



# Manual

# IPEmotion PlugIn CAETEC dataLog



V18.02.00

Date of issue: December 5, 2018

© Copyright 2007 - 2018 by:

CAETEC GmbH  
Industriestrasse 1  
82140 Olching  
Germany

All rights reserved. Any reprinting, photocopying or translation of this manual, in whole or in part, requires advance written approval of CAETEC.

Pictures and sketches are for illustration purposes only and are not to be used as design drawings nor to serve as offer or assembly drawings.

All specifications are based on the technical status of December 5, 2018. We reserve the right to make any changes required to technically improve the equipment.

This manual has been produced with all due diligence.

CAETEC shall not be held liable for any damage resulting from the use of this manual, providing it is not due to gross negligence on our own part or the part of our legal representative or vicarious agent, and to the extent that the damage does not stem from personal injury, bodily harm or damage to health.

All related registered brands and trademarks are the property of the respective owners.

# Contents

<b>1</b>	<b>Foreword</b>	<b>13</b>
<b>2</b>	<b>Configuration</b>	<b>14</b>
2.1	Symbols . . . . .	14
2.2	References . . . . .	14
<b>3</b>	<b>Product description</b>	<b>15</b>
3.1	Installation . . . . .	15
3.1.1	System requirements . . . . .	15
3.1.2	Where to get the installation file? . . . . .	15
3.1.3	How to know the right version? . . . . .	15
3.1.4	Installation on Windows . . . . .	16
3.2	User interface . . . . .	19
3.2.1	Menu bar . . . . .	19
3.2.2	"File" menu . . . . .	20
3.2.3	Working with the Ribbon . . . . .	29
3.2.4	"Signals" tab . . . . .	32
3.2.5	Quick Access Toolbar . . . . .	33
3.2.6	Message area . . . . .	33
<b>4</b>	<b>Setting up a logger system</b>	<b>34</b>
4.1	Choosing a logger-system . . . . .	34
4.2	The measurement task workspace . . . . .	35
4.2.1	The measurement task tree . . . . .	35
4.2.2	The details area . . . . .	36
4.2.3	The grid area . . . . .	39
4.3	Plugin version . . . . .	39
4.3.1	Column chooser . . . . .	40
4.3.2	Filter editor . . . . .	41
4.4	Changing the logger system with the <i>change into</i> -command . . . . .	42
4.5	Importing and exporting the <i>datalog.cmc</i> . . . . .	43
4.5.1	Ignore errors and warnings at export . . . . .	44
4.6	Online communication with the logger . . . . .	45
4.6.1	Communication settings . . . . .	45
4.6.2	Online functions . . . . .	47
4.6.3	Licence information . . . . .	48
4.6.4	Licence check . . . . .	48
4.7	Changing a system's default tree elements . . . . .	50
<b>5</b>	<b>Project settings</b>	<b>52</b>
5.1	Adding project parameters . . . . .	52
5.2	Assigning a template of project parameters . . . . .	53
5.3	Tree elements for Project settings . . . . .	54
5.4	Grid area for Project settings . . . . .	54
5.5	Details area for Project settings . . . . .	55
5.6	Using project parameters as variables in CAETEC <i>dataLog</i> Plugin . . . . .	56

<b>6</b>	<b>UPS (Uninterruptible power supply)</b>	<b>58</b>
6.1	Storage method	58
6.2	Adding the UPS interface	59
6.3	Configuring the UPS interface	59
6.3.1	Tree elements for the UPS interface	59
6.3.2	Details area for the UPS interface	59
6.4	UPS signal properties	61
6.4.1	Grid area for UPS signals	61
6.4.2	Overview of UPS signals	61
6.4.3	Details area for UPS signals	62
<b>7</b>	<b>Drivebay (Arcos 1.5 only)</b>	<b>66</b>
7.1	Adding the Drivebay interface	66
7.2	Tree elements for Drivebay	66
7.3	Details area for Drivebay	67
<b>8</b>	<b>Signal Acquisition</b>	<b>68</b>
8.1	CAN/CAN FD channels	68
8.1.1	Storage method	68
8.1.2	Adding CAN/CAN FD channels	69
8.1.3	CAN settings	70
8.1.3.1	General	70
8.1.3.2	CAN	71
8.1.3.3	Bit timing	72
8.1.3.4	Wake On CAN	72
8.1.3.5	Hardware (Channel number)	79
8.1.4	Virtual CAN settings	80
8.1.4.1	General	80
8.1.4.2	Hardware (Channel number)	81
8.1.5	CAN channel Bus statistic	81
8.1.5.1	Adding Bus statistics	81
8.1.5.2	Bus statistic signals	82
8.2	CAN signals	83
8.2.1	Storage method	83
8.2.2	Importing CAN signals	83
8.2.3	Import properties	86
8.2.4	Signal properties	88
8.2.4.1	Tree elements for CAN signals	88
8.2.4.2	Grid area for CAN signals	89
8.2.4.3	Details area for CAN signals	89
8.3	CCP/XCP signals	95
8.3.1	Storage method	95
8.3.2	Importing CCP/XCP signals	95
8.3.3	Import properties	98
8.3.4	Signal properties	99
8.3.4.1	Tree elements for CCP/XCP signals	99
8.3.4.2	Grid area for CCP/XCP signals	102
8.3.4.3	Details area for CCP/XCP signals	102
8.4	UDS signals	111
8.4.1	Storage method	111



8.4.2	Importing UDS signals	111
8.4.3	Import properties	113
8.4.4	Signal properties	114
8.4.4.1	Tree elements for UDS signals	114
8.4.4.2	Grid area for UDS signals	115
8.4.4.3	Details area for UDS signals	115
8.5	OBD signals	120
8.5.1	Storage method	120
8.5.2	Adding the OBD signals interface	120
8.5.3	User-defined OBD signals	120
8.5.4	Signal properties	121
8.5.4.1	Tree elements for OBD signals	121
8.5.4.2	Grid area for OBD signals	121
8.5.4.3	Details area for OBD signals	122
8.6	Gateways	129
8.6.1	Adding a gateway	129
8.6.2	Adding an ID filter	129
8.6.3	Gateway settings	130
8.6.3.1	Tree elements for Gateways	130
8.6.3.2	Grid area for Gateways	130
8.6.3.3	Details area for Gateways	130
8.6.4	ID filter settings	131
8.7	Runstate	132
8.7.1	Add Runstate	132
8.7.2	Tree elements for Runstate	133
8.7.3	Grid area for Runstate	133
8.7.4	Details area for Runstate	134
8.7.5	Export Runstate	134
8.8	ETH channels	135
8.8.1	Storage method	135
8.8.2	Adding ETH channels	136
8.8.3	ETH settings	137
8.8.3.1	General	138
8.8.3.2	LAN	138
8.8.3.3	Settings	138
8.8.4	ETH channel Bus statistic	139
8.8.4.1	Adding Bus statistics	139
8.8.4.2	Bus statistic signals	139
8.9	ETH signals	141
8.9.1	Storage method	141
8.9.2	Importing ETH signals	141
8.9.2.1	Importing Fibex files (SOME/IP)	141
8.9.2.2	Importing A2L files (XCPonUDP)	142
8.9.2.3	IP settings when Importing A2L files (XCPonUDP)	144
8.9.3	Signal properties	146
8.9.3.1	Signal properties for SOME/IP	146
8.9.3.2	Signal properties for XCPonUDP	146
8.10	LIN channels	147
8.10.1	Storage method	147
8.10.2	Adding LIN channels	148

8.10.3	LIN settings	148
8.10.3.1	General	149
8.10.3.2	LIN	150
8.10.3.3	Wake On LIN	150
8.10.3.4	Hardware (Channel number)	151
8.10.4	LIN channel Bus statistic	151
8.11	LIN signals	152
8.11.1	Storage method	152
8.11.2	Importing LIN signals	152
8.11.3	Import properties	154
8.11.4	Signal properties	155
8.11.4.1	Tree elements for LIN signals	155
8.11.4.2	Grid area for LIN signals	156
8.11.4.3	Details area for LIN signals	157
8.12	FlexRay channels	162
8.12.1	Storage method	162
8.12.2	Adding FlexRay channels	163
8.12.3	FlexRay settings	163
8.12.3.1	General	164
8.12.3.2	Wake On FlexRay	164
8.12.3.3	Hardware (Channel number)	165
8.12.4	FlexRay channel Bus statistic	166
8.12.4.1	Adding Bus statistics	166
8.12.4.2	Bus statistic signals	167
8.13	FlexRay signals	168
8.13.1	Storage method	168
8.13.2	Importing FlexRay signals	168
8.13.2.1	Importing Autosar and Fibex files	168
8.13.2.2	Importing A2L files (XCP on FlexRay)	171
8.13.3	Import properties	173
8.13.4	Signal properties	175
8.13.4.1	Tree elements for FlexRay signals	175
8.13.4.2	Grid area for FlexRay signals	176
8.13.4.3	Details area for FlexRay signals	176
8.14	GPS Signals	182
8.14.1	Storage method	182
8.14.2	Adding GPS Signals	182
8.14.2.1	CAETEC GPS module	182
8.14.2.2	Other GPS signals (Assigning GPS signals)	183
8.14.3	Signal properties	184
8.14.3.1	Tree elements for GPS signals	184
8.14.3.2	Grid area for GPS signals	184
8.14.3.3	Details area for GPS signals	184
8.15	Video devices	189
8.15.1	Storage method	189
8.15.2	Video Interface	189
8.15.2.1	Adding the Video Interface	190
8.15.2.2	Tree elements for the Video Interface	190
8.15.2.3	Grid area for the Video Interface	191
8.15.2.4	Details area for the Video Interface	192

8.15.3	USB camera	194
8.15.3.1	Adding a USB camera	194
8.15.3.2	Tree elements for USB camera	195
8.15.3.3	Grid area for USB camera	195
8.15.3.4	Details area for USB camera	195
8.15.4	Ethernet camera	198
8.15.4.1	Adding an ETH camera	198
8.15.4.2	Tree elements for ETH camera	201
8.15.4.3	Grid area for ETH camera	201
8.15.4.4	Details area for ETH camera	201
8.16	Video signals	204
8.16.1	Storage method	204
8.16.2	Settings for video signals	204
8.17	Audio recording	206
8.17.1	Storage method	206
8.17.2	Adding an Audio recording	206
8.17.3	Tree elements for Audio recordings	207
8.17.3.1	Grid area for Audio recordings	208
8.17.4	Details area for Audio recording	209
8.17.5	Microphone settings	212
8.17.5.1	Signals	212
8.17.5.2	LEDs	212
8.17.5.3	Buttons	213
8.18	DIN (Digital input signals)	214
8.18.1	Storage method	214
8.18.2	Adding the DIN-Interface	214
8.18.3	Signal properties	215
8.18.3.1	Tree elements for DIN signals	216
8.18.3.2	Grid area for DIN signals	216
8.18.3.3	Details area for DIN signals	216
8.19	DOUT (Digital output signals)	221
8.19.1	Storage method	221
8.19.2	Adding the DOUT-Interface	221
8.19.3	Signal properties	222
8.19.3.1	Tree elements for DOUT signals	223
8.19.3.2	Grid area for DOUT signals	223
8.19.3.3	Details area for DOUT signals	223
8.20	Analog signals	227
8.20.1	Storage method	227
8.20.2	Adding the Analog Interface	228
8.20.3	Signal properties	231
8.20.3.1	Tree elements for Analog signals	231
8.20.3.2	Grid area for Analog signals	231
8.20.3.3	Details area for Analog signals (Voltage)	231
8.20.3.4	Details area for Analog signals (Counter/frequency)	235
8.20.3.5	Details area for Analog signals (Duty cycle)	239
8.20.4	Analog signals on $\mu$ Cros SL	243
8.21	Thermo	245
8.21.1	Storage method	245
8.21.2	Adding the Thermo-Interface	245

8.21.3	Signal properties	247
8.21.3.1	Tree elements for Thermo signals	247
8.21.3.2	Grid area for Thermo signals	247
8.21.3.3	Details area for Thermo signals	247
8.22	Internal signals	252
8.22.1	Storage method	252
8.22.2	Accessing internal signals	253
8.22.3	Internal signals properties	254
8.22.3.1	Tree elements for Internal signals	254
8.22.3.2	Grid area for Internal signals	254
8.22.3.3	Details area for Internal signals	254
8.22.4	Run state	258
8.22.5	System info	260
8.22.6	Time	262
8.23	Formulas	263
8.23.1	Formulas (Calculated signals)	263
8.23.1.1	Storage method	264
8.23.1.2	Adding a formula	265
8.23.1.3	Grid area for formulas	265
8.23.1.4	Details area for formulas	266
8.23.2	Script expressions	273
8.23.2.1	Adding a Script expression	273
8.23.2.2	Script expression editor	274
8.24	Synchronizing signals	278
8.25	Transferring measurement tasks to the logger	280
<b>9</b>	<b>Triggers</b>	<b>281</b>
9.1	Adding a trigger	282
9.2	Tree elements for triggers	283
9.3	Grid area for Triggers	284
9.4	Details area for Triggers	285
9.5	Standard Triggers	286
9.6	Level Triggers	287
9.7	Cyclic Triggers	290
9.8	Trigger groups	291
<b>10</b>	<b>Scripts</b>	<b>293</b>
10.1	Adding the Scripts-Interface	293
10.2	Adding a script	294
10.3	Importing a script	294
10.4	Tree elements for Scripts	295
10.5	Grid area for Scripts (Composing a script)	296
10.5.0.1	Edit script code	297
10.5.0.2	Prefabricated code blocks	298
10.5.0.3	Syntax check	299
10.5.1	Triggers in Scripts	300
10.5.2	Signals in Scripts	300
10.5.3	Methods in Scripts	301
10.6	Details area for Scripts	301
10.7	Exporting a script	302

<b>11 Includes</b>	<b>303</b>
11.1 Adding the Includes-Interface	303
11.2 Tree elements for Includes	304
11.3 Grid area for Includes	304
11.4 Details area for Includes	304
<b>12 External files</b>	<b>306</b>
12.1 Automatically add external files	306
12.2 Adding the External files interface	308
12.3 Adding an external file	308
12.4 Tree elements for External files	309
12.5 Grid area for External files	309
12.6 Details area for External files	310
<b>13 Surveillance</b>	<b>311</b>
13.1 Displays	311
13.1.1 Adding a display	311
13.1.2 The "Displays" interface	313
13.1.3 CAETEC Display-specific settings	314
13.1.4 openABK Display-specific settings	315
13.1.5 General Display settings	316
13.1.5.1 Tree elements for a Display	316
13.1.5.2 Grid area for a Display	316
13.1.5.3 Details area for a Display	317
13.1.5.4 Signals for Display	318
13.1.5.5 Buttons for Display	320
13.1.5.6 Messages for Display	321
13.2 E-mails	323
13.2.1 Setting up the E-mails interface	323
13.2.1.1 Adding the E-mails interface	323
13.2.1.2 Configure SMTP	324
13.2.2 Composing e-mails	325
13.2.2.1 Creating a new e-mail	325
13.2.2.2 Tree elements for E-mails	325
13.2.2.3 Grid area for E-mails	326
13.2.2.4 Details area for E-mails (Composing)	326
13.2.3 E-mail attachments	328
13.2.3.1 Signal attachments	328
13.2.3.2 Datafile attachments	330
13.2.3.3 Logfile attachments	331
13.2.3.4 Dataset attachments	332
13.3 Log file messages	334
13.3.1 Adding the Log file messages interface	334
13.3.2 Create a new Log file messages interface	334
13.3.3 Composing Log file messages	335
13.3.3.1 Grid area for Log file messages	335
13.3.3.2 Details area for Log file messages (Composing)	336
13.4 Monitoring	338
13.4.1 Tree elements for Monitoring	338
13.4.2 Booleans	339

13.4.2.1	Adding Booleans . . . . .	339
13.4.2.2	Grid area for Booleans . . . . .	339
13.4.2.3	Details area for Booleans . . . . .	339
13.4.3	Limit value . . . . .	343
13.4.3.1	Adding a limit value . . . . .	343
13.4.3.2	Grid area for Limit values . . . . .	343
13.4.3.3	Details area for Limit values . . . . .	344
13.4.4	Range . . . . .	347
13.4.4.1	Adding a Range . . . . .	347
13.4.4.2	Grid area for Ranges . . . . .	347
13.4.4.3	Details area for Ranges . . . . .	348
13.5	XCP slave . . . . .	351
13.5.1	Adding XCP slave . . . . .	351
13.5.2	Tree elements for XCP slave . . . . .	351
13.5.3	Grid area for XCP slave . . . . .	351
13.5.4	Details area for XCP slave . . . . .	352
<b>14</b>	<b>Datasets</b>	<b>354</b>
14.1	Dataset . . . . .	354
14.1.1	Adding extra datasets . . . . .	354
14.1.2	Tree elements for Datasets . . . . .	354
14.1.3	Details area for Datasets . . . . .	355
14.1.4	Setting up a dataset . . . . .	360
14.2	Ring buffer . . . . .	362
14.2.1	Adding a ring buffer . . . . .	362
14.2.2	Setting up a ring buffer . . . . .	362
14.2.3	Tree elements for ring buffer . . . . .	363
14.2.4	Grid area for ring buffer . . . . .	363
14.2.5	Details area for Ring buffer . . . . .	363
14.3	Dataset Project settings . . . . .	366
14.3.1	Adding project parameters . . . . .	366
14.3.2	Assigning a template of project parameters . . . . .	367
14.3.3	Tree elements for Project settings . . . . .	367
14.3.4	Grid area for Project settings . . . . .	368
14.3.5	Details area for Project settings . . . . .	368
14.4	Includes . . . . .	370
14.4.1	Adding the Includes-Interface . . . . .	370
14.4.2	Tree elements for Includes . . . . .	370
14.4.3	Grid area for Includes . . . . .	371
14.4.4	Details area for Includes . . . . .	371
14.5	ATFX . . . . .	372
14.5.1	Tree elements for ATFX . . . . .	372
14.5.2	Grid area for ATFX . . . . .	372
14.5.3	Details area for ATFX . . . . .	373
14.5.3.1	ATFX file . . . . .	373
14.5.3.2	ATFX Timelog . . . . .	375
14.5.3.3	ATFX Signal Group . . . . .	377
14.5.4	Working with Signal Groups for ATFX . . . . .	379
14.6	MDF 4.0 . . . . .	381
14.6.1	Multiple timelogs in MDF 4.0 . . . . .	381



14.6.2	Bus trace in MDF 4.0	382
14.6.3	Tree elements for MDF 4.0	382
14.6.4	Grid area for MDF 4.0	383
14.6.5	Details area for MDF 4.0	383
14.6.5.1	MDF 4.0 File	383
14.6.5.2	MDF 4.0 Timelog	386
14.6.5.3	MDF 4.0 Signal Group	388
14.6.6	Working with Signal Groups for MDF 4.0	390
14.7	MDF 4.1	392
14.7.1	File compression in MDF 4.1	392
14.7.2	Header profiles in MDF 4.1	392
14.7.2.1	Comments reference	393
14.7.2.2	Overview of header profiles and their differences	393
14.7.3	Video attachments in MDF 4.1	395
14.7.3.1	Attaching a video	395
14.7.3.2	Details area for video in MDF 4.1	396
14.8	Vector BLF / Vector ASCII / Vector ASCII compressed	398
14.8.1	Tree elements for bus tracing	398
14.8.2	Grid area for bus tracing	399
14.8.3	Details area for bus tracing	400
14.8.3.1	Bus tracing file	400
14.8.3.2	Bus trace	402
14.8.3.3	Traceable Bus channel	404
14.8.4	Bus trace ID Filter	405
14.8.5	Details area for bus tracing (Ring buffer)	407
14.8.5.1	Bus tracing file	407
14.8.5.2	Bus trace	409
14.8.5.3	Traceable Bus channel	411
14.9	PCAP	413
14.9.1	Tree elements for PCAP	413
14.9.2	Grid area for PCAP	414
14.9.3	Details area for PCAP	414
14.9.3.1	PCAP file	414
14.9.3.2	Eth trace	417
14.9.4	Details area for PCAP (Ring buffer)	420
14.9.4.1	PCAP file	420
14.9.4.2	ETH trace	422
14.9.4.3	Traceable ETH channel	424
14.10	AVI	425
14.10.1	Including a video signal in the Video Stream	425
14.10.2	Tree elements for AVI	426
14.10.3	Grid area for AVI	427
14.10.4	Details area for AVI	427
14.10.4.1	AVI File	427
14.10.4.2	Video	430
14.10.4.3	Video Stream	432
14.10.5	Details area for AVI (Ring buffer)	433
14.10.5.1	AVI File	433
14.10.5.2	Video	435
14.10.5.3	Video Stream	437

14.11 WAV	438
14.11.1 Including an audio signal in the audio Stream	438
14.11.2 Tree elements for WAV	439
14.11.3 Grid area for WAV	440
14.11.4 Details area for WAV	440
14.12 GPX	442
14.12.1 Assigning GPS signals	442
14.12.2 Tree elements for GPX	442
14.12.3 Grid area for GPX	443
14.12.4 Details area for GPX	443
14.12.4.1 GPX File	443
14.12.4.2 GPS Tracking	444
14.13 CAETEC binary (Classings)	447
14.13.1 Tree elements for CAETEC binary	447
14.13.2 Details area for CAETEC binary	448
14.13.3 Adding a classing	449
14.14 CAETEC ASCII (Classings)	450
14.14.1 Tree elements for CAETEC ASCII	450
14.14.2 Details area for CAETEC ASCII	451
14.14.3 Adding a classing	452
14.15 Classing methods	453
14.16 Script file	454
14.16.1 Including a Script file as a target in a script	454
14.16.2 Tree elements for Script file	456
14.16.3 Details area for Script file	456
<b>15 Datatransfer</b>	<b>458</b>
15.1 Transfer events	458
15.1.1 General Information about transfer events	459
15.1.2 Export order	460
15.1.3 Transfer priority	460
15.1.4 Trigger events	461
15.1.4.1 Tree elements for Trigger events	463
15.1.4.2 Grid area for Trigger events	463
15.1.4.3 Details area for Trigger events	464
15.1.5 Time events	465
15.1.5.1 Tree elements for Time events	465
15.1.5.2 Grid area for Time events	466
15.1.5.3 Details area for Time events	466
15.1.6 System events	467
15.1.6.1 Tree elements for System events	468
15.1.6.2 Grid area for System events	468
15.1.6.3 Details area for System events	468
15.2 Transfer event targets	469
15.2.1 Tree elements for transfer event targets	469
15.2.2 Grid area for transfer event targets	469
15.2.3 Details area for transfer event targets	470
15.3 Transfer connections	471
15.3.1 Data transfer via USB	471
15.3.1.1 Details area for USB	471

---

15.3.2	Data transfer via WIFI	473
15.3.2.1	Tree elements for WIFI connections	474
15.3.2.2	Grid area for WIFI connections	474
15.3.2.3	Details area for WIFI	475
15.3.3	Data transfer via LAN	479
15.3.3.1	Tree elements for LAN connections	479
15.3.3.2	Details area for LAN	479
15.3.4	Data transfer via PPP/UMTS	481
15.3.4.1	Setting up a PPP/UMTS connection	481
15.3.4.2	Details area for PPP/UMTS	482
15.3.4.3	PPP status signals	483
15.3.5	Wake on Call/Text	485
15.3.5.1	Tree elements for Wake on Call/Text	486
15.3.5.2	Grid area for Wake on Call/Text	486
15.3.5.3	Details area for Wake on Call/Text	486
15.3.6	Setting up a Fileserver	488
15.3.6.1	Multiple File servers	489
15.3.6.2	Tree elements for File servers	489
15.3.6.3	Grid area for File servers	490
15.3.6.4	Details area for File servers	490
16	Setting up a time server	492
17	Obtaining extended support	493

# 1 Foreword

## 2 Configuration

### 2.1 Symbols

Various paragraphs in this manual are marked with special symbols. These symbols have the following meanings:



This symbol highlights important information that, if ignored, may prevent successful use of the program.



This symbol refers to additional information supplementing this manual.

### 2.2 References

References to other sections of this manual are generally placed in brackets and are indicated by an arrow:

(→ [2.1](#)) refers to Section 2.1.

When this manual is read in digital form, a mouse-click on such a reference accesses the particular section of the book.

## 3 Product description

### 3.1 Installation

#### 3.1.1 System requirements

**Minimum:**

Screen resolution: 1080 x 800 pixel  
Processor: 2 GHz  
RAM: 2048 MB  
DirectX 9

**Recommended:**

Screen resolution: 1920 x 1200 pixel  
Processor: 3 GHz Multi-Core  
RAM: 6144 MB  
Storage medium type: SSD  
DirectX 11

**Supported platforms:**

Microsoft Windows 10 (32 Bit and 64 Bit operating systems)  
Microsoft Windows 8.1 (32 Bit and 64 Bit operating systems)  
Microsoft Windows 8 (32 Bit and 64 Bit operating systems)  
Microsoft Windows 7 (32 Bit and 64 Bit operating systems)

**The following additional software is required:**

Microsoft .NET 4.5.1 Framework

#### 3.1.2 Where to get the installation file?

The installation file can be downloaded from <https://myipe.ipetronik.com/> or here <https://www.ipetronik.com/software/plugins>. Once you have logged in with your username and password you will be able to locate the file in the section **UP- & DOWNLOADS**.

#### 3.1.3 How to know the right version?

The format of the Plugin version is always Vxx.xx.xx (e.g. V15.10.00) and has to match the target system's dataLog firmware-version. To find out which firmware-version your target system is running you can either use the the data logger's web interface or open with a text editor the logger's logfile and look for the firmware version there. For further information please also refer to your data logger's documentation.



Multiple Plugin versions of the CAETEC Plugin for IPEmotion can be installed at the same time. This allows you to work with various data loggers that are running on different firmware versions. Once you have multiple Plugin versions installed, you will always have to ensure that your currently used Plugin version and target system's firmware version match. Please refer to **Switching between Plugin versions** (→ [3.2.2](#)).



## 3.1 INSTALLATION

### 3.1.4 Installation on Windows

Navigate to your download directory and extract the downloaded zip-file. For 32-bit versions of Microsoft Windows choose the following file for installation.

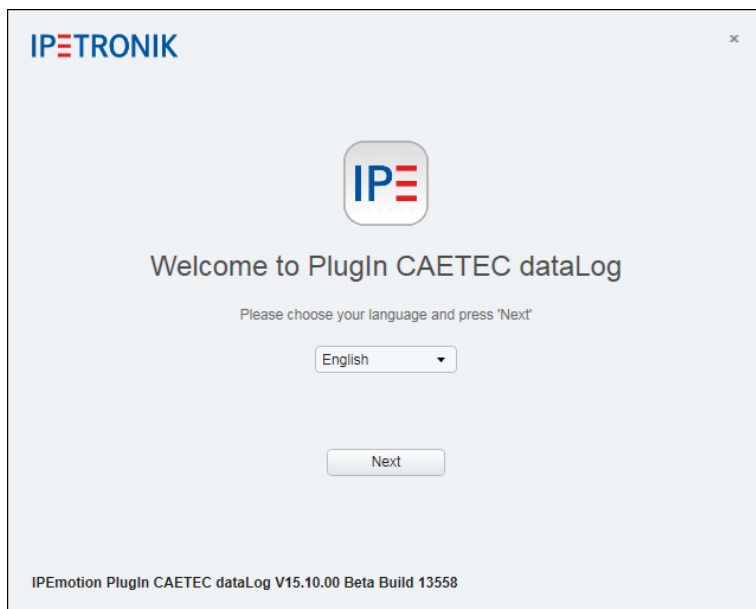
 Setup IPEmotion PlugIn CAETEC dataLog.exe	30.10.2017 09:21	Anwendung	32.090 KB
-----------------------------------------------------------------------------------------------------------------------------	------------------	-----------	-----------

For 64-bit versions of Microsoft Windows choose the following file for installation.

 Setup IPEmotion PlugIn CAETEC dataLog (x64).exe	30.10.2017 09:21	Anwendung	25.966 KB
-----------------------------------------------------------------------------------------------------------------------------------	------------------	-----------	-----------

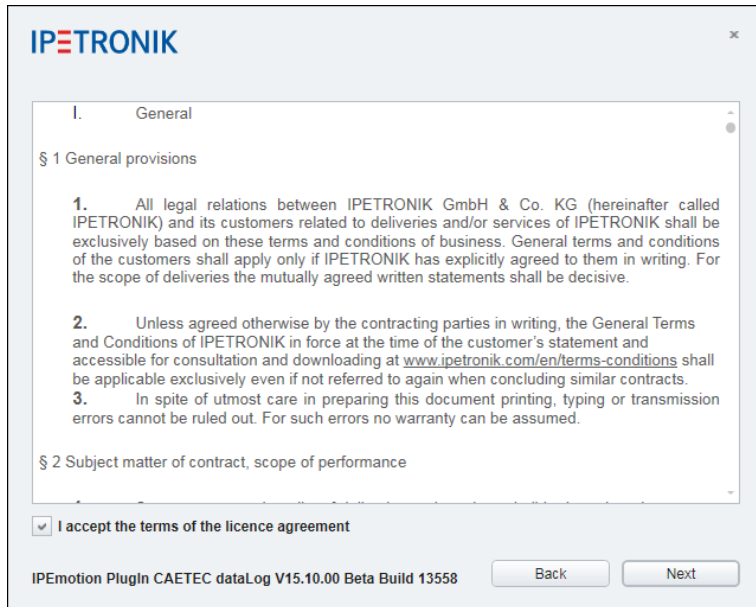
You may be asked to confirm execution of the program. If so, please confirm and type in your Windows user password if asked to do so.

In the next window you may choose the language in which you wish to install the plugin. The language can be changed later on (→ [3.2.2](#)).

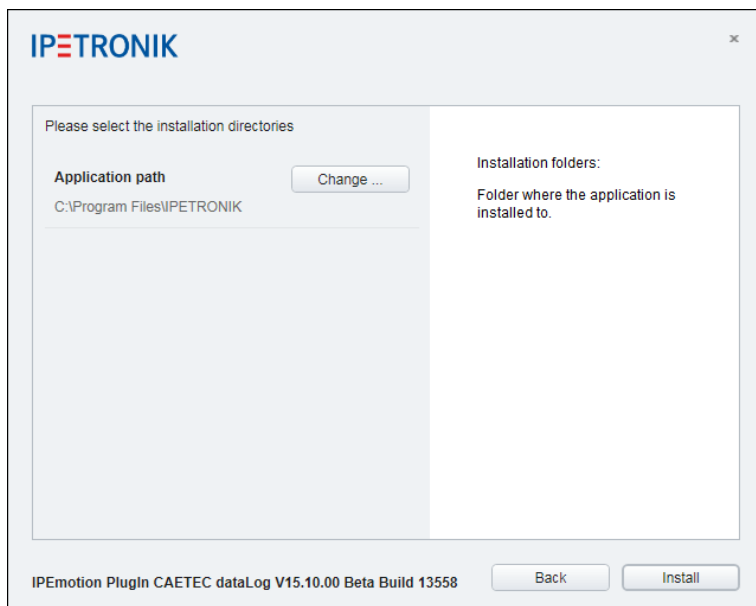


### 3.1 INSTALLATION

The next window presents the terms of the licence agreement. Check the box **I accept the terms of the licence agreement** and proceed to the next window by clicking next.



In the next window you select the target folder where the plugin is to be installed. The standard installation path is set as default. Normally you can simply accept it. Click Install to continue and start the installation process. Again you may be asked for confirmation and/or your Windows user or admin password. Please type it in and click OK to continue.



Once installation has been completed successfully you will need to activate the Plugin in the Options dialogue in order to start working with it. To do so please refer to the section **Activating the plugin** (→3.2.2).



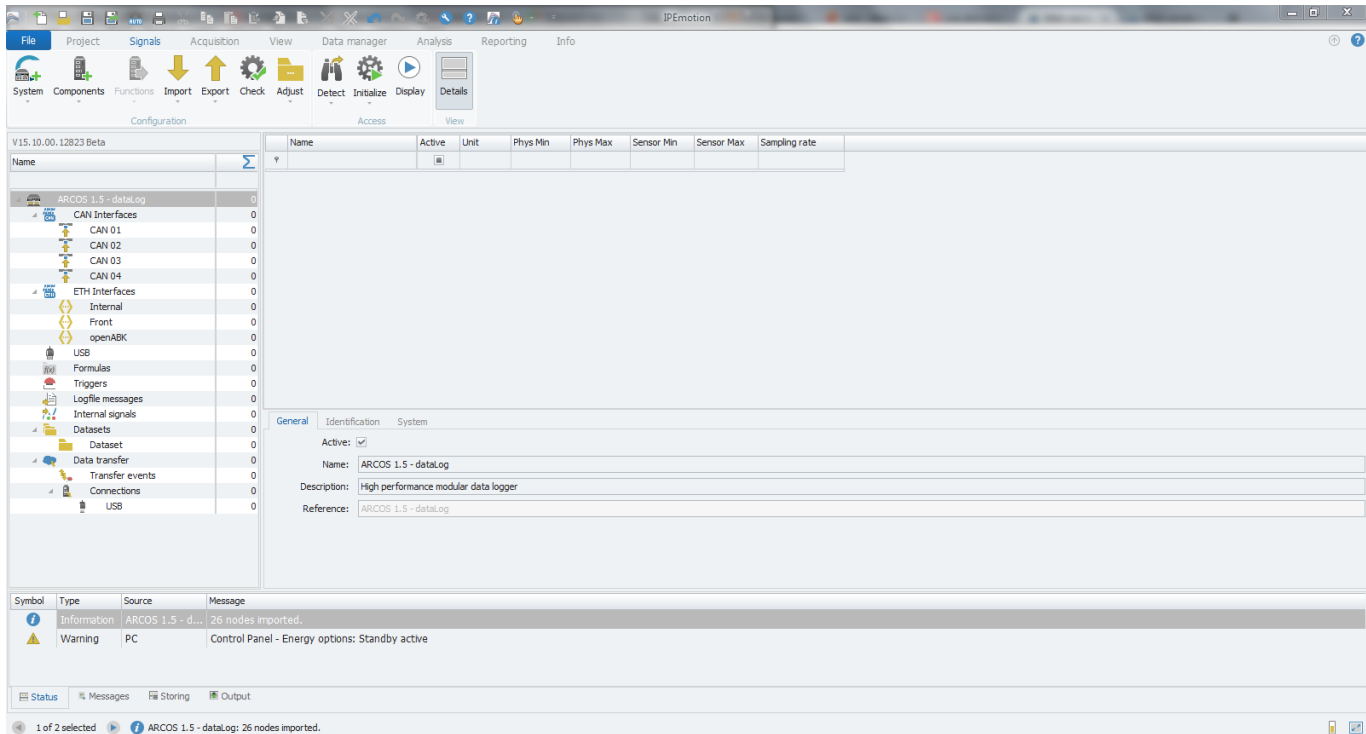
After completion of the installation we strongly advise you to activate the expert mode and afterwards activate the extended tabs option and change the number of maximum polling lists in the expert settings. For detailed information refer to the sections **Expert mode**, **Extended tabs** and **Maximum polling lists** (→ [3.2.2](#)).



The plugin allows for a customizable user guidance by giving you the option to choose which tree elements you would like to be available in the measurement task tree. For detailed information refer to the section **Customizing tree elements** (→ [3.2.2](#)).

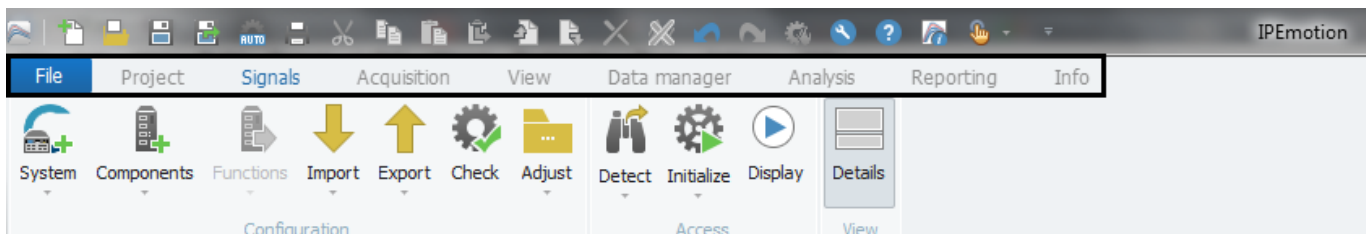
### 3.2 User interface

This section describes the general appearance and functionality of the CAETEC Plugin for IPEmotion. However it will not describe specifics on how to create a measurement task or on the single elements available to configure your data logger. For information on these topics please refer to Chapter (→ 8).



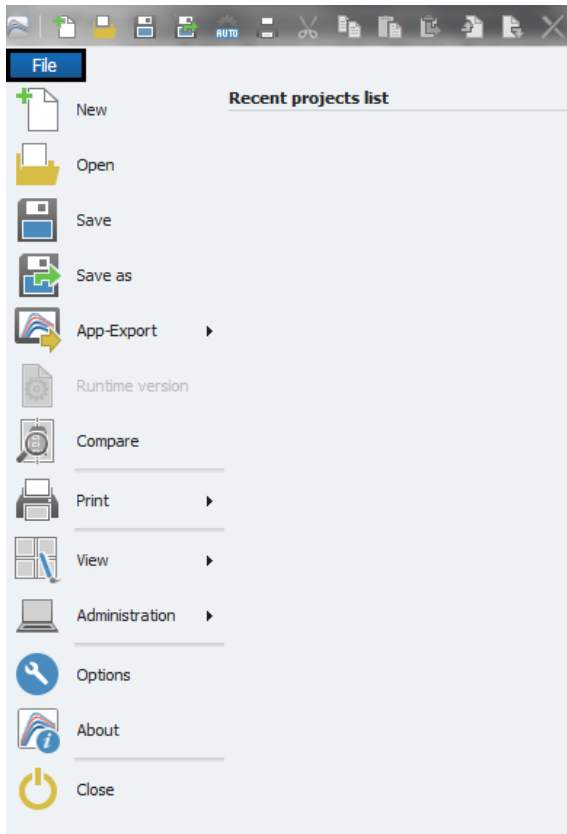
#### 3.2.1 Menu bar

The menu bar provides the core functionality of IPEmotion. However when working with the CAETEC Plugin you will only need to make use of the "File" menu and the "Signals" tab, as the Plugin offers a closed working environment which includes all functionality needed in order to work with your data logger inside the "Signals"-tab.



### 3.2.2 “File” menu

A dropdown menu with basic IPEmotion functionality. Only the item relevant for working with the CAETEC Plugin for IPEmotion will be explained here. For information on items not explained here, please refer to the IPEmotion documentation.



#### New

Creates a new configuration/project.

#### Open

Opens a previously saved configuration/project. You can open three different file-types. The **IPEmotion configuration file (\*.iwf)** is a container holding the entire IPEmotion-Project and its settings. The **IPEmotion acquisition configuration file (\*.iac)** holds only the configuration contained in the “Signals” tab. The **System file (\*.isf)** is currently not supported by this plugin.

#### Save

Saves the current configuration/project.

#### Save as

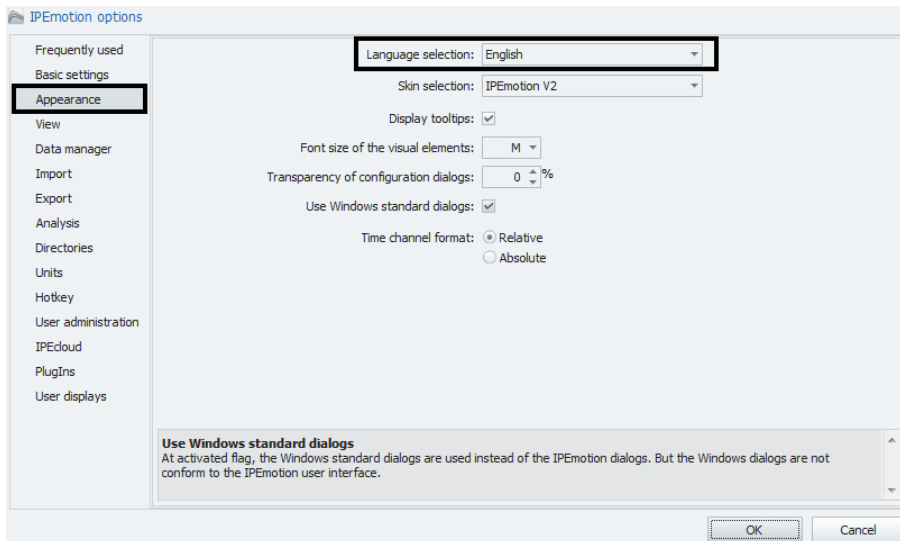
Save the current configuration/project under a new name and/or in a new location. You can choose between two different file-types to save your projects, which will determine the information your saved file will hold. The **IPEmotion configuration file (\*.iwf)** is a container holding the entire IPEmotion-Project and its settings. The **IPEmotion acquisition configuration file (\*.iac)** holds only the configuration contained in the “Signals” tab.

## Options

Opens a window with options that affect the behaviour of IPEmotion. This manual will only address the options important for working with the CAETEC Plugin for IPEmotion.

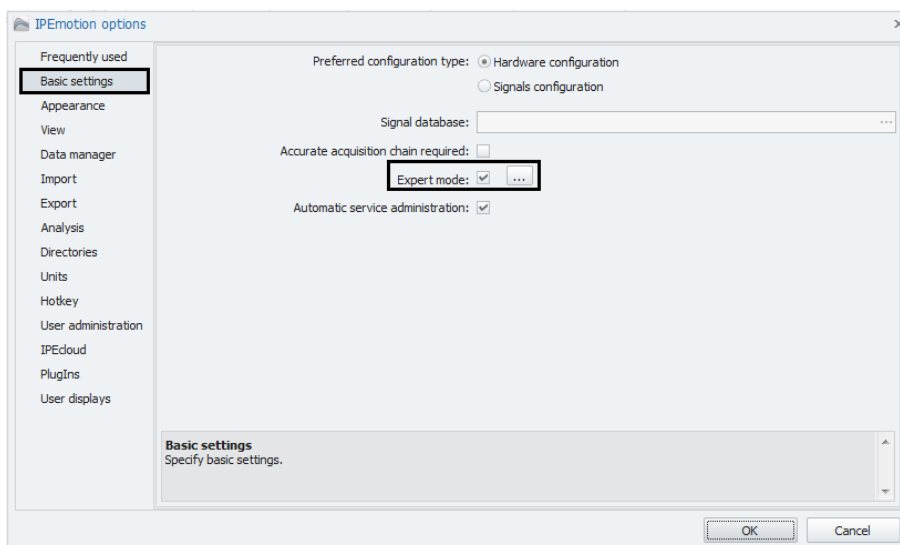
- **Change language**

The language can be changed in the options window by choosing the tab **“Appearance”** on the left and then choosing the desired language from the dropdown menu **“Language selection”** on the right.



- **Expert mode**

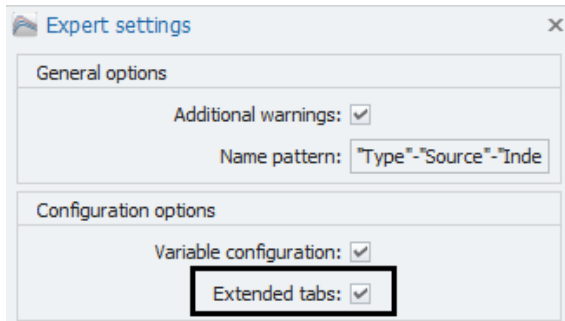
The Expert mode can be activated in the options window by choosing the tab **“Basic settings”** on the left and then checking the checkbox for Expert mode. Once you activated the Expert mode, **Expert settings** will be accessible by clicking on the button with the three dots next to the Expert mode checkbox.





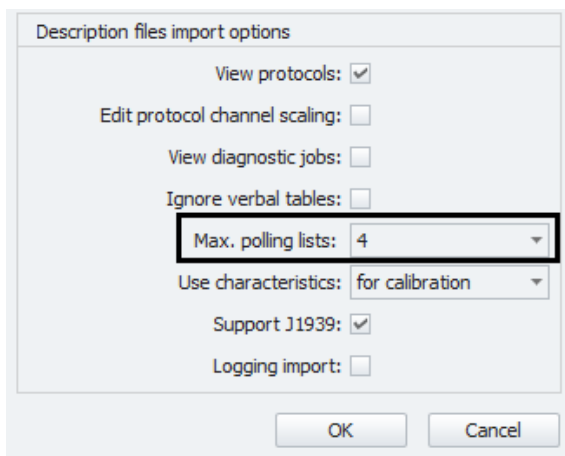
- **Extended tabs**

This Option should be activated. It will provide additional tabs in the Details area for some elements of the Measurement task tree. Extended tabs can be activated by checking the checkbox for **Extended tabs** in Expert settings.



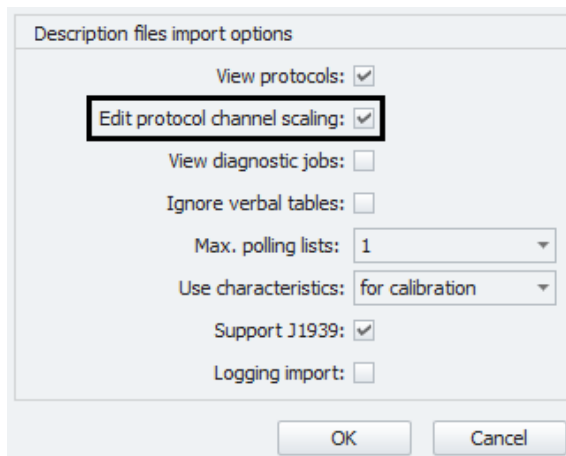
- **Maximum polling lists**

Defines the maximum number of polling lists. Multiple polling lists must be supported by the connector. The value should be set to a maximum of 4 polling list. The maximum number of polling lists can be changed by choosing the desired number from the **Max. polling lists** dropdown menu in Expert settings.



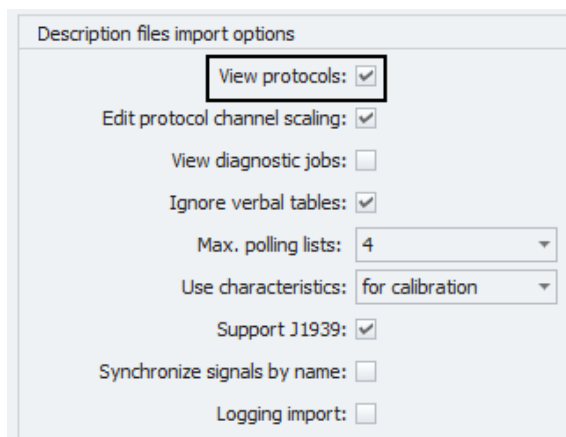
- **Edit protocol channel scaling**

This option has to be activated in Expert settings in order for the scaling functionality to be available for protocol channels as well.



- **View protocols**

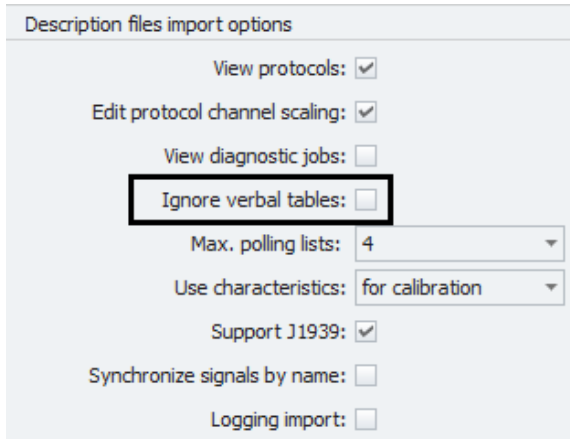
Activating this expert setting will allow you to see not only the ECU in the “Measurement task tree”, but also the associated daq lists and polling lists.



- **Ignore verbal tables (VTAB)**

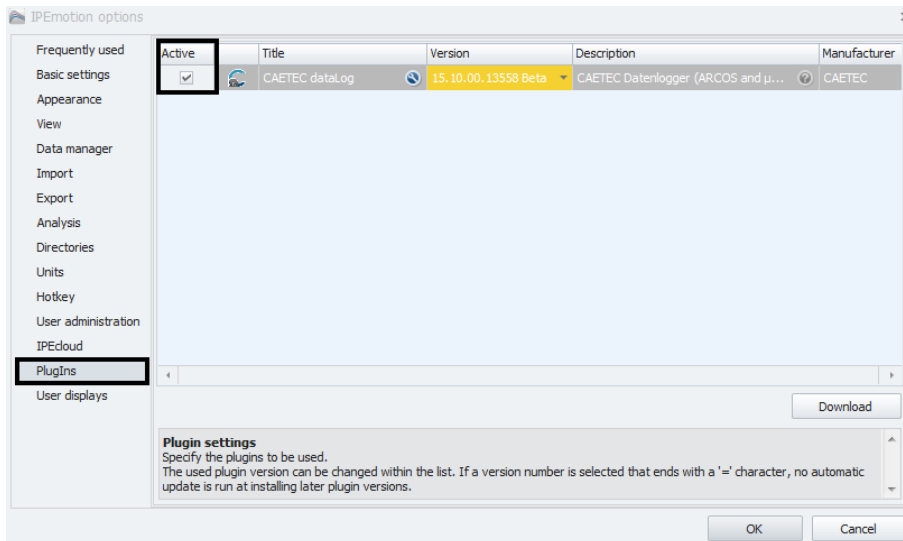
In the documentation for earlier version of this PlugIn it was recommended, to activate this setting, as the PlugIn did not support verbal tables.

From V17.10.00 on the PlugIn does now support verbal tables for single values/integers, so it is recommended to deactivate the "Ignore verbal tables" setting. However, verbal table ranges are still not supported by this PlugIn and will be transformed to a factor/offset scaling at export.



- **Activating the Plugin**

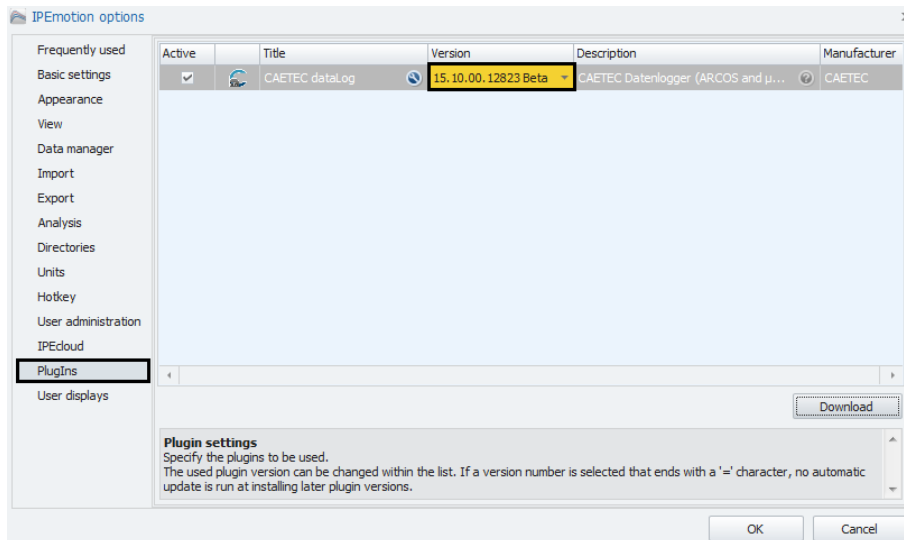
In order to work with the CAETEC PlugIn you need to activate it inside IPEmotion. To do so, choose **PluIns** on the left side of the Options window and then tick the checkbox saying *Active* for the CAETEC dataLog PlugIn.



- **Switching between Plugin versions**

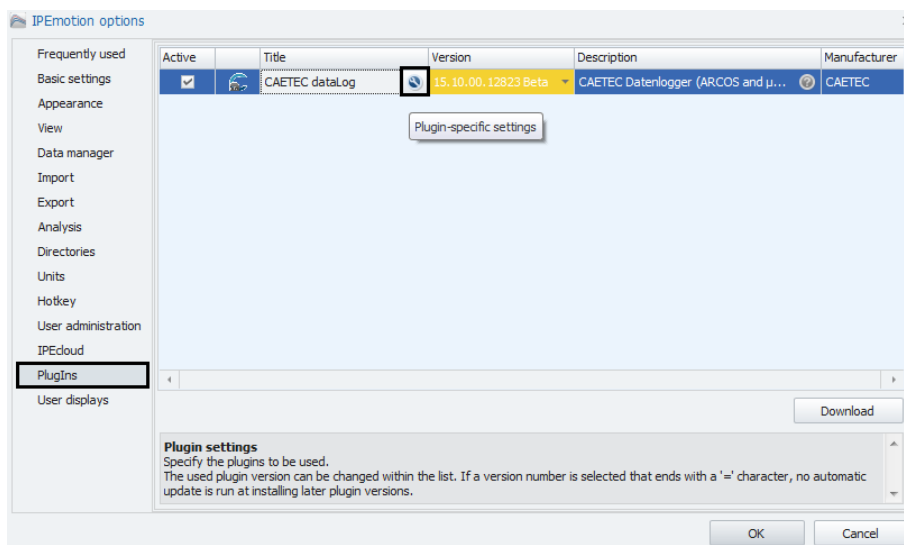
The CAETEC Plugin allows for various versions of the plugin in to be installed at the same time. It offers you the option to switch between different Plugin versions according to the firmware version your dataLog system is running, and therefore allows the use of various dataLog systems with different firmware versions.

In order to switch between Plugin versions choose **Plugins** on the left side of the Options window and then you can choose the desired version in the yellow marked dropdown menu on the right.

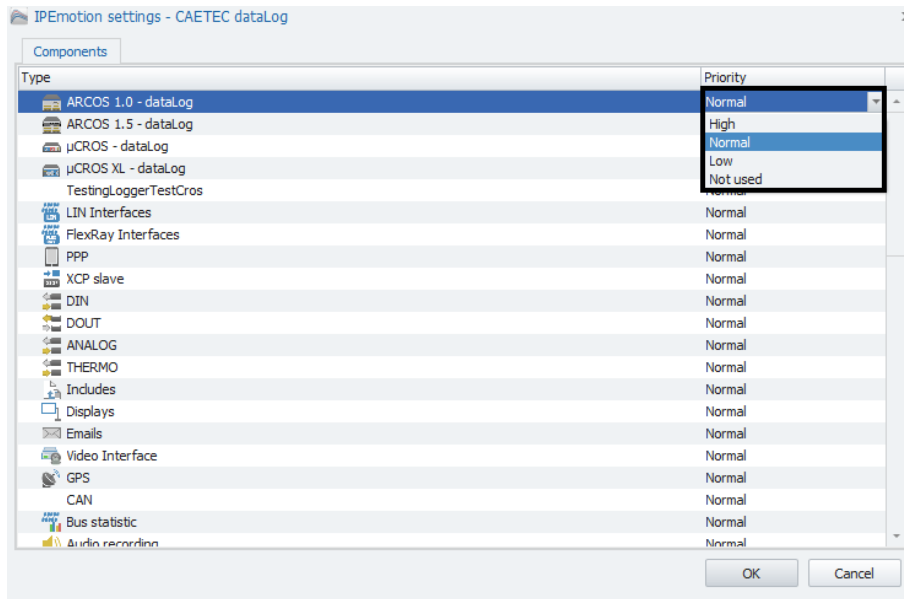


- **Customizing tree elements**

The CAETEC Plugin for IPEmotion allows for a customizable user guidance by letting you specifically determine which elements of the Measurement task tree will be available or not. In order to do so, choose **Plugins** on the left side of the Options window and then press the small button with the blue wrench symbol to open the **Plugin-specific settings**.



By choosing an element in the “Components” tab of the following window and setting its priority to “**Not used**”, as shown below, this element will not be available for the respective Plugin version anymore. If you would like to make available an element which you have previously set to “**Not used**” then you just have to set its priority back to “**Normal**”.

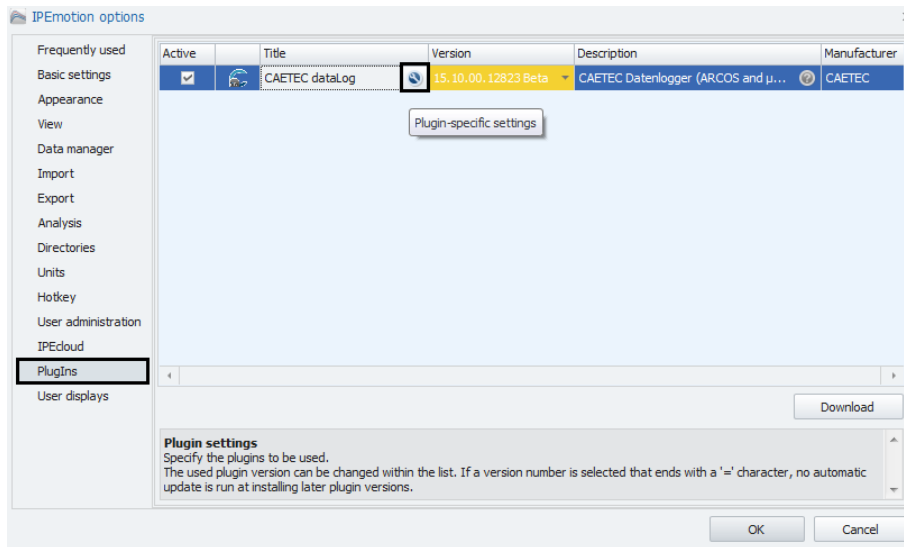


- **Create and synchronize external files**

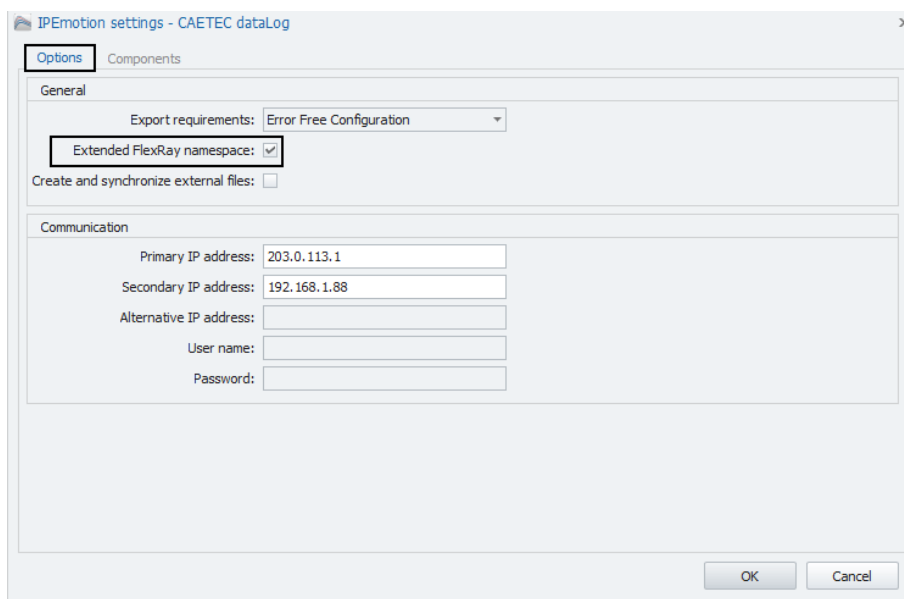
It is possible to set up the CAETEC dataLog Plugin in such a way, that imported signal databases will automatically be added as external file to their respective signal channel. This option is described in detail in the chapter “External files”. Please refer to (→ [12.1](#)).

- **Extended FlexRay namespaces**

The CAETEC Plugin for IPeMotion provides the option to work with an extended FlexRay namespace, including base cycle and cycle repetition parameters of signals in the namespace. This may be useful, if the regular FlexRay namespace creates ambiguous names. In order to do so, choose **Plugins** on the left side of the Options window and then press the small button with the blue wrench symbol to open the **Plugin-specific settings**.



In the following window activate the option by marking active the “Extended FlexRay namespace” tickbox in the “Options” tab.



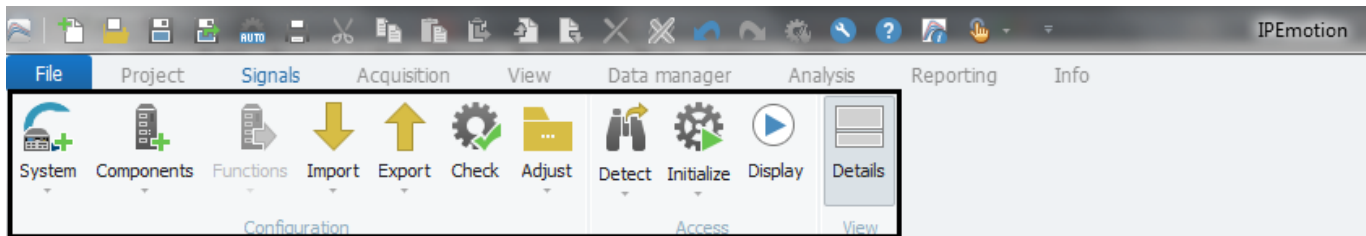


- **Communication settings**

In order for online communication via Ethernet between The CAETEC Plugin for IPE-motion and a PC to work, it may be necessary to specify some communication settings. For instructions on how to do so, please refer to (→[4.6](#)).

### 3.2.3 Working with the Ribbon

This area contains general functions regarding your configuration.



The “Ribbon’s” functionalities “**Components**”, “**Functions**”, “**Import**” and “**Export**” are also accessible by rightclicking on the tree element to which you would like to apply one of these options and then choosing the respective option from the context menu.

#### System

A dropdown menu that lets you choose which data logger system you want to configure. You can change the logger system into a different system at any given moment by right-clicking on your current logger system in the measurement task tree and selecting **Change into**.

#### Components

Offers a choice of additional components which are available for the element currently selected in the measurement task tree. You add the desired component by clicking on it. Each activated component will appear in the measurement task tree as a child element to your previously selected element. The same functionality is accessible through right-clicking an element in the measurement task tree and then choosing **Components**.

#### Functions

Offers a choice of additional functions which are available for the element currently selected in the measurement task tree. You activate the desired function by clicking on it. The same functionality is accessible through right-clicking an element in the measurement task tree and then choosing **Functions**.

#### Import

Offers a choice of import-options which are available for the element currently selected in the measurement task tree. You choose the desired import-option by clicking on it. The same functionality is accessible through right-clicking an element in the measurement task tree and then choosing **Import**.

### Export

Offers a choice of export-options which are available for the element currently selected in the measurement task tree. You choose the desired export-option by clicking on it. The same functionality is accessible through right-clicking an element in the measurement task tree and then choosing **Export**.

### Check

Perform a check on your current configuration's validity. The results will be automatically presented in a pop-up window once the check has finished. In the pop-up window you have the option of rerunning the check by clicking **refresh** and to export the results as a csv-file or html-file. The check function will be automatically performed each time the **datalog.cfg** is exported.

### Adjust

This functionality is currently not supported by this plugin.

### Detect

If a logger is connected to your PC, this function detects any connected logger and imports the configuration currently in use on the logger. This gives you the possibility to modify a pre-existing loggerconfiguration without the need of setting it up from scratch.

If there is no valid configuration in use on the connected logger, the plugin will detect all the available interfaces of the logger and adjust the measurement task tree elements accordingly.



If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([→3.2.2](#)).

### Initialize

If a logger is connected to your PC, this function exports the configuration currently in use in IPEmotion to the logger. During export the configuration will be checked for validity and give notice if any errors occur.



If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([→3.2.2](#)).

### Reset

If a logger is connected to your PC, this function deploys a basic configuration compatible with your logger model to the logger.



If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([→3.2.2](#)).

### Display

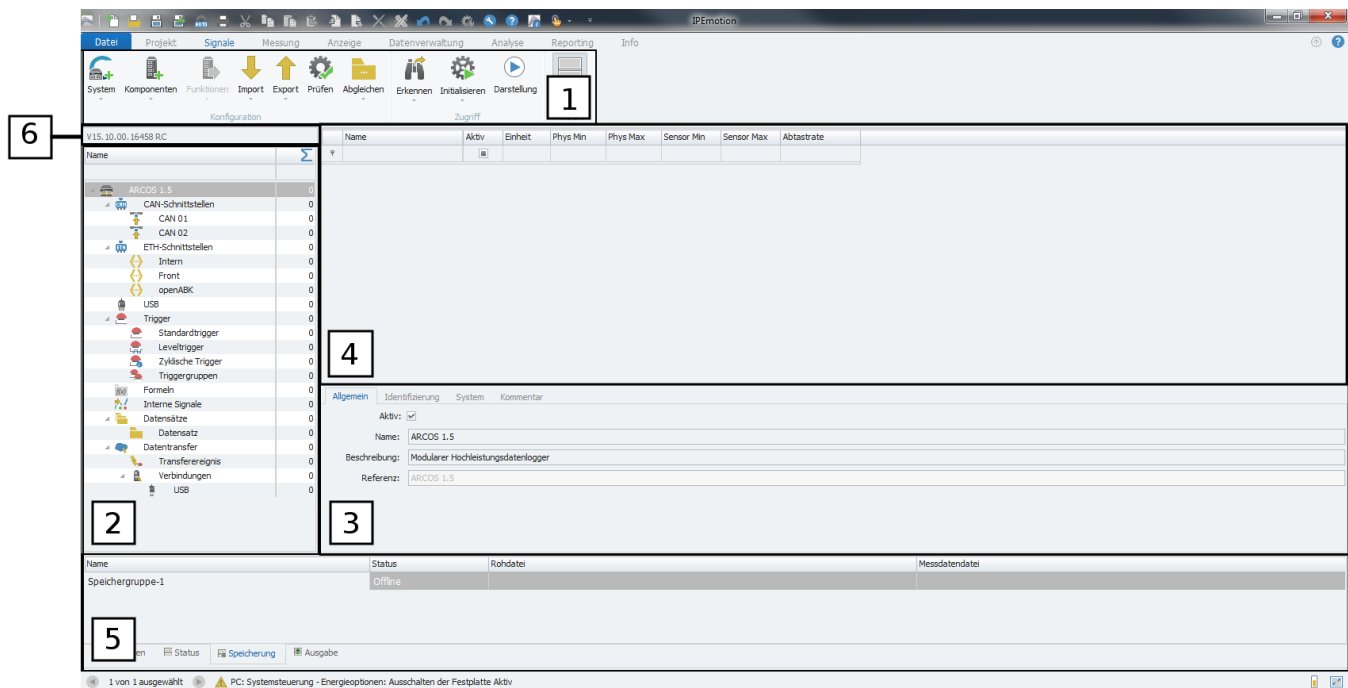
This functionality is currently not supported by this plugin.

### Details

Allows you to hide/show the details area of your current configuration.

### 3.2.4 "Signals" tab

The "Signals" tab, which is located in the menu bar (→ 3.2.1), contains your main workspace when working with the CAETEC Plugin for IPEmotion. It is divided in various areas which allow you to create measurement tasks and configure your data logger.



**1** **Ribbon**  
A strip of icons that can be clicked for quick access to certain functions and tools. (→ 3.2.3)

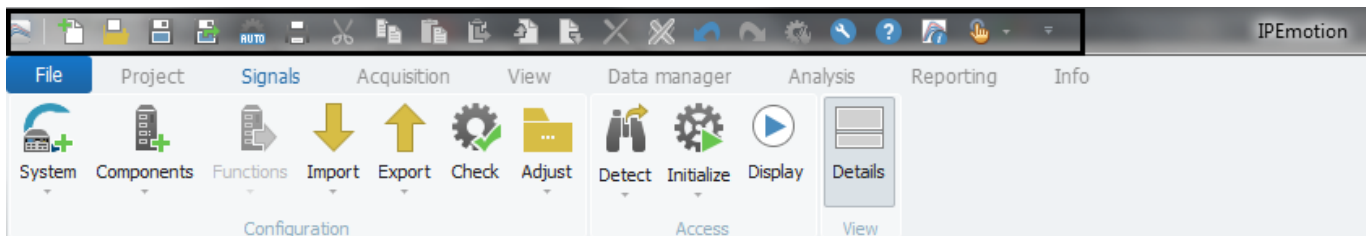
**2** **Measurement task tree**  
The measurement task tree shows a hierarchical view of the individual configuration sections for the opened measurement task. Specific information on working with the measurement task tree will be given in chapter (→ 4.2.1).

**3** **Details area**  
This area contains, for the selected section of the measurement task tree, a field and/or tabs that allows you to set the parameter settings for your selected section. Specific information on working with the Details area will be given in chapter (→ 4.2.2).

**4** **Grid area**  
This area contains, for the selected section of the measurement task tree and its child elements, a grid which shows an overview of available measurement channels. Specific information on working with the grid area will be given in chapter (→ 4.2.3).

- 5 Message area**  
This area contains messages about errors, warnings and information of the current configuration of the measurement task (→ [3.2.6](#)).
- 6 Plugin version**  
The current Plugin version in use is shown in a field located above the measurement task tree. The format of the version is always Vxx.xx.xx and has to match the target system's dataLog firmware-version (→ [4.3](#)).

### 3.2.5 Quick Access Toolbar



A customizable toolbar which allows you to quickly access your most frequently used tools. For further information please refer to the IPeMotion documentation.

### 3.2.6 Message area

This area contains important messages, the status, potential conflicts and errors of the current configuration of the system.

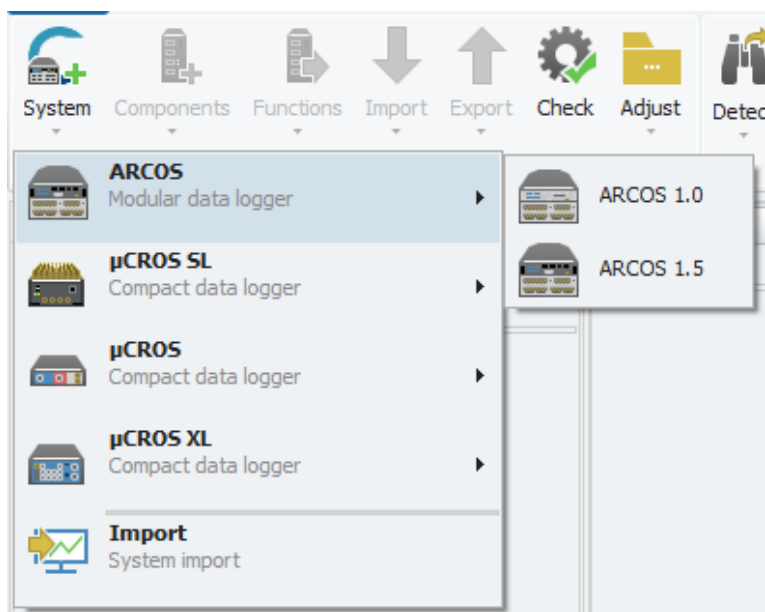
## 4 Setting up a logger system

This section explains the workspace and the steps to set up the right logger system for your project/configuration. It will also explain options to customize the use of your logger system.

### 4.1 Choosing a logger-system

Once you have activated the Plugin (→3.2.2) choose the desired logger-system. Click the **System** button on the top left in the ribbon and then select your desired system.

The system you choose should match your target system (e.g. if your hardware is an ARCOS 1.5 then you should choose the ARCOS 1.5 as the system you wish to configure).



According to your hardware's possible interfaces a preconfigured workspace will be opened inside the Signals tab. This workspace will be explained in more detail in the following.

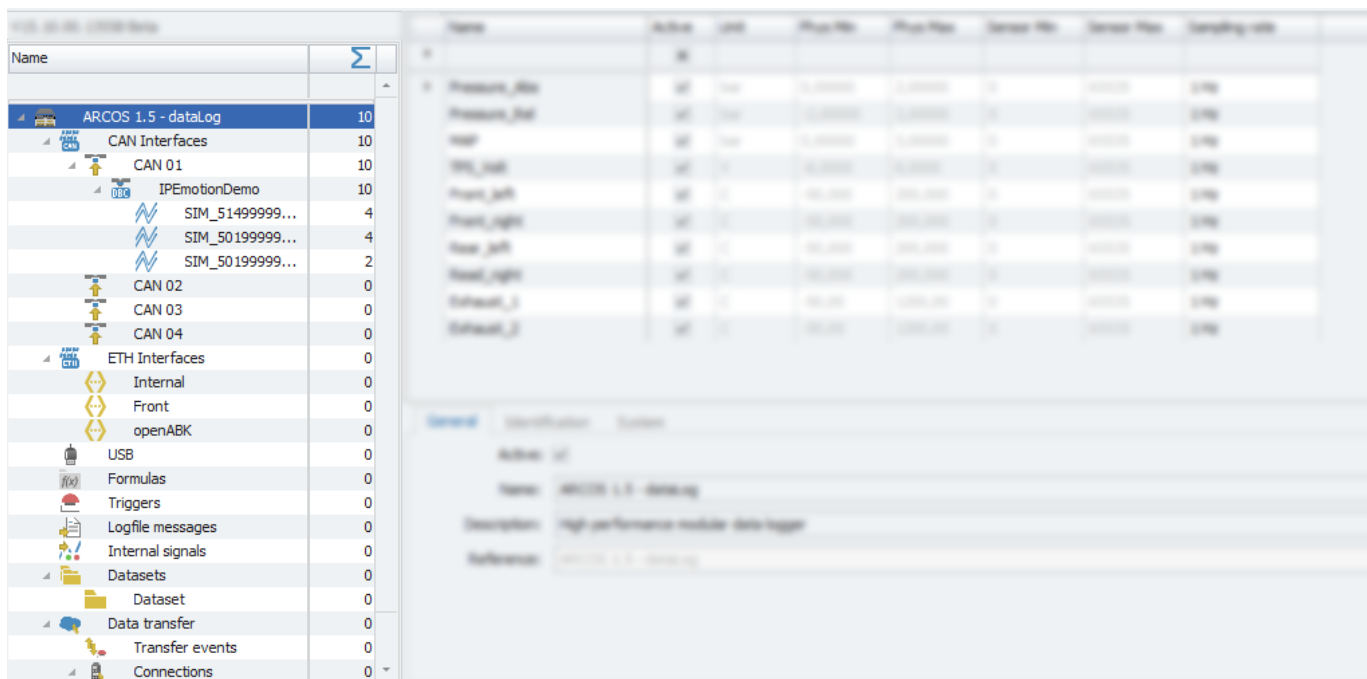
### 4.2 The measurement task workspace

The main parts of the measurement task workspace are the **Measurement task tree** (→ 4.2.1), the **details area** (→ 4.2.2) and the **grid area** (→ 4.2.3). For information on other parts please refer to (→ 3.2.4).

#### 4.2.1 The measurement task tree

The measurement task tree shows a hierarchical view of the individual measurement task-configuration pages of the Plugin. Clicking an item in the tree opens the corresponding configuration pages in the details area and the grid area to the right of the tree, where you can perform the desired settings.

Right-clicking an element in the measurement task trees opens a context menu showing options for that element. Depending on the type of element, this context menu gives you the option of adding additional child elements below the clicked element, for example, or lets you remove or disable optional configuration elements.



The screenshot displays the measurement task workspace. On the left is a tree view showing a hierarchical structure of measurement tasks. The root node is 'ARCOS 1.5 - dataLog' with a count of 10. It branches into 'CAN Interfaces' (10), 'ETH Interfaces' (0), 'USB' (0), 'Formulas' (0), 'Triggers' (0), 'Logfile messages' (0), 'Internal signals' (0), 'Datasets' (0), 'Data transfer' (0), 'Transfer events' (0), and 'Connections' (0). Under 'CAN Interfaces', there is 'CAN 01' (10), which contains 'IPEmotionDemo' (10) and three simulation nodes: 'SIM\_51499999...' (4), 'SIM\_50199999...' (4), and 'SIM\_50199999...' (2). Other CAN interfaces (CAN 02, CAN 03, CAN 04) have counts of 0. Under 'ETH Interfaces', there are 'Internal', 'Front', and 'openABK' nodes, all with counts of 0.

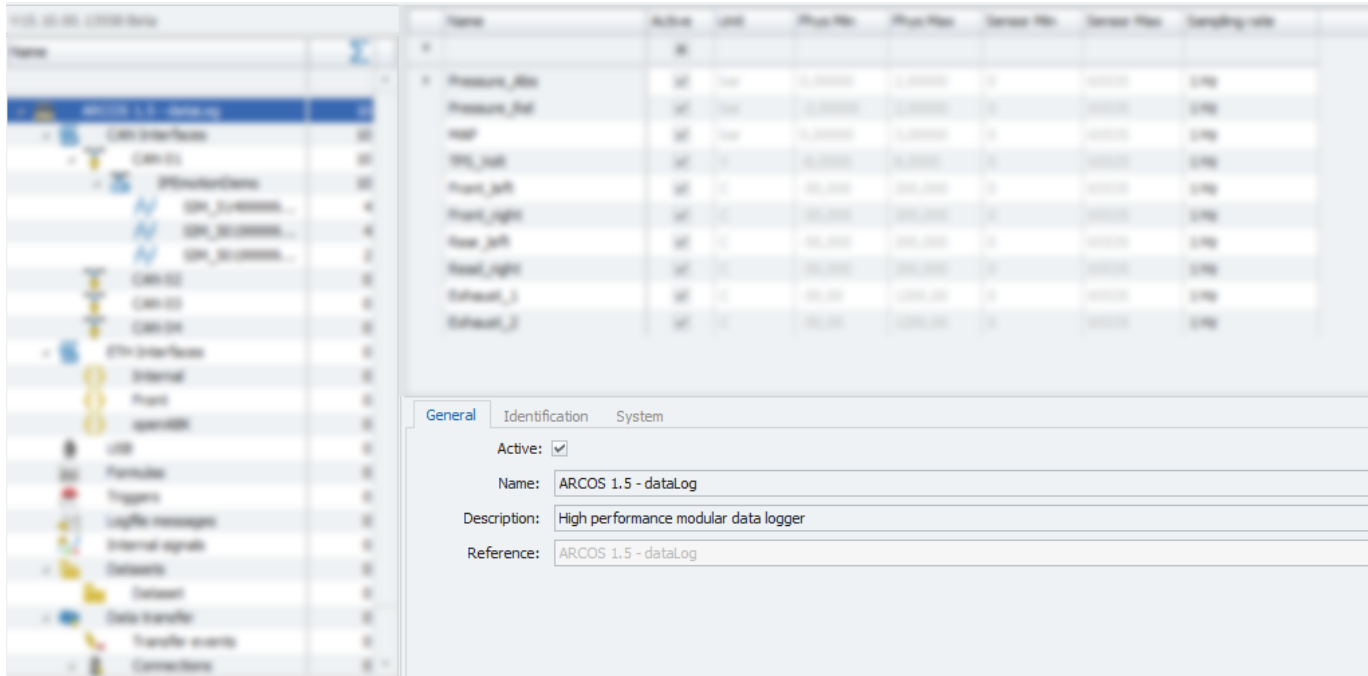
On the right, the details area shows a table of configuration parameters for the selected 'IPEmotionDemo' element. The table has columns for 'Name', 'Active', 'Unit', 'Phys. Min.', 'Phys. Max.', 'Sensor Min.', 'Sensor Max.', and 'Sampling rate'. The parameters listed are: 'Pressure\_00a', 'Pressure\_00b', 'WSP', 'WSP\_Low', 'Front\_00a', 'Front\_00b', 'Rear\_00a', 'Rear\_00b', 'Wheel\_0', and 'Wheel\_1'. All 'Active' checkboxes are checked, and all 'Sampling rate' values are 100.

Below the table, the details area shows a 'General' tab with the following information: 'Name: ARCOS 1.5 - dataLog', 'Description: High performance modular data logger', and 'Reference: ARCOS 1.5 - dataLog'.



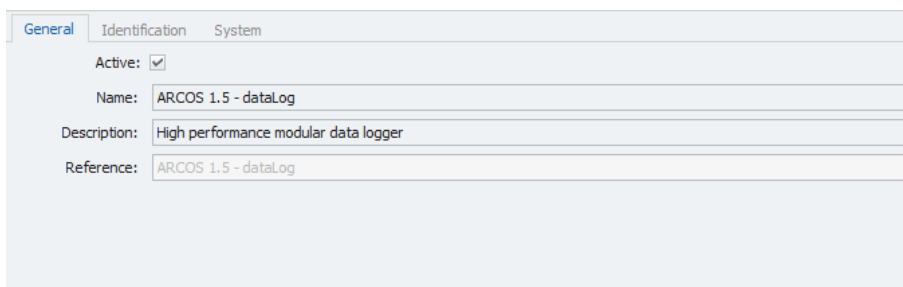
### 4.2.2 The details area

The details area contains tabs which allow for additional settings for the selected tree element. In this section we will quickly describe the Details settings available for your CAETEC dataLog system, which you will see, once you clicked on your system on top of the measurement task tree. More detailed settings for other tree elements will be handled in the respective sections of this manual.



- **General**

This tab allows you to activate or deactivate the entire system by ticking/unticking the checkbox, give a user specific name to your system if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.



- **Identification**

This tab allows for more specific Identification of your system. The **Configuration name** serves as an identifier of your configuration for the user. This is the name of your system that will be visible in the webinterface of your datalogger.

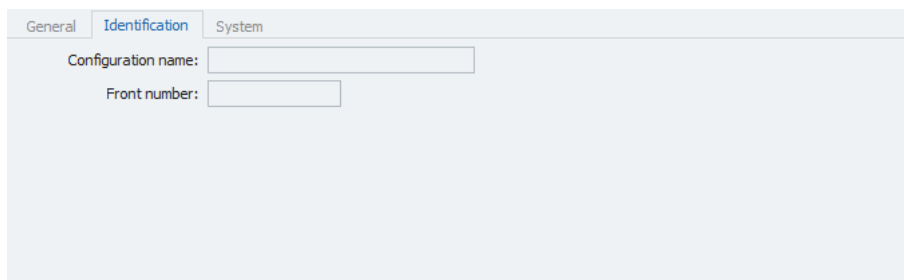
The **Front number** field allows you to enter a target system's unique front number, which will have the effect, that the created datalog.cfg will only work with that specific target system.



In the field "Configuration name" project parameters can be used as variables. For more information please refer to (→[5.6](#)).



If a front number is entered, the configuration will exclusively function for the logger with the corresponding front number. It will not work on any other logger.



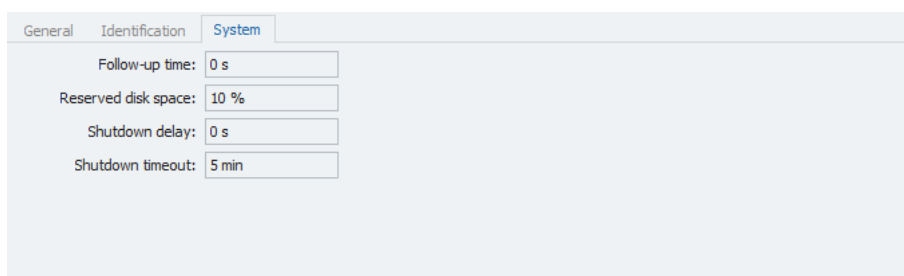
General Identification System

Configuration name:

Front number:

- **System**

This tab allows for system specific settings. **Follow-up time** defines how long the logger keeps measuring after the last wake-up condition went away. **Reserved disk space** determines the amount of disk space that is reserved for internal processing (e.g. zip-compression). **Shutdown delay** sets how long the shutdown of the whole system will be delayed after having ended current measuring. During this time a new measurement can be started by a valid wake-up condition. **Shutdown timeout** determines the maximum allowed time for a shutdown of the system. If this time is exceeded, a hard shutdown will be forced and all processes, including datatransfers, will be cancelled.



General Identification System

Follow-up time:

Reserved disk space:

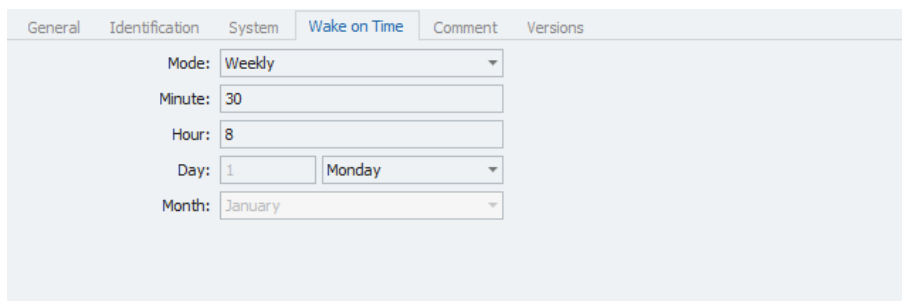
Shutdown delay:

Shutdown timeout:

- **Wake on time**

The “Wake on time” option allows you to set your logger to wake up on a specific point of time. Whenever this point of time occurs the logger will wake up, if it is not active already. I.e. if you set the mode to hourly at minute “0”, the logger will wake up at every full hour. If you set it to hourly and minute “30”, the logger will wake up always at half past the full hour.

If in another case you would like the logger to wake up every monday at 8.30 in the morning, the setting would be **Mode: weekly, Minute: 30, Hour: 8, Day: Monday**, as seen in the figure below.



The screenshot shows the 'Wake on Time' configuration tab. It contains the following fields:

- Mode: Weekly (dropdown menu)
- Minute: 30 (text input)
- Hour: 8 (text input)
- Day: 1 (text input) and Monday (dropdown menu)
- Month: January (dropdown menu)

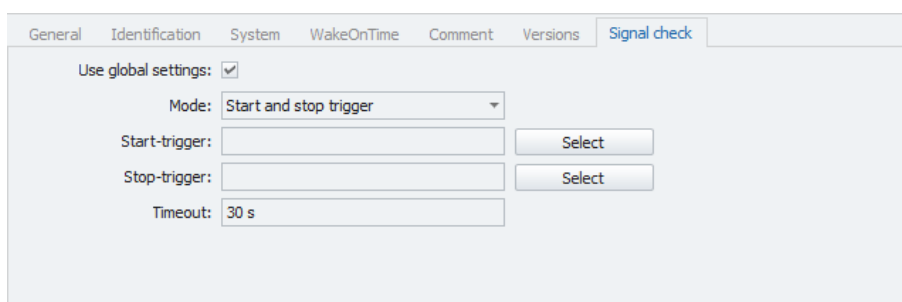
- **Signal check**

If set active, the “Signal check” tab allows you to define a set of global signal settings, which can then later be applied to any signal. That way you can make sure, that all signals which are supposed to be configured in the same way, will actually use the same settings.

The “Signal check” function allows to set one of three modes: “Continuous”, “Start and stop trigger” and “Stop is inverted start”.

It also allows to set the Start-trigger and the Stop-trigger and the timeout.

If any of these parameters gets changed, it will change also for any signal to which is using the global signal check settings.



The screenshot shows the 'Signal check' configuration tab. It contains the following fields:

- Use global settings:
- Mode: Start and stop trigger (dropdown menu)
- Start-trigger: [text input] [Select button]
- Stop-trigger: [text input] [Select button]
- Timeout: 30 s (text input)

### 4.2.3 The grid area

The grid area shows all available signals of the selected tree element. It also allows to activate certain signals, to rename them and sort by the different column's parameters. The "column chooser" allows you to customize which columns will be shown in the grid area. For more information on the "column chooser" please refer to (→[4.3.1](#)).

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
Pressure_Abs	<input checked="" type="checkbox"/>	bar	0,00000	2,00000	0	65535	1 Hz
Pressure_Rel	<input checked="" type="checkbox"/>	bar	-2,00000	2,00000	0	65535	1 Hz
MAP	<input checked="" type="checkbox"/>	bar	0,00000	3,00000	0	65535	1 Hz
TPS_Volt	<input checked="" type="checkbox"/>	V	-8,0000	8,0000	0	65535	1 Hz
Front_left	<input checked="" type="checkbox"/>	C	-50,000	200,000	0	65535	1 Hz
Front_right	<input checked="" type="checkbox"/>	C	-50,000	200,000	0	65535	1 Hz
Rear_left	<input checked="" type="checkbox"/>	C	-50,000	200,000	0	65535	1 Hz
Rear_right	<input checked="" type="checkbox"/>	C	-50,000	200,000	0	65535	1 Hz
Exhaust_1	<input checked="" type="checkbox"/>	C	-50,00	1200,00	0	65535	1 Hz
Exhaust_2	<input checked="" type="checkbox"/>	C	-50,00	1200,00	0	65535	1 Hz

## 4.3 Plugin version

The current Plugin version in use is shown in a field located above the measurement task tree. The format of the plugin version is always Vxx.xx.xx (Vmajor.minor.hotfix). The plugin version has to match the target system's dataLog firmware-version.



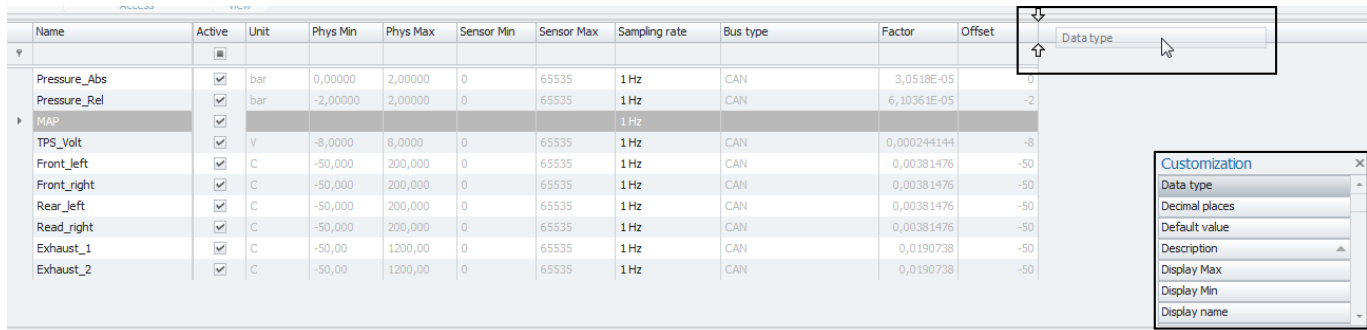
If you have multiple Plugin versions installed, you may switch between versions. For detailed information please refer to **Switching between Plugin versions** (→ [3.2.2](#)).

## 4.3 PLUGIN VERSION

### 4.3.1 Column chooser

The “column chooser” allows you to customize which columns will be visible in the grid area. You can customize your “grid area” in such a way, that every setting you can adjust in the “details area” for a given tree element can be made accessible through the “grid area” of this tree element.

In order to access the “Column chooser” rightclick on any point of the column title bar and choose “Column chooser” from the context menu.

