE = alarm*	32960	9	(0x04)
259		**rac**	bMsgBrinePressureLow	brine pressure have to be checked, low pressure (prio=2)	R			bool	TRUE = alarm	32960	10	(0x04)
260		**rac**	bMsgBrinePressureCritical	critical brine pressure (prio=3)	R			bool	TRUE = alarm	32960	11	(0x04)
261		**rac**	bMsgAutoPumpOff	auto pump stop (prio=2)	R			bool	TRUE = alarm	32960	12	(0x04)
262		**rac**	bStateHeatExchangerOperation	state of the operation signal be released	R			bool	TRUE = on	32960	13	(0x04)
263		**rac**	bStateFastMode	state of the fast cool- or heating mode active	R			bool	TRUE = on	32960	14	(0x04)
264		**rac**	bStateCoolingFeed	state of the cooling feed feeding is active	R			bool	TRUE = on	32960	15	(0x04)
265		**rac**	bStateHeatingFeed	state of the heating feed feeding is active	R			bool	TRUE = on	32961	0	(0x04)
266		***rac***	*bMsgPumpWarning*	*pump error outputted a warning signal (prio=2)*	*R*			*bool*	*TRUE = alarm*	32961	1	(0x04)
267		**rac**	bActuatingValueFeedCoolingPump	pump release cooling feed	R			bool	TRUE = on	32961	2	(0x04)
268		**rac**	bActuatingValueFeedHeatingPump	pump release heating feed	R			bool	TRUE = on	32961	3	(0x04)
269		**rac**	bReleaseColdProvider	release cold provider	R			bool	TRUE = on	32961	4	(0x04)
270		**rac**	bReleaseHeatProvider	release heat provider	R			bool	TRUE = on	32961	5	(0x04)
271		**rac**	rActuatingValuePump	controlled value pump speed	R	%		uint		32962		(0x04)
272		**rac**	rActuatingValuePowerValve	controlled value run around coil power valve	R	%		uint		32963		(0x04)
273		**rac**	rActuatingValueFrostProtectionValve	controlled value run around coil frost protection valve	R	%		uint		32964		(0x04)
274		**rac**	rTempSUPIn	current value inlet temperature of the fresh air heat exchanger	R	°C	0.1	int		32965		(0x04)
275		**rac**	rTempSUPOut	current value outlet temperature of the fresh air heat exchanger	R	°C	0.1	int		32966		(0x04)
276		**rac**	rTempETAIn	current value inlet temperature of exhaust air heat exchanger	R	°C	0.1	int		32967		(0x04)
277		**rac**	rTempETAOut	current value return temperature exhaust air heat exchanger	R	°C	0.1	int		32968		(0x04)
278		**rac**	rTempPreFeed	current value brine temperature	R	°C	0.1	int		32969		(0x04)
279		**rac**	rBrineVolumeFlow	present value brine volume flow	R	m³/h	0.01	uint		32970		(0x04)
280		**rac**	rThermalPowerSUP	current value thermal power of the fresh air heat exchanger	R	kW		int		32971		(0x04)
281		**rac**	rThermalPowerETA	current value thermal power of the exhaust air heat exchanger	R	kW		int		32972		(0x04)
282		**rac**	rThermalPowerFeedHeat	current value thermal power of the heating feed	R	kW		int		32973		(0x04)
283		**rac**	rThermalPowerFeedCool	current value thermal power of the cooling feed	R	kW		int		32974		(0x04)
284		**rac**	rActuatingValueFeedCoolingValve	controlled value run around coil cooling feed valve	R	%		uint		32975		(0x04)
285		**rac**	rActuatingValueFeedHeatingValve	controlled value run around coil heating feed valve	R	%		uint		32976		(0x04)
286		**rac**	rTempCoolFeedIn	current value inlet temperature of cooling feed	R	°C	0.1	int		32977		(0x04)
287		**rac**	rTempHeatFeedOut	current value return temperature of heating feed	R	°C	0.1	int		32978		(0x04)
288		**rac**	rActuatingValueDehumCoolValve	controlled value run around coil dehumdifier cooling valve	R	%		uint		32979		(0x04)
289		**rac**	rActuatingValueDehumReaheater	controlled value run around coil dehumdifier cooling recovery valve	R	%		uint		32980		(0x04)
290		**integrated refrigerating**	bMsgErrorSuperHeatingController	error super heating controller	R			bool	TRUE = alarm	32985	0	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
291		**integrated refrigerating**	bMsgOilManagement	oil management error	R			bool	TRUE = alarm	32985	1	(0x04)
292		**integrated refrigerating**	bMsgHighPressure	high pressure error	R			bool	TRUE = alarm	32985	2	(0x04)
293		**integrated refrigerating**	bMsgLowPressure	low pressure error	R			bool	TRUE = alarm	32985	3	(0x04)
294		**integrated refrigerating**	bCompressorPWM [1]	actuating value pwm compressor 1	R			bool	TRUE = on	32985	4	(0x04)
295		**integrated refrigerating**	bCompressorPWM [2]	actuating value pwm compressor 2	R			bool	TRUE = on	32985	5	(0x04)
296		**integrated refrigerating**	arCompMotorProtection [1]	motor protection triggered first compressor	R			bool	TRUE = alarm	32985	7	(0x04)
297		**integrated refrigerating**	arCompMotorProtection [2]	motor protection triggered second compressor	R			bool	TRUE = alarm	32985	8	(0x04)
298		**integrated refrigerating**	arSoftStarterOn [1]	actuating value soft starter compressor 1	R			bool	TRUE = on	32985	10	(0x04)
299		**integrated refrigerating**	arSoftStarterOn [2]	actuating value soft starter compressor 2	R			bool	TRUE = on	32985	11	(0x04)
300		**integrated refrigerating**	bMsgHighTemp	error - high compressor end temperature	R			bool	TRUE = alarm	32985	13	(0x04)
301		**integrated refrigerating**	bEAPComError	comm. error with the EAP-Modul	R			bool	TRUE = alarm	32985	14	(0x04)
302	***S***	***fire damper***	*bCriticalSmokeDetectorOrFireDampe*	*collective fault of smoke detectors and fire dampers*	*R*			*bool*	*TRUE = alarm*	32990	0	(0x04)
303	***S***	***fire damper***	*bFireDampersOK*	*all fire dampers are ok*	*R*			*bool*	*TRUE = ok*	32990	1	(0x04)
304	***S***	***fire damper***	*bSmokeDetectorsOK*	*all smoke detectors are ok*	*R*			*bool*	*TRUE = ok*	32990	2	(0x04)
305		**fire damper**	bFireDamperClosed1	contact fire damper 1 closed	R			bool	TRUE = closed	32991	0	(0x04)
306		**fire damper**	bFireDamperClosed2	contact fire damper 2 closed	R			bool	TRUE = closed	32991	1	(0x04)
307		**fire damper**	bFireDamperClosed3	contact fire damper 3 closed	R			bool	TRUE = closed	32991	2	(0x04)
308		**fire damper**	bFireDamperClosed4	contact fire damper 4 closed	R			bool	TRUE = closed	32991	3	(0x04)
309		**fire damper**	bFireDamperClosed5	contact fire damper 5 closed	R			bool	TRUE = closed	32991	4	(0x04)
310		**fire damper**	bFireDamperClosed6	contact fire damper 6 closed	R			bool	TRUE = closed	32991	5	(0x04)
311		**fire damper**	bFireDamperClosed7	contact fire damper 7 closed	R			bool	TRUE = closed	32991	6	(0x04)
312		**fire damper**	bFireDamperClosed8	contact fire damper 8 closed	R			bool	TRUE = closed	32991	7	(0x04)
313		**fire damper**	bFireDamperClosed9	contact fire damper 9 closed	R			bool	TRUE = closed	32991	8	(0x04)
314		**fire damper**	bFireDamperClosed10	contact fire damper 10 closed	R			bool	TRUE = closed	32991	9	(0x04)
315		**fire damper**	bFireDamperClosed11	contact fire damper 11 closed	R			bool	TRUE = closed	32991	10	(0x04)
316		**fire damper**	bFireDamperClosed12	contact fire damper 12 closed	R			bool	TRUE = closed	32991	11	(0x04)
317		**fire damper**	bFireDamperClosed13	contact fire damper 13 closed	R			bool	TRUE = closed	32991	12	(0x04)
318		**fire damper**	bFireDamperClosed14	contact fire damper 14 closed	R			bool	TRUE = closed	32991	13	(0x04)
319		**fire damper**	bFireDamperClosed15	contact fire damper 15 closed	R			bool	TRUE = closed	32991	14	(0x04)
320		**fire damper**	bFireDamperClosed16	contact fire damper 16 closed	R			bool	TRUE = closed	32991	15	(0x04)
321		**fire damper**	bFireDamperClosed17	contact fire damper 17 closed	R			bool	TRUE = closed	32992	0	(0x04)
322		**fire damper**	bFireDamperClosed18	contact fire damper 18 closed	R			bool	TRUE = closed	32992	1	(0x04)
323		**fire damper**	bFireDamperClosed19	contact fire damper 19 closed	R			bool	TRUE = closed	32992	2	(0x04)
324		**fire damper**	bFireDamperClosed20	contact fire damper 20 closed	R			bool	TRUE = closed	32992	3	(0x04)
325		**fire damper**	bFireDamperClosed21	contact fire damper 21 closed	R			bool	TRUE = closed	32992	4	(0x04)
326		**fire damper**	bFireDamperClosed22	contact fire damper 22 closed	R			bool	TRUE = closed	32992	5	(0x04)
327		**fire damper**	bFireDamperClosed23	contact fire damper 23 closed	R			bool	TRUE = closed	32992	6	(0x04)
328		**fire damper**	bFireDamperClosed24	contact fire damper 24 closed	R			bool	TRUE = closed	32992	7	(0x04)
329		**fire damper**	bFireDamperClosed25	contact fire damper 25 closed	R			bool	TRUE = closed	32992	8	(0x04)
330		**fire damper**	bFireDamperClosed26	contact fire damper 26 closed	R			bool	TRUE = closed	32992	9	(0x04)
331		**fire damper**	bFireDamperClosed27	contact fire damper 27 closed	R			bool	TRUE = closed	32992	10	(0x04)
332		**fire damper**	bFireDamperClosed28	contact fire damper 28 closed	R			bool	TRUE = closed	32992	11	(0x04)
333		**fire damper**	bFireDamperClosed29	contact fire damper 29 closed	R			bool	TRUE = closed	32992	12	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
334		**fire damper**	bFireDamperClosed30	contact fire damper 30 closed	R			bool	TRUE = closed	32992	13	(0x04)
335		**fire damper**	bFireDamperClosed31	contact fire damper 31 closed	R			bool	TRUE = closed	32992	14	(0x04)
336		**fire damper**	bFireDamperClosed32	contact fire damper 32 closed	R			bool	TRUE = closed	32992	15	(0x04)
337		**fire damper**	bFireDamperClosed33	contact fire damper 33 closed	R			bool	TRUE = closed	32993	0	(0x04)
338		**fire damper**	bFireDamperClosed34	contact fire damper 34 closed	R			bool	TRUE = closed	32993	1	(0x04)
339		**fire damper**	bFireDamperClosed35	contact fire damper 35 closed	R			bool	TRUE = closed	32993	2	(0x04)
340		**fire damper**	bFireDamperClosed36	contact fire damper 36 closed	R			bool	TRUE = closed	32993	3	(0x04)
341		**fire damper**	bFireDamperClosed37	contact fire damper 37 closed	R			bool	TRUE = closed	32993	4	(0x04)
342		**fire damper**	bFireDamperClosed38	contact fire damper 38 closed	R			bool	TRUE = closed	32993	5	(0x04)
343		**fire damper**	bFireDamperClosed39	contact fire damper 39 closed	R			bool	TRUE = closed	32993	6	(0x04)
344		**fire damper**	bFireDamperClosed40	contact fire damper 40 closed	R			bool	TRUE = closed	32993	7	(0x04)
345		**fire damper**	bFireDamperClosed41	contact fire damper 41 closed	R			bool	TRUE = closed	32993	8	(0x04)
346		**fire damper**	bFireDamperClosed42	contact fire damper 42 closed	R			bool	TRUE = closed	32993	9	(0x04)
347		**fire damper**	bFireDamperClosed43	contact fire damper 43 closed	R			bool	TRUE = closed	32993	10	(0x04)
348		**fire damper**	bFireDamperClosed44	contact fire damper 44 closed	R			bool	TRUE = closed	32993	11	(0x04)
349		**fire damper**	bFireDamperClosed45	contact fire damper 45 closed	R			bool	TRUE = closed	32993	12	(0x04)
350		**fire damper**	bFireDamperClosed46	contact fire damper 46 closed	R			bool	TRUE = closed	32993	13	(0x04)
351		**fire damper**	bFireDamperClosed47	contact fire damper 47 closed	R			bool	TRUE = closed	32993	14	(0x04)
352		**fire damper**	bFireDamperClosed48	contact fire damper 48 closed	R			bool	TRUE = closed	32993	15	(0x04)
353		**fire damper**	bFireDamperClosed49	contact fire damper 49 closed	R			bool	TRUE = closed	32994	0	(0x04)
354		**fire damper**	bFireDamperClosed50	contact fire damper 50 closed	R			bool	TRUE = closed	32994	1	(0x04)
355		**fire damper**	bFireDamperClosed51	contact fire damper 51 closed	R			bool	TRUE = closed	32994	2	(0x04)
356		**fire damper**	bFireDamperClosed52	contact fire damper 52 closed	R			bool	TRUE = closed	32994	3	(0x04)
357		**fire damper**	bFireDamperClosed53	contact fire damper 53 closed	R			bool	TRUE = closed	32994	4	(0x04)
358		**fire damper**	bFireDamperClosed54	contact fire damper 54 closed	R			bool	TRUE = closed	32994	5	(0x04)
359		**fire damper**	bFireDamperClosed55	contact fire damper 55 closed	R			bool	TRUE = closed	32994	6	(0x04)
360		**fire damper**	bFireDamperClosed56	contact fire damper 56 closed	R			bool	TRUE = closed	32994	7	(0x04)
361		**fire damper**	bFireDamperClosed57	contact fire damper 57 closed	R			bool	TRUE = closed	32994	8	(0x04)
362		**fire damper**	bFireDamperClosed58	contact fire damper 58 closed	R			bool	TRUE = closed	32994	9	(0x04)
363		**fire damper**	bFireDamperClosed59	contact fire damper 59 closed	R			bool	TRUE = closed	32994	10	(0x04)
364		**fire damper**	bFireDamperClosed60	contact fire damper 60 closed	R			bool	TRUE = closed	32994	11	(0x04)
365		**fire damper**	bFireDamperClosed61	contact fire damper 61 closed	R			bool	TRUE = closed	32994	12	(0x04)
366		**fire damper**	bFireDamperClosed62	contact fire damper 62 closed	R			bool	TRUE = closed	32994	13	(0x04)
367		**fire damper**	bFireDamperOpened1	contact fire damper 1 Opened	R			bool	TRUE = open	32995	0	(0x04)
368		**fire damper**	bFireDamperOpened2	contact fire damper 2 Opened	R			bool	TRUE = open	32995	1	(0x04)
369		**fire damper**	bFireDamperOpened3	contact fire damper 3 Opened	R			bool	TRUE = open	32995	2	(0x04)
370		**fire damper**	bFireDamperOpened4	contact fire damper 4 Opened	R			bool	TRUE = open	32995	3	(0x04)
371		**fire damper**	bFireDamperOpened5	contact fire damper 5 Opened	R			bool	TRUE = open	32995	4	(0x04)
372		**fire damper**	bFireDamperOpened6	contact fire damper 6 Opened	R			bool	TRUE = open	32995	5	(0x04)
373		**fire damper**	bFireDamperOpened7	contact fire damper 7 Opened	R			bool	TRUE = open	32995	6	(0x04)
374		**fire damper**	bFireDamperOpened8	contact fire damper 8 Opened	R			bool	TRUE = open	32995	7	(0x04)
375		**fire damper**	bFireDamperOpened9	contact fire damper 9 Opened	R			bool	TRUE = open	32995	8	(0x04)
376		**fire damper**	bFireDamperOpened10	contact fire damper 10 Opened	R			bool	TRUE = open	32995	9	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
377		**fire damper**	bFireDamperOpened11	contact fire damper 11 Opened	R			bool	TRUE = open	32995	10	(0x04)
378		**fire damper**	bFireDamperOpened12	contact fire damper 12 Opened	R			bool	TRUE = open	32995	11	(0x04)
379		**fire damper**	bFireDamperOpened13	contact fire damper 13 Opened	R			bool	TRUE = open	32995	12	(0x04)
380		**fire damper**	bFireDamperOpened14	contact fire damper 14 Opened	R			bool	TRUE = open	32995	13	(0x04)
381		**fire damper**	bFireDamperOpened15	contact fire damper 15 Opened	R			bool	TRUE = open	32995	14	(0x04)
382		**fire damper**	bFireDamperOpened16	contact fire damper 16 Opened	R			bool	TRUE = open	32995	15	(0x04)
383		**fire damper**	bFireDamperOpened17	contact fire damper 17 Opened	R			bool	TRUE = open	32996	0	(0x04)
384		**fire damper**	bFireDamperOpened18	contact fire damper 18 Opened	R			bool	TRUE = open	32996	1	(0x04)
385		**fire damper**	bFireDamperOpened19	contact fire damper 19 Opened	R			bool	TRUE = open	32996	2	(0x04)
386		**fire damper**	bFireDamperOpened20	contact fire damper 20 Opened	R			bool	TRUE = open	32996	3	(0x04)
387		**fire damper**	bFireDamperOpened21	contact fire damper 21 Opened	R			bool	TRUE = open	32996	4	(0x04)
388		**fire damper**	bFireDamperOpened22	contact fire damper 22 Opened	R			bool	TRUE = open	32996	5	(0x04)
389		**fire damper**	bFireDamperOpened23	contact fire damper 23 Opened	R			bool	TRUE = open	32996	6	(0x04)
390		**fire damper**	bFireDamperOpened24	contact fire damper 24 Opened	R			bool	TRUE = open	32996	7	(0x04)
391		**fire damper**	bFireDamperOpened25	contact fire damper 25 Opened	R			bool	TRUE = open	32996	8	(0x04)
392		**fire damper**	bFireDamperOpened26	contact fire damper 26 Opened	R			bool	TRUE = open	32996	9	(0x04)
393		**fire damper**	bFireDamperOpened27	contact fire damper 27 Opened	R			bool	TRUE = open	32996	10	(0x04)
394		**fire damper**	bFireDamperOpened28	contact fire damper 28 Opened	R			bool	TRUE = open	32996	11	(0x04)
395		**fire damper**	bFireDamperOpened29	contact fire damper 29 Opened	R			bool	TRUE = open	32996	12	(0x04)
396		**fire damper**	bFireDamperOpened30	contact fire damper 30 Opened	R			bool	TRUE = open	32996	13	(0x04)
397		**fire damper**	bFireDamperOpened31	contact fire damper 31 Opened	R			bool	TRUE = open	32996	14	(0x04)
398		**fire damper**	bFireDamperOpened32	contact fire damper 32 Opened	R			bool	TRUE = open	32996	15	(0x04)
399		**fire damper**	bFireDamperOpened33	contact fire damper 33 Opened	R			bool	TRUE = open	32997	0	(0x04)
400		**fire damper**	bFireDamperOpened34	contact fire damper 34 Opened	R			bool	TRUE = open	32997	1	(0x04)
401		**fire damper**	bFireDamperOpened35	contact fire damper 35 Opened	R			bool	TRUE = open	32997	2	(0x04)
402		**fire damper**	bFireDamperOpened36	contact fire damper 36 Opened	R			bool	TRUE = open	32997	3	(0x04)
403		**fire damper**	bFireDamperOpened37	contact fire damper 37 Opened	R			bool	TRUE = open	32997	4	(0x04)
404		**fire damper**	bFireDamperOpened38	contact fire damper 38 Opened	R			bool	TRUE = open	32997	5	(0x04)
405		**fire damper**	bFireDamperOpened39	contact fire damper 39 Opened	R			bool	TRUE = open	32997	6	(0x04)
406		**fire damper**	bFireDamperOpened40	contact fire damper 40 Opened	R			bool	TRUE = open	32997	7	(0x04)
407		**fire damper**	bFireDamperOpened41	contact fire damper 41 Opened	R			bool	TRUE = open	32997	8	(0x04)
408		**fire damper**	bFireDamperOpened42	contact fire damper 42 Opened	R			bool	TRUE = open	32997	9	(0x04)
409		**fire damper**	bFireDamperOpened43	contact fire damper 43 Opened	R			bool	TRUE = open	32997	10	(0x04)
410		**fire damper**	bFireDamperOpened44	contact fire damper 44 Opened	R			bool	TRUE = open	32997	11	(0x04)
411		**fire damper**	bFireDamperOpened45	contact fire damper 45 Opened	R			bool	TRUE = open	32997	12	(0x04)
412		**fire damper**	bFireDamperOpened46	contact fire damper 46 Opened	R			bool	TRUE = open	32997	13	(0x04)
413		**fire damper**	bFireDamperOpened47	contact fire damper 47 Opened	R			bool	TRUE = open	32997	14	(0x04)
414		**fire damper**	bFireDamperOpened48	contact fire damper 48 Opened	R			bool	TRUE = open	32997	15	(0x04)
415		**fire damper**	bFireDamperOpened49	contact fire damper 49 Opened	R			bool	TRUE = open	32998	0	(0x04)
416		**fire damper**	bFireDamperOpened50	contact fire damper 50 Opened	R			bool	TRUE = open	32998	1	(0x04)
417		**fire damper**	bFireDamperOpened51	contact fire damper 51 Opened	R			bool	TRUE = open	32998	2	(0x04)
418		**fire damper**	bFireDamperOpened52	contact fire damper 52 Opened	R			bool	TRUE = open	32998	3	(0x04)
419		**fire damper**	bFireDamperOpened53	contact fire damper 53 Opened	R			bool	TRUE = open	32998	4	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
420		**fire damper**	bFireDamperOpened54	contact fire damper 54 Opened	R			bool	TRUE = open	32998	5	(0x04)
421		**fire damper**	bFireDamperOpened55	contact fire damper 55 Opened	R			bool	TRUE = open	32998	6	(0x04)
422		**fire damper**	bFireDamperOpened56	contact fire damper 56 Opened	R			bool	TRUE = open	32998	7	(0x04)
423		**fire damper**	bFireDamperOpened57	contact fire damper 57 Opened	R			bool	TRUE = open	32998	8	(0x04)
424		**fire damper**	bFireDamperOpened58	contact fire damper 58 Opened	R			bool	TRUE = open	32998	9	(0x04)
425		**fire damper**	bFireDamperOpened59	contact fire damper 59 Opened	R			bool	TRUE = open	32998	10	(0x04)
426		**fire damper**	bFireDamperOpened60	contact fire damper 60 Opened	R			bool	TRUE = open	32998	11	(0x04)
427		**fire damper**	bFireDamperOpened61	contact fire damper 61 Opened	R			bool	TRUE = open	32998	12	(0x04)
428		**fire damper**	bFireDamperOpened62	contact fire damper 62 Opened	R			bool	TRUE = open	32998	13	(0x04)
429		**fire damper**	bFireDamperErrorClosingRuntime1	Error closing runtime fire damper 1	R			bool	TRUE = ok	32999	0	(0x04)
430		**fire damper**	bFireDamperErrorClosingRuntime2	Error closing runtime fire damper 2	R			bool	TRUE = ok	32999	1	(0x04)
431		**fire damper**	bFireDamperErrorClosingRuntime3	Error closing runtime fire damper 3	R			bool	TRUE = ok	32999	2	(0x04)
432		**fire damper**	bFireDamperErrorClosingRuntime4	Error closing runtime fire damper 4	R			bool	TRUE = ok	32999	3	(0x04)
433		**fire damper**	bFireDamperErrorClosingRuntime5	Error closing runtime fire damper 5	R			bool	TRUE = ok	32999	4	(0x04)