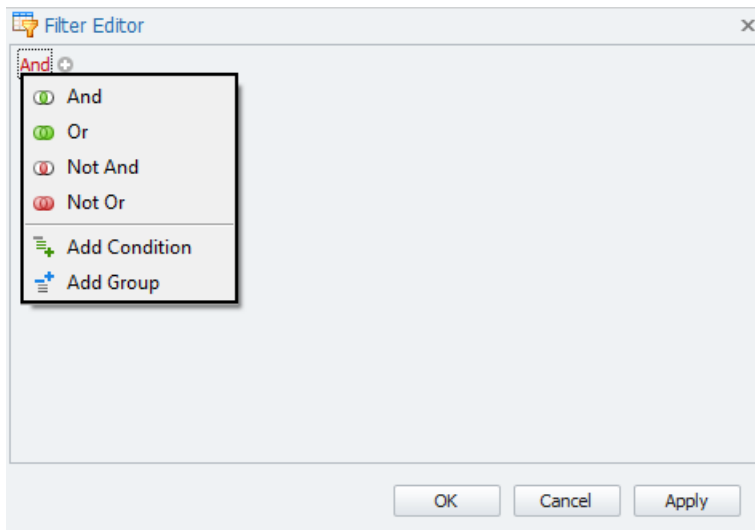


The resulting window on the right called “Customization” will present you with a selection of the available columns. In order to add an extra column, choose the one you would like to add from the “Customization” window and drag it to the column bar as shown above.

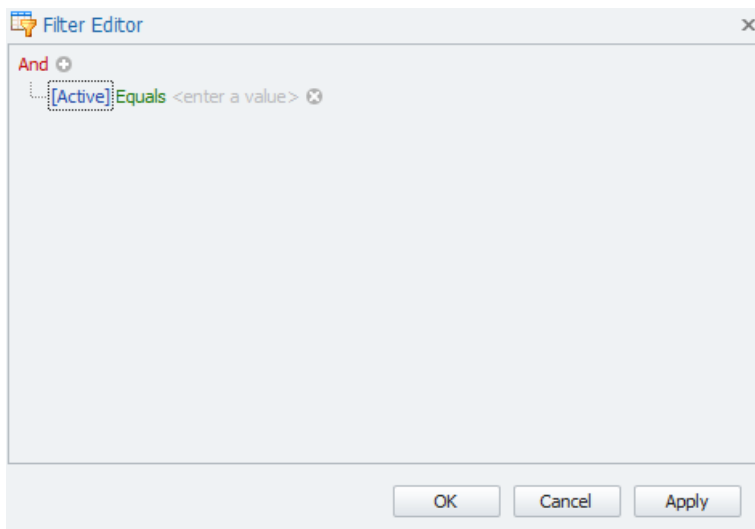
If you wish to remove a column just drag it to any point outside of the column title bar until you see a big black cross and then release it.

### 4.3.2 Filter editor

The “Filter editor” allows you to apply customized filter rules to your signals. In order to access the “Filter editor” rightclick on any point of the column title bar and choose “Filter editor” from the context menu.

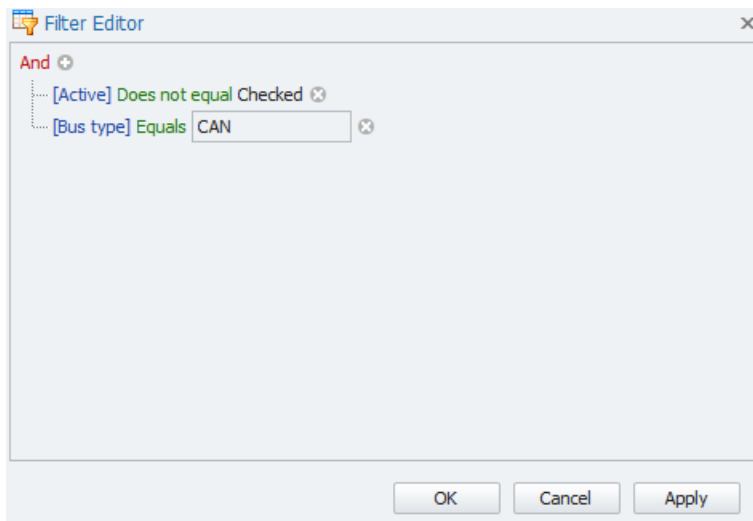


Once the “Filter editor” has opened, click on the red “And” in the top left corner. The resulting context menu allows you to choose the method of combining the different filter conditions (And, Or, Not And, Not). It also allows you to add a condition (which can alternatively be achieved by clicking the “+” sign right of the “And”) or add a new group of filter conditions.



Once you have added a filter condition, you will now need to specify this condition in order to function properly. There are three fields you will need to set.

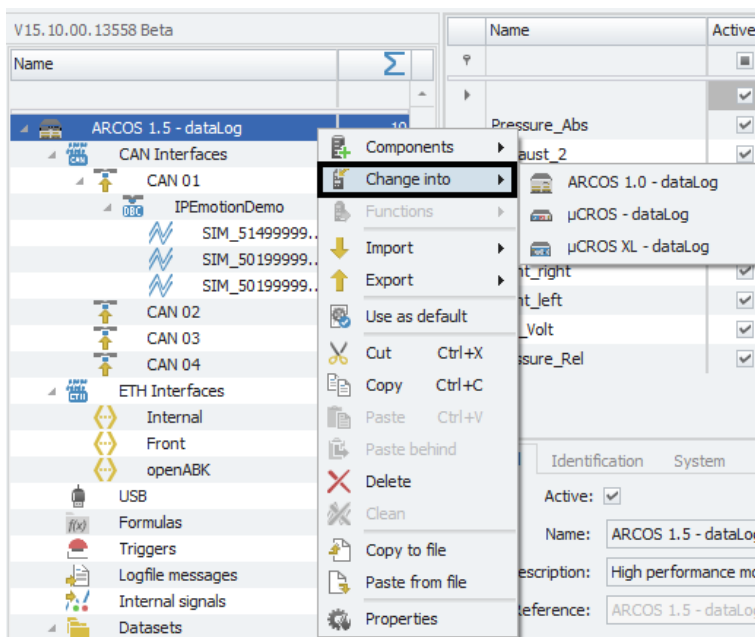
- The field on the left with **blue letters** defines to which column the filter will apply.
- The field in the middle with **green letters** sets a parameter for your filter condition.
- The field on the right with **grey letters** sets a value for the chosen parameter.



You can combine different filter conditions or groups of filter conditions to obtain the desired filter result. In the example shown above, the filter will select all CAN signals which are not checked "Active".

#### 4.4 Changing the logger system with the *change into*-command

A previously set up logger system can still be changed into another one later. To do so, right-click on your system in the measurement task tree, navigate to **change into** and choose the system you would like to change it to.



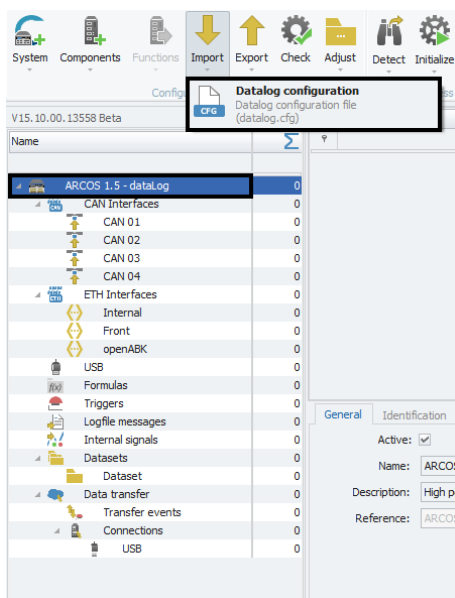
If you are trying to change into another logger system and one of your currently used tree elements is not available in the system you are changing into, you will see a **Warning-message** in the Messages area (→[3.2.6](#)) and this tree element will not be migrated.

## 4.5 Importing and exporting the datalog.ccmc

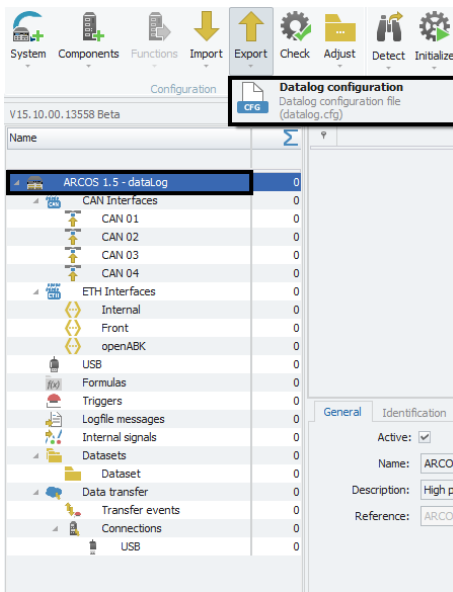
Once you have chosen the system you would like to configure, you will be able to import and/or export the datalog.ccmc. That way you can import a datalog.ccmc previously configured with the plugin and can keep the settings. The export function allows you to export your configuration and then copy it to one or more matching target systems. To do so, just select your system in the measurement task tree and then select either **Import** or **Export** from the Ribbon and choose datalog.ccmc. Alternatively you can right-click on your system in the measurement task tree and select Import or Export from the following context menu.



The filename under which the exported .ccmc-file will be save cannot contain any special characters such as ` , \$ , ö and so forth. The spelling of the filename will be checked during export and any special character in the filename will be replaced with an underscore ("\_").





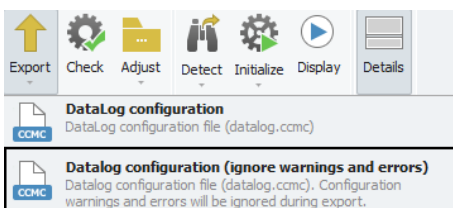


### 4.5.1 Ignore errors and warnings at export

The CAETEC Plugin for IPEmotion checks any configuration for errors and warnings before exporting the ccmc-container or when using the “Check” function in the Ribbon. If a warning occurs, export will still happen, but the element causing the warning will not be exported.

If an error occurs, the export will be canceled.

However, if you click “Export” in the Ribbon and then choose “Datalog configuration (ignore warnings and errors)”, export will happen even if an error or warning occurs. The error/warning will then appear as info-message instead and export will proceed.

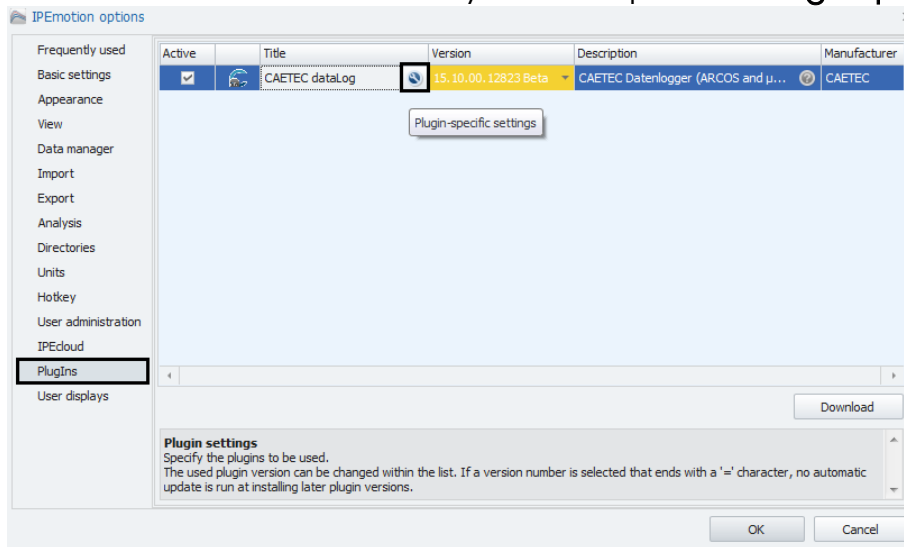


## 4.6 Online communication with the logger

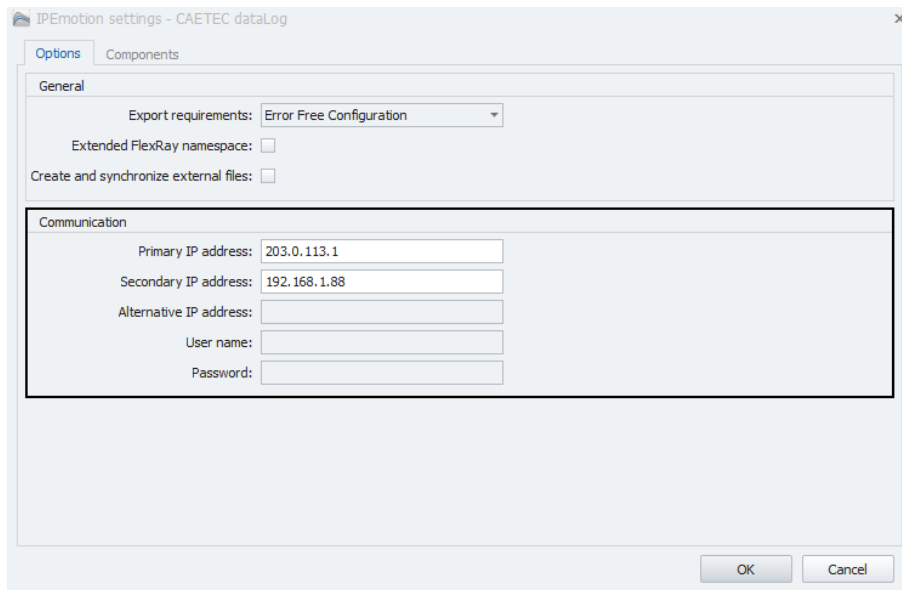
There are several functions that require online communication between the logger and the pc via Ethernet. This section will explain how to set up the logger for online communication and also the different functions that require online communication.

### 4.6.1 Communication settings

In order for online communication via Ethernet between The CAETEC Plugin for IPEmotion and a PC to work, it may be necessary to specify some communication settings. In order to do so, choose **Plugins** on the left side of the Options window and then press the small button with the blue wrench symbol to open the **Plugin-specific settings**.



In the "Options" tabsheet of the following window you will find a field called "Communication", containing all the relevant settings for online communication between a logger and a PC.



### Primary IP address

Primary IP address for communication with a logger. This IP address cannot be changed.

### Secondary IP address

Secondary IP address for communication with a logger. This IP address cannot be changed.

### Alternative IP address

If a logger is communicating on a user-specific IP address, type in this address here to ensure correct communication between the logger and the PC.

### User name and Password

If a logger is protected via user name and password, type them in here to ensure correct communication between the logger and the PC.

### 4.6.2 Online functions

The online functions provided by the plugin can be found in the Ribbon and will be explained in the following.

#### Detect

If a logger is connected to your PC, this function detects any connected logger and imports the configuration currently in use on the logger. This gives you the possibility to modify a pre-existing logger configuration without the need of setting it up from scratch.

If there is no valid configuration in use on the connected logger, the plugin will detect all the available interfaces of the logger and adjust the measurement task tree elements accordingly.



If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([→4.6.1](#)).

#### Initialize

If a logger is connected to your PC, this function exports the configuration currently in use in IPEmotion to the logger. During export the configuration will be checked for validity and give notice if any errors occur.



If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([→4.6.1](#)).

## Reset

If a logger is connected to your PC, this function deploys a basic configuration compatible with your logger model to the logger.

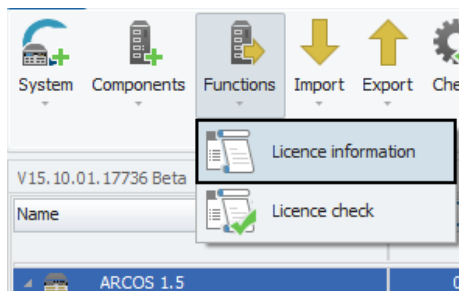


If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([→4.6.1](#)).

### 4.6.3 Licence information

If a logger is connected to the PC, it is possible for the plugin, to access that logger's licence information. Thus you can find out which licence keys are currently installed on your logger and which are still needed for your current configuration to function properly.

To read the licence information out, select your system in the measurement task tree (the topmost element), click the "Functions" button in the Ribbon and then choose "Licence information".

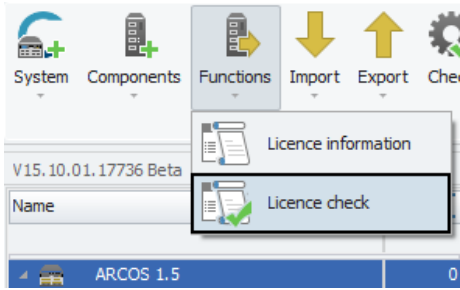


If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([to3.2.2](#)).

### 4.6.4 Licence check

If a logger is connectet to the PC, it is possible for the plugin, to check whether all the necessary licences for your current configuration are installed on the logger.

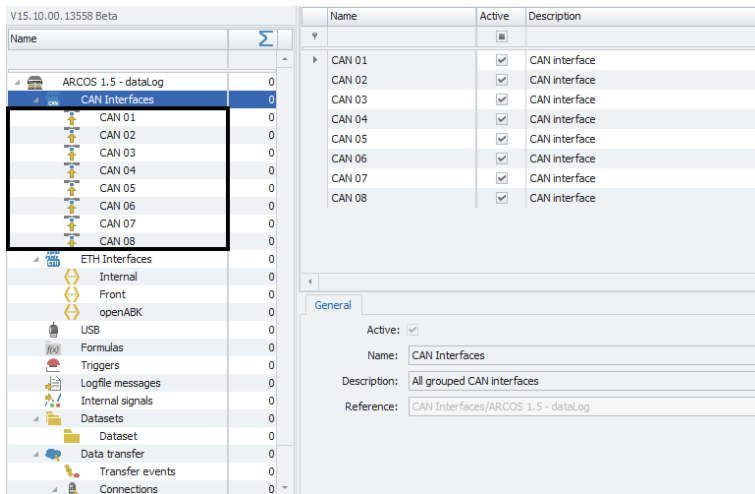
To perform a licence check, select your system in the measurement task tree (the topmost element), click the "Functions" button in the Ribbon and then choose "Licence check".



If a logger is configured with a user-specific IP or protected with user name and password, it may be necessary, to edit the communication settings of the plugin first. Instruction on these settings can be found here ([to3.2.2](#)).

### 4.7 Changing a system's default tree elements

The tree elements available by default for a given system are merely a preference, which can be adjusted via the measurement task tree. If you see, that you usually use 8 instead of 4 CAN busses, you can determine this setting as your default for the currently used system type and the next time you set up a new logger system it will automatically start with your new default settings. To do so, you will first need to add or remove the desired components. (In this case we added another 4 CAN busses in order to get a total of 8).



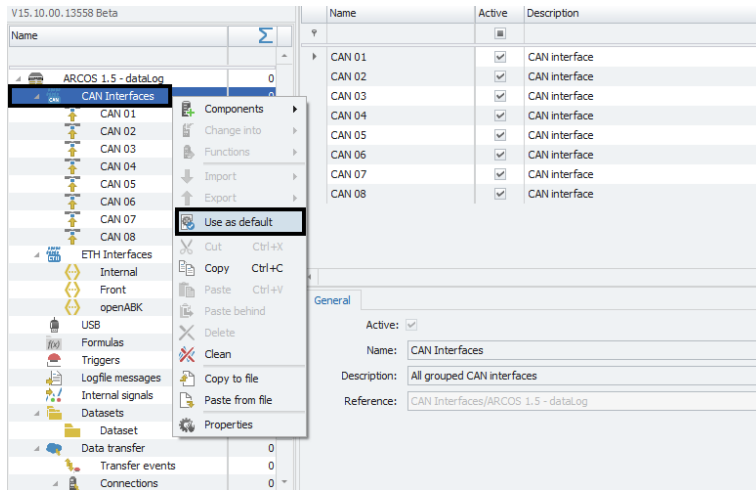
Name	Active	Description
CAN 01	<input checked="" type="checkbox"/>	CAN interface
CAN 02	<input checked="" type="checkbox"/>	CAN interface
CAN 03	<input checked="" type="checkbox"/>	CAN interface
CAN 04	<input checked="" type="checkbox"/>	CAN interface
CAN 05	<input checked="" type="checkbox"/>	CAN interface
CAN 06	<input checked="" type="checkbox"/>	CAN interface
CAN 07	<input checked="" type="checkbox"/>	CAN interface
CAN 08	<input checked="" type="checkbox"/>	CAN interface

General	
Active:	<input checked="" type="checkbox"/>
Name:	CAN Interfaces
Description:	All grouped CAN interfaces
Reference:	CAN Interfaces/ARCOS 1.5 - dataLog

## 4.7 CHANGING A SYSTEM'S DEFAULT TREE ELEMENTS

In the next step you will have to right-click on the category of tree elements for which you wish to define your new default settings as shown below. Note: it will not work, if you right-click the single component or the system itself. Make sure to right-click the category, in this case the **CAN Interfaces** and then choose **Use as default**.



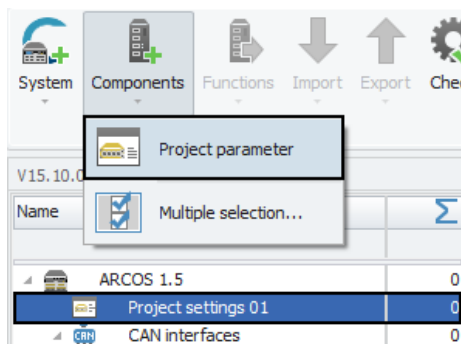


## 5 Project settings

“Project settings” allow you to include project parameters such as company name, serial number, project name etc. in your Configuration. It is also possible to create a set of project settings, specific for a dataset. That way, you can define different project parameters for different datasets. Please refer to the section “Dataset Project settings” (→ 14.3).

### 5.1 Adding project parameters

It is possible, to add user-specific project parameters in addition to the default project parameters. To do so, select the “Project settings” element in the measurement task tree, select the “Components” button in the Ribbon and then choose “Project parameter”.

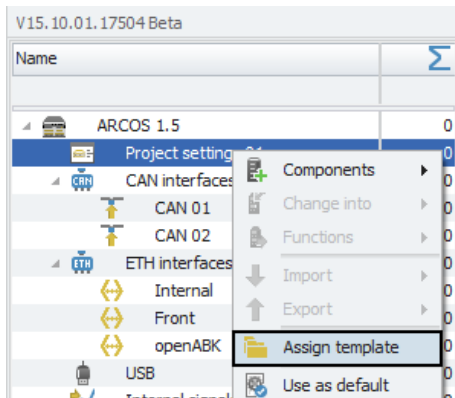


The new parameter will appear in the respective “Project settings xx” Grid area as customizable parameter in the table.

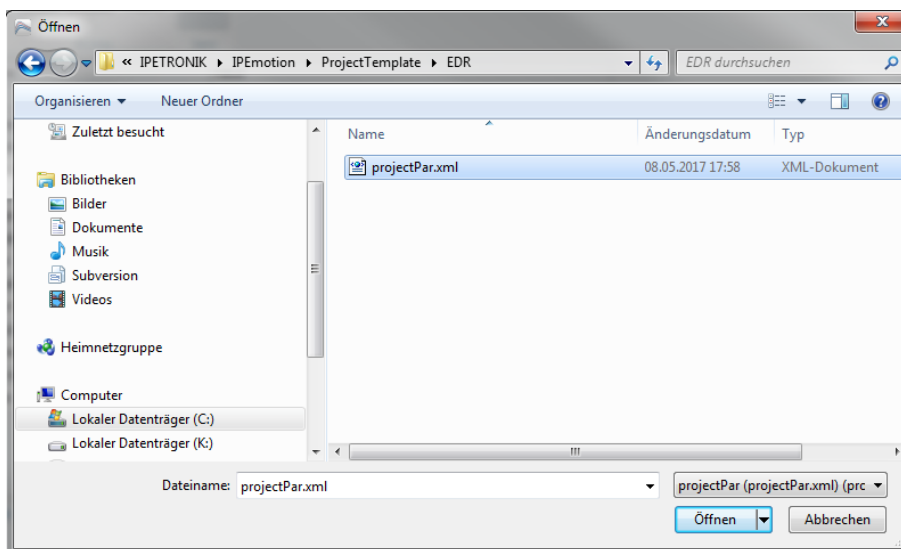
File name	
Date	27.06.2018 10:18:39
Project parameter 14	

## 5.2 Assigning a template of project parameters

It is also possible, to assign a template which contains a predefined set of user-specific project parameters. To do so, right-click on the “Project settings” tree element and then choose “Assign template”.



In the following window you may choose the template file and confirm with “Open”. The file type needs to be .xml.

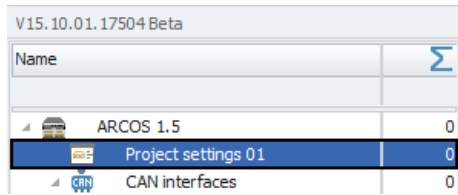


The CAETEC dataLog PlugIn for IPEmotion currently supports two types of templates, the “EDR” type and the “Feger” type. Both templates are installed with the plugin in the format “projectPar.xml” and can be found in the plugin’s installation directory under “UserData/ProjectTemplate”.

If at dataset creation either of these two template types is assigned, the dataset will be treated especially.

### 5.3 Tree elements for Project settings

Including “Project settings” in your configuration will add one new tree element. The tree element is labeled “Project settings”.



V15.10.01.17504 Beta	
Name	
ARCOS 1.5	0
<b>Project settings 01</b>	<b>0</b>
CAN interfaces	0

### 5.4 Grid area for Project settings

If the “Project settings” element is selected in the Measurement task tree, the grid area will provide you with a table, that allows you to access all default or previously defined user-specific project parameters.

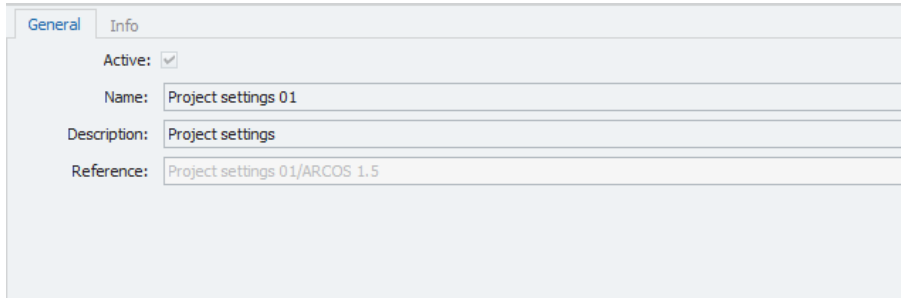
Name	Value
Company name	
Serial number	
Manufacturer	
Project name	
Project manager name	
E-mail address project manager	
User	
E-mail address user	
User login	bachl
Description	
IPEmotion version	V08.00.01
File name	
Date	27.06.2018 10:18:39
Project parameter 14	

## 5.5 Details area for Project settings

If the “Project settings” element has been selected in the measurement task tree, additional settings are available in the details area.

### General

This tab provides general settings for the selected Project settings file.



General	Info
Active:	<input checked="" type="checkbox"/>
Name:	Project settings 01
Description:	Project settings
Reference:	Project settings 01/ARCOS 1.5

- **Name**

Give a user-defined Name to the selected formula/signal.

- **Description**

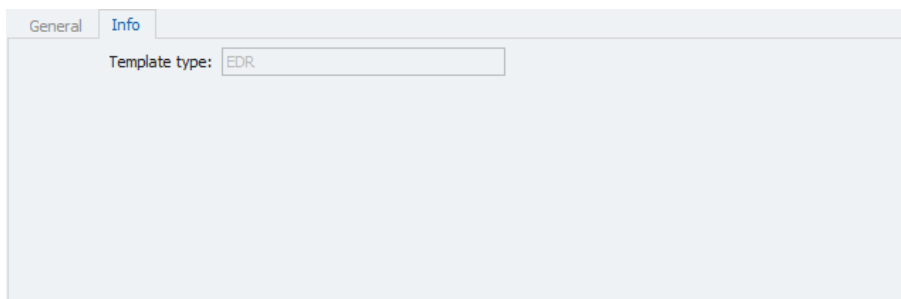
Give a user-defined description to the selected formula/signal.

- **Reference**

This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

### Info

Tells you the type of template that has been assigned.

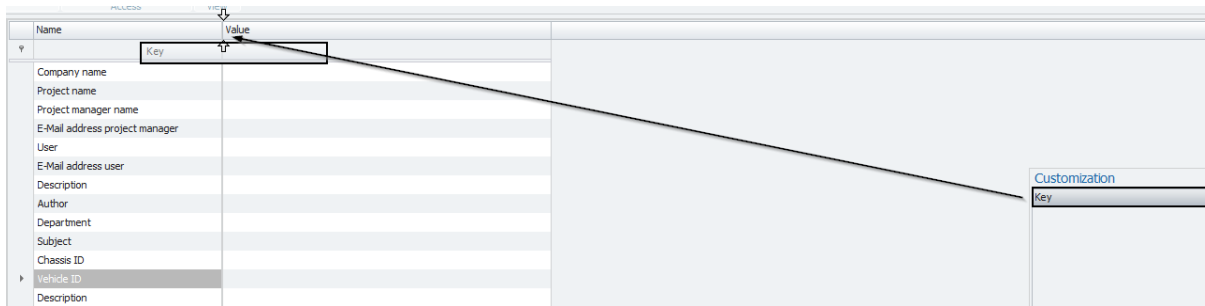


General	Info
Template type:	EDR

## 5.6 Using project parameters as variables in CAETEC dataLog Plugin

If project settings have been added to your configuration, the Keys of these project settings can be used as variables in some text fields. This can be helpful if the same Parameter (e.g. "VehicleID") should be used in many different contexts. It is then enough, to define a value for that key. Whenever that key is used as a variable in the right context, at export of the datalog.ccmc it will be replaced with the corresponding value.

To use this feature, select the "Project settings" interface in the measurement task tree and navigate to the grid area. Access the column chooser (→4.3.1) and add the "Key" column to the grid area.



This will add a new column with the exact keys, that can be used. Now a corresponding value has to be defined for the key you which to use as variable. Click in the corresponding "Value" field for the desired key (e.g. "VehicleID" and type in the Value that shall be used (e.g. "Example Vehicle ID").

Name	Key	Value
Company name	Company	
Project name	Project	
Project manager name	AttendantName	
E-Mail address project manager	AttendantMail	
User	UserName	
E-Mail address user	UserMail	
Description	Description0	
Author	Author	
Department	Department	
Subject	Subject	
Chassis ID	Vehicle	
Vehicle ID	VehicleID	Example Vehicle ID
Description	Description1	

The term "VehicleID" can now be used as a variable in certain text fields and will be replaced with the value "Example Vehicle ID" in the datalog.ccmc.

The syntax for variables is the following:

Variable	Value written in data-log.ccmc
<Key>	"Value"
<VehicleID>	"Example Vehicle ID"



Wherever it is possible to use project parameters as variables, this manual will point that with an information box.

## 6 UPS (Uninterruptible power supply)

The UPS (Uninterruptible power supply) module provides the logger with power for a limited time in case of loss of external power. The UPS can be configured through the root element "UPS", which will also provide a list of status signals about the "UPS" module.

"UPS" status signals are largely treated in the same manner as a regular signal. They can be recorded over time, classed or processed; they can generate alarms or be displayed. Only they can't be directly stored in traces.



If a shut down occurs with no external power supply available, i.e. while the "UPS" is active, the "UPS" will cause an emergency shut down. That means, that the logger will shut down, but all configured data transfers will be ignored.

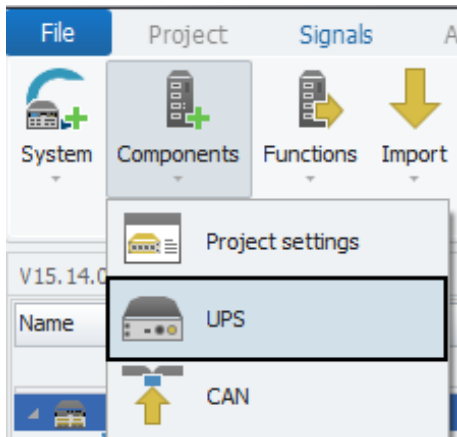
### 6.1 Storage method

In order to store "UPS" status signals use one of the following signal storage methods.

- AFX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))
- CAETEC binary (→ [14.13](#))
- CAETEC ASCII (→ [14.14](#))

## 6.2 Adding the UPS interface

In order to configure your “UPS” and make its signals accessible you will first need to add the “UPS” interface to your system. To do so, select the system in the measurement task tree (the topmost element in the tree), click the “Components” button in the Ribbon and then choose “UPS”.



## 6.3 Configuring the UPS interface

### 6.3.1 Tree elements for the UPS interface

Adding the “UPS” interface to your system will produce two new tree elements: the interface itself labeled “UPS” and its child element labeled “Status”.

ARCOS 1.5	0
UPS	0
Status	0

### 6.3.2 Details area for the UPS interface

Selecting the “UPS” interface in the tree allows you to access two tabs in the details area.

#### General

This tab allows you to activate or deactivate the “UPS” interface and thus to make its signals available for internal Recording and further use (e.g. triggers, formulas, display,...) It also allows you to give a user specific name to your signal if wished and add an additional description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

#### Settings

This tab contains settings regarding the “UPS”.

- **Charge threshold**

Define a threshold in percentage of the battery charge. If the state of battery charge falls below or is already below this threshold at loss of external power, the logger will



General Settings

Active:

Name: UPS

Description: Configuration of a connected UPS

Reference: UPS/ARCOS 1.5

General Settings

Charge threshold:  0 %

Time threshold:  0 s

Ignore duration:  0 s

shut down.

- **Time threshold**

Define a threshold in percentage of the remaining battery time. If the state of remaining battery time falls below or is already below this threshold at loss of external power, the logger will shut down.

- **Ignore duration**

If set, for the defined amount of time, the logger will ignore regular shutdown conditions and only perform an emergency shut down if either the "Charge threshold" or "Time threshold" are met.



If a shut down occurs with no external power supply available, i.e. while the "UPS" is active, the logger will perform an emergency shut down. That means, that the logger will shut down, but all configured data transfers will be ignored.

## 6.4 UPS signal properties

“UPS” status signals do not need to be imported, as they are continuously produced by the “UPS” itself. As soon as the “UPS” interface has been added to the system, they can then be activated in order to be used like regular signals in further processing, for example as triggers or in formulas.

### 6.4.1 Grid area for UPS signals

When selecting any of the “UPS” tree elements, the “Grid area” will present you with an overview of the available “UPS” status signals. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
☐	<input checked="" type="checkbox"/>						
▶ Power lost	<input type="checkbox"/>		0	1	0	1	1 Hz
Capacitator voltage	<input type="checkbox"/>	V	0	29	0	29	1 Hz
State of charge	<input type="checkbox"/>	%	0	100	0	100	1 Hz
Remaining runtime	<input type="checkbox"/>	s	0	10000	0	10000	1 Hz

### 6.4.2 Overview of UPS signals

Signal	Meaning	Unit and/or Possible Values
Power lost	Loss of external power	0 = No 1 = Yes
Capacitator voltage	Current voltage of the UPS capacitors	V 0-29
State of charge	Shows the state of charge in percentage of the battery capacity	% 0-100
Remaining runtime	Shows the remaining runtime of the “UPS” in seconds	s 0-10000

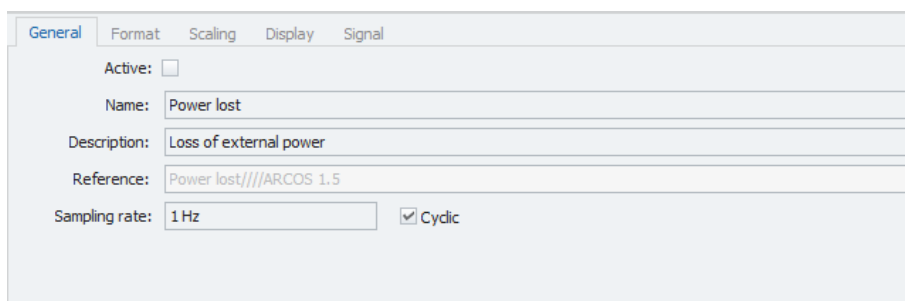
### 6.4.3 Details area for UPS signals

If a "UPS" status signal is selected in the grid area, the details area will additional tabs with settings regarding these signals. These settings will be explained in the following.

#### General

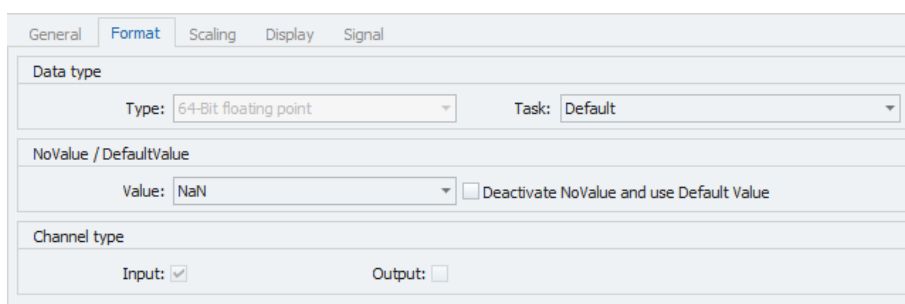
This tab allows you to activate or deactivate the signal by ticking/unticking the checkbox and thus to make it available for internal Recording and further use (e.g. triggers, formulas, display,...)

It also allows you to give a user specific name to your signal if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The "Sampling rate" allows you to set, how frequently a signal should be requested. The tickbox "Cyclic" allows you to switch between cyclic and event controlled sampling.



#### Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case "64-Bit floating point") and allows you to apply special tasks for this signal.

- **NoValue / DefaultValue**

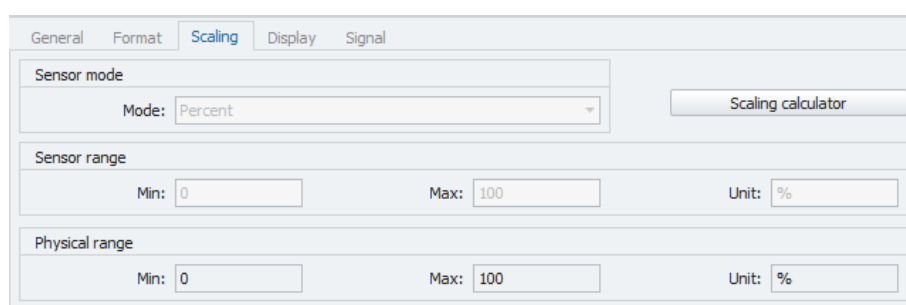
This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a “Input” channel or “Output” channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “Scaling calculator” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the [IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”](#).



The screenshot shows a software interface with a 'Scaling' tab selected. It contains three main sections: 'Sensor mode' with a dropdown menu set to 'Percent'; 'Sensor range' with input fields for 'Min: 0', 'Max: 100', and 'Unit: %'; and 'Physical range' with input fields for 'Min: 0', 'Max: 100', and 'Unit: %'. A 'Scaling calculator' button is located to the right of the 'Sensor mode' dropdown.

- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Percent” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**


Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the [IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”](#).

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

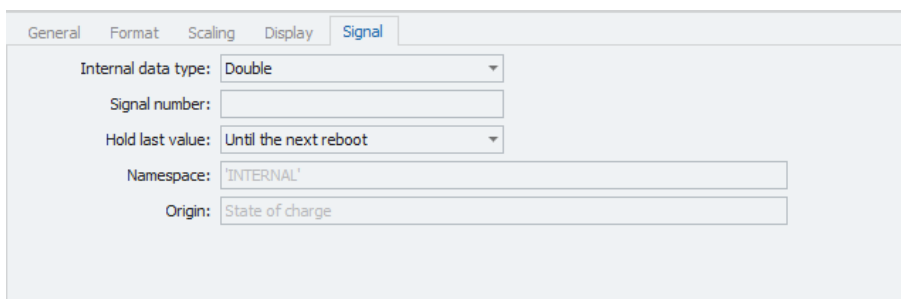
The dropdown menu "**Decimal places**" allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Signal

This tab allows you to define signal settings.



- **Internal data type**

Assign an internal data type to the signal. Available data types are "**Double**" and "**String**".

- **Signal number**  
Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".
- **Hold las value**  
Specify, for how long the last value of the signal will be hold.
- **Namespace**  
The "Namespace" serves as unique identifier for the signal inside the logger.
- **Origin**  
Tells the source of the signal. This can help identifying the source of a signal for which a user defined signal name has been set.

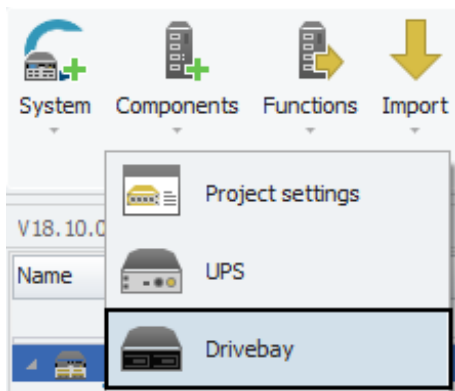
## 7 Drivebay (Arcos 1.5 only)

The “Drivebay” is an extension module only available for the Arcos 1.5 system. It provides your logger with two extraslots, that can be equipped with a 2.5 inch SSD (Solid State Disk) each, in order to expand your logger’s storage capacity.

The equipped SSDs can be defined as storage targets for datasets, so the dataset will be stored on the SSD. For instructions on how to do so please refer to the “Dataset settings” of the “Dataset” chapter (→ [14.1.3](#)).

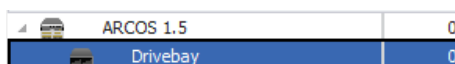
### 7.1 Adding the Drivebay interface

In order to use the equipped SSDs, you will need to add the “Drivebay” interface to your system. To do so, select the System in the measurement task tree, click the “Components” button in the ribbon and then choose “Drivebay”.



### 7.2 Tree elements for Drivebay

Once the “Drivebay” interface has been added to your system, it will appear as a new tree element in the measurement task tree.



## 7.3 Details area for Drivebay

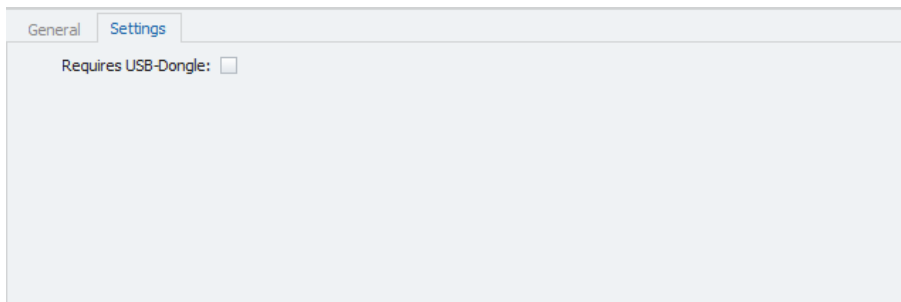
The Details area shows settings either for the “Drivebay” interface.

### General

Please refer to (→[4.2.2](#)).

### Settings

The “Settings” tab contains the “Requires USB-Dongle” setting. If this setting is activated, a USB-Dongle with a signed datalog.key is needed in order to eject an SSD from the “Drivebay” and remove it from the logger.





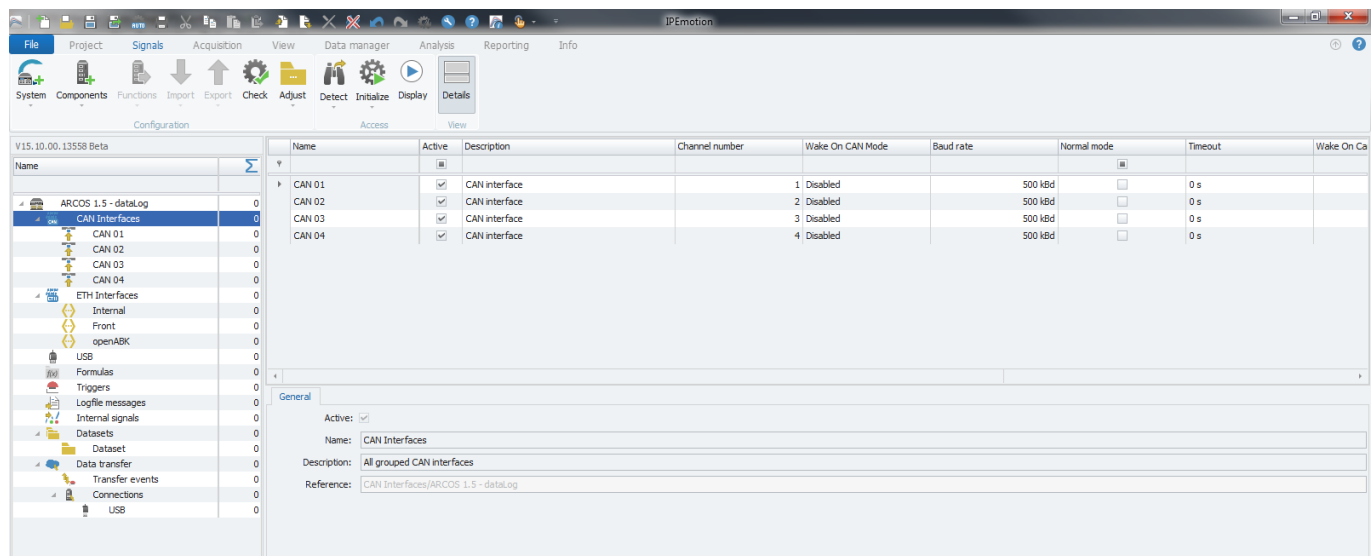
## 8 Signal Acquisition

### 8.1 CAN/CAN FD channels

All the CAN channels for your system are located in the tree element "CAN interfaces". There are two types of CAN channels, physical "CAN" channels and "Virtual CAN" channels.

According to the default settings, the tree element "CAN interfaces" will include a preset number of CAN channels. By clicking the tree element CAN Interfaces you will see all of its channels and signals in the grid area as well as a tab called **General** in the **Details area** which allows you to set a name and description. These settings apply to the entire element "CAN interfaces".

In the following will be described how to add CAN channels and adjust their settings (→8.1.3).

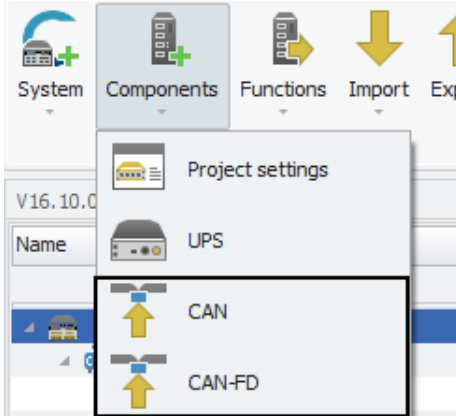


#### 8.1.1 Storage method

In order to store all incoming traffic on a CAN channel use a bus tracing method for storage. Please refer to (→ 14.8).

### 8.1.2 Adding CAN/CAN FD channels

CAN channels can be added by selecting the system in the tree, then clicking the “Components” button and finally choosing the desired type of CAN channel you wish to add.



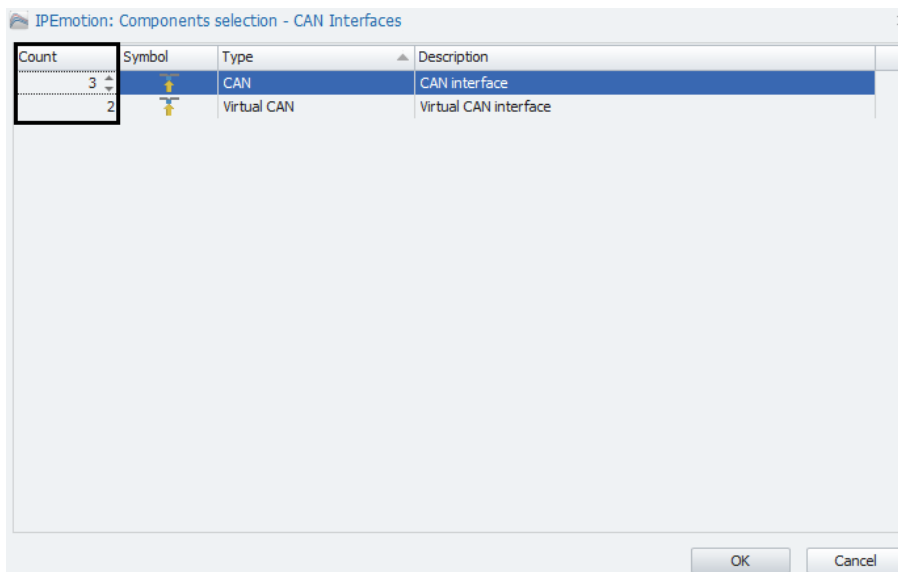
- **CAN/CAN FD**

Adds a CAN channel that corresponds to a physical CAN channel of your logger. For instructions on CAN settings refer to (→8.1.3).

- **Virtual CAN**

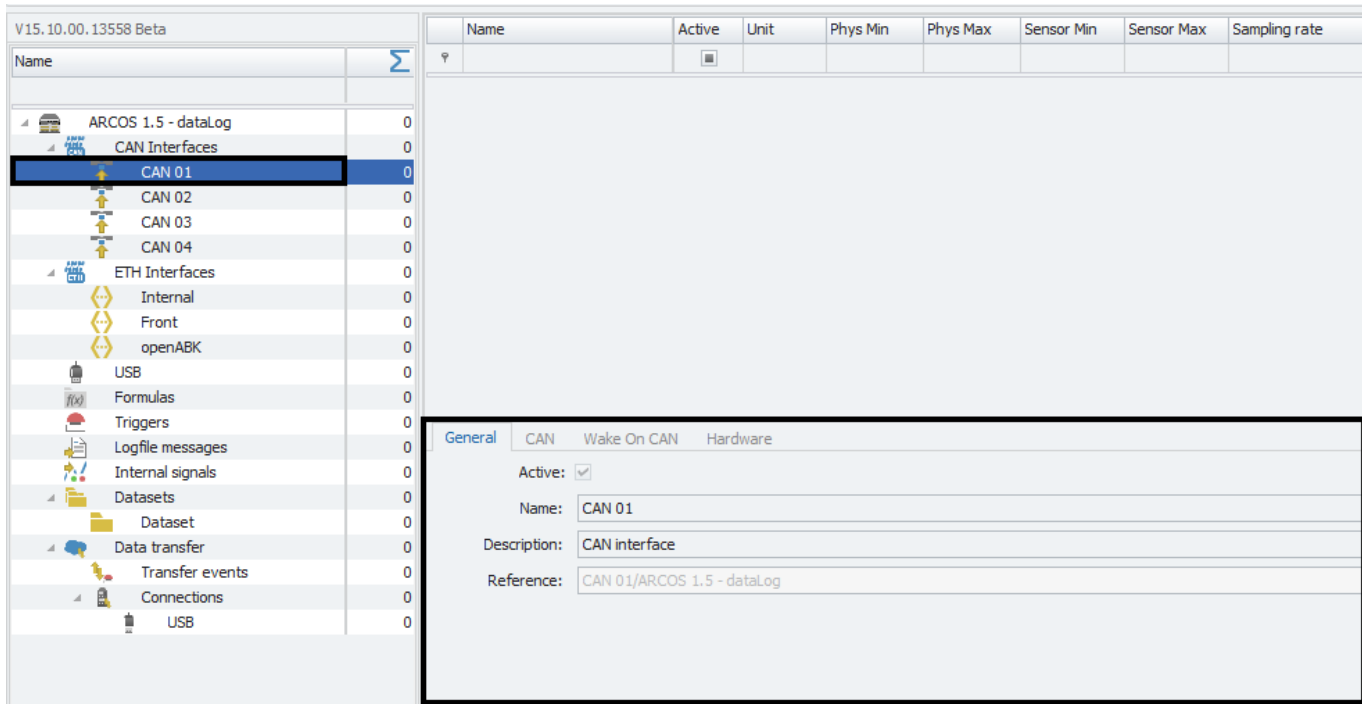
Adds a virtual CAN channel. For instructions on Virtual CAN settings refer to (→8.1.4).

**Multiple selection** Allows you to add multiple CAN channels of both types at the same time. To do so set the counter for each type to the desired number of channels that you wish to add as marked in the figure below.



### 8.1.3 CAN settings

By selecting one of the CAN channels in the tree you will be able to define this channel's settings in the details area.



Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
ARCOS 1.5 - dataLog	0						
CAN Interfaces	0						
CAN 01	0						
CAN 02	0						
CAN 03	0						
CAN 04	0						
ETH Interfaces	0						
Internal	0						
Front	0						
openABK	0						
USB	0						
Formulas	0						
Triggers	0						
Logfile messages	0						
Internal signals	0						
Datasets	0						
Dataset	0						
Data transfer	0						
Transfer events	0						
Connections	0						
USB	0						

General | CAN | Wake On CAN | Hardware

Active:

Name: CAN 01

Description: CAN interface

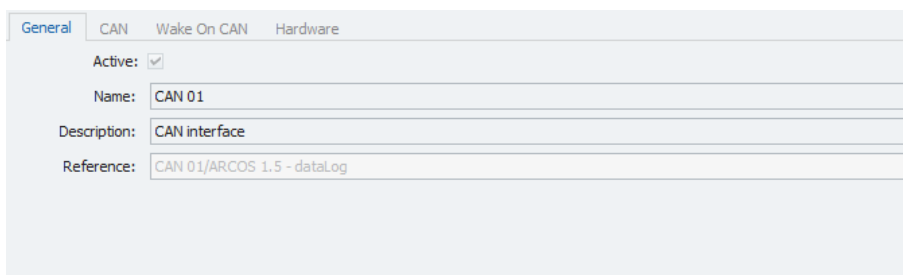
Reference: CAN 01/ARCOS 1.5 - dataLog



The same settings described in this section as part of the Details area can also be adjusted when selecting the tree element **CAN interfaces** and then directly changing the desired setting in the respective field of the **Grid area**.

#### 8.1.3.1 General

This tab allows you to give a user specific name for the selected CAN channel if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The **Active** checkbox is by default marked as activated and cannot be changed.



General | CAN | Wake On CAN | Hardware

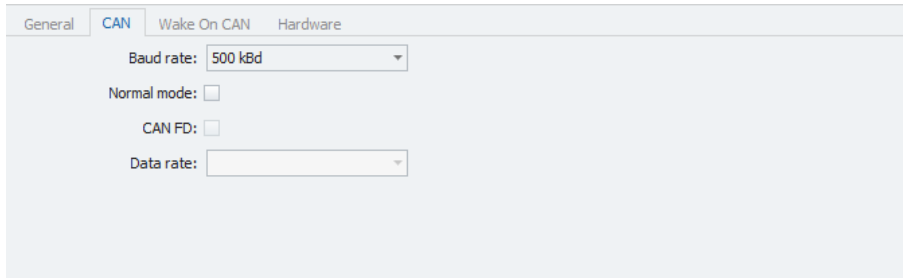
Active:

Name: CAN 01

Description: CAN interface

Reference: CAN 01/ARCOS 1.5 - dataLog

### 8.1.3.2 CAN



The screenshot shows a configuration window with four tabs: General, CAN, Wake On CAN, and Hardware. The CAN tab is selected. It contains the following controls:

- Baud rate: 500 kBd (dropdown menu)
- Normal mode:
- CAN FD:
- Data rate: (empty dropdown menu)

#### Baud rate

The dropdown menu Baud rate allows you to set the Baud rate for the selected CAN channel. The baud rate defines the speed in bits/second at which data can be transmitted through the CAN bus. The speed must be adapted to the source. Only if all the users on the bus are set at the same rate, is transfer possible.

#### Normal mode

The checkbox Normal mode allows you to determine whether the CAN channel may communicate or is in silent mode. If the checkbox is marked, the channel may communicate, if the checkbox is not marked, the channel is in silent mode.

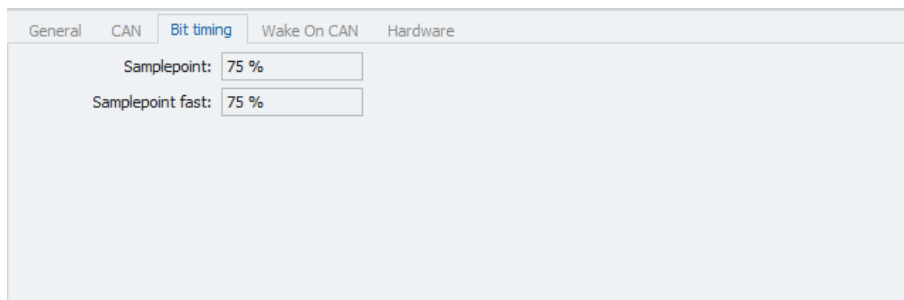
#### CAN FD

This option is only supported on CAN FD channels in the CAN FD interface. If activated, you may customize the fast data rate of the CAN FD channel.

## Data rate

This option is only supported on CAN FD channels in the CAN FD interface. If CAN FD is activated you may set the fast datarate for the channel here.

### 8.1.3.3 Bit timing



General	CAN	Bit timing	Wake On CAN	Hardware
Samplepoint: 75 %				
Samplepoint fast: 75 %				

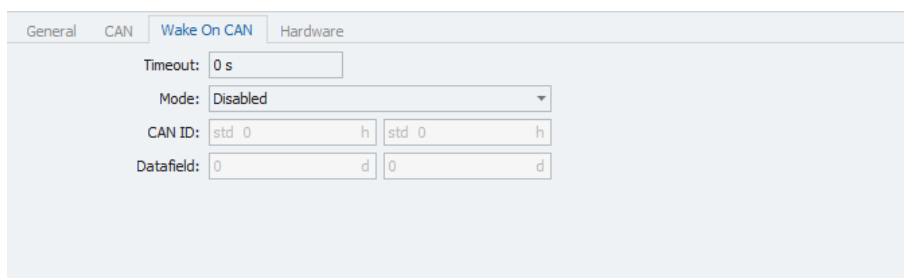
## Sample point

Sample point for normal data rate.

## Sample point fast (CAN FD only)

Sample point for increased data rate.

### 8.1.3.4 Wake On CAN



General	CAN	Wake On CAN	Hardware
Timeout: 0 s			
Mode: Disabled			
CAN ID: std 0 h		std 0 h	
Datafield: 0 d		0 d	

## Timeout

For Wake on CAN, timeout has a special significance. It defines how long a waking channel must be inactive to be recognized so and therefore allow for the logger to shut down. If timeout is recognized, an entry is made in the log file and an error message with an alert appears on the display, which has to be acknowledged.

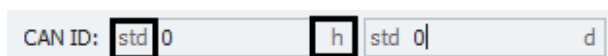
## Mode

This dropdown menu allows you to set the wake-up function for your selected CAN channel.

Wake on CAN type	Characteristics
Disabled	No start on CAN messages, lowest energy consumption.
Enabled	Start on a CAN message, with first messages lost; low energy consumption.
Enabled (no message lost)	Start on CAN message, with no message lost; slightly higher idle current.
Keep awake	The logger starts with Clamp 15, but only shuts down if all the awakening conditions (Clamp 15 and WakeOnX) are no longer fulfilled.

### CAN ID - Settings for starting on a specific CAN ID

The CAN ID consists of two fields in the "Wake on CAN" tab of the CAN channel's details area: The "CAN identifier" on the left and the "CAN ID bitmask" on the right.



The aim of this setting is to wake the logger with a message having a specific ID (or group of IDs), regardless of what the content of the message is. In order to do so, you can define a "CAN identifier" and a "CAN ID bitmask" to limit the identifier. Both parameters are used in their binary form. The "CAN ID bitmask" defines (or masks) the bit positions of a message ID that are to be applied. The "CAN identifier" specifies the contents that must exist at these bit positions in order for a start to take place.

Both fields can be defined as "standard CAN ID" or "extended CAN ID" by clicking the button in the left corner of the field. Both fields can process and show the entered number in its decimal or hexadecimal form. To switch between decimal or hexadecimal just click the button in the right corner of the field as marked in the figure above.



The decimal number is processed in the logger in its binary form and if the number set for the "CAN ID bitmask" = 0, the logger will start on any message.

*Example:*

*CAN identifier = 22 (binary = 10110)*

*CAN ID bitmask = 28 (binary = 11100)*

*In other words, the mask (CAN ID bitmask) specifies that, to start the logger, given values are expected at the positions Bit2, Bit3, Bit4. All other values are irrelevant ("x").*

*The CAN identifier calls for Bit2 to be "1", Bit3 to be "0" and Bit4 to be "1".*

*The following table lists the positions for an 11-bit CAN identifier (in red the masked positions, the numbers signifying the expected values).*

## 8.1 CAN/CAN FD CHANNELS

Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
x	x	x	x	x	x	1	0	1	x	x

Example of values of a CAN ID that would start the logger in this configuration:

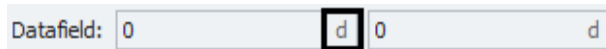
Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	CAN-ID (dec)
		1	0	0	0	1	0	1	0	1	277
				1	0	1	0	1	1	1	87
1	0	0	0	0	0	1	0	1	0	0	1044
			1	1	1	1	0	1	1	1	247
			1	0	0	1	0	1	1	0	150
						1	0	1	0	0	20
						1	0	1	0	1	21

Values of a CAN ID that would **NOT** start the logger in this configuration (example):

Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	CAN-ID (dec)
		1	0	0	0	1	1	0	0	1	281

### Datafield - Settings for starting on a specific message value of a CAN ID

The Datafield consists of two fields in the "Wake on CAN" tab of the CAN channel's details area: The "Datafield content" field, which will contain the "Datafield content value" on the left and the "Datafield bitmask" on the right.



The aim of this setting is to wake the logger with a given content of a message with a particular ID (or a group of IDs).

After defining the ID that is supposed to start the logger, you can follow the above pattern to additionally specify which value within the message of the ID is to be an additional requirement for start. The entire 64-bit message is considered, single bits of which can be defined as start conditions. (In order to specify the values, it is necessary to decode the binary structure of the message. There is no DBC file assistance available here.)

A "Datafield bitmask" defines (masks) which bit positions of a message are to be applied. A "1" marks the bit positions to be used. Positions coded "0" are ignored. The "Datafield content value" specifies the contents checked in the mask by the logger. If "Datafield content value" writes "1" (or "0") in the positions of the mask (Datafield bitmask), then the mask positions of the ID must contain the identical values; i.e. "1" (or "0") too, otherwise the logger is not started.

Both fields can process and show the entered number in its binary, decimal or hexadecimal form. To switch between binary, decimal or hexadecimal just click the button in the right corner of the field as marked in the figure above.



The decimal number is processed in the logger in binary form and if the number set for the “Datafield bitmask” = 0, the logger will start on any message. When the number entered in the “Datafield bitmask” is considered in binary form, “1” defines a bit position that is considered in filtering, “0” means filtering ignored this bit position.

*Example:*

Wake on CAN trigger raw value = 22 (binary = 10110)

Wake on CAN raw value mask = 20 (binary = 10100)

The mask “Datafield bitmask” specifies that, the logger is only started if certain values are found at the positions Bit2 and Bit4. All other values are irrelevant (“x”).

“Datafield content value” calls for Bit2 to be “0” and Bit4 to be “1”.

The following table lists the positions for a 64-bit message (with the masked positions colored red and the numbers signifying the expected values).

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
x	x	x	1	x	0	x	x

Message values that would start the logger in this configuration (example):

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Message value (dec)
			1	0	0	0	1	17
	1	0	1	1	0	1	1	27
			1	0	0	0	0	16
1	1	1	1	1	0	1	1	251
1	0	0	1	0	0	1	0	146
			1	1	0	0	0	24

Negative example of message values that would NOT start the logger in this configuration:

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Message value (dec)
0	0	0	1	1	1	0	0	28

### Settings for starting on a specific signal

The settings described in the previous paragraphs apply to entire messages. This section explains how to apply these settings to a specific signal contained in a message. It describes the procedure for deriving the required WoC parameters from the physical value of a signal. Since a simple formula such as “signal > 30” is not possible, it is necessary to define the start condition at the bit level. The numeric format is important, as well. The order in which the bit positions are counted depends on the numeric format (e.g., Intel or Motorola). Please bear in mind, when selecting the signal and a particular start value, that the start condition must be met during the entire measurement run. Remember: The start signal is also the stop signal. If the start signal is missing for a set period of time,

Changes and errors excepted.



## 8.1 CAN/CAN FD CHANNELS

the logger is stopped. This makes state bits good start signals. Signals such as temperature signals that generally fluctuate, are only suitable providing the definition of the start condition is sufficiently vague. Here vague means that not a specific bit combination switches on the logger, but that a range of bit combinations is possible. When defining filters, be sure to avoid gaps between the significant mask positions (marked), otherwise the covered range of values will also have gaps (see Filter 4), which would shut down the logger. Several filters serve as examples below. The table shows which values start the logger with which filter. In the column for each filter, these values are marked with an "X".

### Filter 1:

Datafield content value = 16 (binary = 10000)

Datafield bitmask = 24 (binary = 11000)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
x	x	x	1	0	x	x	x

### Filter 2:

Datafield content value = 24 (binary = 11000)

Datafield bitmask = 24 (binary = 11000)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
x	x	x	1	1	x	x	x

### Filter 3:

Datafield content value = 16 (binary = 10000)

Datafield bitmask = 16 (binary = 10000)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
x	x	x	1	x	x	x	x

### Filter 4: (negative example)

Datafield content value = 18 (binary = 10010)

Datafield bitmask = 18 (binary = 10010)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
x	x	x	1	x	x	1	x

Dec. value	Binary value	Filter 1	Filter 2	Filter 3	Filter 4
15	000001111				
16	0000010000	x		x	
17	0000010001	x		x	
18	0000010010	x		x	x
19	0000010011	x		x	x
20	0000010100	x		x	
21	0000010101	x		x	
22	0000010110	x		x	x
23	0000010111	x		x	x
24	0000011000		x	x	
25	0000011001		x	x	
26	0000011010		x	x	x
27	0000011011		x	x	x
28	0000011100		x	x	
29	0000011101		x	x	
30	0000011110		x	x	x
31	0000011111		x	x	x
32	0000100000				

The following example of a temperature in Intel format illustrates how to derive the filter settings.

### Example

With the logger operating in the temperature range of approx. 30 - 40 °C, the following signal definition is given:

Deriving Wake on CAN "Datafield bitmask" and Wake on CAN "Datafield content value":

First you use the limit values to derive the raw value and thus the binary value of the operating range.

$$\text{Datafieldcontentvalue}_{30\text{degree}} = \frac{\text{value} - \text{Offset}}{\text{Scale}} = \frac{30 - (-10)}{0.1} = 400 = \text{Binary} : 110010000$$

$$\text{Datafieldcontentvalue}_{40\text{degree}} = \frac{\text{value} - \text{Offset}}{\text{Scale}} = \frac{40 - (-10)}{0.1} = 500 = \text{Binary} : 111110100$$

The highest bit positions that are identical for both values, without a gap, are the two on the left, which have the value 1. This yields, as mask and filter binary: 1 1000 0000

So the lower bound for the Datafield content value is:

- binary 1 1000 0000
- decimal 384
- physical 28,4°C

## 8.1 CAN/CAN FD CHANNELS

So the upper bound for the Datafield content value is:

- binary 1 1111 1111
- decimal 511
- physical 41,1°C

Since the value has 9, but the signal 16 Bit, the 7 most significant Bits have to be 0. This is achieved by the Datafield bitmask.

Filter:

Wake on CAN Datafield content value = 384 (binary = 0000 0001 1000 0000)

Wake on CAN Datafield bitmask = 65408 (binary = 1111 1111 1000 0000)

Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1	1	X	X	X	X	X	X	X

Since, however, the signal lies in the middle of the 64 bit-wide message (bit offset=16), the bit positions to the right of the signal in the message must be filled with "0" (left is equal to Bit0 in the CAN-traffic).

	Bit-Offset				T_Aussen				Don't care								
Binary	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
Intel:	LSB				MSB												
Datafield content value:	0000	0000	0000	0000	0000	1000	0001	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
Hex:	0	0	0	0	0	8	1	0	0	0	0	0	0	0	0	0	0
Datafield bitmask:	0000	0000	0000	0000	0000	1000	1111	1111	0000	0000	0000	0000	0000	0000	0000	0000	0000
Hex:	0	0	0	0	0	8	F	F	0	0	0	0	0	0	0	0	0

So the Datafield content value is equal to:

Datafield content value (hex) = 0xFF800000

Datafield content value (dec) = 4,286,578,688

So the mask is equal to:

Datafield bitmask (hex) = 0x1800000

Datafield bitmask (dec) = 25,165,824

Wake on CAN Datafield content value = Wake on CAN Datafield bitmask

**Deriving Wake on CAN "CAN identifier" and "CAN ID bitmask":**

The precise ID of the message is to be found – i.e. all the positions of the ID that are not leading zeros are significant. All the significant positions are to be assigned the value of the ID.

This means:

**CAN identifier = Message D = 100**

Deriving the significant bit positions of the ID:

ID (dec) = 100 => ID (bin) = 1100100

The ID also has 7 significant positions. So the mask must also be 7 bits long. All positions must be assigned "1".

**"CAN ID bitmask" = 1111111 = 127 (dec)**

Each node can thus be assigned individual start settings. Remember: During the booting phase, at the node that started the logger, the WoC LED on the front plate of the interface blinks quickly for approximately ten seconds.

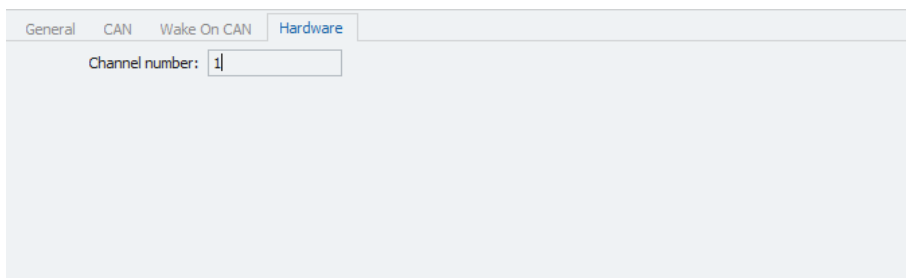


Even if WoC (Wake on CAN) is set as start condition the logger starts on clamp 15. So start on clamp 15 has priority over WoC. This is necessary to have a fallback solution in case a start message or start value can no longer be sent.

If the logger was started with clamp 15, it can likewise only be shut down by “pulling” Clamp 15 – based on the principle, the source that starts, also stops – providing there is not simultaneously another wake condition (e.g. WoC) active.

### 8.1.3.5 Hardware (Channel number)

This tab allows you to set a **Channel number** for the selected CAN channel. This channel number has to be unique within the CAN interface.



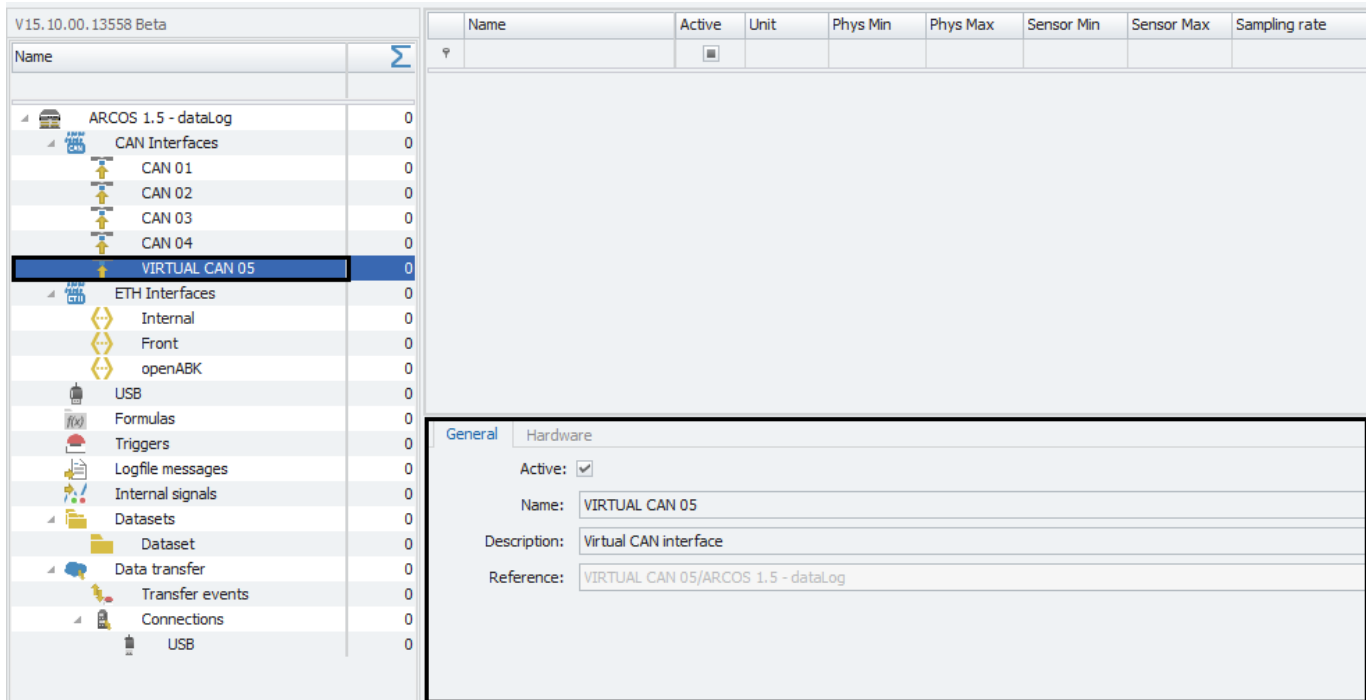
The screenshot shows a software interface with four tabs: 'General', 'CAN', 'Wake On CAN', and 'Hardware'. The 'Hardware' tab is selected. Below the tabs, there is a label 'Channel number:' followed by a text input field containing the number '1'.



For better orientation and in order to avoid confusion regarding Channelnumbers and -names, a Channels physical number can be found in the logger’s **“Web Interface”** and set accordingly.

## 8.1.4 Virtual CAN settings

By selecting one of the Virtual CAN channels in the tree you will be able to define this channel's settings in the details area.



Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
ARCOS 1.5 - dataLog							
CAN Interfaces							
CAN 01							
CAN 02							
CAN 03							
CAN 04							
<b>VIRTUAL CAN 05</b>							
ETH Interfaces							
Internal							
Front							
openABK							
USB							
Formulas							
Triggers							
Logfile messages							
Internal signals							
Datasets							
Dataset							
Data transfer							
Transfer events							
Connections							
USB							

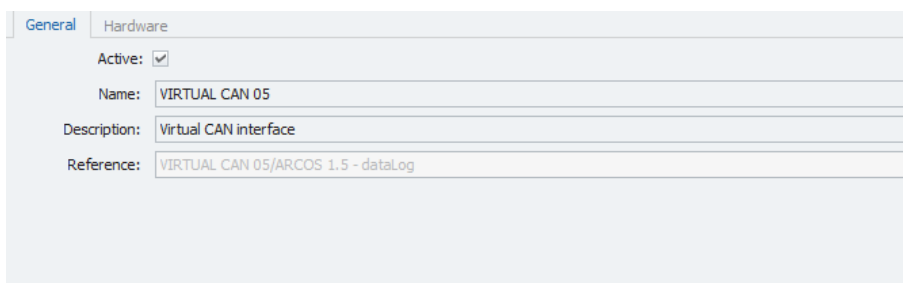
General	Hardware
Active:	<input checked="" type="checkbox"/>
Name:	VIRTUAL CAN 05
Description:	Virtual CAN interface
Reference:	VIRTUAL CAN 05/ARCOS 1.5 - dataLog



The same settings described in this section as part of the Details area can also be adjusted when selecting the tree element **CAN interfaces** and then directly changing the desired setting in the respective field of the **Grid area**.

### 8.1.4.1 General

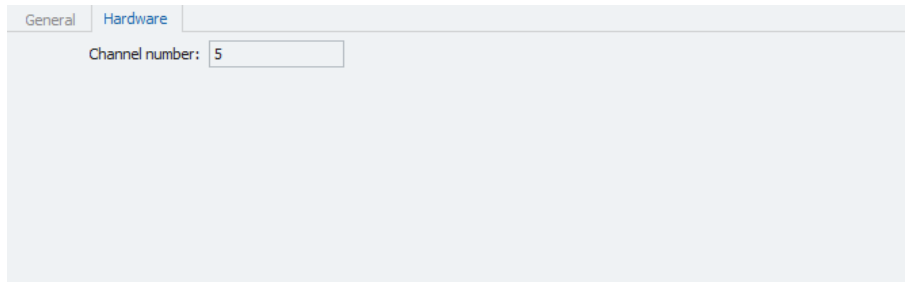
This tab allows you to give a user specific name to your for the selected Virtual CAN channel if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The **"Active"** checkbox allows you to activate or deactivate the selected Virtual CAN channel.



General	Hardware
Active:	<input checked="" type="checkbox"/>
Name:	VIRTUAL CAN 05
Description:	Virtual CAN interface
Reference:	VIRTUAL CAN 05/ARCOS 1.5 - dataLog

### 8.1.4.2 Hardware (Channel number)

This tab allows you to set a **Channel number** for the selected Virtual CAN channel. This channel number has to be unique within the CAN interface.



For better orientation and in order to avoid confusion regarding Channel numbers and -names, a Channels physical number can be found in the logger's **"Web Interface"** and set accordingly.

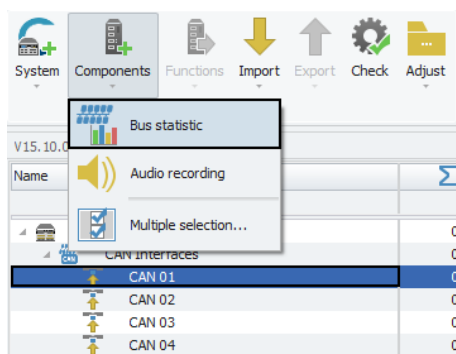
### 8.1.5 CAN channel Bus statistic

The "Bus statistic" provides a range of statistics and status signals for the respective CAN channel. It contains information on the current state of the Bus, the Busload, as well as on the messages that have been received and errors that occurred.



The "Bus statistic" only shows statistics for the channel to which it belongs. For each channel that you would like to see the statistic, you will have to add the component "Bus statistic".

**8.1.5.1 Adding Bus statistics** Select the channel in the tree for which you would like to add "Bus statistic", then click the "Components" button in the Ribbon and choose "Bus statistic".



**8.1.5.2 Bus statistic signals** Once the component “Bus statistic” has been added to your channel, it will appear in the measurement task tree as a child element of this channel and the grid area will give you an overview of the available signals.

The signals included in “Bus statistics” are of the type “Internal signal” and may be adjusted in the same way. For more information on “Internal signals” please refer to (→[8.22](#)).

### Overview of signals

Subtype	Meaning	Unit
Controller state	nan= Channel not available 1= Bus on 2= Bus warning 3= Bus off	-
Busload (%)	Bus load of a CAN/LIN channel	(%)
Number of messages	Number of messages since beginning of measurement	-
Message rate total	Current bus load	(frames/s)
Number of messages with standard ID	Number of messages with standard ID	-
Number of messages with extended ID	Number of messages with extended ID	-
Number of messages with remote standard ID	Number of messages with remote standard ID	-
Number of messages with remote extended ID	Number of messages with remote extended ID	-
Number of error frames	Number of error frames	-
Message rate of standard IDs	Messages with standard ID	(frames/s)
Message rate of extended IDs	Messages with extended ID	(frames/s)
Message rate of standard IDs	Messages with remote standard ID	(frames/s)
Message rate of extended IDs	Messages with remote extended ID	(frames/s)
Error frame rate	Average of errors per second	(frames/s)

## 8.2 CAN signals

### 8.2.1 Storage method

In order to store incoming signals on a CAN channel use one of the following signal storage methods.

- ATEX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

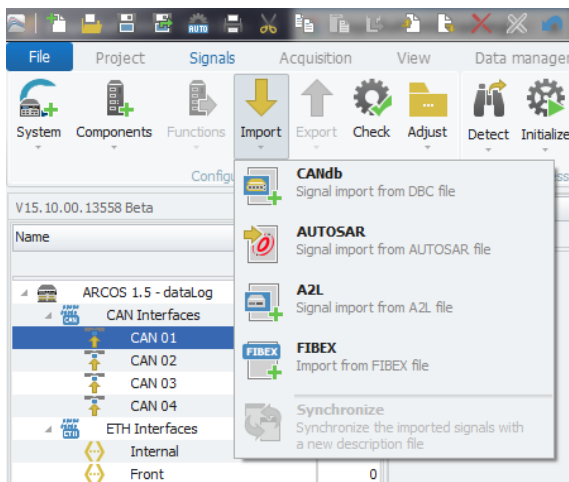
### 8.2.2 Importing CAN signals

This section explains how to import CAN signals. There are three different filetypes which can be used in order to import a single CAN signal or a group of CAN signals:

**CANdb (DBC file), Autosar and Fibex.**

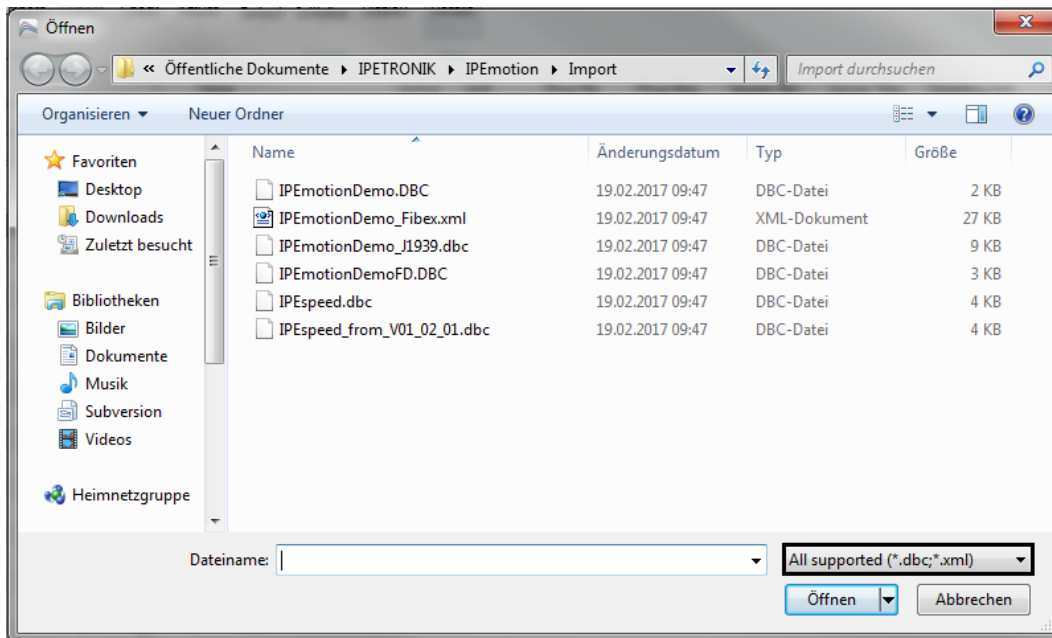
The import procedure for all of these filetypes is the same and will be explained using the example of the CANdb import.

To import Signals, select the CAN channel to which you wish to import your signal in the tree, click the **“Import”** button in the ribbon and then choose which filetype the **“description file”**, you wish to use for the import, has. For more information on the **“description file”** refer to (→[8.2.4.1](#)).

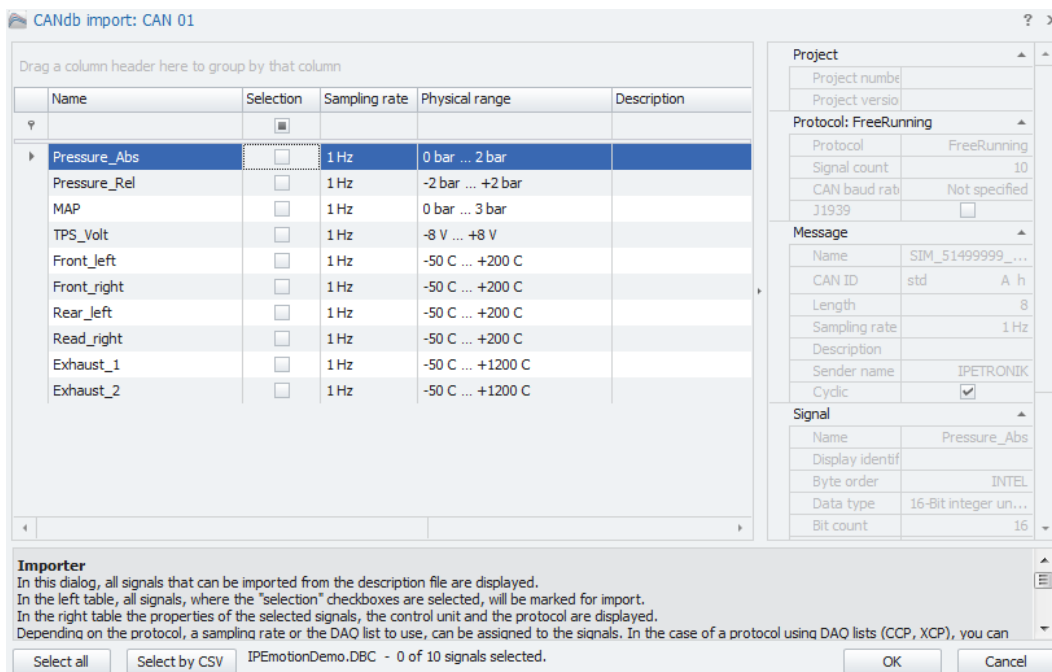




The following window lets you choose which file you wish to import. According to the file-type you have chosen earlier for your import, you will now be able to choose files of the respective filetype. The dropdown menu on the bottom right of the window shows you, which filetypes are available. Choose the file you wish to import and click "Open".



Once you have opened your file, the "Importer" window will appear, that will present you with a range of importing options.



In this dialog, all signals that can be imported from the description file are displayed. In the left table, all signals, where the "selection" checkboxes are selected, will be marked for import. You can either choose manually, which signals to import, you can use the "Select/Deselect all" button on the bottom left, or you can use a CSV-file to determine which signals are to be imported, by clicking "Select by CSV" on the bottom left.

In the right table the metadata, properties of the selected signals, the control unit and the protocol are displayed.

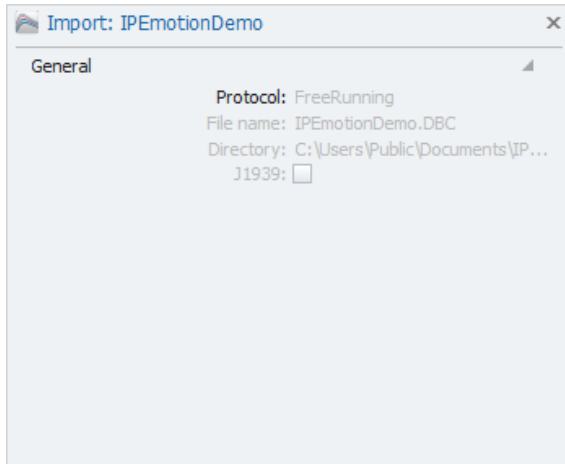
Once you have chosen all the signals you wish to import, click **"OK"** to complete the import procedure.



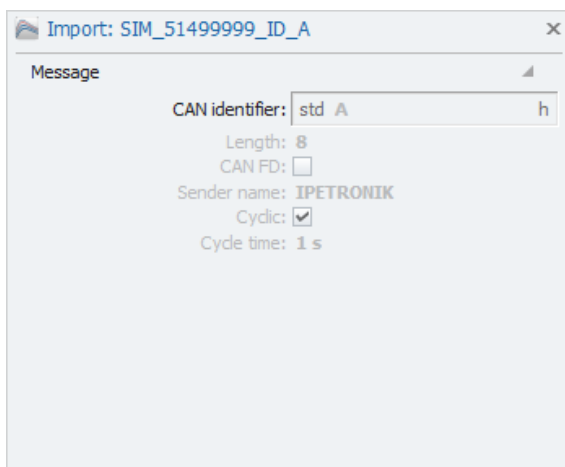
Multiple description files can be imported into the same CAN channel.

### 8.2.3 Import properties

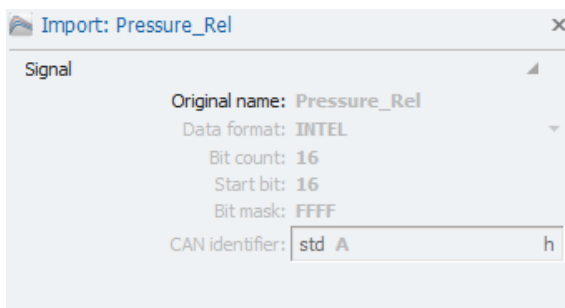
The “Import properties” of a description file, Message or signal allow you to see certain properties such as the Data format, The CAN identifier, the Bit mask, the start bit, bit count and more. It shows the signal’s properties as described by the description file.



Example for “Import properties” of a CAN description file



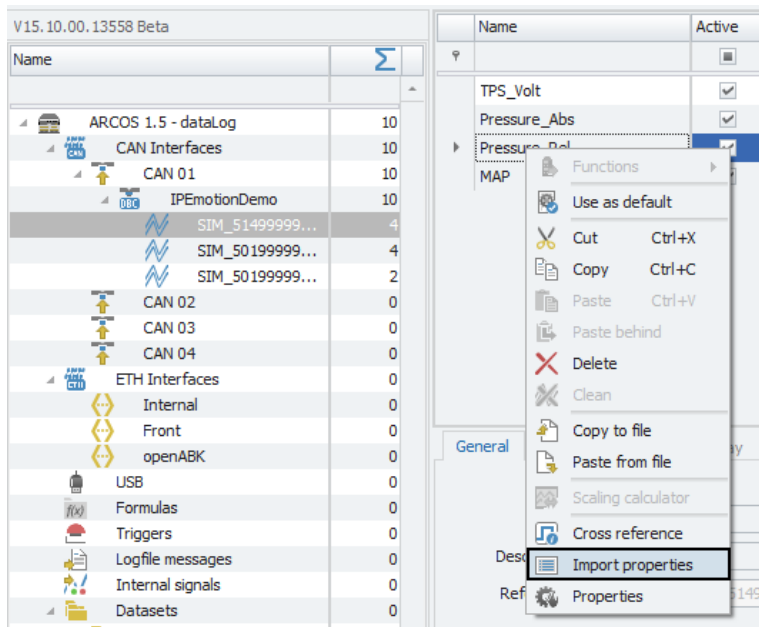
Example for “Import properties” of a CAN Message



Example for “Import properties” of a CAN signal

## 8.2 CAN SIGNALS

To access the “Import properties” rightclick on any desired description file, Message or signal and then choose “Import properties” from the resulting context menu.



## 8.2.4 Signal properties







### 8.2.4.1 Tree elements for CAN signals

After having successfully imported the desired signals to your CAN channel, this channel will contain two new layers of child elements in the measurement task tree: The “**Description file**” and the “**Message**”.

#### Description file

The “description file” is a database file which contains signal information and can be used to import those signals into a Signal channel in IPEmotion. The filetypes which are supported by the CAETEC Plugin for IPEmotion depend on the type of signal you wish to import.

The symbol in the left part of the tree element shows you the type of “description file” you imported (in this case a “DBC” file), then follows the name of the imported “description file” (in this case “IPEmotionDemo”) and on the right is a number indicating how many signals the “description file” contains (in this case 10).

Name		
ARCOS 1.5 - dataLog		10
CAN Interfaces		10
CAN 01		10
DBC IPEmotionDemo		10
SIM_51499999...		4
SIM_50199999...		4
SIM_50199999...		2
CAN 02		0
CAN 03		0
CAN 04		0

## Message

Each “description file” can contain one or more “Messages” (in this case 3), which then contain the actual signals. A “Message” can be found in the “Measurement task tree” as a child element of the “description file”, it belongs to.

Each “Message” can, again, contain one or more signals (in this case the three “Messages” contain 4, 4 and 2 signals), which is indicated by the number on the right of the “Message’s” name.

Name	
ARCOS 1.5 - dataLog	10
CAN Interfaces	10
CAN 01	10
IPEmotionDemo	10
SIM_51499999...	4
SIM_50199999...	4
SIM_50199999...	2
CAN 02	0
CAN 03	0
CAN 04	0

### 8.2.4.2 Grid area for CAN signals

In the “grid area” you will be presented with an overview of your selected CAN channel’s signals. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

### 8.2.4.3 Details area for CAN signals

The Details area shows settings either for the selected tree element (“description file” or “Message”) or the selected signal in the grid area. In case a tree element is selected, the details area will only show the “General” tab. Please refer to (→4.2.2).

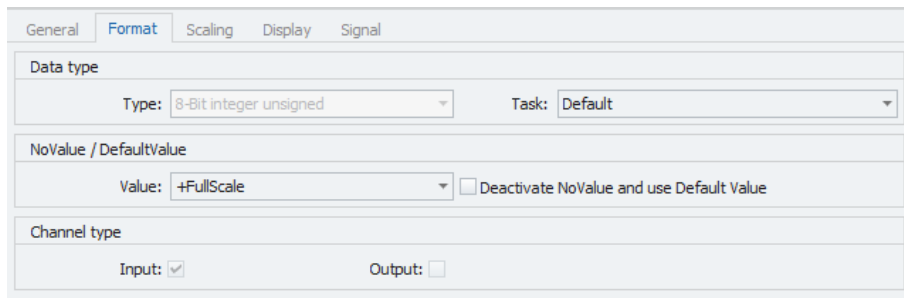
In case a signal is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

Please refer to (→4.2.2).

## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case **“8-Bit integer unsigned”**) and allows you to apply special tasks for this signal such as **“GPS Longitude”**, **“GPS Latitude”**, **“UTC hour”**, **“Audio mono”** and more.

- **NoValue / DefaultValue**

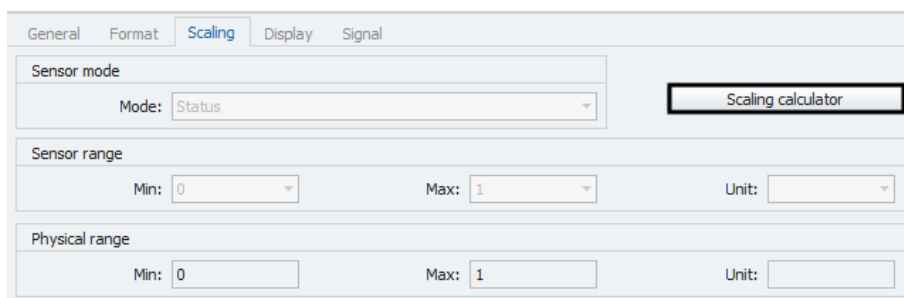
This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a **“Input”** channel or **“Output”** channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The **“Scaling calculator”** allows for more refined scaling options with a large range of functions. For details on how to use the **“Scaling calculator”** please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as "Status", "Voltage", "Frequency" or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

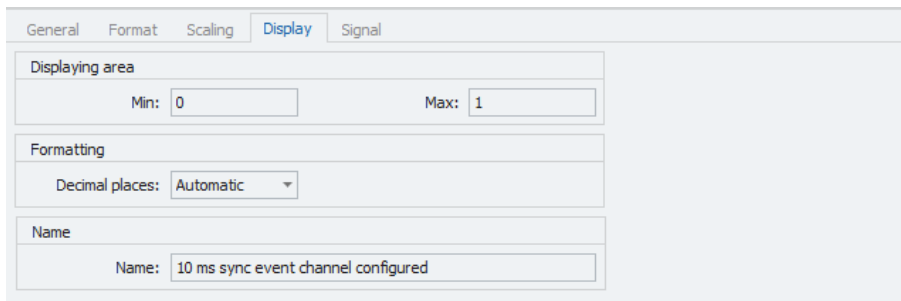
Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to "scale" your signal and also define the unit to use. For more refined scaling please use the "Scaling calculator" and refer to the IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling".

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



The screenshot shows a software interface with five tabs: General, Format, Scaling, Display, and Signal. The 'Display' tab is active. It contains three sections: 'Displaying area' with 'Min: 0' and 'Max: 1' input fields; 'Formatting' with a 'Decimal places' dropdown menu set to 'Automatic'; and 'Name' with an input field containing the text '10 ms sync event channel configured'.

- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

The dropdown menu "Decimal places" allows you to set how many decimal numbers of the value will be shown on a display.

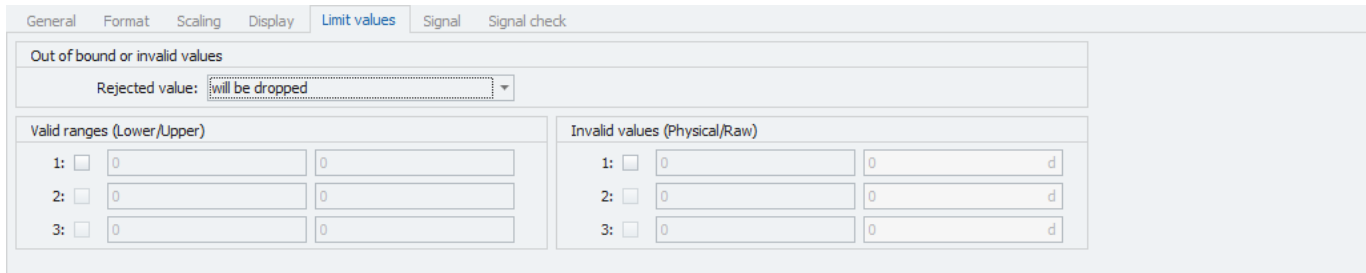
- **Name**

Allows you to set a Name to be shown on a display.



## Limit values

This tab allows you to define limit values for a signal and what action to take upon a limit value violation.



- **Rejected value**

Define what happens to a value, that has been rejected because it is out of bound or invalid. By default this value will be dropped. It can also be written as NaN.

- **Valid ranges (Lower/Upper)**

Define up to three ranges of valid signal values. Activate a range in order to define its upper/lower value (datatype double). Range 2 can only be activated if range 1 is and range 3 can only be activated if range 2 is.

- **Invalid values (Physical/Raw)**

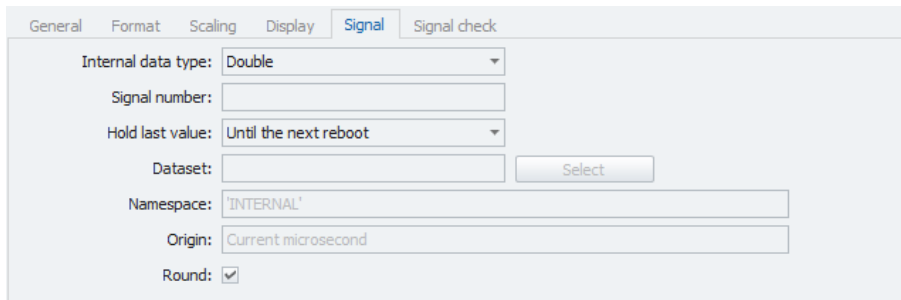
Define up to three invalid values. Activate an invalid value in order to define the physical value (datatype double) or raw value (datatype integer). If one of the two has been typed in, the other will be calculated according to the scale/offset settings in the scaling calculator.

Invalid value 2 can only be activated if invalid value 1 is and invalid value 3 can only be activated if invalid value 2 is.

For details on how to use the "Scaling calculator" please refer to the [IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling"](#).

## Signal

This tab allows you to define signal settings.



- **Internal data type**

Assign an internal data type to the signal. Available data types are “Double” and “String”.

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn’t send data for the specified time period, the value of the signal is set to “NaN (Not a Number)” and will be displayed as “-” in a display.

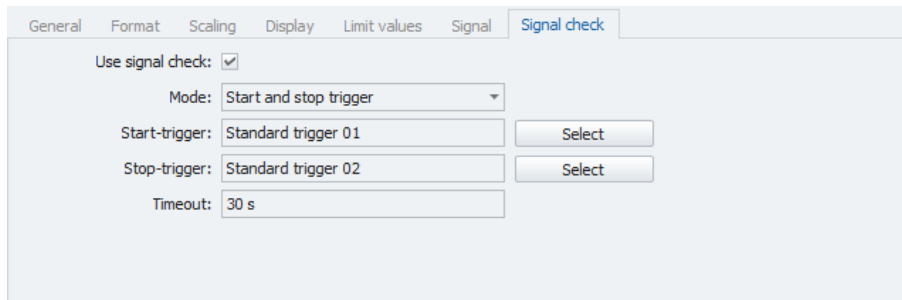
- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.

## Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



The screenshot shows a software interface with a tabbed menu at the top: General, Format, Scaling, Display, Limit values, Signal, and Signal check. The 'Signal check' tab is active. Below the tabs, there are several configuration options:

- 'Use signal check:' with a checked checkbox.
- 'Mode:' with a dropdown menu set to 'Start and stop trigger'.
- 'Start-trigger:' with a text box containing 'Standard trigger 01' and a 'Select' button.
- 'Stop-trigger:' with a text box containing 'Standard trigger 02' and a 'Select' button.
- 'Timeout:' with a text box containing '30 s'.

## 8.3 CCP/XCP signals

### 8.3.1 Storage method

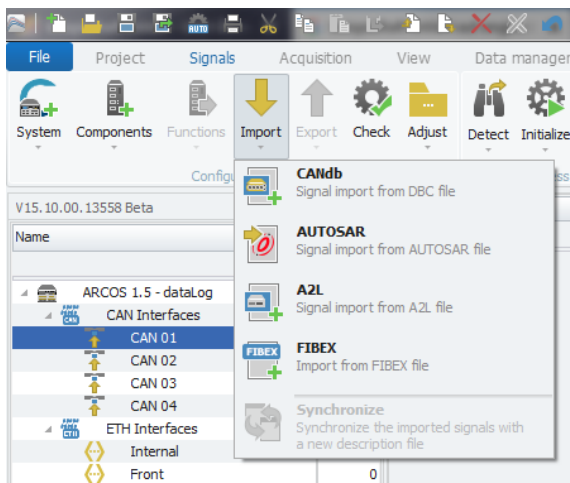
In order to store incoming signals on a CAN channel use one of the following signal storage methods.

- ATEX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

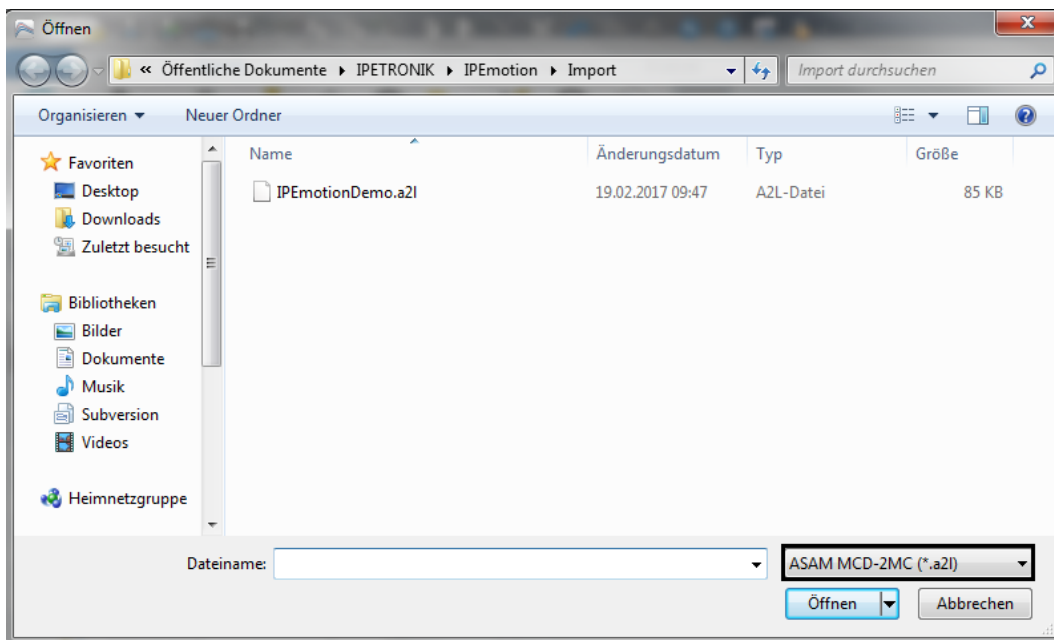
### 8.3.2 Importing CCP/XCP signals

This section explains how to import CCP or XCP signals. The filetype associated with these signals is a "A2L file".

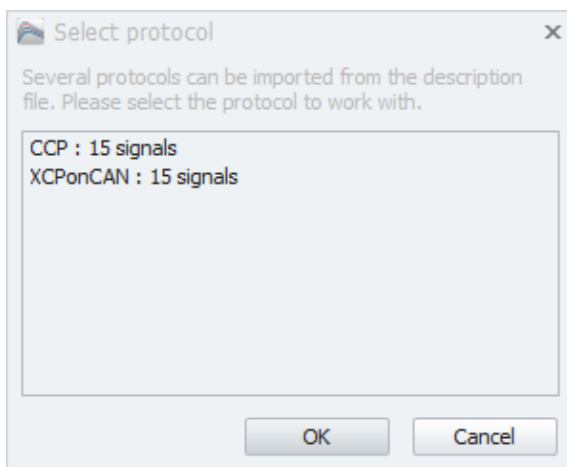
To import a CCP/XCP signal, select the CAN channel to which you wish to import your signal in the tree, click the "Import" button in the ribbon and then choose the "A2L" filetype for the import.



The following window lets you choose which file you wish to import. According to the file-type you have chosen earlier for your import, you will now only be able to choose files of the "A2L" filetype. Choose the file you wish to import and click "Open".



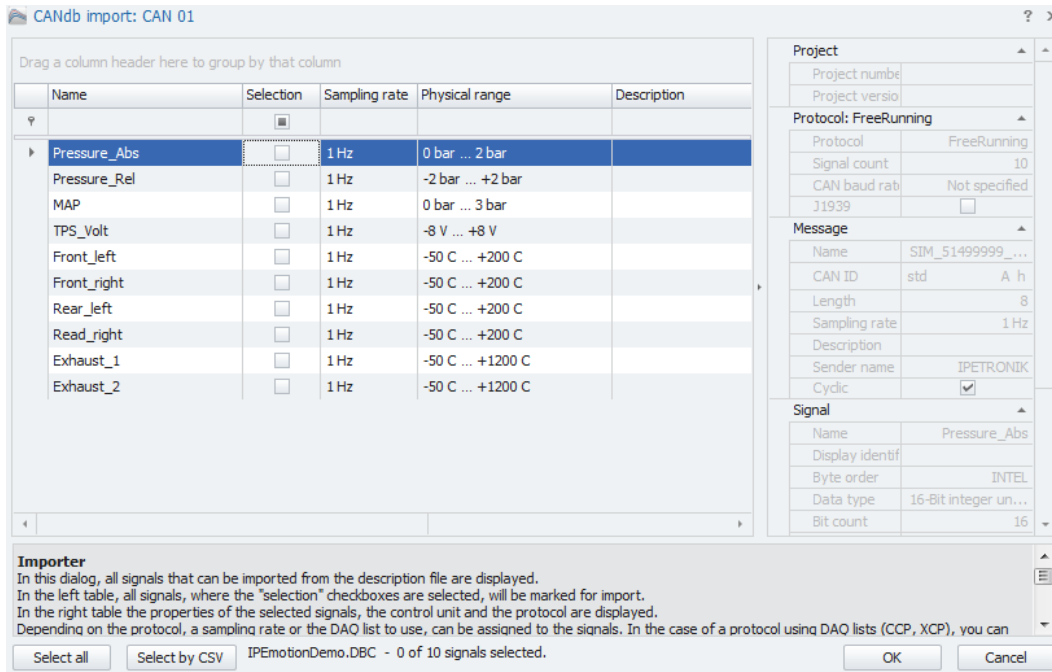
The following window lets you choose whether you want to import a "CCP" signal or a "XCP" signals. Choose the protocol you wish to import and click "OK".



Once you have chosen the protocol and confirmed, the "Importer" window will appear, that will present you with a range of importing options.

In this dialog, all signals that can be imported from the description file are displayed. In the left table, all signals, where the "selection" checkboxes are selected, will be marked for import. You can either choose manually, which signals to import, you can use the "Select/Deselect all" button on the bottom left, or you can use a CSV-file to determine which signals are to be imported, by clicking "Select by CSV" on the bottom left.

In the right table the metadata, properties of the selected signals, the control unit and the protocol are displayed.



Once you have chosen all the signals you wish to import, click “OK” to complete the import procedure.



Depending on the protocol, a sampling rate or the DAQ list to use, can be assigned to the signals.

In the case of a protocol using DAQ lists (CCP, XCP), you can specify via the column selection dialog, if the signals are configured by the sampling rate or a DAQ list. To achieve this you should open the column selection dialog, via the context menu of the table header, and then drag the desired column (“sampling” or “DAQ list”) from the column selection dialog to the table header. The other column is removed automatically.

If in a protocol based on DAQ lists, sampling rates are used for the signal configuration, during import the signals are assigned to the available DAQ list with the most suitable sampling rate.

As of V18.02.00 predefined DAQ lists are supported. In case of a predefined DAQ-list the ECU provides the available signals defined by said DAQ list. It is not possible for the user to customize which signals should be requested by the DAQ list.

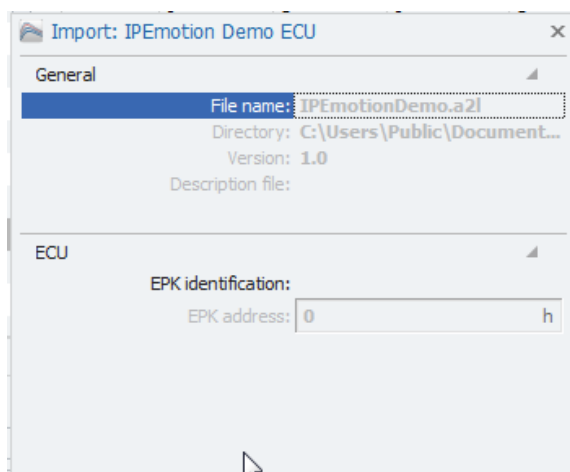
In case of protocols supporting array signals, you can specify via the “split array” column whether all the signals of the array or just the first to be imported. If this column does not appear it can be moved from the column selection dialog into the table.



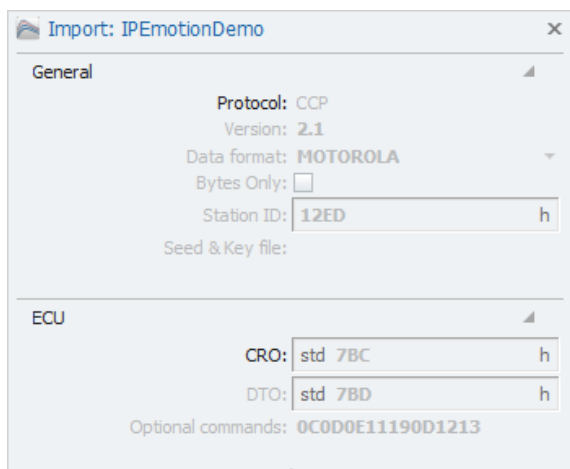
Multiple description files can be imported into the same CAN channel.

### 8.3.3 Import properties

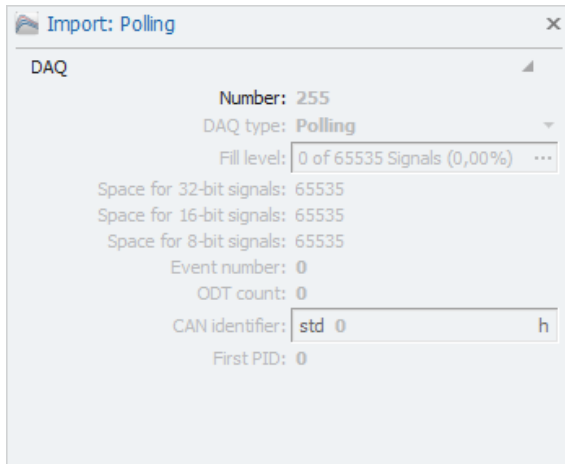
The “Import properties” of an “ECU”, “Description file”, “Polling list”, “DAQ list” or signal allow you to see certain properties such as the Data format, the Adress, the Bit mask, the start bit, bit count and more. It shows the properties as described by the description file.



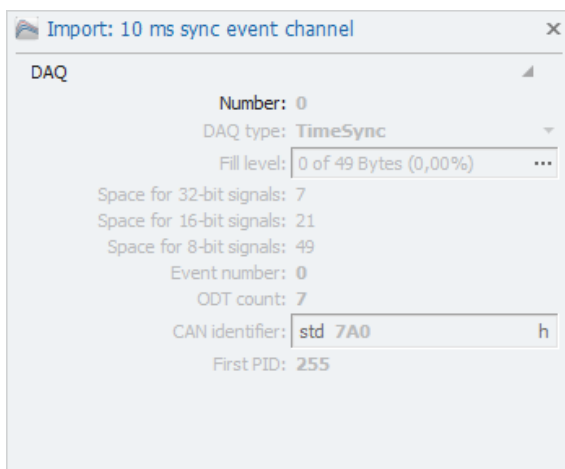
Example for Import properties of a CCP/XCP ECU



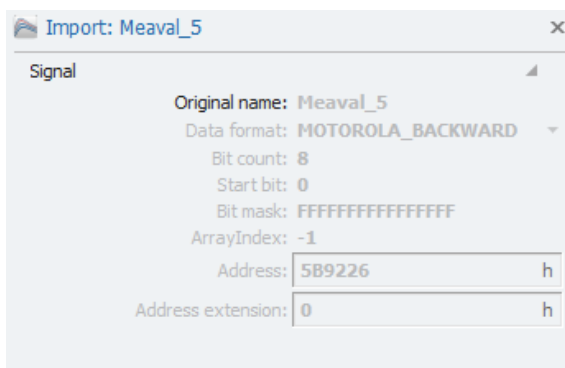
Example for Import properties of a CCP/XCP Description file



Example for Import properties of a CCP/XCP Polling list



Example for signal properties of a CCP/XCP DAQ list



Example for Import properties of a CCP/XCP signal

## 8.3.4 Signal properties

### 8.3.4.1 Tree elements for CCP/XCP signals

After having successfully imported the desired signals to your CAN channel, this channel will contain three new layers of child elements in the measurement task tree: The “ECU”-layer, the “Description file”-layer and the “Message”-layer.



## ECU

The “ECU” (Electronic control unit) represents the control unit inside the vehicle that the logger communicates with. It allows for an active communication between logger and vehicle, where the logger can not only receive messages but also send messages.

Name	
ARCOS 1.5 - dataLog	15
CAN Interfaces	15
CAN 01	15
IPEmotion Demo ECU	15
IPEmotionDemo	15
Status	0
Polling	0
10 ms sync event channel	0
100 ms sync event channel	15
seg sync event channel	0
CAN 02	0
CAN 03	0
CAN 04	0

### Description file (Station)

The “description file” (also called station in case of CCP/XCP) is a database file which contains signal information and can be used to import those signals into a Signal channel in IPEmotion. The filetype associated with CCP/XCP signals is the “A2L file”.

The symbol in the left part of the tree element shows you the type of “database” you imported (CCP or XCP), then follows the name of the imported “description file” (in this case “IPEmotionDemo”) and on the right is a number indicating how many signals the “description file” contains (in this case 15).

Name	
ARCOS 1.5 - dataLog	15
CAN Interfaces	15
CAN 01	15
IPEmotion Demo ECU	15
IPEmotionDemo	15
Status	0
Polling	0
10 ms sync event channel	0
100 ms sync event channel	15
seg sync event channel	0
CAN 02	0
CAN 03	0
CAN 04	0

## Signal lists

Each “description file” can contain one or more Signal lists, which then contain the actual signals. A Signal list can be found in the “Measurement task tree” as a child element of the “description file”, it belongs to.

Each Signal list can, again, contain one or more signals, which is indicated by the number on the right of the Signal list’s name.

CCP/XCP Signal lists group the “description file’s” signals in groups with different functionalities. The groups differ in the way that the logger obtains or requests a signal.

Name	
ARCOS 1.5 - dataLog	15
CAN Interfaces	15
CAN 01	15
IPEmotion Demo ECU	15
IPEmotionDemo	15
Status	0
Polling	0
10 ms sync event channel	0
100 ms sync event channel	15
seg sync event channel	0
CAN 02	0
CAN 03	0
CAN 04	0

- **Status list**

The status list contains signals on the ECU status and the status of DAQ lists included in the description file. Each signal list has two entries. “**Configured**” gives information whether the list has been configured, “**started**” gives information whether the respective list has been started and is running or not.

“**Station connected**” tells you whether the respective ECU is connected and “**EPK Versioncheck result**” lets you know whether the “EPK check” has been passed successfully or not.

### Station connected

Tells you the status of the connection with the respective ECU.

NaN = not started	The ECU has not been started.
0 = failed	The ECU has been started but the connection test has failed.
1 = successful	The ECU has been started and the connection test has been passed successfully.

### EPK Versioncheck result

Tells you about the result of the “EPK check”.

NaN = not started	The “EPK check” has not been performed.
0 = failed	The “EPK check” has been performed but not passed.
1 = successful	The “EPK check” has been performed and passed successfully.

**xxx\_ms\_task\_configured**

Tells you whether the respective DAQ list has been configured.

0 = not yet configured	The DAQ list has not yet been configured.
1 = successful	The DAQ list has been configured.

**xxx\_ms\_task\_started**

Tells you whether the respective DAQ list has been started.

0 = not yet started	The DAQ list has not been started. It is inactive.
1 = successful	The DAQ list has been started.

- **Polling list**

Signals contained in a "Polling list", will be actively requested by the logger. That means, for each signal a sampling rate has to be defined, according to which the logger will request the signals which will then be sent to the logger by the ECU.

- **DAQ list**

Signals contained in a "DAQ list" are assigned with a certain time interval in which the ECU is requested to send these signals. I.e. any signal contained in the DAQ list "**100ms sync event channel**" will be sent to the logger in an interval of 100ms. This guarantees that there is no unintended delay.

### 8.3.4.2 Grid area for CCP/XCP signals

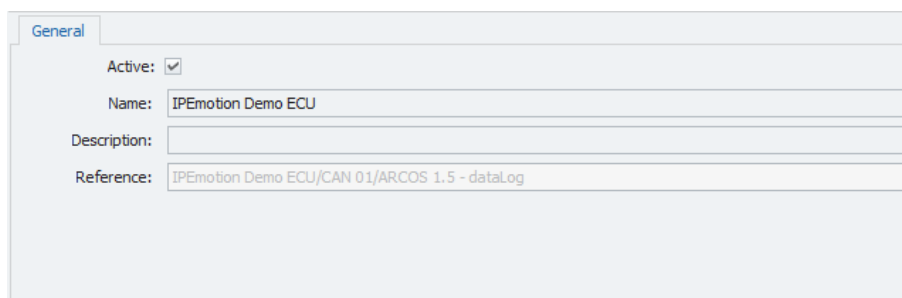
In the "grid area" you will be presented with an overview of your selected CAN channel's signals. Also you can find here two important functions, which are the "**Column chooser**" (→4.3.1) and the "**Filter editor**" (→4.3.2).

### 8.3.4.3 Details area for CCP/XCP signals

The Details area shows settings either for the selected tree element ("ECU", "description file" or "signal list") or the selected signal in the grid area.

- **"ECU" selected**

In this case the details area will only show the "General" tab. Please refer to (→4.2.2).

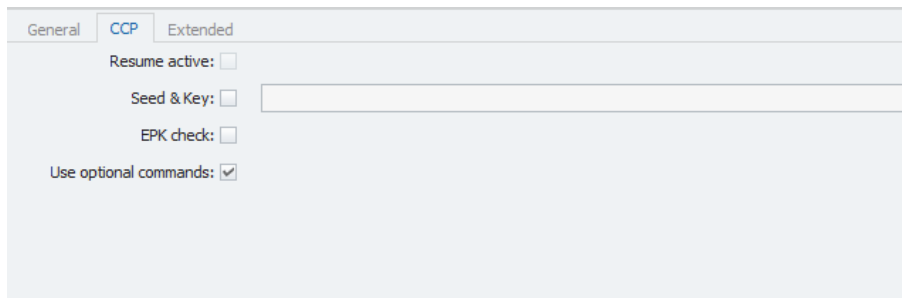


- “description file/station” selected

In this case the details area will contain the “General” tab (→4.2.2) plus additional tabs.

## CCP/XCP

This tab contains CCP and XCP specific options.



- **Resume active**

Activates the start of sending data by the ECU without the need of an initialization. Thereto the configuration is transmitted into the flash of ECU once and stays there, so that the configuration is still available after power loss.

- **Seed & Key**

This field allows you to enter a Seed & Key binary file (\*.skb) which contains the information required to unlock the ECU if necessary.

- **EPK check**

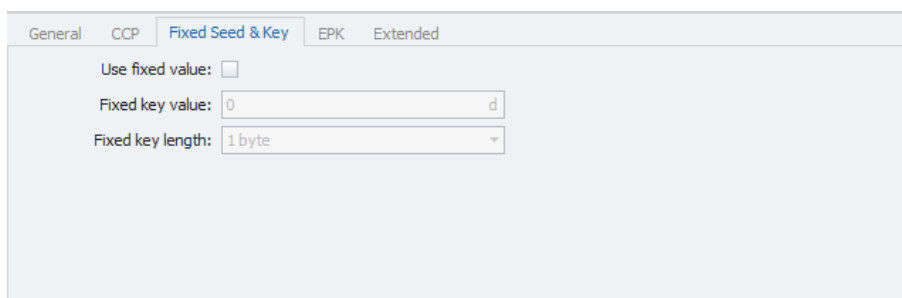
Checks for differences in the checksums of the current configuration and the ECU.

- **Use optional commands (CCP only)**

If marked active, the optional commands defined in the ECU file will be used.

## Fixed Seed & Key

If no Seed & Key file has been specified, this tab allows you to define fixed Seed & Key settings to be used.



- **Use fixed value**

If marked active, a fixed Seed & Key value will be used. The value has to be

defined below.

- **Fixed key value**

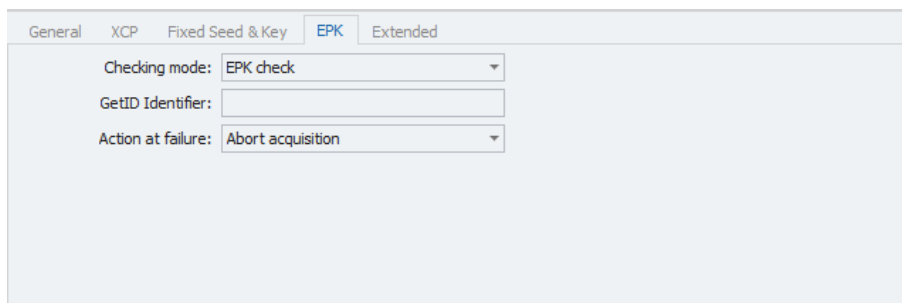
Define the fixed Seed & Key value. The button on the right allows to switch between decimal, hexadecimal or binary mode.

- **Fixed key length**

The dropdown menu allows you to define the byte length of the fixed Seed & Key value.

## EPK

This tab allows for EPK settings.



- **Checking mode**

The dropdown menu allows you to choose the checking mode. It can be either "EPK" or "GetID". "GetID" is only available for XCP.

- **GetID Identifier**

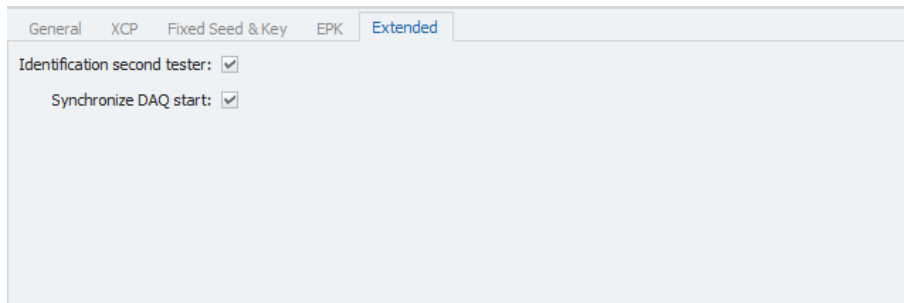
Define the identifier for the GetID mode. The identifier is a string that corresponds to the ECU's software version number. It can be found under "EPK identification" by right-clicking on the ECU in the tree and then clicking "Import properties".

- **Action at failure**

Allows you to define what action to take upon an EPK check failure. Regardless of the selected option the logger will always write an EPK check failure logging message.

## Extended

This tab allows for extended settings.



### – Identification second tester

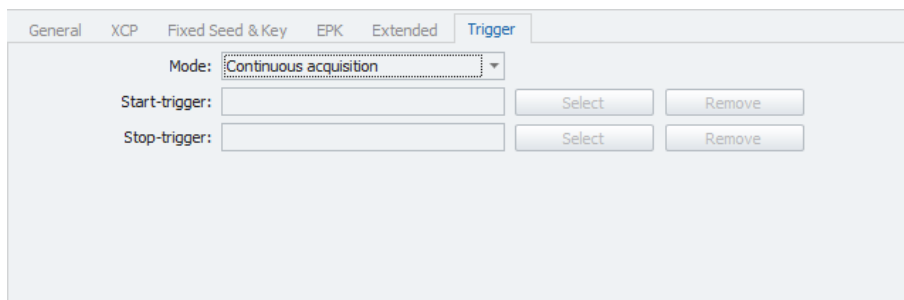
Allows you to activate or deactivate the detection of a second master at the bus.

### – Synchronize DAQ start

Allows you to activate or deactivate the synchronization of the start of all DAQ-lists.

## Trigger

This tab allows to define a trigger in order to stop or start the entire station. Stopping the station means disconnecting from it. No data will be received from the station. In order to stop/start single signals or signal groups, the trigger function of the daq-lists has to be used.



### – Mode

Define whether you wish to continuously acquire data or if you want to start/stop data acquisition via a trigger. There are two modes to control data acquisition via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will acquire data as long as the start trigger condition is met. Once it is no longer met, data acquisition will stop.

#### – Start-trigger

Allows you to choose a trigger upon whose activation the station will be CONNECTED. A trigger has first to be defined. Please refer to (→9).

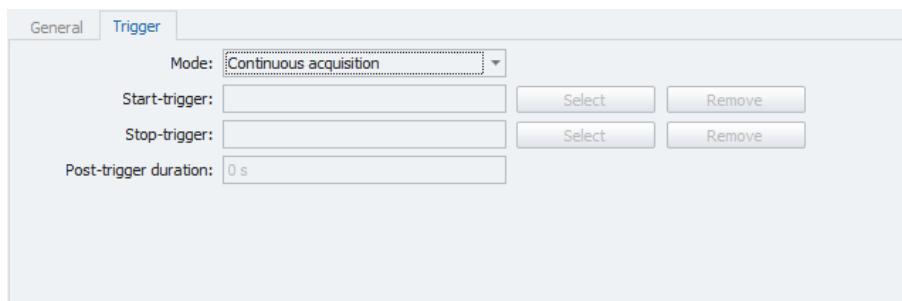
#### – Stop-trigger

Allows you to choose a trigger upon whose activation the station will be DISCONNECTED. A trigger has first to be defined. Please refer to (→9).

If there is no “Stop-trigger” defined the acquisition will be stopped by inverted start condition.

- “Polling list” or “DAQ list” selected

In this case the details area will additionally contain the “Trigger tab”. The “Trigger tab” allows you to set a trigger upon whose activation the signals contained in the list will be requested.



#### – Mode

Define whether you wish to continuously acquire data or if you want to start/stop data acquisition via a trigger. There are two modes to control data acquisition via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will acquire data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data acquisition will stop.

#### – Start-trigger

Allows you to choose a trigger upon whose activation the list’s Signals will be requested. A trigger has first to be defined. Please refer to (→9).

#### – Stop-trigger

Allows you to choose a trigger upon whose activation the list’s Signals will stop being requested. A trigger has first to be defined. Please refer to (→9).

If there is no “Stop-trigger” defined the acquisition will be stopped by inverted start condition.

### – Post-trigger duration

You can determine here for how long after a stop request (either by explicit “Stop-trigger” or inverted start condition) the stop will be delayed and data acquisition will continue.

### • Signal selected

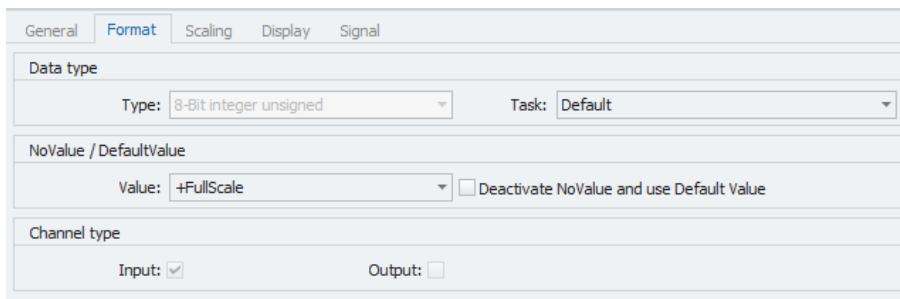
In this case the details area will contain additional tabs which will be explained in the following.

### General

Please refer to (→[4.2.2](#)).

### Format

This tab contains information and options regarding file format, tasks and Channel type.



### – Data type

This field tells you the type of data (in this case “**8-Bit integer unsigned**”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

### – NoValue / DefaultValue

This field allows you to define the value that will be shown if a signal value is read as invalid.

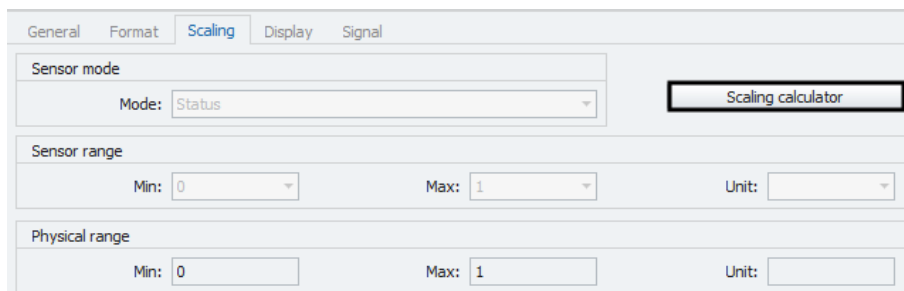
### – Channel type

This field tells you whether you are dealing with a “**Input**” channel or “**Output**” channel.



## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “Scaling calculator” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.



### – Sensor Mode

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

### – Sensor Range

Shows the raw value range of the signal.

### – Physical Range

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



### – Displaying area

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

### – Formatting

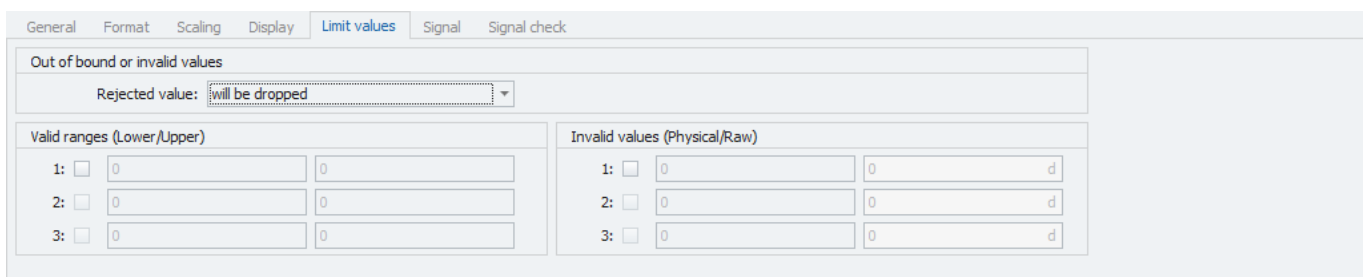
The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

### – Name

Allows you to set a Name to be shown on a display.

## Limit values

This tab allows you to define limit values for a signal and what action to take upon a limit value violation.



The screenshot shows the 'Limit values' configuration window. At the top, there are tabs for 'General', 'Format', 'Scaling', 'Display', 'Limit values', 'Signal', and 'Signal check'. The 'Limit values' tab is active. Below the tabs, there is a section titled 'Out of bound or invalid values' with a dropdown menu showing 'will be dropped'. Underneath, there are two main sections: 'Valid ranges (Lower/Upper)' and 'Invalid values (Physical/Raw)'. Each section contains three rows, each with a checkbox and two input fields. In the 'Valid ranges' section, all checkboxes are unchecked and the input fields contain '0'. In the 'Invalid values' section, all checkboxes are unchecked and the input fields contain '0' and 'd'.

### – Rejected value

Define what happens to a value, that has been rejected because it is out of bound or invalid. By default this value will be dropped. It can also be written as NaN.

### – Valid ranges (Lower/Upper)

Define up to three ranges of valid signal values. Activate a range in order to define its upper/lower value (datatype double). Range 2 can only be activated if range 1 is and range 3 can only be activated if range 2 is.

### – Invalid values (Physical/Raw)

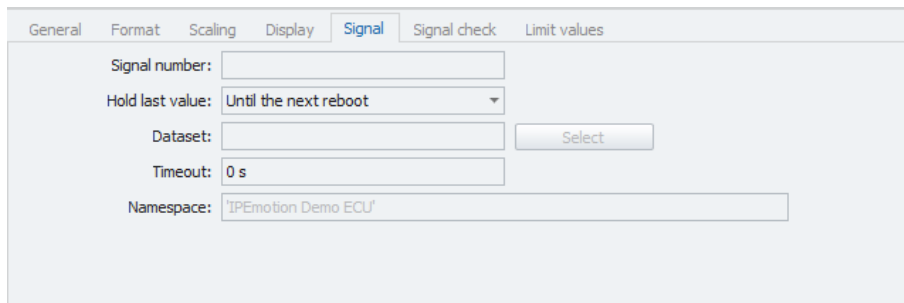
Define up to three invalid values. Activate an invalid value in order to define the physical value (datatype double) or raw value (datatype integer). If one of the two has been typed in, the other will be calculated according to the scale/offset settings in the scaling calculator.

Invalid value 2 can only be activated if invalid value 1 is and invalid value 3 can only be activated if invalid value 2 is.

For details on how to use the “Scaling calculator” please refer to the **IPeMotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

## Signal

This tab allows you to define signal settings.



### – Signal number

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.

### – Hold last value

Specify, for how long the last value of the signal will be hold.

### – Dataset

If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.

### – Timeout

Specify the timeout period for the current signal. If the data source doesn’t send data for the specified time period, the value of the signal is set to “NaN (Not a Number)” and will be displayed as “–” in a display.

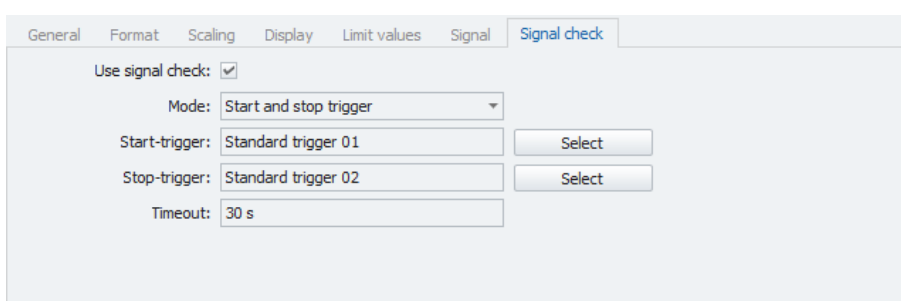
### – Namespace

The “Namespace” serves as unique identifier for the signal inside the logger.

## Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



## 8.4 UDS signals

### 8.4.1 Storage method

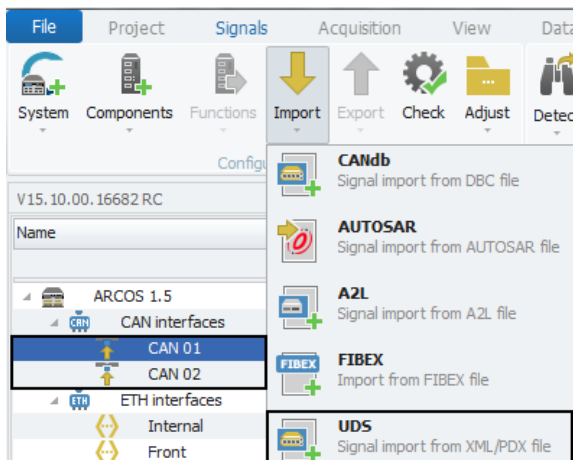
In order to store incoming signals on a CAN channel use one of the following signal storage methods.

- ATEX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

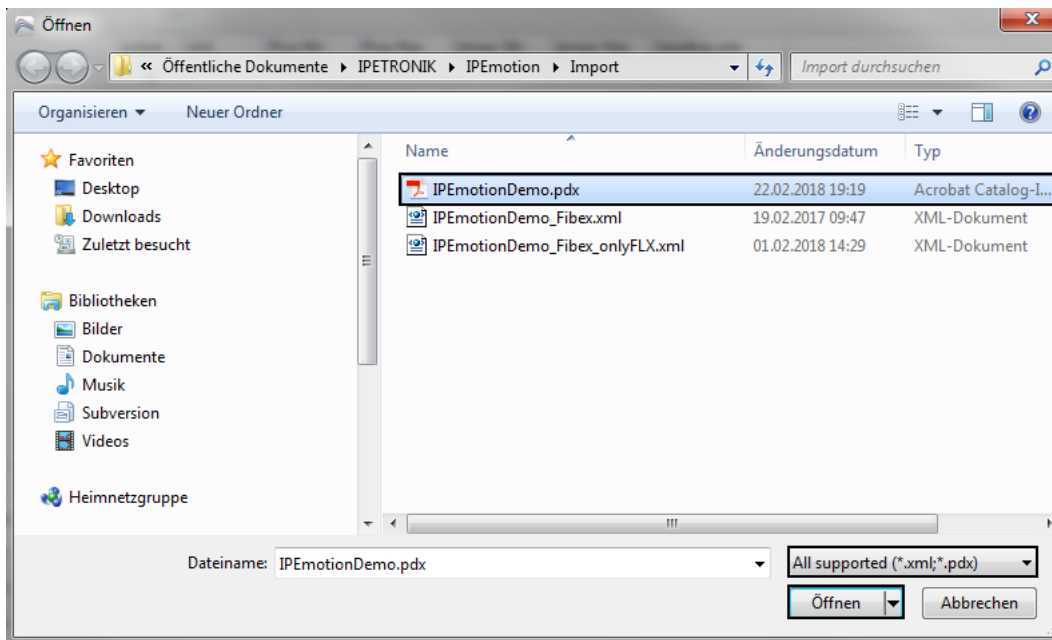
### 8.4.2 Importing UDS signals

This section explains how to import **UDS signals**. The filetype associated with these signals is a “XML/PDX file”.

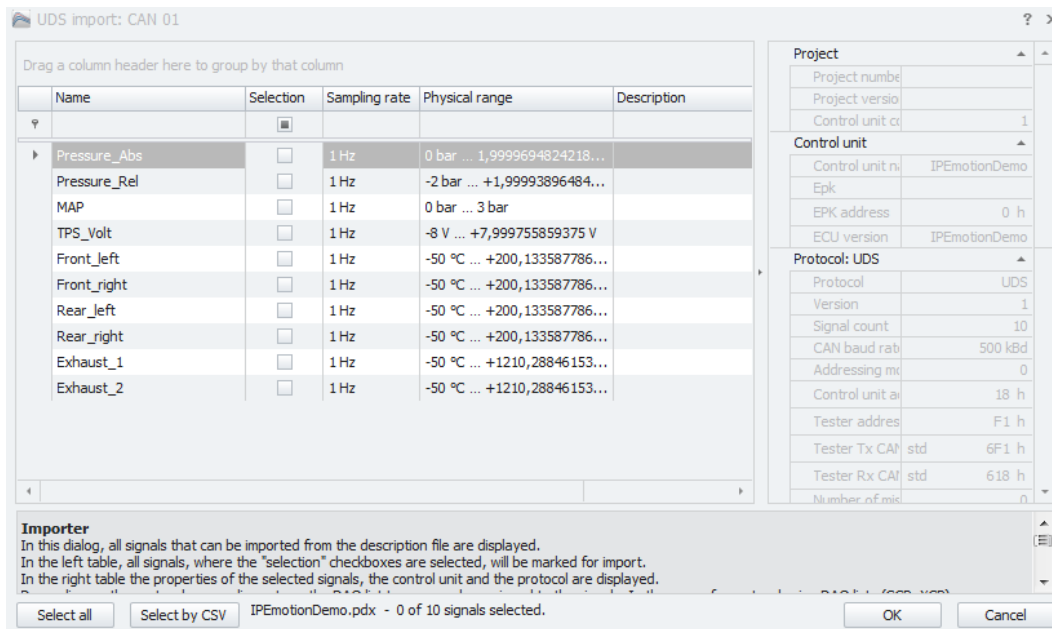
To import a UDS signal, select the CAN channel to which you wish to import your signal in the tree, click the “Import” button in the ribbon and then choose the “UDS” filetype for the import.



The following window lets you choose which file you wish to import. According to the file-type you have chosen earlier for your import, you will now only be able to choose files of the “XML/PDX” filetype. Choose the file you wish to import and click “Open”.



Next the “Importer” window will appear, that will present you with a range of importing options.



In this dialog, all signals that can be imported from the description file are displayed. In the left table, all signals, where the “selection” checkboxes are selected, will be marked for import. You can either choose manually, which signals to import, you can use the “**Select/Deselect all**” button on the bottom left, or you can use a CSV-file to determine which signals are to be imported, by clicking “**Select by CSV**” on the bottom left.

In the right table the metadata, properties of the selected signals, the control unit and the protocol are displayed.

Changes and errors excepted.

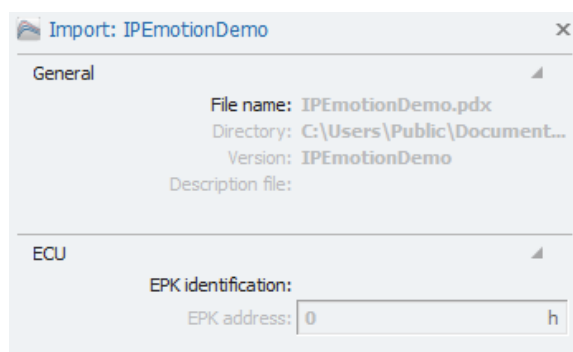
Once you have chosen all the signals you wish to import, click “OK” to complete the import procedure.



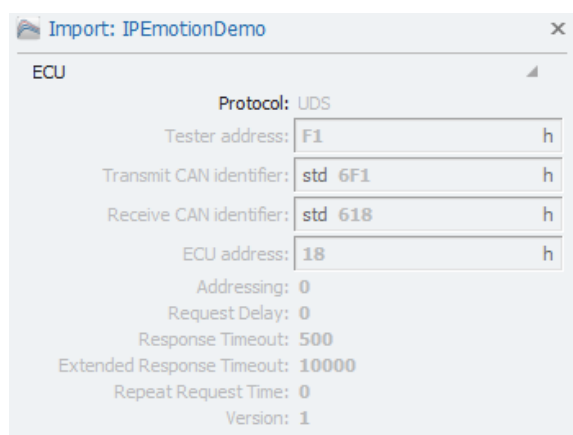
Multiple description files can be imported into the same CAN channel.

### 8.4.3 Import properties

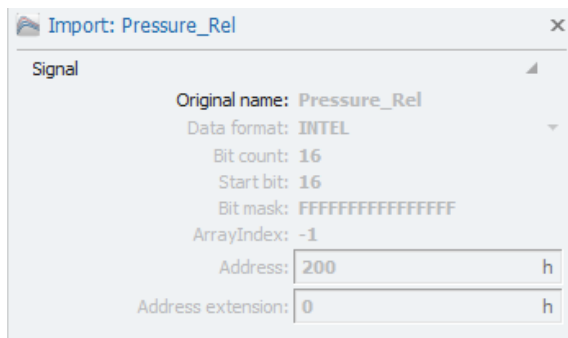
The “Import properties” of an “ECU”, “Description file” or signal allow you to see certain properties such as the Data format, the Address, the Bit mask, the start bit, bit count and more. It shows the properties as described by the description file.



Example for Import properties of a UDS ECU



Example for Import properties of a UDS Description file



Example for Import properties of a UDS signal

### 8.4.4 Signal properties

#### 8.4.4.1 Tree elements for UDS signals

After having successfully imported the desired signals to your CAN channel, this channel will contain two new layers of child elements in the measurement task tree: The "ECU"-layer and the "Description file"-layer.

#### ECU

The "ECU" (Electronic control unit) represents the control unit inside the vehicle that the logger communicates with. It allows for an active communication between logger and vehicle, where the logger can not only receive messages but also send messages.

ARCOS 1.5	10
CAN interfaces	10
CAN 01	10
IPEmotionDemo	10
IPEmotionDemo	10

#### Description file

The "description file" is a database file which contains signal information and can be used to import those signals into a Signal channel in IPEmotion. The filetype associated with UDS signals is the "UDS file".

The symbol in the left part of the tree element shows you the name of the imported "description file" (in this case "IPEmotionDemo") and on the right is a number indicating how many signals the "description file" contains (in this case 10).

ARCOS 1.5	10
CAN interfaces	10
CAN 01	10
IPEmotionDemo	10
IPEmotionDemo	10

### 8.4.4.2 Grid area for UDS signals

In the “grid area” you will be presented with an overview of your selected CAN channel’s signals. Also you can find here two important functions, which are the “**Column chooser**” (→4.3.1) and the “**Filter editor**” (→4.3.2).

### 8.4.4.3 Details area for UDS signals

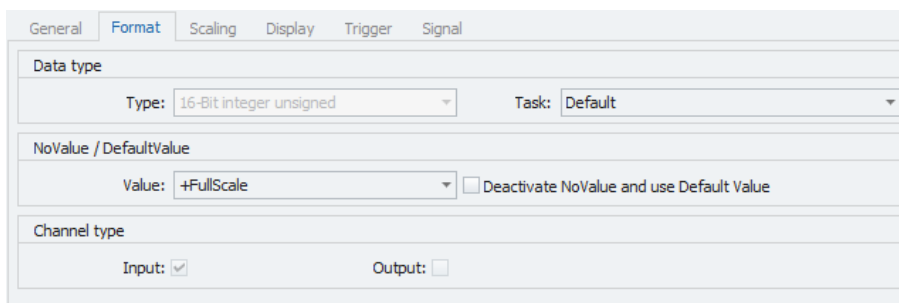
The Details area allows you to access settings for the selected signal in the grid area.

#### General

Please refer to (→4.2.2).

#### Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case “**16-Bit integer unsigned**”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a signal value is read as invalid.

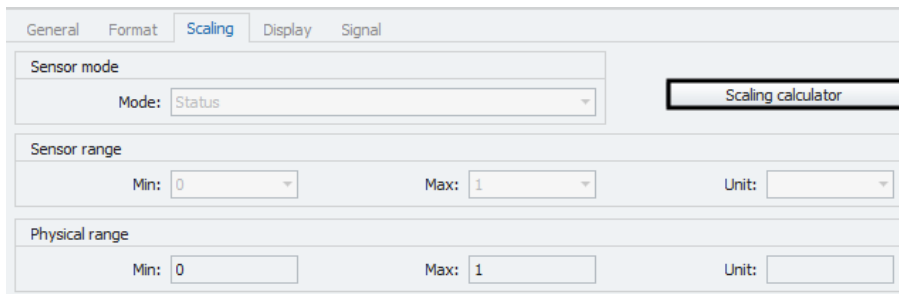
- **Channel type**

This field tells you whether you are dealing with a “**Input**” channel or “**Output**” channel.



## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

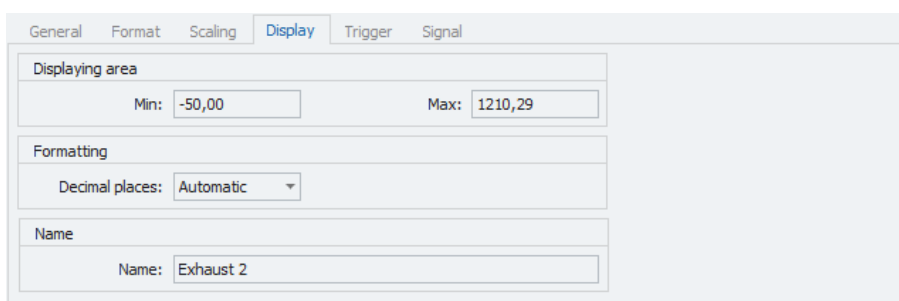
Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

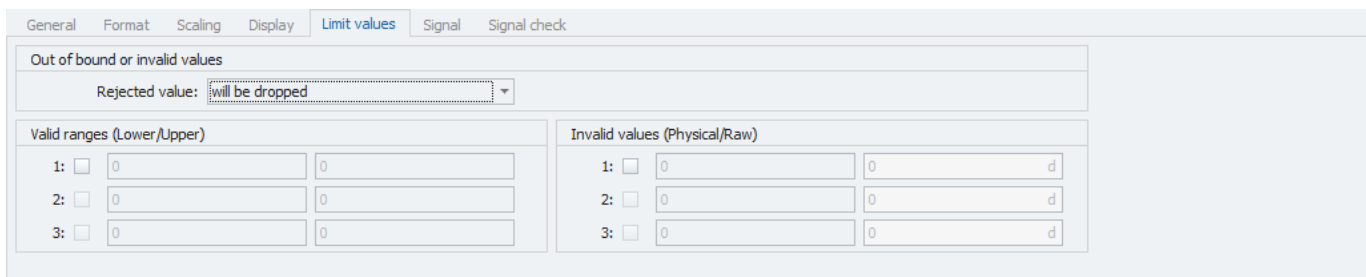
The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

### Limit values

This tab allows you to define limit values for a signal and what action to take upon a limit value violation.



- **Rejected value**

Define what happens to a value, that has been rejected because it is out of bound or invalid. By default this value will be dropped. It can also be written as NaN.

- **Valid ranges (Lower/Upper)**

Define up to three ranges of valid signal values. Activate a range in order to define its upper/lower value (datatype double). Range 2 can only be activated if range 1 is and range 3 can only be activated if range 2 is.

- **Invalid values (Physical/Raw)**

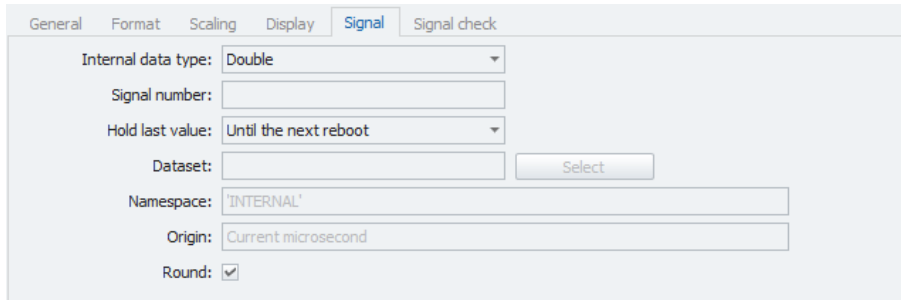
Define up to three invalid values. Activate an invalid value in order to define the physical value (datatype double) or raw value (datatype integer). If one of the two has been typed in, the other will be calculated according to the scale/offset settings in the scaling calculator.

Invalid value 2 can only be activated if invalid value 1 is and invalid value 3 can only be activated if invalid value 2 is.

For details on how to use the “Scaling calculator” please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

## Signal

This tab allows you to define signal settings.

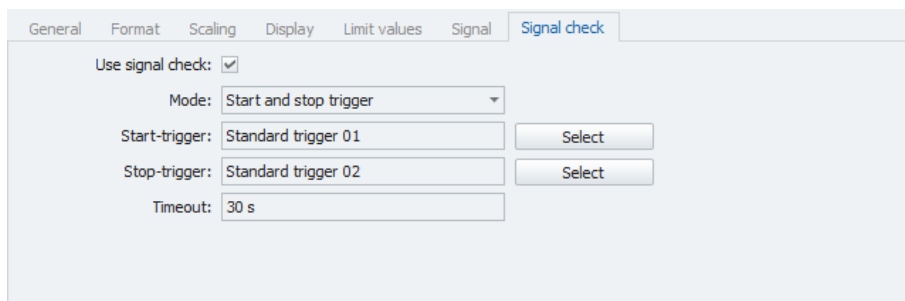


- Internal data type**  
 Assign an internal data type to the signal. Available data types are “**Double**” and “**String**”.
- Signal number**  
 Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.
- Hold last value**  
 Specify, for how long the last value of the signal will be hold.
- Dataset**  
 If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.
- Timeout**  
 Specify the timeout period for the current signal. If the data source doesn’t send data for the specified time period, the value of the signal is set to “NaN (Not a Number)” and will be displayed as “-” in a display.
- Namespace**  
 The “Namespace” serves as unique identifier for the signal inside the logger.

## Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



The screenshot shows a software interface with a tabbed menu at the top: General, Format, Scaling, Display, Limit values, Signal, and Signal check. The 'Signal check' tab is active. Below the tabs, there is a section for configuring signal check parameters:

- Use signal check:**
- Mode:** Start and stop trigger (dropdown menu)
- Start-trigger:** Standard trigger 01 (text input) with a **Select** button
- Stop-trigger:** Standard trigger 02 (text input) with a **Select** button
- Timeout:** 30 s (text input)

## 8.5 OBD signals

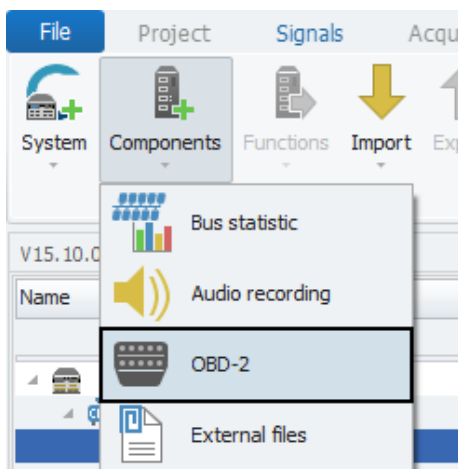
### 8.5.1 Storage method

In order to store all incoming traffic on a CAN channel use a bus tracing method for storage. Please refer to (→ 14.8).

### 8.5.2 Adding the OBD signals interface

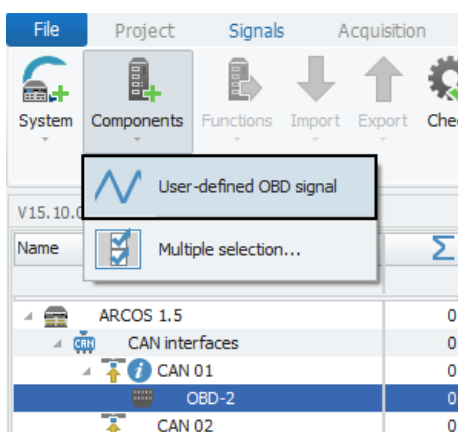
This section explains how to work with OBD (On-Board diagnosis) signals.

In order to work with "OBD signals", you will first need to add the "OBD signals" interface, which will contain all the available "OBD signals". To do so, select a CAN channel in the tree, click the "Components" button in the Ribbon and then choose "OBD-2".



### 8.5.3 User-defined OBD signals

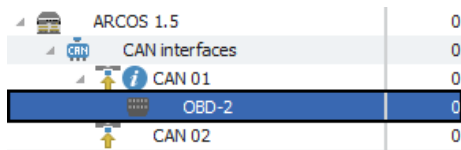
It is possible to define a user-specific "OBD signal". To do so, select the "OBD signals" interface in the measurement task tree, click the "Components" button in the Ribbon and then choose "User-defined OBD signal".



### 8.5.4 Signal properties

#### 8.5.4.1 Tree elements for OBD signals

After having added the “OBD signals” interface, it will appear in the measurement task tree as a new tree element.



#### 8.5.4.2 Grid area for OBD signals

In the “grid area” you will be presented with an overview of all the available “OBD signals”. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

	Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
?		<input type="checkbox"/>						
0.	09 01 - Length of vehicle identification	<input checked="" type="checkbox"/>		0	255	0	255	Event controlled
	09 02 - Vehicle identification number	<input type="checkbox"/>						Event controlled
	002 - DTC that caused required fr...	<input type="checkbox"/>		0	65535	0	65535	1 Hz
	003 - Fuel system status 1 & 2	<input type="checkbox"/>		0	65535	0	65535	1 Hz
	004 - Calculated load value	<input type="checkbox"/>	%	0,0	100,0	0	100	1 Hz
	005 - Engine coolant temperature	<input type="checkbox"/>	°C	-40	215	-40	215	1 Hz
	006 - Short term fuel % trim Bank 1	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	006 - Short term fuel % trim Bank 3	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	007 - Long term fuel % trim Bank 1	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	007 - Long term fuel % trim Bank 3	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	008 - Short term fuel % trim Bank 2	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	008 - Short term fuel % trim Bank 4	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	009 - Long term fuel % trim Bank 2	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	009 - Long term fuel % trim Bank 4	<input type="checkbox"/>	%	-100,0	99,2	-100	99,22	1 Hz
	00A - Fuel pressure	<input type="checkbox"/>	bar	0,00	7,65	0	7,65	1 Hz

### 8.5.4.3 Details area for OBD signals

The Details area allows you to access settings either for the “OBD signals” interface or for the selected “OBD signal” in the grid area.

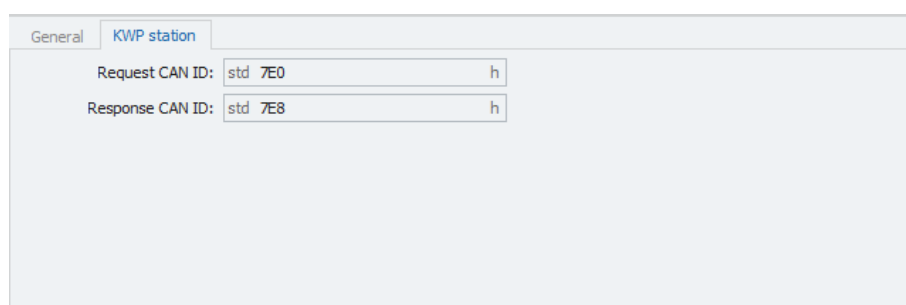
If the interface has been selected in the measurement task tree, the details area will contain two tabs.

#### General

Please refer to (→[4.2.2](#)).

#### KWP station

Define the request and respond CAN ID of the addressed ECU.



The screenshot shows a software interface with two tabs: 'General' and 'KWP station'. The 'KWP station' tab is active. It contains two input fields: 'Request CAN ID' with the value 'std 7E0' and a unit selector 'h', and 'Response CAN ID' with the value 'std 7E8' and a unit selector 'h'.

- **Request CAN ID**

The Request CAN ID is typically between 7E0h and 7E7h.

- **Respond CAN ID**

The Respond CAN ID is typically between 7E8h and 7EFh.

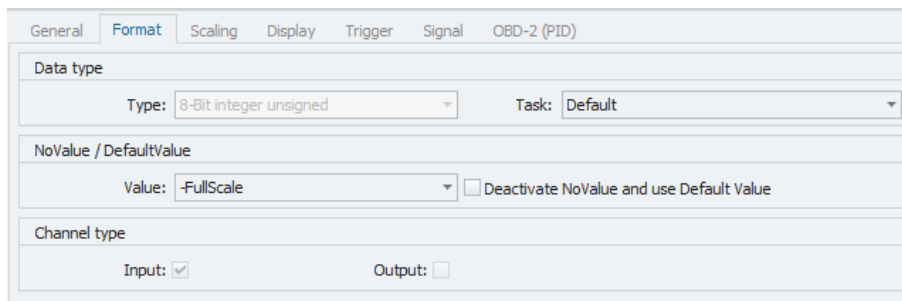
If a signal has been selected in the Grid area, the details area will contain the following tabs.

## General

Please refer to (→[4.2.2](#)).

## Format

This tab contains information and options regarding file format, tasks and Channel type.



General	Format	Scaling	Display	Trigger	Signal	OBD-2 (PID)
Data type						
Type:	8-Bit integer unsigned	Task:	Default			
NoValue / DefaultValue						
Value:	-FullScale	<input type="checkbox"/> Deactivate NoValue and use Default Value				
Channel type						
Input:	<input checked="" type="checkbox"/>	Output:	<input type="checkbox"/>			

- **Data type**

This field tells you the type of data (in this case “**8-bit integer unsigned**”) and allows you to apply special tasks for this formula/signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a formula/signal value is read as invalid.



## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

Changes and errors excepted.

- **Name**

Allows you to set a Name to be shown on a display.

## Trigger

This tab provides settings regarding the trigger for the start and stop of data acquisition on this channel.



- **Mode**

Define whether you wish to continuously acquire data or if you want to start/stop data acquisition via a trigger. There are two modes to control data acquisition via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will acquire data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data acquisition will stop.

- **Start-trigger**

Define a trigger, that will start data acquisition.

- **Stop-trigger**

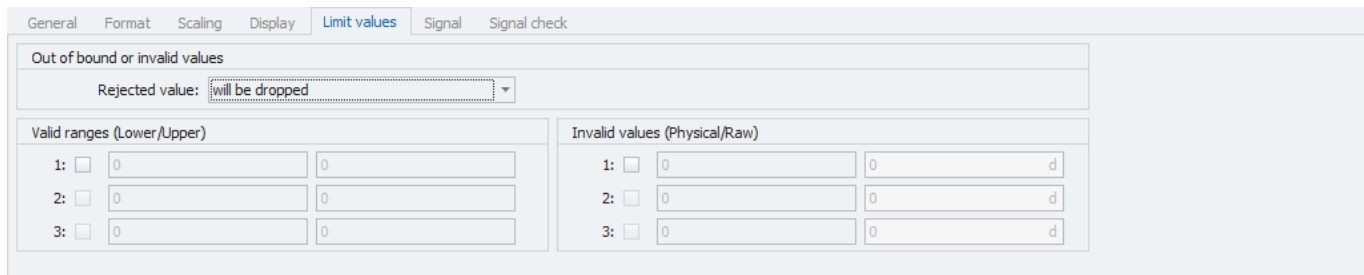
Define a trigger, that will stop data acquisition.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, data acquisition will stop.

## Limit values

This tab allows you to define limit values for a signal and what action to take upon a limit value violation.



- **Rejected value**

Define what happens to a value, that has been rejected because it is out of bound or invalid. By default this value will be dropped. It can also be written as NaN.

- **Valid ranges (Lower/Upper)**

Define up to three ranges of valid signal values. Activate a range in order to define its upper/lower value (datatype double). Range 2 can only be activated if range 1 is and range 3 can only be activated if range 2 is.

- **Invalid values (Physical/Raw)**

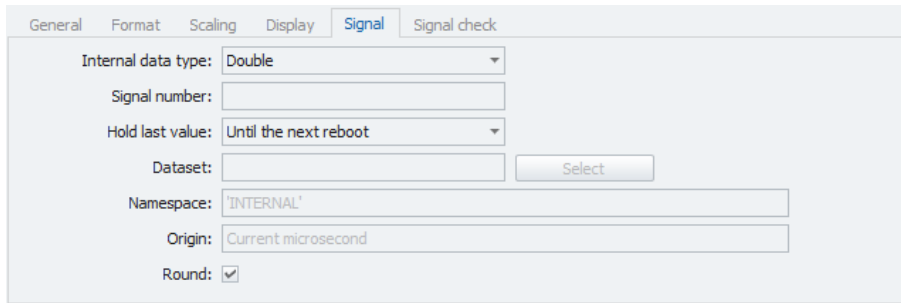
Define up to three invalid values. Activate an invalid value in order to define the physical value (datatype double) or raw value (datatype integer). If one of the two has been typed in, the other will be calculated according to the scale/offset settings in the scaling calculator.

Invalid value 2 can only be activated if invalid value 1 is and invalid value 3 can only be activated if invalid value 2 is.

For details on how to use the "Scaling calculator" please refer to the **IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling"**.

## Signal

This tab allows you to define signal settings.



The screenshot shows the 'Signal' tab in a software interface. The tab is active and shows various configuration options. The options are:

- Internal data type: Double (dropdown menu)
- Signal number: (text input field)
- Hold last value: Until the next reboot (dropdown menu)
- Dataset: (text input field) with a 'Select' button next to it
- Namespace: 'INTERNAL' (text input field)
- Origin: Current microsecond (text input field)
- Round:  (checkbox)

- **Internal data type**

Assign an internal data type to the signal. Available data types are "Double" and "String".

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting "Hold last value" has been set to "Until end of dataset", you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to "NaN (Not a Number)" and will be displayed as "-" in a display.

- **Namespace**

The "Namespace" serves as unique identifier for the signal inside the logger.

## OBD-2(PID)

This tab allows shows extended OBD-2 specific settings.



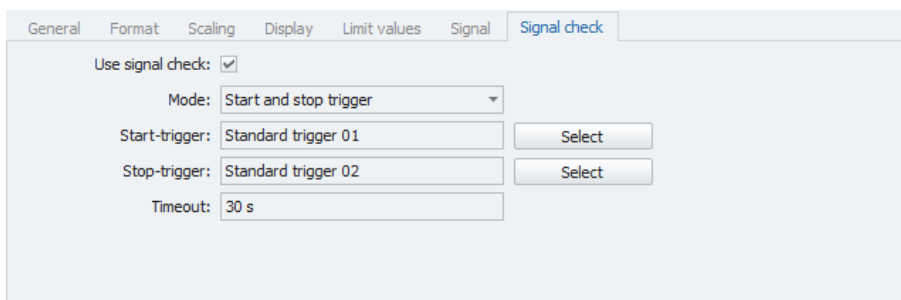
The screenshot shows the 'OBD-2 (PID)' tab with the following settings:

Parameter	Value	Unit
SID:	1	h
PID:	6	h
PID size:	2	
Start bit:	8	
Bit count:	8	

## Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



The screenshot shows the 'Signal check' tab with the following settings:

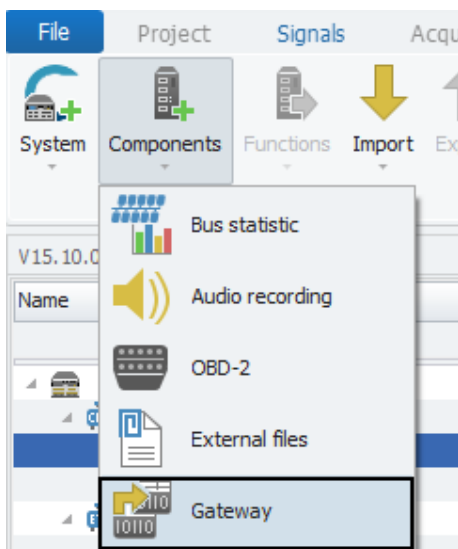
Use signal check:	<input checked="" type="checkbox"/>	
Mode:	Start and stop trigger	
Start-trigger:	Standard trigger 01	Select
Stop-trigger:	Standard trigger 02	Select
Timeout:	30 s	

## 8.6 Gateways

The Gateway method makes it possible to take messages received on one CAN channel (source) and output them on another channel (target). Messages are output as soon as they are received with this method, it is neither possible to influence the time of transmission nor to modify the sent data (this can be achieved by means of programming, through a script). Filters can be used to restrict the messages transmitted.

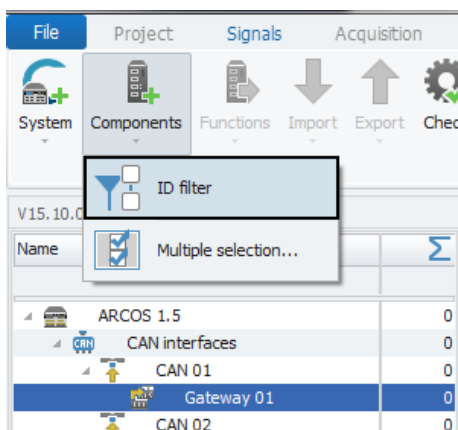
### 8.6.1 Adding a gateway

In order to add a gateway, select the desired source CAN channel in the measurement task tree, click the components button in the Ribbon and then select "Gateway". For instructions regarding the settings of a gateway please refer to (→[8.6.3](#)).



### 8.6.2 Adding an ID filter

In order to limit the traffic passed through the gateway, one or more ID filter per gateway can be defined. In order to add an ID filter select the desired gateway in the measurement task tree, click the "Components" button in the Ribbon and then select "ID filter". For instruction regarding the settings of an ID filter please refer to (→[8.6.4](#)).



### 8.6.3 Gateway settings

#### 8.6.3.1 Tree elements for Gateways

Each Gateway that has been added to a CAN channel will appear in the measurement task tree as a child element to the respective CAN channel.

Name	
ARCOS 1.5	0
CAN interfaces	0
CAN 01	0
Gateway 01	0
Gateway 02	0
CAN 02	0

#### 8.6.3.2 Grid area for Gateways

In the “grid area” you will be presented with an overview of all the previously added ID filters for the selected gateway. You can activate or deactivate an ID filter by ticking/unticking the “Active”-box. Also you can find here two important functions, which are the “**Column chooser**” (→4.3.1) and the “**Filter editor**” (→4.3.2).

	Name	Active	Description
▼		<input type="checkbox"/>	
▶	ID filter 01	<input checked="" type="checkbox"/>	
	ID filter 02	<input checked="" type="checkbox"/>	

#### 8.6.3.3 Details area for Gateways

The details area provides settings regarding the general behaviour of a gateway.

##### General

Please refer to (→4.2.2).

##### Settings

Settings regarding the target CAN channel and filter action.

General Settings

Default filter action: Pass all except specified ID or ID range

Target:  Select

- **Default filter action**

Define whether the gateway should pass or block all traffic except the specified ID or ID range.

Changes and errors excepted.

- **Target**

Select the desired target CAN channel.

#### 8.6.4 ID filter settings

All setting regarding the ID filter can be found in the “Filter settings” tab of the details area of the respective ID filter.

Name	Active	Description
☿	<input type="checkbox"/>	
▶ ID filter 01	<input checked="" type="checkbox"/>	
ID filter 02	<input checked="" type="checkbox"/>	

General Filter settings

Mode: Specific ID ▼

CAN ID: std 0 h

Stop ID: std 0 h

- **Mode**

Define whether the filter will affect a specific ID or an ID range.

- **CAN ID**

Define the specific or start ID for the filter. For detailed instructions on the topic CAN ID please refer to (→[8.1.3.4](#)).

- **CAN ID**

Define the stop ID for the filter if the mode ID range has been selected.



## 8.7 Runstate

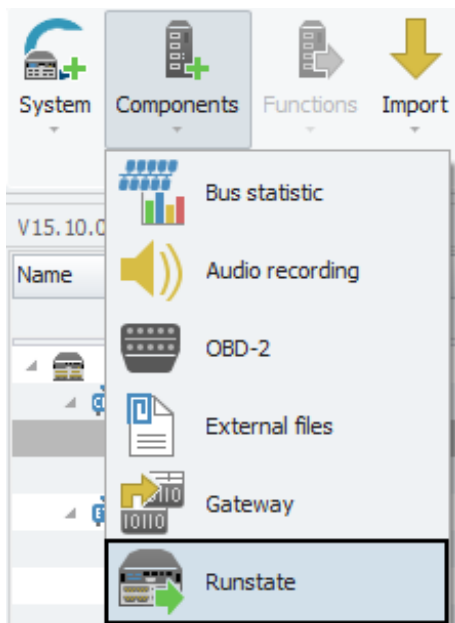
Taking into account the variety of different run states, plus the fact that the ARCOS display is not always used, we have created the possibility of using CAN messages to read out the current logger state and export it into a CANdb file.

For this purpose, the start identifier can be freely selected, while the remaining messages are sequentially put out on the subsequent identifiers.

Moreover, the output rate can be selected, and the content influenced by activating/deactivating individual messages.

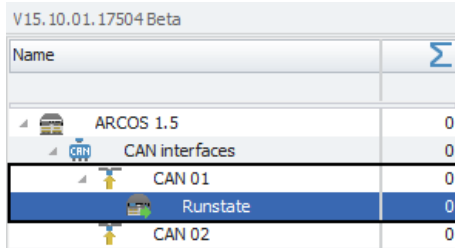
### 8.7.1 Add Runstate

In order to read out the logger state using the "Runstate" functionality, you will first need to add the "Runstate" interface to the CAN channel on which it should be sent. To do so, select the desired CAN channel in the measurement task tree, click the "Components" button in the Ribbon and then choose "Runstate".



### 8.7.2 Tree elements for Runstate

Once the “Runstate” interface has been added to a CAN channel, it will appear in the measurement task tree as a child element to that channel.



V15.10.01.17504 Beta	
Name	Σ
ARCOS 1.5	0
CAN interfaces	0
CAN 01	0
Runstate	0
CAN 02	0

### 8.7.3 Grid area for Runstate

In the “Grid area” you will be presented with an overview of the available signal messages. Each signal message has a subject and contains various signals concerning the respective subject. Marking/demarking active a signal message in the grid area will determine, whether the contained information will be read out and saved or discarded at export.

The description field tells you the subject of a signal message and the CAN ID field tells you the CAN ID in hexadecimal of the signal message.

Also you can find here two important functions, which are the “**Column chooser**” (→[4.3.1](#)) and the “**Filter editor**” (→[4.3.2](#)).

Name	Active	Description	CAN ID
Version information	<input checked="" type="checkbox"/>	Sends version information	0x1
Measurement time in µs	<input checked="" type="checkbox"/>	Sends the measurement time in µs	0x2
Absolute time in µs	<input checked="" type="checkbox"/>	Sends the absolute time in µs	0x3
Measurement delay in ms	<input type="checkbox"/>	Sends the current measurement delay in ms	0x4
Absolute time in s	<input type="checkbox"/>	Sends the absolute time in s	0x4
Absolute time formatted	<input type="checkbox"/>	Sends the absolute time formatted	0x5
System state information	<input type="checkbox"/>	Sends the system state information	0x6
Transfer state information	<input type="checkbox"/>	Sends the transfer state information	0x7

### 8.7.4 Details area for Runstate

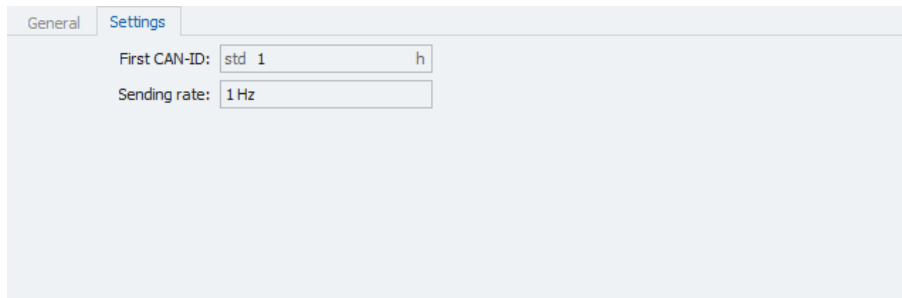
The details area for “Runstate” provides runstate settings.

#### General

Please refer to (→4.2.2).

#### Settings

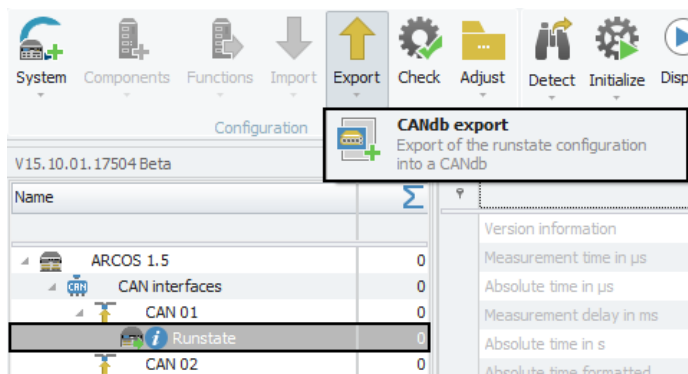
This tab contains the runstate settings.



- **First CAN-ID**  
This field defines the CAN identifier for sending of the first signal message. The remaining messages are sequentially put out on the subsequent identifiers.
- **Sending rate**  
Define the output rate for state information.

### 8.7.5 Export Runstate

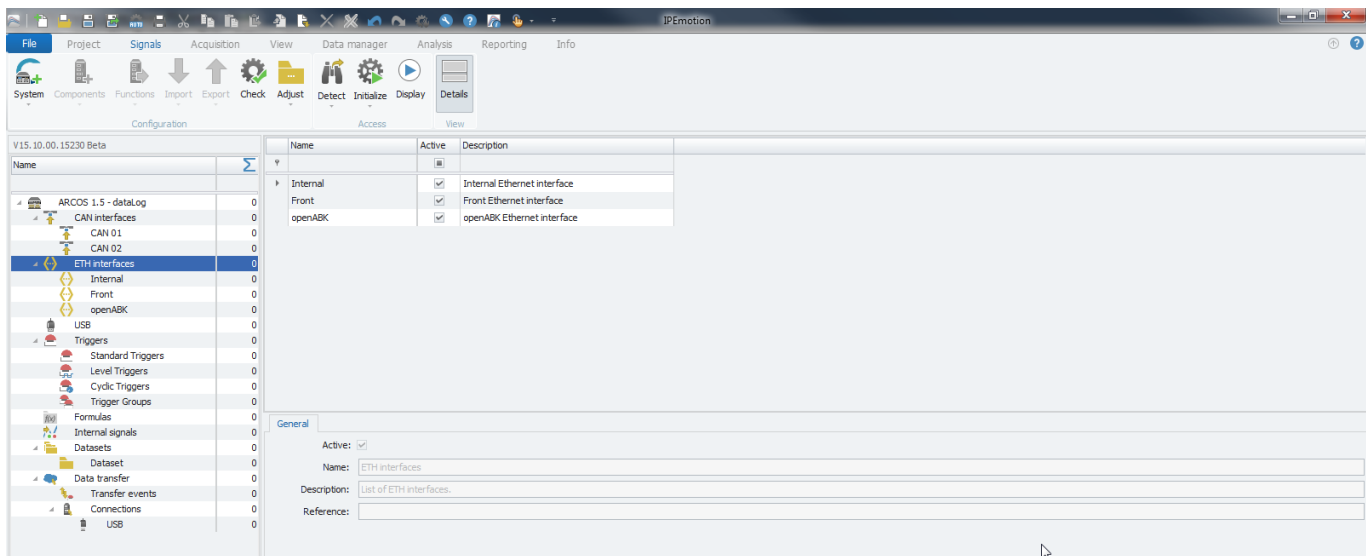
In order to make the loggers runstate signals available for further processing or analysis, they may be exported into a CANdb database. To do so, select the “Runstate” interface in the measurement task tree and then click the “Export” button in the Ribbon and choose “CANdb export”. Only the signal messages that have previously been marked active in the grid area will be exported.



## 8.8 ETH channels

All the ETH channels for your system are located in the tree element "ETH interfaces". According to the default settings, the tree element "ETH interfaces" will include a preset number of ETH channels. By clicking the tree element ETH Interfaces you will see all of its channels and signals in the grid area as well as a tab called **General** in the **Details area** which allows you to set a name and description. These settings apply to the entire element "ETH interfaces".

In the following will be described how to add ETH channels and adjust their settings (→8.8.3).

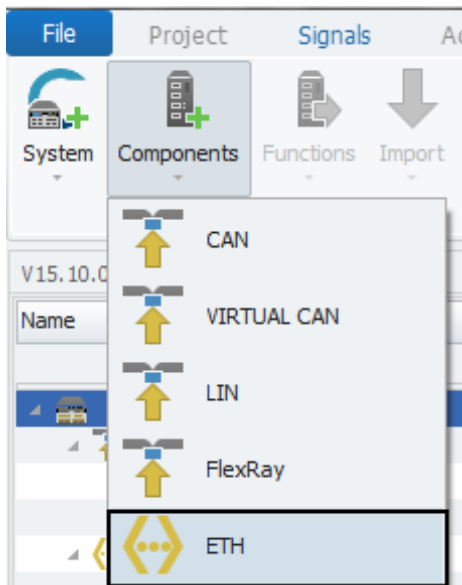


### 8.8.1 Storage method

In order to store all incoming traffic on an ETH channel use "PCAP" as a bus tracing method for storage. Please refer to (→ 14.9).

### 8.8.2 Adding ETH channels

ETH channels can be added by selecting the tree element “**ETH interfaces**”, then clicking the “**Components**” button and finally choosing the desired type of ETH channel you wish to add.



### 8.8.3 ETH settings

By selecting one of the ETH channels in the tree you will be able to define this channel's settings in the details area.