434		**fire damper**	bFireDamperErrorClosingRuntime6	Error closing runtime fire damper 6	R			bool	TRUE = ok	32999	5	(0x04)
435		**fire damper**	bFireDamperErrorClosingRuntime7	Error closing runtime fire damper 7	R			bool	TRUE = ok	32999	6	(0x04)
436		**fire damper**	bFireDamperErrorClosingRuntime8	Error closing runtime fire damper 8	R			bool	TRUE = ok	32999	7	(0x04)
437		**fire damper**	bFireDamperErrorClosingRuntime9	Error closing runtime fire damper 9	R			bool	TRUE = ok	32999	8	(0x04)
438		**fire damper**	bFireDamperErrorClosingRuntime10	Error closing runtime fire damper 10	R			bool	TRUE = ok	32999	9	(0x04)
439		**fire damper**	bFireDamperErrorClosingRuntime11	Error closing runtime fire damper 11	R			bool	TRUE = ok	32999	10	(0x04)
440		**fire damper**	bFireDamperErrorClosingRuntime12	Error closing runtime fire damper 12	R			bool	TRUE = ok	32999	11	(0x04)
441		**fire damper**	bFireDamperErrorClosingRuntime13	Error closing runtime fire damper 13	R			bool	TRUE = ok	32999	12	(0x04)
442		**fire damper**	bFireDamperErrorClosingRuntime14	Error closing runtime fire damper 14	R			bool	TRUE = ok	32999	13	(0x04)
443		**fire damper**	bFireDamperErrorClosingRuntime15	Error closing runtime fire damper 15	R			bool	TRUE = ok	32999	14	(0x04)
444		**fire damper**	bFireDamperErrorClosingRuntime16	Error closing runtime fire damper 16	R			bool	TRUE = ok	32999	15	(0x04)
445		**fire damper**	bFireDamperErrorClosingRuntime17	Error closing runtime fire damper 17	R			bool	TRUE = ok	33000	0	(0x04)
446		**fire damper**	bFireDamperErrorClosingRuntime18	Error closing runtime fire damper 18	R			bool	TRUE = ok	33000	1	(0x04)
447		**fire damper**	bFireDamperErrorClosingRuntime19	Error closing runtime fire damper 19	R			bool	TRUE = ok	33000	2	(0x04)
448		**fire damper**	bFireDamperErrorClosingRuntime20	Error closing runtime fire damper 20	R			bool	TRUE = ok	33000	3	(0x04)
449		**fire damper**	bFireDamperErrorClosingRuntime21	Error closing runtime fire damper 21	R			bool	TRUE = ok	33000	4	(0x04)
450		**fire damper**	bFireDamperErrorClosingRuntime22	Error closing runtime fire damper 22	R			bool	TRUE = ok	33000	5	(0x04)
451		**fire damper**	bFireDamperErrorClosingRuntime23	Error closing runtime fire damper 23	R			bool	TRUE = ok	33000	6	(0x04)
452		**fire damper**	bFireDamperErrorClosingRuntime24	Error closing runtime fire damper 24	R			bool	TRUE = ok	33000	7	(0x04)
453		**fire damper**	bFireDamperErrorClosingRuntime25	Error closing runtime fire damper 25	R			bool	TRUE = ok	33000	8	(0x04)
454		**fire damper**	bFireDamperErrorClosingRuntime26	Error closing runtime fire damper 26	R			bool	TRUE = ok	33000	9	(0x04)
455		**fire damper**	bFireDamperErrorClosingRuntime27	Error closing runtime fire damper 27	R			bool	TRUE = ok	33000	10	(0x04)
456		**fire damper**	bFireDamperErrorClosingRuntime28	Error closing runtime fire damper 28	R			bool	TRUE = ok	33000	11	(0x04)
457		**fire damper**	bFireDamperErrorClosingRuntime29	Error closing runtime fire damper 29	R			bool	TRUE = ok	33000	12	(0x04)
458		**fire damper**	bFireDamperErrorClosingRuntime30	Error closing runtime fire damper 30	R			bool	TRUE = ok	33000	13	(0x04)
459		**fire damper**	bFireDamperErrorClosingRuntime31	Error closing runtime fire damper 31	R			bool	TRUE = ok	33000	14	(0x04)
460		**fire damper**	bFireDamperErrorClosingRuntime32	Error closing runtime fire damper 32	R			bool	TRUE = ok	33000	15	(0x04)
461		**fire damper**	bFireDamperErrorClosingRuntime33	Error closing runtime fire damper 33	R			bool	TRUE = ok	33001	0	(0x04)
462		**fire damper**	bFireDamperErrorClosingRuntime34	Error closing runtime fire damper 34	R			bool	TRUE = ok	33001	1	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
463		**fire damper**	bFireDamperErrorClosingRuntime35	Error closing runtime fire damper 35	R			bool	TRUE = ok	33001	2	(0x04)
464		**fire damper**	bFireDamperErrorClosingRuntime36	Error closing runtime fire damper 36	R			bool	TRUE = ok	33001	3	(0x04)
465		**fire damper**	bFireDamperErrorClosingRuntime37	Error closing runtime fire damper 37	R			bool	TRUE = ok	33001	4	(0x04)
466		**fire damper**	bFireDamperErrorClosingRuntime38	Error closing runtime fire damper 38	R			bool	TRUE = ok	33001	5	(0x04)
467		**fire damper**	bFireDamperErrorClosingRuntime39	Error closing runtime fire damper 39	R			bool	TRUE = ok	33001	6	(0x04)
468		**fire damper**	bFireDamperErrorClosingRuntime40	Error closing runtime fire damper 40	R			bool	TRUE = ok	33001	7	(0x04)
469		**fire damper**	bFireDamperErrorClosingRuntime41	Error closing runtime fire damper 41	R			bool	TRUE = ok	33001	8	(0x04)
470		**fire damper**	bFireDamperErrorClosingRuntime42	Error closing runtime fire damper 42	R			bool	TRUE = ok	33001	9	(0x04)
471		**fire damper**	bFireDamperErrorClosingRuntime43	Error closing runtime fire damper 43	R			bool	TRUE = ok	33001	10	(0x04)
472		**fire damper**	bFireDamperErrorClosingRuntime44	Error closing runtime fire damper 44	R			bool	TRUE = ok	33001	11	(0x04)
473		**fire damper**	bFireDamperErrorClosingRuntime45	Error closing runtime fire damper 45	R			bool	TRUE = ok	33001	12	(0x04)
474		**fire damper**	bFireDamperErrorClosingRuntime46	Error closing runtime fire damper 46	R			bool	TRUE = ok	33001	13	(0x04)
475		**fire damper**	bFireDamperErrorClosingRuntime47	Error closing runtime fire damper 47	R			bool	TRUE = ok	33001	14	(0x04)
476		**fire damper**	bFireDamperErrorClosingRuntime48	Error closing runtime fire damper 48	R			bool	TRUE = ok	33001	15	(0x04)
477		**fire damper**	bFireDamperErrorClosingRuntime49	Error closing runtime fire damper 49	R			bool	TRUE = ok	33002	0	(0x04)
478		**fire damper**	bFireDamperErrorClosingRuntime50	Error closing runtime fire damper 50	R			bool	TRUE = ok	33002	1	(0x04)
479		**fire damper**	bFireDamperErrorClosingRuntime51	Error closing runtime fire damper 51	R			bool	TRUE = ok	33002	2	(0x04)
480		**fire damper**	bFireDamperErrorClosingRuntime52	Error closing runtime fire damper 52	R			bool	TRUE = ok	33002	3	(0x04)
481		**fire damper**	bFireDamperErrorClosingRuntime53	Error closing runtime fire damper 53	R			bool	TRUE = ok	33002	4	(0x04)
482		**fire damper**	bFireDamperErrorClosingRuntime54	Error closing runtime fire damper 54	R			bool	TRUE = ok	33002	5	(0x04)
483		**fire damper**	bFireDamperErrorClosingRuntime55	Error closing runtime fire damper 55	R			bool	TRUE = ok	33002	6	(0x04)
484		**fire damper**	bFireDamperErrorClosingRuntime56	Error closing runtime fire damper 56	R			bool	TRUE = ok	33002	7	(0x04)
485		**fire damper**	bFireDamperErrorClosingRuntime57	Error closing runtime fire damper 57	R			bool	TRUE = ok	33002	8	(0x04)
486		**fire damper**	bFireDamperErrorClosingRuntime58	Error closing runtime fire damper 58	R			bool	TRUE = ok	33002	9	(0x04)
487		**fire damper**	bFireDamperErrorClosingRuntime59	Error closing runtime fire damper 59	R			bool	TRUE = ok	33002	10	(0x04)
488		**fire damper**	bFireDamperErrorClosingRuntime60	Error closing runtime fire damper 60	R			bool	TRUE = ok	33002	11	(0x04)
489		**fire damper**	bFireDamperErrorClosingRuntime61	Error closing runtime fire damper 61	R			bool	TRUE = ok	33002	12	(0x04)
490		**fire damper**	bFireDamperErrorClosingRuntime62	Error closing runtime fire damper 62	R			bool	TRUE = ok	33002	13	(0x04)
491		**fire damper**	bFireDamperErrorOpeningRuntime1	Error Opening runtime fire damper 1	R			bool	TRUE = ok	33003	0	(0x04)
492		**fire damper**	bFireDamperErrorOpeningRuntime2	Error Opening runtime fire damper 2	R			bool	TRUE = ok	33003	1	(0x04)
493		**fire damper**	bFireDamperErrorOpeningRuntime3	Error Opening runtime fire damper 3	R			bool	TRUE = ok	33003	2	(0x04)
494		**fire damper**	bFireDamperErrorOpeningRuntime4	Error Opening runtime fire damper 4	R			bool	TRUE = ok	33003	3	(0x04)
495		**fire damper**	bFireDamperErrorOpeningRuntime5	Error Opening runtime fire damper 5	R			bool	TRUE = ok	33003	4	(0x04)
496		**fire damper**	bFireDamperErrorOpeningRuntime6	Error Opening runtime fire damper 6	R			bool	TRUE = ok	33003	5	(0x04)
497		**fire damper**	bFireDamperErrorOpeningRuntime7	Error Opening runtime fire damper 7	R			bool	TRUE = ok	33003	6	(0x04)
498		**fire damper**	bFireDamperErrorOpeningRuntime8	Error Opening runtime fire damper 8	R			bool	TRUE = ok	33003	7	(0x04)
499		**fire damper**	bFireDamperErrorOpeningRuntime9	Error Opening runtime fire damper 9	R			bool	TRUE = ok	33003	8	(0x04)
500		**fire damper**	bFireDamperErrorOpeningRuntime10	Error Opening runtime fire damper 10	R			bool	TRUE = ok	33003	9	(0x04)
501		**fire damper**	bFireDamperErrorOpeningRuntime11	Error Opening runtime fire damper 11	R			bool	TRUE = ok	33003	10	(0x04)
502		**fire damper**	bFireDamperErrorOpeningRuntime12	Error Opening runtime fire damper 12	R			bool	TRUE = ok	33003	11	(0x04)
503		**fire damper**	bFireDamperErrorOpeningRuntime13	Error Opening runtime fire damper 13	R			bool	TRUE = ok	33003	12	(0x04)
504		**fire damper**	bFireDamperErrorOpeningRuntime14	Error Opening runtime fire damper 14	R			bool	TRUE = ok	33003	13	(0x04)
505		**fire damper**	bFireDamperErrorOpeningRuntime15	Error Opening runtime fire damper 15	R			bool	TRUE = ok	33003	14	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
506		**fire damper**	bFireDamperErrorOpeningRuntime16	Error Opening runtime fire damper 16	R			bool	TRUE = ok	33003	15	(0x04)
507		**fire damper**	bFireDamperErrorOpeningRuntime17	Error Opening runtime fire damper 17	R			bool	TRUE = ok	33004	0	(0x04)
508		**fire damper**	bFireDamperErrorOpeningRuntime18	Error Opening runtime fire damper 18	R			bool	TRUE = ok	33004	1	(0x04)
509		**fire damper**	bFireDamperErrorOpeningRuntime19	Error Opening runtime fire damper 19	R			bool	TRUE = ok	33004	2	(0x04)
510		**fire damper**	bFireDamperErrorOpeningRuntime20	Error Opening runtime fire damper 20	R			bool	TRUE = ok	33004	3	(0x04)