The screenshot shows the software interface with the following components:

- Menu Bar:** File, Project, Signals, Acquisition, View, Data manager, Analysis, Reporting, Info.
- Toolbar:** System, Components, Functions, Import, Export, Check, Adjust, Detect, Initialize, Display, Details.
- Configuration Area:** V15.10.00.15230 Beta
- Tree View:**
  - ARCOS 1.5 - dataLog (0)
    - CAN interfaces (0)
      - CAN 01 (0)
      - CAN 02 (0)
    - ETH interfaces (0)
      - Internal (0)
      - Front (0)
      - openABK (0)** (Selected)
      - ETH 04 (0)
  - USB (0)
    - Triggers (0)
      - Standard Triggers (0)
      - Level Triggers (0)
      - Cyclic Triggers (0)
      - Trigger Groups (0)
    - Formulas (0)
      - Internal signals (0)
    - Datasets (0)
      - Dataset (0)
    - Data transfer (0)
      - Transfer events (0)
    - Connections (0)
      - USB (0)
- Grid Area:**

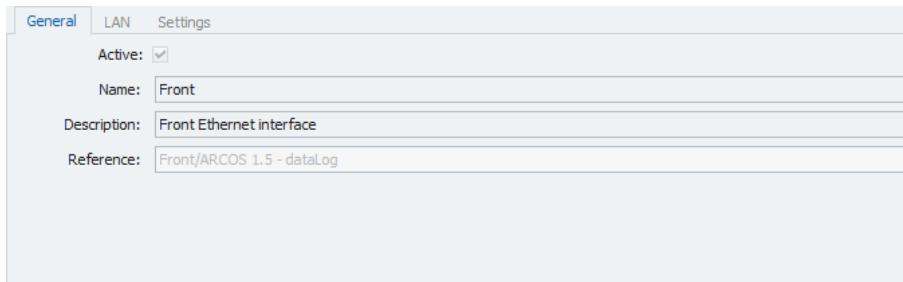
Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
?	<input type="checkbox"/>						
- Details Panel:**
  - General | LAN | Settings
  - Active:
  - Name: openABK
  - Description: openABK Ethernet interface
  - Reference: openABK/ARCOS 1.5 - dataLog



The same settings described in this section as part of the Details area can also be adjusted when selecting the tree element **ETH interfaces** and then directly changing the desired setting in the respective field of the **Grid area**.

### 8.8.3.1 General

This tab allows you to give a user specific name for the selected ETH channel if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The **Active** checkbox is by default marked as activated and cannot be changed.



General | LAN | Settings

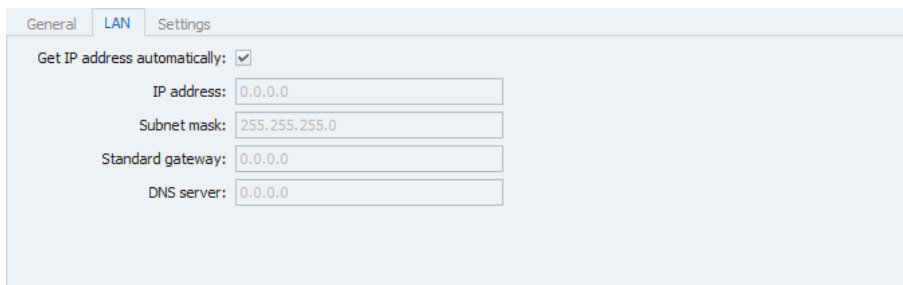
Active:

Name:

Description:

Reference:

### 8.8.3.2 LAN



General | LAN | Settings

Get IP address automatically:

IP address:

Subnet mask:

Standard gateway:

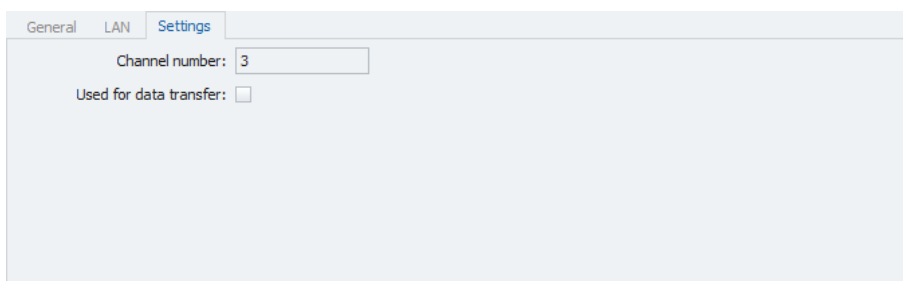
DNS server:

This tab allows you to set your current ETH channel's LAN settings. Whether it should receive an IP address automatically or not and consequently to manually set the channel's IP settings.



For the tree elements "Open ABK" and "Inernal" the IP settings cannot be adjusted.

### 8.8.3.3 Settings



General | LAN | Settings

Channel number:

Used for data transfer:

This tab allows you to set your current ETH channel's physical channel number and whether it should be used for data transfer.



The ETH channel **"Internal"** cannot be used for data transfer.

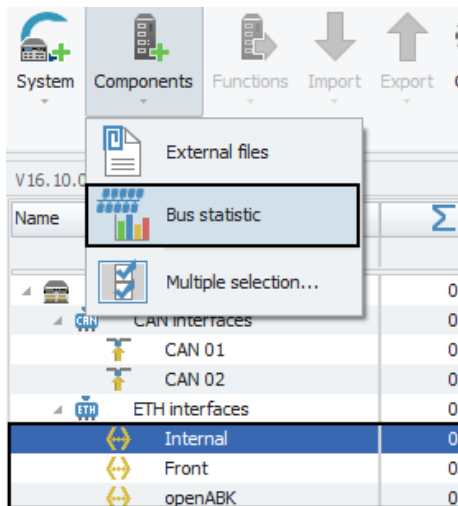
#### 8.8.4 ETH channel Bus statistic

The "Bus statistic" provides a range of statistics and status signals for the respective ETH channel. It contains information on the current state of the Bus, the Busload, as well as on the messages that have been received and errors that occurred.



The "Bus statistic" only shows statistics for the channel to which it belongs. For each channel that you would like to see the statistic, you will have to add the component "Bus statistic".

**8.8.4.1 Adding Bus statistics** Select the channel in the tree for which you would like to add "Bus statistic", then click the "Components" button in the Ribbon and choose "Bus statistic".



**8.8.4.2 Bus statistic signals** Once the component "Bus statistic" has been added to your channel, it will appear in the measurement task tree as a child element of this channel and the grid area will give you an overview of the available signals.

The signals included in "Bus statistics" are of the type "Internal signal" and may be adjusted in the same way. For more information on "Internal signals" please refer to (→8.22).



## Overview of signals

Subtype	Meaning	Unit
Internal Link status	0= Link down 1= Link up	-
Internal Link speed	Speed of the Link	(Mb/s)
Internal Duplex mode	Mode of operation	-
Internal Number of packets	Total number of transferred packets	-
Internal Current packet rate	Current rate of packets per second	(frames/s)
Internal Number of bytes	Total number of transferred bytes	-
Internal Byte rate	Current rate of bytes per second	(frames/s)
Internal Number of error frames	Total number of error frames occurred	-
Internal Error frame rate	Current rate of error frames per second	(frames/s)

## 8.9 ETH signals

For data acquisition you can import **A2L** or **Fibex** databases on the “Front” channel of your “ETH interface” or on any manually added ETH channel. The protocol for importing A2L databases is **XCPonUDP** and the protocol for importing Fibex databases is **SOME/IP**.

### 8.9.1 Storage method

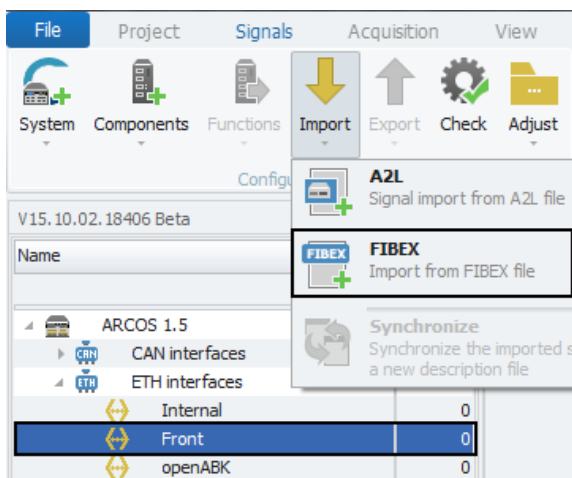
In order to store incoming signals on an ETH channel use one of the following signal storage methods.

- A2L (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

### 8.9.2 Importing ETH signals

#### 8.9.2.1 Importing Fibex files (SOME/IP)

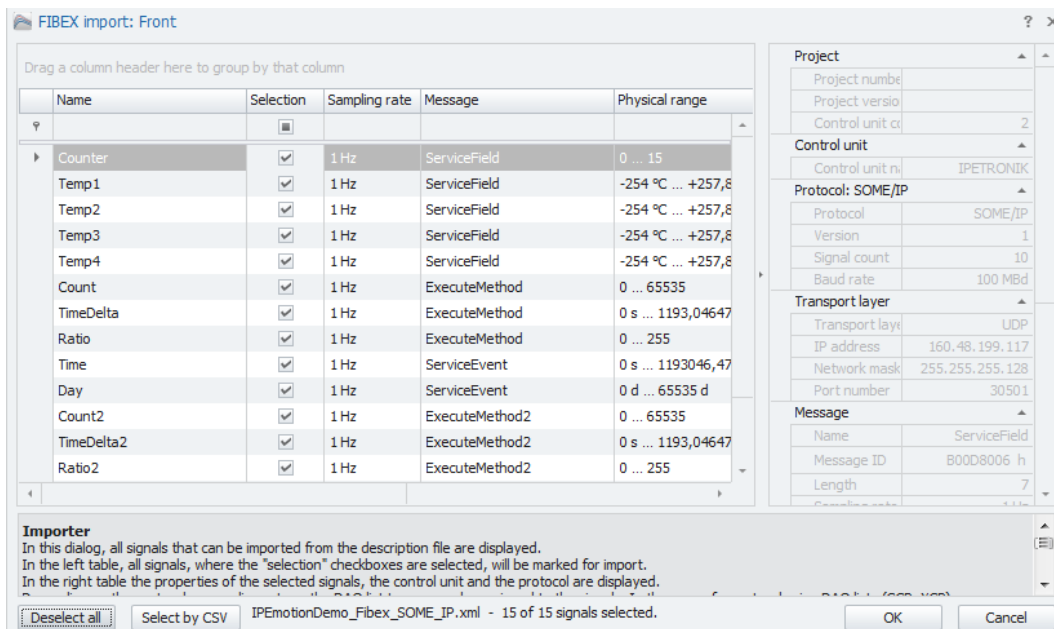
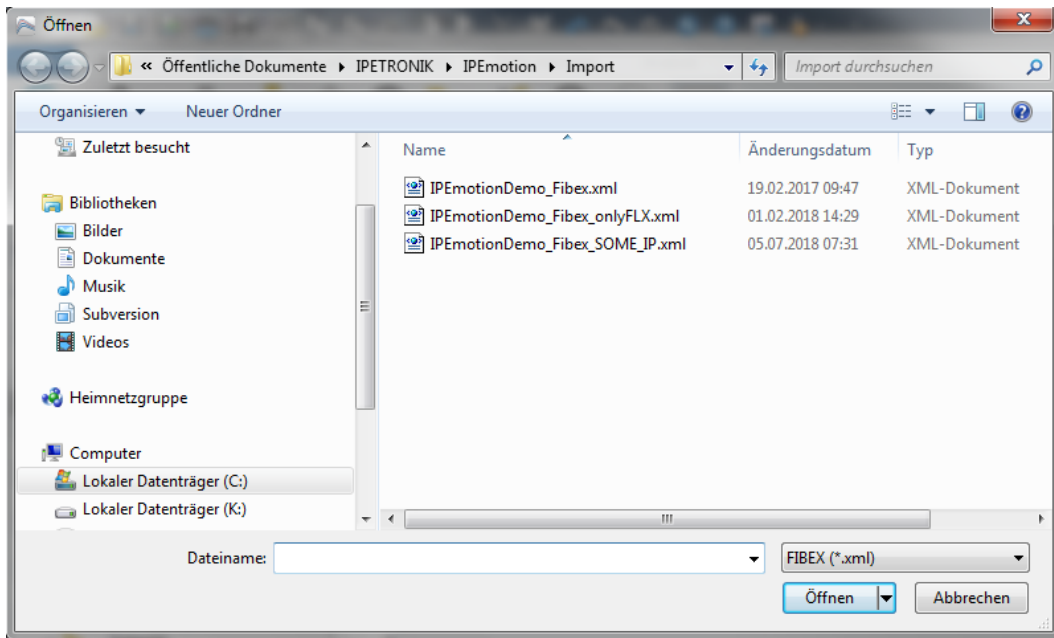
In order to import a Fibex database, select either the “Front” channel or any manually added “ETH xx” channel of your “ETH interface”, click the “Import” button in the Ribbon and then choose “Fibex”.



The following window lets you choose which file you wish to import. Choose the file you wish to import and click “Open”.

Next the “Importer” window will appear, that will present you with a range of importing options.

In this dialog, all signals that can be imported from the description file are displayed. In the left table, all signals, where the “selection” checkboxes are selected, will be marked for import. You can either choose manually, which signals to import, you can use the “Select/Deselect all” button on the bottom left, or you can use a CSV-file to determine which signals are to be imported, by clicking “Select by CSV” on the bottom left.



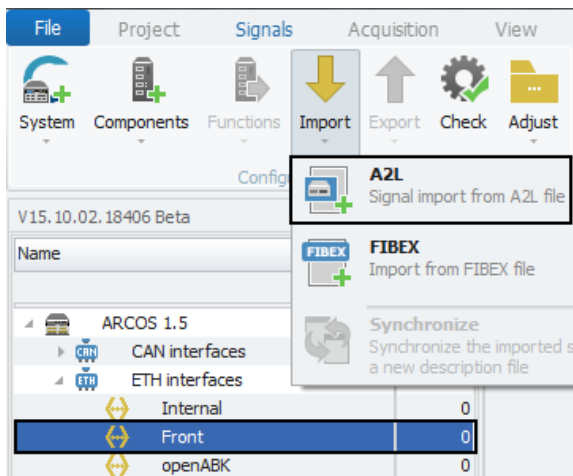
In the right table the metadata, properties of the selected signals, the control unit and the protocol are displayed.

Once you have chosen all the signals you wish to import, click "OK" to complete the import procedure.

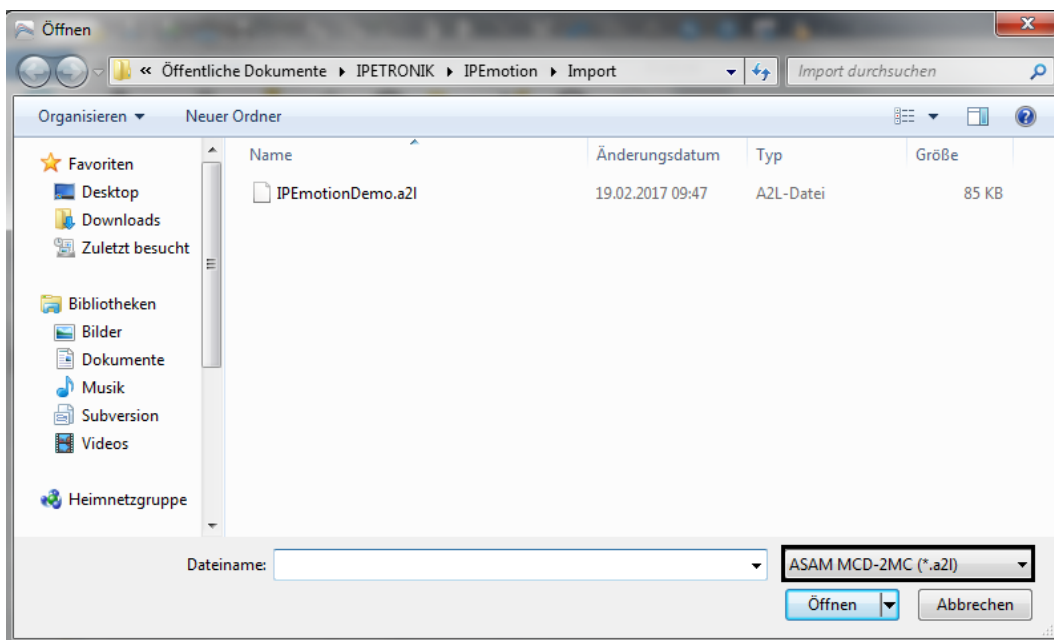
### 8.9.2.2 Importing A2L files (XCPonUDP)

In order to import an A2L database, select either the "Front" channel or any manually added "ETH xx" channel of your "ETH interface", click the "Import" button in the Ribbon and then choose "A2L".

The following window lets you choose which file you wish to import. Choose the file you Changes and errors excepted.



wish to import and click “Open”.

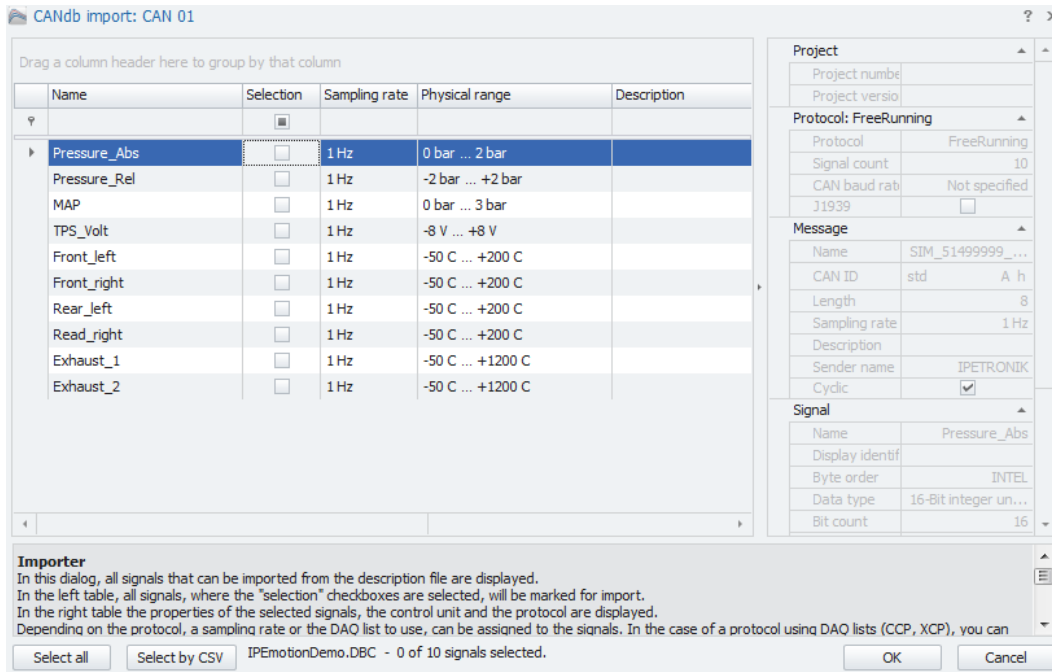


Next the “Importer” window will appear, that will present you with a range of importing options.

In this dialog, all signals that can be imported from the description file are displayed. In the left table, all signals, where the “selection” checkboxes are selected, will be marked for import. You can either choose manually, which signals to import, you can use the “**Select/Deselect all**” button on the bottom left, or you can use a CSV-file to determine which signals are to be imported, by clicking “**Select by CSV**” on the bottom left.

In the right table the metadata, properties of the selected signals, the control unit and the protocol are displayed.

Once you have chosen all the signals you wish to import, click “OK” to complete the import procedure.



Depending on the protocol, a sampling rate or the DAQ list to use, can be assigned to the signals.

In the case of a protocol using DAQ lists (CCP, XCP), you can specify via the column selection dialog, if the signals are configured by the sampling rate or a DAQ list. To achieve this you should open the column selection dialog, via the context menu of the table header, and then drag the desired column ("sampling" or "DAQ list") from the column selection dialog to the table header. The other column is removed automatically.

If in a protocol based on DAQ lists, sampling rates are used for the signal configuration, during import the signals are assigned to the available DAQ list with the most suitable sampling rate.

In case of protocols supporting array signals, you can specify via the "split array" column whether all the signals of the array or just the first to be imported. If this column does not appear it can be moved from the column selection dialog into the table.

### 8.9.2.3 IP settings when Importing A2L files (XCPonUDP)

When importing signals to an ETH channel from an A2L file, the PlugIn will automatically set the description file's source IP address (to be found in the import properties of the description file) as the ETH channel's IP address. If DHCP is activated it will be deactivated and the existing IP address will be replaced with the source IP address. Also if DHCP is deactivated but the IP address is set to 0.0.0.0, the existing IP address will be replaced with the source IP address.

However, if DHCP is deactivated and a user specific IP address **other** than 0.0.0.0 is set, this IP address will **not** be replaced.

Changes and errors excepted.

General LAN Settings

Get IP address automatically:

IP address: 192.168.0.254

Subnet mask: 255.255.255.0

Standard gateway: 0.0.0.0

DNS server: 0.0.0.0

Import: IPEmotionDemo XCP\_V1.4

Command-Timeout: 2000

ReInit Timeout: 5

Time stamp size: 4

Time stamp resolution: DAQ\_TIMESTAMP\_UNIT...

Time stamp ticks: 1

Time stamp fixed:

---

Ethernet

Target IP address: 192.168.0.2

Port: 11000

Hostname:

MediaType: Auto

Source IP address: 192.168.0.254

### 8.9.3 Signal properties

#### 8.9.3.1 Signal properties for SOME/IP

The signal properties for SOME/IP signals are mainly the same as for Fibex signals on Flexray channels. Please refer to (→ [8.13.4](#)).

The only difference between Fibex signals on ETH and on Flexray is, that importing Fibex signals on ETH will result in one extra layer of tree elements called **“Socket x”**.

ETH interfaces	15
Internal	0
Front	15
IPETRONIK	10
IPEmotionDemo...	10
Socket 1	10
Service...	5
Execut...	3
Service...	2

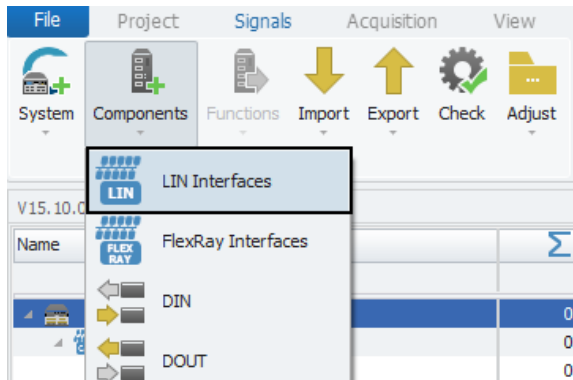
The socket layer then contains the signal messages and offers an extra possibility for grouping these messages.

#### 8.9.3.2 Signal properties for XCPonUDP

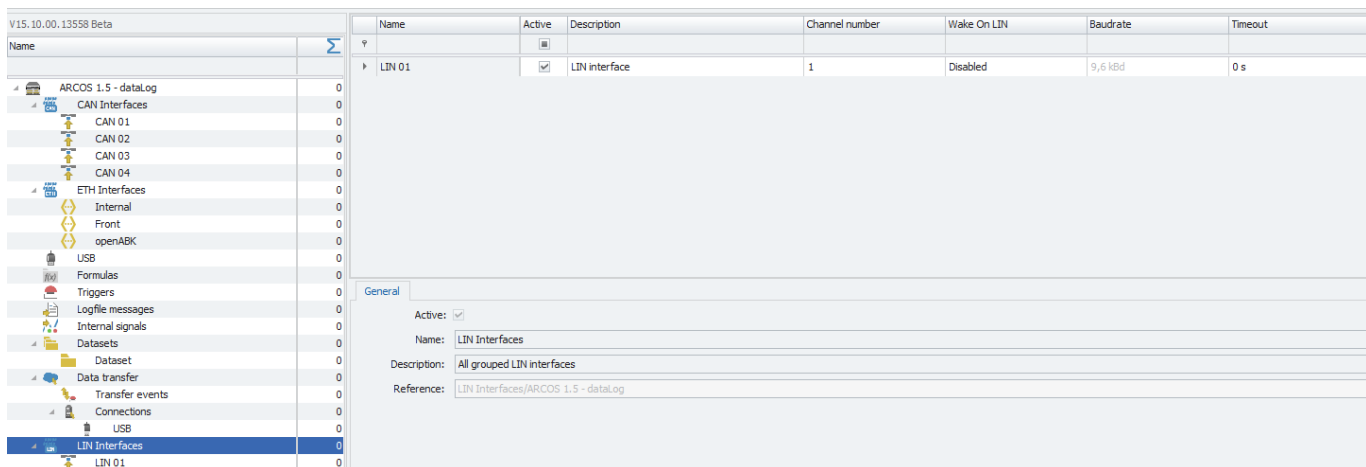
The signal properties for XCPonUDP signals are the same as for XCP signals on CAN channels. Please refer to (→ [8.3.4](#)).

## 8.10 LIN channels

To work with “LIN channels”, you will first have to add the tree element “LIN interfaces” to your measurement task tree. To do so, select your system in the measurement task tree, click the “Components” button in the ribbon and choose “LIN interfaces” from the resulting dropdown menu.



Once the “LIN interfaces” component has been added it will appear in the measurement task tree as a tree element with one “LIN channel” as a child element.



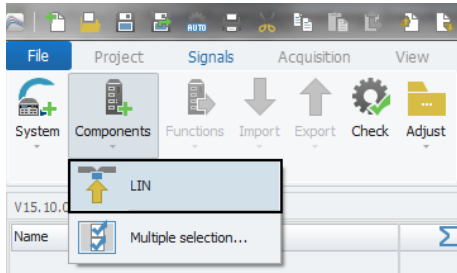
### 8.10.1 Storage method

In order to store all incoming traffic on a LIN channel use a bus tracing method for storage. Please refer to (→ [14.8](#)).



### 8.10.2 Adding LIN channels

LIN channels can be added by selecting the tree element **"LIN interfaces"**, then clicking the **"Components"** button and then choosing **"LIN channel"**.

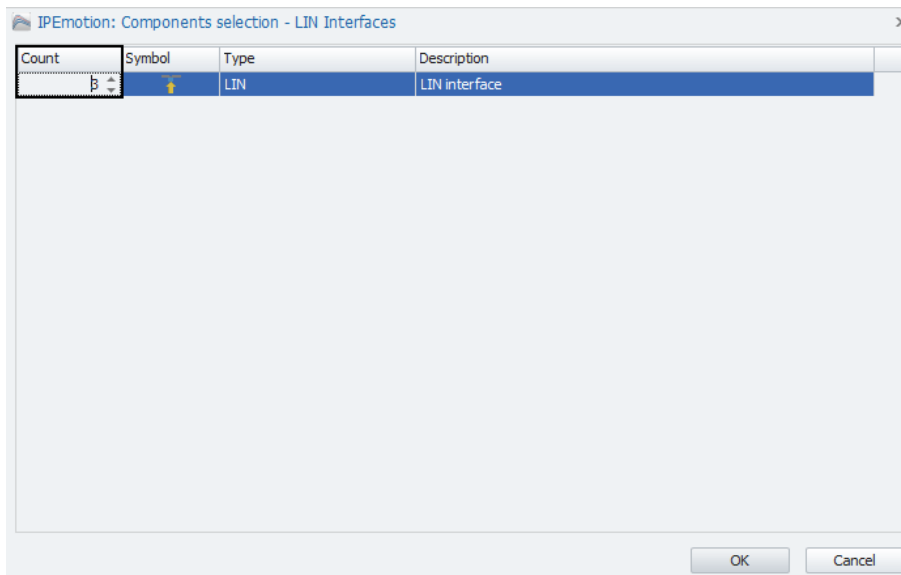


#### LIN

Adds a LIN channel that corresponds to a physical LIN channel of your logger. For instructions on LIN settings refer to (→[8.10.3](#)).

#### Multiple selection

Allows you to add multiple LIN channels of both types at the same time. To do so set the counter for each type to the desired number of channels that you wish to add as marked in the figure below.



### 8.10.3 LIN settings

By selecting one of the LIN channels in the tree you will be able to define this channel's settings in the details area.



The same settings described in this section as part of the Details area can also be adjusted when selecting the tree element **LIN interfaces** and then directly changing the desired setting in the respective field of the **Grid area**.

## 8.10 LIN CHANNELS

V15.10.00.13558 Beta

Name	Σ	Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
CAN 02	0		<input type="checkbox"/>						
CAN 03	0		<input type="checkbox"/>						
CAN 04	0		<input type="checkbox"/>						
ETH Interfaces	0								
Internal	0								
Front	0								
openABK	0								
USB	0								
Formulas	0								
Triggers	0								
Logfile messages	0								
Internal signals	0								
Datasets	0								
Dataset	0								
Data transfer	0								
Transfer events	0								
Connections	0								
USB	0								
LIN Interfaces	0								
LIN 01	0		<input checked="" type="checkbox"/>						
LIN 02	0		<input type="checkbox"/>						
LIN 03	0		<input type="checkbox"/>						
LIN 04	0		<input type="checkbox"/>						

General LIN Wake On LIN Hardware

Active:

Name: LIN 01

Description: LIN interface

Reference: LIN 01/ARCOS 1.5 - dataLog

### 8.10.3.1 General

This tab allows you to give a user specific name for the selected LIN channel if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The **Active** checkbox is by default marked as activated and cannot be changed.

General LIN Wake On LIN Hardware

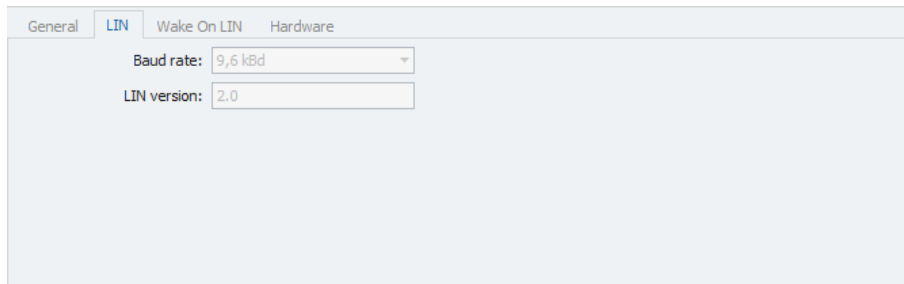
Active:

Name: LIN 01

Description: LIN interface

Reference: LIN 01/ARCOS 1.5 - dataLog

### 8.10.3.2 LIN



The screenshot shows a configuration window with four tabs: General, LIN, Wake On LIN, and Hardware. The LIN tab is active. It contains two fields: 'Baud rate' set to '9,6 kBd' and 'LIN version' set to '2.0'.

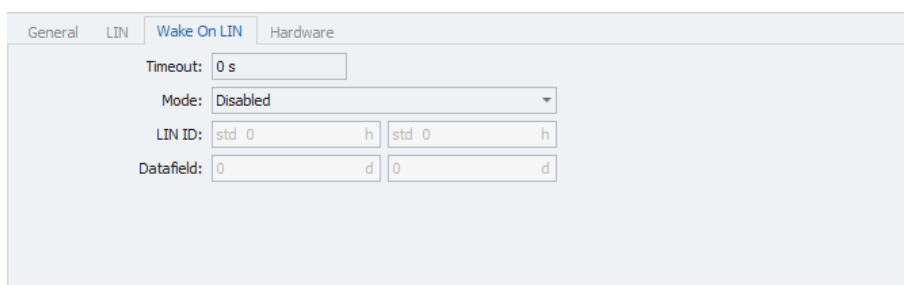
#### Baud rate

The Baud rate is determined automatically by the logger.

#### LIN version

This field shows you the version of the LIN protocol that is used as described in the "Description file". This field cannot be changed.

### 8.10.3.3 Wake On LIN



The screenshot shows a configuration window with four tabs: General, LIN, Wake On LIN, and Hardware. The Wake On LIN tab is active. It contains several fields: 'Timeout' set to '0 s', 'Mode' set to 'Disabled', 'LIN ID' with two fields both set to 'std 0', and 'Datafield' with two fields both set to '0'.

#### Timeout

For Wake On LIN, timeout has a special significance. It defines how long a waking channel must be inactive to be recognized so and therefore allow for the logger to shutdown. If timeout is recognized, an entry is made in the log file and an error message with an alert appears on the display, which has to be acknowledged.

#### Mode

This dropdown menu allows you to set the wake-up function for your selected LIN channel.

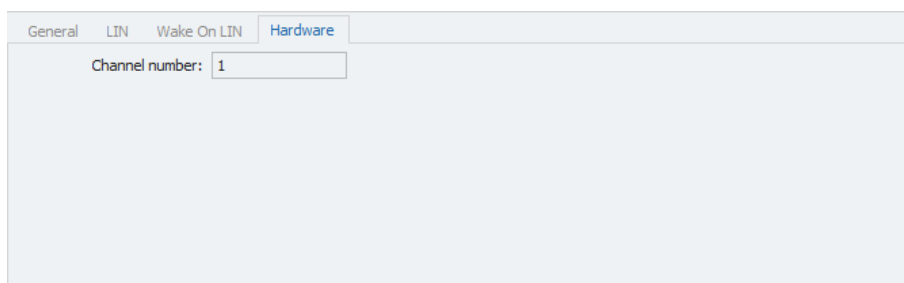
Wake on LIN type	Characteristics
Disabled	No start on LIN messages, lowest energy consumption.
Enabled	Start on a LIN message, with first messages lost; low energy consumption.
Enabled (no message lost)	Start on LIN message, with no message lost; slightly higher idle current.
Keep awake	The logger starts to other awakenings, but only shuts down if all the awakening-conditions are no longer fulfilled and if the keep awake condition is no longer fulfilled.

### LIN ID - Settings for starting on a specific LIN ID

This functionality is the same for CAN and LIN interfaces and has been described in great detail earlier. Please refer to (→[8.1.3.4](#)).

#### 8.10.3.4 Hardware (Channel number)

This tab allows you to set a **Channel number** for the selected LIN channel. This channel number has to be unique within the LIN interface.



The screenshot shows a software interface with four tabs: 'General', 'LIN', 'Wake On LIN', and 'Hardware'. The 'Hardware' tab is selected. Below the tabs, there is a label 'Channel number:' followed by a text input field containing the number '1'.



For better orientation and in order to avoid confusion regarding Channelnumbers and -names, a Channels physical number can be found in the logger's **"Web Interface"** and set accordingly.

#### 8.10.4 LIN channel Bus statistic

The "Bus statistic" provides a range of statistics and status signals for the respective LIN channel. It contains information on the current state of the Bus, the Busload, as well as on the messages that have been received and errors that occurred.

This functionality is the same for CAN and LIN interfaces and has been described in great detail earlier. Please refer to (→[8.1.5](#)).

## 8.11 LIN signals

### 8.11.1 Storage method

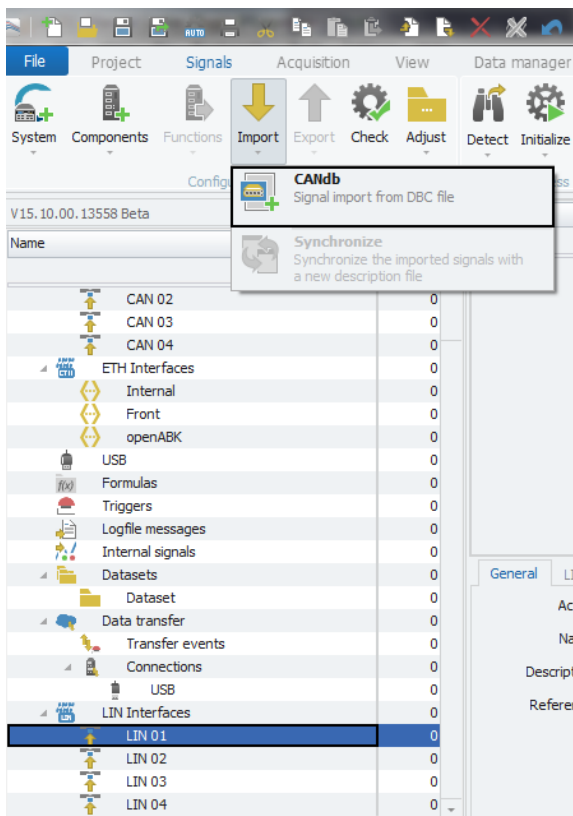
In order to store incoming signals on a LIN channel use one of the following signal storage methods.

- ATEX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

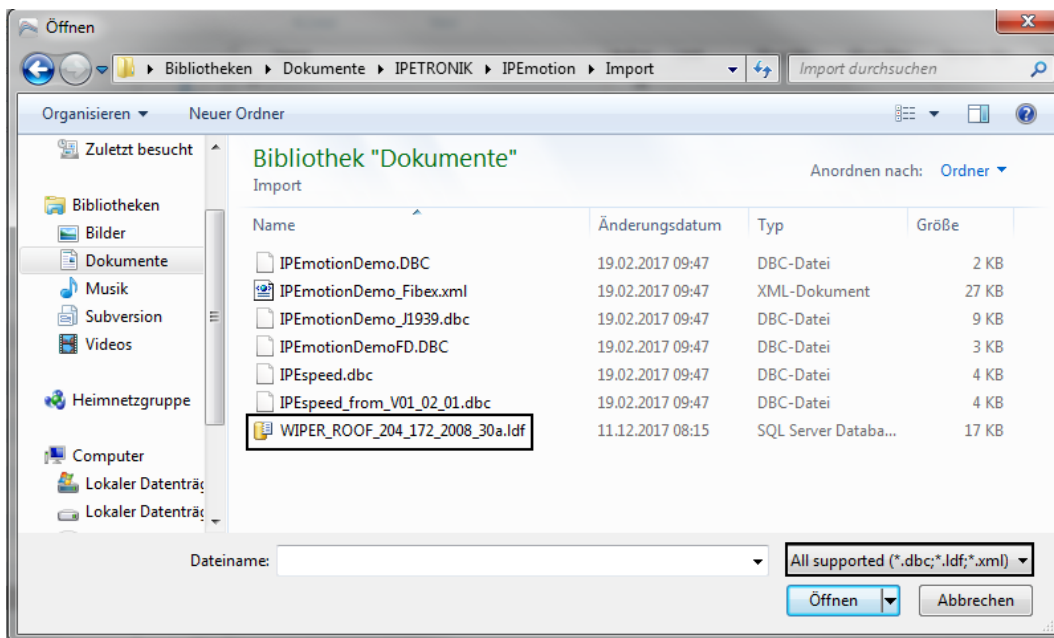
### 8.11.2 Importing LIN signals

This section explains how to import LIN signals. The filetype to be used for importing LIN signals is a **LINdb (LDF file)**.

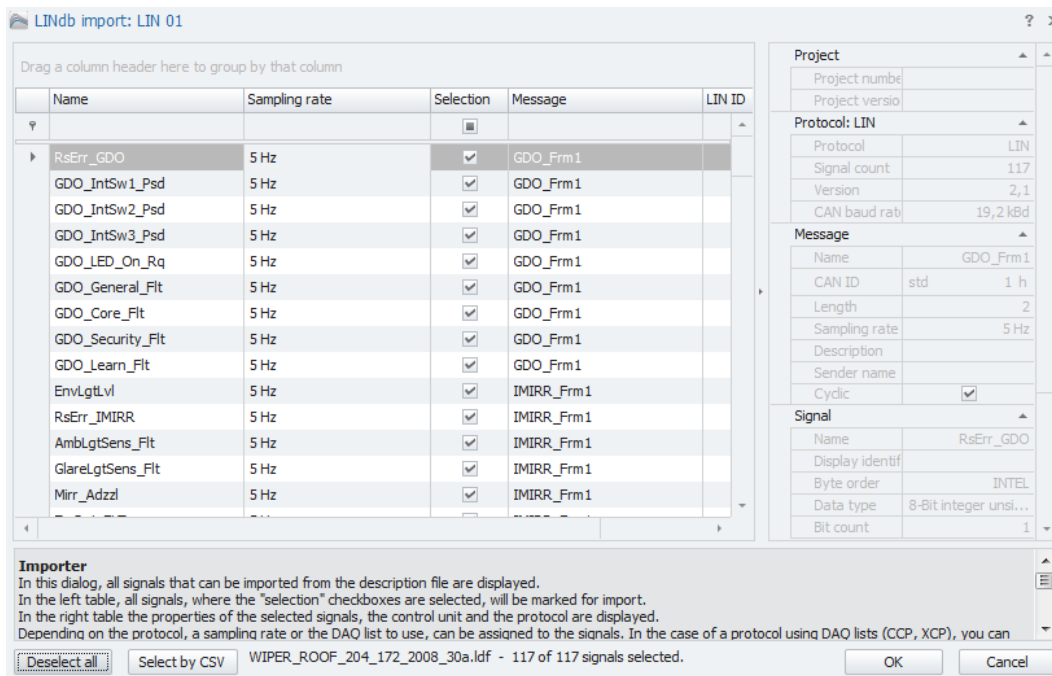
To import Signals, select the LIN channel to which you wish to import your signal in the tree, click the **“Import”** button in the ribbon and then choose **“CANdb”** as description file for the import. For more information on the **“description file”** refer to (→[8.2.4.1](#)).



The following window lets you choose which file you wish to import. The dropdown menu on the bottom right of the window shows you, which filetypes are available. Choose the file you wish to import and click “Open”.



Once you have opened your file, the “Importer” window will appear, that will present you with a range of importing options.



In this dialog, all signals that can be imported from the description file are displayed. In the left table, all signals, where the “selection” checkboxes are selected, will be marked for import. You can either choose manually, which signals to import, you can use the “Select/Deselect all” button on the bottom left, or you can use a CSV-file to determine which signals are to be imported, by clicking “Select by CSV” on the bottom left.

In the right table the properties of the selected signals, the control unit and the protocol are displayed.

Once you have chosen all the signals you wish to import, click “OK” to complete the import procedure.



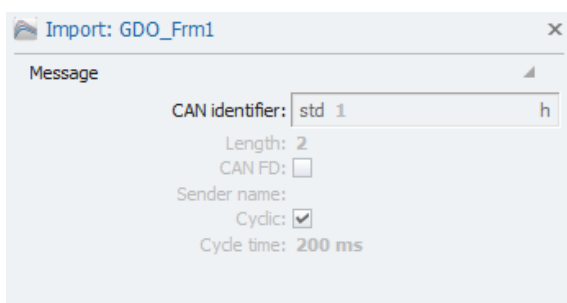
Multiple description files can be imported into the same LIN channel.

### 8.11.3 Import properties

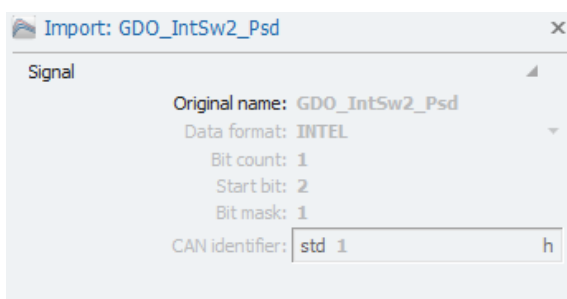
The “Import properties” of a description file, Message or signal allow you to see certain properties such as the Data format, the LIN identifier, the Bit mask, the start bit, bit count and more. It shows the signal’s properties as described by the description file.



Example for “Import properties” of a LIN description file

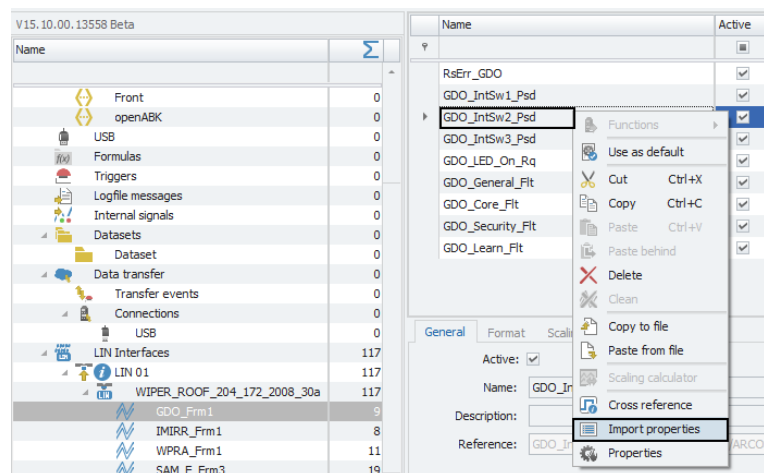


Example for “Import properties” of a LIN Message



Example for “Import properties” of a LIN signal

To access the “Import properties” rightclick on any desired description file, Message or signal and then choose “Import properties” from the resulting context menu.



## 8.11.4 Signal properties














### 8.11.4.1 Tree elements for LIN signals

After having successfully imported the desired signals to your LIN channel, this channel will contain two new layers of child elements in the measurement task tree: The “**Description file**” and the “**Message**”.

#### Description file

The “description file” is a database file which contains signal information and can be used to import those signals into a Signal channel in IPEmotion. The filetypes which are supported by the CAETEC Plugin for IPEmotion depend on the type of signal you wish to import.

The symbol in the left part of the tree element shows you the type of “description file” you imported (in this case a “LDF” file), then follows the name of the imported “description file” (in this case “WIPER\_ROOF\_...”) and on the right is a number indicating how many signals the “description file” contains (in this case 117).

 LIN Interfaces	117
 LIN 01	117
 WIPER_ROOF_204_172_2008_30a	117
 GDO_Frm1	9
 IMIRR_Frm1	8
 WPRA_Frm1	11
 SAM_F_Frm3	19
 OHCM_Frm	20
 LRSM_Frm1	21
 SAM_F_Frm1	20
 SAM_F_Frm2	9
 LIN 02	0
 LIN 03	0



## Message

Each “description file” can contain one or more “Messages”, which then contain the actual signals. A “Message” can be found in the “Measurement task tree” as a child element of the “description file”, it belongs to.

Each “Message” can, again, contain one or more signals, which is indicated by the number on the right of the “Message’s” name.

LIN Interfaces	117
LIN 01	117
LIN WIPER_ROOF_204_172_2008_30a	117
GDO_Frm1	9
IMIRR_Frm1	8
WPRA_Frm1	11
SAM_F_Frm3	19
OHCM_Frm	20
LRSM_Frm1	21
SAM_F_Frm1	20
SAM_F_Frm2	9
LIN 02	0
LIN 03	0

### 8.11.4.2 Grid area for LIN signals

In the “grid area” you will be presented with an overview of your selected LIN channel’s signals. Also you can find here two important functions, which are the “**Column chooser**” (→4.3.1) and the “**Filter editor**” (→4.3.2).

### 8.11.4.3 Details area for LIN signals

The Details area shows settings either for the selected tree element (“description file” or “Message”) or the selected signal in the grid area. In case a tree element is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

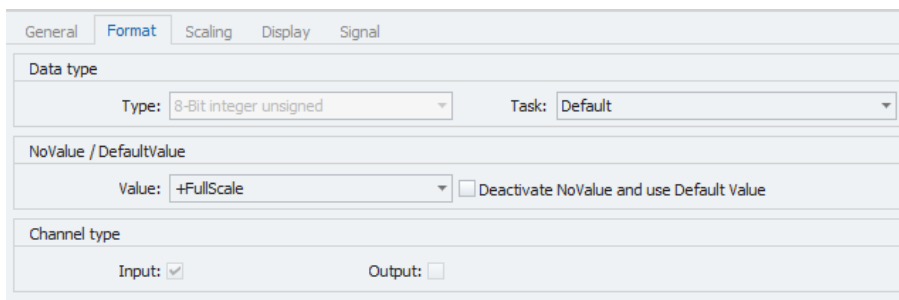
In case a signal is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

Please refer to (→[4.2.2](#)).

#### Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case “8-Bit integer unsigned”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

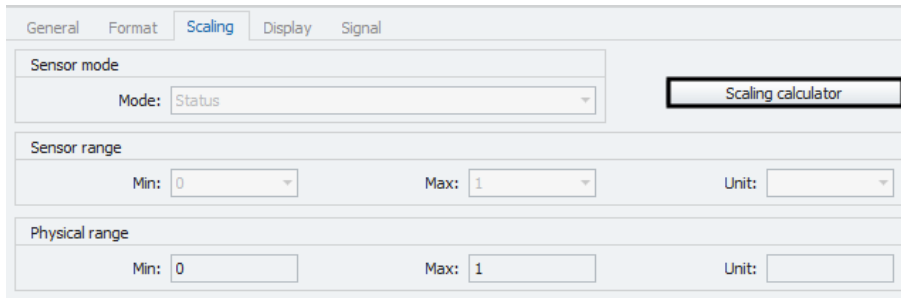
This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a “Input” channel or “Output” channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

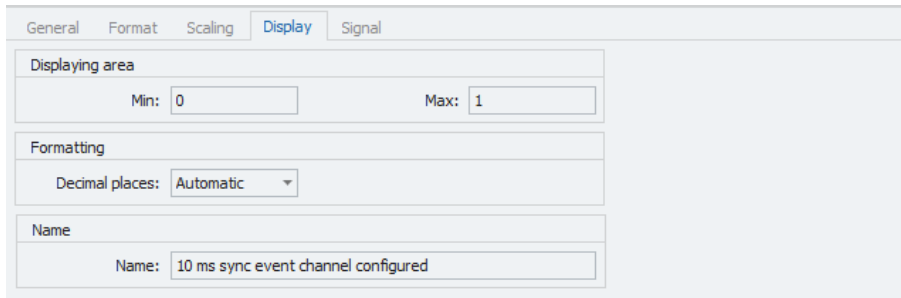
Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

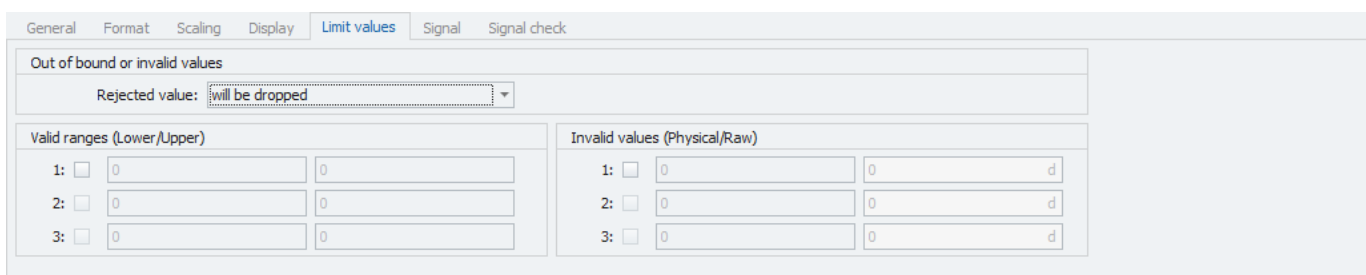
The dropdown menu "**Decimal places**" allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Limit values

This tab allows you to define limit values for a signal and what action to take upon a limit value violation.



- **Rejected value**

Define what happens to a value, that has been rejected because it is out of bound or invalid. By default this value will be dropped. It can also be written as NaN.

- **Valid ranges (Lower/Upper)**

Define up to three ranges of valid signal values. Activate a range in order to define its upper/lower value (datatype double). Range 2 can only be activated if range 1 is and range 3 can only be activated if range 2 is.

- **Invalid values (Physical/Raw)**

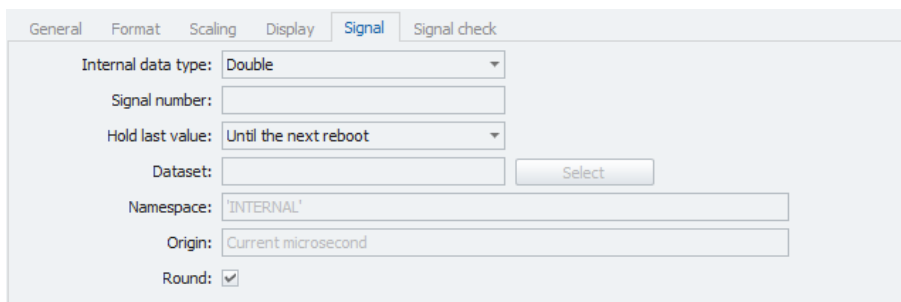
Define up to three invalid values. Activate an invalid value in order to define the physical value (datatype double) or raw value (datatype integer). If one of the two has been typed in, the other will be calculated according to the scale/offset settings in the scaling calculator.

Invalid value 2 can only be activated if invalid value 1 is and invalid value 3 can only be activated if invalid value 2 is.

For details on how to use the "Scaling calculator" please refer to the [IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling"](#).

## Signal

This tab allows you to define signal settings.



- **Internal data type**

Assign an internal data type to the signal. Available data types are "Double" and "String".

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting "Hold last value" has been set to "Until end of dataset", you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to "NaN (Not a Number)" and will be displayed as "-" in a display.

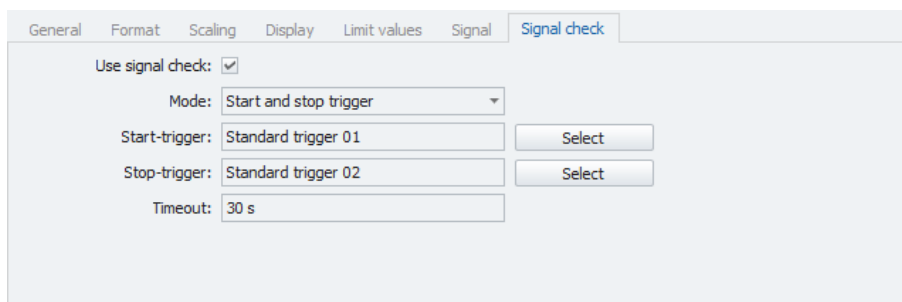
- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.

### Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.

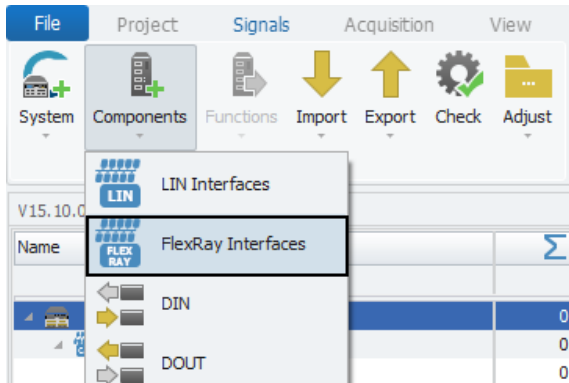


The screenshot shows a software window with several tabs: General, Format, Scaling, Display, Limit values, Signal, and Signal check. The 'Signal check' tab is active. It contains the following settings:

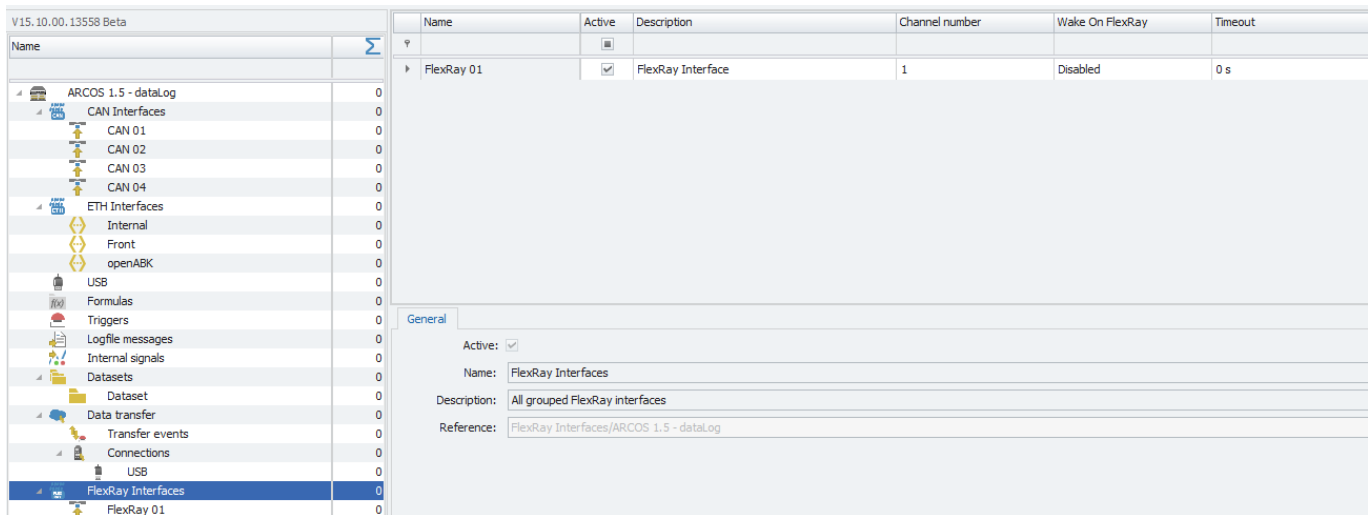
- Use signal check:
- Mode: Start and stop trigger (dropdown menu)
- Start-trigger: Standard trigger 01 (text field) with a Select button
- Stop-trigger: Standard trigger 02 (text field) with a Select button
- Timeout: 30 s (text field)

## 8.12 FlexRay channels

To work with “FlexRay channels”, you will first have to add the tree element “FlexRay interfaces” to your measurement task tree. To do so, select your system in the measurement task tree, click the “Components” button in the ribbon and choose “FlexRay interfaces” from the resulting dropdown menu.



Once the “FlexRay interfaces” component has been added it will appear in the measurement task tree as a tree element with one “FlexRay channel” as a child element.



### 8.12.1 Storage method

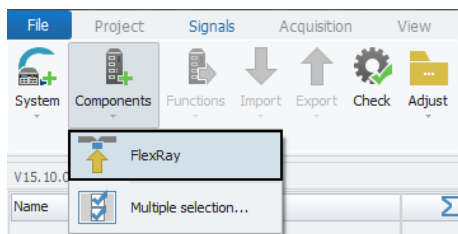
In order to store all incoming traffic on a FlexRay channel use a bus tracing method for storage. Please refer to (→ [14.8](#)).

### 8.12.2 Adding FlexRay channels

FlexRay channels can be added by selecting the tree element "FlexRay interfaces", then clicking the "Components" button and then choosing "FlexRay channel".

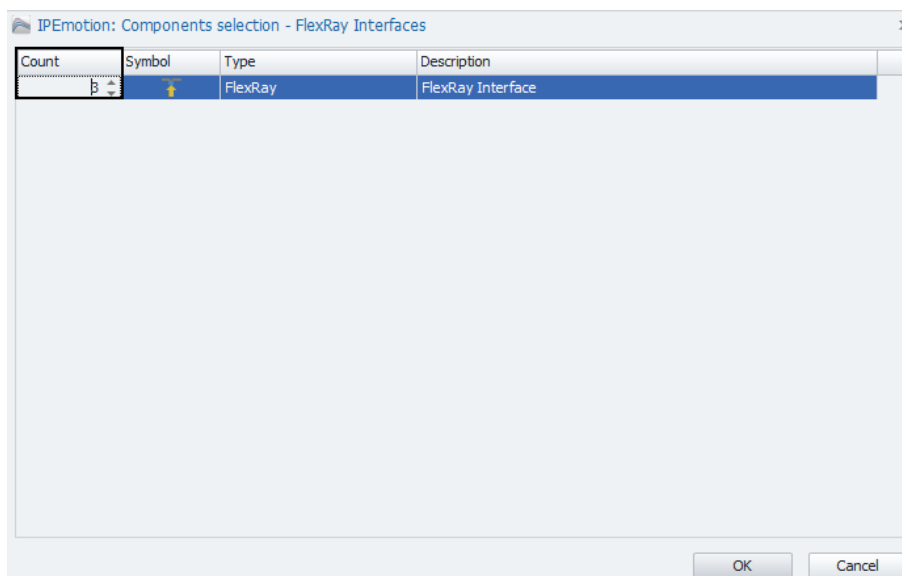
#### FlexRay

Adds a FlexRay channel that corresponds to a physical FlexRay channel of your logger. For instructions on FlexRay settings refer to (→8.12.3).



#### Multiple selection

Allows you to add multiple FlexRay channels of both types at the same time. To do so set the counter for each type to the desired number of channels that you wish to add as marked in the figure below.



### 8.12.3 FlexRay settings

By selecting the "FlexRay Interfaces" in the tree and then choosing one of the FlexRay channels in the grid area you will be able to define this channel's settings in the details area.



The same settings described in this section as part of the Details area can also be adjusted when selecting the desired channel in the grid area and then directly changing the desired setting in the respective field of the **Grid area**.



V15.10.00.13558 Beta

Name	Active	Description	Channel number	Wake On FlexRay	Timeout
FlexRay 01	<input checked="" type="checkbox"/>	FlexRay Interface	1	Disabled	0 s
FlexRay 02	<input checked="" type="checkbox"/>	FlexRay Interface	2	Disabled	0 s
FlexRay 03	<input checked="" type="checkbox"/>	FlexRay Interface	3	Disabled	0 s
FlexRay 04	<input checked="" type="checkbox"/>	FlexRay Interface	4	Disabled	0 s

General Wake On FlexRay Hardware

Active:

Name: FlexRay 01

Description: FlexRay Interface

Reference: FlexRay 01/ARCOS 1.5 - dataLog

### 8.12.3.1 General

This tab allows you to give a user specific name for the selected FlexRay channel if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The **Active** checkbox is by default marked as activated and cannot be changed.

General Wake On FlexRay Hardware

Active:

Name: FlexRay 01

Description: FlexRay Interface

Reference: FlexRay 01/ARCOS 1.5 - dataLog

### 8.12.3.2 Wake On FlexRay

General Wake On FlexRay Hardware

Timeout: 0 s

Mode: Disabled

#### Timeout

Timeout defines how long a waking channel must be inactive to be recognized so and therefore allow for the logger to shutdown. If timeout is recognized, an entry is made in the log file and an error message with an alert appears on the display, which has to be acknowledged.

Changes and errors excepted.

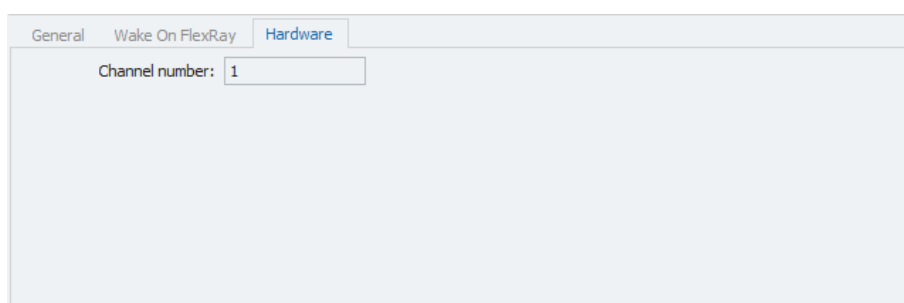
## Mode

This dropdown menu allows you to set the wake-up function for your selected FlexRay channel.

Wake on FlexRay type	Characteristics
Disabled	No start on FlexRay messages, lowest energy consumption.
Enabled	Start on a FlexRay message, with first messages lost; low energy consumption.
Keep awake	The logger starts to other awakenings, but only shuts down if all the awakening-conditions are no longer fulfilled and if the keep awake condition is no longer fulfilled.

### 8.12.3.3 Hardware (Channel number)

This tab allows you to set a **Channel number** for the selected FlexRay channel. This channel number has to be unique within the FlexRay interface.



General Wake On FlexRay **Hardware**

Channel number:



For better orientation and in order to avoid confusion regarding Channelnumbers and -names, a Channels physical number can be found in the logger's **"Web Interface"** and set accordingly.

### 8.12.4 FlexRay channel Bus statistic

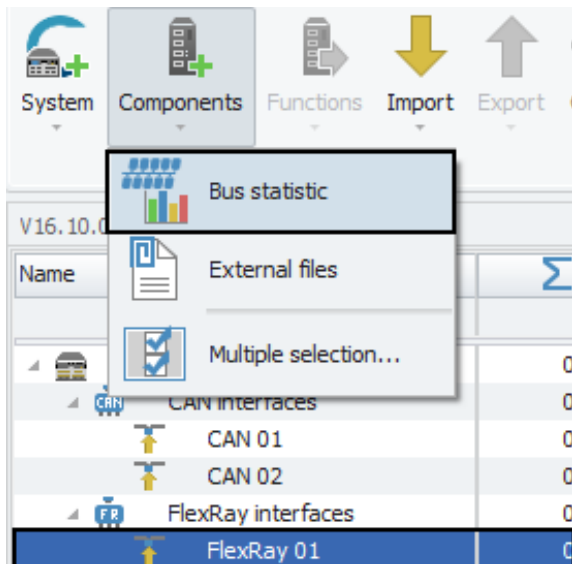
The “Bus statistic” provides a range of statistics and status signals for the respective FlexRay channel. It contains information on the current state of the Bus, the Busload, as well as on the messages that have been received and errors that occurred.



The “Bus statistic” only shows statistics for the channel to which it belongs. For each channel that you would like to see the statistic, you will have to add the component “Bus statistic”.

#### 8.12.4.1 Adding Bus statistics

Select the channel in the tree for which you would like to add “Bus statistic”, then click the “Components” button in the Ribbon and choose “Bus statistic”.



### 8.12.4.2 Bus statistic signals

Once the component “Bus statistic” has been added to your channel, it will appear in the measurement task tree as a child element of this channel and the grid area will give you an overview of the available signals.

The signals included in “Bus statistics” are of the type “Internal signal” and may be adjusted in the same way. For more information on “Internal signals” please refer to (→[8.22](#)).

#### Overview of signals

Subtype	Meaning	Unit
FlexRay xx Controller state	nan= Channel not available 1= Bus on 2= Bus warning 3= Bus off	-
FlexRay xx Number of static messages	Number of static messages since beginning of measurement	-
FlexRay xx Number of dynamic messages	Number of dynamic messages since beginning of measurement	-
FlexRay xx Number of null frames	Number of frames since beginning of measurement	-
FlexRay xx Message rate total	Average of messages of all types per second	(frames/s)
FlexRay xx Number of error frames	Number of error frames	-
FlexRay xx Error frame rate	Average of errors per second	(frames/s)
FlexRay xx Message rate of static messages	Average of static messages per second	(frames/s)
FlexRay xx Message rate of dynamic messages	Average of dynamic messages per second	(frames/s)
FlexRay xx Null frame rate	Average of null frames per second	(frames/s)

## 8.13 FlexRay signals

### 8.13.1 Storage method

In order to store incoming signals on a FlexRay channel use one of the following signal storage methods.

- ATEX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

### 8.13.2 Importing FlexRay signals

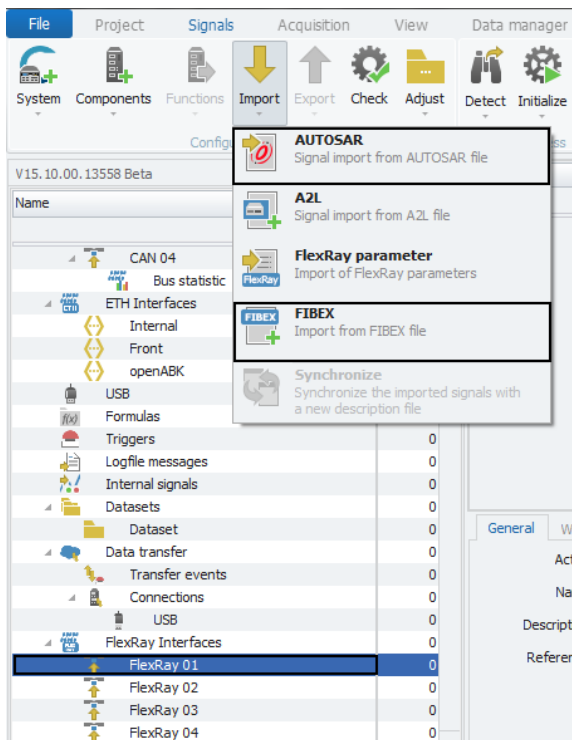
This section explains how to import FlexRay signals. There are three different filetypes which can be used in order to import a single FlexRay signal or a group of FlexRay signals:

- Autosar files
- A2L files
- Fibex files

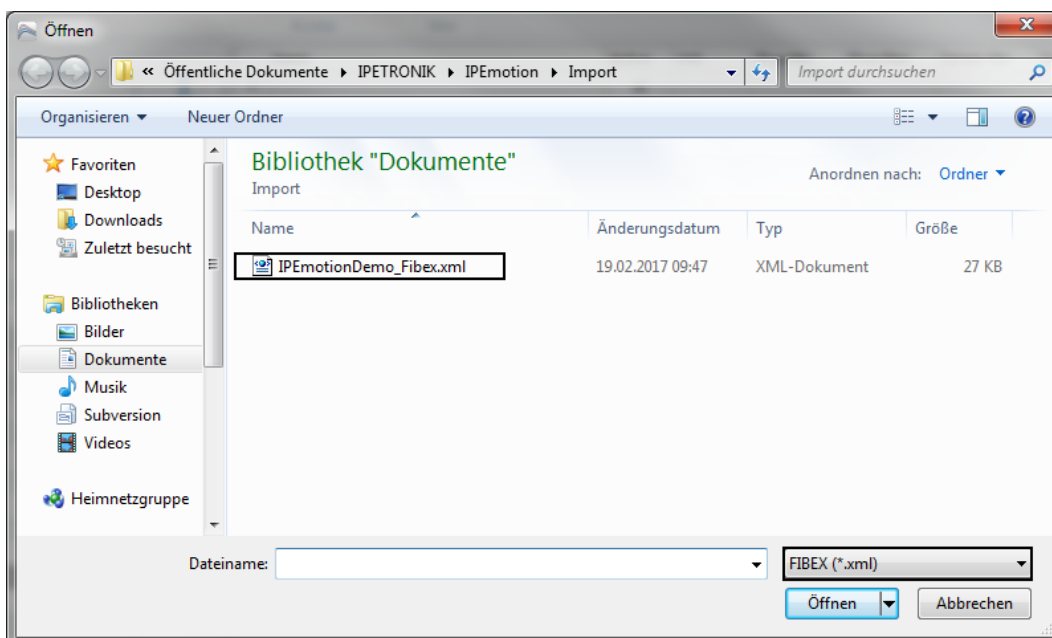
#### 8.13.2.1 Importing Autosar and Fibex files

The procedure for both filetypes is the same and will be exemplary explained in the following via the "Fibex" import.

To import Signals, select the FlexRay channel to which you wish to import your signal in the tree, click the "Import" button in the ribbon and then choose "Autosar" or "Fibex" for the import. For more information on the "description file" refer to (→[8.2.4.1](#)).

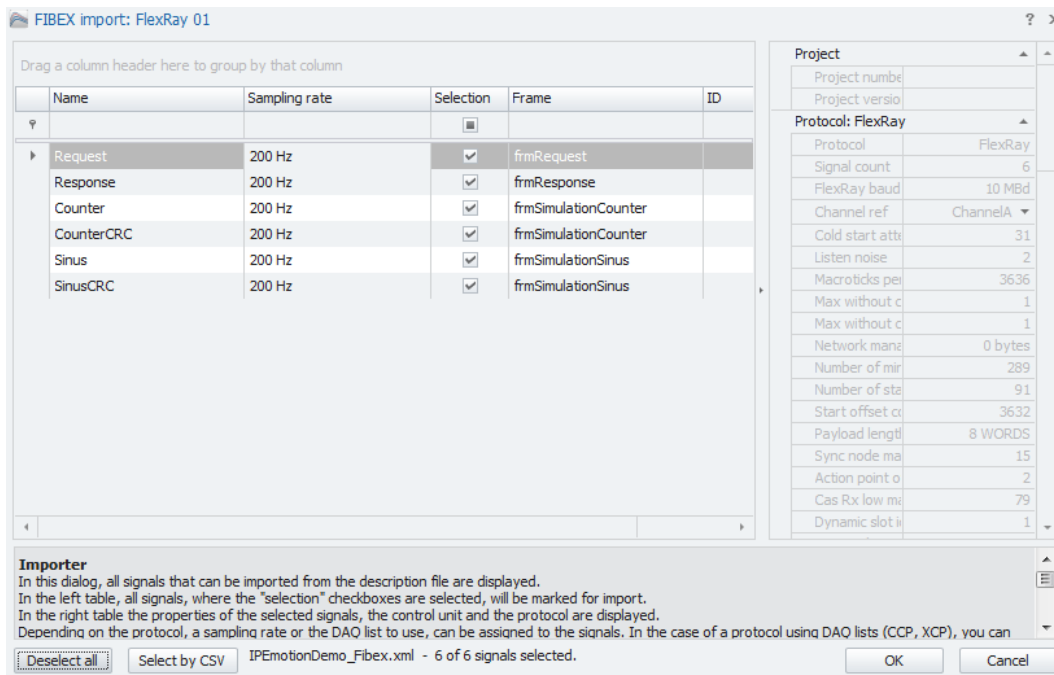


The following window lets you choose which file you wish to import. The dropdown menu on the bottom right of the window shows you, which filetypes are available. Choose the file you wish to import and click "Open".



Once you have opened your file, the "Importer" window will appear, that will present you with a range of importing options.

In this dialog, all signals that can be imported from the description file are displayed. In the left table, all signals, where the "selection" checkboxes are selected, will be marked for import. You can either choose manually, which signals to import, you can use the "Select/Deselect all" button on the bottom left, or you can use a CSV-file to determine which signals are to be imported, by clicking "Select by CSV" on the bottom left.



In the right table the properties of the selected signals, the control unit and the protocol are displayed.

Once you have chosen all the signals you wish to import, click "OK" to complete the import procedure.



Multiple description files can be imported into the same FlexRay channel.

### 8.13.2.2 Importing A2L files (XCP on FlexRay)

The Import of CCP/XCP databases via A2L files for FlexRay follows the same procedure as the CCP/XCP import via A2L file for CAN. Please refer to (→8.3.2).

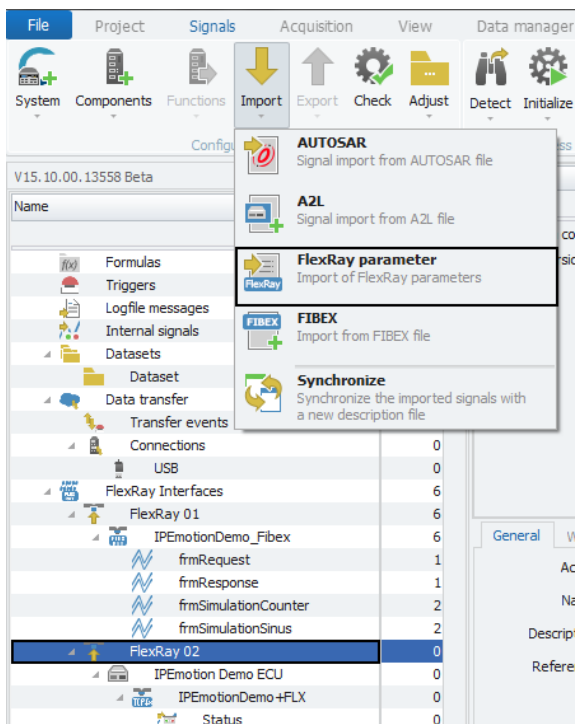


When importing signals from a CCP/XCP database via A2L file, you will need to additionally import the flexray parameters from a Fibex file as described below.

#### Importing FlexRay parameters

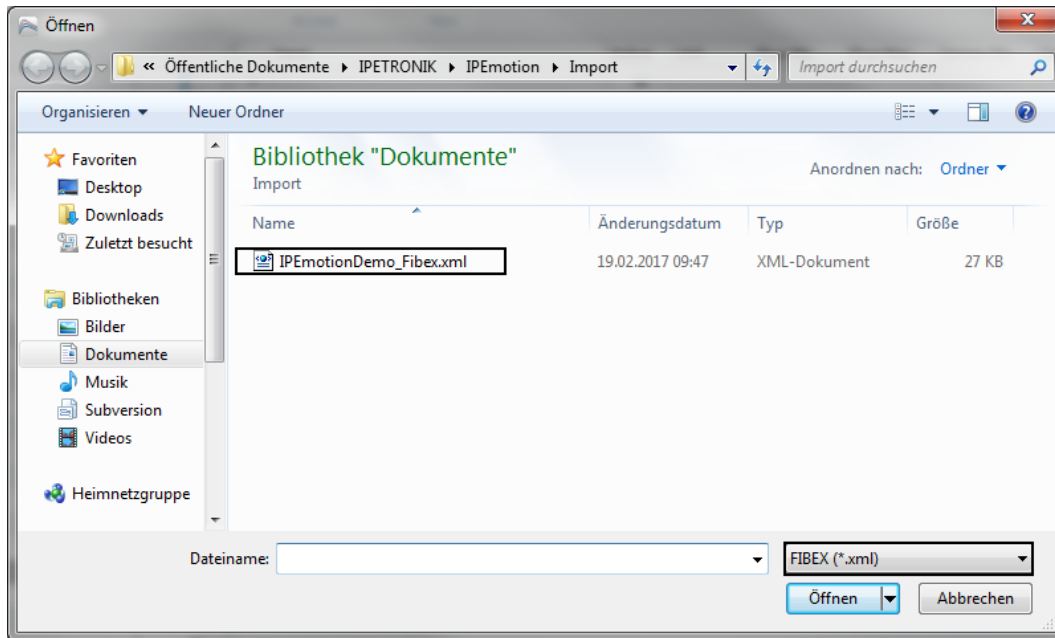
When creating a FlexRay XCP measurement task via an A2L file, it is necessary to import the FlexRay parameters via a Fibex file once the A2L file has been imported. Otherwise, the communication between ECU and FlexRay bus cannot be established.

To import FlexRay parameters, select the FlexRay channel to which you wish to import your signal in the tree, click the **Import** button in the ribbon and then choose **FlexRay parameters** for the import.

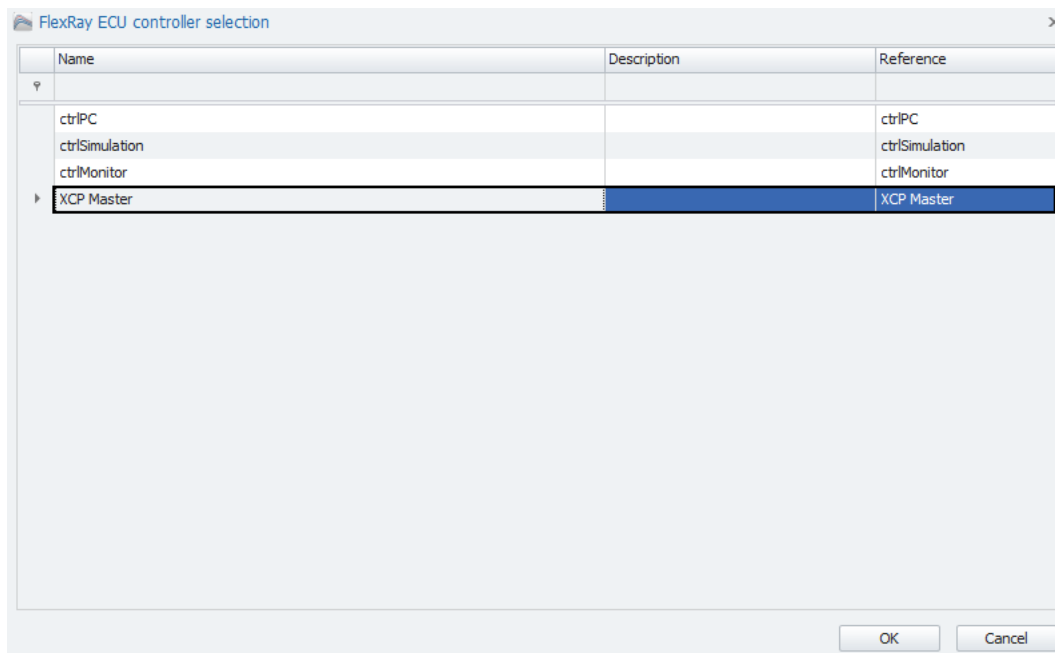




The following window lets you choose which Fibex file you wish to import. Choose the file you wish to import and click **“Open”**.

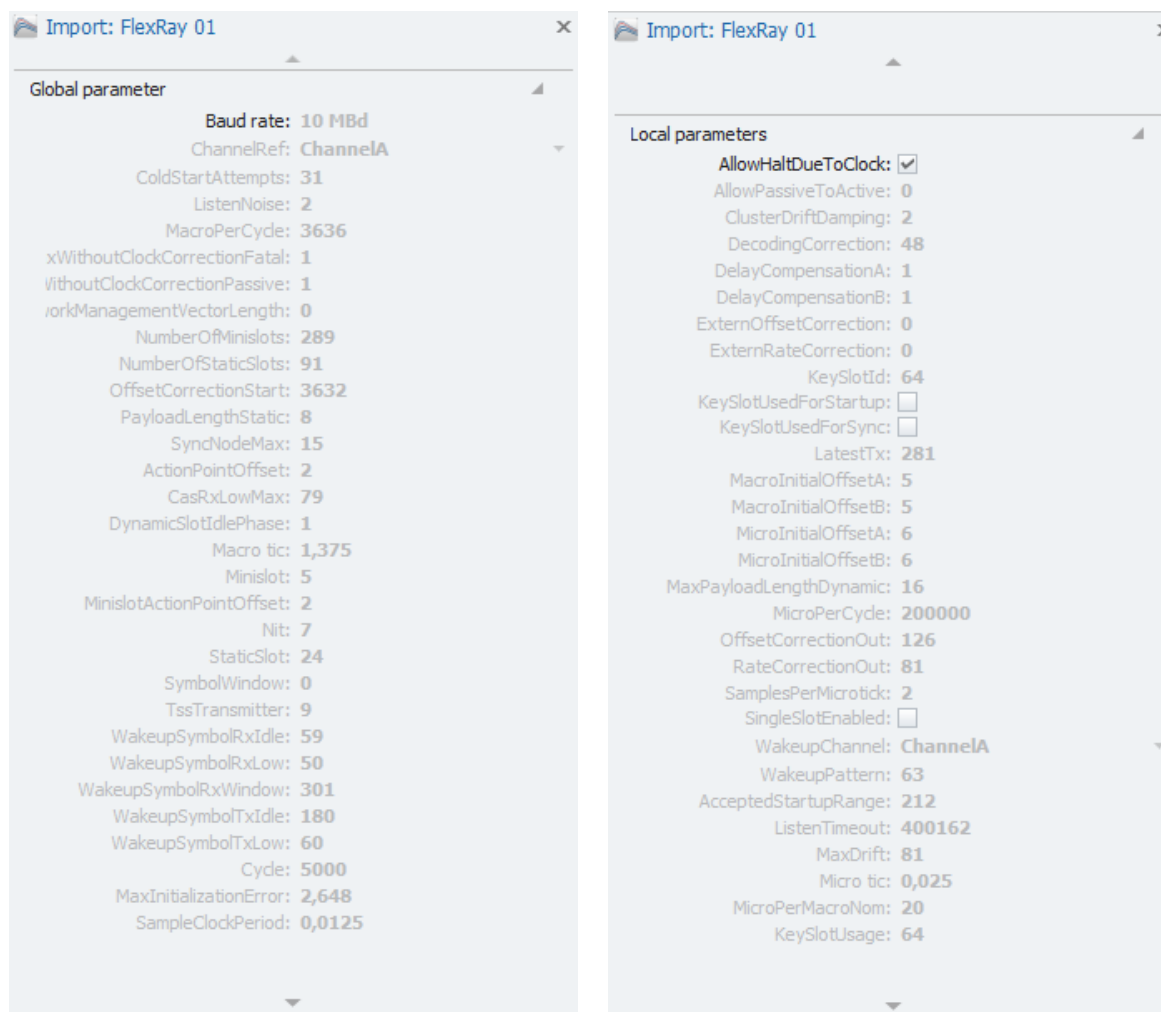


The following window lets you choose the ECU controller with which the FlexRay bus is supposed to communicate. Normally the right controller to choose is called **“XCP Master”**. Select your desired controller and confirm with **“OK”**.

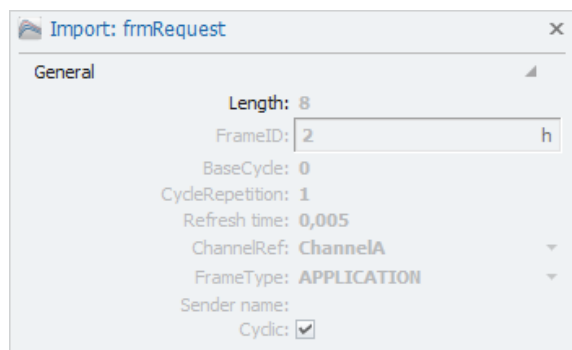


### 8.13.3 Import properties

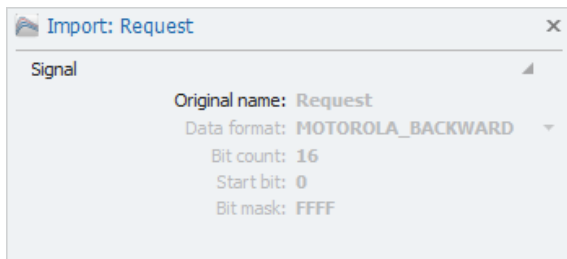
The “Import properties” of a description file, Message or signal allow you to see certain properties such as the Data format, the FlexRay identifier, the Bit mask, the start bit, bit count, optional commands and more. It shows the signal’s properties as described by the description file.



Example for “Import properties” of a FlexRay description file

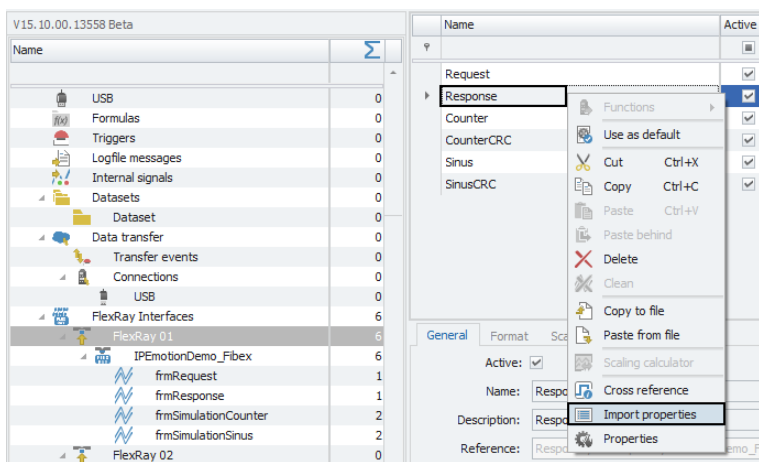


Example for “Import properties” of a FlexRay Message



Example for "Import properties" of a FlexRay signal

To access the "Import properties" rightclick on any desired description file, Message or signal and then choose "Import properties" from the resulting context menu.



## 8.13.4 Signal properties

### 8.13.4.1 Tree elements for FlexRay signals

The Tree elements available after import for your FlexRay channel depend on the Method you used for importing the signals.

If you imported an A2L file, the resulting tree elements will be equal to a CCP/XCP import on a CAN channel. Please refer to (→8.3.4.1).

If you imported a Fibex file to your FlexRay channel, this channel will contain two new layers of child elements in the measurement task tree: The **“Description file”** and the **“Message”**.

#### Description file

The “description file” is the database file which contains signal information and can be used to import those signals into a Signal channel in IPEmotion.

The tree element shows the “description file’s” name (in this case “IPEmotionDemo\_Fibex”) and, on the right, the number of signals it contains (in this case 6).

FlexRay Interfaces	6
FlexRay 01	6
IPEmotionDemo_Fibex	6
frmRequest	1
frmResponse	1
frmSimulationCounter	2
frmSimulationSinus	2
FlexRay 02	0
FlexRay 03	0
FlexRay 04	0

#### Message

Each “description file” can contain one or more “Messages”, which then contain the actual signals. A “Message” can be found in the “Measurement task tree” as a child element of the “description file”, it belongs to.

Each “Message” can, again, contain one or more signals, which is indicated by the number on the right of the “Message’s” name.

FlexRay Interfaces	6
FlexRay 01	6
IPEmotionDemo_Fibex	6
frmRequest	1
frmResponse	1
frmSimulationCounter	2
frmSimulationSinus	2
FlexRay 02	0
FlexRay 03	0
FlexRay 04	0

### 8.13.4.2 Grid area for FlexRay signals

In the “grid area” you will be presented with an overview of your selected FlexRay channel’s signals. Also you can find here two important functions, which are the “**Column chooser**” (→4.3.1) and the “**Filter editor**” (→4.3.2).

### 8.13.4.3 Details area for FlexRay signals

The Details area shows settings either for the selected tree element (“description file” or “Message”) or the selected signal in the grid area. In case a tree element is selected, the details area will only show the “General” tab. Please refer to (→4.2.2).

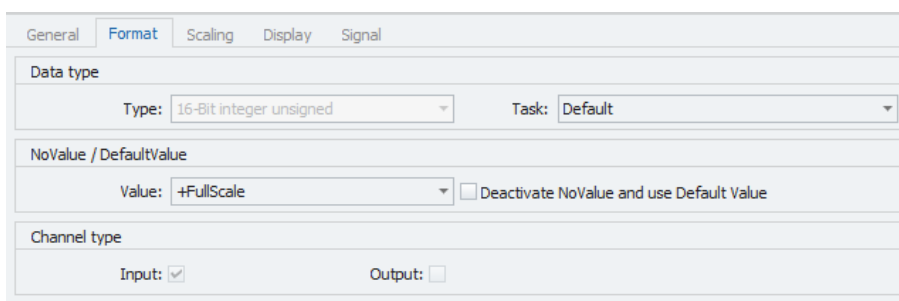
In case a signal is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

This tab allows you to activate or deactivate the entire signal by ticking/unticking the checkbox, give a user specific name to your signal if wished and add an additional description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed. The “Sampling rate” allows you to set, how frequently a signal should be requested.

#### Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case “**16-Bit integer unsigned**”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

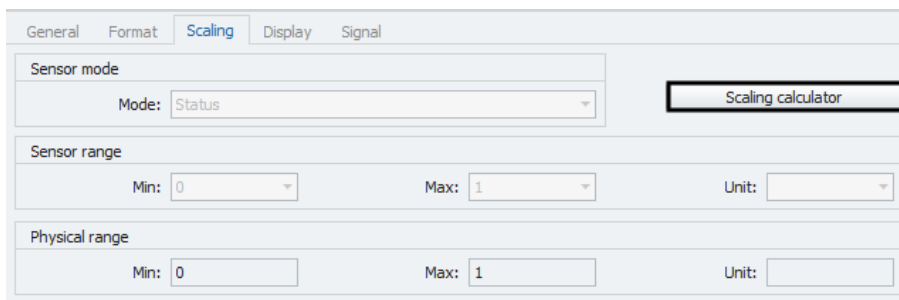
This field tells you whether you are dealing with a “**Input**” channel or “**Output**” channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.