511		**fire damper**	bFireDamperErrorOpeningRuntime21	Error Opening runtime fire damper 21	R			bool	TRUE = ok	33004	4	(0x04)
512		**fire damper**	bFireDamperErrorOpeningRuntime22	Error Opening runtime fire damper 22	R			bool	TRUE = ok	33004	5	(0x04)
513		**fire damper**	bFireDamperErrorOpeningRuntime23	Error Opening runtime fire damper 23	R			bool	TRUE = ok	33004	6	(0x04)
514		**fire damper**	bFireDamperErrorOpeningRuntime24	Error Opening runtime fire damper 24	R			bool	TRUE = ok	33004	7	(0x04)
515		**fire damper**	bFireDamperErrorOpeningRuntime25	Error Opening runtime fire damper 25	R			bool	TRUE = ok	33004	8	(0x04)
516		**fire damper**	bFireDamperErrorOpeningRuntime26	Error Opening runtime fire damper 26	R			bool	TRUE = ok	33004	9	(0x04)
517		**fire damper**	bFireDamperErrorOpeningRuntime27	Error Opening runtime fire damper 27	R			bool	TRUE = ok	33004	10	(0x04)
518		**fire damper**	bFireDamperErrorOpeningRuntime28	Error Opening runtime fire damper 28	R			bool	TRUE = ok	33004	11	(0x04)
519		**fire damper**	bFireDamperErrorOpeningRuntime29	Error Opening runtime fire damper 29	R			bool	TRUE = ok	33004	12	(0x04)
520		**fire damper**	bFireDamperErrorOpeningRuntime30	Error Opening runtime fire damper 30	R			bool	TRUE = ok	33004	13	(0x04)
521		**fire damper**	bFireDamperErrorOpeningRuntime31	Error Opening runtime fire damper 31	R			bool	TRUE = ok	33004	14	(0x04)
522		**fire damper**	bFireDamperErrorOpeningRuntime32	Error Opening runtime fire damper 32	R			bool	TRUE = ok	33004	15	(0x04)
523		**fire damper**	bFireDamperErrorOpeningRuntime33	Error Opening runtime fire damper 33	R			bool	TRUE = ok	33005	0	(0x04)
524		**fire damper**	bFireDamperErrorOpeningRuntime34	Error Opening runtime fire damper 34	R			bool	TRUE = ok	33005	1	(0x04)
525		**fire damper**	bFireDamperErrorOpeningRuntime35	Error Opening runtime fire damper 35	R			bool	TRUE = ok	33005	2	(0x04)
526		**fire damper**	bFireDamperErrorOpeningRuntime36	Error Opening runtime fire damper 36	R			bool	TRUE = ok	33005	3	(0x04)
527		**fire damper**	bFireDamperErrorOpeningRuntime37	Error Opening runtime fire damper 37	R			bool	TRUE = ok	33005	4	(0x04)
528		**fire damper**	bFireDamperErrorOpeningRuntime38	Error Opening runtime fire damper 38	R			bool	TRUE = ok	33005	5	(0x04)
529		**fire damper**	bFireDamperErrorOpeningRuntime39	Error Opening runtime fire damper 39	R			bool	TRUE = ok	33005	6	(0x04)
530		**fire damper**	bFireDamperErrorOpeningRuntime40	Error Opening runtime fire damper 40	R			bool	TRUE = ok	33005	7	(0x04)
531		**fire damper**	bFireDamperErrorOpeningRuntime41	Error Opening runtime fire damper 41	R			bool	TRUE = ok	33005	8	(0x04)
532		**fire damper**	bFireDamperErrorOpeningRuntime42	Error Opening runtime fire damper 42	R			bool	TRUE = ok	33005	9	(0x04)
533		**fire damper**	bFireDamperErrorOpeningRuntime43	Error Opening runtime fire damper 43	R			bool	TRUE = ok	33005	10	(0x04)
534		**fire damper**	bFireDamperErrorOpeningRuntime44	Error Opening runtime fire damper 44	R			bool	TRUE = ok	33005	11	(0x04)
535		**fire damper**	bFireDamperErrorOpeningRuntime45	Error Opening runtime fire damper 45	R			bool	TRUE = ok	33005	12	(0x04)
536		**fire damper**	bFireDamperErrorOpeningRuntime46	Error Opening runtime fire damper 46	R			bool	TRUE = ok	33005	13	(0x04)
537		**fire damper**	bFireDamperErrorOpeningRuntime47	Error Opening runtime fire damper 47	R			bool	TRUE = ok	33005	14	(0x04)
538		**fire damper**	bFireDamperErrorOpeningRuntime48	Error Opening runtime fire damper 48	R			bool	TRUE = ok	33005	15	(0x04)
539		**fire damper**	bFireDamperErrorOpeningRuntime49	Error Opening runtime fire damper 49	R			bool	TRUE = ok	33006	0	(0x04)
540		**fire damper**	bFireDamperErrorOpeningRuntime50	Error Opening runtime fire damper 50	R			bool	TRUE = ok	33006	1	(0x04)
541		**fire damper**	bFireDamperErrorOpeningRuntime51	Error Opening runtime fire damper 51	R			bool	TRUE = ok	33006	2	(0x04)
542		**fire damper**	bFireDamperErrorOpeningRuntime52	Error Opening runtime fire damper 52	R			bool	TRUE = ok	33006	3	(0x04)
543		**fire damper**	bFireDamperErrorOpeningRuntime53	Error Opening runtime fire damper 53	R			bool	TRUE = ok	33006	4	(0x04)
544		**fire damper**	bFireDamperErrorOpeningRuntime54	Error Opening runtime fire damper 54	R			bool	TRUE = ok	33006	5	(0x04)
545		**fire damper**	bFireDamperErrorOpeningRuntime55	Error Opening runtime fire damper 55	R			bool	TRUE = ok	33006	6	(0x04)
546		**fire damper**	bFireDamperErrorOpeningRuntime56	Error Opening runtime fire damper 56	R			bool	TRUE = ok	33006	7	(0x04)
547		**fire damper**	bFireDamperErrorOpeningRuntime57	Error Opening runtime fire damper 57	R			bool	TRUE = ok	33006	8	(0x04)
548		**fire damper**	bFireDamperErrorOpeningRuntime58	Error Opening runtime fire damper 58	R			bool	TRUE = ok	33006	9	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
549		**fire damper**	bFireDamperErrorOpeningRuntime59	Error Opening runtime fire damper 59	R			bool	TRUE = ok	33006	10	(0x04)
550		**fire damper**	bFireDamperErrorOpeningRuntime60	Error Opening runtime fire damper 60	R			bool	TRUE = ok	33006	11	(0x04)
551		**fire damper**	bFireDamperErrorOpeningRuntime61	Error Opening runtime fire damper 61	R			bool	TRUE = ok	33006	12	(0x04)
552		**fire damper**	bFireDamperErrorOpeningRuntime62	Error Opening runtime fire damper 62	R			bool	TRUE = ok	33006	13	(0x04)
553		**fire damper**	bFireDamperErrorPositionIndicator1	Error end switch fire damper 1	R			bool	TRUE = ok	33007	0	(0x04)
554		**fire damper**	bFireDamperErrorPositionIndicator2	Error end switch fire damper 2	R			bool	TRUE = ok	33007	1	(0x04)
555		**fire damper**	bFireDamperErrorPositionIndicator3	Error end switch fire damper 3	R			bool	TRUE = ok	33007	2	(0x04)
556		**fire damper**	bFireDamperErrorPositionIndicator4	Error end switch fire damper 4	R			bool	TRUE = ok	33007	3	(0x04)
557		**fire damper**	bFireDamperErrorPositionIndicator5	Error end switch fire damper 5	R			bool	TRUE = ok	33007	4	(0x04)
558		**fire damper**	bFireDamperErrorPositionIndicator6	Error end switch fire damper 6	R			bool	TRUE = ok	33007	5	(0x04)
559		**fire damper**	bFireDamperErrorPositionIndicator7	Error end switch fire damper 7	R			bool	TRUE = ok	33007	6	(0x04)
560		**fire damper**	bFireDamperErrorPositionIndicator8	Error end switch fire damper 8	R			bool	TRUE = ok	33007	7	(0x04)
561		**fire damper**	bFireDamperErrorPositionIndicator9	Error end switch fire damper 9	R			bool	TRUE = ok	33007	8	(0x04)
562		**fire damper**	bFireDamperErrorPositionIndicator10	Error end switch fire damper 10	R			bool	TRUE = ok	33007	9	(0x04)
563		**fire damper**	bFireDamperErrorPositionIndicator11	Error end switch fire damper 11	R			bool	TRUE = ok	33007	10	(0x04)
564		**fire damper**	bFireDamperErrorPositionIndicator12	Error end switch fire damper 12	R			bool	TRUE = ok	33007	11	(0x04)
565		**fire damper**	bFireDamperErrorPositionIndicator13	Error end switch fire damper 13	R			bool	TRUE = ok	33007	12	(0x04)
566		**fire damper**	bFireDamperErrorPositionIndicator14	Error end switch fire damper 14	R			bool	TRUE = ok	33007	13	(0x04)
567		**fire damper**	bFireDamperErrorPositionIndicator15	Error end switch fire damper 15	R			bool	TRUE = ok	33007	14	(0x04)
568		**fire damper**	bFireDamperErrorPositionIndicator16	Error end switch fire damper 16	R			bool	TRUE = ok	33007	15	(0x04)
569		**fire damper**	bFireDamperErrorPositionIndicator17	Error end switch fire damper 17	R			bool	TRUE = ok	33008	0	(0x04)
570		**fire damper**	bFireDamperErrorPositionIndicator18	Error end switch fire damper 18	R			bool	TRUE = ok	33008	1	(0x04)
571		**fire damper**	bFireDamperErrorPositionIndicator19	Error end switch fire damper 19	R			bool	TRUE = ok	33008	2	(0x04)
572		**fire damper**	bFireDamperErrorPositionIndicator20	Error end switch fire damper 20	R			bool	TRUE = ok	33008	3	(0x04)
573		**fire damper**	bFireDamperErrorPositionIndicator21	Error end switch fire damper 21	R			bool	TRUE = ok	33008	4	(0x04)
574		**fire damper**	bFireDamperErrorPositionIndicator22	Error end switch fire damper 22	R			bool	TRUE = ok	33008	5	(0x04)
575		**fire damper**	bFireDamperErrorPositionIndicator23	Error end switch fire damper 23	R			bool	TRUE = ok	33008	6	(0x04)
576		**fire damper**	bFireDamperErrorPositionIndicator24	Error end switch fire damper 24	R			bool	TRUE = ok	33008	7	(0x04)
577		**fire damper**	bFireDamperErrorPositionIndicator25	Error end switch fire damper 25	R			bool	TRUE = ok	33008	8	(0x04)
578		**fire damper**	bFireDamperErrorPositionIndicator26	Error end switch fire damper 26	R			bool	TRUE = ok	33008	9	(0x04)
579		**fire damper**	bFireDamperErrorPositionIndicator27	Error end switch fire damper 27	R			bool	TRUE = ok	33008	10	(0x04)
580		**fire damper**	bFireDamperErrorPositionIndicator28	Error end switch fire damper 28	R			bool	TRUE = ok	33008	11	(0x04)
581		**fire damper**	bFireDamperErrorPositionIndicator29	Error end switch fire damper 29	R			bool	TRUE = ok	33008	12	(0x04)
582		**fire damper**	bFireDamperErrorPositionIndicator30	Error end switch fire damper 30	R			bool	TRUE = ok	33008	13	(0x04)
583		**fire damper**	bFireDamperErrorPositionIndicator31	Error end switch fire damper 31	R			bool	TRUE = ok	33008	14	(0x04)
584		**fire damper**	bFireDamperErrorPositionIndicator32	Error end switch fire damper 32	R			bool	TRUE = ok	33008	15	(0x04)
585		**fire damper**	bFireDamperErrorPositionIndicator33	Error end switch fire damper 33	R			bool	TRUE = ok	33009	0	(0x04)
586		**fire damper**	bFireDamperErrorPositionIndicator34	Error end switch fire damper 34	R			bool	TRUE = ok	33009	1	(0x04)
587		**fire damper**	bFireDamperErrorPositionIndicator35	Error end switch fire damper 35	R			bool	TRUE = ok	33009	2	(0x04)
588		**fire damper**	bFireDamperErrorPositionIndicator36	Error end switch fire damper 36	R			bool	TRUE = ok	33009	3	(0x04)
589		**fire damper**	bFireDamperErrorPositionIndicator37	Error end switch fire damper 37	R			bool	TRUE = ok	33009	4	(0x04)
590		**fire damper**	bFireDamperErrorPositionIndicator38	Error end switch fire damper 38	R			bool	TRUE = ok	33009	5	(0x04)
591		**fire damper**	bFireDamperErrorPositionIndicator39	Error end switch fire damper 39	R			bool	TRUE = ok	33009	6	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
592		**fire damper**	bFireDamperErrorPositionIndicator40	Error end switch fire damper 40	R			bool	TRUE = ok	33009	7	(0x04)
593		**fire damper**	bFireDamperErrorPositionIndicator41	Error end switch fire damper 41	R			bool	TRUE = ok	33009	8	(0x04)
594		**fire damper**	bFireDamperErrorPositionIndicator42	Error end switch fire damper 42	R			bool	TRUE = ok	33009	9	(0x04)
595		**fire damper**	bFireDamperErrorPositionIndicator43	Error end switch fire damper 43	R			bool	TRUE = ok	33009	10	(0x04)
596		**fire damper**	bFireDamperErrorPositionIndicator44	Error end switch fire damper 44	R			bool	TRUE = ok	33009	11	(0x04)
597		**fire damper**	bFireDamperErrorPositionIndicator45	Error end switch fire damper 45	R			bool	TRUE = ok	33009	12	(0x04)
598		**fire damper**	bFireDamperErrorPositionIndicator46	Error end switch fire damper 46	R			bool	TRUE = ok	33009	13	(0x04)
599		**fire damper**	bFireDamperErrorPositionIndicator47	Error end switch fire damper 47	R			bool	TRUE = ok	33009	14	(0x04)
600		**fire damper**	bFireDamperErrorPositionIndicator48	Error end switch fire damper 48	R			bool	TRUE = ok	33009	15	(0x04)
601		**fire damper**	bFireDamperErrorPositionIndicator49	Error end switch fire damper 49	R			bool	TRUE = ok	33010	0	(0x04)
602		**fire damper**	bFireDamperErrorPositionIndicator50	Error end switch fire damper 50	R			bool	TRUE = ok	33010	1	(0x04)
603		**fire damper**	bFireDamperErrorPositionIndicator51	Error end switch fire damper 51	R			bool	TRUE = ok	33010	2	(0x04)
604		**fire damper**	bFireDamperErrorPositionIndicator52	Error end switch fire damper 52	R			bool	TRUE = ok	33010	3	(0x04)
605		**fire damper**	bFireDamperErrorPositionIndicator53	Error end switch fire damper 53	R			bool	TRUE = ok	33010	4	(0x04)
606		**fire damper**	bFireDamperErrorPositionIndicator54	Error end switch fire damper 54	R			bool	TRUE = ok	33010	5	(0x04)
607		**fire damper**	bFireDamperErrorPositionIndicator55	Error end switch fire damper 55	R			bool	TRUE = ok	33010	6	(0x04)
608		**fire damper**	bFireDamperErrorPositionIndicator56	Error end switch fire damper 56	R			bool	TRUE = ok	33010	7	(0x04)
609		**fire damper**	bFireDamperErrorPositionIndicator57	Error end switch fire damper 57	R			bool	TRUE = ok	33010	8	(0x04)
610		**fire damper**	bFireDamperErrorPositionIndicator58	Error end switch fire damper 58	R			bool	TRUE = ok	33010	9	(0x04)
611		**fire damper**	bFireDamperErrorPositionIndicator59	Error end switch fire damper 59	R			bool	TRUE = ok	33010	10	(0x04)
612		**fire damper**	bFireDamperErrorPositionIndicator60	Error end switch fire damper 60	R			bool	TRUE = ok	33010	11	(0x04)
613		**fire damper**	bFireDamperErrorPositionIndicator61	Error end switch fire damper 61	R			bool	TRUE = ok	33010	12	(0x04)
614		**fire damper**	bFireDamperErrorPositionIndicator62	Error end switch fire damper 62	R			bool	TRUE = ok	33010	13	(0x04)
615		**fire damper**	bFireDamper1	Error triggered fire damper 1	R			bool	TRUE = ok	33011	0	(0x04)
616		**fire damper**	bFireDamper2	Error triggered fire damper 2	R			bool	TRUE = ok	33011	1	(0x04)
617		**fire damper**	bFireDamper3	Error triggered fire damper 3	R			bool	TRUE = ok	33011	2	(0x04)
618		**fire damper**	bFireDamper4	Error triggered fire damper 4	R			bool	TRUE = ok	33011	3	(0x04)
619		**fire damper**	bFireDamper5	Error triggered fire damper 5	R			bool	TRUE = ok	33011	4	(0x04)
620		**fire damper**	bFireDamper6	Error triggered fire damper 6	R			bool	TRUE = ok	33011	5	(0x04)
621		**fire damper**	bFireDamper7	Error triggered fire damper 7	R			bool	TRUE = ok	33011	6	(0x04)
622		**fire damper**	bFireDamper8	Error triggered fire damper 8	R			bool	TRUE = ok	33011	7	(0x04)
623		**fire damper**	bFireDamper9	Error triggered fire damper 9	R			bool	TRUE = ok	33011	8	(0x04)
624		**fire damper**	bFireDamper10	Error triggered fire damper 10	R			bool	TRUE = ok	33011	9	(0x04)
625		**fire damper**	bFireDamper11	Error triggered fire damper 11	R			bool	TRUE = ok	33011	10	(0x04)
626		**fire damper**	bFireDamper12	Error triggered fire damper 12	R			bool	TRUE = ok	33011	11	(0x04)
627		**fire damper**	bFireDamper13	Error triggered fire damper 13	R			bool	TRUE = ok	33011	12	(0x04)
628		**fire damper**	bFireDamper14	Error triggered fire damper 14	R			bool	TRUE = ok	33011	13	(0x04)
629		**fire damper**	bFireDamper15	Error triggered fire damper 15	R			bool	TRUE = ok	33011	14	(0x04)
630		**fire damper**	bFireDamper16	Error triggered fire damper 16	R			bool	TRUE = ok	33011	15	(0x04)
631		**fire damper**	bFireDamper17	Error triggered fire damper 17	R			bool	TRUE = ok	33012	0	(0x04)
632		**fire damper**	bFireDamper18	Error triggered fire damper 18	R			bool	TRUE = ok	33012	1	(0x04)
633		**fire damper**	bFireDamper19	Error triggered fire damper 19	R			bool	TRUE = ok	33012	2	(0x04)
634		**fire damper**	bFireDamper20	Error triggered fire damper 20	R			bool	TRUE = ok	33012	3	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
635		**fire damper**	bFireDamper21	Error triggered fire damper 21	R			bool	TRUE = ok	33012	4	(0x04)
636		**fire damper**	bFireDamper22	Error triggered fire damper 22	R			bool	TRUE = ok	33012	5	(0x04)
637		**fire damper**	bFireDamper23	Error triggered fire damper 23	R			bool	TRUE = ok	33012	6	(0x04)
638		**fire damper**	bFireDamper24	Error triggered fire damper 24	R			bool	TRUE = ok	33012	7	(0x04)
639		**fire damper**	bFireDamper25	Error triggered fire damper 25	R			bool	TRUE = ok	33012	8	(0x04)
640		**fire damper**	bFireDamper26	Error triggered fire damper 26	R			bool	TRUE = ok	33012	9	(0x04)
641		**fire damper**	bFireDamper27	Error triggered fire damper 27	R			bool	TRUE = ok	33012	10	(0x04)
642		**fire damper**	bFireDamper28	Error triggered fire damper 28	R			bool	TRUE = ok	33012	11	(0x04)
643		**fire damper**	bFireDamper29	Error triggered fire damper 29	R			bool	TRUE = ok	33012	12	(0x04)
644		**fire damper**	bFireDamper30	Error triggered fire damper 30	R			bool	TRUE = ok	33012	13	(0x04)
645		**fire damper**	bFireDamper31	Error triggered fire damper 31	R			bool	TRUE = ok	33012	14	(0x04)
646		**fire damper**	bFireDamper32	Error triggered fire damper 32	R			bool	TRUE = ok	33012	15	(0x04)
647		**fire damper**	bFireDamper33	Error triggered fire damper 33	R			bool	TRUE = ok	33013	0	(0x04)
648		**fire damper**	bFireDamper34	Error triggered fire damper 34	R			bool	TRUE = ok	33013	1	(0x04)
649		**fire damper**	bFireDamper35	Error triggered fire damper 35	R			bool	TRUE = ok	33013	2	(0x04)
650		**fire damper**	bFireDamper36	Error triggered fire damper 36	R			bool	TRUE = ok	33013	3	(0x04)
651		**fire damper**	bFireDamper37	Error triggered fire damper 37	R			bool	TRUE = ok	33013	4	(0x04)
652		**fire damper**	bFireDamper38	Error triggered fire damper 38	R			bool	TRUE = ok	33013	5	(0x04)
653		**fire damper**	bFireDamper39	Error triggered fire damper 39	R			bool	TRUE = ok	33013	6	(0x04)
654		**fire damper**	bFireDamper40	Error triggered fire damper 40	R			bool	TRUE = ok	33013	7	(0x04)
655		**fire damper**	bFireDamper41	Error triggered fire damper 41	R			bool	TRUE = ok	33013	8	(0x04)
656		**fire damper**	bFireDamper42	Error triggered fire damper 42	R			bool	TRUE = ok	33013	9	(0x04)
657		**fire damper**	bFireDamper43	Error triggered fire damper 43	R			bool	TRUE = ok	33013	10	(0x04)
658		**fire damper**	bFireDamper44	Error triggered fire damper 44	R			bool	TRUE = ok	33013	11	(0x04)
659		**fire damper**	bFireDamper45	Error triggered fire damper 45	R			bool	TRUE = ok	33013	12	(0x04)
660		**fire damper**	bFireDamper46	Error triggered fire damper 46	R			bool	TRUE = ok	33013	13	(0x04)
661		**fire damper**	bFireDamper47	Error triggered fire damper 47	R			bool	TRUE = ok	33013	14	(0x04)
662		**fire damper**	bFireDamper48	Error triggered fire damper 48	R			bool	TRUE = ok	33013	15	(0x04)
663		**fire damper**	bFireDamper49	Error triggered fire damper 49	R			bool	TRUE = ok	33014	0	(0x04)
664		**fire damper**	bFireDamper50	Error triggered fire damper 50	R			bool	TRUE = ok	33014	1	(0x04)
665		**fire damper**	bFireDamper51	Error triggered fire damper 51	R			bool	TRUE = ok	33014	2	(0x04)
666		**fire damper**	bFireDamper52	Error triggered fire damper 52	R			bool	TRUE = ok	33014	3	(0x04)
667		**fire damper**	bFireDamper53	Error triggered fire damper 53	R			bool	TRUE = ok	33014	4	(0x04)
668		**fire damper**	bFireDamper54	Error triggered fire damper 54	R			bool	TRUE = ok	33014	5	(0x04)
669		**fire damper**	bFireDamper55	Error triggered fire damper 55	R			bool	TRUE = ok	33014	6	(0x04)
670		**fire damper**	bFireDamper56	Error triggered fire damper 56	R			bool	TRUE = ok	33014	7	(0x04)
671		**fire damper**	bFireDamper57	Error triggered fire damper 57	R			bool	TRUE = ok	33014	8	(0x04)
672		**fire damper**	bFireDamper58	Error triggered fire damper 58	R			bool	TRUE = ok	33014	9	(0x04)
673		**fire damper**	bFireDamper59	Error triggered fire damper 59	R			bool	TRUE = ok	33014	10	(0x04)
674		**fire damper**	bFireDamper60	Error triggered fire damper 60	R			bool	TRUE = ok	33014	11	(0x04)
675		**fire damper**	bFireDamper61	Error triggered fire damper 61	R			bool	TRUE = ok	33014	12	(0x04)
676		**fire damper**	bFireDamper62	Error triggered fire damper 62	R			bool	TRUE = ok	33014	13	(0x04)
677		**supply air fan 3 to 16**	bFanCtrlOperationSUP3	controlled value operation signal of the supply air fan 3	R			bool	TRUE = on	33015	0	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