In some cases it may be necessary to activate editing of protocol channel scaling in order to gain full access to the scaling functionality. For instructions please refer to the point “**Edit protocol channel scaling**” of the Expert settings (→ [3.2.2](#)).



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

The dropdown menu "**Decimal places**" allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

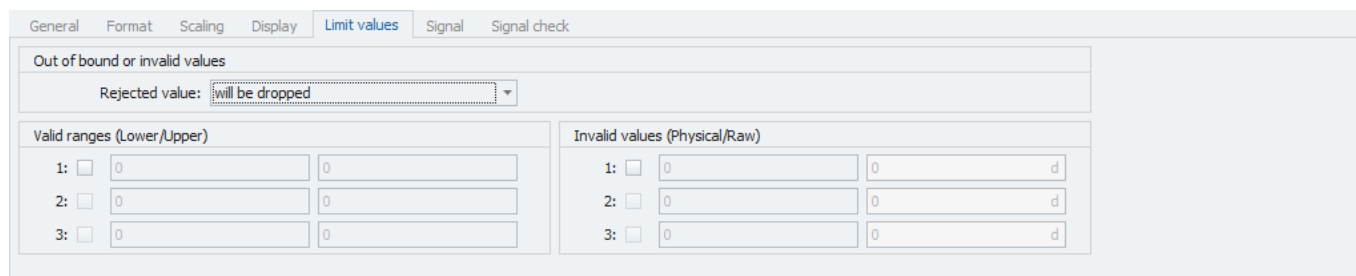
Allows you to set a Name to be shown on a display.



If the regular FlexRay namespace creates ambiguous names, it may be useful, to work with extended FlexRay namespaces. For instructions please refer to (→[3.2.2](#)).

## Limit values

This tab allows you to define limit values for a signal and what action to take upon a limit value violation.



- **Rejected value**

Define what happens to a value, that has been rejected because it is out of bound or invalid. By default this value will be dropped. It can also be written as NaN.

- **Valid ranges (Lower/Upper)**

Define up to three ranges of valid signal values. Activate a range in order to define its upper/lower value (datatype double). Range 2 can only be activated if range 1 is and range 3 can only be activated if range 2 is.

- **Invalid values (Physical/Raw)**

Define up to three invalid values. Activate an invalid value in order to define the physical value (datatype double) or raw value (datatype integer). If one of the two has been typed in, the other will be calculated according to the scale/offset settings in the scaling calculator.

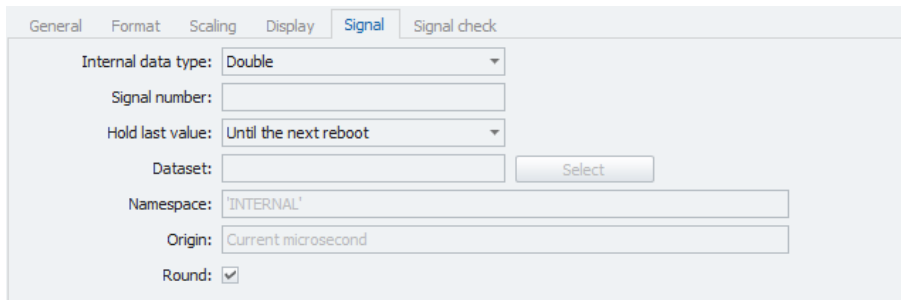
Invalid value 2 can only be activated if invalid value 1 is and invalid value 3 can only be activated if invalid value 2 is.

For details on how to use the "Scaling calculator" please refer to the [IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling"](#).



## Signal

This tab allows you to define signal settings.



The screenshot shows a software interface with several tabs: General, Format, Scaling, Display, Signal (selected), and Signal check. The Signal tab contains the following settings:

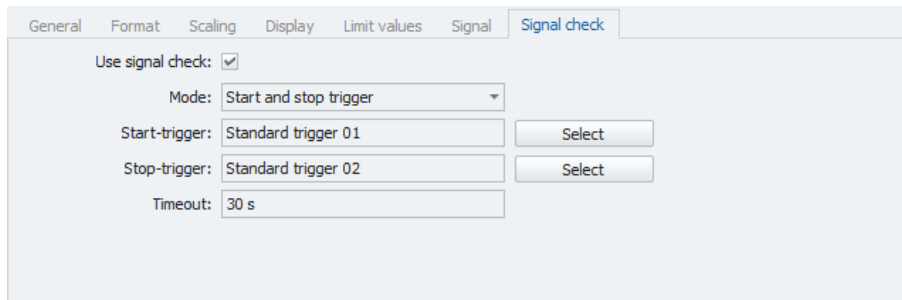
- Internal data type: Double
- Signal number: (empty text box)
- Hold last value: Until the next reboot
- Dataset: (empty text box) with a Select button
- Namespace: INTERNAL
- Origin: Current microsecond
- Round:

- Internal data type**  
 Assign an internal data type to the signal. Available data types are “Double” and “String”.
- Signal number**  
 Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.
- Hold last value**  
 Specify, for how long the last value of the signal will be hold.
- Dataset**  
 If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.
- Timeout**  
 Specify the timeout period for the current signal. If the data source doesn’t send data for the specified time period, the value of the signal is set to “NaN (Not a Number)” and will be displayed as “-” in a display.
- Namespace**  
 The “Namespace” serves as unique identifier for the signal inside the logger.

## Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



The screenshot shows a software interface with a tabbed menu at the top: General, Format, Scaling, Display, Limit values, Signal, and Signal check. The 'Signal check' tab is active. Below the tabs, there is a section for configuring signal check parameters:

- Use signal check:**
- Mode:** Start and stop trigger (dropdown menu)
- Start-trigger:** Standard trigger 01 (text input) with a **Select** button
- Stop-trigger:** Standard trigger 02 (text input) with a **Select** button
- Timeout:** 30 s (text input)

## 8.14 GPS Signals

The GPS module sends a constant stream of values to the logger. Its configuration defines which values from this data stream are to be evaluated and made available for further use.

### 8.14.1 Storage method

In order to store incoming GPS signals you can use the GPX storage method. These signals will then be stored in a separate file only containing your GPS signals.

- GPX (→ [14.12](#))

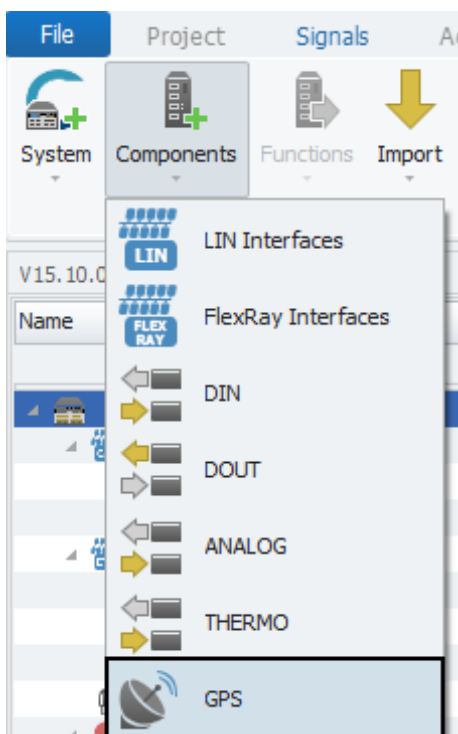
You can also store your GPS signals together with signals from other buses. To do so use one of the following signal storage methods.

- ATFX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

### 8.14.2 Adding GPS Signals

#### 8.14.2.1 CAETEC GPS module

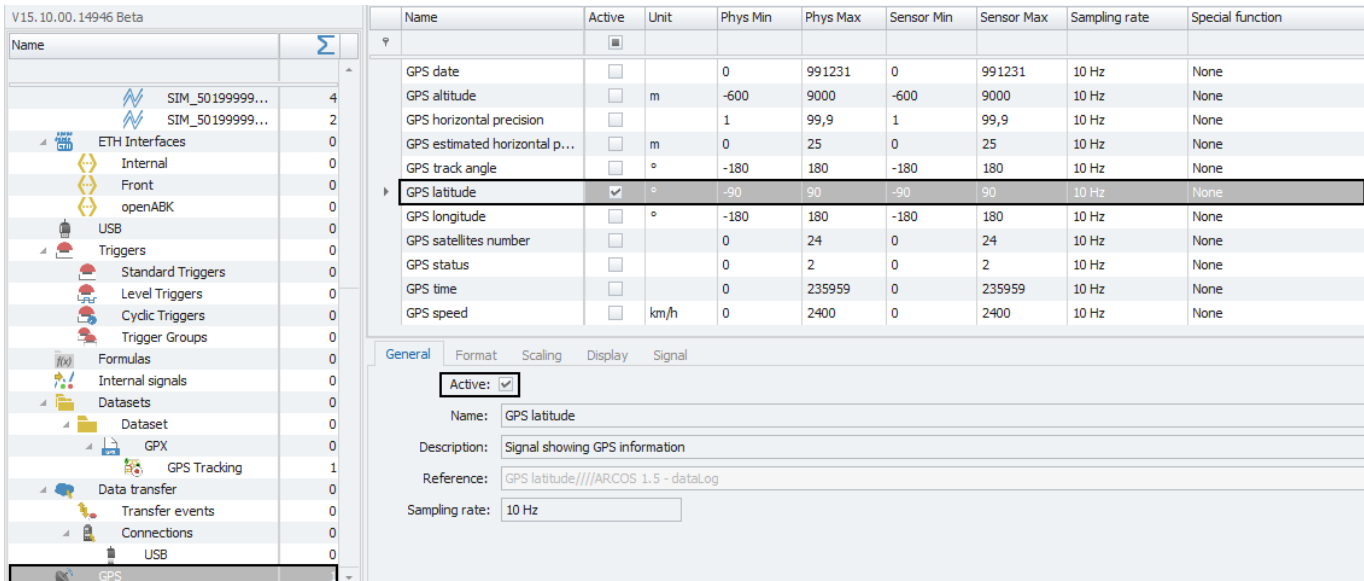
In order to use the CAETEC GPS module, select your system (Arcos 1.x,  $\mu$ cross) in the “Measurement task tree”, click the “Components” button in the Ribbon and choose “GPS”.



### 8.14.2.2 Other GPS signals (Assigning GPS signals)

If you are not using the CAETEC GPS module, but are receiving GPS signals on your logger GPS tasks have to be manually assigned to the respective signals in order to store those signals in a GPX file.

To assign a GPS task to a signal, select the desired signal in the “Grid area” and then activate it by ticking the “Active” box in the signal’s “Details area”.



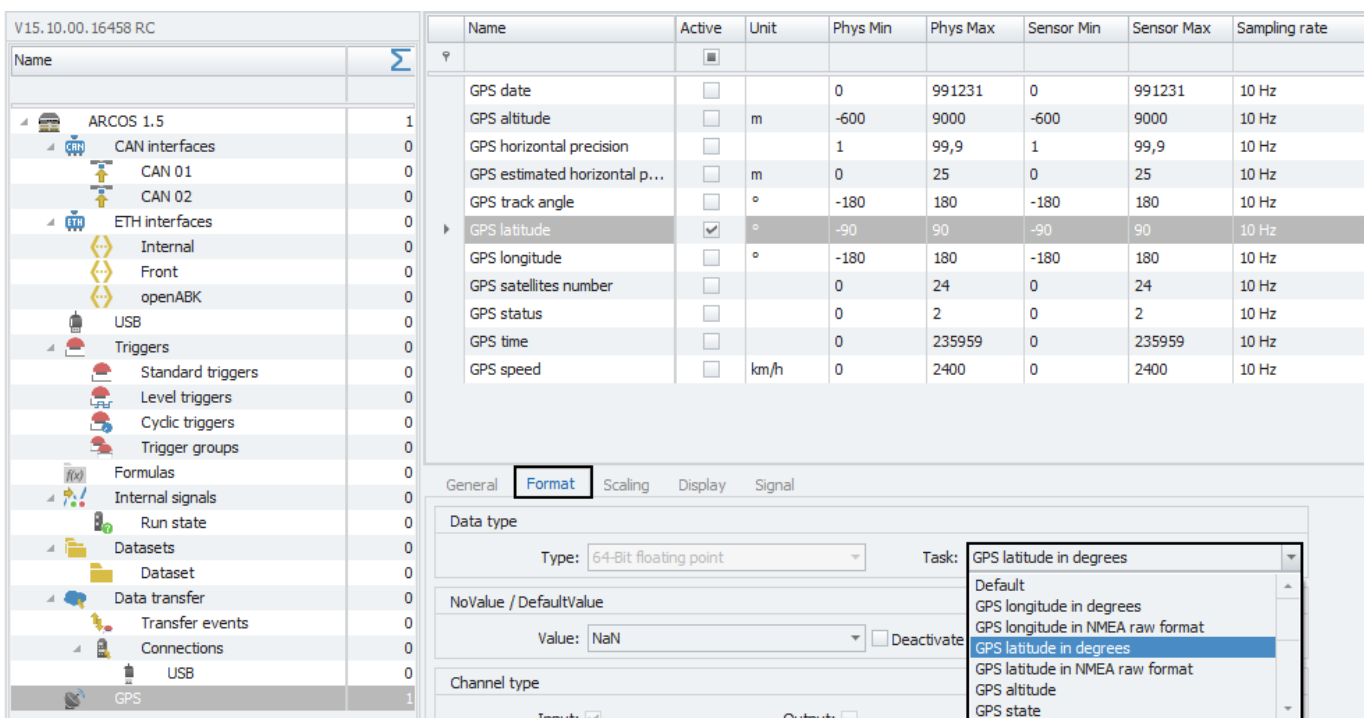
The screenshot shows the software interface with a tree view on the left and a signal configuration table on the right. The 'GPS latitude' signal is selected and its details are shown in the 'Details area'.

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate	Special function
GPS date	<input type="checkbox"/>		0	991231	0	991231	10 Hz	None
GPS altitude	<input type="checkbox"/>	m	-600	9000	-600	9000	10 Hz	None
GPS horizontal precision	<input type="checkbox"/>		1	99,9	1	99,9	10 Hz	None
GPS estimated horizontal p...	<input type="checkbox"/>	m	0	25	0	25	10 Hz	None
GPS track angle	<input type="checkbox"/>	°	-180	180	-180	180	10 Hz	None
<b>GPS latitude</b>	<input checked="" type="checkbox"/>	°	-90	90	-90	90	10 Hz	None
GPS longitude	<input type="checkbox"/>	°	-180	180	-180	180	10 Hz	None
GPS satellites number	<input type="checkbox"/>		0	24	0	24	10 Hz	None
GPS status	<input type="checkbox"/>		0	2	0	2	10 Hz	None
GPS time	<input type="checkbox"/>		0	235959	0	235959	10 Hz	None
GPS speed	<input type="checkbox"/>	km/h	0	2400	0	2400	10 Hz	None

The 'Details area' for 'GPS latitude' shows:

- Active:
- Name: GPS latitude
- Description: Signal showing GPS information
- Reference: GPS latitude:///ARCOS 1.5 - dataLog
- Sampling rate: 10 Hz

Then navigate to the “Format” tab in the signal’s “Details area” and use the dropdown menu “Tasks” to assign the desired GPS task.



The screenshot shows the 'Format' tab for the 'GPS latitude' signal. The 'Task' dropdown menu is open, showing the following options:

- GPS latitude in degrees
- Default
- GPS longitude in degrees
- GPS longitude in NMEA raw format
- GPS latitude in degrees
- GPS latitude in NMEA raw format
- GPS altitude
- GPS state

Each GPS task can only be assigned to one signal. That means, if you reassign a previously already assigned GPS task to a new signal, make sure to unassign this task from its previously assigned signal.

Using the “Check” function will tell you, whether you have multiply assigned tasks.

### 8.14.3 Signal properties

#### 8.14.3.1 Tree elements for GPS signals

There is one tree element for GPS signals, called “GPS”. In the right table of the “Measurement task tree” a number will indicate how many active signals it contains (in this case 2).



#### 8.14.3.2 Grid area for GPS signals

In the “Grid area” you will be presented with an overview of the available GPS signals. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

*Overview of GPS signals*

Subtype	Meaning	Unit
GPS date	GPS date UTC yymmdd	-
GPS altitude	Elevation above sea level	(m)
GPS horizontal precision	Horizontal Dilution of Precision	-
GPS estimated horizontal precision	Estimation of horizontal dilution of precision (probability 95%)	(m)
GPS track angle	Inclination of the track	(°)
GPS latitude	Latitude	(°)
GPS longitude	Longitude	(°)
GPS satellites number	Number of received satellites	-
GPS status	0 = no connection 1 = connection 2 = Egnos active	-
GPS time	GPS time UTC hhmmss	-
GPS speed	Current speed	(km/h)

#### 8.14.3.3 Details area for GPS signals

The Details area shows settings either for the tree element “GPS” or a selected signal in the grid area. In case the tree element “GPS” is selected, the details area will only show the “General” tab. Please refer to (→4.2.2).

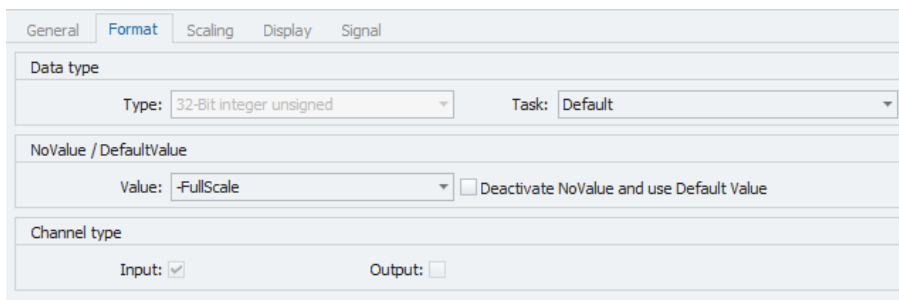
In case a signal is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

## General

This tab allows you to activate or deactivate the entire signal by ticking/unticking the checkbox, give a user specific name to your signal if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The "Sampling rate" allows you to set, how frequently a signal should be requested.

## Format

This tab contains information and options regarding file format, tasks and Channel type.



The screenshot shows the 'Format' tab of a software interface. It contains three main sections:

- Data type:** A dropdown menu for 'Type' is set to '32-Bit integer unsigned', and a dropdown menu for 'Task' is set to 'Default'.
- NoValue / DefaultValue:** A dropdown menu for 'Value' is set to '-FullScale', and there is an unchecked checkbox labeled 'Deactivate NoValue and use Default Value'.
- Channel type:** There are two checkboxes: 'Input' is checked, and 'Output' is unchecked.

- **Data type**

This field tells you the type of data (in this case "32-Bit integer unsigned") and allows you to apply special tasks for this signal such as "GPS Longitude", "GPS Latitude", "UTC hour", "Audio mono" and more.

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

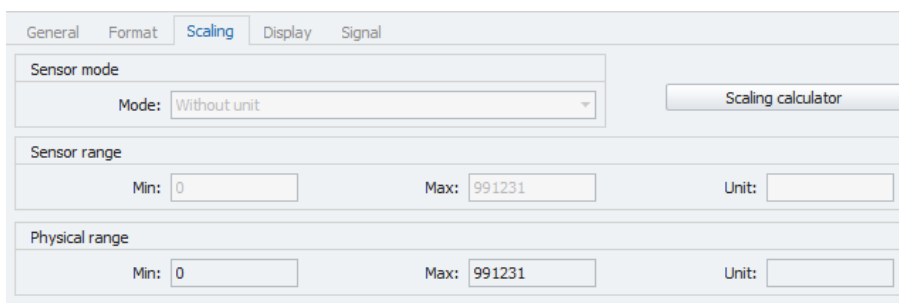
This field tells you whether you are dealing with a "Input" channel or "Output" channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.



In some cases it may be necessary to activate editing of protocol channel scaling in order to gain full access to the scaling functionality. For instructions please refer to the point “**Edit protocol channel scaling**” of the Expert settings (→ [3.2.2](#)).



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

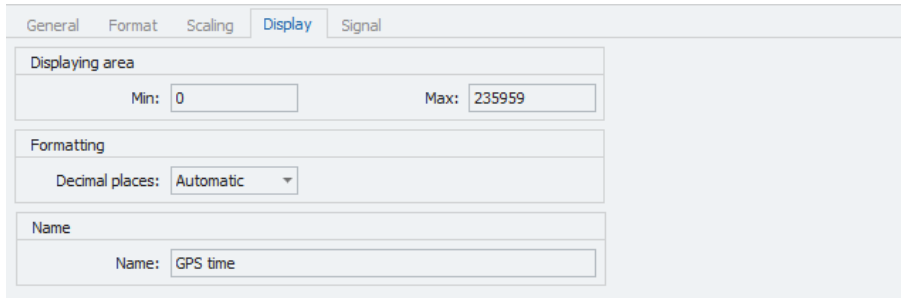
Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

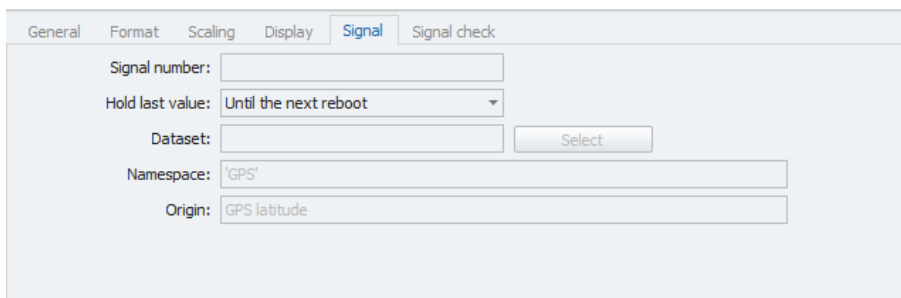
The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Signal

This tab allows you to define signal settings.



- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

Changes and errors excepted.



- **Dataset**

If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.

- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.

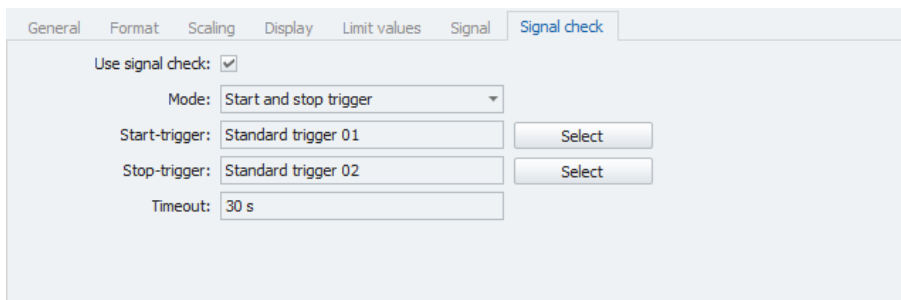
- **Origin**

Tells the source of the signal. This can help identifying the source of a signal for which a user defined signal name has been set.

### Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



The screenshot shows a software interface with a tabbed menu at the top: General, Format, Scaling, Display, Limit values, Signal, and Signal check. The 'Signal check' tab is active. Below the tabs, there is a section for configuring signal check settings. It includes a checkbox labeled 'Use signal check:' which is checked. Below this is a dropdown menu for 'Mode:' set to 'Start and stop trigger'. There are two rows for triggers: 'Start-trigger:' with a text box containing 'Standard trigger 01' and a 'Select' button; and 'Stop-trigger:' with a text box containing 'Standard trigger 02' and a 'Select' button. At the bottom, there is a 'Timeout:' field with a text box containing '30 s'.

## 8.15 Video devices

In order to work with video signals, you will first need to add a video device to your system. There are three different video devices, that you can work with:

- Video Interface (→[8.15.2](#))
- USB camera (→[8.15.3](#))
- Ethernet camera (→[8.15.4](#))

### 8.15.1 Storage method

In order to store an incoming signal on a video device use "AVI" as storage method. Please refer to (→ [14.10](#)).

### 8.15.2 Video Interface

The "Video Interface" provides a set of four analog cameras, which are connected to your logger. It provides five signals, one for each camera and a combination of all four camera signals called "Quad camera". The "Quad camera" provides a single video signal which contains all four original signals at a quarter of their original resolution, and presents a 4 in 1 picture.

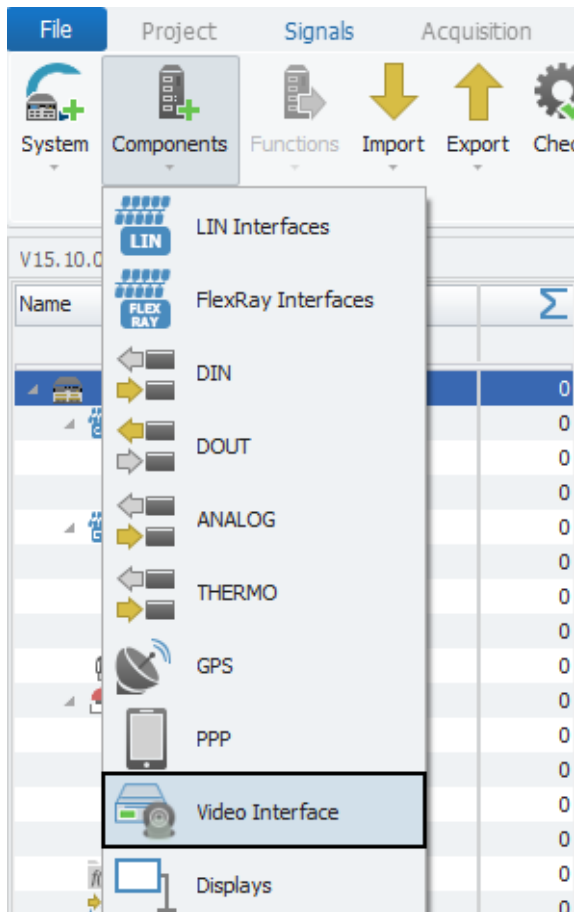
For communication between the cameras and the interface it needs to have its own subnet, in which only the "Video Interface" operates.



The "Video Interface" is a deprecated product and is not being sold anymore.

### 8.15.2.1 Adding the Video Interface

The “Video Interface” can be added by selecting the system in the “Measurement task tree”, then clicking the “Components” button and finally choosing “Video Interface”



### 8.15.2.2 Tree elements for the Video Interface

Adding the “Video Interface” to your system will add seven new elements to your “Measurement task tree”:

- **Video Interface xx**  
This item represents the entire “Video Interface” and all the included child elements.
- **Cameras**  
This item is a child element of the “Video Interfaces” element and represents all the included cameras.
- **Camera xx**  
There are four elements named “Camera xx”, they represent the four physical cameras connected to the logger.
- **Quad Camera**  
The “Quad Camera” is a virtual camera, that combines the four signals of the four

physical cameras in one single image. The image is divided into four quarters. Each quarter contains one of the physical cameras images.

Video Interface 01	5
Cameras	5
Camera 01	1
Camera 02	1
Camera 03	1
Camera 04	1
Quad Camera	1

### 8.15.2.3 Grid area for the Video Interface

In the “grid area” you will be presented with an overview of the “Video Interface’s” signals. There will be five signals. **Signal five is the signal from the “Quad Camera”**. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

The screenshot shows the software interface with a tree view on the left and a table of video signals on the right. The tree view shows the following structure:

- openABK: 0
- USB: 1
- Camera: 1
  - USB Video: 1
- Triggers: 0
  - Standard Triggers: 0
  - Level Triggers: 0
  - Cyclic Triggers: 0
  - Trigger Groups: 0
- Formulas: 0
- Internal signals: 0
- Datasets: 0
  - Dataset: 0
- Data transfer: 0
  - Transfer events: 0
- Connections: 0
  - USB: 0
- Logfile messages: 0
- Video Interface 01: 5
  - Cameras: 5
    - Camera 01: 1
    - Camera 02: 1
    - Camera 03: 1
    - Camera 04: 1
    - Quad Camera: 1

The table on the right lists the video signals:

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
Video signal 01	<input checked="" type="checkbox"/>						1 Hz
Video signal 02	<input checked="" type="checkbox"/>						1 Hz
Video signal 03	<input checked="" type="checkbox"/>						1 Hz
Video signal 04	<input checked="" type="checkbox"/>						1 Hz
Video signal 05	<input checked="" type="checkbox"/>						1 Hz

### 8.15.2.4 Details area for the Video Interface

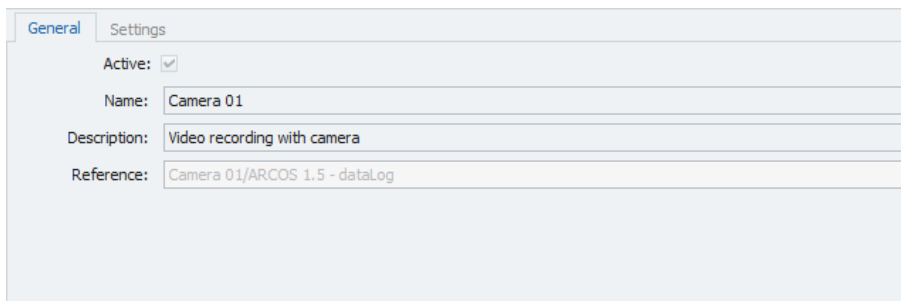
The Details area shows settings either for the tree element “**Video Interface xx**” or one of its child elements. In case the tree element “Cameras” is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

The “Details area” for video signal will be handled in a separate chapter. Please refer to the Chapter “Video signals” (→[8.16](#)).

In case the “Video Interface” or one of the “Camera” elements is selected in the tree, the “Details area” will contain additional tabs, which will be explained in the following.

#### General

This tab allows you to activate or deactivate the entire tree element by ticking/unticking the checkbox, give a user specific name to tree element if wished and add an additional description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

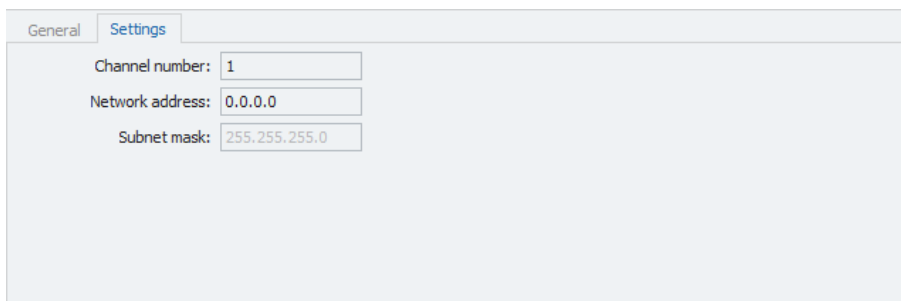


The screenshot shows the 'General' tab of a settings window. It contains the following fields:

- Active:** A checked checkbox.
- Name:** A text input field containing 'Camera 01'.
- Description:** A text input field containing 'Video recording with camera'.
- Reference:** A text input field containing 'Camera 01/ARCOS 1.5 - dataLog'.

#### Settings (for Video Interface)

This tab allows you to define the settings for the entire “Video Interface”.



The screenshot shows the 'Settings' tab of a settings window. It contains the following fields:

- Channel number:** A text input field containing '1'.
- Network address:** A text input field containing '0.0.0.0'.
- Subnet mask:** A text input field containing '255.255.255.0'.

- **Channel number**

Define the number of the “Video Interface” hardware channel.



For better orientation and in order to avoid confusion regarding Channel numbers and -names, a Channels physical number can be found in the logger's "**Web Interface**" and set accordingly.

- **Network address**

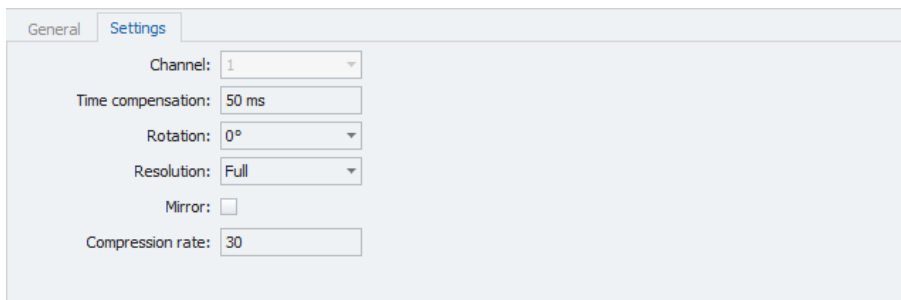
Define the network address of the subnet in which your "Video Interface" communicates.

- **Subnet mask**

The subnet mask for your "Video Interface". It cannot be changed.

### Settings (for Camera elements)

This tab allows you to define the settings for the entire "Camera" child elements of your "Video Interface".



- **Channel**

The physical channel, on which the camera operates. It cannot be changed.

- **Time compensation**

Define the compensation of the video capture latency. This value is experience based and can vary.

- **Rotation**

Allows you to rotate the video in steps of 90°.

- **Resolution**

Allows you to set the video's resolution to either full or one quarter.

- **Mirror**  
Checking the box will mirror the image of the video.
- **Compression rate**  
Allows you to define the compression rate of your video.

### 8.15.3 USB camera

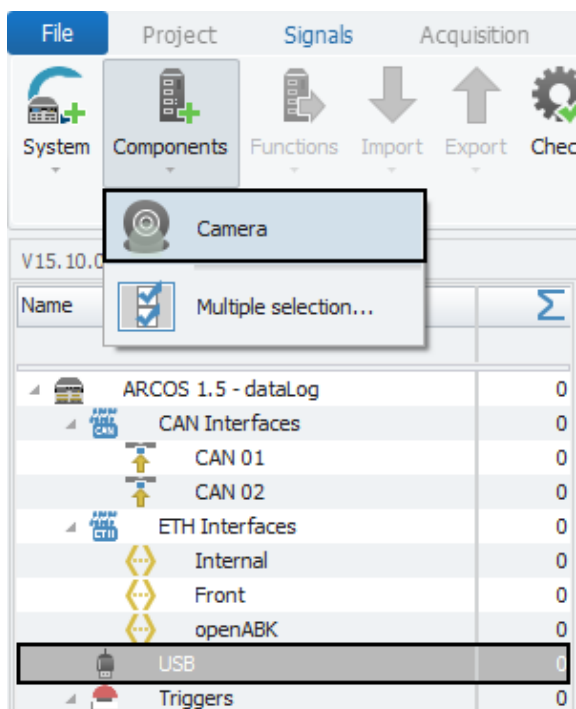
The "Camera" component for USB allows you to connect a digital video camera via USB to your logger and control it.



It is possible to add multiple USB cameras to your logger.

#### 8.15.3.1 Adding a USB camera

The "Camera" component for USB can be added by selecting the tree element "USB" in the "Measurement task tree", then clicking the "Components" button and finally choosing "Camera"



### 8.15.3.2 Tree elements for USB camera

Adding the “Camera” component for USB to your system will add two new elements to your “Measurement task tree”:

USB	1
Camera	1
USB Video	1

- **Camera**  
This item represents the camera itself.
- **USB Video**  
This item represents the video, that you will receive from the camera.

### 8.15.3.3 Grid area for USB camera

In the “grid area” you will see the video signal coming from your connected USB camera.

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
Video signal 01	<input checked="" type="checkbox"/>						1 Hz

### 8.15.3.4 Details area for USB camera

The Details area shows settings either for the tree element “**Camera**” or its child element “USB Video”.



Video signal settings will be handled in a separate chapter. Please refer to the Chapter “Video signals” (→[8.16](#)).



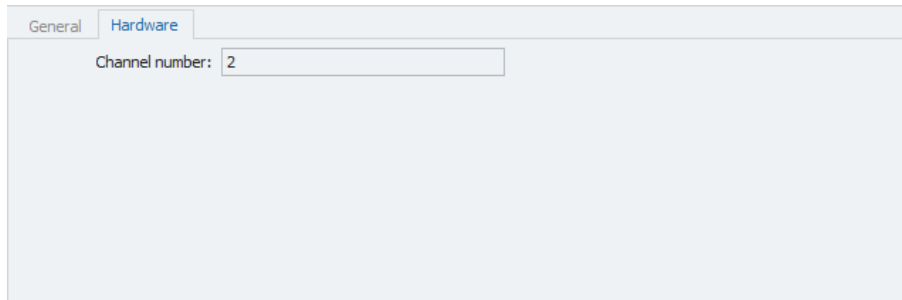
## Settings for “Camera”

### General

“General” tab. Please refer to (→[4.2.2](#)).

### Hardware

Set the digital hardware channel number, that corresponds with the camera’s physical hardware channel number. To find out the corresponding hardware channel numbers please consult your loggers webinterface.

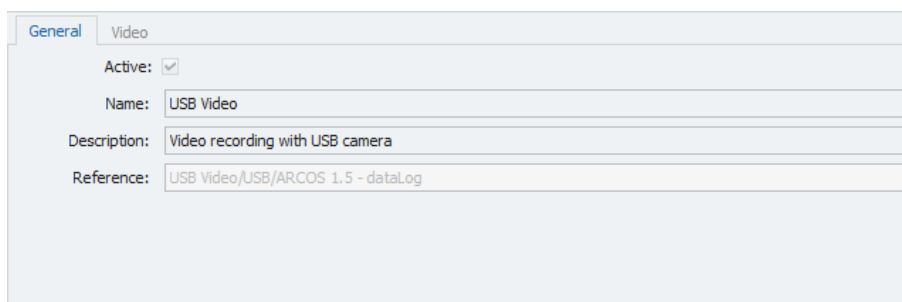


The screenshot shows the 'Hardware' tab of the 'Camera' settings. It features a 'Channel number' label followed by a text input field containing the value '2'.

## Settings for “USB Video”

### General

This tab allows you to activate or deactivate the entire tree element by ticking/unticking the checkbox, give a user specific name to tree element if wished and add an additional description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

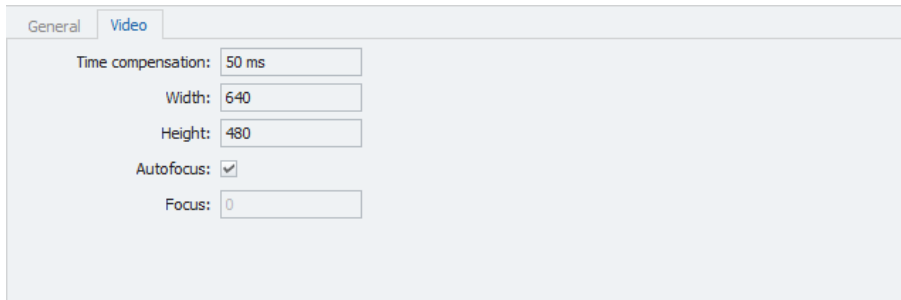


The screenshot shows the 'General' tab of the 'USB Video' settings. It includes the following fields:

- Active:** A checked checkbox.
- Name:** A text input field containing 'USB Video'.
- Description:** A text input field containing 'Video recording with USB camera'.
- Reference:** A text input field containing 'USB Video/USB/ARCOS 1.5 - dataLog'.

## Video

This tab allows you to define the settings for the incoming video.



The screenshot shows a software interface with two tabs: 'General' and 'Video'. The 'Video' tab is active. It contains the following settings:

- Time compensation: 50 ms
- Width: 640
- Height: 480
- Autofocus:
- Focus: 0

- **Time compensation**

Define the compensation of the video capture latency. This value is experience based and can vary.

- **Width**

Define the video capture resolution width in pixel.

- **Height**

Define the video capture resolution height in pixel.

- **Autofocus**

Allows you to enable or disable the the camera's autofocus.

- **Focus**

When "Autofocus" has been disabled, this field allow you to set the focus of the camera.

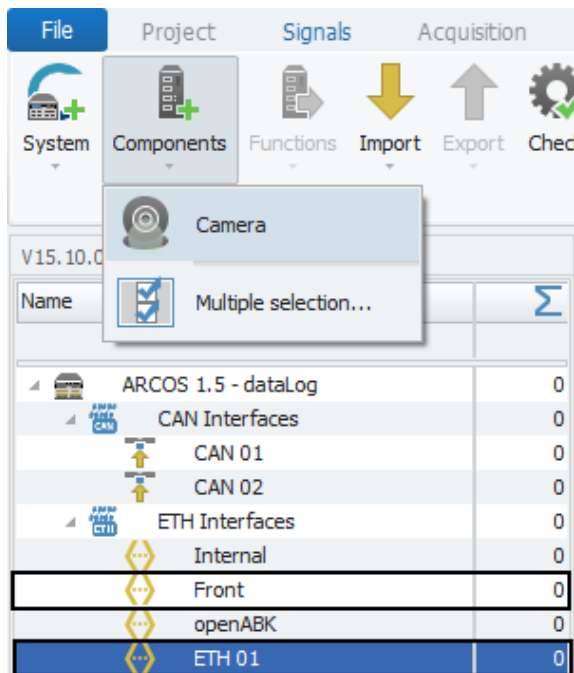
APPROXIMATE and CALIBRATED devices report the focus metadata in units of diopters (1/meter), so 0.0f represents focusing at infinity, and increasing positive numbers represent focusing closer and closer to the camera device. The focus distance control also uses diopters on these devices.

## 8.15.4 Ethernet camera

The “Camera” component for Ethernet allows you to connect a digital video camera via Ethernet to your logger and control it.

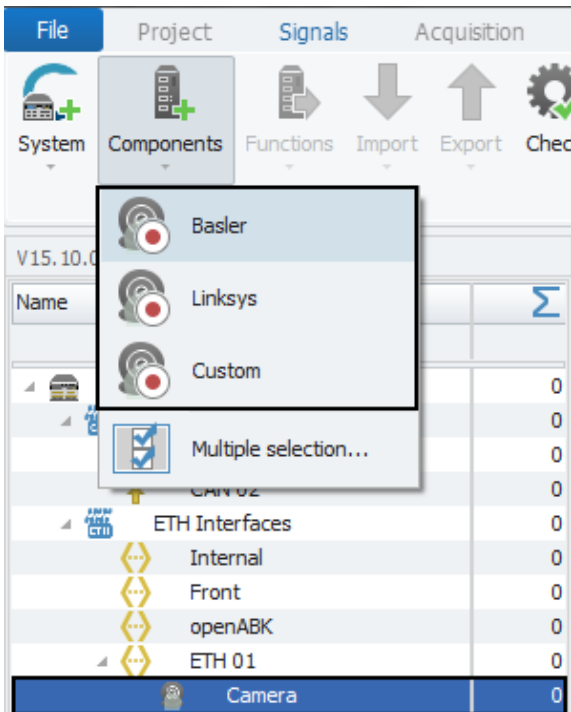
### 8.15.4.1 Adding an ETH camera

The “Camera” component for ETH can be added by selecting the tree element “Front”, which is a child element to the tree element “ETH Interfaces” in the “Measurement task tree” or by selecting any manually added “ETH” channel, then clicking the “Components” button and finally choosing “Camera”.



In the next step you select the newly created tree element "Camera", then click the "Components" button and finally choose the desired camera model.

At the moment the Plugin support two third party cameras, "Basler" and "Linksys". If your camera is not listed, you can simply choose "Custom" and then set the necessary setting manually.





Once a new camera for Ethernet has been added, you may encounter problems regarding the cameras IP settings.

In order to resolve this problem, DHCP has to be disabled for the tree element "Camera" and IP addresses have to be set for both, the tree element "Camera" and its child element, the cameratype you have previously chosen. The IP addresses must belong to the same subnet, and this subnet must not be used by any other operator of the system.

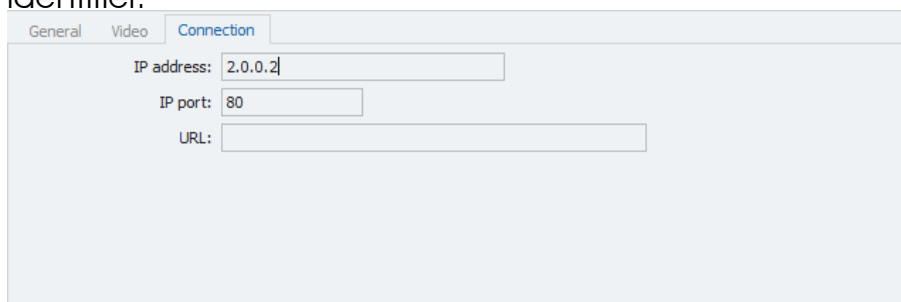
To do so, select the tree element that contains your ETH camera navigate to the "LAN" tab in the details area, untick the checkbox "Get IP address automatically" and set a new IP address.

The first three numbers of your IP address mark the subnet, so they cannot be equal to any other operators IP address, that is not a child-element to the currently selected tree element. The last number marks the client inside the subnet. It has to be higher equal or higher than "1" and unique inside its respective subnet.



General	LAN	Settings
Get IP address automatically: <input type="checkbox"/>		
IP address: <input type="text" value="2.0.0.1"/>		
Subnet mask: <input type="text" value="255.255.255.0"/>		
Standard gateway: <input type="text" value="0.0.0.0"/>		
DNS server: <input type="text" value="0.0.0.0"/>		

Then select your chosen cameratype in the tree (Basler, Linksys or Custom) and navigate to the "Connection" tab in the details area. Here you need to set an IP address that belongs to the same subnet as the one defined in the last step, but, again, with a unique client identifier.



General	Video	Connection
IP address: <input type="text" value="2.0.0.2"/>		
IP port: <input type="text" value="80"/>		
URL: <input type="text"/>		

### 8.15.4.2 Tree elements for ETH camera

Adding the “Camera” component for ETH to your system will add two new elements to your “Measurement task tree”.

- **Camera**  
This item represents the camera interface.
- **Basler/Linksys/Custom**  
This item represents the specific type of camera, that you have connected are configuring.

ETH Interfaces	1
Internal	0
Front	0
openABK	0
ETH 01	1
Camera	1
Basler	1

### 8.15.4.3 Grid area for ETH camera

In the “grid area” you will see the video signal coming from your connected ETH camera.

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
Video signal 01	<input checked="" type="checkbox"/>						1 Hz

### 8.15.4.4 Details area for ETH camera

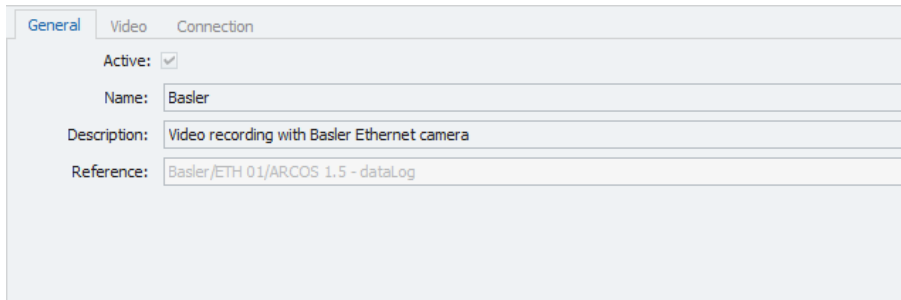
The Details area shows settings either for the tree element “**Camera**” or its child element “Basler/Linksys/Custom”. In case the tree element “Camera” is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

The “Details area” for video signals will be handled in a separate chapter. Please refer to the Chapter “Video signals” (→[8.16](#)).

In case the child element “Basler/Linksys/Custom” is selected in the tree, the “Details area” will contain additional tabs, which will be explained in the following.

## General

This tab allows you to activate or deactivate the entire tree element by ticking/unticking the checkbox, give a user specific name to tree element if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

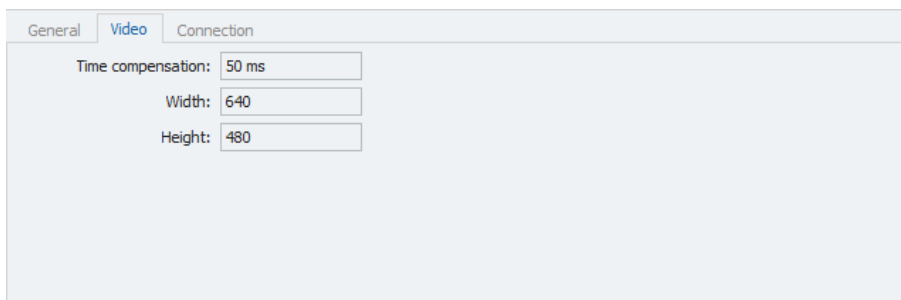


The screenshot shows the 'General' tab of a configuration window. It contains the following fields:

- Active:** A checked checkbox.
- Name:** A text input field containing 'Basler'.
- Description:** A text input field containing 'Video recording with Basler Ethernet camera'.
- Reference:** A text input field containing 'Basler/ETH 01/ARCOS 1.5 - dataLog'.

## Video

This tab allows you to define the settings for the incoming video.



The screenshot shows the 'Video' tab of a configuration window. It contains the following fields:

- Time compensation:** A text input field containing '50 ms'.
- Width:** A text input field containing '640'.
- Height:** A text input field containing '480'.

- **Time compensation**

Define the compensation of the video capture latency. This value is experience based and can vary.

- **Width**

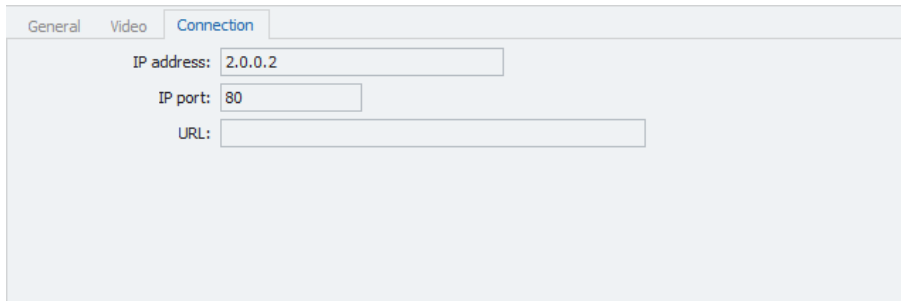
Define the video capture resolution width in pixel.

- **Height**

Define the video capture resolution height in pixel.

## Connection

This tab contains settings according the connection of the camera with the ETH interface.



The screenshot shows a configuration window with three tabs: 'General', 'Video', and 'Connection'. The 'Connection' tab is active. It contains three input fields: 'IP address' with the value '2.0.0.2', 'IP port' with the value '80', and 'URL' which is currently empty.

- **IP address**

Define the IP address of your camera. It has to belong to the same unique subnet (The first three out of the four numbers define the subnet) as the ETH channel to which it belongs and it needs to have a unique client identifier (the last out of the four number defines the client identifier).

- **IP port**

Define the IP port of the video stream.

- **URL**

Set the URL on which your cameras stream is to be found. This information should be found in your camera's manual or on the manufacturer's website.



When working with one of the two already supported models (Basler or Linksys) this field will not be accessible, as the URL is already defined. When working with the Custom profile for other cameras you will need to fill in the URL of your camera stream, otherwise the logger will not know where to look for it and will not receive a video signal.



## 8.16 Video signals

The video signals, received from any of the three camera types is are equal in functionality. The setting for these signals can be set over the “Details area” for each respective signal and will be explained in the following.

### 8.16.1 Storage method

In order to store a video signal use “AVI” as storage method. Please refer to (→ [14.10](#)).

### 8.16.2 Settings for video signals

#### General

This tab allows you to activate or deactivate the entire signal by ticking/unticking the checkbox, give a user specific name to your signal if wished and add an additional description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.



The screenshot shows a software interface with four tabs: 'General', 'Format', 'Display', and 'Signal'. The 'General' tab is active and contains the following fields:

- Active:** A checked checkbox.
- Name:** A text input field containing 'Video signal 01'.
- Description:** A text input field containing 'Video signal'.
- Reference:** A text input field containing 'Video signal 01//Basler/ARCOS 1.5 - dataLog'.



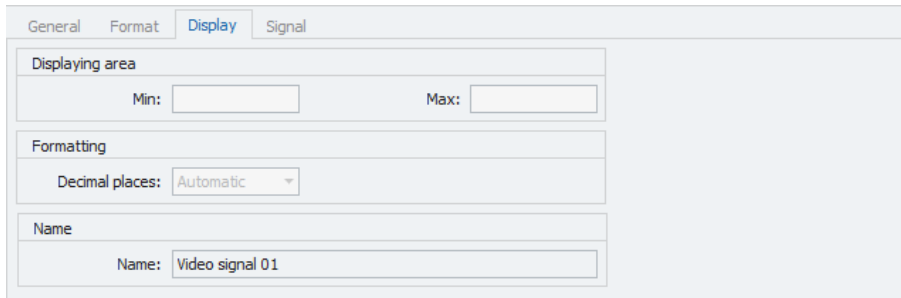
In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).

#### Format

Settings in the “Format”-tab do not have any effect on video signals and will therefore not be explained here.

## Display

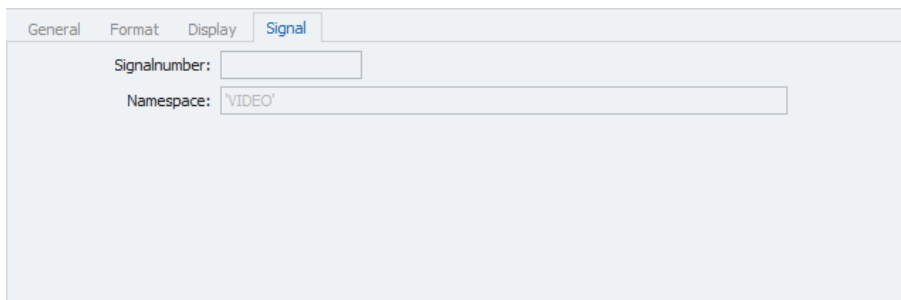
The only setting in the “Display”-tab relevant for video signals is the “Name” setting. It allows you to set a Name to be shown on a display.



The screenshot shows a configuration window with four tabs: General, Format, Display, and Signal. The 'Display' tab is active. It contains three sections: 'Displaying area' with 'Min:' and 'Max:' input fields; 'Formatting' with a 'Decimal places:' dropdown menu set to 'Automatic'; and 'Name' with a text input field containing 'Video signal 01'.

## Signal

This tab allows you to define signal settings.



The screenshot shows the same configuration window with the 'Signal' tab selected. It contains two fields: 'Signalnumber:' with an empty input field, and 'Namespace:' with an input field containing the text 'VIDEO'.

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.

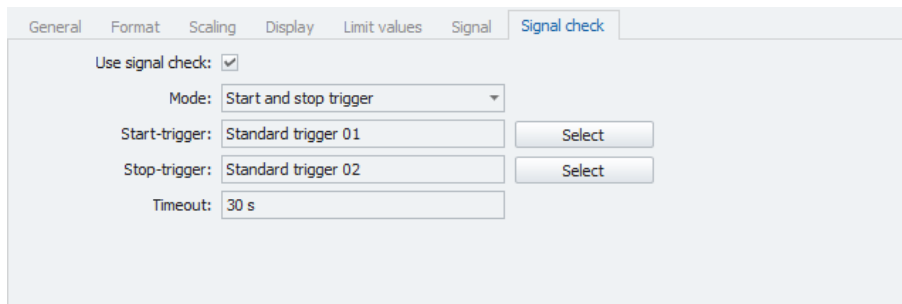
- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.

## Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



## 8.17 Audio recording

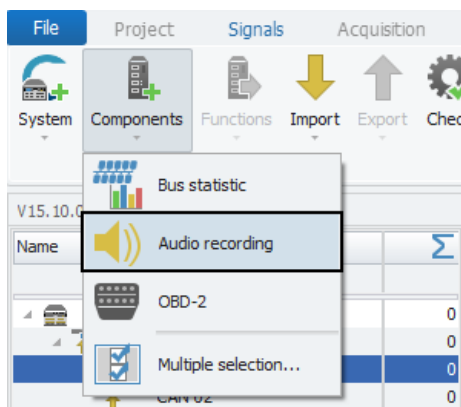
For audio recordings you can connect a microphone to the logger via CAN. The logger will receive the audio signals on one of the CAN channels.

### 8.17.1 Storage method

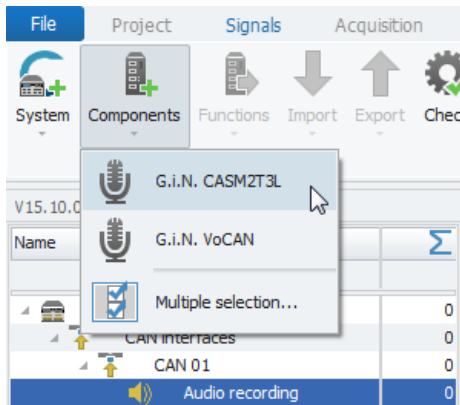
In order to store an audio recording use **“WAV”** as storage method. Please refer to (→ [14.11](#)).

### 8.17.2 Adding an Audio recording

To add an “Audio recording” select the CAN channel to which you wish to add the recording, click the “Components” button in the Ribbon and then choose “Audio recording”.



Once the “Audio recording” component has been added, select the newly created tree element “Audio recording” in the respective CAN channel, click the “Components” button in the Ribbon and then choose one of the two available microphones.



For instructions on microphone specific settings please refer to (→[8.17.5](#)).

### 8.17.3 Tree elements for Audio recordings

Adding an “Audio recording” to your system will add five new elements to your “Measurement task tree”:

- **Audio recording**  
This item represents the entire “Audio recording” and all the included child elements.
- **G.I.N. CASM2T3L / G.I.N. VoCAN**  
This item represents the microphone which you have connected to the logger. At the moment these are the only supported microphone models.
- **Signals**  
This element contains the incoming audio signal.
- **LEDs**  
This element represents the microphone’s LEDs. By selecting this item you will be able to give user specific names to the single LEDs in the details area.
- **Buttons**  
This element represents the microphone’s Buttons. By selecting this item you will be able to give user specific names to the single Buttons in the details area.

V15.10.00.15230 Beta

Name	
ARCOS 1.5 - dataLog	1
CAN interfaces	1
CAN 01	1
Audio recording	1
G.i.N. CASM2T3L	1
Signals	1
LEDs	0
Buttons	0

### 8.17.3.1 Grid area for Audio recordings

In the “grid area” you will see the incoming audio signal. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

V15.10.00.15230 Beta

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
Audio signal 01	<input checked="" type="checkbox"/>		-32768	32767	-32768	32767	1 Hz

### 8.17.4 Details area for Audio recording

When selecting the audio signal in the grid area, you will be able to access the signal's settings in the details area.

#### General

This tab allows you to activate or deactivate the entire signal by ticking/unticking the checkbox, give a user specific name to your signal if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

The sampling rate allows you to set the frequency in which the logger will receive the signal.



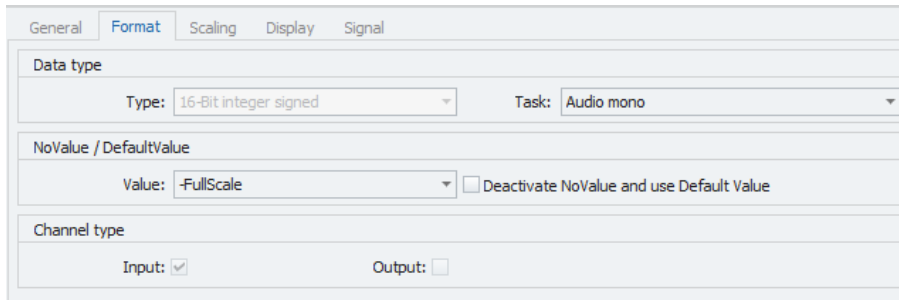
General	Format	Scaling	Display	Signal
Active:	<input checked="" type="checkbox"/>			
Name:	Audio signal 01			
Description:	Audio signal			
Reference:	Audio signal 01///Signals/ARCOS 1.5 - dataLog			
Sampling rate:	1 Hz			



In the field "Name" project parameters can be used as variables. For more information please refer to ([→5.6](#)).

## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case “16-Bit integer unsigned”) and allows you to special tasks for this signal such as “Audio mono”.

- **NoValue / DefaultValue**


This field allows you to define the value that will be shown if a signal value is read as invalid.

## Scaling

The settings in the “Scaling”-tab are not relevant for working with audio recordings.

## Display

The only setting in the “Display”-tab relevant for audio recordings is the “Name” setting. It allows you to set a name to be shown on a display.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Signal

This tab allows you to define signal settings.



- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.



## 8.17.5 Microphone settings

Both supported microphones, the G.I.N. VoCAN and the G.I.N. CASM2T3L allow for some user specific configuration, which will be explained in the following. With the exception of a few functions these settings are the same for both models. Whenever there is a function specific to one of the models, this will be noted in parenthesis.

### 8.17.5.1 Signals

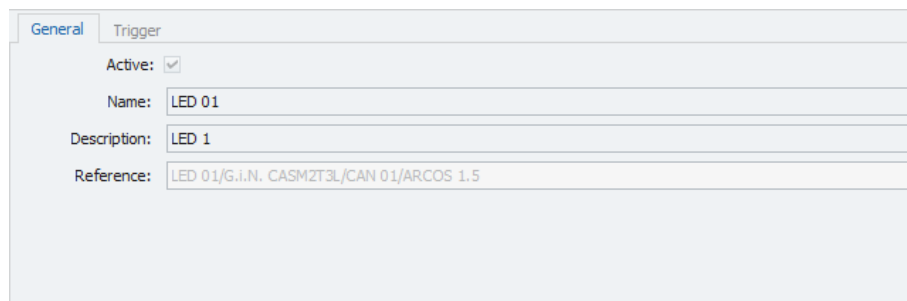
The signal settings for Audio recordings have been explained at the beginning of this chapter in the section **Grid area for Audio recording** (→[8.17.3.1](#)) and the section **Details area for Audio recording** (→[8.17.4](#)).

### 8.17.5.2 LEDs

Both models come with a number of LEDs (G.I.N. VoCAN with 4 LEDs and G.I.N. CASM2T3L with 3 LEDs), whose behaviour can be customized. To adapt the settings for a desired LED, click on the “LEDs” element in the tree, select the desired LED in the grid area and navigate to the details area.

#### General

This tab allows you to give a user specific name and description to the LED.

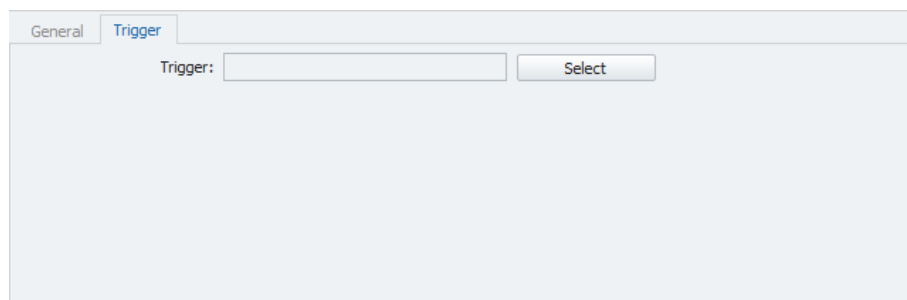


The screenshot shows the 'General' tab of an LED configuration window. It contains the following fields:

- Active:** A checked checkbox.
- Name:** A text input field containing 'LED 01'.
- Description:** A text input field containing 'LED 1'.
- Reference:** A text input field containing 'LED 01/G.I.N. CASM2T3L/CAN 01/ARCOS 1.5'.

#### Trigger

This tab allows you to assign a trigger to the LED, upon whose firing the LED will light up.



The screenshot shows the 'Trigger' tab of the same LED configuration window. It contains a 'Trigger:' label followed by an empty text input field and a 'Select' button.

### 8.17.5.3 Buttons

Both models come with a number of buttons, whose activation will set a trigger. The associated triggers cannot be changed. To see an overview of the available buttons select the "Buttons" element in the tree and navigate to the grid area.

Name	Active	Description	Index
☐	<input type="checkbox"/>		
▶ Red button 01	<input checked="" type="checkbox"/>	Red button	1
Microphone button 01	<input checked="" type="checkbox"/>	Microphone button	2

*Overview of buttons per microphone model*

Microphone model	Overview of buttons
G.I.N. VoCAN	Red button 01 Microphone button 01
G.I.N. CASM2T3L	Red button 01

## 8.18 DIN (Digital input signals)

The “DIN” module offers a digital bit-channel, which is directly configured as a digital signal and then acquired. This means they can be used in the logger configuration in the same way as conventional bus signals.

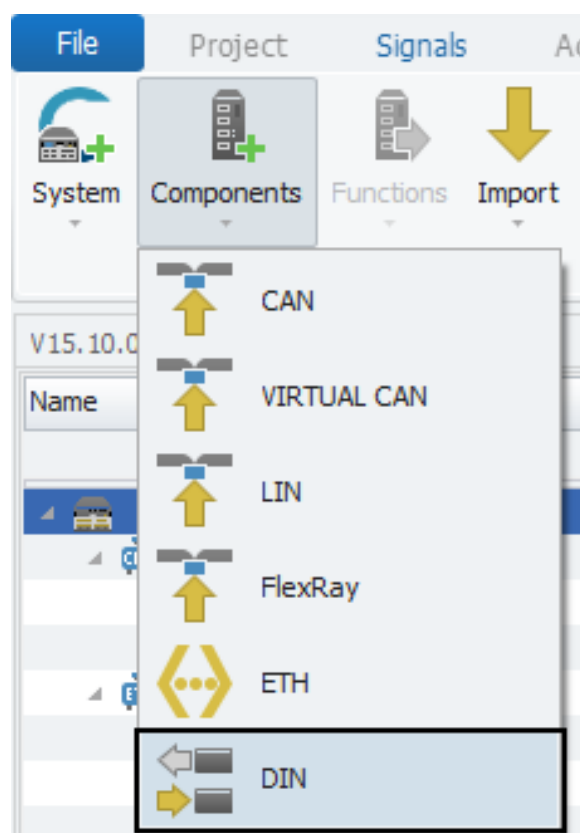
### 8.18.1 Storage method

In order to store incoming signals on a DIN channel use one of the following signal storage methods.

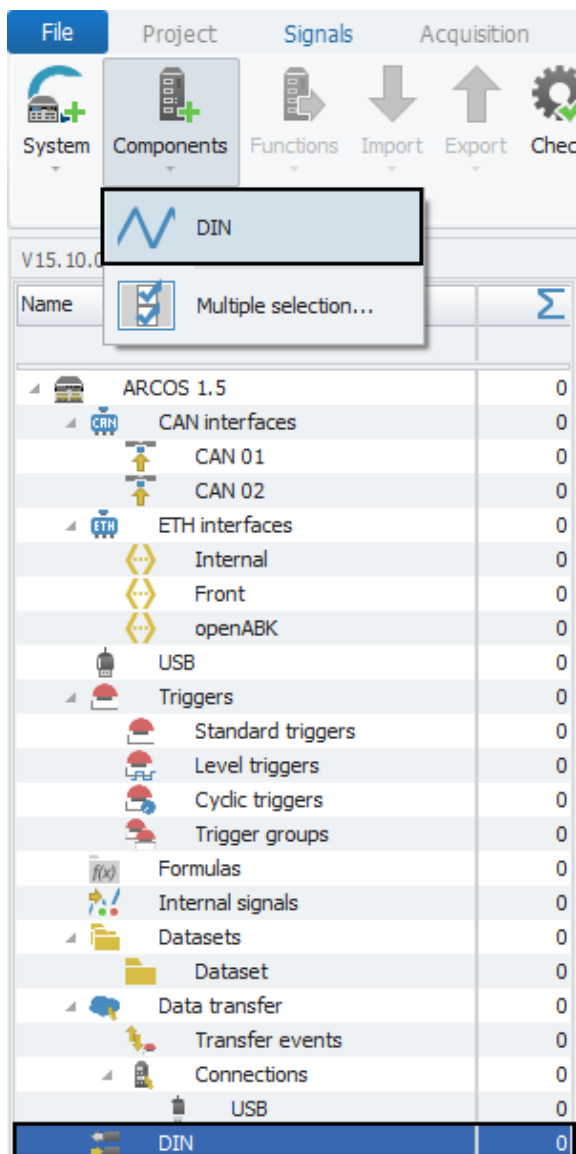
- ATEX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

### 8.18.2 Adding the DIN-Interface

In order to work with digital input signals, you will first need to add the “DIN” interface to your system. To do so, select the system in the tree, click the “Components” button in the Ribbon and then choose “DIN”.



Once the “DIN” interface has been added to your system, you can then add multiple “DIN” channels, in order to acquire digital signals. To do so, select the “DIN” interface in the tree, click the “Components” button in the Ribbon and then choose “DIN”.



In order to acquire digital signals via “DIN” channels, each “DIN” channel needs to be matched with the corresponding digital hardware channels of the logger. To find out the digital hardware channel numbers please consult your loggers webinterface.

Once you know the hardware channel number, navigate to the “Signals” tab in “Details area” of the corresponding “DIN” channel in the grid area and enter the number.

### 8.18.3 Signal properties

Changes and errors excepted.

## 8.18 DIN (DIGITAL INPUT SIGNALS)

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max
?	<input type="checkbox"/>					
▶ DIN 01	<input checked="" type="checkbox"/>		0	1	0	1

General   Format   Scaling   Display   **Signal**

Signal number:

Hold last value:

Timeout:

Hardware channel:

Namespace:

### 8.18.3.1 Tree elements for DIN signals

After having added the “DIN” interface to your system it will appear as a tree element with the name “DIN”.

▶ Data transfer	0
Transfer events	0
▶ Connections	0
USB	0
<b>DIN</b>	<b>1</b>

### 8.18.3.2 Grid area for DIN signals

In the “grid area” you will be presented with an overview of the DIN channels which have been added to your system so far. Each DIN channel can only receive one signal and therefore each DIN channel is treated as a signal.

Also you can find here two important functions, which are the “**Column chooser**” (→4.3.1) and the “**Filter editor**” (→4.3.2).

### 8.18.3.3 Details area for DIN signals

The Details area shows settings either for the selected tree element “DIN” or the selected “DIN” channel in the grid area. In case the tree element is selected, the details area will

only show the “General” tab. Please refer to (→[4.2.2](#)).

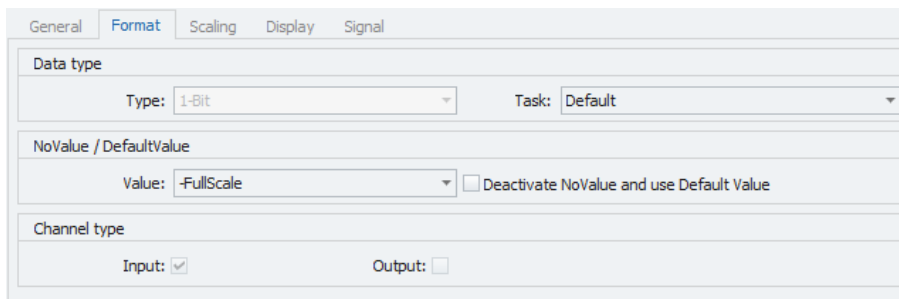
In case a “DIN” channel is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

## General

Please refer to (→[4.2.2](#)).

## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case “1-Bit”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a “Input” channel or “Output” channel.

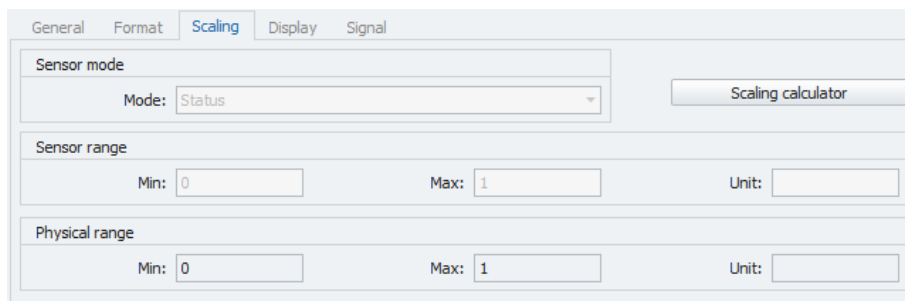
## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”,

Changes and errors excepted.



The screenshot shows a software interface with a 'Scaling' tab selected. It contains the following elements:

- Sensor mode:** A dropdown menu currently showing 'Status'. To its right is a 'Scaling calculator' button.
- Sensor range:** A section with three input fields: 'Min:' containing '0', 'Max:' containing '1', and 'Unit:' which is empty.
- Physical range:** A section with three input fields: 'Min:' containing '0', 'Max:' containing '1', and 'Unit:' which is empty.

“Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

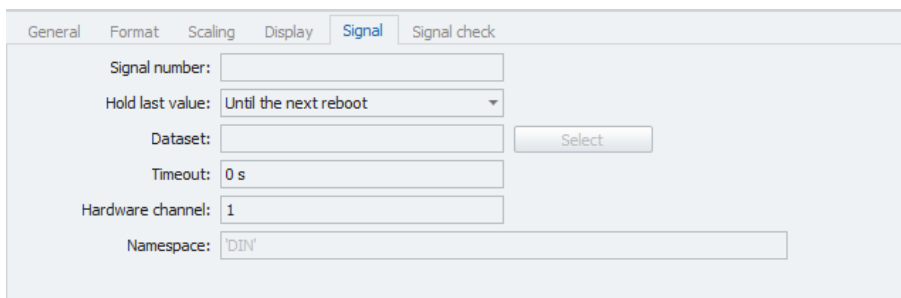
The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Signal

This tab allows you to define signal settings.



- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

Changes and errors excepted.



- **Dataset**

If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to “NaN (Not a Number)” and will be displayed as “-” in a display.

- **Hardware channel**

Assign the corresponding digital hardware channel number from which you would like to acquire data. The hardware channel number can be found out via the logger's webinterface.

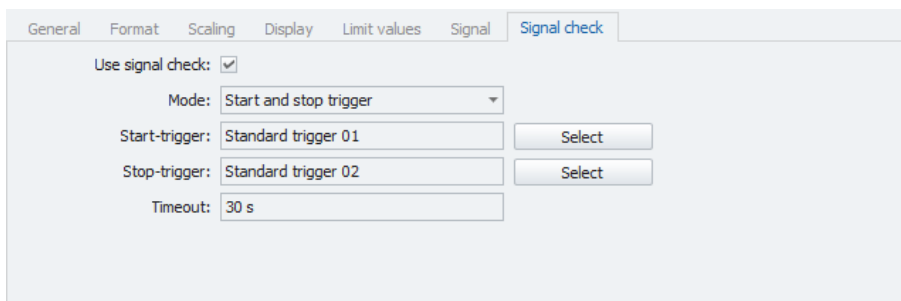
- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.

## Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



The screenshot shows a software interface with several tabs: General, Format, Scaling, Display, Limit values, Signal, and Signal check. The 'Signal check' tab is active. It contains the following settings:

- Use signal check:
- Mode: Start and stop trigger (dropdown menu)
- Start-trigger: Standard trigger 01 (text input) with a 'Select' button
- Stop-trigger: Standard trigger 02 (text input) with a 'Select' button
- Timeout: 30 s (text input)

## 8.19 DOUT (Digital output signals)

The “DOUT” module offers a digital bit-channel on which a calculated digital signal can be put out.

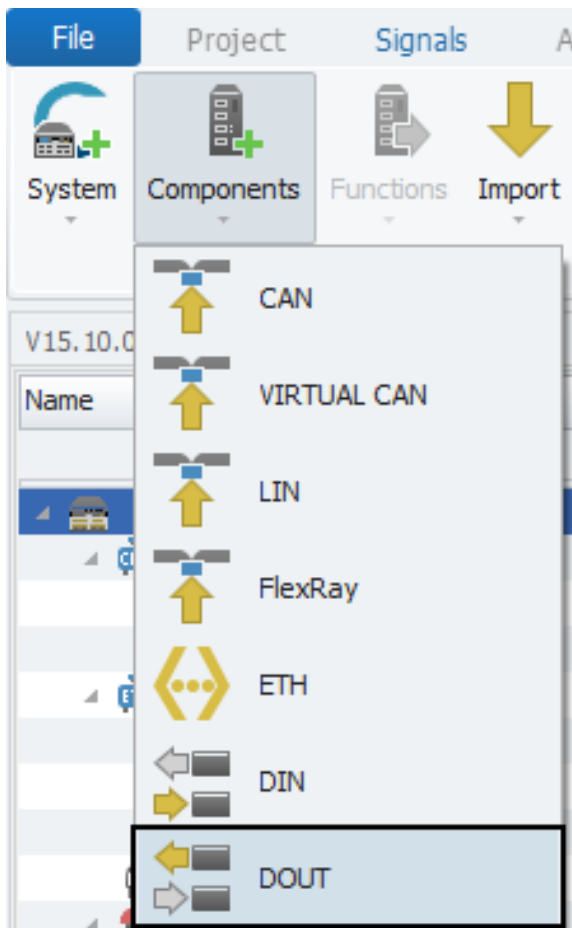
### 8.19.1 Storage method

In order to store outgoing signals on a DOUT channel use one of the following signal storage methods.

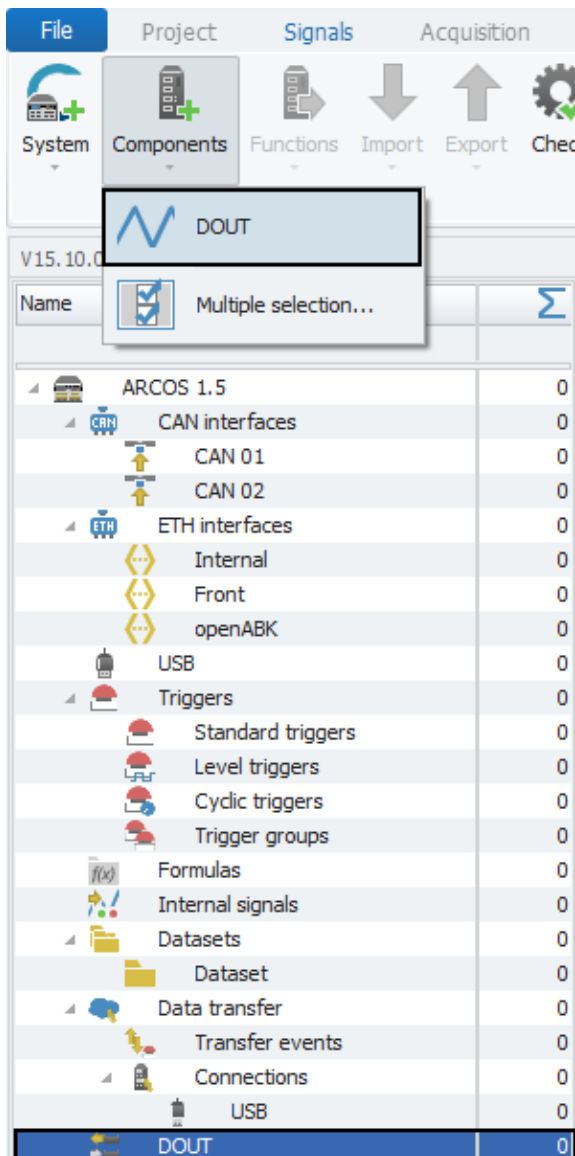
- ATFX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

### 8.19.2 Adding the DOUT-Interface

In order to work with digital output signals, you will first need to add the “DOUT” interface to your system. To do so, select the system in the tree, click the “Components” button in the Ribbon and then choose “DOUT”.



Once the “DOUT” interface has been added to your system, you can then add multiple “DOUT” channels, in order to put out digital signals. To do so, select the “DOUT” interface in the tree, click the “Components” button in the Ribbon and then choose “DOUT”.



In order to put out digital signals via “DOUT” channels, each “DOUT” channel needs to be matched with the corresponding digital hardware channels of the logger. To find out the digital hardware channel numbers please consult your loggers webinterface.

Once you know the hardware channel number, navigate to the “Signals” tab in “Details area” of the corresponding “DOUT” channel in the grid area and enter the number.

### 8.19.3 Signal properties

Changes and errors excepted.

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max
☿	<input type="checkbox"/>					
▶ DOUT 01	<input checked="" type="checkbox"/>		0	1	0	1

General   Format   Scaling   Display   Calculation   **Signal**

Signal number:

Hold last value: Until the next reboot ▼

Hardware channel:

Namespace:

### 8.19.3.1 Tree elements for DOUT signals

After having added the “DOUT” interface to your system it will appear as a tree element with the name “DOUT”.

☰ Data transfer	0
☰ Transfer events	0
☰ Connections	0
☰ USB	0
<b>☰ DOUT</b>	<b>1</b>

### 8.19.3.2 Grid area for DOUT signals

In the “grid area” you will be presented with an overview of the DOUT channels which have been added to your system so far. Each DOUT channel can only receive one signal and therefore each DOUT channel is treated as a signal.

Also you can find here two important functions, which are the “**Column chooser**” (→[4.3.1](#)) and the “**Filter editor**” (→[4.3.2](#)).

### 8.19.3.3 Details area for DOUT signals

The Details area shows settings either for the selected tree element “DOUT” or the selected “DOUT” channel in the grid area. In case the tree element is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

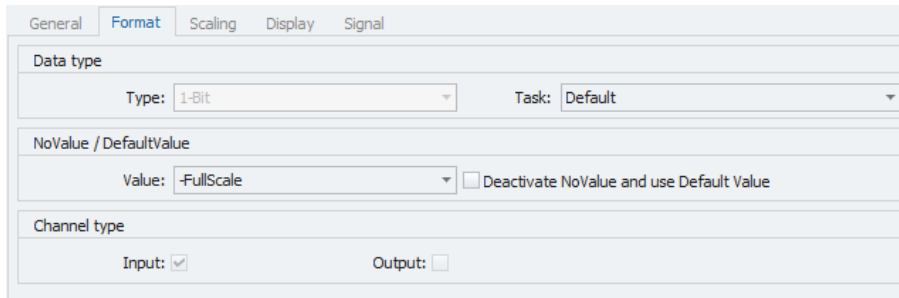
In case a “DOUT” channel is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

## General

Please refer to (→[4.2.2](#)).

## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case “1-Bit”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

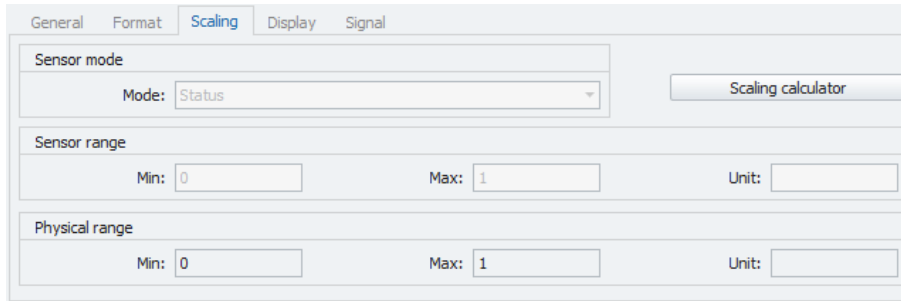
This field tells you whether you are dealing with a “Input” channel or “Output” channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “Scaling calculator” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

- **Sensor Mode**

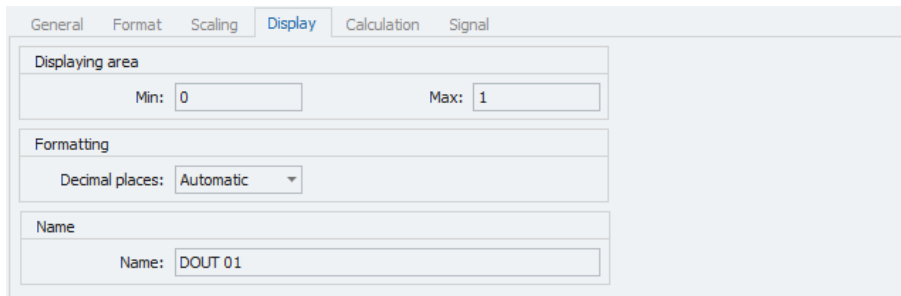
The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.



- **Sensor Range**  
Shows the raw value range of the signal.
- **Physical Range**  
Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the [IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”](#).

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



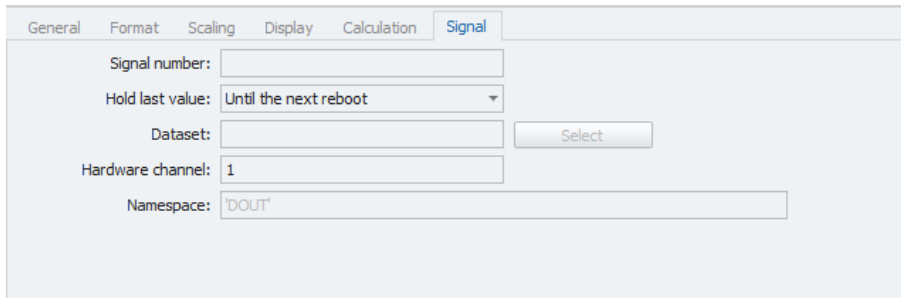
- **Displaying area**  
Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.
- **Formatting**  
The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.
- **Name**  
Allows you to set a Name to be shown on a display.

## Calculation

In this tab you set the formula to calculate the “DOUT” signal. This functionality has been explained in depth in the “Formulas” section. Please refer to (→[8.23.1.4](#)).

## Signal

This tab allows you to define signal settings.



The screenshot shows the 'Signal' configuration tab with the following fields:

- Signal number:
- Hold last value:
- Dataset:
- Hardware channel:
- Namespace:

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn’t send data for the specified time period, the value of the signal is set to “NaN (Not a Number)” and will be displayed as “-” in a display.

- **Hardware channel**

Assign the corresponding digital hardware channel number from which you would like to put out data. The hardware channel number can be found out via the logger’s webinterface.

- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.

## 8.20 Analog signals

Your logger is equipped with a number of analog input channels and the “Analog” interface of the plugin allows for direct acquisition of analog signals coming in on these channels. There are three types of analog signals that can be acquired via the “Analog” interface:

- **Voltage** (→[8.20.3.3](#)) Allows you to directly import a raw voltage and, with the help of the “Scaling calculator”, to transform it into a signal type of your desire.
- **Counter/frequency** (→[8.20.3.4](#)) Allows you to define a voltage-threshold for the incoming signal and thus transform the raw voltage into a counter or frequency.
- **Duty cycle** (→[8.20.3.5](#)) Allows you to acquire the hightime or the lowtime in percent of a signal.

### 8.20.1 Storage method

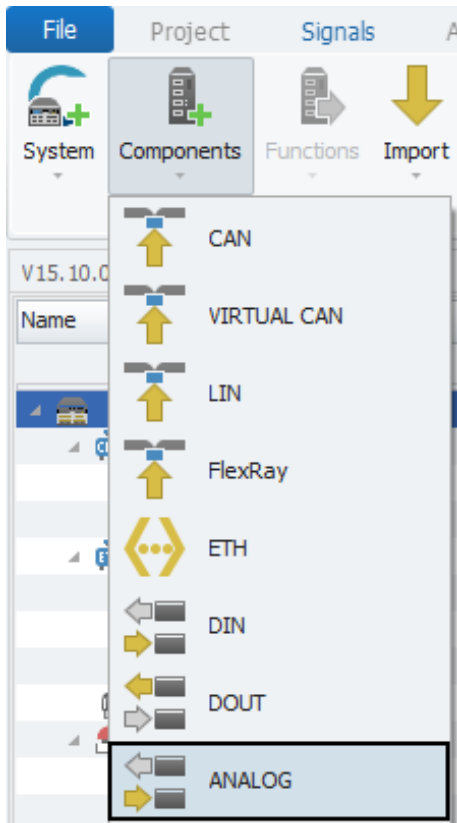
In order to store incoming signals on an analog channel use one of the following signal storage methods.

- ATEX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))



## 8.20.2 Adding the Analog Interface

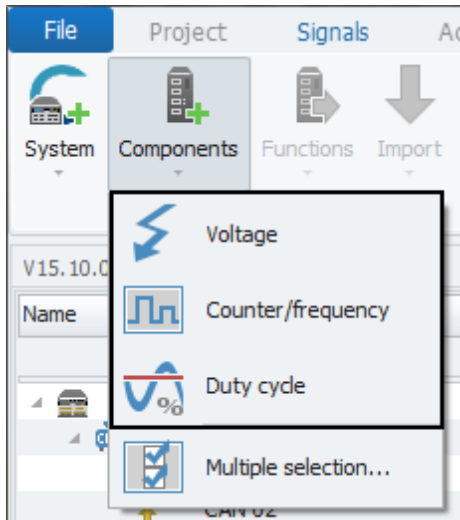
In order to work with incoming analog signals, you will first need to add the “Analog” interface to your system. To do so, select the system in the tree, click the “Components” button in the Ribbon and then choose “Analog”.



Once the “Analog” interface has been added to your system, you can then add multiple “Analog” channels of three different types (each type will acquire a different type of signal).

To do so, select the “Analog” interface in the tree, click the “Components” button in the Ribbon and then choose one of the three types “Voltage” (→[8.20.3.3](#)), “Counter/frequency” (→[8.20.3.4](#)) or “Duty cycle” (→[8.20.3.5](#)).

For specifics on the configuration for each of these signal types please click on the respective links in the preceding paragraph.





In order to acquire signals on analog channels, each analog channel needs to be matched with the corresponding analog hardware channels of the logger. To find out the analog hardware channel numbers please consult your loggers webinterface.

Once you know the hardware channel number, navigate to the "Signals" tab in "Details area" of the corresponding analog channel in the grid area and enter the number.

	Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max
⌵		<input type="checkbox"/>					
⌵	ANALOG 01	<input checked="" type="checkbox"/>	V	-60,000	60,000	-60	60

General   Format   Scaling   Display   Excitation   **Signal**

Signal number:

Hold last value:

Timeout:

Hardware channel:

Namespace:

## 8.20.3 Signal properties

### 8.20.3.1 Tree elements for Analog signals

After having added the “Analog” interface to your system it will appear as a tree element with the name “Analog”.

Data transfer	0
Transfer events	0
Connections	0
USB	0
<b>ANALOG</b>	<b>3</b>

### 8.20.3.2 Grid area for Analog signals

In the “grid area” you will be presented with an overview of the Analog channels which have been added to your system so far. Each Analog channel can only receive one signal and therefore each Analog channel is treated as a signal.

Also you can find here two important functions, which are the “Column chooser” (→[4.3.1](#)) and the “Filter editor” (→[4.3.2](#)).

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
ANALOG 01	<input checked="" type="checkbox"/>	V	-60,000	60,000	-60	60	1 Hz

### 8.20.3.3 Details area for Analog signals (Voltage)

The Details area shows settings either for the selected tree element “Analog” or the selected “Voltage” channel in the grid area. In case the tree element is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

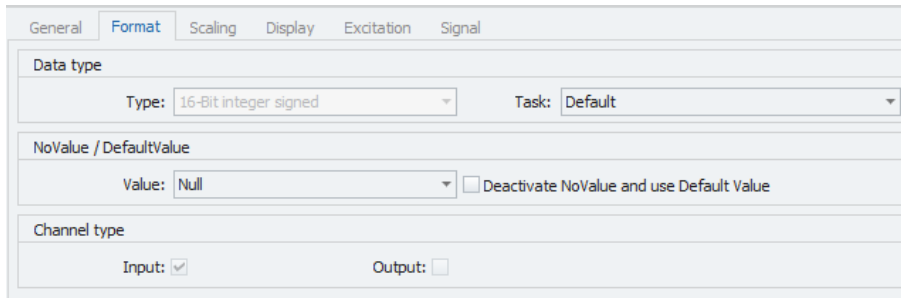
In case a “Voltage” channel is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

Please refer to (→[4.2.2](#)).

## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case “16-Bit integer signed”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

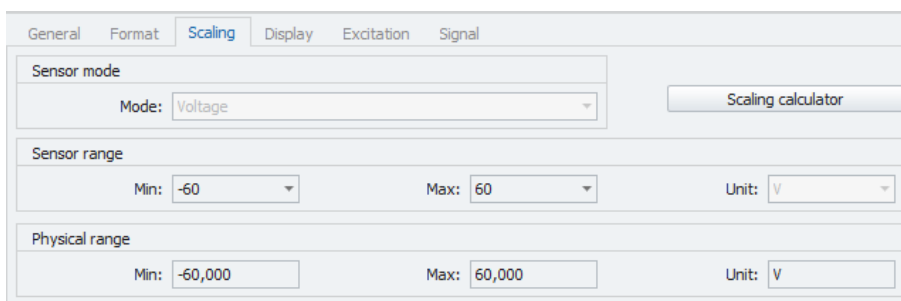
This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a “Input” channel or “Output” channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “Scaling calculator” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the [IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”](#).



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as "Status", "Voltage", "Frequency" or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

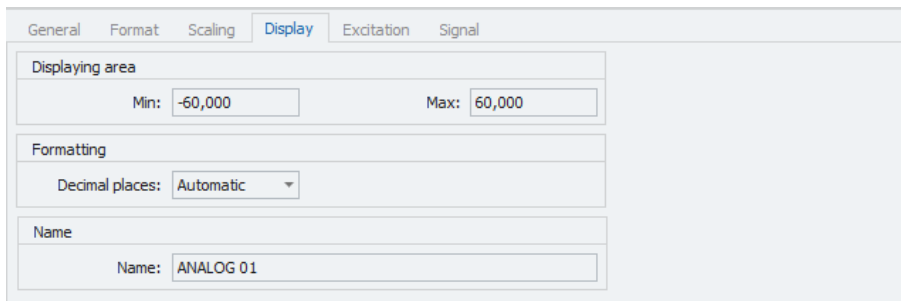
Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to "scale" your signal and also define the unit to use. For more refined scaling please use the "Scaling calculator" and refer to the IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling".

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



The screenshot shows a software interface with several tabs: General, Format, Scaling, Display (selected), Excitation, and Signal. The 'Display' tab contains three sections: 'Displaying area' with 'Min: -60,000' and 'Max: 60,000'; 'Formatting' with 'Decimal places: Automatic'; and 'Name' with 'Name: ANALOG 01'.

- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

The dropdown menu "Decimal places" allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Excitation

This tab allows you to provide the analog sensor with excitation if necessary and also to set the voltage of the excitation.

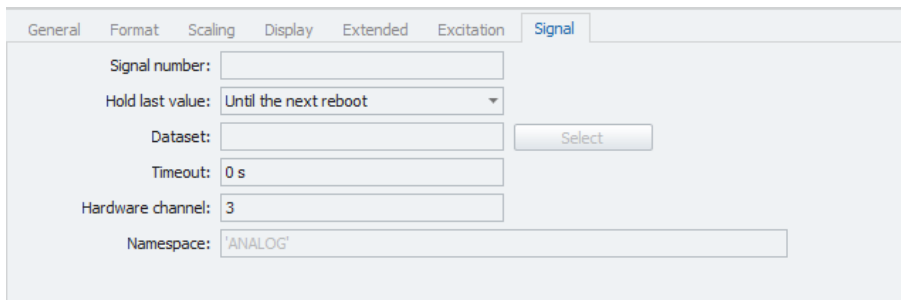


The screenshot shows the 'Excitation' tab with the following settings:

- Sensor excitation:  5 V

## Signal

This tab allows you to define signal settings.



The screenshot shows the 'Signal' tab with the following settings:

- Signal number:
- Hold last value:
- Dataset:
- Timeout:
- Hardware channel:
- Namespace:

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting "Hold last value" has been set to "Until end of dataset", you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to "NaN (Not a Number)" and will be displayed as "-" in a display.

- **Hardware channel**

Assign the corresponding digital hardware channel number from which you would like to put out data. The hardware channel number can be found out via the logger's webinterface.

- **Namespace**

The “Namespace” serves as unique identifier for the signal inside the logger.

#### 8.20.3.4 Details area for Analog signals (Counter/frequency)

The Details area shows settings either for the selected tree element “Analog” or the selected “Counter/frequency” channel in the grid area. In case the tree element is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

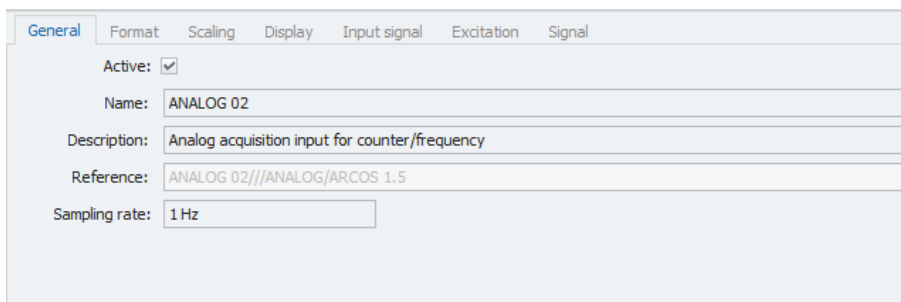
In case a “Counter/frequency” channel is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

Please refer to (→[4.2.2](#)).

#### Format

This tab contains information and options regarding file format, tasks and Channel type.



The screenshot shows a software interface with a tabbed menu at the top containing 'General', 'Format', 'Scaling', 'Display', 'Input signal', 'Excitation', and 'Signal'. The 'General' tab is active and displays the following settings:

- Active:
- Name: ANALOG 02
- Description: Analog acquisition input for counter/frequency
- Reference: ANALOG 02///ANALOG/ARCOS 1.5
- Sampling rate: 1 Hz

- **Data type**

This field tells you the type of data (in this case “32-Bit integer unsigned”) and allows you to apply special tasks for this signal such as “GPS Longitude”, “GPS Latitude”, “UTC hour”, “Audio mono” and more.

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a signal value is read as invalid.

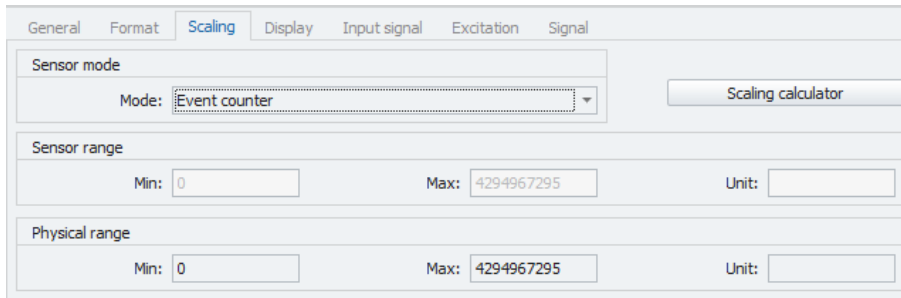
- **Channel type**

This field tells you whether you are dealing with a “Input” channel or “Output” channel.



## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.



- **Sensor Mode**

The sensor mode lets you further specify, in which way the signal should be used. There are three options:

**Frequency** Determines the frequency of an analog input.

**Event counter** Determines the occurrence of an event for an analog input.

**Event counter with direction** Determines the occurrence of an event for an analog input, including a directional input to count either up or down. This mode requires two analog channels, the prefix of the second channel determines the direction of the event counter.

- **Sensor Range**

Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

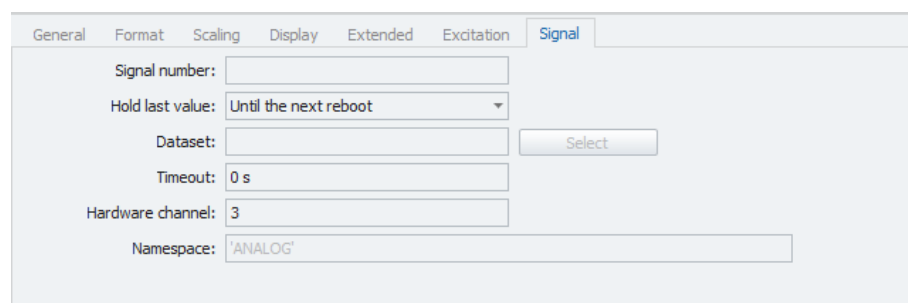
The dropdown menu "**Decimal places**" allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Input signal

This tab provides settings regarding the Input signal. These settings are crucial for a correct functionality of the Counter/frequency mode.



- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".

- **Hold last value**  
Specify, for how long the last value of the signal will be hold.
- **Dataset**  
If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.
- **Timeout**  
Specify the timeout period for the current signal. If the data source doesn’t send data for the specified time period, the value of the signal is set to “NaN (Not a Number)” and will be displayed as “-” in a display.
- **Hardware channel**  
Assign the corresponding digital hardware channel number from which you would like to put out data. The hardware channel number can be found out via the logger’s webinterface.
- **Namespace**  
The “Namespace” serves as unique identifier for the signal inside the logger.

## Excitation

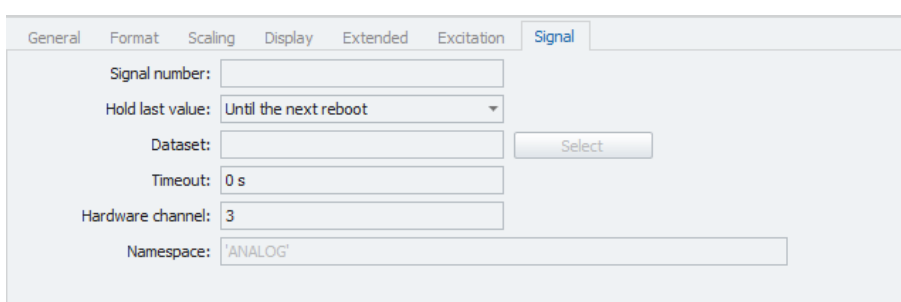
This tab allows you to provide the analog sensor with excitation if necessary and also to set the voltage of the excitation.



The screenshot shows the 'Excitation' tab selected in a software interface. The tab bar includes 'General', 'Format', 'Scaling', 'Display', 'Excitation', and 'Signal'. The 'Excitation' tab contains a 'Sensor excitation' section with a checkbox and a text input field containing '5 V'.

## Signal

This tab allows you to define signal settings.



The screenshot shows the 'Signal' tab selected in a software interface. The tab bar includes 'General', 'Format', 'Scaling', 'Display', 'Extended', 'Excitation', and 'Signal'. The 'Signal' tab contains several settings: 'Signal number' (empty text field), 'Hold last value' (dropdown menu set to 'Until the next reboot'), 'Dataset' (empty text field with a 'Select' button), 'Timeout' (text field set to '0 s'), 'Hardware channel' (text field set to '3'), and 'Namespace' (text field set to 'ANALOG').

- **Signal number**  
Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".
- **Hold last value**  
Specify, for how long the last value of the signal will be hold.
- **Timeout**  
Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to "NaN (Not a Number)" and will be displayed as "-" in a display.
- **Hardware channel**  
Assign the corresponding digital hardware channel number from which you would like to put out data. The hardware channel number can be found out via the logger's webinterface.
- **Namespace**  
The "Namespace" serves as unique identifier for the signal inside the logger.

### 8.20.3.5 Details area for Analog signals (Duty cycle)

The Details area shows settings either for the selected tree element "Analog" or the selected "Duty cycle" channel in the grid area. In case the tree element is selected, the details area will only show the "General" tab. Please refer to (→[4.2.2](#)).

In case a "Duty cycle" channel is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

Please refer to (→[4.2.2](#)).

#### Format

This tab contains information and options regarding file format, tasks and Channel type.

- **Data type**  
This field tells you the type of data (in this case "16-Bit integer unsigned") and allows you to apply special tasks for this signal such as "GPS Longitude", "GPS Latitude", "UTC hour", "Audio mono" and more.

General	Format	Scaling	Display	Extended	Excitation	Signal
Active: <input checked="" type="checkbox"/>						
Name: ANALOG 03						
Description: Analog acquisition input for duty cycle						
Reference: ANALOG 03///ANALOG/ARCOS 1.5						
Sampling rate: 1 Hz						

- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a "Input" channel or "Output" channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The "Scaling calculator" allows for more refined scaling options with a large range of functions. For details on how to use the "Scaling calculator" please refer to the [IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling"](#).

General	Format	Scaling	Display	Extended	Excitation	Signal
Sensor mode						
Mode: Duty cycle						Scaling calculator
Sensor range						
Min: 0		Max: 100		Unit: %		
Physical range						
Min: 0,0		Max: 100,0		Unit: %		

- **Sensor Mode**

The sensor mode tells the type of signal. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

Shows the raw value range of the signal.

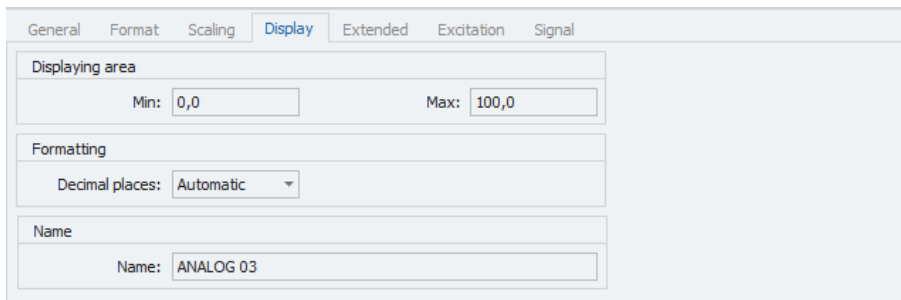
- **Physical Range**

Allows you to set a range to which you would like to "scale" your signal and also define the unit to use. For more refined scaling please use the "Scaling calculator" and

refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



The screenshot shows the 'Display' configuration window with the following settings:

- Displaying area:** Min: 0,0; Max: 100,0
- Formatting:** Decimal places: Automatic
- Name:** ANALOG 03

- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

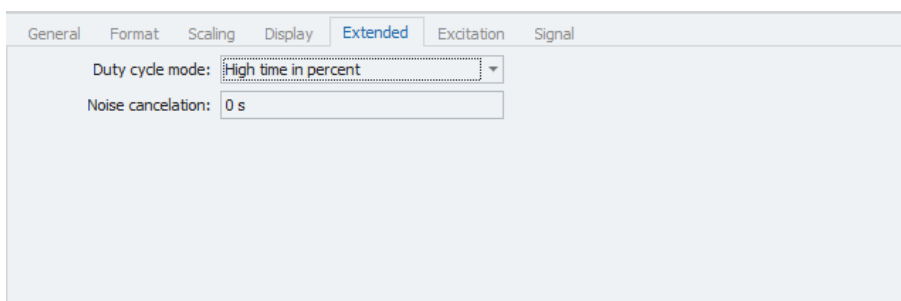
The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Extended

This tab provides extended settings regarding the duty cycle mode.



The screenshot shows the 'Extended' configuration window with the following settings:

- Duty cycle mode:** High time in percent
- Noise cancelation:** 0 s

- **Duty cycle mode**

Define whether the high time or the low time will be put out in percent.

- **Noise cancellation**

Define the noise cancellation delay time.

### Excitation

This tab allows you to provide the analog sensor with excitation if necessary and also to set the voltage of the excitation.

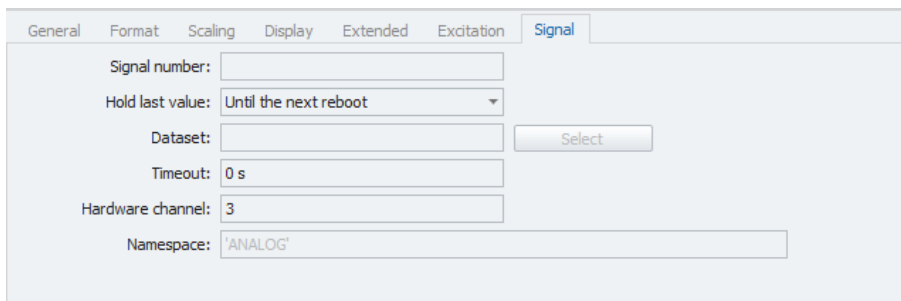


General Format Scaling Display **Excitation** Signal

Sensor excitation:  5 V

### Signal

This tab allows you to define signal settings.



General Format Scaling Display Extended Excitation **Signal**

Signal number:

Hold last value:

Dataset:

Timeout:

Hardware channel:

Namespace:

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting "Hold last value" has been set to "Until end of dataset", you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to "NaN (Not a Number)" and will be displayed as "-" in a display.

- **Hardware channel**

Assign the corresponding digital hardware channel number from which you would like to put out data. The hardware channel number can be found out via the logger's webinterface.

- **Namespace**

The "Namespace" serves as unique identifier for the signal inside the logger.

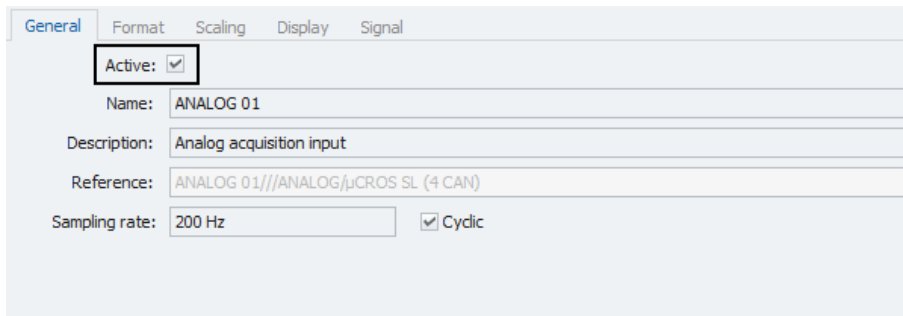
#### 8.20.4 Analog signals on $\mu$ Cros SL

The logger-model  $\mu$ Cros SL supports analog signals and by default the "Analog" interface is activated and can be found in the measurement task tree. These analog signals for  $\mu$ Cros SL are of the type "Voltage" but they have some differences compared to regular analog voltage-signals on other loggers.

This section of the manual will explain how they differ from regular analog voltage signals.

#### Overview of the differences compared to regular analog voltage-signals

- Analog signals on  $\mu$ Cros SL are always available in the "Analog" interface and cannot be deleted. They are by default inactive and need to be marked active in order to be used.



General | Format | Scaling | Display | Signal

Active:

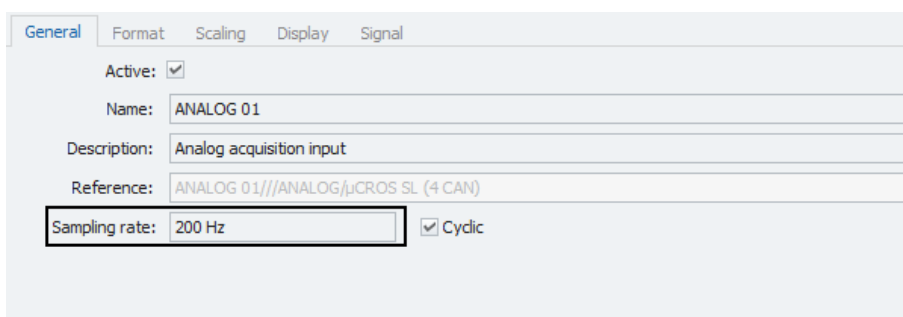
Name: ANALOG 01

Description: Analog acquisition input

Reference: ANALOG 01///ANALOG/ $\mu$ CROS SL (4 CAN)

Sampling rate: 200 Hz  Cyclic

- The sampling rate for analog signals on  $\mu$ Cros SL cannot exceed a maximum of 200 Hz.



General | Format | Scaling | Display | Signal

Active:

Name: ANALOG 01

Description: Analog acquisition input

Reference: ANALOG 01///ANALOG/ $\mu$ CROS SL (4 CAN)

Sampling rate: 200 Hz  Cyclic



- The sensor rate of analog signals on  $\mu$ Cros SL is set fixed from 0 to 40 V. It cannot be changed.



The screenshot shows a software interface with tabs for 'General', 'Format', 'Scaling', 'Display', and 'Signal'. The 'Scaling' tab is active. It contains a 'Sensor mode' dropdown menu set to 'Voltage' and a 'Scaling calculator' button. Below this is a 'Sensor range' section, which is highlighted with a black border. It includes three dropdown menus: 'Min:' set to '0', 'Max:' set to '40', and 'Unit:' set to 'V'. At the bottom is a 'Physical range' section with three input fields: 'Min:' containing '0,00', 'Max:' containing '40,00', and 'Unit:' set to 'V'.

- The resolution of the signal is 10 bit.
- The  $\mu$ Cros SL is not able to put out voltage on the analog channel, therefore the tab "Excitation" does not exist.

Apart from these differences the analog signals on  $\mu$ Cros SL are identical to regular analog voltage signals and can be treated in the same way. For more information regarding regular analog voltage signals please refer to ( $\rightarrow$ [8.20.3.3](#)).

## 8.21 Thermo

The “Thermo” module offers an analog channel, through which a “Thermo” signal can be acquired.

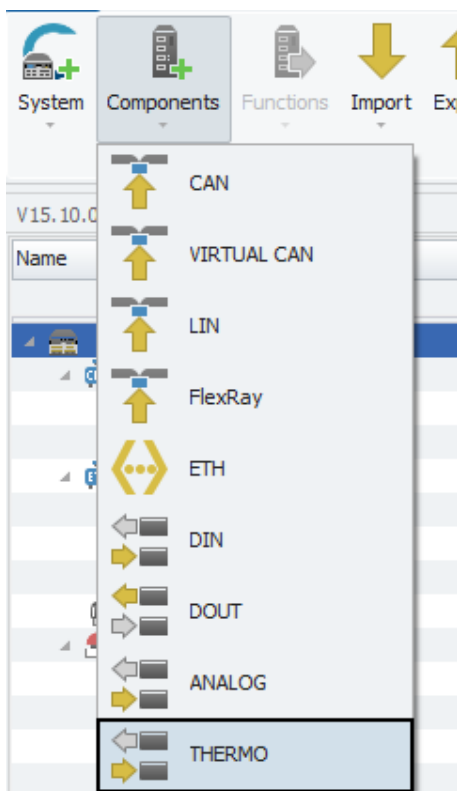
### 8.21.1 Storage method

In order to store incoming signals on a Thermo channel use one of the following signal storage methods.

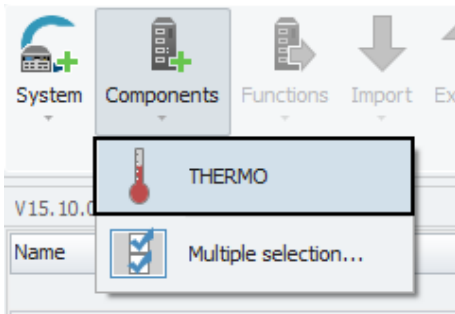
- AFX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

### 8.21.2 Adding the Thermo-Interface

In order to work with “Thermo” signals, you will first need to add the “Thermo” interface to your system. To do so, select the system in the tree, click the “Components” button in the Ribbon and then choose “Thermo”.



Once the “Thermo” interface has been added to your system, you can then add multiple “Thermo” channels, in order to acquire digital signals. To do so, select the “Thermo” interface in the tree, click the “Components” button in the Ribbon and then choose “Thermo”.



In order to acquire “Thermo” signals via “Thermo” channels, each “Thermo” channel needs to be matched with the corresponding hardware channels of the logger. To find out the hardware channel numbers please consult your loggers webinterface.

Once you know the hardware channel number, navigate to the “Signals” tab in “Details area” of the corresponding “Thermo” channel in the grid area and enter the number.

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
THERMO.01	<input checked="" type="checkbox"/>	°C	-50,00	1200,00	-50	1200	1 Hz

General   Format   Scaling   Terminal   Display   **Signal**

Signal number:

Hold last value:

Timeout:

Hardware channel:

Namespace:

## 8.21.3 Signal properties

### 8.21.3.1 Tree elements for Thermo signals

After having added the “Thermo” interface to your system it will appear as a tree element with the name “Thermo”.

Data transfer	0
Transfer events	0
Connections	0
USB	0
<b>THERMO</b>	<b>1</b>

### 8.21.3.2 Grid area for Thermo signals

In the “grid area” you will be presented with an overview of the Thermo channels which have been added to your system so far. Each Thermo channel can only receive one signal and therefore each Thermo channel is treated as a signal.

Also you can find here two important functions, which are the “**Column chooser**” (→[4.3.1](#)) and the “**Filter editor**” (→[4.3.2](#)).

### 8.21.3.3 Details area for Thermo signals

The Details area shows settings either for the selected tree element “Thermo” or the selected “Thermo” channel in the grid area. In case the tree element is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

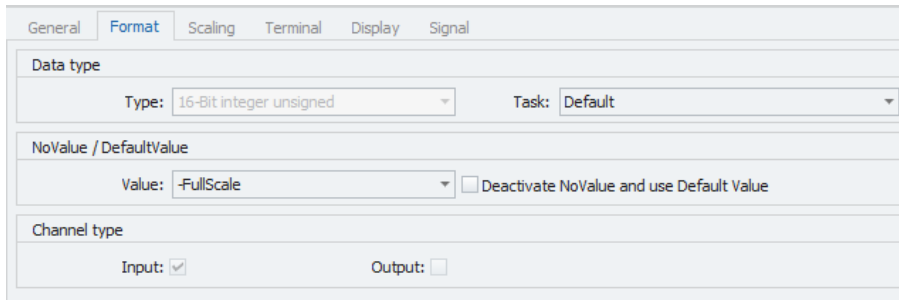
In case a “Thermo” channel is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

Please refer to (→[4.2.2](#)).

## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case "16-Bit integer unsigned") and allows you to apply special tasks for this signal such as "GPS Longitude", "GPS Latitude", "UTC hour", "Audio mono" and more.

- **NoValue / DefaultValue**

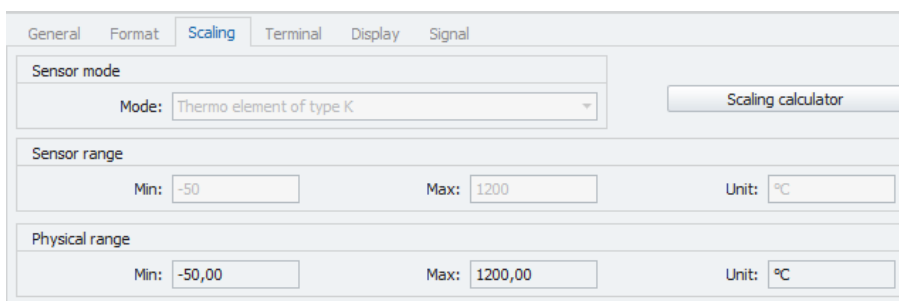
This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a "Input" channel or "Output" channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The "Scaling calculator" allows for more refined scaling options with a large range of functions. For details on how to use the "Scaling calculator" please refer to the [IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling"](#).



- **Sensor Mode**

The sensor mode tells the type of signal. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

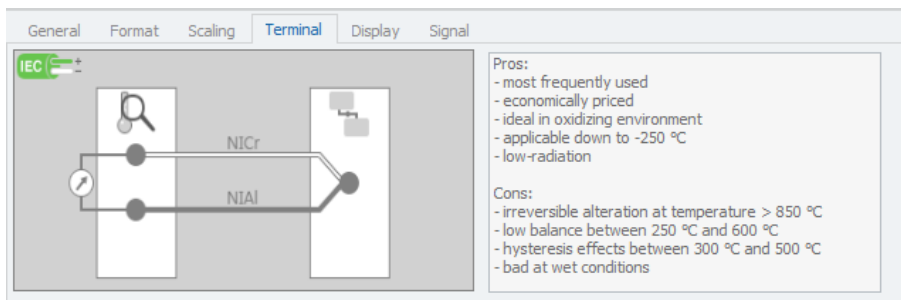
Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to "scale" your signal and also define the unit to use. For more refined scaling please use the "Scaling calculator" and refer to the IPEmotion Documentation - Section 3.4.5 "Channel configuration and scaling".

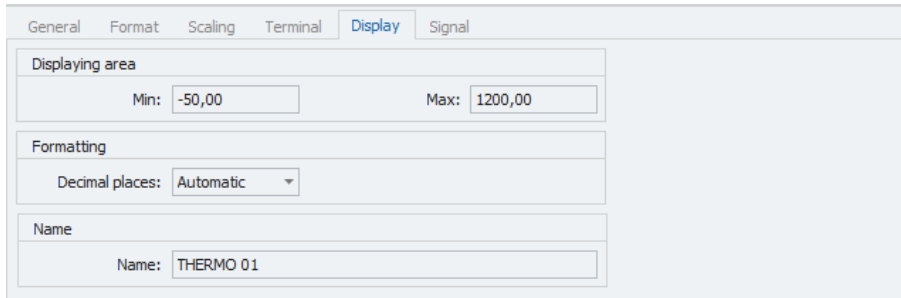
## Terminal

Additional information regarding the kind of thermoelement, that is being used.



## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

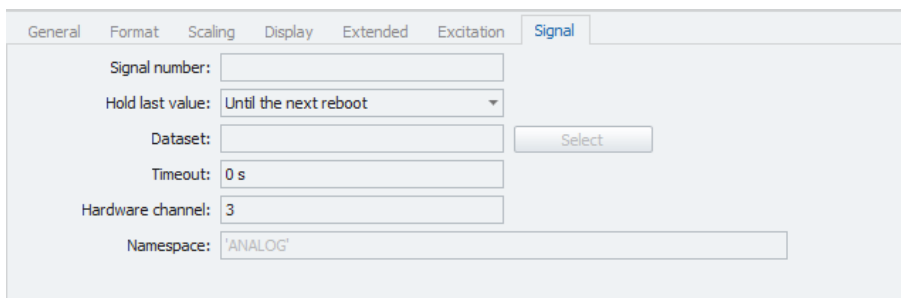
The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Signal

This tab allows you to define signal settings.



- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their “Signal numbers”.

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

Changes and errors excepted.

- **Dataset**

If the setting "Hold last value" has been set to "Until end of dataset", you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to "NaN (Not a Number)" and will be displayed as "-" in a display.

- **Hardware channel**

Assign the corresponding digital hardware channel number from which you would like to put out data. The hardware channel number can be found out via the logger's webinterface.

- **Namespace**

The "Namespace" serves as unique identifier for the signal inside the logger.



## 8.22 Internal signals

Internal signals are values that are not fed into the logger from the outside (CAN, GPS...), but are generated within the logger and provide information about internal system states.

Internal signals are largely treated in the same manner as a CAN signal. They can be recorded over time, classed or processed; they can generate alarms or be displayed. Only they can't be directly stored in traces, since the values, with the exception of the bus statistics, are not, as required, in the form of bus messages.

The internal signals are divided into three groups and the details on each group can be found in the respective section:

- Run state (→[8.22.4](#))
- System info (→[8.22.5](#))
- Time (→[8.22.6](#))

### 8.22.1 Storage method

In order to store internal signals use one of the following signal storage methods.

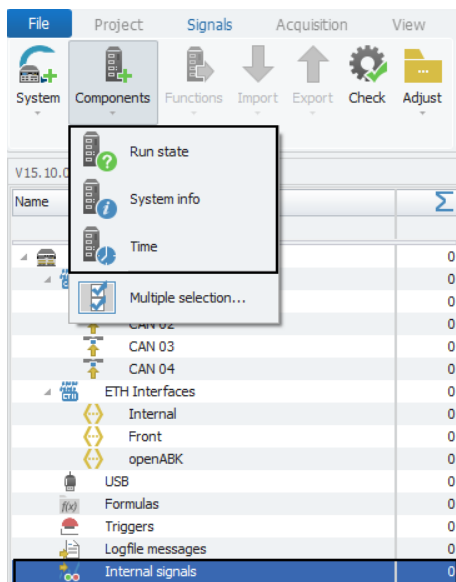
- ATFX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

### 8.22.2 Accessing internal signals

“Internal signals” do not need to be imported, as they are continuously produced by the logger itself. They only need to be made accessible and can then be activated for to be used liked regular signals in further processing, for example as triggers or in formulas.

To access “Internal signals” select the tree element “Internal signals”, click the “Components” button in the Ribbon and then choose, which of the three categories of internal signals you wish to access.

If you wish to acces two or all three categories, you can either access them one by one or through the button **“Multiple selection...”**.



### 8.22.3 Internal signals properties

#### 8.22.3.1 Tree elements for Internal signals

Each category of “Internal signals” that has been accessed, will appear in the tree as a child element to the tree element “Internal signals” with its respective name. In the right part of the tree it will also show, how many signals have been activated for further processing.

Name	Σ
ARCOS 1.5 - dataLog	0
CAN Interfaces	0
CAN 01	0
CAN 02	0
CAN 03	0
CAN 04	0
Data transfer	0
Connections	0
USB	0
Transfer events	0
Datasets	0
Dataset	0
ETH Interfaces	0
Front	0
Internal	0
openABK	0
Formulas	0
Internal signals	0
Run state	0
System info	0
Time	0

#### 8.22.3.2 Grid area for Internal signals

In the “Grid area” you will be presented with an overview of the available “Internal signals”. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

#### 8.22.3.3 Details area for Internal signals

The Details area shows settings either for the selected tree element (“Internal signals”, “Run state”, “System info” or “Time”) or the selected signal in the grid area. In case a tree element is selected, the details area will only show the “General” tab. Please refer to (→4.2.2).

In case a signal is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

## General

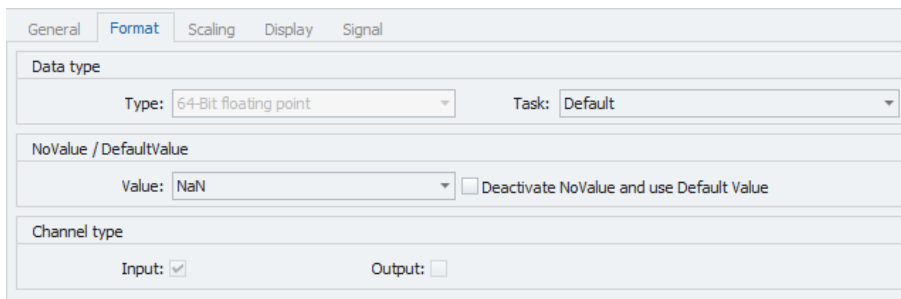
This tab allows you to activate or deactivate the signal by ticking/unticking the checkbox and thus to make it available for internal Recording and further use (e.g. triggers, formulas, display,...)

It also allows you to give a user specific name to your signal if wished and add an additional description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed. The "Sampling rate" allows you to set, how frequently a signal should be requested.



## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case "64-Bit floating point") and allows you to apply special tasks for this signal such as "GPS Longitude", "GPS Latitude", "UTC hour", "Audio mono" and more.

- **NoValue / DefaultValue**

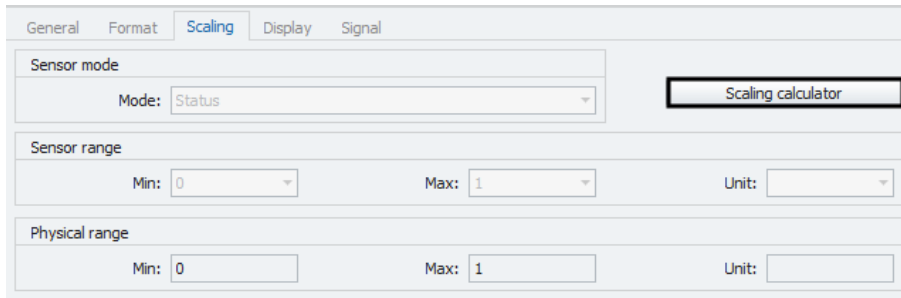
This field allows you to define the value that will be shown if a signal value is read as invalid.

- **Channel type**

This field tells you whether you are dealing with a "Input" channel or "Output" channel.

## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.



- **Sensor Mode**

The sensor mode tells the type of signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of signal it is dealing with.

- **Sensor Range**

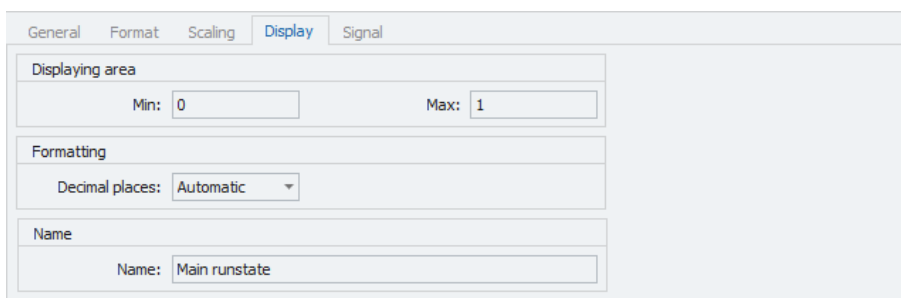
Shows the raw value range of the signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”.

## Display

This tab allows you to define what information about the current signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

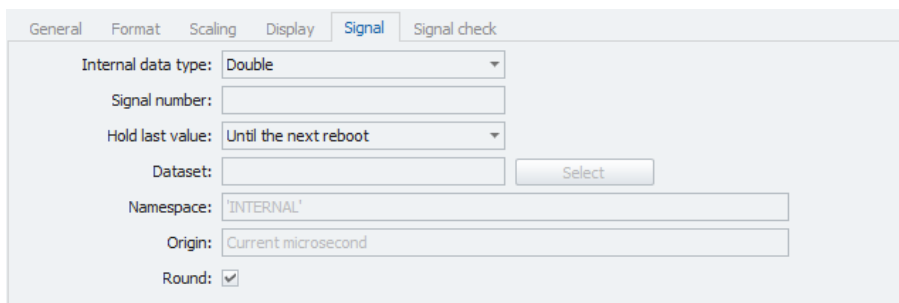
The dropdown menu "**Decimal places**" allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

## Signal

This tab allows you to define signal settings.



- **Internal data type**

Assign an internal data type to the signal. Available data types are "Double" and "String".

- **Signal number**

Assign a number to the current signal. This way you will later be able to sort the signals in the grid according to their "Signal numbers".

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting "Hold last value" has been set to "Until end of dataset", you may here select the dataset, to which this setting will refer.

- **Timeout**

Specify the timeout period for the current signal. If the data source doesn't send data for the specified time period, the value of the signal is set to "NaN (Not a Number)" and will be displayed as "-" in a display.

- **Namespace**

The "Namespace" serves as unique identifier for the signal inside the logger.

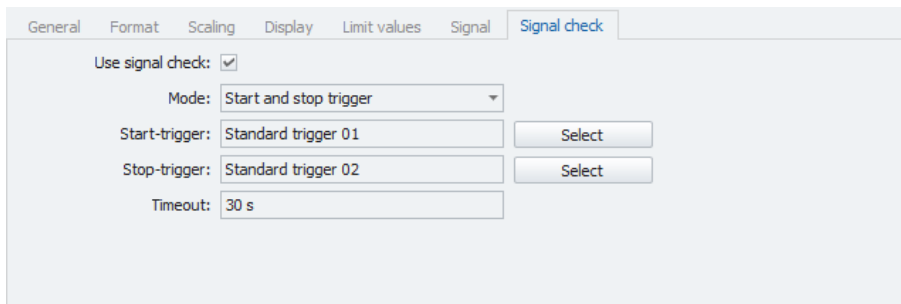
- **Round (Time status signals only)**

Signals showing time in split units will be round to full units.

### Signal check

This tab allows to apply the global signal check settings to this signal. In order to do so, global signal check settings have to be defined, please refer to (→[4.2.2](#)).

The parameters defined by the global signal check settings may be manually overridden for each signal.



General   Format   Scaling   Display   Limit values   Signal   **Signal check**

Use signal check:

Mode: Start and stop trigger

Start-trigger: Standard trigger 01  

Stop-trigger: Standard trigger 02  

Timeout: 30 s

#### 8.22.4 Run state

This group of internal signals provides information about the states of external signals and switches, as well as about the wake conditions of the buses that started the logger.

These internal channels can be used as a supplement to the standard logger responses, for example, to start or stop defined methods on signal states. They can also set off warning messages and alarms.

If a bus wake condition is defined, it is output even if this condition was not used to start the current measurement. This makes it possible, during operation, to check when wake conditions are met and trigger methods/messages/alarms.

The following table gives an overview and explanation of “Run state” signals.

Signal	Meaning	Possible Values
Main runstate	Main runstate	0 = off 1 = on
Active wake up condition	Type of wake condition that currently started the logger	1 = A-switch 2 = Cl. 15 3 = WoX
State system switch		0 = off 1 = on
State Clamp 15		0 = off 1 = on
State wake up condition		0 = not met 1 = met
Measuring time delay	Delay between the current logger time and the timestamp of the data being processed.	Value in (ms)
Error	Tells you whether an error has occurred in your present configuration.	0 = no error 1 = error
Error missing channel		0 = no error 1 = error
Error initialized interface		0 = no error 1 = error
Error missing interface		0 = no error 1 = error
Error persistencefile		0 = no error 1 = error
Dataset size	Tells you the current size of the dataset.	Value in MB
Measuring time delay	Tells you the current measurement delay.	Value in ms
File I/O buffer time delay	Tells you the current delay for writing data from the buffer to disk.	Value in ms



### 8.22.5 System info

These signals provide information on CPU status and allocation of both working memory (mem) and hard drive (disk). Sizes are specified in absolute numbers (in MB or in °C) or as relative values (in %).

This category also contains the loggers frontnumber.

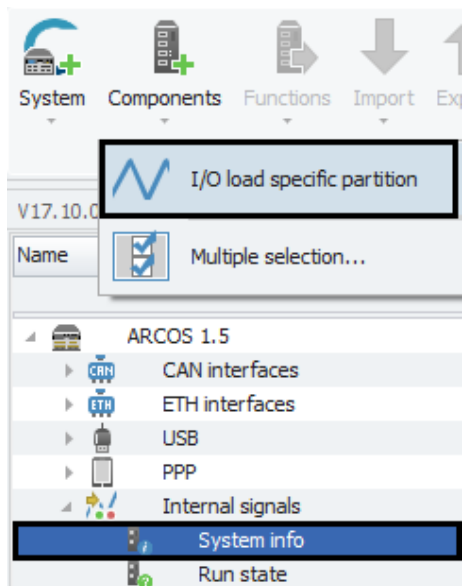
The following table gives an overview and explanation of “System info” signals.

Signal	Meaning	Unit
Total memory	total working memory (RAM)	(MByte)
Free memory	free working memory	(%)
Used memory	used working memory	(%)
Total disk space	total disk space	(MByte)
Free disk space	free disk space	(%)
Used disk space	used disk space	(%)
Total external disk space	total disk space	(MByte)
Free external disk space	free disk space	(%)
Used external disk space	used disk space	(%)
CPU load	processor load	(%)
CPU temperature	processor temperature	(°C)
Frontnumber	Tells you the loggers unique frontnumber	Value
I/O load internal disk	Tells you the writing load to the internal disk	(%)
I/O load external disk	Tells you the writing load to the external disk	(%)
I/O load specific partition	Tells you the writing load to a specific partition	(%)



### Regarding the signal "I/O load specific partition"

As it is possible to define multiple custom partitions, the "I/O load specific partition" can also exist in multiple instances. To add a new instance of the signal select the "System info" tree element, click the "Components" button in the Ribbon and choose "I/O load specific partition".



In order to then assign the correct partition to the signal, navigate to the "Partition" tab in the signal's details area and fill in the partition's name under "Name".



### 8.22.6 Time

These signals provide information on time and date. They allow you to record a timeline in order to trace the occurrence of events in the logger.

The following table gives an overview and explanation of "Time" signals.

Subtype	Meaning	Unit
Time since beginning of day	Time since 00:00:00 h UTC	(s)
Time since firmware start	Time since the firmware was started (values <0 represent values during booting phase)	(s)
Time since START MEAS	Time since measurement has started	(s)
Current date	Current date in the format ddmmyy	Value
Current year	Current year in the format yyyy	Value
Current month	Current month in the format mm	Value
Current day	Current day in the format dd	Value
Current time	Current time in the format hhmmss	Value
Current hour	Current hour in the format hh	(h)
Current minute	Current minute in the format mm	(min)
Current second	Current second in the format ss	(s)
Current microseconds	Current microseconds	( $\mu$ s)

## 8.23 Formulas

### 8.23.1 Formulas (Calculated signals)

A formula is a calculated signal. Apart from the fact of being calculated it has largely the same properties as physical signals and can be modified or used for further processing in the same way. It allows you, to combine already existing signals into a new signal. Therefore the existence of physical signals is requirement in order for formulas to function properly.

All the signals/quantities, that have been individually defined can be further processed in formulas. This also applies to internal signals.

A formula is a one-line term made up of operators and functions that are applied to numbers and signals, which will have a calculated signal as a result. Calculated signals can in turn be used as normal signals in another formula. Not only numbers and signals can serve as function arguments, but also the name of any already defined formula.

The operator priorities used by the formula interpreter are listed in the Table In addition, the interpreter observes the "multiplication/division before addition/subtraction" rule. When uncertain about priorities, you should use brackets.

Bear in mind that signal names and operators, in particular, are case-sensitive. Throughout the signals, whether they be bus, internal or computed, no name may be used twice. The functions can have multiple applications within a formula – with the exception of integration (INT\_STD), differential (DIFF) and moving average (MEAN). **So each formula may apply only once the function INT\_STD, DIFF or MEAN.**

Operator	Meaning	Priority
AND	Logical and	1
OR	Logical or	1
XOR	Logical, exclusive or	1
<=	Less than or equal to	2
>=	Greater than or equal to	2
<>	Not equal to	2
=	Equal to (comparison)	2
>	Greater than	2
<	Less than	2
+	Addition	3
-	Subtraction	3
*	Multiplication	4
/	Division	4
^	Exponentiation (2^3 =>"raise 2 to the power of 3")	5

#### Special features of moving averages

Unlike the other formulas, the moving average (MEAN) uses not only the current value but also a certain number of previous values. This number is defined by the parameter "Delay depth (values)". The number is theoretically unlimited, but in practice it is limited by the working memory and processing speed.

Assuming the number is =100, then this computes the average over the last 100 samples. At the next sampling instance, the oldest of the 100 values is dropped and the current value is included. At start-up no samples are available for review, so the buffer is still empty. The buffer is filled up with the first valid value and then moves through the sample values. In the event of a signal timeout ("Not a Number", abbreviated NaN, or as a value, also called NoValue), this review is interrupted. As long as the value of the signal is NaN, the moving average is also equal to NoValue (processing a NoValue yields another NoValue). Once the signal goes back to a valid value, the buffer – just like at the start of measurement – is filled with the first valid value, thus resuming processing of a valid average. In a triggered timelog, the moving average is based on the continuous, untriggered value stream. Once the start trigger is activated, the moving average buffer is filled, according to the depth of averaging, with the values gathered prior to the trigger time. This means the moving average at trigger time is computed from values obtained before the start trigger.

### Special features of "logical" operators

Basic rules for applying the logical operators from are:

- Note upper/lower case: always lowercase logical operators
- Always bracket operands, if you use logical operators. Example: ('signal1') or ('signal2').

The results derived by an operator fundamentally depends on the type of data to which it is applied.

The formula "(('signal1') and ('signal2'))" applies the logical operator "and" to two channels with rational numbers. In this case, the integer parts of the respective channels are linked bit-wise. So if:

Signal1= 6 (dec) or 110 (bin)

Signal2= 3 (dec) or 011 (bin)

Then "(('signal1') and ('signal2'))" yields the result:

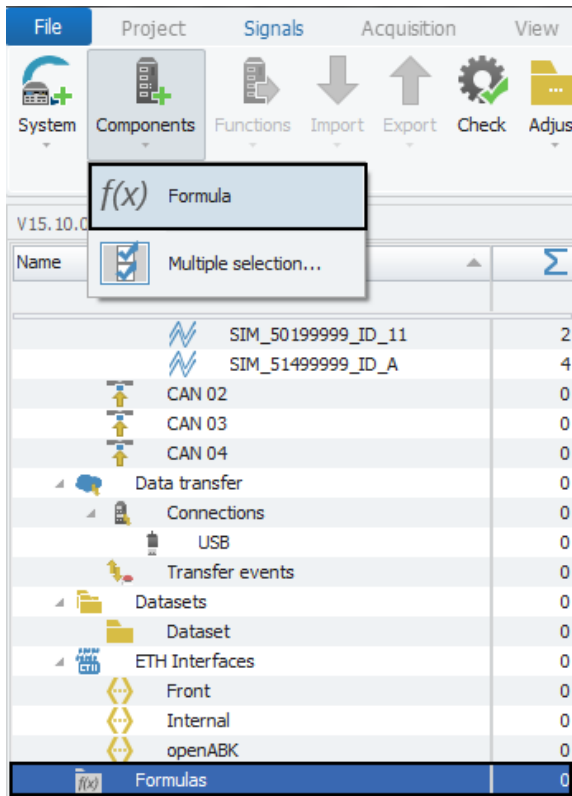
$$6 \text{ and } 3 = 2 \text{ (dec) or } 110 \text{ and } 011 = 10 \text{ (bin)}$$

The formula "(('signal1'>5) and ('signal2'>2))" applies the logical operator "and" to the binary intermediate results of two channels with rational numbers. The intermediate results (signal1>5) and (signal2>2) yield binary "0" or "1", depending on the value of the channels. The link "and" merely links these values and can have only "0" or "1" as a result.

**8.23.1.1 Storage method** In order to store calculated signals use one of the following signal storage methods.

- AFX (→ [14.5](#))
- MDF 4.0 (→ [14.6](#))
- MDF 4.1 (→ [14.7](#))

**8.23.1.2 Adding a formula** To add a new formula select the tree element “Formulas”, click on the “Components” button in the Ribbon and then choose “Formula”



This will add a generic formula, which in the beginning will have the value “1”. Instructions on how to work with that formula and modify it, will be explained in the section “Calculation” (→8.23.1.4).

**8.23.1.3 Grid area for formulas** All the formulas, that have been added to your system so far, will be presented in an overview in the grid area. Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate	Bus type	Factor	Offset	Signalnumber
Formula 01	<input checked="" type="checkbox"/>				-1,797693...	1,7976931...	1 Hz				
Formula 02	<input checked="" type="checkbox"/>		-1,797693...	1,797693...	-1,797693...	1,7976931...	1 Hz	NONE	1	0	
Formula 03	<input checked="" type="checkbox"/>		-1,797693...	1,797693...	-1,797693...	1,7976931...	1 Hz	NONE	1	0	
Formula 04	<input checked="" type="checkbox"/>		-1,797693...	1,797693...	-1,797693...	1,7976931...	1 Hz	NONE	1	0	

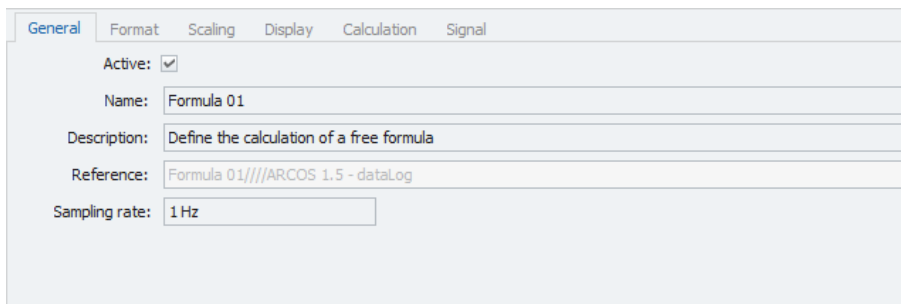
### 8.23.1.4 Details area for formulas

The Details area shows settings either for the tree element “Formulas” or for a single formula/signal which has been selected in the grid area. In case a tree element is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

In case a single formula/signal is selected in the grid area, the details area will contain additional tabs which will be explained in the following.

#### General

This tab provides general settings for the selected formula/signal.



General	Format	Scaling	Display	Calculation	Signal
Active:	<input checked="" type="checkbox"/>				
Name:	Formula 01				
Description:	Define the calculation of a free formula				
Reference:	Formula 01////ARCOS 1.5 - dataLog				
Sampling rate:	1 Hz				

- **Active**  
Allows you to activate or deactivate the selected formula/signal.
- **Name**  
Give a user-defined Name to the selected formula/signal.
- **Description**  
Give a user-defined description to the selected formula/signal.
- **Reference**  
This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

- **Sampling rate**

This field allows you to set the formula/signal's processing rate.



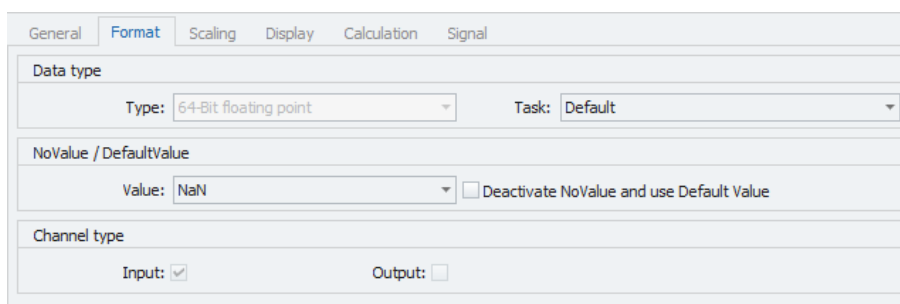
Formulas are executed in cycles. The processing frequency can be defined for each formula. This rate can be set independently of the recording rate. The computation process is applied to the values of the input quantities actually registered at the time of the processing phase. If the processing rate is higher than the signal rate of an input quantity, then the last value of the variable is used again. If the processing rate is lower, then some of the values of the quantity are omitted. (An analogous procedure applies when the recording rate is higher or lower than the processing rate.)

Particularly with such channel processing as min, max, sum or avg, periodic processing means that results are derived for the current sampling instance and not over the entire channel. If, for example, the minimum over an entire channel is required, the minimum/maximum value recording must be used.

Processing a timeout value ( Not a Number , NaN) also yields a NaN value.

## Format

This tab contains information and options regarding file format, tasks and Channel type.



- **Data type**

This field tells you the type of data (in this case "64-Bit floating point") and allows you to apply special tasks for this formula/signal such as "GPS Longitude", "GPS Latitude", "UTC hour", "Audio mono" and more.

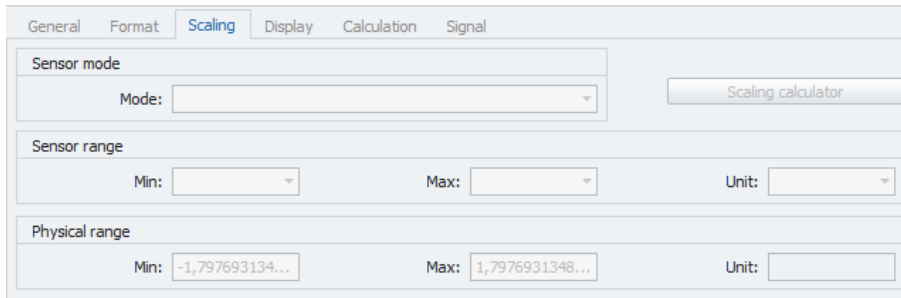
- **NoValue / DefaultValue**

This field allows you to define the value that will be shown if a formula/signal value is read as invalid.



## Scaling

The fields accessible directly through the tab allow for basic scaling operations to convert analog measurement in engineering units. The “**Scaling calculator**” allows for more refined scaling options with a large range of functions. For details on how to use the “Scaling calculator” please refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.



- **Sensor Mode**

The sensor mode tells the type of formula/signal. It can be of different types such as “Status”, “Voltage”, “Frequency” or others. It cannot be changed and serves for IPEmotion to know what kind of formula/signal it is dealing with.

- **Sensor Range**

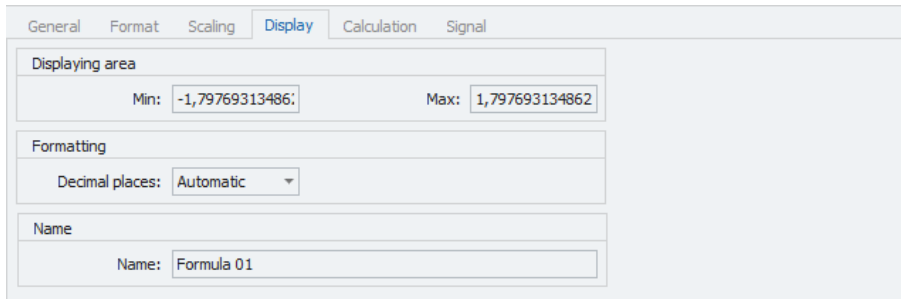
Shows the raw value range of the formula/signal.

- **Physical Range**

Allows you to set a range to which you would like to “scale” your formula/signal and also define the unit to use. For more refined scaling please use the “Scaling calculator” and refer to the **IPEmotion Documentation - Section 3.4.5 “Channel configuration and scaling”**.

## Display

This tab allows you to define what information about the current formula/signal will be shown on a display if one is connected.



- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the "Physical range" from the "Scaling" tab.

- **Formatting**

The dropdown menu "Decimal places" allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

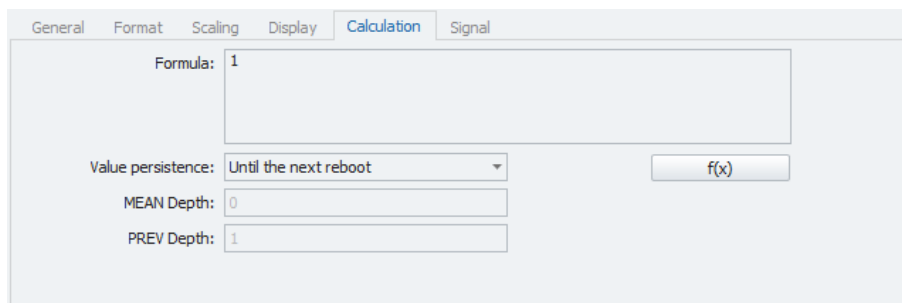
## Calculation

This tab provides the main functionality for working with formulas. Here you can define the functions and operators of your formula and what numbers and/or signals they should be applied to.

The operator priorities used by the formula interpreter are listed in the Table In addition, the interpreter observes the "multiplication/division before addition/subtraction" rule. When uncertain about priorities, you should use brackets.

Bear in mind that signal names and operators, in particular, are case-sensitive. Throughout the signals, whether they be bus, internal or computed, no name may be used twice. The functions can have multiple applications within a formula – with the exception of integration (INT\_STD), differential (DIFF) and moving average (MEAN). **So each formula may apply only once the function INT\_STD, DIFF or MEAN.**

Operator	Meaning	Priority
AND	Logical and	1
OR	Logical or	1
XOR	Logical, exclusive or	1
<=	Less than or equal to	2
>=	Greater than or equal to	2
<>	Not equal to	2
=	Equal to (comparison)	2
>	Greater than	2
<	Less than	2
+	Addition	3
-	Subtraction	3
*	Multiplication	4
/	Division	4
^	Exponentiation (2^3 => "raise 2 to the power of 3")	5

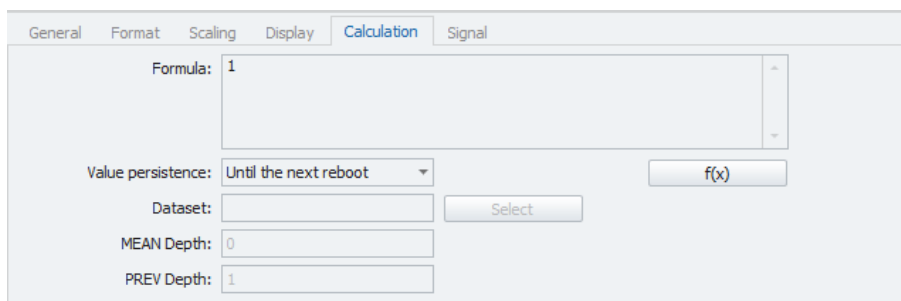


- **Formula**

This field allows you to manually enter a formula.

- **Formula editor**

Although there is the possibility to manually define formulas, most of the time it will be more convenient to define formulas using the "Formula editor".

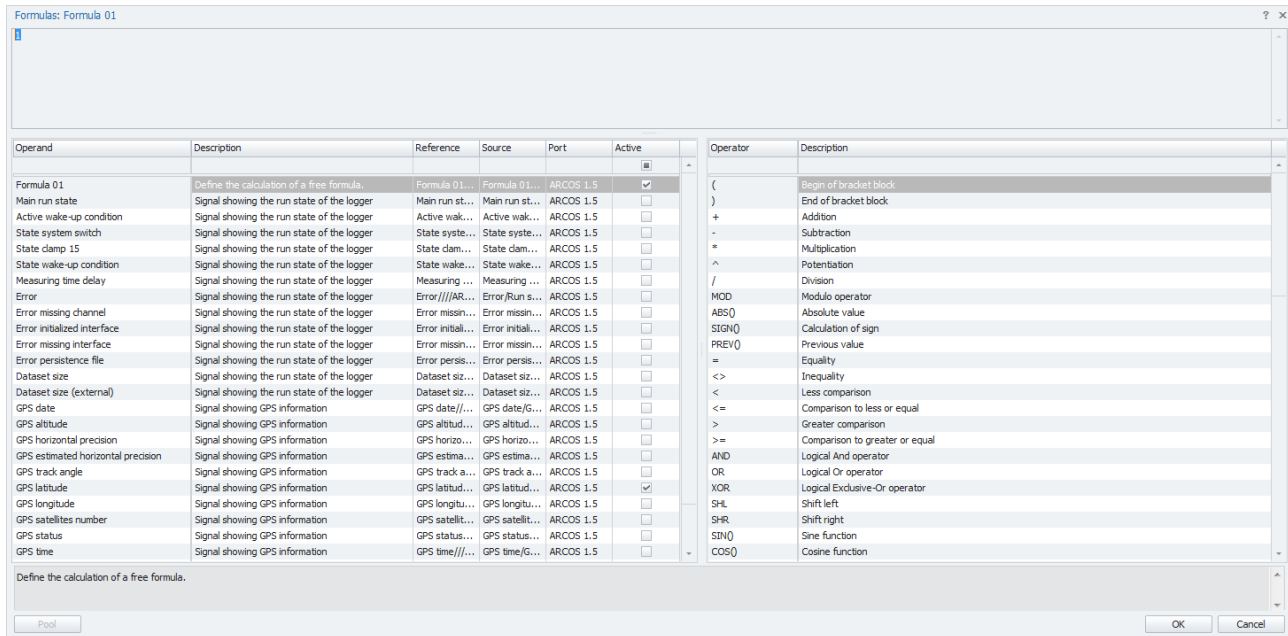


The "Formula editor" provides an overview of your current formula (the upper table), as well as the possible "Operands" (left table) and "Operators" (right table) and a short description for each item in the lower table.

You can add items to your formula either by doubleclicking on them or per drag and drop.

If you wish to delete an item from the formula, you can simply mark it with the mouse and then delete it.

If you are uncertain about the operator priorities please refer to the table above. When confirming a formula that has been defined in the "Formula editor" by clicking **OK**, the editor will automatically validate the formula's correctness. If the formula is not correct, that will result in an error message.



- **Value persistence**

Defines the persistence of the signal, so that the value of the signal is sustained even beyond logger restart.

- **Dataset**

If the setting "Value persistence" has been set to "Until end of dataset", you may here select the dataset, to which this setting will refer.

- **Moving average buffer size**

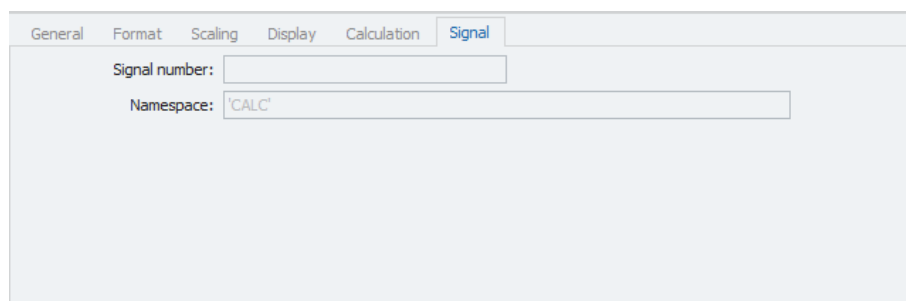
Defines buffer size of the MEAN() filter.

- **Delay buffer size**

Defines the size of the delay() buffer.

## Signal

This tab allows you to define formula/signal settings.



The screenshot shows a software interface with a tabbed menu at the top. The tabs are labeled 'General', 'Format', 'Scaling', 'Display', 'Calculation', and 'Signal'. The 'Signal' tab is currently selected and highlighted. Below the tabs, there are two input fields. The first is labeled 'Signal number:' and is empty. The second is labeled 'Namespace:' and contains the text 'CALC'.

- **Signal number**

Assign a number to the current formula/signal. This way you will later be able to sort the formulas/signals in the grid according to their "Signal numbers".

- **Namespace**

The "Namespace" serves as unique identifier for the formula/signal inside the logger.

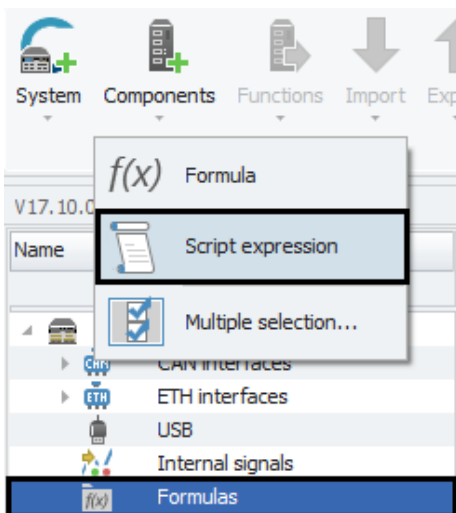
### 8.23.2 Script expressions

As of V17.10.00 the PlugIn supports, additional to regular formulas, also Script expressions as formulas. Script expressions mainly function in the same way as regular formulas, meaning they result in a calculated signal that can than be used like any other signal. For further details on settings for formulas please refer to (→[8.23.1](#)).

The main difference is the editor, that is used to compose the Script expression, which has a wider range of operators. An overview of the available operators for the “Script expression” editor can be found here (→[8.23.2.2](#)) in this section.

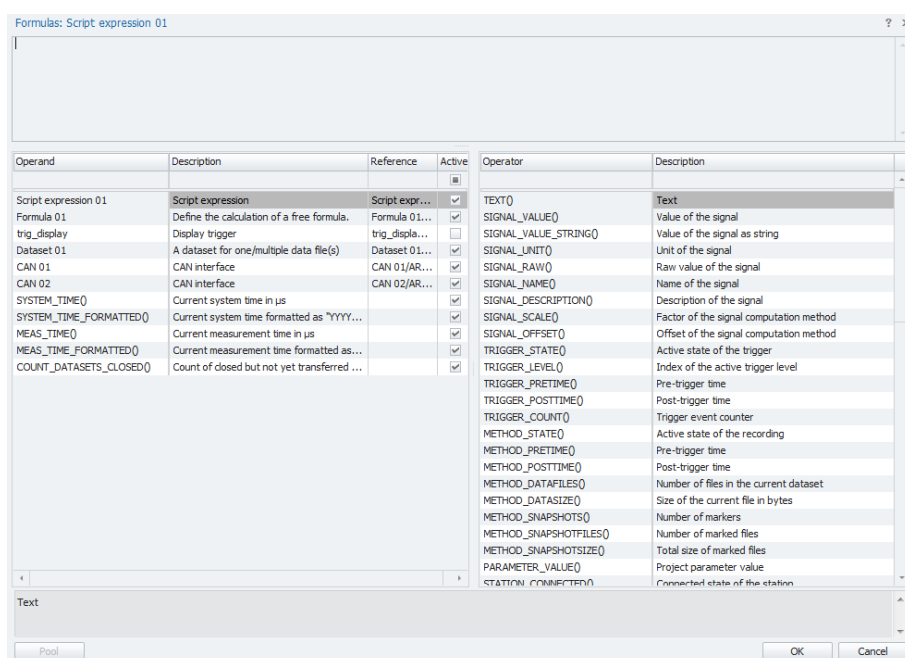
#### 8.23.2.1 Adding a Script expression

To add a “Script expression”, select the “Formulas” interface in the tree, click on the “Components” button in the Ribbon and then select “Script expression”.



### 8.23.2.2 Script expression editor

The “Script expression editor” allows you to compose dynamic messages using operands and operators. As operands all signals, triggers and methods will be available. The available operators are listed in the tables below with a short description.



#### Signal operators

Operator	Characteristics
SYSTEM_TIME()	Current time in microseconds as integer
SYSTEM_TIME_FORMATTED()	“%Y-%m-%d %H:%M:%S”
MEAS_TIME()	Current time (minus Measdelay) in microseconds as integer
MEAS_TIME_FORMATTED()	“%Y-%m-%d %H:%M:%S”
TO_CHAR(255) TO_CHAR( SIGNAL_VALUE(“signal”) ) TO_CHAR( ( 3 - SIGNAL_VALUE(“signal”) ) * 5 )	Number to ASCII character
TO_HEX( 0 ) TO_HEX( SIGNAL_VALUE(“signal”) )	Hex value as string (e.g. 0x0a)
TO_HEX_FULL( 255 ) TO_HEX_FULL( 255; 5 ) TO_HEX_FULL( SIGNAL_RAW(“signal”); SIGNAL_VALUE(“signal2”) )	Hex value as string (e.g. a)
TO_ASCII( 5; 10 ) TO_ASCII( 5; 10; 3 )	Decimal value as string
TO_INTEGER( “asdf” ) TO_INTEGER( SY- STEM_TIME_FORMATTED() )	ASCII value of the string as integer

Operator	Characteristics
COUNT_DATASETS_CLOSED()	Number of closed datasets, that have not been transferred yet
CHECK_BUS( 10; "CAN 01" ) CHECK_BUS( SIGNAL_VALUE("signal"); "LIN 05" ) CHECK_BUS( ( EXP( 13 ) / 4 ); "ETH 02" )	True when bus has no timeout
IS_VALID_SIGNAL( "name" ) IS_VALID_SIGNAL( "name" + 13 )	Reports "True" when the string is a valid signal name
IS_VALID_METHOD( "name" )	Reports "True" when the string is a valid method name
IS_VALID_DATAFILE( "name" )	Reports "True" when the string is a valid datafile name
COS(255) COS( SIGNAL_VALUE("signal" ) ) COS( ( 7 + COUNT_DATASETS_CLOSED() ) * 12 )	
SIN(1)	
TAN(0)	
ACOS(255)	
ASIN(0)	
ATAN(1)	
COSH(2)	
SINH(3)	
TANH(4)	
EXP(5)	
LOG(6)	
LN(7)	
POW(8;9)	Second operand is exponent
SQRT(10)	
CEIL(11.1234)	
FLOOR(12.9876)	
LN(7)	
ABS(-13)	
CRC8_SUM( 1; 2 ) CRC8_SUM( 1; 2; 3; 4; 5; 6; 7; 8 ) CRC8_SUM( 1; 2; 3; ... 255 )	Calculation of the CRC8 sum



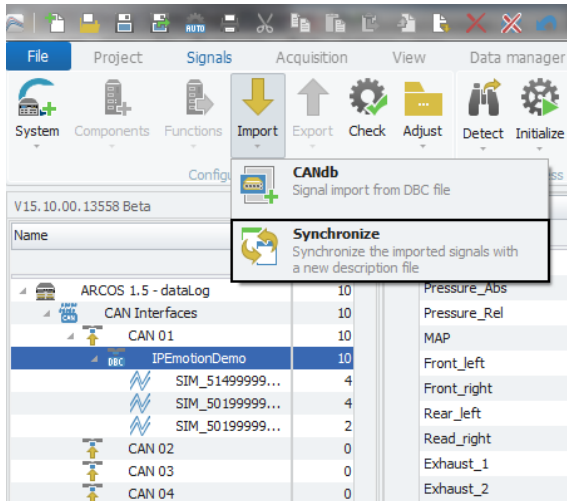
Operator	Characteristics
FRONTNUMBER() DATASET_ID_GLOBAL( "name" ) DATASET_ID_LOCAL( "name" ) SPECIAL_VIN() SPECIAL_ODO()	e.g. specialtagvalue("fn")  The following tags are supported: <ul style="list-style-type: none"> <li>• \$(fn)/\$(sn): The logger's frontnumber</li> <li>• \$(vin): Value of the signal with special role "vin" (special=vin)</li> <li>• \$(odo): Value of the signal with special role "odo" (special=odo, supported from 2016.06)</li> </ul> The following tags are supported from V 2017.10 inside of a dataset and refer to the respective dataset: <ul style="list-style-type: none"> <li>• \$(datasetglobalid): global index (unique for all datasets)</li> <li>• \$(datasetlocalid): local index (unique for datasets within the same namespace)</li> </ul>
SIGNAL_VALUE("signal")	
SIGNAL_VALUE_STRING("signal")	as of V17.10.00
SIGNAL_UNIT("signal")	
SIGNAL_RAW("signal")	
SIGNAL_NAME("signal")	
SIGNAL_DESCRIPTION("signal")	
SIGNAL_SCALE("signal")	
SIGNAL_OFFSET("signal")	
STATION_CONNECTED( "station" )	as of V17.10.00
STATION_VERSIONCHECK_RESULT( "station" )	as of V17.10.00
STATION_VERSIONCHECK_VERSION( "station" )	as of V17.10.00
DAQ_CONFIGURED( "station"; "daqlist" )	the specified DAQ list has to belong to the specified station
DAQ_STARTED( "station"; "daqlist" )	the specified DAQ list has to belong to the specified station
PARTITION_TOTAL( "internal" )	in MB
PARTITION_TOTAL( "customStorageLabel" )	as of V17.10.00
PARTITION_USED_MB( "partition" )	in MB as of V17.10.00
PARTITION_USED_PERCENT( "partition" )	in percent as of V17.10.00
TRIGGER_STATE()	Reports the state of the trigger

Operator	Characteristics
TRIGGER_LEVEL()	Index of the active trigger level
TRIGGER_PRETIME()	Pre-trigger time
TRIGGER_POSTTIME()	Post-trigger time
TRIGGER_COUNT()	Trigger event counter
METHOD_STATE()	Reports the state of the method
METHOD_PRETIME()	Pre-trigger time
METHOD_POSTTIME()	Post-trigger time
METHOD_DATAFILES()	Number fo files in the current dataset
METHOD_DATASIZE()	Size of the current file in bytes
METHOD_SNAPSHOTS()	Number of markers
METHOD_SNAPSHOTFILES()	Number of marked files
METHOD_SNAPSHOTSIZE()	Total size of marked files
DATASET_SIZE( "dataset" )	
TEXT( "hello world!" )	
SIGNAL_VALUE( "signal" ) == 5 ( 1 + 2 ) == ( 3 - 4 )	equal
1 <> 0	not equal
1 > 0	greater
1 < 0	less than
1 >= 0	not less than
1 <= 0	equal or less than
1 AND 0	AND
1 OR 0	OR
1 XOR 0	XOR
1 ORB 0	Bitwise-OR
1 XORB 0	Bitwise-XOR
1 ANDB 0	Bitwise-AND
NOTB( 1 ) NOTB( SIGNAL_VALUE( "signal" ) )	Bitwise-NOT
1 + 0	Plus
1 - 0	Minus
1 * 0	Times
1 / 0	Divide
1 MOD 0	Modulo
PARAMETER_VALUE()	Fill in any project parameter value

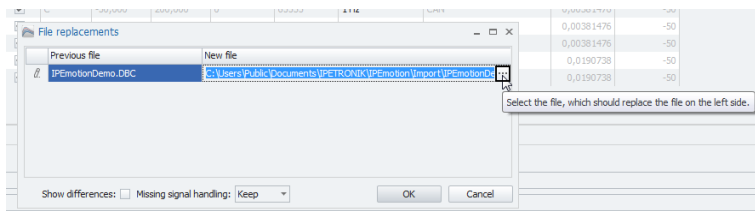
## 8.24 Synchronizing signals

This option allows you to synchronize the signals of a already imported description file with a newer or older version of the same description file.

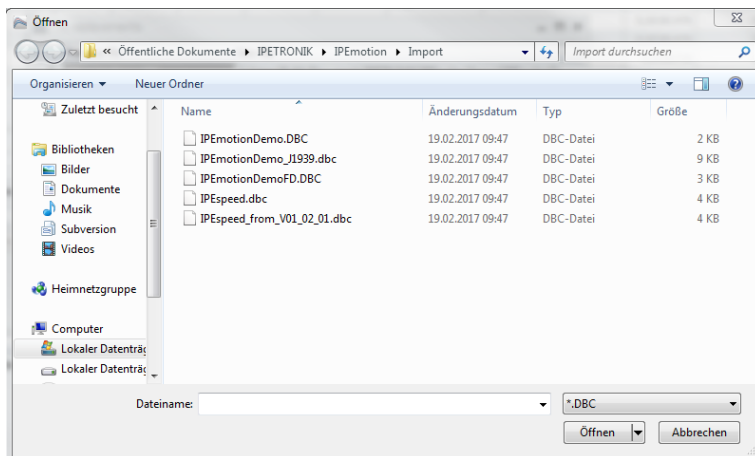
To do so, select the CAN channel or description file you wish to synchronize in the measurement task tree, click “Import” in the ribbon and then choose “Synchronize”.



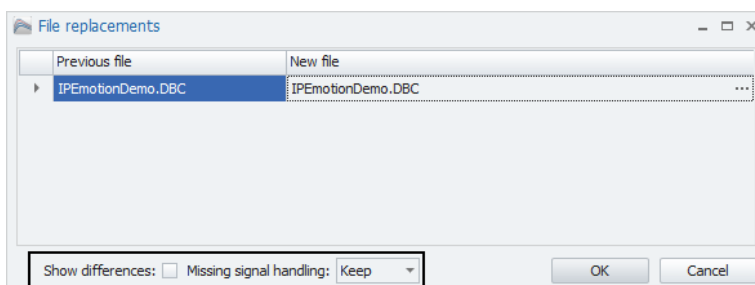
In the resulting window you will be shown which is the file previously used under “Previous file” and you will be able to choose a new file with which you would like to synchronize. To do so, click in the field for the new file and then click the button with the three dots as shown in the following figure.



In the resulting window you will be able to choose the file with which you would like to synchronize and confirm by clicking “Open”.



Before you complete the synchronization process, you may adjust two settings on how to handle differences between the two description files.



- **Show differences**

If marked active and there are differences between the current configuration and the new description file, a dialog with all these differences will be displayed before the synchronization is performed. Properties, that cannot be edited, will not be taken into account in this comparison.

- **Missing signal handling**

This dropdown menu allows you to specify how signals, that are no longer available in the new description file, should be handled.

To complete the synchronization process click "OK".

---

## 8.25 Transferring measurement tasks to the logger

## 9 Triggers

Triggers define a certain event or condition in a way, that they can trigger an action. They possess two distinguishing features compared to formulas:

- They are not a signal but a **trigger event**
- A trigger always has to be a **truth condition** and therefore can only have two possible values: true or false

Other than that, they are similar to formulas and can mainly be treated in the same way:

A trigger allows you, to combine already existing signals. Therefore the existence of physical signals is requirement in order for triggers to function properly.

A trigger is a one-line term made up of operators and operands that are applied to numbers and signals, which will have a calculated condition (trigger event) as a result.

All the signals/quantities, that have been individually defined can be further processed as operands in a trigger's formula. This also applies to internal signals.

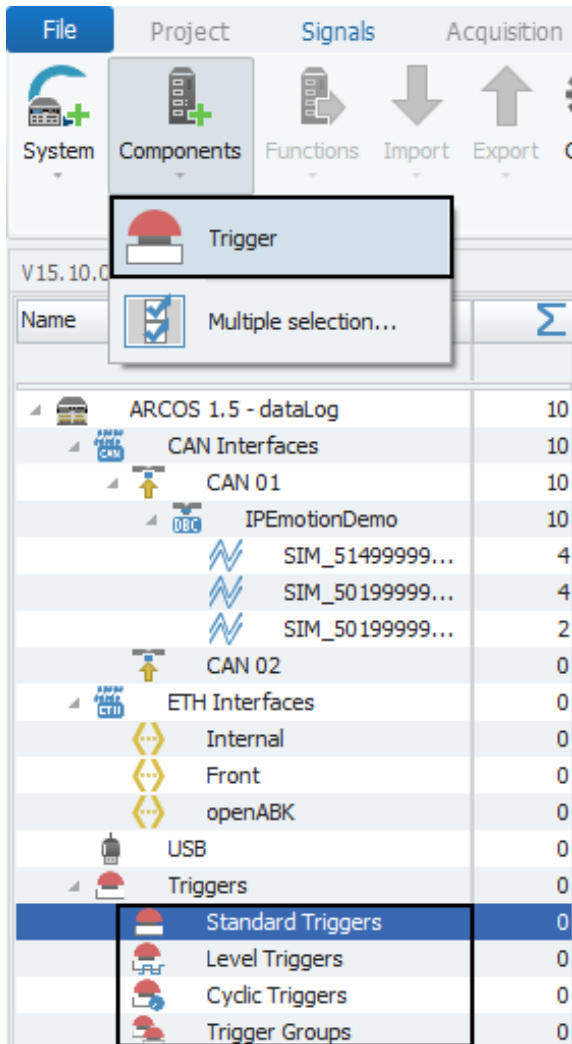
An existing trigger event can again be used as operand in a different trigger's formula.

The triggers are divided into four groups and the details on each group can be found in the respective section:

- Standard Triggers (→[9.5](#))
- Level Triggers (→[9.6](#))
- Cyclic Triggers (→[9.7](#))
- Trigger groups (→[9.8](#))

## 9.1 Adding a trigger

To add a new trigger select the desired type of trigger in the tree, click on the “**Components**” button in the Ribbon and then again choose the desired type of trigger.



This will add a generic trigger condition, which in the beginning will have the value “1”. Technically this trigger condition works the same way as a formula and instructions on how to modify it will be explained in the section “**Calculation**” (→8.23.1.4).

## 9.2 Tree elements for triggers

By default the “Measurement task tree” will contain all four categories of triggers.

Triggers	1
Standard Triggers	1
Level Triggers	0
Cyclic Triggers	0
Trigger Groups	0

Each trigger that you add to your system will be shown and accessible in the measurement task tree in its respective trigger category.

Triggers	0
Standard triggers	0
Standard trigger 01	0
Standard trigger 02	0
Level triggers	0
Level trigger 01	0
Cyclic triggers	0
Cyclic trigger 01	0
Trigger groups	0
Trigger group 01	0



### 9.3 Grid area for Triggers

The grid area for each Trigger category will present you with an overview of the triggers that have been added to your system so far.

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

The screenshot shows the CAETEC software interface. On the left is a tree view of the system configuration. The 'Triggers' section is expanded, showing 'Standard triggers' with a count of 0. Below it are 'Standard trigger 01' and 'Standard trigger 02', both with counts of 0. Other trigger categories include 'Level triggers', 'Cyclic triggers', and 'Trigger groups', all with counts of 0.

On the right, a detailed view of 'Standard trigger 01' is shown. It has a table with the following data:

Name	Active	Description
Standard trigger 01	<input checked="" type="checkbox"/>	Standard trigger
Standard trigger 02	<input checked="" type="checkbox"/>	Standard trigger

Below the table, there are fields for 'Name' (Standard trigger), 'Description', and 'Reference'.

## 9.4 Details area for Triggers

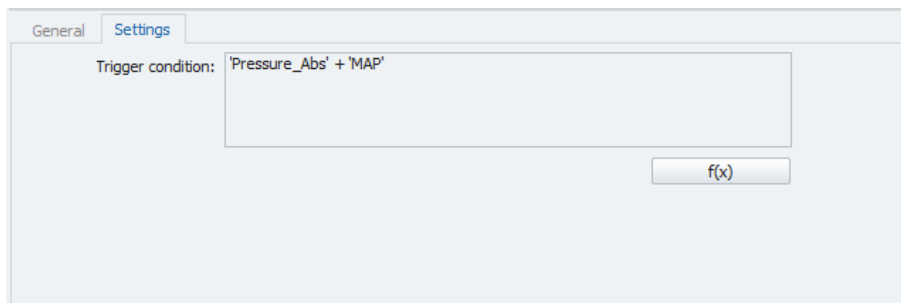
The Details area shows settings either for a selected tree element. In case the “Triggers” element or one of the four categories (“Standard Trigger”, “Level Triggers”, “Cyclic Triggers” or “Trigger groups”) is selected, the details area will only show the “General” tab. Please refer to (→[4.2.2](#)).

### Settings

In case a single trigger is selected in the tree or the grid area, the details area will additionally contain the “Settings” tab.

This tab provides the main functionality for working with Triggers. Here you can define the functions and operators for the formula of your trigger and what numbers and/or signals they should be applied to, as well as cycling rates, levels and groups.

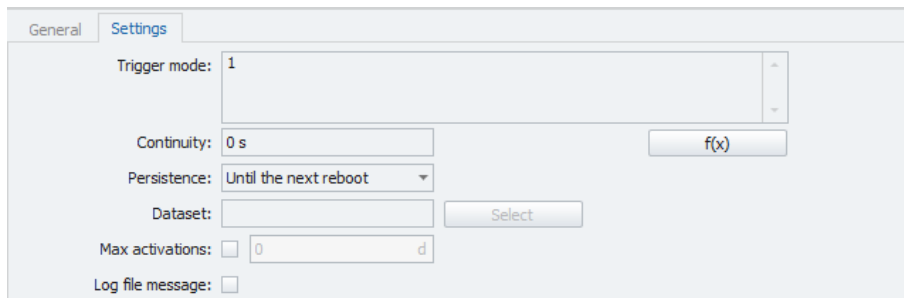
As this tab is different for each trigger category, it will be explained in the respective section for each trigger category:



- Standard Triggers (→[9.5](#))
- Level Triggers (→[9.6](#))
- Cyclic Triggers (→[9.7](#))
- Trigger groups (→[9.8](#))

## 9.5 Standard Triggers

For standard triggers, only an activation condition is specified. If the condition is met, the trigger is set; once it is no longer met, the trigger is reset. This makes the standard trigger a simple and quick way to define a trigger. In the following will be explained how work with “Standard Triggers”.



The screenshot shows the 'Settings' tab of a configuration window. It contains the following fields and controls:

- Trigger mode:** A dropdown menu showing the value '1'.
- Continuity:** A text input field containing '0 s' and a button labeled 'f(x)'.
- Persistence:** A dropdown menu showing 'Until the next reboot'.
- Dataset:** An empty text input field and a 'Select' button.
- Max activations:** A checkbox followed by a text input field containing '0' and a unit 'd'.
- Log file message:** An unchecked checkbox.

- **Trigger mode**

This field allows you to manually enter a formula for your trigger condition.

- **Formula editor**

Although there is the possibility to manually define formulas, most of the time it will be more convenient to define formulas using the “**Formula editor**”. To do so, please refer to the section “**Formula editor**” (→[8.23.1.4](#)).

- **Continuity**

Define the required minimum duration of trigger condition being met.

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.

- **max activations**

Specify a maximum number of trigger activations. Once this number has been reached, the trigger will not be activated even if the trigger condition is met.

- **Logfile**

If set, a message will be written into the logfile at trigger occurrence.

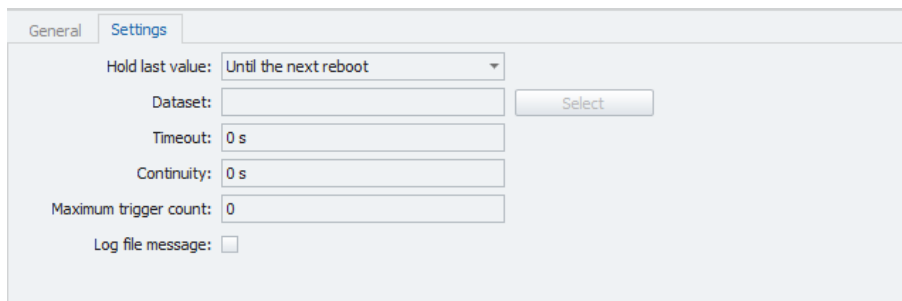
## 9.6 Level Triggers

A “Level Trigger” is an event with multiple levels of conditions, which have to be set consecutively, in a specified order. As trigger condition for one level, you can either define a single condition, which if met activates the level, and once it is no longer met, resets the trigger. Or else two conditions are defined, one to activate the level (set condition) and one to reset the trigger (reset condition). The program always checks the reset condition of the current level and the set condition of the next level, and reacts accordingly. The trigger is set once the highest defined level is reached, and remains set until the reset condition of this level is met.

In the following will be explained how to work with “Level Triggers”.

### Settings

If a “Level trigger” is selected in the tree or the grid are, the settings tab allows for settings regarding the general behaviour of a level trigger.



General Settings

Hold last value:

Dataset:

Timeout:

Continuity:

Maximum trigger count:

Log file message:

- **Hold last value**

Specify, for how long the last value of the signal will be hold.

- **Dataset**

If the setting “Hold last value” has been set to “Until end of dataset”, you may here select the dataset, to which this setting will refer.

- **Timeout**

Timeout (in ms), after which the trigger is reset, even if neither the highest level is reached nor another reset condition is met.

- **Continuity**

Minimum duration of the trigger condition (in ms) before the trigger is activated.