678		**supply air fan 3 to 16**	bFanCtrlOperationSUP4	controlled value operation signal of the supply air fan 4	R			bool	TRUE = on	33015	1	(0x04)
679		**supply air fan 3 to 16**	bFanCtrlOperationSUP5	controlled value operation signal of the supply air fan 5	R			bool	TRUE = on	33015	2	(0x04)
680		**supply air fan 3 to 16**	bFanCtrlOperationSUP6	controlled value operation signal of the supply air fan 6	R			bool	TRUE = on	33015	3	(0x04)
681		**supply air fan 3 to 16**	bFanCtrlOperationSUP7	controlled value operation signal of the supply air fan 7	R			bool	TRUE = on	33015	4	(0x04)
682		**supply air fan 3 to 16**	bFanCtrlOperationSUP8	controlled value operation signal of the supply air fan 8	R			bool	TRUE = on	33015	5	(0x04)
683		**supply air fan 3 to 16**	bFanCtrlOperationSUP9	controlled value operation signal of the supply air fan 9	R			bool	TRUE = on	33015	6	(0x04)
684		**supply air fan 3 to 16**	bFanCtrlOperationSUP10	controlled value operation signal of the supply air fan 10	R			bool	TRUE = on	33015	7	(0x04)
685		**supply air fan 3 to 16**	bFanCtrlOperationSUP11	controlled value operation signal of the supply air fan 11	R			bool	TRUE = on	33015	8	(0x04)
686		**supply air fan 3 to 16**	bFanCtrlOperationSUP12	controlled value operation signal of the supply air fan 12	R			bool	TRUE = on	33015	9	(0x04)
687		**supply air fan 3 to 16**	bFanCtrlOperationSUP13	controlled value operation signal of the supply air fan 13	R			bool	TRUE = on	33015	10	(0x04)
688		**supply air fan 3 to 16**	bFanCtrlOperationSUP14	controlled value operation signal of the supply air fan 14	R			bool	TRUE = on	33015	11	(0x04)
689		**supply air fan 3 to 16**	bFanCtrlOperationSUP15	controlled value operation signal of the supply air fan 15	R			bool	TRUE = on	33015	12	(0x04)
690		**supply air fan 3 to 16**	bFanCtrlOperationSUP16	controlled value operation signal of the supply air fan 16	R			bool	TRUE = on	33015	13	(0x04)
691		**supply air fan 3 to 16**	bFanStateErrorSUP3	internal error supply air fan 3	R			bool	TRUE = ok	33016	0	(0x04)
692		**supply air fan 3 to 16**	bFanStateErrorSUP4	internal error supply air fan 4	R			bool	TRUE = ok	33016	1	(0x04)
693		**supply air fan 3 to 16**	bFanStateErrorSUP5	internal error supply air fan 5	R			bool	TRUE = ok	33016	2	(0x04)
694		**supply air fan 3 to 16**	bFanStateErrorSUP6	internal error supply air fan 6	R			bool	TRUE = ok	33016	3	(0x04)
695		**supply air fan 3 to 16**	bFanStateErrorSUP7	internal error supply air fan 7	R			bool	TRUE = ok	33016	4	(0x04)
696		**supply air fan 3 to 16**	bFanStateErrorSUP8	internal error supply air fan 8	R			bool	TRUE = ok	33016	5	(0x04)
697		**supply air fan 3 to 16**	bFanStateErrorSUP9	internal error supply air fan 9	R			bool	TRUE = ok	33016	6	(0x04)
698		**supply air fan 3 to 16**	bFanStateErrorSUP10	internal error supply air fan 10	R			bool	TRUE = ok	33016	7	(0x04)
699		**supply air fan 3 to 16**	bFanStateErrorSUP11	internal error supply air fan 11	R			bool	TRUE = ok	33016	8	(0x04)
700		**supply air fan 3 to 16**	bFanStateErrorSUP12	internal error supply air fan 12	R			bool	TRUE = ok	33016	9	(0x04)
701		**supply air fan 3 to 16**	bFanStateErrorSUP13	internal error supply air fan 13	R			bool	TRUE = ok	33016	10	(0x04)
702		**supply air fan 3 to 16**	bFanStateErrorSUP14	internal error supply air fan 14	R			bool	TRUE = ok	33016	11	(0x04)
703		**supply air fan 3 to 16**	bFanStateErrorSUP15	internal error supply air fan 15	R			bool	TRUE = ok	33016	12	(0x04)
704		**supply air fan 3 to 16**	bFanStateErrorSUP16	internal error supply air fan 16	R			bool	TRUE = ok	33016	13	(0x04)
705		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP3	modbus comm. error with the supply air fan 3	R			bool	TRUE = ok	33017	0	(0x04)
706		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP4	modbus comm. error with the supply air fan 4	R			bool	TRUE = ok	33017	1	(0x04)
707		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP5	modbus comm. error with the supply air fan 5	R			bool	TRUE = ok	33017	2	(0x04)
708		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP6	modbus comm. error with the supply air fan 6	R			bool	TRUE = ok	33017	3	(0x04)
709		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP7	modbus comm. error with the supply air fan 7	R			bool	TRUE = ok	33017	4	(0x04)
710		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP8	modbus comm. error with the supply air fan 8	R			bool	TRUE = ok	33017	5	(0x04)
711		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP9	modbus comm. error with the supply air fan 9	R			bool	TRUE = ok	33017	6	(0x04)
712		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP10	modbus comm. error with the supply air fan 10	R			bool	TRUE = ok	33017	7	(0x04)
713		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP11	modbus comm. error with the supply air fan 11	R			bool	TRUE = ok	33017	8	(0x04)
714		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP12	modbus comm. error with the supply air fan 12	R			bool	TRUE = ok	33017	9	(0x04)
715		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP13	modbus comm. error with the supply air fan 13	R			bool	TRUE = ok	33017	10	(0x04)
716		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP14	modbus comm. error with the supply air fan 14	R			bool	TRUE = ok	33017	11	(0x04)
717		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP15	modbus comm. error with the supply air fan 15	R			bool	TRUE = ok	33017	12	(0x04)
718		**supply air fan 3 to 16**	bFanStateErrModbusFuSUP16	modbus comm. error with the supply air fan 16	R			bool	TRUE = ok	33017	13	(0x04)
719		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP3	repair switch of the supply air fan is active 3	R			bool	TRUE = ok	33018	0	(0x04)
720		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP4	repair switch of the supply air fan is active 4	R			bool	TRUE = ok	33018	1	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
721		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP5	repair switch of the supply air fan is active 5	R			bool	TRUE = ok	33018	2	(0x04)
722		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP6	repair switch of the supply air fan is active 6	R			bool	TRUE = ok	33018	3	(0x04)
723		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP7	repair switch of the supply air fan is active 7	R			bool	TRUE = ok	33018	4	(0x04)
724		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP8	repair switch of the supply air fan is active 8	R			bool	TRUE = ok	33018	5	(0x04)
725		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP9	repair switch of the supply air fan is active 9	R			bool	TRUE = ok	33018	6	(0x04)
726		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP10	repair switch of the supply air fan is active 10	R			bool	TRUE = ok	33018	7	(0x04)
727		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP11	repair switch of the supply air fan is active 11	R			bool	TRUE = ok	33018	8	(0x04)
728		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP12	repair switch of the supply air fan is active 12	R			bool	TRUE = ok	33018	9	(0x04)
729		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP13	repair switch of the supply air fan is active 13	R			bool	TRUE = ok	33018	10	(0x04)
730		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP14	repair switch of the supply air fan is active 14	R			bool	TRUE = ok	33018	11	(0x04)
731		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP15	repair switch of the supply air fan is active 15	R			bool	TRUE = ok	33018	12	(0x04)
732		**supply air fan 3 to 16**	bFanStateRepairSwitchSUP16	repair switch of the supply air fan is active 16	R			bool	TRUE = ok	33018	13	(0x04)
733		**extract air fan 3 to 16**	bFanCtrlOperationETA3	controlled value operation signal of the extract air fan 3	R			bool	TRUE = on	33019	0	(0x04)
734		**extract air fan 3 to 16**	bFanCtrlOperationETA4	controlled value operation signal of the extract air fan 4	R			bool	TRUE = on	33019	1	(0x04)
735		**extract air fan 3 to 16**	bFanCtrlOperationETA5	controlled value operation signal of the extract air fan 5	R			bool	TRUE = on	33019	2	(0x04)
736		**extract air fan 3 to 16**	bFanCtrlOperationETA6	controlled value operation signal of the extract air fan 6	R			bool	TRUE = on	33019	3	(0x04)
737		**extract air fan 3 to 16**	bFanCtrlOperationETA7	controlled value operation signal of the extract air fan 7	R			bool	TRUE = on	33019	4	(0x04)
738		**extract air fan 3 to 16**	bFanCtrlOperationETA8	controlled value operation signal of the extract air fan 8	R			bool	TRUE = on	33019	5	(0x04)
739		**extract air fan 3 to 16**	bFanCtrlOperationETA9	controlled value operation signal of the extract air fan 9	R			bool	TRUE = on	33019	6	(0x04)
740		**extract air fan 3 to 16**	bFanCtrlOperationETA10	controlled value operation signal of the extract air fan 10	R			bool	TRUE = on	33019	7	(0x04)
741		**extract air fan 3 to 16**	bFanCtrlOperationETA11	controlled value operation signal of the extract air fan 11	R			bool	TRUE = on	33019	8	(0x04)
742		**extract air fan 3 to 16**	bFanCtrlOperationETA12	controlled value operation signal of the extract air fan 12	R			bool	TRUE = on	33019	9	(0x04)
743		**extract air fan 3 to 16**	bFanCtrlOperationETA13	controlled value operation signal of the extract air fan 13	R			bool	TRUE = on	33019	10	(0x04)
744		**extract air fan 3 to 16**	bFanCtrlOperationETA14	controlled value operation signal of the extract air fan 14	R			bool	TRUE = on	33019	11	(0x04)
745		**extract air fan 3 to 16**	bFanCtrlOperationETA15	controlled value operation signal of the extract air fan 15	R			bool	TRUE = on	33019	12	(0x04)
746		**extract air fan 3 to 16**	bFanCtrlOperationETA16	controlled value operation signal of the extract air fan 16	R			bool	TRUE = on	33019	13	(0x04)
747		**extract air fan 3 to 16**	bFanStateErrorETA3	internal error extract air fan 3	R			bool	TRUE = ok	33020	0	(0x04)
748		**extract air fan 3 to 16**	bFanStateErrorETA4	internal error extract air fan 4	R			bool	TRUE = ok	33020	1	(0x04)
749		**extract air fan 3 to 16**	bFanStateErrorETA5	internal error extract air fan 5	R			bool	TRUE = ok	33020	2	(0x04)
750		**extract air fan 3 to 16**	bFanStateErrorETA6	internal error extract air fan 6	R			bool	TRUE = ok	33020	3	(0x04)
751		**extract air fan 3 to 16**	bFanStateErrorETA7	internal error extract air fan 7	R			bool	TRUE = ok	33020	4	(0x04)
752		**extract air fan 3 to 16**	bFanStateErrorETA8	internal error extract air fan 8	R			bool	TRUE = ok	33020	5	(0x04)
753		**extract air fan 3 to 16**	bFanStateErrorETA9	internal error extract air fan 9	R			bool	TRUE = ok	33020	6	(0x04)
754		**extract air fan 3 to 16**	bFanStateErrorETA10	internal error extract air fan 10	R			bool	TRUE = ok	33020	7	(0x04)
755		**extract air fan 3 to 16**	bFanStateErrorETA11	internal error extract air fan 11	R			bool	TRUE = ok	33020	8	(0x04)
756		**extract air fan 3 to 16**	bFanStateErrorETA12	internal error extract air fan 12	R			bool	TRUE = ok	33020	9	(0x04)
757		**extract air fan 3 to 16**	bFanStateErrorETA13	internal error extract air fan 13	R			bool	TRUE = ok	33020	10	(0x04)
758		**extract air fan 3 to 16**	bFanStateErrorETA14	internal error extract air fan 14	R			bool	TRUE = ok	33020	11	(0x04)
759		**extract air fan 3 to 16**	bFanStateErrorETA15	internal error extract air fan 15	R			bool	TRUE = ok	33020	12	(0x04)
760		**extract air fan 3 to 16**	bFanStateErrorETA16	internal error extract air fan 16	R			bool	TRUE = ok	33020	13	(0x04)
761		**extract air fan 3 to 16**	bFanStateErrModbusFuETA3	modbus comm. error with the extract air fan 3	R			bool	TRUE = ok	33021	0	(0x04)
762		**extract air fan 3 to 16**	bFanStateErrModbusFuETA4	modbus comm. error with the extract air fan 4	R			bool	TRUE = ok	33021	1	(0x04)
763		**extract air fan 3 to 16**	bFanStateErrModbusFuETA5	modbus comm. error with the extract air fan 5	R			bool	TRUE = ok	33021	2	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
764		**extract air fan 3 to 16**	bFanStateErrModbusFuETA6	modbus comm. error with the extract air fan 6	R			bool	TRUE = ok	33021	3	(0x04)
765		**extract air fan 3 to 16**	bFanStateErrModbusFuETA7	modbus comm. error with the extract air fan 7	R			bool	TRUE = ok	33021	4	(0x04)
766		**extract air fan 3 to 16**	bFanStateErrModbusFuETA8	modbus comm. error with the extract air fan 8	R			bool	TRUE = ok	33021	5	(0x04)
767		**extract air fan 3 to 16**	bFanStateErrModbusFuETA9	modbus comm. error with the extract air fan 9	R			bool	TRUE = ok	33021	6	(0x04)
768		**extract air fan 3 to 16**	bFanStateErrModbusFuETA10	modbus comm. error with the extract air fan 10	R			bool	TRUE = ok	33021	7	(0x04)
769		**extract air fan 3 to 16**	bFanStateErrModbusFuETA11	modbus comm. error with the extract air fan 11	R			bool	TRUE = ok	33021	8	(0x04)
770		**extract air fan 3 to 16**	bFanStateErrModbusFuETA12	modbus comm. error with the extract air fan 12	R			bool	TRUE = ok	33021	9	(0x04)
771		**extract air fan 3 to 16**	bFanStateErrModbusFuETA13	modbus comm. error with the extract air fan 13	R			bool	TRUE = ok	33021	10	(0x04)
772		**extract air fan 3 to 16**	bFanStateErrModbusFuETA14	modbus comm. error with the extract air fan 14	R			bool	TRUE = ok	33021	11	(0x04)
773		**extract air fan 3 to 16**	bFanStateErrModbusFuETA15	modbus comm. error with the extract air fan 15	R			bool	TRUE = ok	33021	12	(0x04)
774		**extract air fan 3 to 16**	bFanStateErrModbusFuETA16	modbus comm. error with the extract air fan 16	R			bool	TRUE = ok	33021	13	(0x04)
775		**extract air fan 3 to 16**	bFanStateRepairSwitchETA3	repair switch of the extract air fan is active 3	R			bool	TRUE = ok	33022	0	(0x04)
776		**extract air fan 3 to 16**	bFanStateRepairSwitchETA4	repair switch of the extract air fan is active 4	R			bool	TRUE = ok	33022	1	(0x04)
777		**extract air fan 3 to 16**	bFanStateRepairSwitchETA5	repair switch of the extract air fan is active 5	R			bool	TRUE = ok	33022	2	(0x04)
778		**extract air fan 3 to 16**	bFanStateRepairSwitchETA6	repair switch of the extract air fan is active 6	R			bool	TRUE = ok	33022	3	(0x04)
779		**extract air fan 3 to 16**	bFanStateRepairSwitchETA7	repair switch of the extract air fan is active 7	R			bool	TRUE = ok	33022	4	(0x04)
780		**extract air fan 3 to 16**	bFanStateRepairSwitchETA8	repair switch of the extract air fan is active 8	R			bool	TRUE = ok	33022	5	(0x04)
781		**extract air fan 3 to 16**	bFanStateRepairSwitchETA9	repair switch of the extract air fan is active 9	R			bool	TRUE = ok	33022	6	(0x04)
782		**extract air fan 3 to 16**	bFanStateRepairSwitchETA10	repair switch of the extract air fan is active 10	R			bool	TRUE = ok	33022	7	(0x04)
783		**extract air fan 3 to 16**	bFanStateRepairSwitchETA11	repair switch of the extract air fan is active 11	R			bool	TRUE = ok	33022	8	(0x04)
784		**extract air fan 3 to 16**	bFanStateRepairSwitchETA12	repair switch of the extract air fan is active 12	R			bool	TRUE = ok	33022	9	(0x04)
785		**extract air fan 3 to 16**	bFanStateRepairSwitchETA13	repair switch of the extract air fan is active 13	R			bool	TRUE = ok	33022	10	(0x04)
786		**extract air fan 3 to 16**	bFanStateRepairSwitchETA14	repair switch of the extract air fan is active 14	R			bool	TRUE = ok	33022	11	(0x04)
787		**extract air fan 3 to 16**	bFanStateRepairSwitchETA15	repair switch of the extract air fan is active 15	R			bool	TRUE = ok	33022	12	(0x04)
788		**extract air fan 3 to 16**	bFanStateRepairSwitchETA16	repair switch of the extract air fan is active 16	R			bool	TRUE = ok	33022	13	(0x04)
789		**electric preheater**	bPreEHeatStateFlowDetector	electric preheater flow detector	R			bool	TRUE = ok	33024	0	(0x04)
790		**electric preheater**	bPreEHeatStateErrTempLimiter	electric preheater temperature limiter	R			bool	TRUE = ok	33024	1	(0x04)
791		**electric preheater**	bPreEHeatStateErrTempWarning	electric preheater temperature monitor	R			bool	TRUE = ok	33024	2	(0x04)
792	**S**	**electric preheater**	bPreEHeatCtrlON	controlled value operation signal of the electric preheater	R			bool	TRUE = on	33024	3	(0x04)
793	**S**	**electric preheater**	fPreEHeatCtrlPower	controlled value of the electric preheater power [0..100%]	R	%		uint		33025		(0x04)
794		**electric reheater**	bReEHeatStateFlowDetector	electric reheater flow detector	R			bool	TRUE = ok	33028	0	(0x04)
795		**electric reheater**	bReEHeatStateErrTempLimiter	electric reheater temperature limiter	R			bool	TRUE = ok	33028	1	(0x04)
796		**electric reheater**	bReEHeatStateErrTempWarning	electric reheater temperature monitor	R			bool	TRUE = ok	33028	2	(0x04)
797	**S**	**electric reheater**	bReEHeatCtrlON	controlled value operation signal of the electric reheater	R			bool	TRUE = on	33028	3	(0x04)
798	**S**	**electric reheater**	fReEHeatCtrlPower	controlled value of the electric reheater power [0..100%]	R	%		uint		33029		(0x04)
799	**S**	**external chiller**	bExternalChillerError	ext. chiller error	R			bool	TRUE = ok	33032	0	(0x04)
800	**S**	**external chiller**	bExternalChillerStateOn	ext. chiller operating	R			bool	TRUE = on	33032	1	(0x04)
801	**S**	**external chiller**	bExternalChillerCtrlOn	controlled value operation signal of the ext. chiller	R			bool	TRUE = on	33032	2	(0x04)
802	**S**	**external chiller**	fExternalChillerCtrlPower	controlled value of the ext. chiller power [0..100%]	R	%		uint		33033		(0x04)
803		**humidifier**	bHumidifierStateActive	humidifier active	R			bool	TRUE = active	33037	0	(0x04)
804	**S**	**humidifier**	bHumidifierStateError	humidifier error	R			bool	TRUE = ok	33037	1	(0x04)
805	**S**	**humidifier**	bHumidifierStateOn	humidifier operating	R			bool	TRUE = on	33037	2	(0x04)
806		**humidifier**	bHumidifierStateHygrostat	humidifier hygrostat	R			bool	TRUE = ok	33037	3	(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
807		**humidifier**	bHumidifierStateService	humidifier service	R			bool	TRUE = on	33037	4	(0x04)
808		**humidifier**	bHumidifierCtrlClean	controlled value to clean humidifier	R			bool	TRUE = on	33037	5	(0x04)
809	**S**	**humidifier**	bHumidifierCtrlOn	controlled value operation signal of the humidifier	R			bool	TRUE = on	33037	6	(0x04)
810	**S**	**humidifier**	fHumidifierCtrlPower	controlled value of the humidifier power [0..100%]	R	%		uint		33038		(0x04)
811	**S**	**heat pump 1**	bHeatPumpStateError1	heatpump 1 error	R			bool	TRUE = ok	33042	0	(0x04)
812	**S**	**heat pump 1**	bHeatPumpStateDeicing1	heatpump 1 deicing	R			bool	TRUE = active	33042	1	(0x04)
813	**S**	**heat pump 1**	bHeatPumpStateHeating1	heatpump 1 heating	R			bool	TRUE = active	33042	2	(0x04)
814	**S**	**heat pump 1**	bHeatPumpStateOperation1	heatpump 1 operating	R			bool	TRUE = active	33042	3	(0x04)
815	**S**	**heat pump 1**	bHeatPumpCtrlCooling1	controlled value cooling signal of the heatpump 1	R			bool	TRUE = on	33042	4	(0x04)
816	**S**	**heat pump 1**	bHeatPumpCtrlHeating1	controlled value heating signal of the heatpump 1	R			bool	TRUE = on	33042	5	(0x04)
817	**S**	**heat pump 1**	bHeatPumpCtrlRelease1	controlled value operation signal of the heatpump 1	R			bool	TRUE = on	33042	6	(0x04)
818	**S**	**heat pump 1**	fHeatPumpCtrlPower1	controlled value of the heatpump 1 power [0..100%]	R	%		uint		33043		(0x04)
819	**S**	**heat pump 2**	bHeatPumpStateError2	heatpump 2 error	R			bool	TRUE = ok	33047	0	(0x04)
820	**S**	**heat pump 2**	bHeatPumpStateDeicing2	heatpump 2 deicing	R			bool	TRUE = active	33047	1	(0x04)
821	**S**	**heat pump 2**	bHeatPumpStateHeating2	heatpump 2 heating	R			bool	TRUE = active	33047	2	(0x04)
822	**S**	**heat pump 2**	bHeatPumpStateOperation2	heatpump 2 operating	R			bool	TRUE = active	33047	3	(0x04)