- **Maximum trigger count**

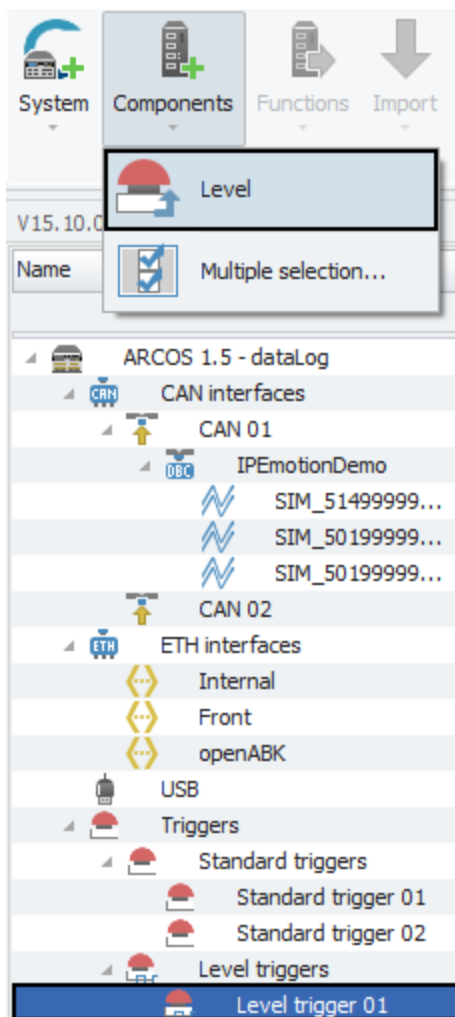
Maximum number of activations for this trigger (0 = unlimited)

- **Logfile message**

If activated, an information on this trigger event will be written to the logfile.

### Adding trigger levels

Working with a “Level trigger”, allows you to add additional trigger levels, for which you can later define trigger conditions and settings. To add a trigger level, select the desired “Level trigger” in the tree, click the “Components” button of the Ribbon and choose “Level”.



An overview of all Levels of a “Level trigger” will be presented in the grid area of the respective “Level trigger”.

### Level xx

The settings for each separate level of a “Level trigger” can be accessed by selecting the desired Level in the grid area and then navigating to the “Settings” tab in the details area.

- **Priority**

The priority defines in which order the single levels of a “Level trigger” have to be met.

V15.10.00.15774 Beta

Name	Value	Name	Active	Description	Priority
ARCOS 1.5 - dataLog	10	Level 01	<input checked="" type="checkbox"/>	Trigger level	1
CAN interfaces	10	Level 02	<input checked="" type="checkbox"/>	Trigger level	2
CAN 01	10	Level 03	<input checked="" type="checkbox"/>	Trigger level	3
CAN 02	0				
ETH interfaces	0				
Internal	0				
Front	0				
openABK	0				
USB	0				
Triggers	0				
Standard triggers	0				
Level triggers	0				
Level trigger 01	0				

Settings

Priority:

Mode:

Set condition:

Reset condition:

Timeout:

A “Level trigger” will always ascend in priority starting from “Priority 1” to “Priority 2” and so on until the final defined priority is met or a reset condition is met.

- **Mode**

Define the Mode of this trigger level.

- “Inactive” means, this trigger level is deactivated.
- “Reset is inverted set” means, this trigger level is activated and will be reset, as soon as the defined set condition is no longer met.
- “Set- and Reset-condition” means that you can define a set condition as well, as a reset condition.

- **Set condition**

Formula for the condition to activate this level.

The result of the formula must always be 0 (not met) or 1 (met), apart from that creating a formula is described in section (→8.23).

- **Reset condition**

Formula for the condition to reset the trigger from this level. Once the reset condition is met, the entire trigger is reset and must therefore run through all levels again. If reset term =1 is set for the highest level, the trigger is immediately reset, i.e. only a single trigger impulse is generated. If no reset term is specified, the end of the set condition is automatically used as the reset condition. This can be prevented by setting =0.

The result of the formula must always be 0 (not met) or 1 (met), apart from that creating a formula is described in section (→8.23).

- **Timeout**

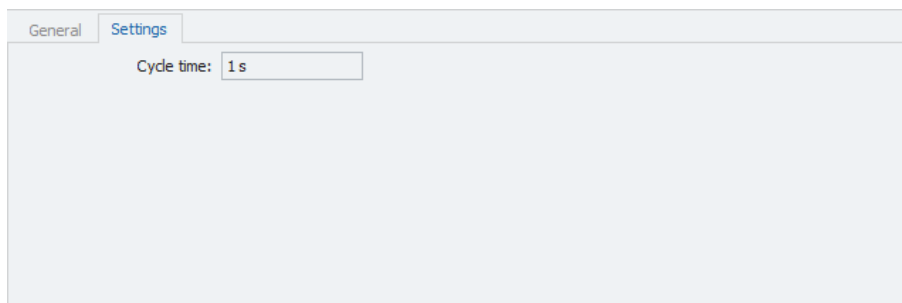
Define a level timeout after which the trigger is reset, regardless of whether the reset condition has been met or the highest level has been reached.

## 9.7 Cyclic Triggers

A “Cyclic Trigger” is an event, that is not defined by a formula. Its only condition is the cycle time to which it is set. According to this time, the trigger will be set periodically. In the following will be explained how to work with “Level Triggers”.

### Settings

The “Settings” tab in the details area of the Cyclic Trigger allows you to set the Cycle time according to which the trigger will be set periodically.



The screenshot shows a software interface with two tabs: "General" and "Settings". The "Settings" tab is active. Below the tabs, there is a label "Cycle time:" followed by a text input field containing the value "1 s".

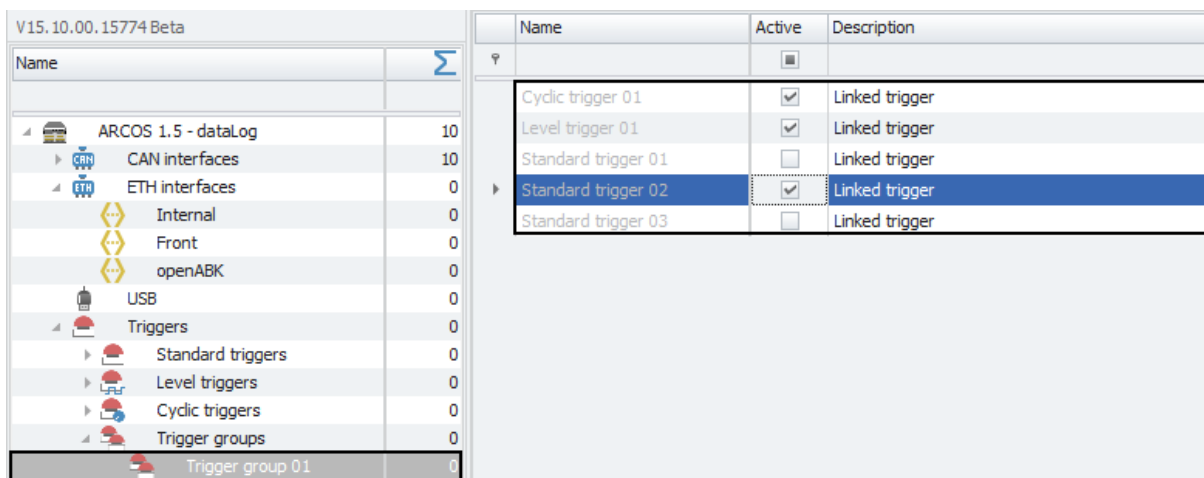
## 9.8 Trigger groups

“Trigger groups” allow you to combine two or more existing triggers and thus create a new trigger condition. Source trigger signals can be combined via “disjunction (or)” or “conjunction (and)”.

### Selecting Triggers

In order for a “Trigger group” to function you will need to select at least two existing triggers to be part of the “Trigger group”. To do so, select the “Trigger group” you wish to work with in the measurement task tree.

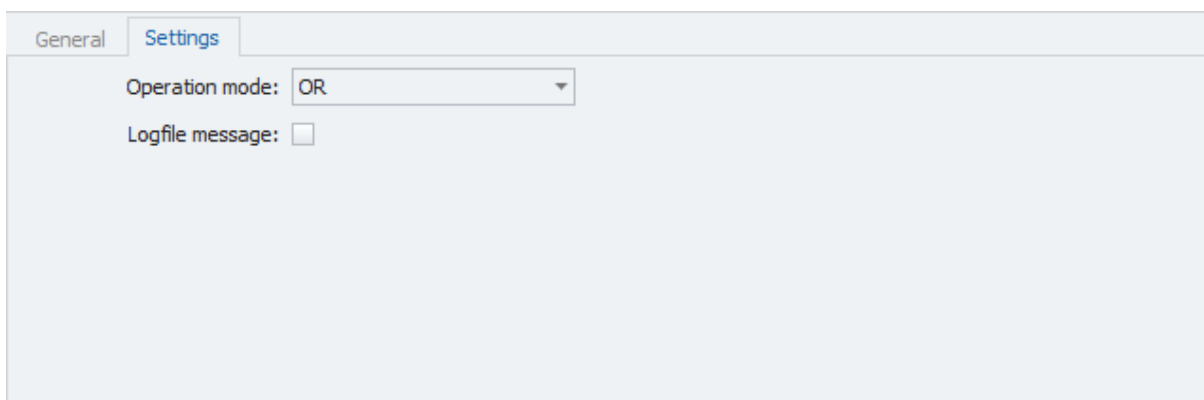
This will present you with an overview of all available triggers in the details area. Just mark active the ones you wish to add to the trigger group.



Name	Active	Description
Cyclic trigger 01	<input checked="" type="checkbox"/>	Linked trigger
Level trigger 01	<input checked="" type="checkbox"/>	Linked trigger
Standard trigger 01	<input type="checkbox"/>	Linked trigger
Standard trigger 02	<input checked="" type="checkbox"/>	Linked trigger
Standard trigger 03	<input type="checkbox"/>	Linked trigger

### Settings

en The “Settings” tab in the details area of your “Trigger group” allows you to set the operation mode of the group and whether a logfile should be created.



General Settings

Operation mode: OR

Logfile message:

- **Operation Mode**

Allows you to set the operation mode of the “Trigger group”. You may choose between “AND” or “OR”.



- **Logfile message**

If activated, an information on this trigger event will be written to the logfile.

## 10 Scripts

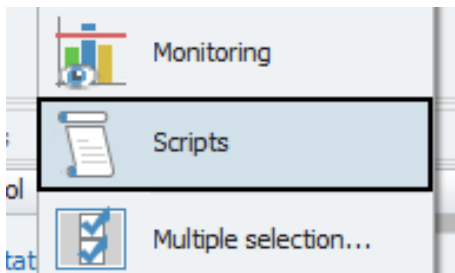
The “Scripts” interface allows you to write customized scripts for a configuration or to import existing scripts in the format “DataLog script” (\*.dls).



For detailed instruction on codewriting for scripts please refer to the CA-ETEC script manual.

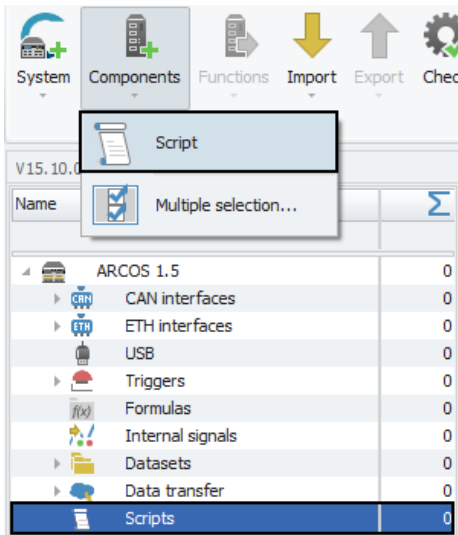
### 10.1 Adding the Scripts-Interface

In order to work with scripts, you will first need to add the “Scripts” interface to your system. To do so, select the system in the tree, click the “Components” button in the Ribbon and then choose “Scripts”.



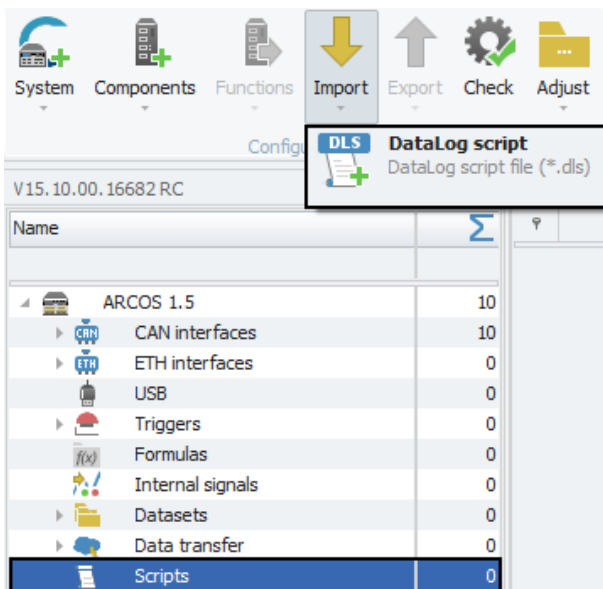
## 10.2 Adding a script

Once the “Scripts” interface has been added to your system, you can then add one or multiple scripts. To do so, select the “Scripts” interface in the tree, click the “Components” button in the Ribbon and then choose “Script”.



## 10.3 Importing a script

It is also possible to import previously written scripts of the format “DataLog script file (\*.dls)”. To do so, select the “Scripts” interface in the tree, click the “Import” button in the Ribbon and then choose “DataLog script”.



The following window lets you choose the desired script file and import it by clicking “open”.

## 10.4 Tree elements for Scripts

After having added the “Scripts” interface to your system it will appear as a tree element with the name “Scripts”.

Connections	0
USB	0
Scripts	0

Once a script has been created or imported, it will appear as a child element to the “Scripts” interface with three child elements of its own.

The tree child elements labeled “Triggers”, “Signals” and “Methods” are non-interactive elements. Their purpose is to show, how many triggers, signals or methods have been defined by the script.

Scripts	0
Script 01	0
Triggers	0
Signals	0
Methods	0

## 10.5 Grid area for Scripts (Composing a script)

The grid area for a script is where the actual composing of the script (writing the code) happens. The grid area provides an editor for writing the code or composing it out of prefabricated block of code, and also for checking for syntax errors of the code. The functions of the editor will be explained in the following.

For detailed instructions on codewriting in scripts please refer to the CAETEC script manual. This can be found in the Script editor toolbar next to the syntax check.



The functions of the editor, which can all be found in the toolbar at the top of the editor window, are divided in three groups:

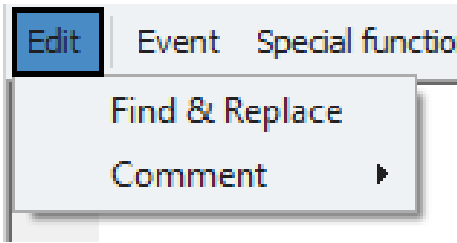
- Edit script code (→ [10.5.0.1](#))
- Prefabricated code blocks (→ [10.5.0.2](#))
- Syntax check (→ [10.5.0.3](#))



If either the “Triggers”, “Signals” or “Methods” element has been selected in the tree, the grid area will provide you with extra functionalities. These will be explained in the sections “Triggers in Scripts” (→ [10.5.1](#)), “Signals in Scripts” (→ [10.5.2](#)) and “Methods in Scripts” (→ [10.5.3](#)) of this chapter.

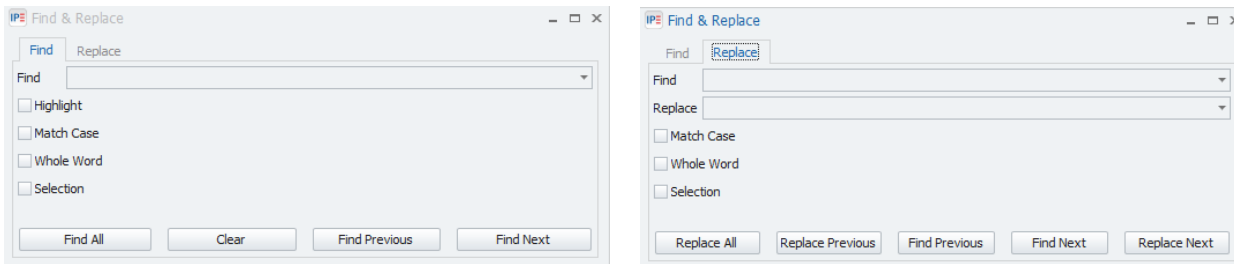
### 10.5.0.1 Edit script code

Here you can find the functions “Find & Replace” and “Comment”.



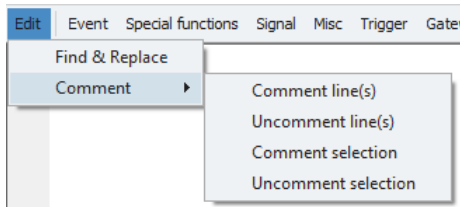
#### Find & Replace

The “Find & Replace” function allows you to search your code or a selection of it for characters/words. It also allows you to directly replace the search result.



#### Comment

The “Comment” function allows you to comment or uncomment an entire line or a selection. A line/selection, that is commented, is not part of the code.

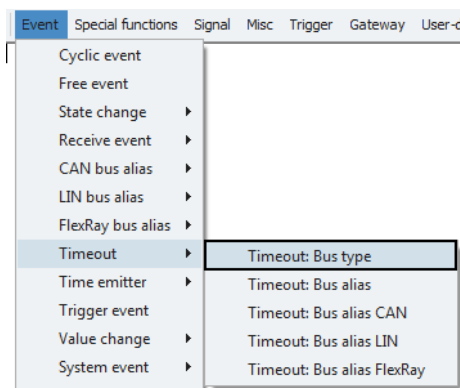


### 10.5.0.2 Prefabricated code blocks

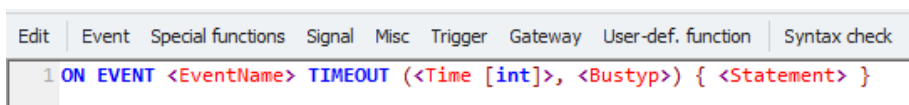
In the middle part of the editor's toolbar you will find a number of dropdown menus that offer prefabricated blocks of code. This allows you to completely compose your script of these blocks instead of writing it manually.



Any code block that is selected in one of these menus will be added to the script at the cursor's current position. For example choosing "Event->Timeout->Timeout: Bus type", like seen below



will insert the following line of code in the script.



If entering code manually, the editor offers an autocompletion of words. However, this only completes words, but does not check whether the result is an errorfree syntax.

### 10.5.0.3 Syntax check

The “Syntax check” function will check whether the syntax of the code is correct. The result of the syntax check will be presented in a pop-up window.

As you can see in the picture below, this pop-up window also offers a line jump feature. That means, if you click on an error in the pop-up window, the editor will automatically jump to the corresponding line and highlight it, so it is easier to correct any errors.

```
Edit | Event | Special functions | Signal | Misc | Trigger | Gateway | User-def. function | Syntax check
1 ON EVENT <EventName> CYCLE (<Time [int]>) { <Statement> }
2 useevent(<Name [string]>, <Value>);
3 set_bus_power(<BusType>, <ChannelNumber>, <State [bool]>);
```

Syntax check

```
Line 1: syntax error near "ON EVENT <"
Line 1: syntax error near "ON EVENT <EventName> CYCLE (<Time [int]>) { <"
Line 3: syntax error near "set_bus_power(<"
Line 3: syntax error near "set_bus_power(<BusType>, <ChannelNumber>, <State [bool]>);"
parsing aborted due to unrecoverable parsing error, script may be incomplete
```



### 10.5.1 Triggers in Scripts

When the “Triggers” tree element of a script is selected, the grid area will present you with an overview of all the triggers, that have been defined by the script. You can also see, whether a trigger is active or not, the description and type of the trigger.

Name	Active	Description	Trigger type
test_trigger1	<input checked="" type="checkbox"/>	Trigger defined by the script	Standard
test_trigger2	<input checked="" type="checkbox"/>	Trigger defined by the script	Level

### 10.5.2 Signals in Scripts

When the “Signals” tree element of a script is selected, the grid area will present you with an overview of all the Signals, that have been defined by the script. You can also see, whether a signal is active or not and relevant settings, regarding the signal, such as Phy. Min/Max or Sampling Rate. Some of these setting can be edited directly in the grid area.

Name	Active	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate
test_signal1	<input checked="" type="checkbox"/>		-1,797693...	1,797693...	-1,797693...	1,7976931...	100 Hz

Additional display settings regarding the signal can be accessed via the details area “Display” tab.

General   Format   Scaling   **Display**   Signal

---

**Displaying area**

Min:       Max:

---

**Formatting**

Decimal places:

---

**Name**

Name:

- **Displaying area**

Shows the value range which will be shown on a display. It usually should match the “Physical range” from the “Scaling” tab.

- **Formatting**

The dropdown menu “**Decimal places**” allows you to set how many decimal numbers of the value will be shown on a display.

- **Name**

Allows you to set a Name to be shown on a display.

### 10.5.3 Methods in Scripts

When the “Methods” treeelement of a script is selected, the grid area will present you with an overview of all the methods, that have been defined by the script. You can also see, whether a method is active or not, the description and type of the method as well as the file type.

Name	Active	Description	Method type	File type
test_blf	<input checked="" type="checkbox"/>	Method defined by the script	Trace	Vector BLF
test_asc	<input checked="" type="checkbox"/>	Method defined by the script	Trace	Vector ASCII
test_asc_compressed	<input checked="" type="checkbox"/>	Method defined by the script	Trace	Vector ASCII compressed
test_pcap	<input checked="" type="checkbox"/>	Method defined by the script	Trace	PCAP

## 10.6 Details area for Scripts

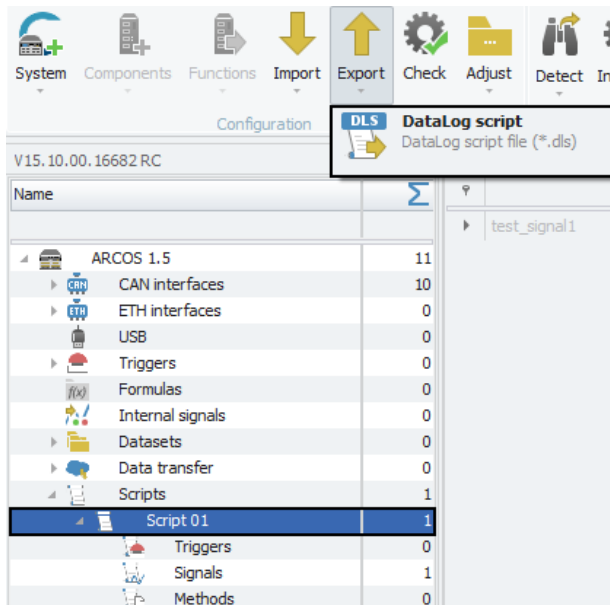
The Details area provides settings for the script, that has been selected in the measurement task tree.

### General

Please refer to (→[4.2.2](#)).

## 10.7 Exporting a script

It is also possible to export a script and thus make it available for use in other configurations. To do so, select the desired script in the measurement task tree, click the “Export” button in the Ribbon and then select “DataLog script”.



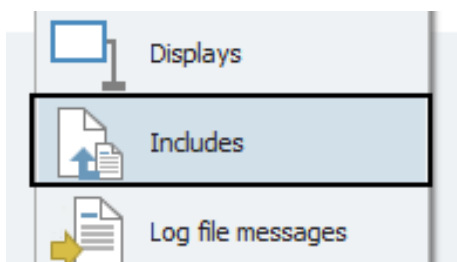
The following window lets you choose the path, where to save the script file. Confirm with “Save”.

## 11 Includes

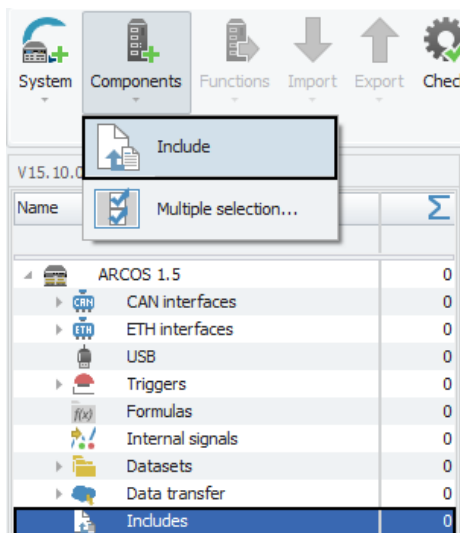
The “Includes” interface allows you to include partial configurations in your system. This can be especially helpful for components of a configuration that are likely to change over time, such as Wifi accesspoints, and are used by a large number of loggers at the same time.

### 11.1 Adding the Includes-Interface

In order to work with Includes, you will first need to add the “Includes” interface to your system. To do so, select the system in the tree, click the “Components” button in the Ribbon and then choose “Includes”.

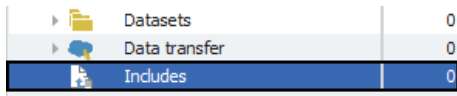


Once the “Includes” interface has been added to your system, you can then add multiple “Includes”. To do so, select the “Includes” interface in the tree, click the “Components” button in the Ribbon and then choose “Include”.



## 11.2 Tree elements for Includes

After having added the “Includes” interface to your system it will appear as a tree element with the name “Includes”.



## 11.3 Grid area for Includes

If the “Includes” interface is selected in the tree, the Grid area will present you with an overview of the Includes which have been added to your system so far.

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

Name	Active	Description	Mode	Path	Copy to dataset
Include 01	<input checked="" type="checkbox"/>	Included partial configuration	Include a specific file		<input checked="" type="checkbox"/>
Include 02	<input checked="" type="checkbox"/>	Included partial configuration	Include a specific file		<input checked="" type="checkbox"/>

## 11.4 Details area for Includes

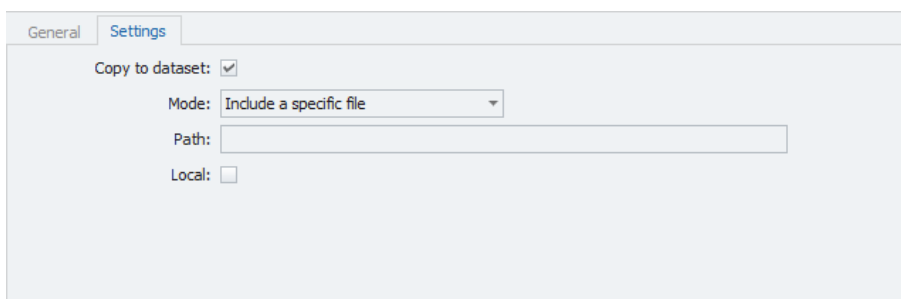
The Details area provides settings for the Include, that has been selected in the grid area.

### General

Please refer to (→4.2.2).

### Settings

This tab provides settings regarding the include file.



A screenshot of the 'Settings' tab in the details area. It shows the following settings: 'Copy to dataset' is checked; 'Mode' is set to 'Include a specific file' in a dropdown menu; 'Path' is an empty text input field; and 'Local' is unchecked.

- **Copy to dataset**

Activating this setting will include a copy of this include file in the dataset for traceability.

- **Mode**

Define, whether you want to include a specific file or the entire directory of the include path.

- **Path**

Define the cfginclude file path relative to "(cfgdir (see data transfer))/includes/". The file path must end with a slash (/). Further information regarding the path of the configuration directory (cfgdir) can be found in the chapter "Transfer connections" here (→ [15.3.1.1](#)).

- **Local**

Marks this include file as a local include that is specific to this logger. For local includes the Path must be relative to "(cfgdir (see data transfer))/(frontnumber)/includes/".

## 12 External files

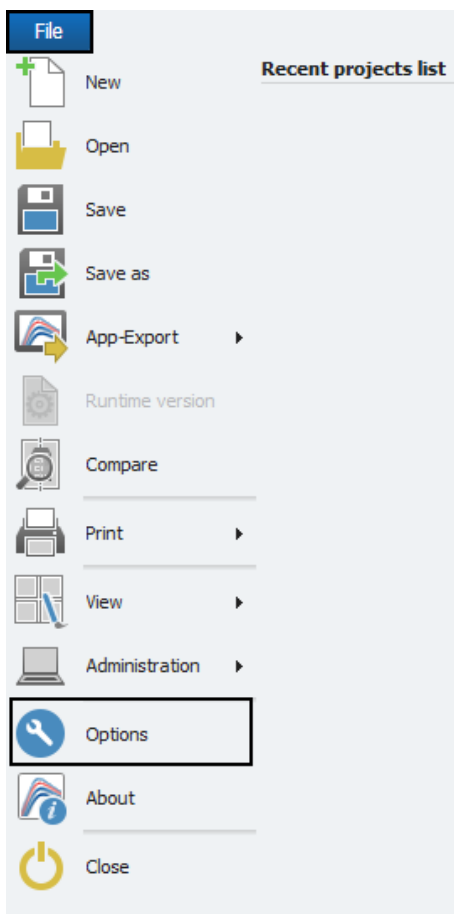
The function “External files” allows to include in the .ccmc a database file (DBC, AUTOSAR, FIBEX etc.), that has been used to import signals into a signal channel (CAN, LIN, FlexRay or ETH), as external file. When exporting your configuration, the external file will be included in the .ccmc container and can thus be made available for later use in acquisition data analysis or can be included in a dataset. All external files will be stored on the logger alongside the configuration file.

The function “External files” is available for CAN, LIN, FlexRay and ETH.

### 12.1 Automatically add external files

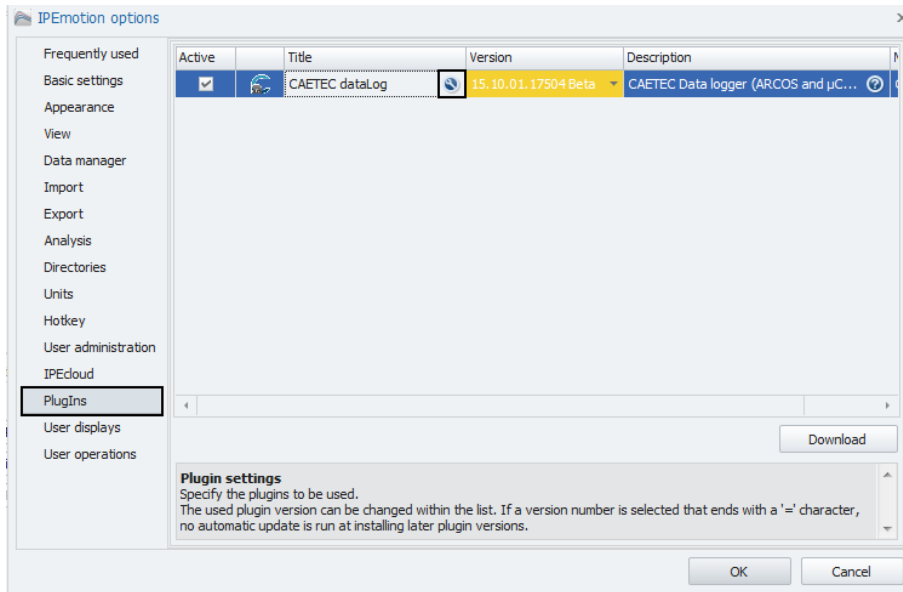
In addition to manually adding a database as external file to your configuration, it is possible to set up a configuration in such a way, that every signal database that gets imported in any one signal channel will automatically be added as external file to that signal channel.

To enable this setting, navigate to the “Options” window of IPEmotion.

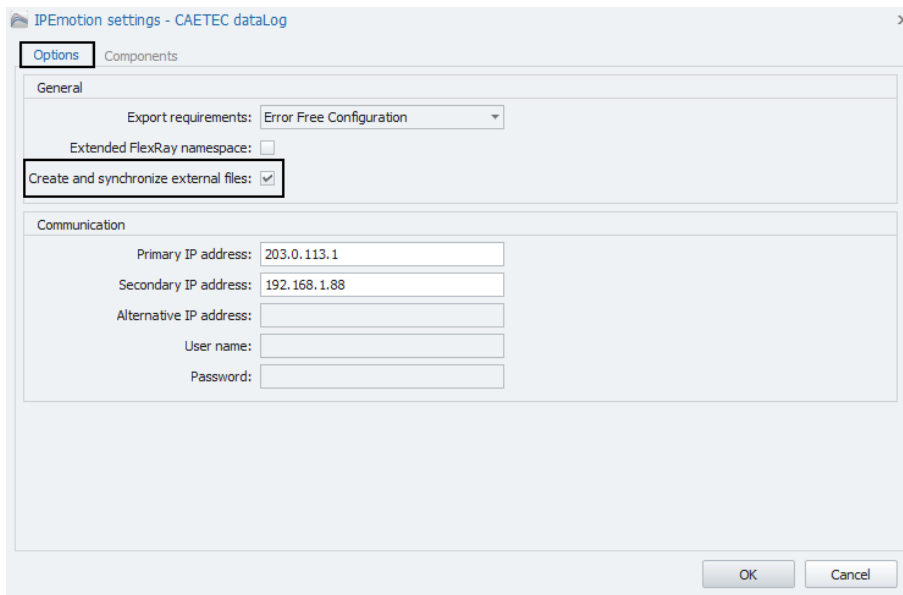


## 12.1 AUTOMATICALLY ADD EXTERNAL FILES

In the following window navigate to the “PlugIns” tab of the sidebar and access the plugin-specific settings for “CAETEC dataLog” by clicking the button with the blue screw wrench symbol right next to “CAETEC dataLog”.



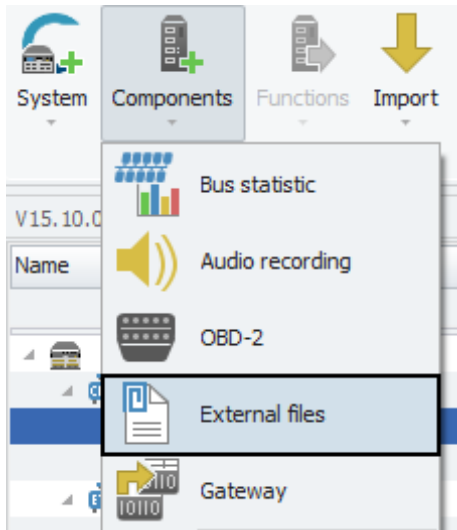
In the following window navigate to the “Options” tab and activate the tickbox for the setting “Create and synchronize external files”. In this way, whenever you import a signal database to a channel, it will be automatically be added as external file.





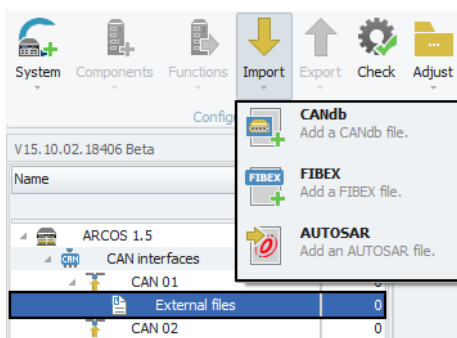
## 12.2 Adding the External files interface

To add the “External files” interface to a signal channel, select the desired channel in the measurement task tree, click the “Components” button in the Ribbon and then choose “External files”.



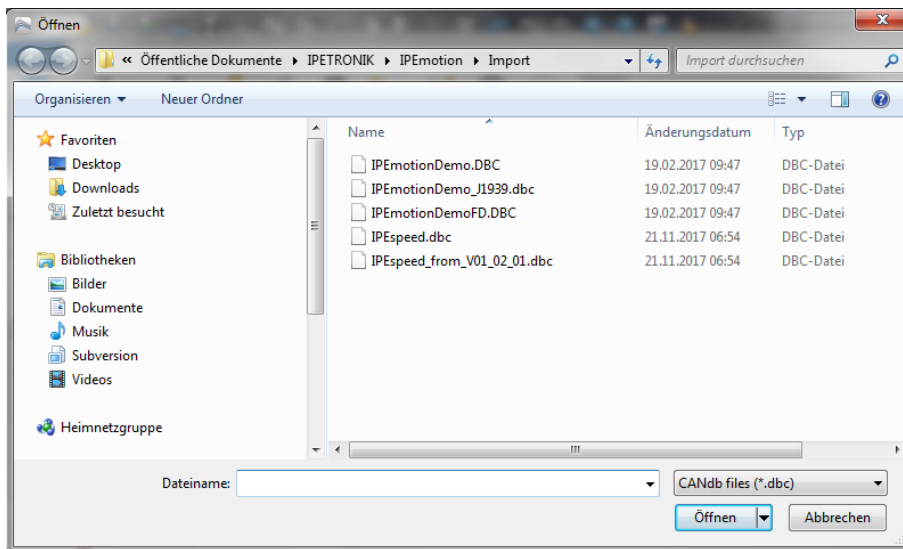
## 12.3 Adding an external file

Once the interface has been added to the signal channel, you can then add one or multiple external files. To do so, select the respective “External files” interface in the measurement task tree, click the “Import” button in the Ribbon and then choose the desired database type.



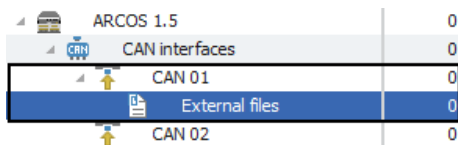
The available database types for external files depend on the type of signal channel, to which they will be linked. So while an external file for a CAN channel may be of the type CANdb, AUTOSAR or FIBEX, an external file for an Ethernet channel can only be of the type AUTOSAR or FIBEX.

In the following window you may then choose your database file and confirm with “OK”.



## 12.4 Tree elements for External files

Once the “External files” interface has been added to a signal channel it will appear as a child element to the respective channel in the measurement task tree.



## 12.5 Grid area for External files

The grid area for “External files” will provide you with an overview of the added external files for a signal channel. It also provides information on the type of database and the location of the original database file to be included.

Name	Active	Description	Location
CANdb	<input checked="" type="checkbox"/>	CANdb file	
AUTOSAR	<input checked="" type="checkbox"/>	AUTOSAR file	
FIBEX	<input checked="" type="checkbox"/>	FIBEX file	

## 12.6 Details area for External files

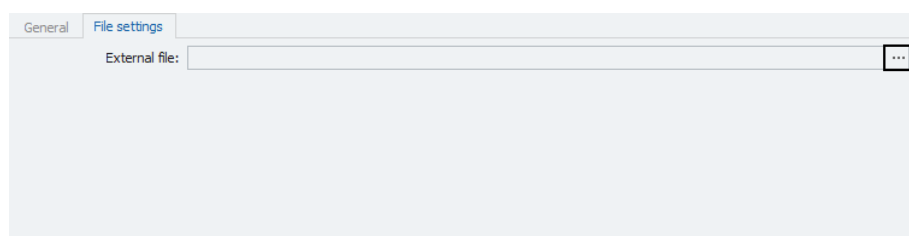
The details area for “External files” will provide you with settings regarding a single database file that has been selected in the grid area.

### General

Please refer to (→[4.2.2](#)).

### File settings

This tab allows you to select the database file, you wish to include. To do so, click the three dots at the right of the field. In the following window navigate to the location of your database file and confirm.



## 13 Surveillance

There are multiple functions for monitoring data acquisition and logger activity. This section will give an overview of these functions and explain their functionality in detail.

### 13.1 Displays

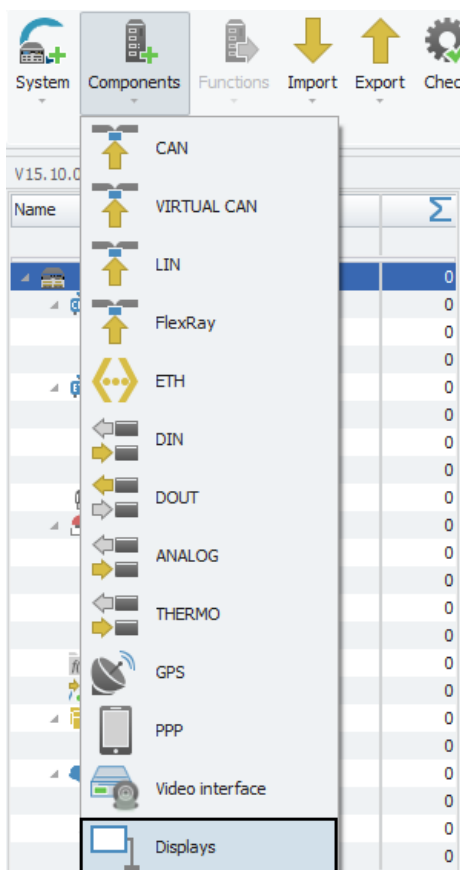
You can connect a display to your logger in order to display certain displays, events, video streams and messages. CAETEC dataloggers and the CAETEC dataLog PlugIn for IPEmotion support two types of displays:

- CAETEC display (→ [13.1.3](#))
- third party openABK display (→ [13.1.4](#))

Configuration of these displays via the plugin will be explained in detail in the following.

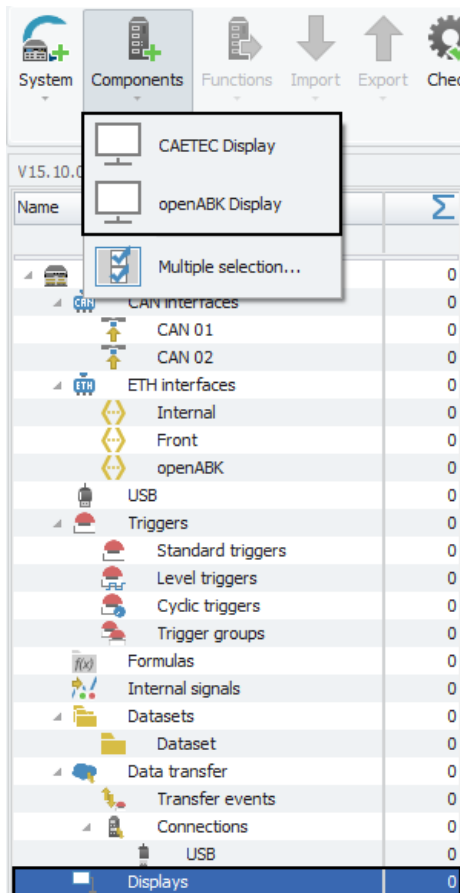
#### 13.1.1 Adding a display

In order to add a display to your system, you will first need to add the “Displays” interface. To do so, select the system in the measurement task tree (the topmost element in the tree, in this case ARCOS 1.5), click the “Components” button in the Ribbon and select “Displays”.



## 13.1 DISPLAYS

In the next step you can add one of the two available display types to your system. To do so, select the “Displays” interface in the tree, click the “Components” button in the Ribbon and then select the display type you wish to add.



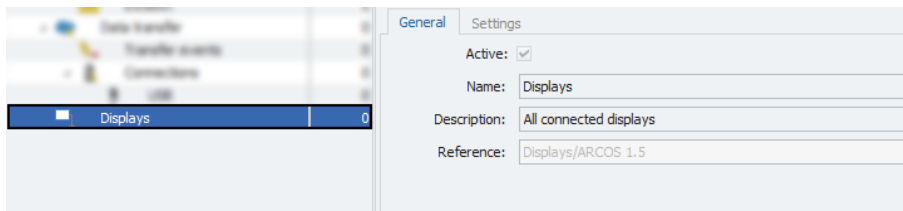
For instructions regarding the configuration of the single display types please refer to the respective sections of this manual:

- CAETEC display (→ [13.1.3](#))
- third party openABK display (→ [13.1.4](#))

For instructions regarding the configuration of the entire “Displays” interface please keep reading on below.

### 13.1.2 The “Displays” interface

In order to access the settings regarding the entire “Displays” interface including all additionally connected display types, select the tree element “Displays” and navigate to the details area. The details area contains two tabs which will be explained here.

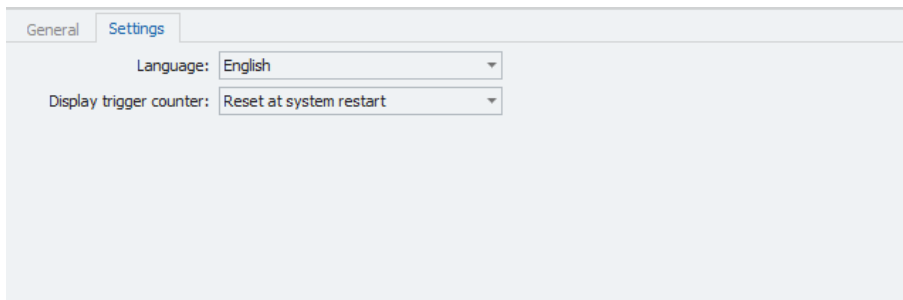


#### General

Please refer to (→[4.2.2](#)).

#### Settings

This tab contains settings specific for the “Displays” interface.



- **Language**  
Choose the standard display language.
- **Display trigger counter** Set the behaviour of the display trigger counter. There are three available settings.

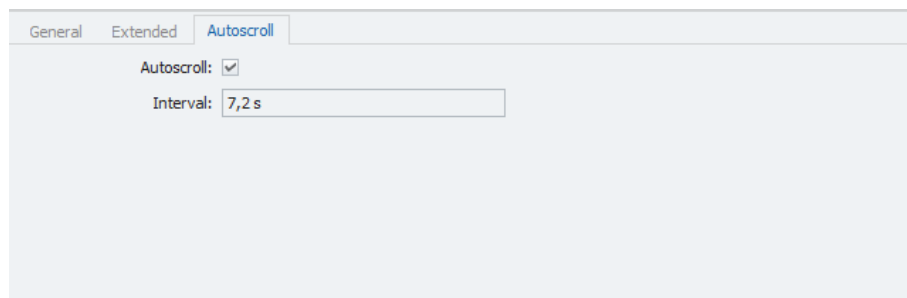
Display trigger counter setting	Characteristics
Reset at new configuration	The trigger counter will be reset, when the logger receives a new configuration.
Reset at system start	The trigger counter will be reset, whenever the logger restarts.
Never reset	The trigger counter will never be reset.

### 13.1.3 CAETEC Display-specific settings

This section will explain settings specific for the CAETEC Display. These settings are to be found in the details area of the CAETEC Display.

#### Autoscroll

Settings regarding the display's autoscroll function. Pressing the trigger button will interrupt autoscroll for a defined time interval.



The screenshot shows a settings window with three tabs: 'General', 'Extended', and 'Autoscroll'. The 'Autoscroll' tab is active. Below the tabs, there are two settings: 'Autoscroll:' with a checked checkbox, and 'Interval:' with a text input field containing '7,2 s'.

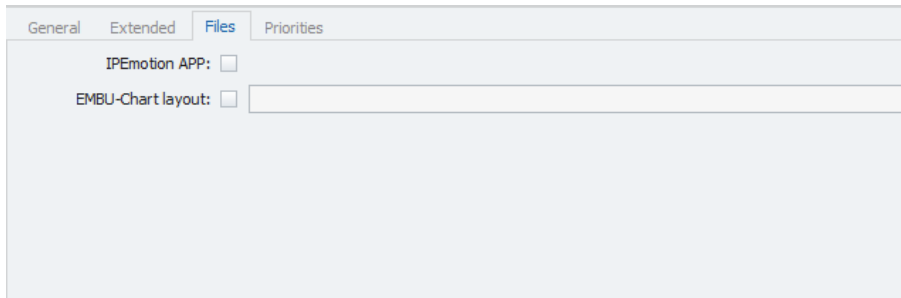
- **Autoscroll**  
Activate or deactivate the autoscroll function.
- **Interval**  
Time interval after which autoscroll will be reactivated. Setting this parameter to 0 will also deactivate the autoscroll function.

### 13.1.4 openABK Display-specific settings

This section will explain settings specific for the openABK Display. These settings are to be found in the details area of the openABK Display.

#### Files

Define which display-configuration files will be included in the loggers configuration.



- **Export IPEmotion APP**

This setting allows you to export a display-configuration for the IPEmotion APP. The display-configuration file will be included in the logger configuration. As soon as a display gets connected to the logger, the display will check whether a corresponding display-configuration file is available on the logger. If so, the display will automatically download the display-configuration and apply it.



For instructions on how to configure the IPEmotion APP as a display device, please refer to the IPEmotion documentation.

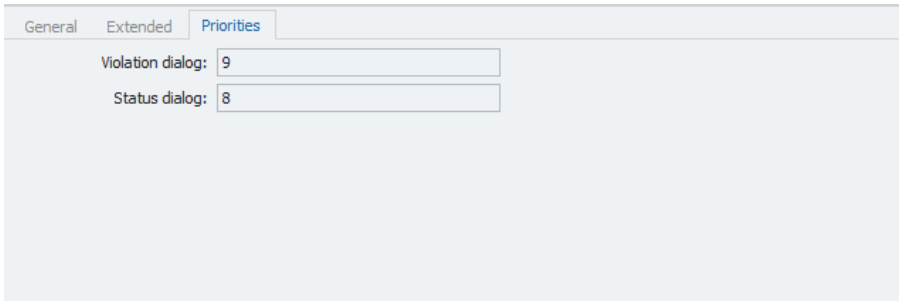
- **Export EMBU-Chart**

This setting allows you to export a display-configuration for an EMBU display device. The display-configuration file will be included in the logger configuration. As soon as a display gets connected to the logger, the display will check whether a corresponding display-configuration file is available on the logger. If so, the display will automatically download the display-configuration and apply it.



## Priorities

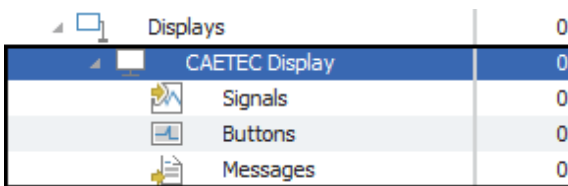
Set the priorities for the "Violation" and the "Status" dialog. The dialog with the higher priority will be shown if both dialogs appear at the same time.



### 13.1.5 General Display settings

#### 13.1.5.1 Tree elements for a Display

Adding the a to your system will add one new child element with the name "CAETEC/openABK Display" to the "Displays" interface. The "CAETEC/openABK Display" will again possess the three child elements: **Signals, Buttons, Messages**.



#### 13.1.5.2 Grid area for a Display

In the "Grid area" you will be presented with an overview of the available Signals, Buttons or Display messages, depending which tree element has been selected. Also you can find here two important functions, which are the "Column chooser" (→4.3.1) and the "Filter editor" (→4.3.2).

	Name	Active	Description
⌵		<input type="checkbox"/>	
▶	Display message 01	<input checked="" type="checkbox"/>	
	Display message 02	<input checked="" type="checkbox"/>	

	Name	Active	Description
?		<input type="checkbox"/>	
▶	Display message 01	<input checked="" type="checkbox"/>	
	Display message 02	<input checked="" type="checkbox"/>	

### 13.1.5.3 Details area for a Display

The details area for a display provides settings for the behaviour of the display.

#### General

Please refer to (→[4.2.2](#)).

#### Extended

Settings regarding the display of the selected signal.

General	Extended	Files	Priorities
Export display name: <input type="checkbox"/> Add signals automatically: <input type="checkbox"/>			

- **Export display name**

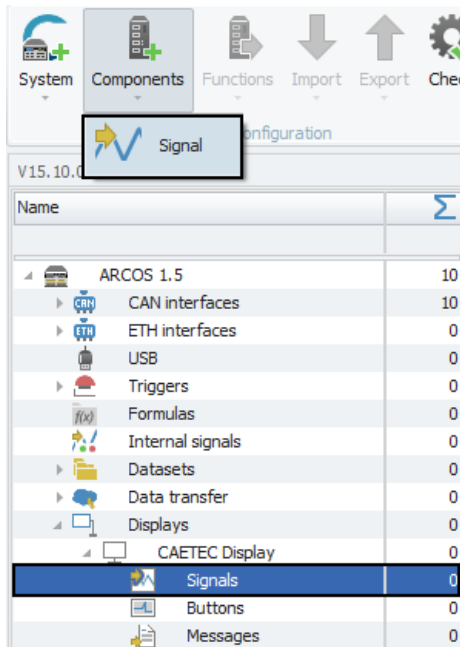
Set checkbox to force export of the parameter display name to the connected display device.

- **Add signals automatically**

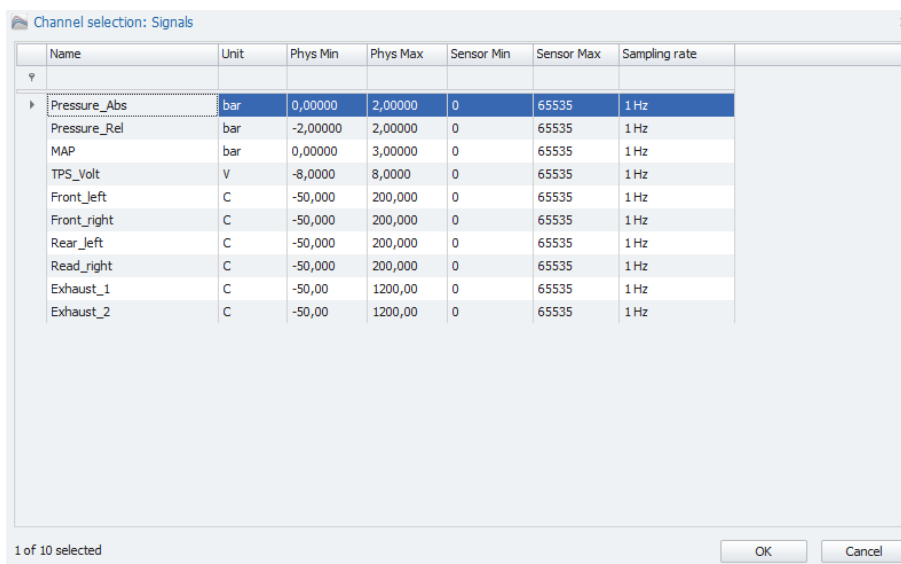
If set, all eligible signals will be automatically added to the display. Any signals that have already been added will not be removed if they become ineligible.

## 13.1.5.4 Signals for Display

The "Signals" element allows you to select one or more signals to be displayed on the connected display-device. To do so, select the "Signals" element, click on the "Components" button in the Ribbon and then choose "Signal".



In the following window you will be presented with an overview of all the available signals for display. Choose one or more signals that you wish to display and confirm with "OK".



In order to access the settings regarding the signals to be displayed please proceed as follows. Select the treeelement "Signals", then select the desired signal in the grid area and access the settings in the details area.

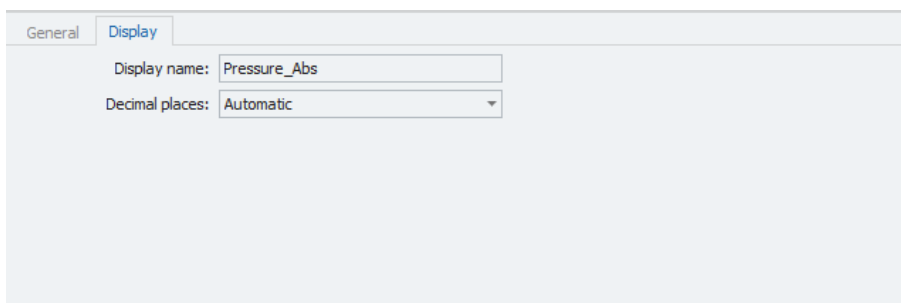
## General

The description field allows you to give a user specific description. The Reference field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.



## Display

Settings regarding the display of the selected signal.



- **Display name**  
Define a name for the selected signal, which will be displayed if the function "Export display name" has been enabled.
- **Decimal places**  
Define the count of decimal places for the signal.

### 13.1.5.5 Buttons for Display

The CAETEC and openABK Displays possess one “Trigger button”, which will be shown in the grid area when selecting the “Buttons” element in the measurement task tree. To access settings regarding this “Trigger button”, select the treeelement “Buttons” and then select the “Trigger button” in the grid area. The setting can be found in the “Trigger button’s” details area.

#### General

The name field allows you to give a user specific name of the trigger button and the description field allows you to give a user specific description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

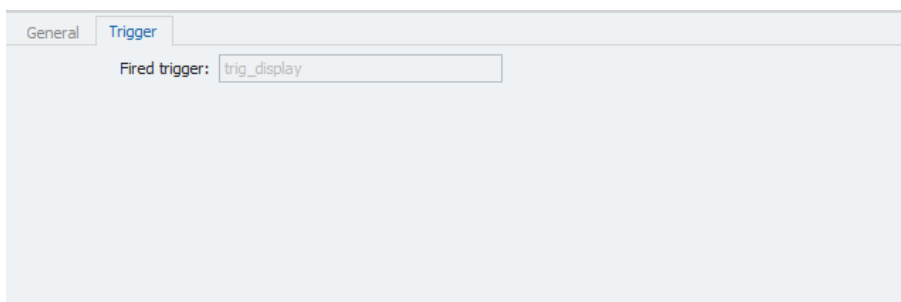


The screenshot shows a configuration window with two tabs: 'General' and 'Trigger'. The 'General' tab is active. It contains the following fields:

- Active:** A checked checkbox.
- Name:** A text input field containing 'Trigger button'.
- Description:** A text input field containing 'Trigger button'.
- Reference:** A text input field containing 'Trigger button/CAETEC Display/ARCOS 1.5'.

#### Trigger

The field “Fired trigger” tells you the trigger to be fired if the button is pressed.



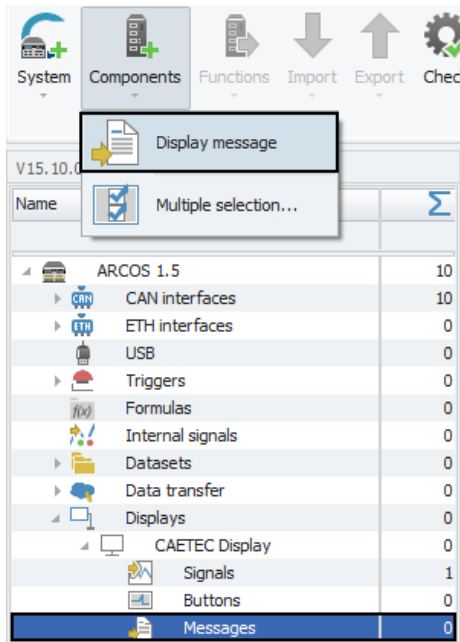
The screenshot shows the 'Trigger' tab of the configuration window. It contains one field:

- Fired trigger:** A text input field containing 'trig\_display'.

### 13.1.5.6 Messages for Display

The “Messages” element allows you to set up one or more customized “Display messages” which will be displayed when triggered. To do so, select the “Messages” element, click on the “Components” button in the Ribbon and then choose “Display message”. Multiple “Display messages” can be defined.

The content and trigger of the message may be defined in the message’s settings in the details area.



To access a message’s settings, select it in the grid area and then navigate to the details area.

#### General

This tab allows you to activate or deactivate a Display message by ticking/unticking the checkbox.

It also allows you to give a user specific name if wished and add an additional description. The Reference field serves as the unique identifier inside the measurement task tree. It cannot be changed.

The screenshot shows the 'General' settings tab for a display message. It includes the following fields:

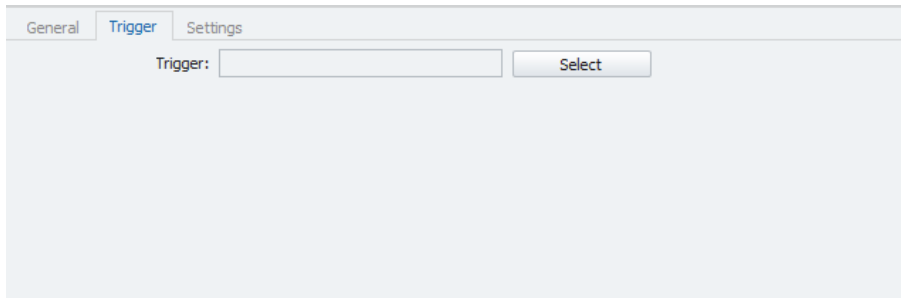
- Active:**
- Name:** Display message 02
- Description:** (empty text box)
- Reference:** Display message 02/CAETEC Display/ARCOS 1.5



In the field “Name” project parameters can be used as variables. For more information please refer to (→5.6).

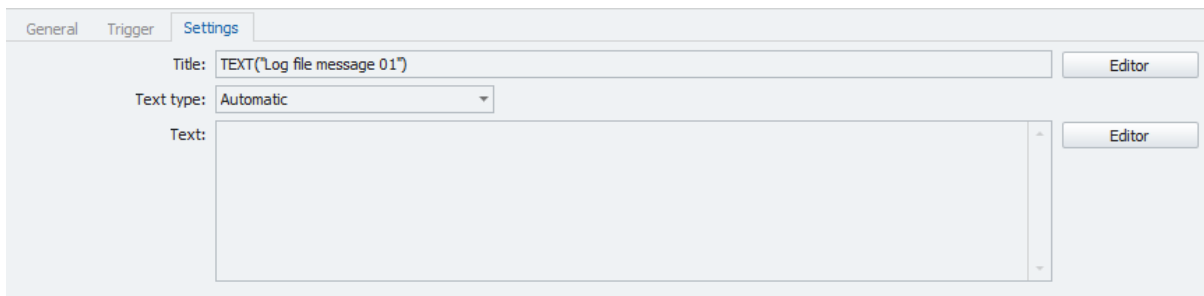
## Trigger

In this tab you can define which trigger will cause the display of a “Display message”.



## Settings

In the “Text” field you may type in the message you would like to be displayed.



- **Title**

Fill in the title of the message. You may use the Script expression editor (→[8.23.2.2](#)) to compose a message using the available variables, operands and operators.

- **Text type**

Define whether the email text is plain text or contains a complete HTML document.

- **Body**

Fill in the text, that will be contained in the message. You may use the Script expression editor (→[8.23.2.2](#)) to compose a message using the available variables, operands and operators.



In the field “Text” project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## 13.2 E-mails

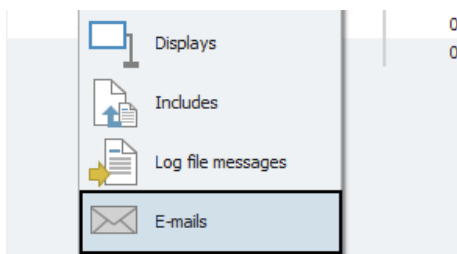
For surveillance-purposes it is possible to configure e-mails with user-specific content. Sending of these e-mails will be triggered by user-defined events.

### 13.2.1 Setting up the E-mails interface

In order to set up the “E-mails” interface, you will first need to add it to your system and then configure its SMTP settings. These steps will be explained in the following.

#### 13.2.1.1 Adding the E-mails interface

In order to add the “E-mails” interface select your system in the measurement task tree (the topmost element of the tree), click the “Components” button in the Ribbon and then select “E-mails”.

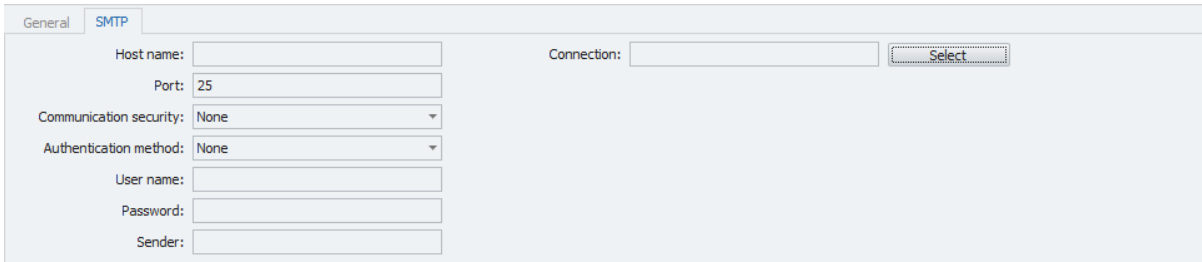




### 13.2.1.2 Configure SMTP

In order for the logger to be able to send e-mails, the SMTP settings have to be correctly set. To do so, fill in your e-mail provider's SMTP server details as well as your user name, password and sender.

To do so, select the "E-mails" interface in the measurement task tree, navigate to the "SMTP" tab in the details area and fill in the fields.



General SMTP

Host name:

Port: 25

Communication security: None

Authentication method: None

User name:

Password:

Sender:

Connection:

#### Connection

The field connection allows you to set the connection you wish to use for this SMTP configuration. Possible Connections are Ethernet and PPP connections.



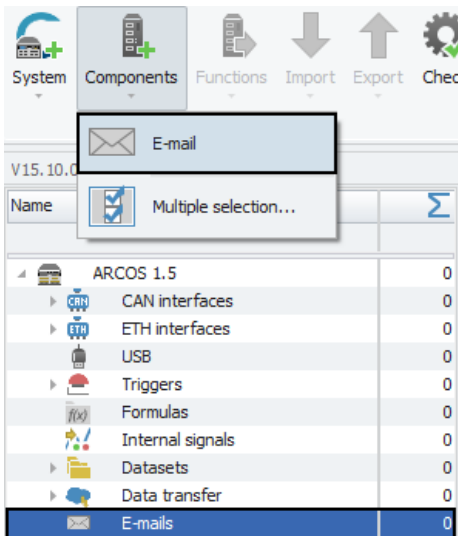
In the field "Sender" project parameters can be used as variables. For more information please refer to ([→5.6](#)).

### 13.2.2 Composing e-mails

This section will explain how to compose new e-mails and how to define a trigger for sending an e-mail.

#### 13.2.2.1 Creating a new e-mail

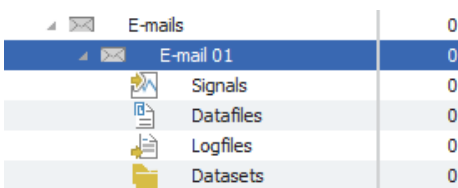
In order to compose an e-mail, you will first need to create it. To do so, select the “E-mails” interface in the measurement task tree, click on the “Components” button in the Ribbon and choose “E-mail”:



#### 13.2.2.2 Tree elements for E-mails

Each e-mail, that you have created in the “E-mails” interface will appear as a child element to the “E-mails” interface in the tree.

Each of these e-mail elements will possess four child elements itself, that will allow you to attach certain files to an e-mail. This will be explained in the chapter “E-mail attachments” (→ 13.2.3).



### 13.2.2.3 Grid area for E-mails

In the “Grid area” you will be presented with an overview of all the e-mails, that have been created so far, as well as details regarding those e-mails, such as subject, recipient, trigger etc. Also you can find here two important functions, which are the “[Column chooser](#)” (→[4.3.1](#)) and the “[Filter editor](#)” (→[4.3.2](#)).

Name	Active	Description	Subject	Text	Trigger	Recipient
<input type="checkbox"/> E-mail 01 E-mail 02	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	E-mail configuraton E-mail configuration	Notification E-mail 01 @ [tri... Notification E-mail 02 @ [tri...			

### 13.2.2.4 Details area for E-mails (Composing)

The Details area shows settings for “E-mails” allows you to compose an e-mail and set its trigger, as well as general settings.

#### General

This tab allows you to activate or deactivate the e-mail by ticking/unticking the checkbox. It also allows you to give a user specific name if wished and add an additional description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

General
Trigger
Settings

Active:

Name:

Description:

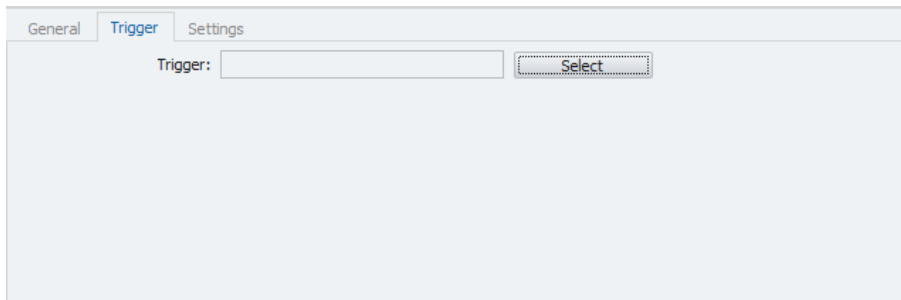
Reference:



In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).

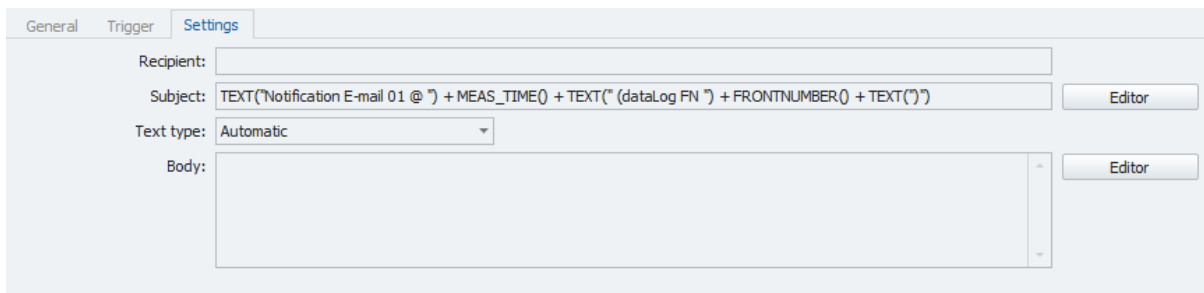
## Trigger

This tab allows you to set the trigger which will cause an e-mail to be sent. For each e-mail a trigger needs to be defined. And while each e-mail can only have one trigger defined, the same trigger can be defined for multiple mails.



## Settings

This is the tab, where the actual composing of the e-mail happens.



- **Recipient**

Fill in the e-mail address, that will receive the e-mail. Multiple recipients are possible. They have to be separated by comma, semicolon or a white space.

- **Subject**

Fill in a subject for the e-mail. You may use the Script expression editor (→[8.23.2.2](#)) to compose a dynamic message using the available variables, operands and operators.

- **Text type**

Define whether the email text is plain text or contains a complete HTML document.

- **Body**

Fill in the text, that will be contained in the e-mail. You may use the Script expression editor (→[8.23.2.2](#)) to compose a dynamic message using the available variables, operands and operators.



In the field "Body" project parameters can be used as variables. For more information please refer to (→5.6).

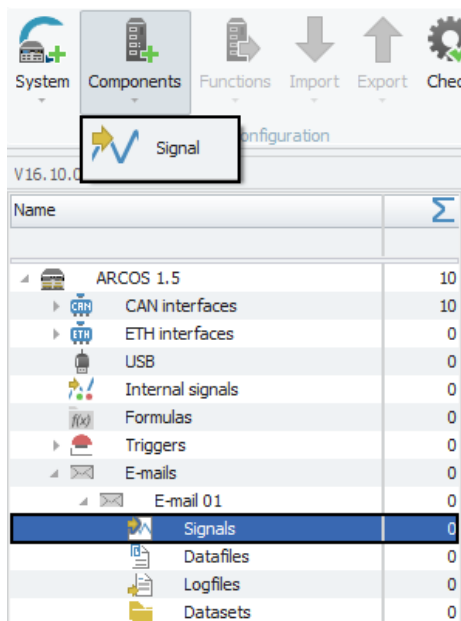
### 13.2.3 E-mail attachments

It is possible to attach files to e-mails and therefore make some parts of the acquired measurement data available for remote analysis on the fly. There are four different types of attachments available, that will be explained in the following.

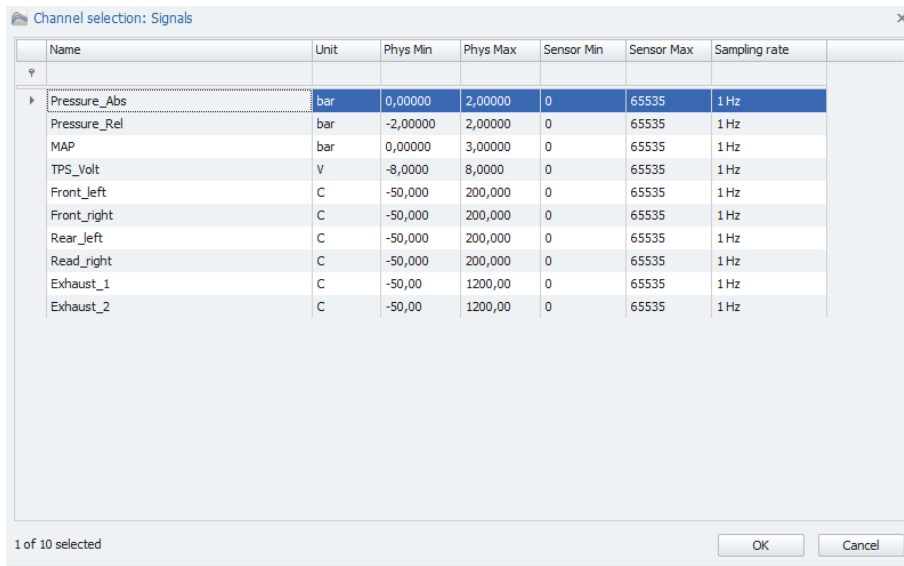
- Signals(→13.2.3.1)
- Datafiles(→13.2.3.2)
- Logfiles(→13.2.3.3)
- Datasets(→13.2.3.4)

#### 13.2.3.1 Signal attachments

To attach signals to an e-mail, select the "Signals" child element of the e-mail, to which you wish to attach signals, in the tree, click the "Components" button in the Ribbon and then choose "Signal".



The following window allows you to choose all the signals you wish to attach to the e-mail. Select all the signal you wish to attach and confirm with "OK".



The grid area of the child element "Signals" will present you with an overview of all the signals that have been selected for attachment.

Channel	Index	Active	Unit	Sampling rate
		<input type="checkbox"/>		
Pressure_Abs	1	<input checked="" type="checkbox"/>	bar	1 Hz
Pressure_Rel	2	<input checked="" type="checkbox"/>	bar	1 Hz
MAP	3	<input checked="" type="checkbox"/>	bar	1 Hz
TPS_Volt	4	<input checked="" type="checkbox"/>	V	1 Hz

### 13.2.3.2 Datafile attachments

To attach datafiles to an e-mail, select the “Datafiles” child element of the e-mail, to which you wish to attach signals, in the tree and navigate to the grid area. Here you will be presented with an overview of all the datafiles that currently exist within your configuration (if you haven't added any datafile yet, the grid area will remain empty).

Select the datafiles you wish to attach via the tickbox labeled “Attach to mail”.



“Script file” datafiles are no longer available as e-mail attachments.

Name	Active	Description	Attach to mail	Number of data files	Maximum size
ATFX 01	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	1	1 MB
Vector ASCII 01	<input checked="" type="checkbox"/>		<input type="checkbox"/>	1	1 MB
MDF 4.1 01	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	1	1 MB

If you select any of the datafiles in the grid area and navigate to the details area, you may access settings regarding the attached file.

#### General

Please refer to (→[4.2.2](#)).

#### Settings

Datafile attachment specific settings.

General
Settings

Attach to mail:

Number of data files:  d

Maximum size:

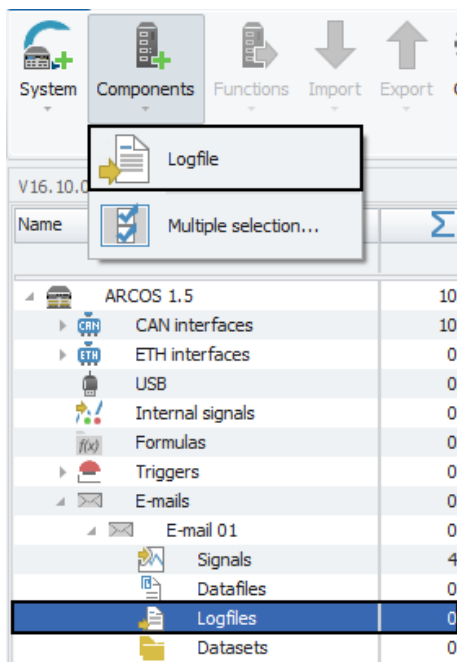
- **Attach to mail.**  
Same functionality as in the grid area. Mark active to attach this file to mail.
- **Number of data files**  
Define the maximum number of datafiles included in the attachment. Newer datafiles take precedence over older ones if the maximum size has been reached.

- **Maximum size**

Define the maximum total size of the attachment. This size is a total limit for all included data files. If the “Maximum size” is set to 1 Mb and data file 1 has 900 Kb, there will be only 100 Kb remaining for all the other data file attachments.

### 13.2.3.3 Logfile attachments

To attach logfiles to an e-mail, select the “Logfiles” child element of the e-mail, to which you wish to attach signals, in the tree, click the “Components” button in the Ribbon and then choose “Logfile”.



The grid area of the child element “Logfiles” will present you with an overview of all the logfiles that have attached.

Name	Active	Description	Logfile type	Dataset name	Maximum size
Logfile 01	<input checked="" type="checkbox"/>		Current run		1 MB
Logfile 02	<input checked="" type="checkbox"/>		Current run		1 MB



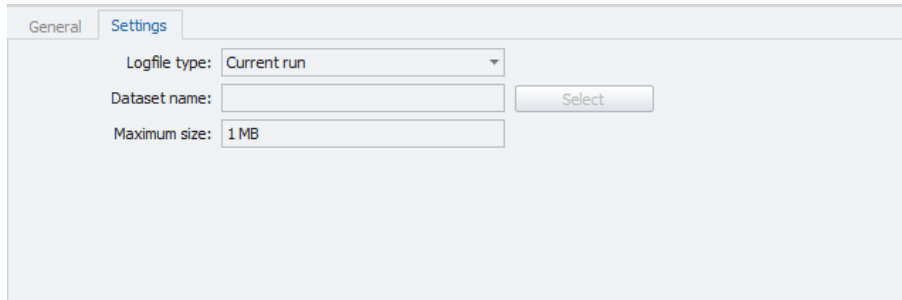
If you select any of the logfiles in the grid area and navigate to the details area, you may access settings regarding the attached file.

### General

Please refer to (→[4.2.2](#)).

### Settings

Logfile attachment specific settings.



- **Logfile type**

Define whether the logfile of the current run should be attached or a logfile from a specific dataset.

- **Dataset name**

If the logfile type has been set to dataset, choose here the dataset from which you wish to include the logfile.

- **Maximum size**

Define the maximum total size of the attachment.

#### 13.2.3.4 Dataset attachments

To attach datasets to an e-mail, select the “Datasets” child element of the e-mail, to which you wish to attach signals, in the tree and navigate to the grid area. Here you will be presented with an overview of all the datasets that currently exist within your configuration. Select the datafiles you wish to attach via the tickbox labeled “Attach to mail”.

Name	Active	Description	Attach to mail	Include running	Number of datasets	Maximum size
Dataset 01	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	1	1 MB
Dataset 02	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>	1	1 MB
Ring buffer 01	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>	1	1 MB

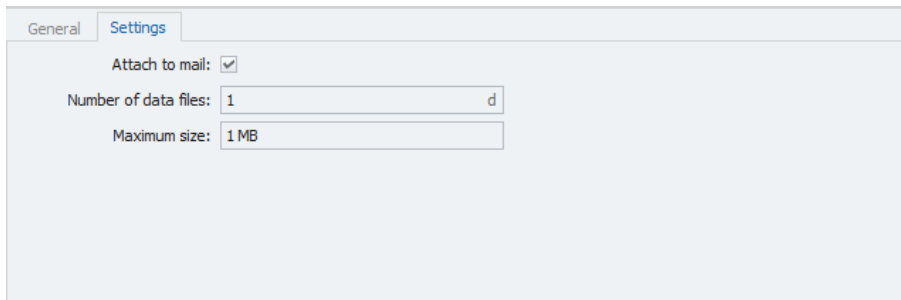
If you select any of the datasets in the grid area and navigate to the details area, you may access settings regarding the attached file.

## General

Please refer to (→[4.2.2](#)).

## Settings

Datafile attachment specific settings.



The screenshot shows a settings panel with two tabs: 'General' and 'Settings'. The 'Settings' tab is active. It contains three settings:

- 'Attach to mail:' with a checked checkbox.
- 'Number of data files:' with a text input field containing '1' and a dropdown menu set to 'd'.
- 'Maximum size:' with a text input field containing '1 MB'.

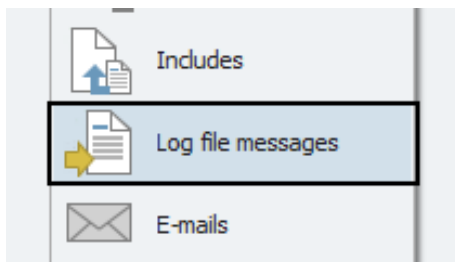
- **Attach to mail**  
Same functionality as in the grid area. Mark active to attach this file to mail.
- **Include running**  
If set, the running dataset is finished and included in the attachment.
- **Number of data files**  
Define the maximum number of datafiles included in the attachment. Newer datafiles take precedence over older ones if the maximum size has been reached.
- **Maximum size**  
Define the maximum total size of the attachment. This size is a total limit for all included datasets. If the "Maximum size" is set to 1 Mb and dataset 1 has 900 Kb, there will be only 100 Kb remaining for all the other dataset attachments.

## 13.3 Log file messages

For surveillance-purposes it is possible to configure log file messages with user-specific content. These messages will be written into the log file when triggered by a user defined trigger.

### 13.3.1 Adding the Log file messages interface

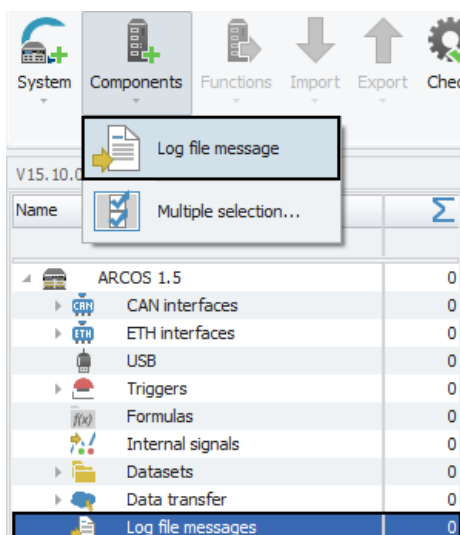
In order to create log file messages, you will first need to add the "Log file messages" interface to your system. To do so, select the system in the measurement task tree (the topmost element of the tree), click the "Components" button in the Ribbon and then choose "Log file messages".



### 13.3.2 Create a new Log file messages interface

Once the "Log file message" interface has been added to your system, you can now create a new message. To do so, select the "Log file messages" interface in the measurement task tree, click the "Components" button in the Ribbon and then choose "Log file message".

Composing and configuration of the message will be explained in the section "Details area for Log file messages" (→ [13.3.3.2](#)).



### 13.3.3 Composing Log file messages

This section will explain how to compose log file messages and how to define a trigger for writing the message to the log file.

#### 13.3.3.1 Grid area for Log file messages

In the "Grid area" you will be presented with an overview of all the Log file messages, that have been created so far, as well as details regarding those Log file messages, such as the trigger and text of the message.

Also you can find here two important functions, which are the "Column chooser" (→[4.3.1](#)) and the "Filter editor" (→[4.3.2](#)).

Name	Active	Description	Trigger	Text
☿	<input type="checkbox"/>			
▶ Log file message 01	<input checked="" type="checkbox"/>	Log file configuration		
Log file message 02	<input checked="" type="checkbox"/>	Log file configuration		
Log file message 03	<input checked="" type="checkbox"/>	Log file configuration		

### 13.3.3.2 Details area for Log file messages (Composing)

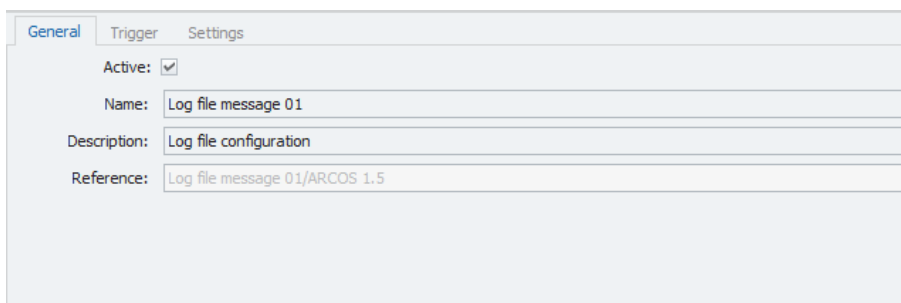
The Details area for “Log file messages” allows you to compose the message and set its trigger, as well as general settings.

To access the details area for a certain “Log file message”, select the message in the grid area and then navigate to the grid area.

#### General

This tab allows you to activate or deactivate the Log file message by ticking/unticking the checkbox.

It also allows you to give a user specific name if wished and add an additional description. The Reference field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.



The screenshot shows the 'General' tab of a configuration window. It has three sub-tabs: 'General', 'Trigger', and 'Settings'. The 'General' tab is active. It contains the following fields:

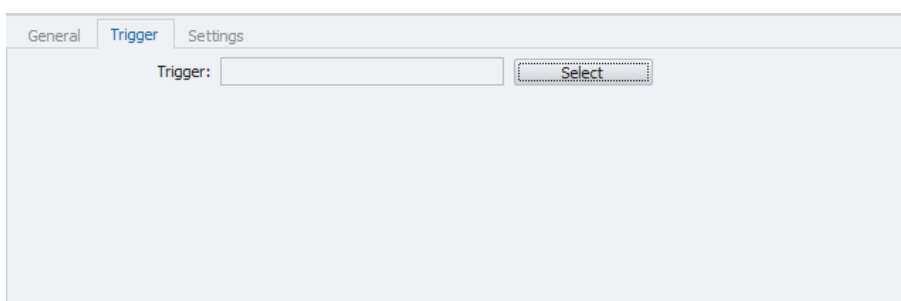
- Active:** A checked checkbox.
- Name:** A text field containing 'Log file message 01'.
- Description:** A text field containing 'Log file configuration'.
- Reference:** A text field containing 'Log file message 01/ARCOS 1.5'.



In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).

#### Trigger

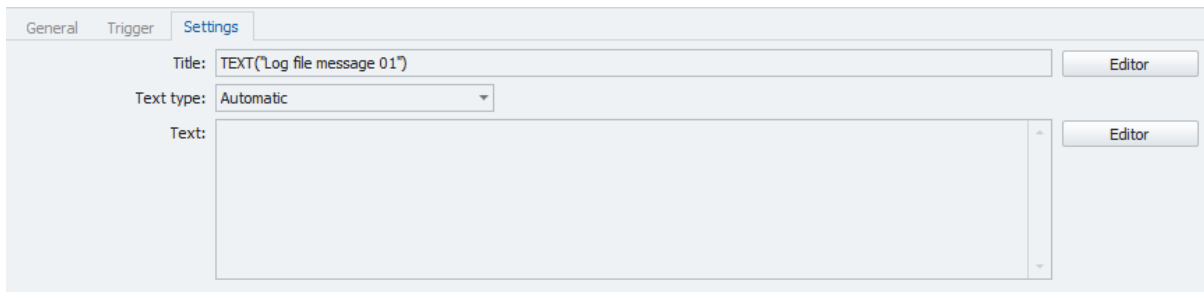
This tab allows you to set the trigger which will cause a Log file message to be written to the log file. For each Log file message a trigger needs to be defined. And while each Log file message can only have one trigger defined, the same trigger can be defined for multiple mails.



The screenshot shows the 'Trigger' tab of the configuration window. It has three sub-tabs: 'General', 'Trigger', and 'Settings'. The 'Trigger' tab is active. It contains a 'Trigger:' label followed by an empty text input field and a 'Select' button.

## Settings

This is the tab, where the actual composing of the Log file message happens. Write in any text you wish in the field labeled "Text". This text will then be written as a message to the log file when triggered.



- **Title**

Fill in the title of the message. You may use the Script expression editor (→[8.23.2.2](#)) to compose a dynamic title using the available variables, operands and operators.

- **Text type**

Define whether the email text is plain text or contains a complete HTML document.

- **Body**

Fill in the text, that will be contained in the message. You may use the Script expression editor (→[8.23.2.2](#)) to compose a dynamic message using the available variables, operands and operators.

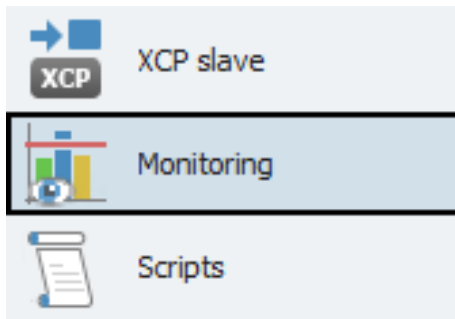


In the field "Text" project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## 13.4 Monitoring

The “Monitoring” interface allows you to define certain limit values or ranges for a signal or channel and monitor if these limits are being violated. The result of monitoring operations can be displayed live on any connected display.

In order to do so, you will first need to add the “Monitoring” interface to your system. Select the system (the topmost element of the tree) in the measurement task tree, click the “Components” button in the Ribbon and then choose “Monitoring”.



Next you will need add and configure either a “Limit value” or a “Range”. To do so please refer to the respective sections of this chapter:

- Booleans (→ [13.4.2](#))
- Limit value (→ [13.4.3](#))
- Range (→ [13.4.4](#))

### 13.4.1 Tree elements for Monitoring

Once the “Monitoring” interface has been added to your system, it will appear as a new tree element in the measurement task tree. It will also contain three child elements called “Booleans”, “Limit value” and “Range”.

These child elements will contain all the “Booleans”, “Limit values” and “Ranges”, that you later add to your system.

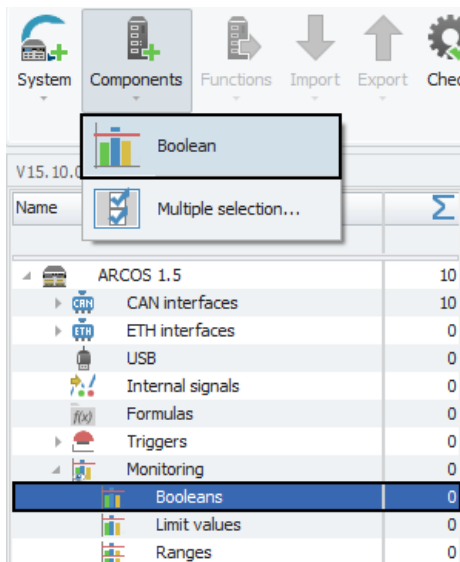
Triggers	0
Monitoring	0
Booleans	0
Limit values	0
Ranges	0
Datasets	0

### 13.4.2 Booleans

The “Booleans” function in “Monitoring” allows you to define a boolean condition in connection with a signal. If the boolean condition becomes true, a user-specific action will be executed.

#### 13.4.2.1 Adding Booleans

In order to add a boolean, select the tree element “Booleans”, click on the “Components” button in the Ribbon and then choose “Boolean”.



#### 13.4.2.2 Grid area for Booleans

In the “Grid area” you will be presented with an overview of all the Booleans, that have been created so far, as well as details regarding those Booleans, such as signal, operation, etc..

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

Name	Active	Description	Channel	Operation	Output
Limit value 01	<input checked="" type="checkbox"/>			=	No output
Limit value 02	<input checked="" type="checkbox"/>			=	No output

#### 13.4.2.3 Details area for Booleans

The Details area contains settings regarding the Booleans.