823	**S**	**heat pump 2**	bHeatPumpCtrlCooling2	controlled value cooling signal of the heatpump 2	R			bool	TRUE = on	33047	4	(0x04)
824	**S**	**heat pump 2**	bHeatPumpCtrlHeating2	controlled value heating signal of the heatpump 2	R			bool	TRUE = on	33047	5	(0x04)
825	**S**	**heat pump 2**	bHeatPumpCtrlRelease2	controlled value operation signal of the heatpump 2	R			bool	TRUE = on	33047	6	(0x04)
826	**S**	**heat pump 2**	fHeatPumpCtrlPower2	controlled value of the heatpump 2 power [0..200%]	R	%		uint		33048		(0x04)
827		**heat pump 3**	bHeatPumpStateError3	heatpump 3 error	R			bool	TRUE = ok	33052	0	(0x04)
828		**heat pump 3**	bHeatPumpStateDeicing3	heatpump 3 deicing	R			bool	TRUE = active	33052	1	(0x04)
829		**heat pump 3**	bHeatPumpStateHeating3	heatpump 3 heating	R			bool	TRUE = active	33052	2	(0x04)
830		**heat pump 3**	bHeatPumpStateOperation3	heatpump 3 operating	R			bool	TRUE = active	33052	3	(0x04)
831		**heat pump 3**	bHeatPumpCtrlCooling3	controlled value cooling signal of the heatpump 3	R			bool	TRUE = on	33052	4	(0x04)
832		**heat pump 3**	bHeatPumpCtrlHeating3	controlled value heating signal of the heatpump 3	R			bool	TRUE = on	33052	5	(0x04)
833		**heat pump 3**	bHeatPumpCtrlRelease3	controlled value operation signal of the heatpump 3	R			bool	TRUE = on	33052	6	(0x04)
834		**heat pump 3**	fHeatPumpCtrlPower3	controlled value of the heatpump 3 power [0..300%]	R	%		uint		33053		(0x04)
835		**heat pump 4**	bHeatPumpStateError4	heatpump 4 error	R			bool	TRUE = ok	33057	0	(0x04)
836		**heat pump 4**	bHeatPumpStateDeicing4	heatpump 4 deicing	R			bool	TRUE = active	33057	1	(0x04)
837		**heat pump 4**	bHeatPumpStateHeating4	heatpump 4 heating	R			bool	TRUE = active	33057	2	(0x04)
838		**heat pump 4**	bHeatPumpStateOperation4	heatpump 4 operating	R			bool	TRUE = active	33057	3	(0x04)
839		**heat pump 4**	bHeatPumpCtrlCooling4	controlled value cooling signal of the heatpump 4	R			bool	TRUE = on	33057	4	(0x04)
840		**heat pump 4**	bHeatPumpCtrlHeating4	controlled value heating signal of the heatpump 4	R			bool	TRUE = on	33057	5	(0x04)
841		**heat pump 4**	bHeatPumpCtrlRelease4	controlled value operation signal of the heatpump 4	R			bool	TRUE = on	33057	6	(0x04)
842		**heat pump 4**	fHeatPumpCtrlPower4	controlled value of the heatpump 4 power [0..400%]	R	%		uint		33058		(0x04)
843		***ext. run around coil system***	*bRacExtStateMotorProtection*	*error motor protection ext. rac pump*	*R*			*bool*	*TRUE = ok*	33062	0	(0x04)
844		***ext. run around coil system***	*bRacExtStateErrModbusValve*	*modbus comm. error with the ext. rac valve*	*R*			*bool*	*TRUE = ok*	33062	1	(0x04)
845		***ext. run around coil system***	*bRacExtCtrlPump*	*controlled value to switch on/off the ext. rac pump*	*R*			*bool*	*TRUE = on*	33062	2	(0x04)
846		***ext. run around coil system***	*fRacExtStateValve*	*current position of the ext. rac valve*	*R*	*%*		*uint*		33063		(0x04)
847		***ext. run around coil system***	*fRacExtMeaWaterTemp*	*present value of the ext. rac inlet temperature*	*R*	*°C*	*0.1*	*int*		33064		(0x04)
848		***ext. run around coil system***	*fRacExtCtrlValve*	*controlled value of the ext. rac valve*	*R*	*%*		*uint*		33065		(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
849		loop controller fans	nFanControlStrategy	control strategy fans	R			uint	0 = no control; 1 = supply pressure; 2 = extract pressure; 3 = pressure difference; 4 = CO2; 5 = VOC; 6 = supply flow; 7 = extract flow; 8 = extract temperature; 9 = supply and extract pressure; 10 = supply and extract flow; 11 = room temperature; 12 = supply flow and extract pressure; 13 = fan optimizer via X-AIRCONTROL	33071		(0x04)
850	S	loop controller fans	fFanSupSetpointAirflow	setpoint supply air flow	R	m³/h	10	uint		33072		(0x04)
851	S	loop controller fans	fFanSupCurrentValueAirflow	present value supply air flow	R	m³/h	10	uint		33073		(0x04)
852	S	loop controller fans	fFanSupSetpointPressure	setpoint supply air duct pressure	R	Pa		uint		33074		(0x04)
853	S	loop controller fans	fFanSupCurrentValuePressure	present value supply air duct pressure	R	Pa		uint		33075		(0x04)
854		loop controller fans	fFanSupActuatingValue	controlled value of the supply air fan	R	%		uint		33076		(0x04)
855	S	loop controller fans	fFanEtaSetpointAirflow	setpoint extract air flow	R	m³/h	10	uint		33082		(0x04)
856	S	loop controller fans	fFanEtaCurrentValueAirflow	present value extract air flow	R	m³/h	10	uint		33083		(0x04)
857	S	loop controller fans	fFanEtaSetpointPressure	setpoint extract air duct pressure	R	Pa		uint		33084		(0x04)
858	S	loop controller fans	fFanEtaCurrentValuePressure	present value extract air duct pressure	R	Pa		uint		33085		(0x04)
859		loop controller fans	fFanEtaActuatingValue	controlled value of the extract air fan	R	%		uint		33086		(0x04)
860		loop controller temperature	nTemperatureControlStrategy	control strategy temperature	R			uint	0 = no control; 1 = supply; 2 = extract; 3 = supply-extract cascade; 4 = room; 5 = supply-room cascade; 6 = supply via X-AIRCONTROL	33092		(0x04)
861		loop controller temperature	fSupTempSetpointLowLimit	min. supply air temperature setpoint	R	°C	0.1	uint		33093		(0x04)
862		loop controller temperature	fSupTempSetpointHighLimit	max. supply air temperature setpoint	R	°C	0.1	uint		33094		(0x04)
863	S	loop controller temperature	fSupTempSetpointCurrent	current supply air temperature setpoint	R	°C	0.1	uint		33095		(0x04)
864	S	loop controller temperature	fSupTempCurrentValue	present value supply air temperature	R	°C	0.1	uint		33096		(0x04)
865		loop controller temperature	fEtaTempSetpointLowLimit	min. extract air temperature setpoint	R	°C	0.1	uint		33097		(0x04)
866		loop controller temperature	fEtaTempSetpointHighLimit	max. extract air temperature setpoint	R	°C	0.1	uint		33098		(0x04)
867	S	loop controller temperature	fEtaTempSetpointCurrent	current extract air temperature setpoint	R	°C	0.1	uint		33099		(0x04)
868	S	loop controller temperature	fEtaTempCurrentValue	present value extract air temperature	R	°C	0.1	uint		33100		(0x04)
869		loop controller temperature	fRoomTempSetpointLowLimit	min. room air temperature setpoint	R	°C	0.1	uint		33101		(0x04)
870		loop controller temperature	fRoomTempSetpointHighLimit	max. room air temperature setpoint	R	°C	0.1	uint		33102		(0x04)
871	S	loop controller temperature	fRoomTempSetpointCurrent	current room air temperature setpoint	R	°C	0.1	uint		33103		(0x04)
872	S	loop controller temperature	fRoomTempCurrentValue	present value room air temperature	R	°C	0.1	uint		33104		(0x04)
873		loop controller temperature	fTempMeanActuatingvalue	mean value of actuating values temperature control	R	%		uint		33105		(0x04)

Data point list Modbus/TCP X-CUBE Control

TROX® TECHNIK
The art of handling air

V006 - 09/2021

D: important data points
S: system specific data points

default IP address: 192.168.0.180 OR 192.168.0.200
Modbus/TCP port: 502

no.	D/S	group	datapoint	description	R/W	unit	scale	data type	values	register	bit	function code
874		***loop controller humidity***	*nHumidityControlStrategy*	*control strategy humidity*	*R*			*uint*	*0 = no control; 1 = supply; 2 = extract; 3 = supply-extract cascade; 4 = room; 5 = supply-room cascade;*	33111		(0x04)
875		***loop controller humidity***	*fSupHumSetpointLowLimit*	*min. supply air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33112		(0x04)
876		***loop controller humidity***	*fSupHumSetpointHighLimit*	*max. supply air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33113		(0x04)
877	***S***	***loop controller humidity***	*fSupHumSetpointCurrent*	*current supply air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33114		(0x04)
878	***S***	***loop controller humidity***	*fSupHumCurrentValue*	*present value supply air humidity*	*R*	*g/kg*	*0.01*	*uint*		33115		(0x04)
879		***loop controller humidity***	*fEtaHumSetpointLowLimit*	*min. extract air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33116		(0x04)
880		***loop controller humidity***	*fEtaHumSetpointHighLimit*	*max. extract air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33117		(0x04)
881	***S***	***loop controller humidity***	*fEtaHumSetpointCurrent*	*current extract air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33118		(0x04)
882	***S***	***loop controller humidity***	*fEtaHumCurrentValue*	*present value extract air humidity*	*R*	*g/kg*	*0.01*	*uint*		33119		(0x04)
883		***loop controller humidity***	*fRoomHumSetpointLowLimit*	*min. room air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33120		(0x04)
884		***loop controller humidity***	*fRoomHumSetpointHighLimit*	*max. room air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33121		(0x04)
885	***S***	***loop controller humidity***	*fRoomHumSetpointCurrent*	*current room air humidity setpoint*	*R*	*g/kg*	*0.01*	*uint*		33122		(0x04)
886	***S***	***loop controller humidity***	*fRoomHumCurrentValue*	*present value room air humidity*	*R*	*g/kg*	*0.01*	*uint*		33123		(0x04)
887		***loop controller humidity***	*fHumMeanActuatingvalue*	*mean value of actuating values humidity control*	*R*	*%*		*uint*		33124		(0x04)
888	***S***	***loop controller air quality***	*fAirQualitySetpoint*	*setpoint air quality*	*R*	*ppm*		*uint*		33130		(0x04)
889	***S***	***loop controller air quality***	*fAirQualityCurrentValue*	*present value air quality*	*R*	*ppm*		*uint*		33131		(0x04)
890		***loop controller air quality***	*fAirQualityActuatingValueRCA*	*Actuating value recirculation air damper*	*R*	*%*		*uint*		33132		(0x04)
891		***loop controller air quality***	*fAirQualityActuatingValueSupFan*	*Actuating value supply air fan*	*R*	*m³/h*	*10*	*uint*		33133		(0x04)
892		***loop controller air quality***	*fAirQualityActuatingValueEtaFan*	*Actuating value extract air fan*	*R*	*m³/h*	*10*	*uint*		33134		(0x04)