##### General

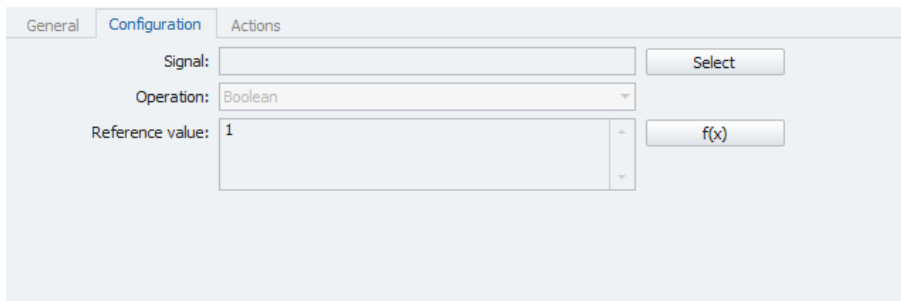
Please refer to (→4.2.2).

Changes and errors excepted.



## Configuration

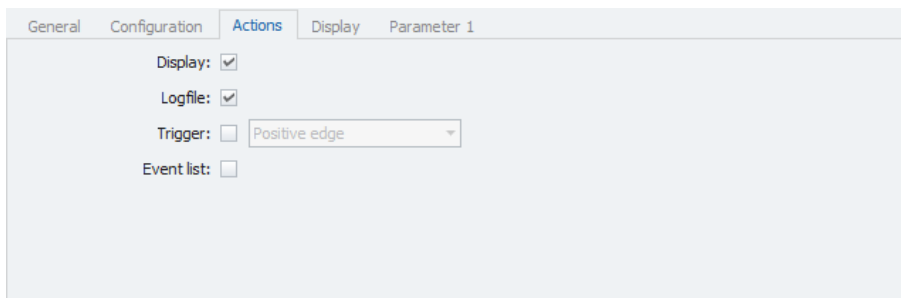
This tab allows you to define to which signal a boolean should apply.



- **Signal**  
Choose the Signal to which the boolean will apply.
- **Operation**  
This setting is by default set to boolean and cannot be changed.
- **Reference value**  
The reference value is the boolean condition. It can be a fixed value, a signal or a formula. For details on working with the formula editor please refer to (→[8.23.1.4](#)).

## Actions

This tab allows you to define what action will be taken, should a boolean condition become true.



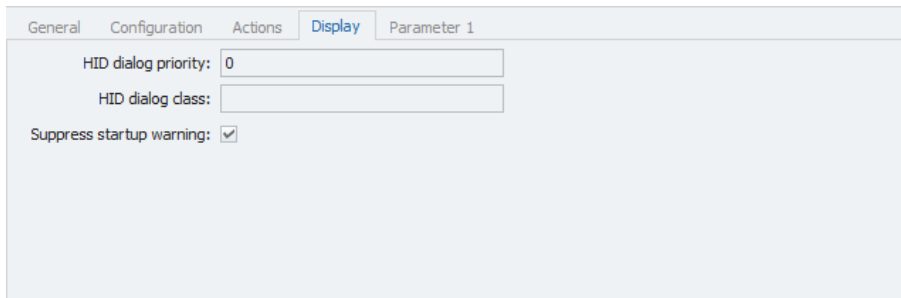
- **Display**  
If activated, a Display option will be used. The Display option comes with further settings in extra tabs, that need to be set. These extra settings will be explained in the sections concerning the "Display" tabsheet (→[13.4.2.3](#)) and the "Parameter x" tabsheet (→[13.4.2.3](#)).
- **Logfile**  
If set active, a logfile entry will be created upon limit violation or if a boolean condition becomes true.
- **Trigger**  
If set, a trigger action will be used. You may further set the trigger mode to **Positive/Negative Edge** or **Stateful**.

- **Eventlist**

If set, an eventlist entry will be created, if either the "EVENT" or the "FEGER" header is used in the dataset. Please refer to (→ [14.1.3](#)).

## Display

Define the display output settings.



- **HID dialog priority**

Define the HID dialog priority. Higher priorities are ranked higher and will thus be preferred if there is a multiple occurrence of display actions.

- **HID dialog class**

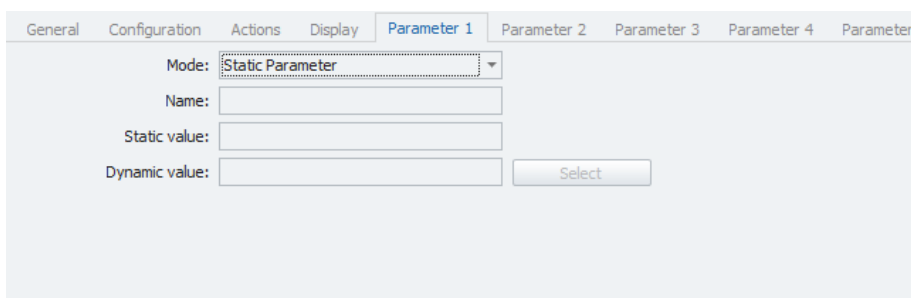
Define the HID dialog class, that will be displayed when a display action occurs.

- **Suppress startup warning**

If this tickbox is marked active, no warning will be shown on the display, if the limit is already exceeded at startup.

## Parameter x

This tab allows you define the type fo parameter that should be displayed and its content. Up to 5 parameters can be defined.



- **Mode**

Define whether no parameter, a static parameter or a dynamic parameter should be displayed.

- **Name**

Define the parameter name.

Changes and errors excepted.

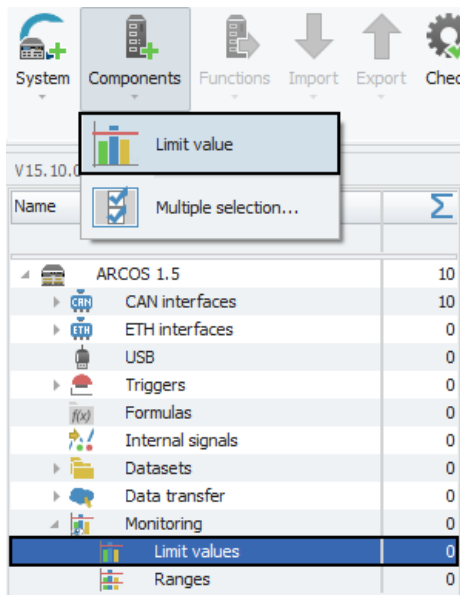
- **Static value**  
Define the parameter value for a static parameter.
- **Dynamic value**  
Select a signal as the parameter value for a dynamic parameter.

### 13.4.3 Limit value

The “Limit value” function in “Monitoring” allows you to define a single value for a signal and to compare the signal to this value. The result of this comparison will be put out to the connected display.

#### 13.4.3.1 Adding a limit value

In order to add a limit value, select the tree element “Limit value”, click on the “Components” button in the Ribbon and then choose “Limit value”.



#### 13.4.3.2 Grid area for Limit values

In the “Grid area” you will be presented with an overview of all the limit values, that have been created so far, as well as details regarding those limit values, such as channel, operation, output etc..

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

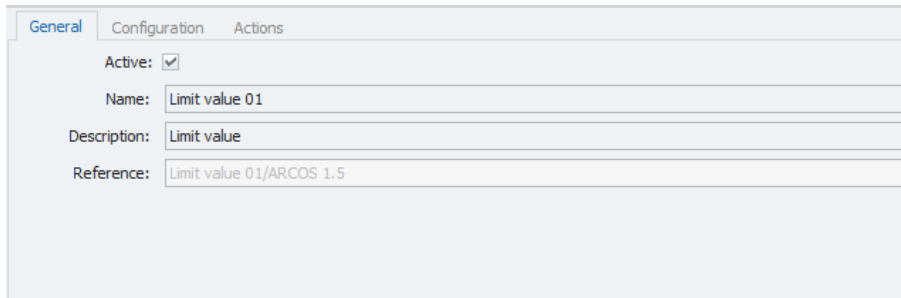
Name	Active	Description	Channel	Operation	Output
Limit value 01	<input checked="" type="checkbox"/>			=	No output
Limit value 02	<input checked="" type="checkbox"/>			=	No output

### 13.4.3.3 Details area for Limit values

The Details area contains settings regarding the limit value.

#### General

This tab provides general settings for a limit value.



- **Name**

Give a user-defined Name to the selected Limit value.

- **Description**

Give a user-defined description to the selected Limit value.

- **Reference**

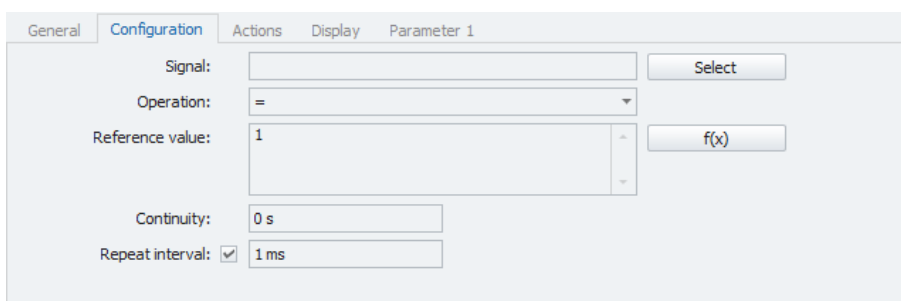
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.



In the field "Name" project parameters can be used as variables. For more information please refer to ([→5.6](#)).

#### Configuration

This tab allows you to define to which signal a limit value should apply, as well as in what way.



- **Signal**

Choose the Signal to which the limit value will apply.

- **Operation**

Select the boolean operator by which the signal will be compared to the limit value. This will determine whether you will get a result when the signal hits the reference value, exceeds it or falls below it.

- **Reference value**

The reference value is the value to which the signal will be compared. It can be a fixed value, a channel or a formula. For details on working with the formula editor please refer to (→[8.23.1.4](#)).

- **Continuity**

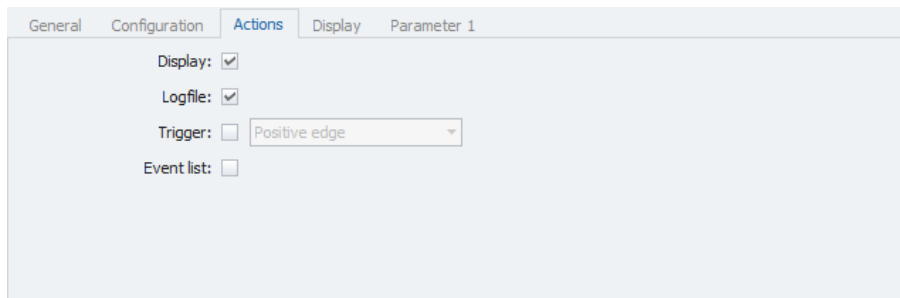
Define the required minimum duration of a limit violation or boolean true-value in order to be put out .

- **Repeat interval**

Interval before reactivation of the event if the limit violation is still existent or the boolean condition is still true.

## Actions

This tab allows you to define what action will be taken, should a limit violation occur.



General Configuration **Actions** Display Parameter 1

Display:

Logfile:

Trigger:  Positive edge

Event list:

- **Display**

If activated, a Display option will be used. The Display option comes with further settings in extra tabs, that need to be set. These extra settings will be explained parts of this section concerning the "Display" tabsheet (→[13.4.3.3](#)) and the "Parameter x" tabsheet (→[13.4.3.3](#)).

- **Logfile**

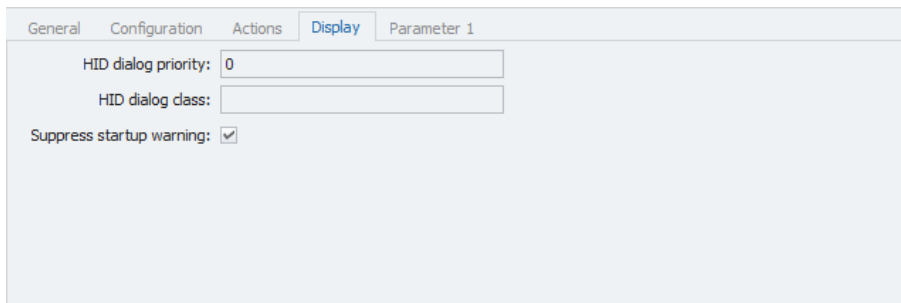
If set active, a logfile entry will be created upon limit violation or if a boolean condition becomes true.

- **Trigger**

If set, a trigger action will be used. You may further set the trigger mode to **Positive/Negative Edge** or **Stateful**.

- **Eventlist**

If set, an eventlist entry will be created, if either the "EVENT" or the "FEGER" header is used in the dataset. Please refer to (→[14.1.3](#)).



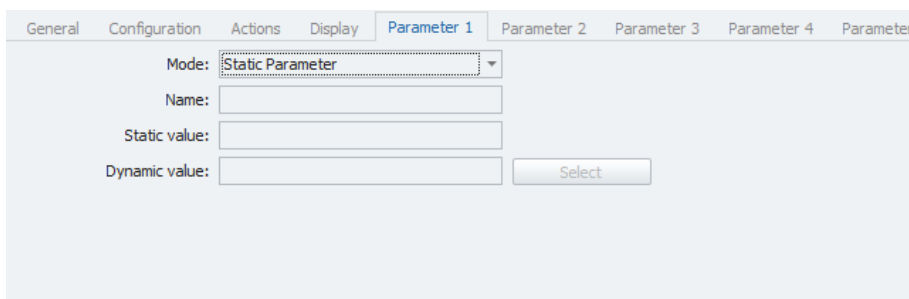
## Display

Define the display output settings.

- **HID dialog priority**  
Define the HID dialog priority. Higher priorities are ranked higher and will thus be preferred if there is a multiple occurrence of display actions.
- **HID dialog class**  
Define the HID dialog class, that will be displayed when a display action occurs.
- **Suppress startup warning**  
If this tickbox is marked active, no warning will be shown on the display, if the limit is already exceeded at startup.

## Parameter x

This tab allows you define the type fo parameter that should be displayed and its content. Up to 5 parameters can be defined.



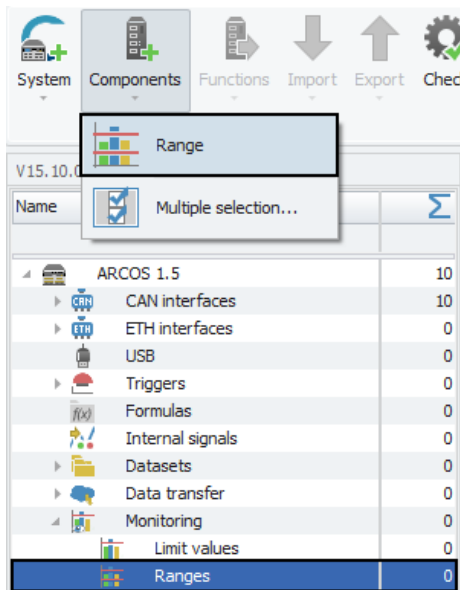
- **Mode**  
Define whether no parameter, a static parameter or a dynamic parameter should be displayed.
- **Name**  
Define the parameter name.
- **Static value**  
Define the parameter value for a static parameter.
- **Dynamic value**  
Select a signal as the parameter value for a dynamic parameter.

### 13.4.4 Range

The “Range” function in “Monitoring” allows you to define a double set of limit values. You can thus create a Range of values to which to compare the signal. The result of this comparison will be put out to the log file and can additionally be displayed on a connected display.

#### 13.4.4.1 Adding a Range

In order to add a Range, select the tree element “Range”, click on the “Components” button in the Ribbon and then choose “Range”.



#### 13.4.4.2 Grid area for Ranges

In the “Grid area” you will be presented with an overview of all the Ranges, that have been created so far, as well as details regarding those Ranges, such as channel, operation, output etc..

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

Name	Active	Description	Channel	Operation	Top output
Range 01	<input checked="" type="checkbox"/>			Limit violation out of range	No output
Range 02	<input checked="" type="checkbox"/>			Limit violation out of range	No output
Range 03	<input checked="" type="checkbox"/>			Limit violation out of range	No output



### 13.4.4.3 Details area for Ranges

The Details area contains settings regarding the Range.

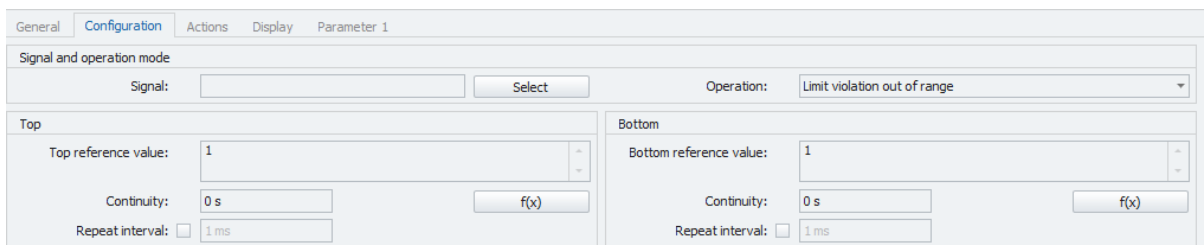
#### General

Please refer to (→[4.2.2](#)).

#### Configuration

This tab allows you to define to which signal a Range should apply, as well as in what way.

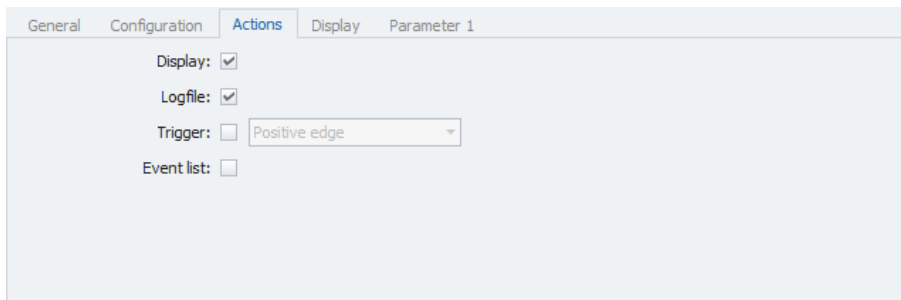
For a Range you need to define a top limit value and a bottom limit value. You can then compare the signal to that range and see if it is inside or outside of the range.



- **Signal**  
Choose the Signal to which the Range will apply.
- **Operation**  
Select the boolean operator by which the signal will be compared to the Range.
- **(Top/Bottom) Reference value**  
The reference value is the value to which the signal will be compared. It can be a fixed value, a channel or a formula. For details on working with the formula editor please refer to (→[8.23.1.4](#)).
- **Continuity**  
Define the required minimum duration of a limit violation or boolean true-value in order to be put out.
- **Repeat interval**  
Interval before reactivation of the event if the limit violation is still existent or the boolean condition is still true.

## Actions

This tab allows you to define what action will be taken, should a limit violation occur.



General Configuration **Actions** Display Parameter 1

Display:

Logfile:

Trigger:  Positive edge

Event list:

- **Display**

If activated, a Display option will be used. The Display option comes with further settings in extra tabs, that need to be set. These extra settings will be explained parts of this section concerning the “Display” tabsheet (→ [13.4.4.3](#)) and the “Parameter x” tabsheet (→ [13.4.4.3](#)).

- **Logfile**

If set active, a logfile entry will be created upon limit violation or if a boolean condition becomes true.

- **Trigger**

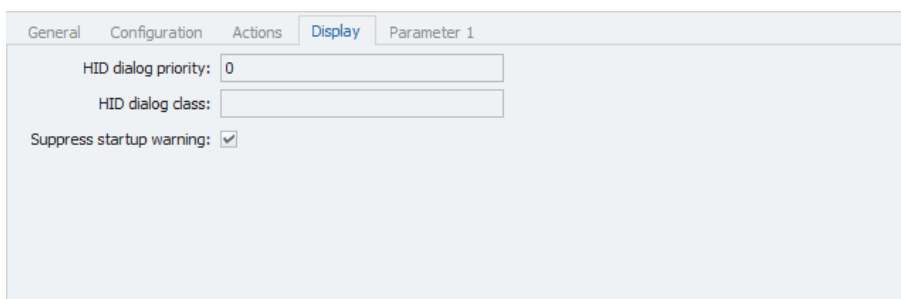
If set, a trigger action will be used. You may further set the trigger mode to **Positive/Negative Edge** or **Stateful**.

- **Eventlist**

If set, an eventlist entry will be created, if either the “EVENT” or the “FEGER” header is used in the dataset. Please refer to (→ [14.1.3](#)).

## Display

Define the display output settings.



General Configuration Actions **Display** Parameter 1

HID dialog priority: 0

HID dialog class:

Suppress startup warning:

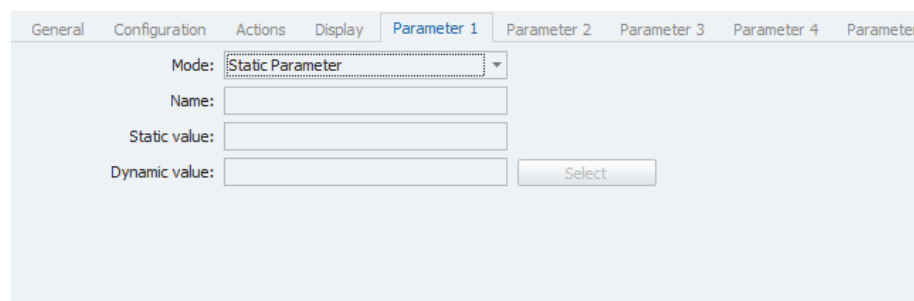
- **HID dialog priority**

Define the HID dialog priority. Higher priorities are ranked higher and will thus be preferred if there is a multiple occurrence of display actions.

- **HID dialog class**  
Define the HID dialog class, that will be displayed when a display action occurs.
- **Suppress startup warning**  
If this tickbox is marked active, no warning will be shown on the display, if the limit is already exceeded at startup.

### Parameter x

This tab allows you define the type fo parameter that should be displayed and its content. Up to 5 parameters can be defined.



The screenshot shows a software interface with a tabbed menu at the top: General, Configuration, Actions, Display, Parameter 1, Parameter 2, Parameter 3, Parameter 4, and Parameter 5. The 'Parameter 1' tab is active. Below the tabs, there are four input fields: 'Mode:' with a dropdown menu showing 'Static Parameter', 'Name:', 'Static value:', and 'Dynamic value:'. A 'Select' button is positioned to the right of the 'Dynamic value:' field.

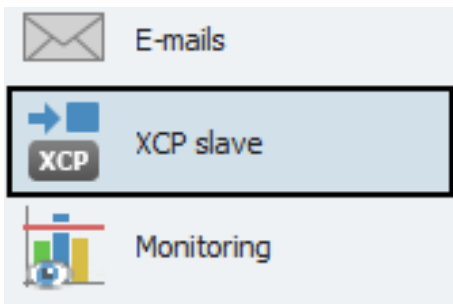
- **Mode**  
Define whether no parameter, a static parameter or a dynamic parameter should be displayed.
- **Name**  
Define the parameter name.
- **Static value**  
Define the parameter value for a static parameter.
- **Dynamic value**  
Select a signal as the parameter value for a dynamic parameter.

## 13.5 XCP slave

“XCP slave” allows you to connect the logger to a PC via ethernet and let the PC function as XCP master. That means, the PC will be requesting and receiving signals from the logger.

### 13.5.1 Adding XCP slave

In order to add XCP slave, select the system in the measurement task tree (the topmost element of the tree), click the “Components” button in the Ribbon and then choose “XCP slave”.



### 13.5.2 Tree elements for XCP slave

Adding “XCP slave” to the system will add one new tree element called “XCP slave”.

▶	Datasets	0
▶	Data transfer	0
	XCP slave	0

### 13.5.3 Grid area for XCP slave

The Grid area provides you with an overview of the ethernet channels available for XCP slave. Here you need to select the ethernet channel you wish to use by ticking the “Active”

checkbox.



It is possible to select multiple ethernet channels for XCP slave, but if you choose more than one channel, all available channels will be activated automatically.

Name	Active	Description	Channel number
Front	<input checked="" type="checkbox"/>	Ethernet channel	3
openABK	<input type="checkbox"/>	Ethernet channel	2

### 13.5.4 Details area for XCP slave

The details area for XCP slave contains all the important settings regarding the xcp slave connection.



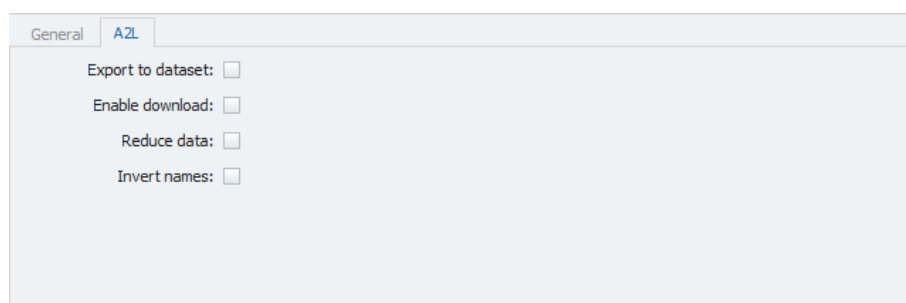
The important settings for XCP slave are located in the details area of the tree element "XCP slave", not in the details area of any of the ethernet connections in the grid area.

#### General

Please refer to (→[4.2.2](#)).

#### A2L

This tab provides settings regarding the type and location of the A2L file provided to a PC for establishing a connection with the ECU.



- **Export to dataset**

Activating this option will store the entire A2L file of the logger, including all bus-, ECU- and logger-signals, inside the dataset. From there it can be copied to your PC and used for XCP slave.

- **Enable download**

Activating this option will make the A2L file of the logger, including all bus-, ECU- and logger-signals, available for download from the web interface. From there it can be downloaded to your PC and used for XCP slave.

- **Reduce data**

If this option is activated, only a reduced file size A2L file will be available for copying/download. This A2L file will only contain the information necessary for the PC to connect to the logger, from where it can then download the entire A2L file including all bus-, ECU- and logger-signals.

- **Invert names**

If this option is set, the measurement names will be written in reversed order. This can be useful for working with some third-party tools.

*Example*

*Regular order: 'some'::'namespace'::'signalname'*

*Reversed order: 'signalname'::'namespace'::'some'*

## 14 Datasets

Data acquired throughout a measurement task will be stored in a dataset or a ring buffer by the logger. This chapter will explain how to configure datasets and ring buffers.

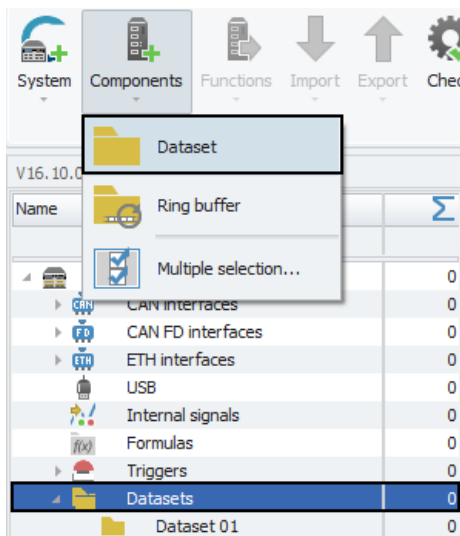
### 14.1 Dataset

This section will explain how to configure a regular dataset.

#### 14.1.1 Adding extra datasets

By default a system is configured with one dataset. It is possible however, to configure multiple datasets.

To do so, select the "Datasets" interface in the measurement task tree, click the "Components" button in the Ribbon and then choose "Dataset".



#### 14.1.2 Tree elements for Datasets

The tree element "Datasets" will contain all the datasets you configure for your system. The child element "Dataset" then contains the single components of your dataset.

## 14.1 DATASET

Name	
ARCOS 1.5 - dataLog	0
CAN Interfaces	0
CAN 01	0
CAN 02	0
ETH Interfaces	0
Internal	0
Front	0
openABK	0
USB	0
Triggers	0
Standard Triggers	0
Level Triggers	0
Cyclic Triggers	0
Trigger Groups	0
Formulas	0
Internal signals	0
<b>Datasets</b>	<b>0</b>
Dataset	0

### 14.1.3 Details area for Datasets

This section contains settings regarding the overall behaviour of your dataset. These settings are global and will affect all components of your dataset.

In case the parent tree element **"Dataset"** selected, the details area will only show the **"General"** tab. Please refer to (→[4.2.2](#)).

In case the child element **"Dataset"** is selected, the details area will contain additional tabs which will be explained in the following.

General	File settings	Dataset settings	Generate	Header
Active: <input checked="" type="checkbox"/>				
Name: Dataset				
Description: A dataset for one/multiple datafile(s)				
Reference: Dataset/ARCOS 1.5 - dataLog				

#### General

Please refer to (→[4.2.2](#)).

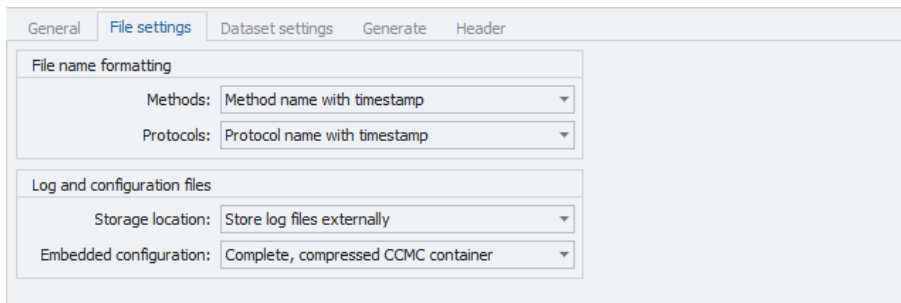


In the field **"Name"** project parameters can be used as variables. For more information please refer to (→[5.6](#)).

#### File settings

This tab contains settings regarding filename and location.





- **Filename formatting**

This field allows you to define the filename formatting of the final dataset file. Both methods and protocols have to possible formatting options:

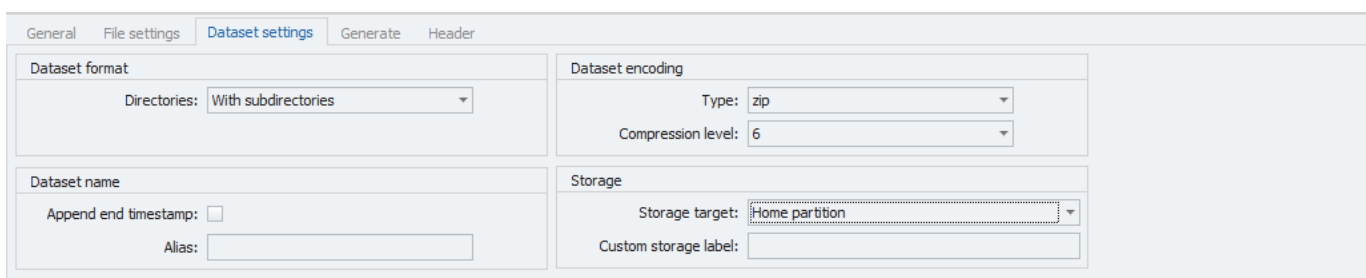
- Method/Protocol name with timestamp: protocol name with timestamp (date and time)(default setting)
- Timestamp, trigger name and counter: Appends the name of the activating trigger and its number/counter to the right of the timestamp in the file name.

- **Log and configuration files** This field allows you to set whether your dataset file will be stored externally or internally and what type of configuration to embed. There are four types of configuration available:

- Complete, compressed CCMC container: Entire container is stored as is.
- Uncompressed contents of CCMC container: Content of the container is unpacked and stored.
- Only CFG: Only the cfg is extracted.
- Complete, compressed CCMC container and uncompressed contents: Entire container is stored as is and the unpacked content of the container is stored as well.

## Dataset settings

This tab allows for settings regarding the dataset itself.



- **Dataset format**

Define whether the directory structure of data within a dataset contains subdirectories or not.

- **Dataset encoding**

Set the type of encoding and compression level for the dataset.

- **Dataset name**

This field allows you to select, whether a timestamp will be appended to the dataset name.

Also you can define an alias for this dataset. The alias can then be used to reference the dataset in configuration includes. The alias has to be unique within all datasets.

- **Storage target**

This dropdown menu allows you to select, where the dataset shall be stored.

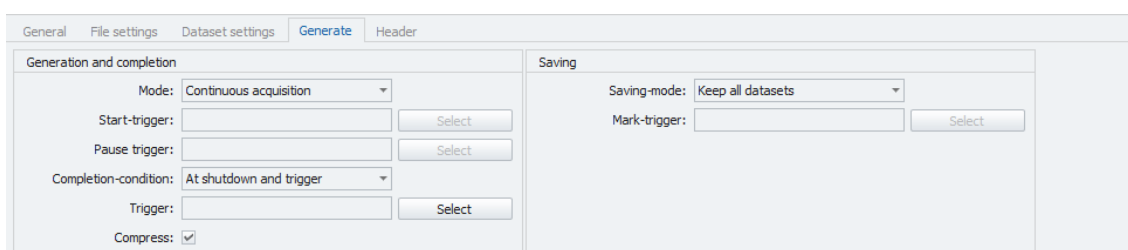
Storage target	Description
Home partition	Dataset will be stored on the internal home partition
Partition with datalog_extension_cfg	Dataset will be stored on an external drive with a datalog_extension_cfg
Left/Right drivebay slot	Dataset will be stored on an extension disk installed in the left/right drivebay slot
Custom storage label	Dataset will be stored to a custom partition. Custom partition can be created and managed via the loggers webinterface.

- **Custom storage label**

Here you can fill in the name of the custom partition if the option "Custom storage label" has been selected.

## Generate

Define settings regarding the creation of the dataset.



The screenshot shows the 'Generate' tab in a software interface. It is divided into two main sections: 'Generation and completion' and 'Saving'.

**Generation and completion:**

- Mode: Continuous acquisition (dropdown menu)
- Start-trigger: [text input] [Select button]
- Pause trigger: [text input] [Select button]
- Completion-condition: At shutdown and trigger (dropdown menu)
- Trigger: [text input] [Select button]
- Compress:

**Saving:**

- Saving-mode: Keep all datasets (dropdown menu)
- Mark-trigger: [text input] [Select button]

- **Mode**

Choose between continuous data acquisition or triggered data acquisition.

- **Start- and Pause-trigger**

This field allows you, to set triggers to start or pause dataset creation. If you do not

set a start trigger, then by default dataset creation is always active during measurement.

- **Completion**

The dropdown menu “**Condition**” allows you to set a condition under which the dataset should be closed and completed. In case your condition for completion requires a trigger, you may choose the trigger in the below field.

If you choose “**Compress after completion**”, the file will be compressed directly after completion. If you do not choose this option, the dataset will be compressed before transfer.

- **Saving-mode**

Select the saving-mode. The different options are listed below.

Saving-mode	Characteristics
Keep all datasets	All datasets will be saved.
Keep marked datasets	Choose a trigger to mark the dataset if the trigger is fired. Only marked datasets will be saved, all other datasets will be discarded.
Discard marked datasets	Choose a trigger to mark the dataset if the trigger is fired. All marked datasets will be discarded, all other datasets will be saved.

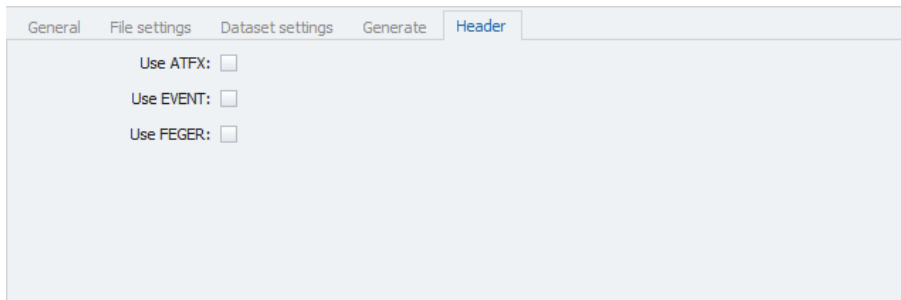
- **Mark-trigger**

Select a trigger to mark a dataset.

### Header

When storing the data, the logger saves general information about the measurement (identification, comments, start, stop) and information about the data files (name, structure) in a header file. For additional functionality a various headers with specific options can be attached. The options of each activated header-type will be accessible in a separate tab with the header-type's name.

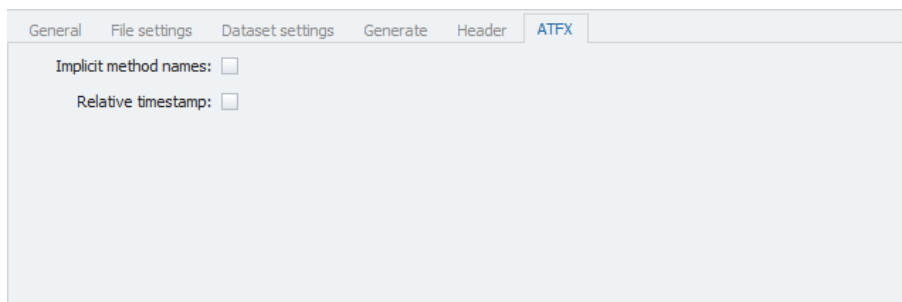
Multiple header-types can be activated at the same time.



The screenshot shows the 'Header' tab selected in a settings window. The window has tabs for 'General', 'File settings', 'Dataset settings', 'Generate', and 'Header'. The 'Header' tab contains three options, all of which are unchecked:

- Use ATFX:
- Use EVENT:
- Use FEGER:

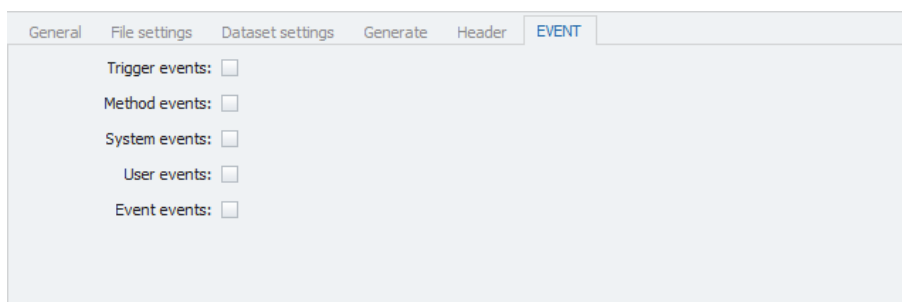
- **ATFX**



The screenshot shows the 'ATFX' tab selected in the settings window. The window has tabs for 'General', 'File settings', 'Dataset settings', 'Generate', 'Header', and 'ATFX'. The 'ATFX' tab contains two options, both of which are unchecked:

- Implicit method names:
- Relative timestamp:

- **EVENT**



The screenshot shows the 'EVENT' tab selected in the settings window. The window has tabs for 'General', 'File settings', 'Dataset settings', 'Generate', 'Header', and 'EVENT'. The 'EVENT' tab contains five options, all of which are unchecked:

- Trigger events:
- Method events:
- System events:
- User events:
- Event events:

- **FEGER**



The screenshot shows the 'FEGER' tab selected in the settings window. The window has tabs for 'General', 'File settings', 'Dataset settings', 'Generate', 'Header', and 'FEGER'. The 'FEGER' tab contains six options:

















- Trigger events:
- Method events:
- System events:
- User events:
- Event events:
- Persistent variables:

At the bottom of the tab, there is a 'Mode' dropdown menu currently set to 'Multilog'.

### 14.1.4 Setting up a dataset

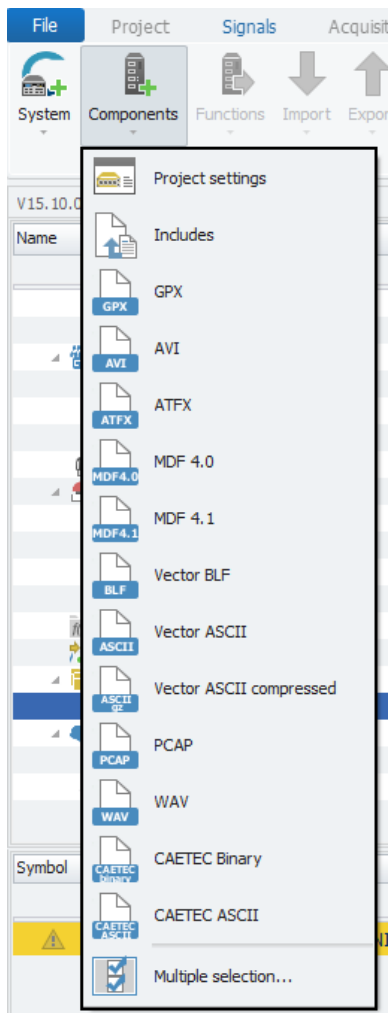
A dataset can be set up using different filetypes, according to what information you want it to contain. Each filetype you include in your dataset has specific functionalities and for each included filetype will later be included a file in the exported dataset with its previously configured name and the according filetype extension.

In order to include a filetype in your dataset, select the tree element "Dataset".

Name	$\Sigma$
 CAN 01	0
 CAN 02	0
 ETH Interfaces	0
 Internal	0
 Front	0
 openABK	0
 USB	0
 Triggers	0
 Standard Triggers	0
 Level Triggers	0
 Cyclic Triggers	0
 Trigger Groups	0
 Formulas	0
 Internal signals	0
 Datasets	0
 Dataset	0

## 14.1 DATASET

Then click the “Components” button in the Ribbon and choose the desired filetype from the resulting menu.



The following sections will explain in detail the different filetypes.

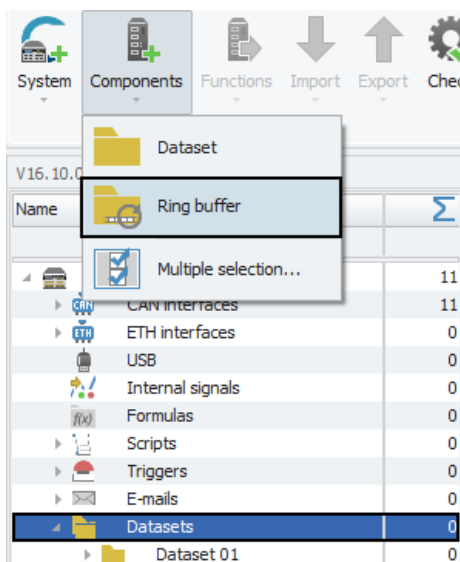
## 14.2 Ring buffer

A ring buffer is a dataset, that can continuously store data from bus trace files and AVI files. Once the defined maximum file size has been reached, old acquisition data from the beginning of the dataset will be erased in order to store new acquisition data. In this way, the ring buffer always keeps a specified amount of the most recent acquisition.

It is, however, possible to define certain triggers that will mark a certain datafile, that will then not be deleted. Please refer to (→ [14.8.5.1](#)).

### 14.2.1 Adding a ring buffer

To add a ring buffer to your configuration, select the “Datasets” interface in the measurement task tree, click the “Components” button in the Ribbon and then choose “Ring buffer”.



### 14.2.2 Setting up a ring buffer

The datafiles that will be included in a ring buffer can be set up in the same way as in a dataset. Please refer to the chapter “Setting up a dataset” (→ [14.1.4](#)).

Other than a dataset, a ring buffer can only contain the following filetypes:

- Vector BLF / Vector ASCII / Vector ASCII compressed (→ [14.8](#))
- PCAP (→ [14.9](#))
- AVI (→ [14.10](#))

The sections of this manual for each of the three filetype mentioned above will contain a part that will explain the specifics of that filetype when used in a ring buffer.

### 14.2.3 Tree elements for ring buffer

Once a ring buffer has been added, it will appear as a child element to the “Dataset” interface in the measurement task tree.

└─ Datasets	0
└─ Dataset 01	0
└─ Ring buffer 01	0

### 14.2.4 Grid area for ring buffer

In the “Grid area” you will be presented with an overview of all the datafiles, that have been added to the ring buffer and you can activate or deactivate single filetypes for storage in the ring buffer.

Also you can find here two important functions, which are the “**Column chooser**” (→[4.3.1](#)) and the “**Filter editor**” (→[4.3.2](#)).

	Name	Active	Description	File type
?		<input type="checkbox"/>		
▶	AVI 01	<input checked="" type="checkbox"/>	File for saving video data	AVI
	Vector BLF 01	<input checked="" type="checkbox"/>	File for saving acquisition data	BLF

### 14.2.5 Details area for Ring buffer

This section contains settings regarding the behaviour of your ring buffer.

General	File settings	Dataset settings	Generate
Active: <input checked="" type="checkbox"/>			
Name: Ring buffer 01			
Description: Ringbuffer Dataset			
Reference: Ring buffer 01/ARCOS 1.5			

#### General

Please refer to (→[4.2.2](#)).

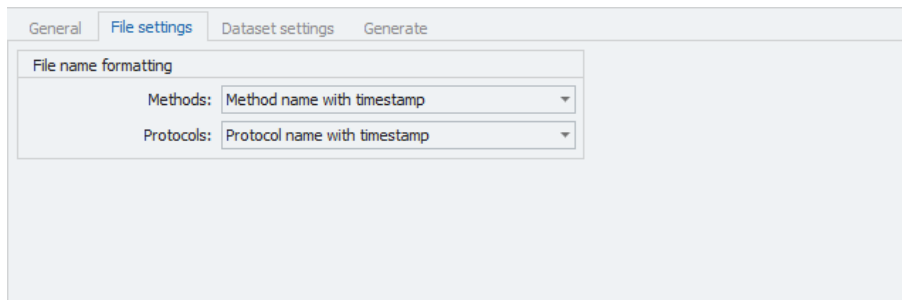


In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).



## File settings

This tab contains settings regarding file name formatting.



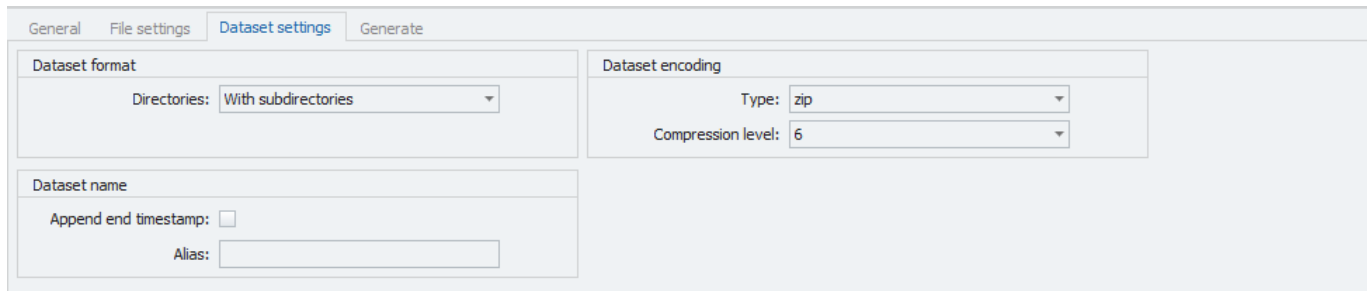
- **Filename formatting**

This field allows you to define the filename formatting of the final ring buffer file. Both methods and protocols have two possible formatting options:

- Method name/Protocol name with timestamp: protocol name with timestamp (date and time)(default setting)
- Timestamp, trigger name and counter: Appends the name of the activating trigger and its number/counter to the right of the timestamp in the file name.

## Dataset settings

This tab allows for settings regarding the ring buffer itself.



- **Dataset format**

Define whether the directory structure of data within a ring buffer contains subdirectories or not.

- **Dataset encoding**

Set the type of encoding and compression level for the ring buffer.

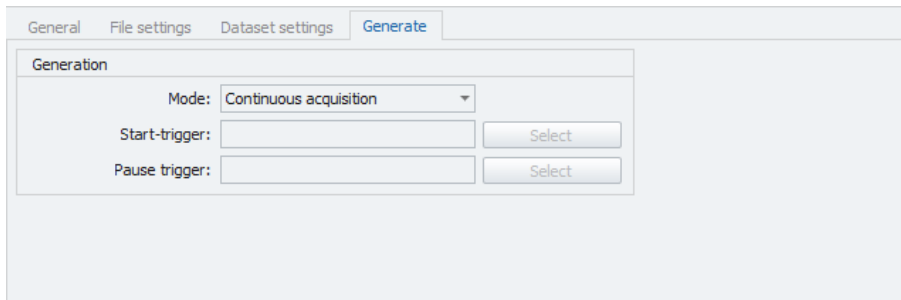
- **Dataset name**

This field allows you to select, whether a timestamp will be appended to the ring buffer name.

Also you can define an alias for this ring buffer. The alias can then be used to reference the ring buffer in configuration includes. The alias has to be unique within all ring buffers.

## Generate

Define settings regarding the creation of the ring buffer.



- **Mode**

Choose between continuous data acquisition or triggered data acquisition.

- **Start- and Pause-trigger**

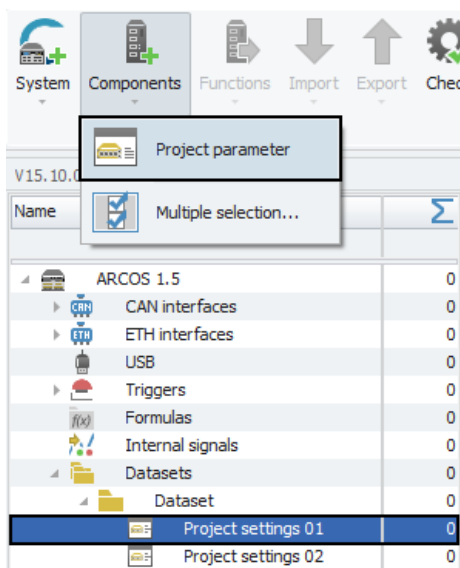
This field allows you, to set triggers to start or pause ring buffer creation. If you do not set a start trigger, then by default ring buffer creation is always active during measurement.

## 14.3 Dataset Project settings

The “Project settings” filetype is meant for including Project information such as company name, serial number, project name etc. in your dataset. Multiple “Project settings” files can be included in your dataset. The “Project settings” filetype for dataset creates a set of project parameters that will be included only in the respective dataset. In order to define “Project settings” that apply globally to the entire configuration and that get exported with every dataset, please refer to (→5).

### 14.3.1 Adding project parameters

It is possible, to add user-specific project parameters in addition to the default project parameters. To do so, select the desired “Project settings xx” element in the measurement task tree, select the “Components” button in the Ribbon and then choose “Project parameter”.

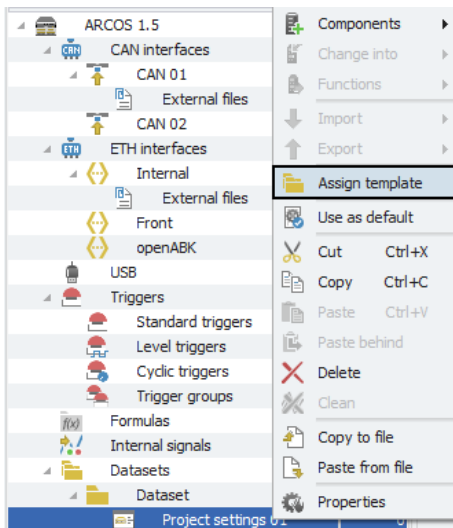


The new parameter will appear in the respective “Project settings xx” Grid area as customizable parameter in the table.

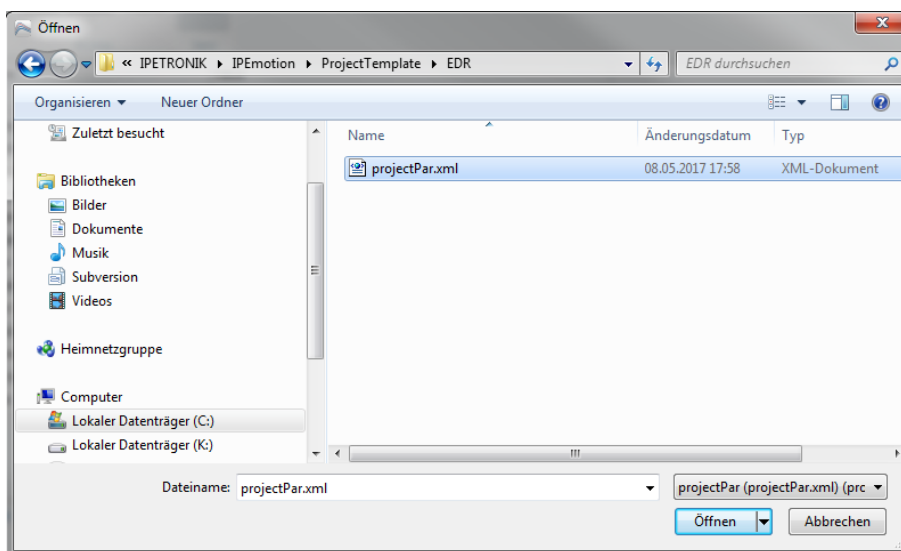
File name	
Date	27.06.2018 10:18:39
Project parameter 14	

### 14.3.2 Assigning a template of project parameters

It is also possible, to assign a template which contains a predefined set of user-specific project parameters. To do so, right-click on the "Project settings xx" tree element and then choose "Assign template".

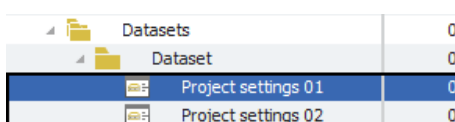


In the following window you may choose the template file and confirm with "Open". The file type needs to be .xml.



### 14.3.3 Tree elements for Project settings

Including a "Project settings" file in your dataset will add one new child element per included "Project settings" file to your tree element "Dataset". The tree element is labeled "Project settings xx".



Changes and errors excepted.

### 14.3.4 Grid area for Project settings

If a "Project settings xx" element is selected in the Measurement task tree, the grid area will provide you with a table, that allows you to access all default or previously defined user-specific project parameters.

Name	Value
Company name	
Serial number	
Manufacturer	
Project name	
Project manager name	
E-mail address project manager	
User	
E-mail address user	
User login	bachl
Description	
IPEmotion version	V08.00.01
File name	
Date	27.06.2018 10:18:39
Project parameter 14	

### 14.3.5 Details area for Project settings

If a "Project settings xx" element has been selected in the measurement task tree, additional settings are available in the details area.

#### General

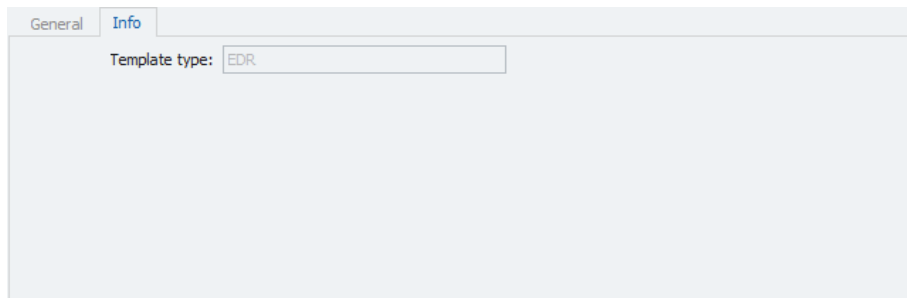
This tab provides general settings for the selected Project settings file.

General	Info
Active:	<input checked="" type="checkbox"/>
Name:	Project settings 01
Description:	Project settings
Reference:	Project settings 01/ARCOS 1.5

- **Name**  
Give a user-defined Name to the selected formula/signal.
- **Description**  
Give a user-defined description to the selected formula/signal.
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

### Info

Tells you the type of template that has been assigned.



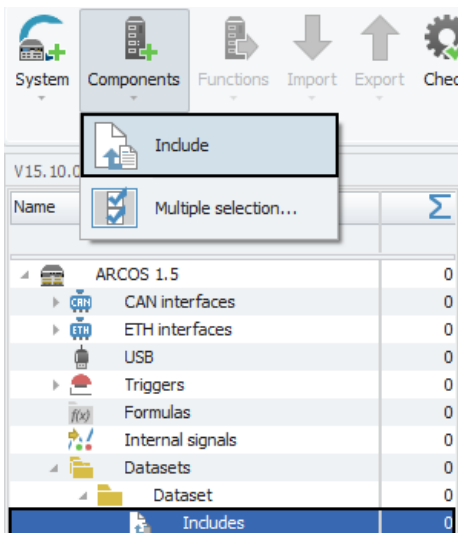
## 14.4 Includes

The “Includes” filetype allows you to include partial configurations in your dataset. This can be especially helpful for components of a configuration that are likely to change over time, such as Wifi accesspoints, and are used by a large number of loggers at the same time.

### 14.4.1 Adding the Includes-Interface

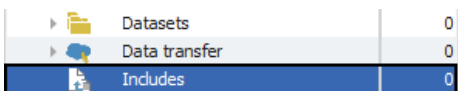
Adding the “Includes” filetype as described in the chapter “Setting up a dataset” (→ 14.1.4), will add the “Includes” interface to your dataset.

Once the “Includes” interface has been added to your dataset, you can then add multiple “Includes”. To do so, select the “Includes” interface in the tree, click the “Components” button in the Ribbon and then choose “Include”.



### 14.4.2 Tree elements for Includes

After having added the “Includes” interface to your dataset it will appear as a child element to the dataset with the name “Includes”.



### 14.4.3 Grid area for Includes

If the “Includes” interface is selected in the tree, the Grid area will present you with an overview of the Includes which have been added to your system so far.

Also you can find here two important functions, which are the “**Column chooser**” (→4.3.1) and the “**Filter editor**” (→4.3.2).

Name	Active	Description	Mode	Path	Copy to dataset
Include 01	<input checked="" type="checkbox"/>	Included partial configuration	Include a specific file		<input checked="" type="checkbox"/>
Include 02	<input checked="" type="checkbox"/>	Included partial configuration	Include a specific file		<input checked="" type="checkbox"/>

### 14.4.4 Details area for Includes

The Details area provides settings for the Include, that has been selected in the grid area.

#### General

Please refer to (→4.2.2).

#### Settings

This tab provides settings regarding the include file.

General Settings

Copy to dataset:

Mode: Include a specific file

Path:

Local:

- **Copy to dataset**

Activating this setting will include a copy of this include file in the dataset for traceability.

- **Mode**

Define, whether you want to include a specific file or the entire directory of the include path.

- **Path**

Define the cfginclude file path relative to “(cfgdir (see data transfer))/includes/”. The file path must end with a slash (/).



## 14.5 AFX

The “AFX” filetype is meant for recording of signals. It is compatible with all signals that produce values that can be represented on a 2-D graph. It is not compatible with video or audio files.

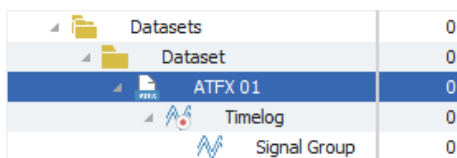
The recorded signal values will be represented in a timelog.

### 14.5.1 Tree elements for AFX

Including the “AFX” filetype in your dataset will add three new child elements to your tree element “Dataset”:

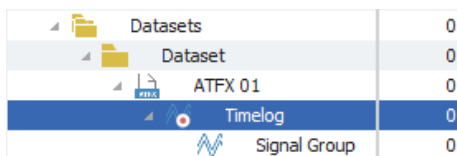
- **AFX xx**

This element represents the AFX file, which will later be included in your exported dataset. You may add multiple files of the same filetype.



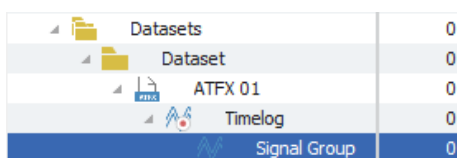
- **Timelog**

This element represents the timelog for recording signal values.



- **Signal Group**

The signal is a group of all the signals that the timelog will record and that will ultimately be included in the exported dataset.



### 14.5.2 Grid area for AFX

If the “Signal Group” is selected in the Measurement task tree, the grid area will show an overview of the signals included in the “Signal Group”

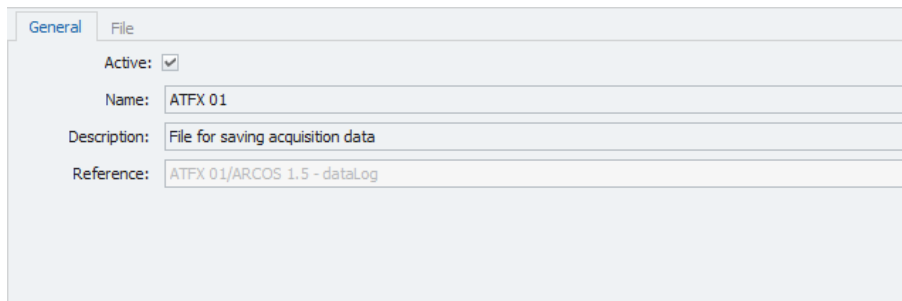
### 14.5.3 Details area for AFX

The details area contains settings for the behaviour of your AFX file, timelog or signal group.

#### 14.5.3.1 AFX file

##### General

This tab provides general settings for the selected AFX file.



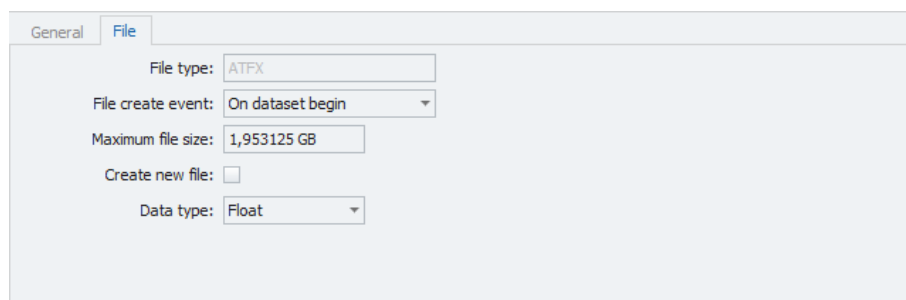
The screenshot shows a dialog box with two tabs: 'General' (selected) and 'File'. The 'General' tab contains the following fields:

- Active:** A checked checkbox.
- Name:** A text field containing 'AFX 01'.
- Description:** A text field containing 'File for saving acquisition data'.
- Reference:** A text field containing 'AFX 01/ARCOS 1.5 - dataLog'.

- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected formula/signal.
- **Description**  
Give a user-defined description to the selected formula/signal.
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## File

This tab provides settings regarding the creation of the file.



- **File type**

Tells you the type of the created file.

- **File create event**

Define, when the AFX file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset ist started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.5.3.2</a> ). This will result in a splitting of the current AFX file into multiple files, as a new file is created for each timethe trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the timelog settings → <a href="#">14.5.3.2</a> . Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the timelog settings → <a href="#">14.5.3.2</a> . Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party analysis tools cannot handle files, that are larger.

- **Create new file**

If this box is marked active, a new file will be created, if the current file exceeds the maximum file size.

- **Data type**

This dropdown menu allows you to switch the data type between float or double.

### 14.5.3.2 AFX Timelog

#### General

This tab provides general settings for the selected AFX timelog.

General	Trigger	User
Active:	<input checked="" type="checkbox"/>	
Name:	Timelog	
Description:	Contains one/multiple signal group(s)	
Reference:	Timelog/AFX 01/ARCOS 1.5 - dataLog	

- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.

- **Description**

Give a user-defined description to the selected file.

- **Reference**

This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

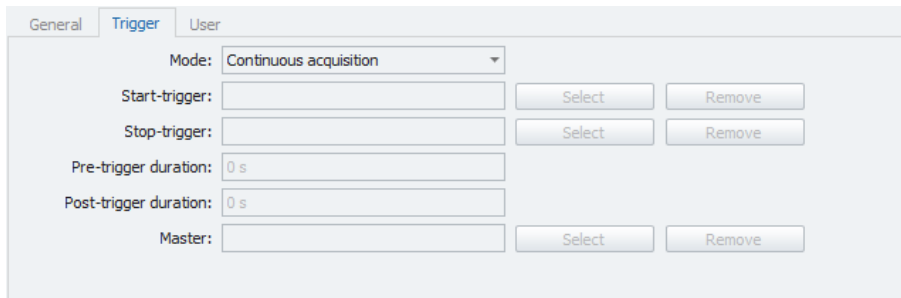


In the field "Name" project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## Trigger

This tab provides settings regarding the trigger for the start and stop of the timelog. When the timelog is started, all signals contained in the signal group will be stored to the ATFX file according to their settings.

Furthermore will this trigger provoke the creation of the ATFX file, if you have chosen either "On trigger" or "On first trigger" as "File create event".



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the timelog.

- **Stop-trigger**

Define a trigger, that will stop the timelog.

- **Pre-trigger duration**

Pre-trigger duration allows you to define, how long before the start trigger was set, the timelog will start.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, the timelog will stop.

## User

The field "Identifier" allows you to give a user identifier to the timelog. It does not have any effect other than helping the user identify a specific timelog.

## 14.5 AFX

General Trigger **User**

Identifier:

### 14.5.3.3 AFX Signal Group

#### General

This tab provides general settings for the selected AFX timelog.

General **Storage**

Active:

Name:

Description:

Reference:

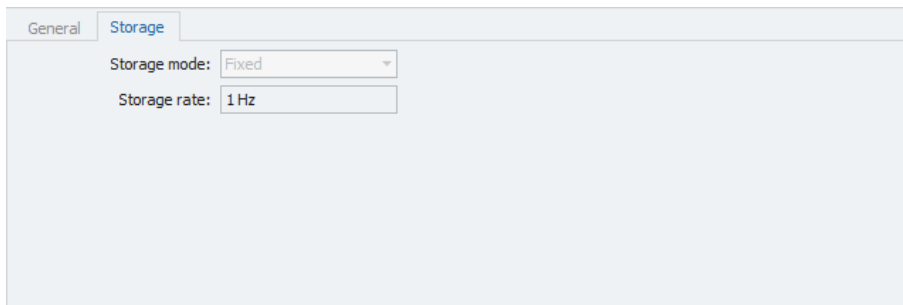
- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected formula/signal.
- **Description**  
Give a user-defined description to the selected formula/signal.
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

#### Storage

This tab allows for setting regarding the storage of the contained signals inside the AFX file.

- **Storage mode**  
The storage mode in AFX is fixed. It cannot be changed.

Changes and errors excepted.



- **Storage rate**

The storage rate defines how often the signals contained in this “Signal Group” will be stored.

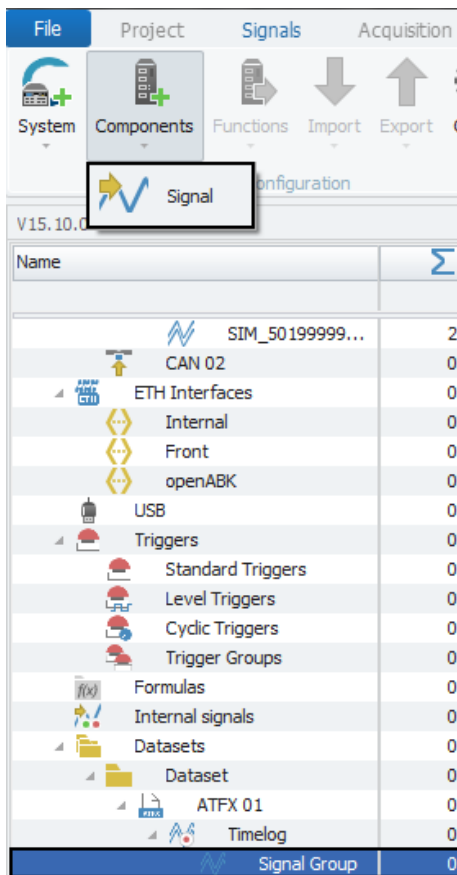


Storage rates with decimal places will be rounded to three decimal places.

### 14.5.4 Working with Signal Groups for AFX

For filetypes intended for signal recording such as AFX, MDF 4.0 and MDF 4.1, signals need to be included in a "Signal Group" belonging to the timelog, to which you would like the signal to be included. The signals, that are included in a "Signal Group" can then be stored.

To achieve this, select the "Signal group" in the "Measurement task tree", click the "Components" button in the Ribbon and choose "Signal".





## 14.5 ATFX

In the following window you may choose all the available signals, that you wish to include within this group and confirm by clicking "OK".

Channel selection: Signal Group

Name	Unit	Phys Min	Phys Max	Sensor Min	Sensor Max	Sampling rate	Special function
Pressure_Abs	bar	0,00000	2,00000	0	65535	1 Hz	None
Pressure_Rel	bar	-2,00000	2,00000	0	65535	1 Hz	None
MAP	bar	0,00000	3,00000	0	65535	1 Hz	None
TPS_Volt	V	-8,0000	8,0000	0	65535	1 Hz	None
Front_left	C	-50,000	200,000	0	65535	1 Hz	None
Front_right	C	-50,000	200,000	0	65535	1 Hz	None
Rear_left	C	-50,000	200,000	0	65535	1 Hz	None
Rear_right	C	-50,000	200,000	0	65535	1 Hz	None
Exhaust_1	C	-50,00	1200,00	0	65535	1 Hz	None
Exhaust_2	C	-50,00	1200,00	0	65535	1 Hz	None

1 of 10 selected

OK Cancel



ATFX can only contain one "Signal Group" and this "Signal Group" can only have one storage rate. That means, all signals, that are contained in the same "Signal Group", will be stored with the same storage rate.

## 14.6 MDF 4.0

The “MDF 4.0” filetype is meant for recording of signals. It is compatible with all signals that produce values that can be represented on a 2-D graph. It is not compatible with video or audio files.

The recorded signal values will be represented in a timelog.

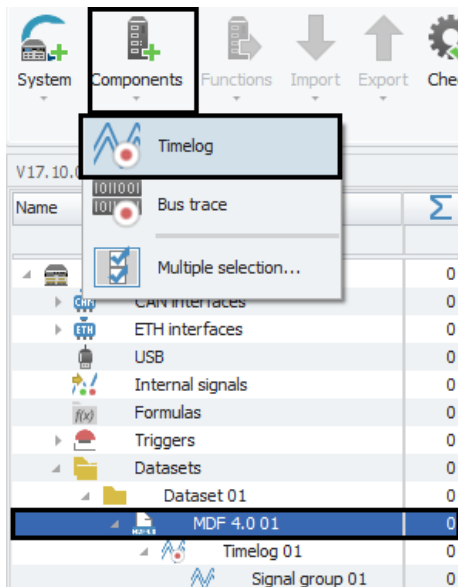


MDF 4.0 and MDF 4.1 files can have (Other than ATFX) multiple “Signal Groups” and some of those groups allow for include signals to be stored in various rates.

### 14.6.1 Multiple timelogs in MDF 4.0

It is possible, to configure multiple timelogs within one MDF file. All timelogs that are configured within one MDF file will later be stored in said MDF file.

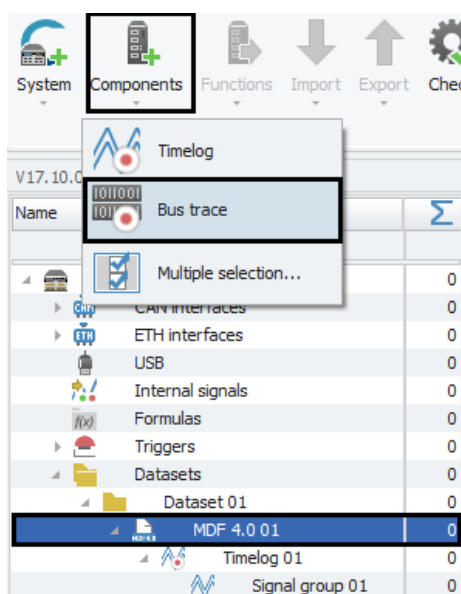
To add another timelog in your MDF file, select the MDF file in the measurement task tree, click the “Components” button in the Ribbon and then select “Timelog”.



### 14.6.2 Bus trace in MDF 4.0

It is possible, to configure a bus trace within an MDF file. This bus trace will be stored alongside other contained timelogs or bus traces within the MDF file. The bustrace for MDF works the same as regular bus tracing methods, such as “Vector BLF” or “Vector ASCII”, for details please refer to the chapter “Vector BLF/Vector ASCII/Vector ASCII compressed” (→14.8).

To add a bus trace in your MDF file, select the MDF file in the measurement task tree, click the “Components” button in the Ribbon and then select “Bus trace”.



### 14.6.3 Tree elements for MDF 4.0

Including the “MDF 4.0” filetype in your dataset will add three new child elements to your tree element “Dataset”:

- **MDF 4.0 xx**

This element represents the MDF 4.0 file, which will later be included in your exported dataset. You may add multiple files of the same filetype.

└─ Datasets	0
└─ Dataset	0
└─ MDF 4.0 01	0
└─ Timelog	0
└─ Signal Group	0
└─ Signal Group	0

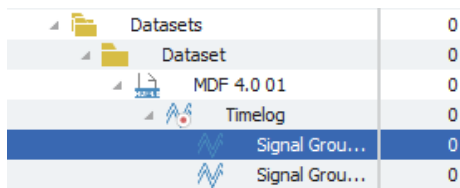
- **Timelog**

This element represents the timelog for recording signal values.

└─ Datasets	0
└─ Dataset	0
└─ MDF 4.0 01	0
└─ Timelog	0
└─ Signal Group	0
└─ Signal Group	0

- **Signal Group**

The signal is a group of all the signals that the timelog will record and that will ultimately be included in the exported dataset.



#### 14.6.4 Grid area for MDF 4.0

If the "Signal Group" is selected in the Measurement task tree, the grid area will show an overview of the signals included in the "Signal Group".

Also you can access settings regarding the signals sampling rate and typ and storage rate. If the signal group's storage mode is set to "Individual" (→ 14.6.5.3) the column "Sample type" allows you to set each signal's sample type.

Channel	Index	Active	Unit	Sampling rate	Storage rate	Sample type
Pressure_Abs	1	<input checked="" type="checkbox"/>	bar	1 Hz	1 Hz	On receive
Pressure_Rel	2	<input checked="" type="checkbox"/>	bar	1 Hz	1 Hz	On value change
MAP	3	<input checked="" type="checkbox"/>	bar	1 Hz	1 Hz	Periodic
TPS_Volt	4	<input checked="" type="checkbox"/>	V	1 Hz	1 Hz	Periodic

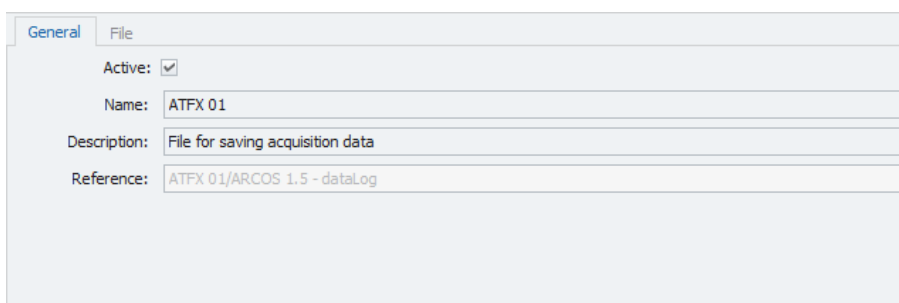
#### 14.6.5 Details area for MDF 4.0

The details area contains settings for the behaviour of your MDF 4.0 file, timelog or signal group.

##### 14.6.5.1 MDF 4.0 File

###### General

This tab provides general settings for the selected MDF 4.0 file.



- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.

- **Description**

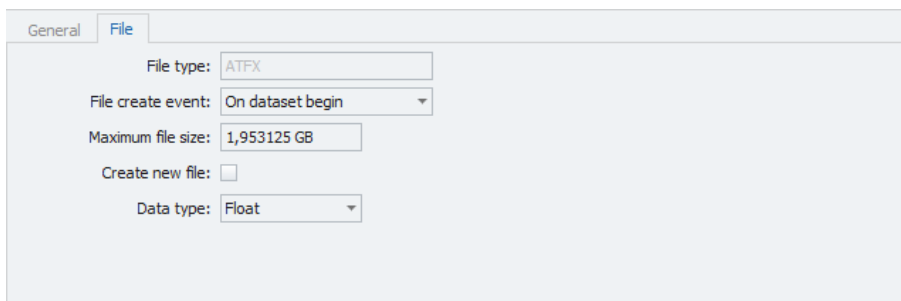
Give a user-defined description to the selected file.

- **Reference**

This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## File

This tab provides settings regarding the creation of the file.



The screenshot shows a software interface with two tabs: 'General' and 'File'. The 'File' tab is active. It contains the following settings:

- File type: ATFX
- File create event: On dataset begin
- Maximum file size: 1,953125 GB
- Create new file:
- Data type: Float

- **File type**

Tells you the type of the created file.

- **File create event**

Define, when the MDF 4.0 file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset is started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.6.5.2</a> ). This will result in a splitting of the current ATFX file into multiple files, as a new file is created for each time the trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.6.5.2</a> ). Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.6.5.2</a> ). Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party evaluation tools cannot handle files, that are larger.

- **Create new file**

If this box is marked active, a new file will be created, if the current file exceeds the maximum file size.

- **Data type**

This dropdown menu allows you to switch the data type between float or double.

## 14.6.5.2 MDF 4.0 Timelog

### General

This tab provides general settings for the selected MDF 4.0 timelog.



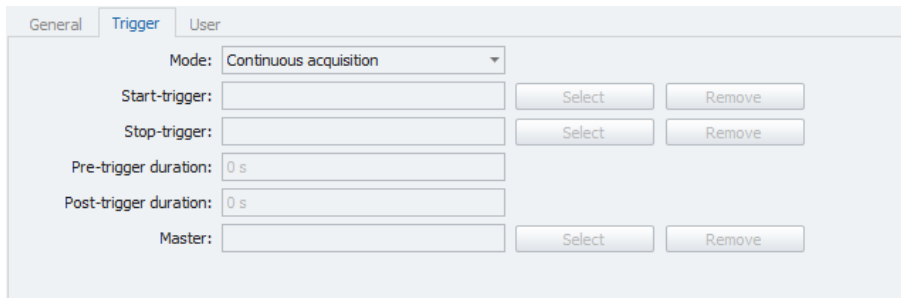
General	Trigger	User
Active:	<input checked="" type="checkbox"/>	
Name:	Timelog	
Description:	Contains one/multiple signal group(s)	
Reference:	Timelog/ATFX 01/ARCOS 1.5 - dataLog	

- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## Trigger

This tab provides settings regarding the trigger for the start and stop of the timelog. When the timelog is started, all signals contained in the signal group will be stored to the MDF 4.0 file according to their settings.

Furthermore will this trigger provoke the creation of the MDF 4.0 file, if you have chosen either "On trigger" or "On first trigger" as "File create event".



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the timelog.

- **Stop-trigger**

Define a trigger, that will stop the timelog.

- **Pre-trigger duration**

Pre-trigger duration allows you to define, how long before the start trigger was set, the timelog will start.

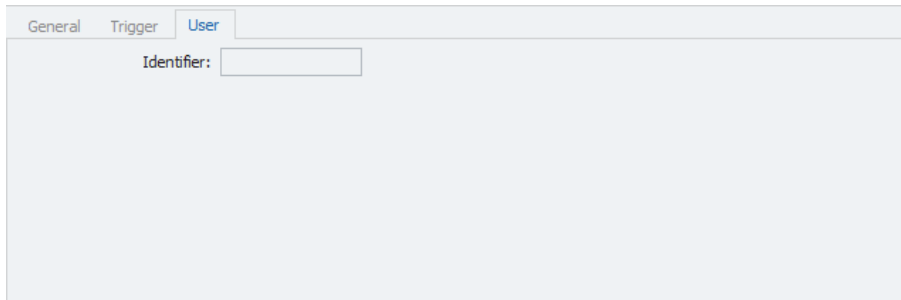
- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, the timelog will stop.



## User

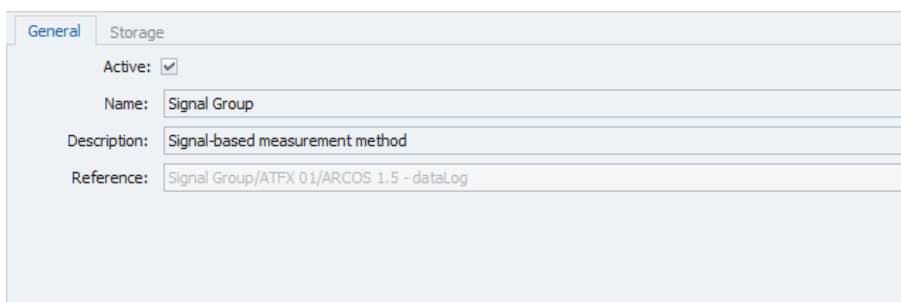
The field “Identifier” allows you to give a user identifier to the timelog. It does not have any effect other than helping the user identify a specific timelog.



### 14.6.5.3 MDF 4.0 Signal Group

#### General

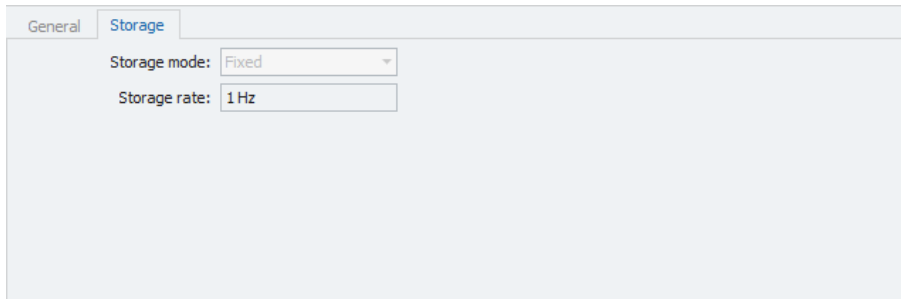
This tab provides general settings for the selected MDF 4.0 timelog.



- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected formula/signal.
- **Description**  
Give a user-defined description to the selected formula/signal.
- **Reference**  
This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

## Storage

This tab allows for setting regarding the storage of the contained signals inside the MDF 4.0 file.



- **Storage mode**

Storage mode	Characteristics
Fixed	The rate, at which the signals included in the "Signal Group" will be stored, is the same for all included signals. You may set the rate below at "Storage Rate"
From channel	The rate, at which the signals included in the "Signal Group" will be stored, is the same as each signal's source channel. This may result in a "Signal Group" with different storage rates for different signals, according to their source channel's sampling rate.
Individual	The rate, at which the signals included in the "Signal Group" will be stored, can be individually set for each signal. This may result in a "Signal Group" with different storage rates for different signals. In "Individual" mode, the sample type for the single signals can be set in the grid area (→ <a href="#">14.6.4</a> ).
On receive	A signal with the storage mode "On receive" will be stored, whenever it is received by the logger. The setting for the storage rate can be ignored.
On value change	A signal with the storage mode "On value change" will be stored, whenever the incoming value of the signal is different than the previous one. The setting for the storage rate can be ignored.

- **Storage rate**

The storage rate defines how often the signals contained in this "Signal Group" will be stored.



Storage rates with decimal places will be rounded to three decimal places.

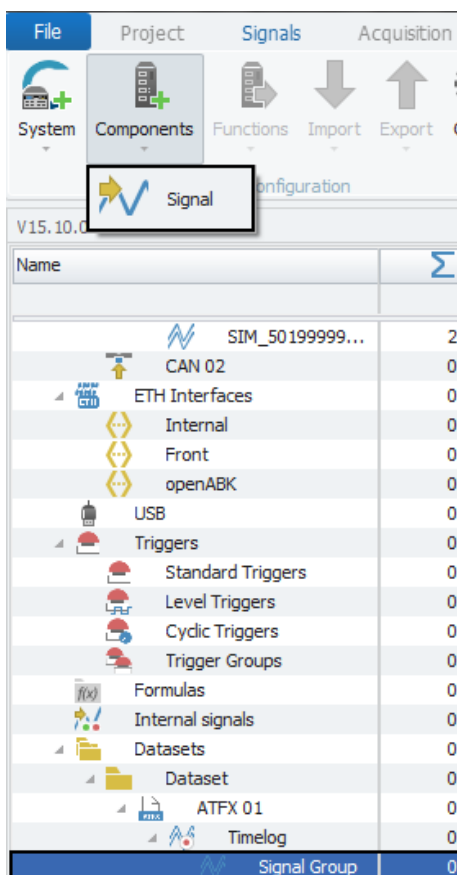
### 14.6.6 Working with Signal Groups for MDF 4.0

For filetypes intended for signal recording such as ATFX, MDF 4.0 and MDF 4.1, signals need to be included in a "Signal Group" belonging to the timelog, to which you would like the signal to be included. The signals, that are included in a "Signal Group" can then be stored.

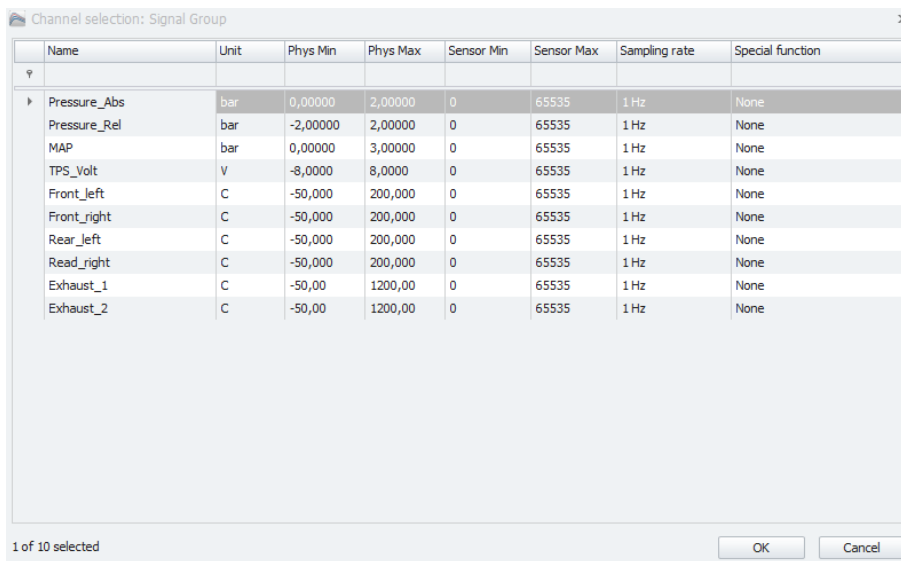
Other than for ATFX files, MDF 4.0/4.1 files support multiple "Signal Groups" and storage modes with different characteristics and therefore allow for a more flexible storage of your data. For details on the different storage modes please refer to (→ [14.6.5.3](#)).

This allows for example to create one "Signal Group" with a fixed storage rate of 10Hz, one that stores signals according to their channel's sampling rate and one that has an individual storage rate for each signal.

To include signals within one group, select the "Signal group" in the "Measurement task tree", click the "Components" button in the Ribbon and choose "Signal".



In the following window you may choose from all the available signals the ones, that you wish to include within this group and confirm by clicking "OK".



Even though the Plugin allows for different storage rates within one "Signal Group" in the case of the storage modes "From channel" and "Individual", technically each rate will be exported as a separate group in the loggerconfig.

That means, if you have a "Signal Group" called "**examplegroup**" with three different storage rates (1Hz, 10Hz, 100Hz), at export will be created three groups named "examplegroup\_1Hz", "examplegroup\_10Hz" and "examplegroup\_100Hz" (namemethod: signalgroup\_storage\_rate).

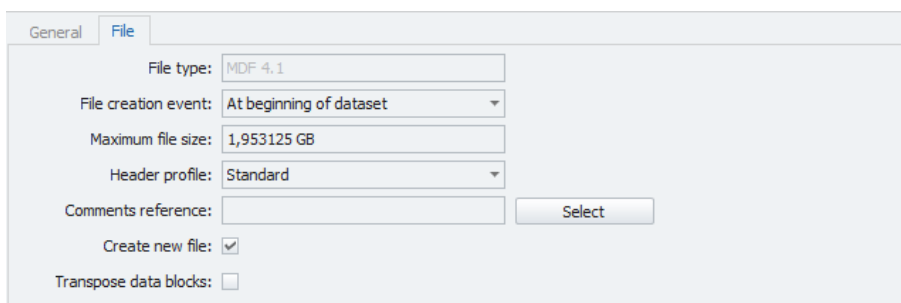
## 14.7 MDF 4.1

MDF 4.1 can largely be treated the same as MDF 4.0, with some differences, that will be explained in the following.

### 14.7.1 File compression in MDF 4.1

MDF 4.1 files are by default compressed. The standard compression rate is 7-10.

For further compression MDF 4.1 offers the option "Transpose data blocks". To access this option, select the tree element "MDF 4.1 xx" and open the "File" tab in the details section.

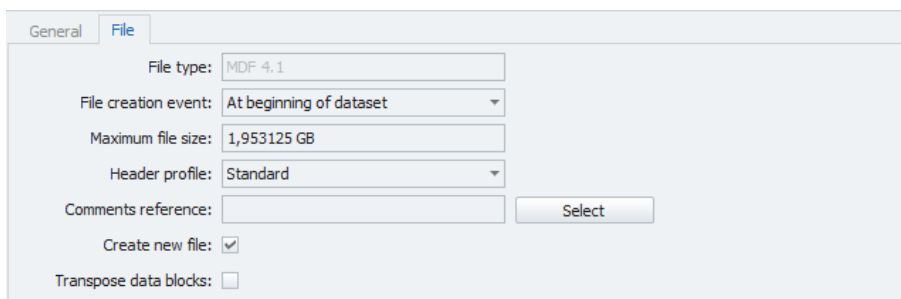


### 14.7.2 Header profiles in MDF 4.1

The dropdown menu "Header profile" in the "File" tab of your MDF 4.1 file allows you to assign a header profile to the file. The available profiles are:

- Standard
- CANape
- MDA
- EDR

If an MDF 4.1 file, that has been created by a dataLog 2015.10.xx or 2015.14.xx, is being read, the Header profile will be automatically set to EDR.

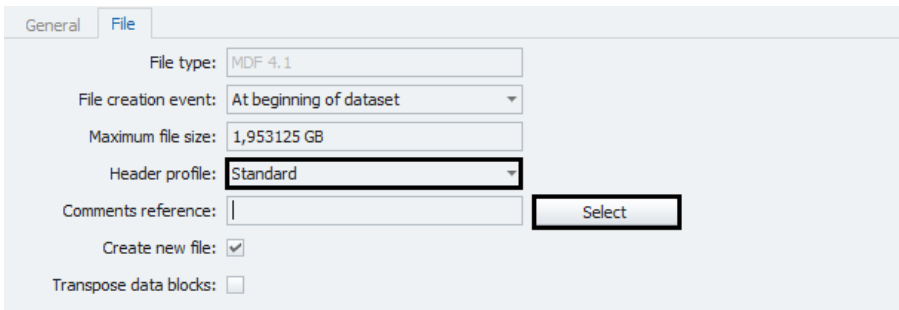



If the header profile is set to EDR and an EDR template has been assigned in the project settings (→5.2), the EDR profile will automatically fetch the required information from the project settings.

### 14.7.2.1 Comments reference

For the header profile Standard, CANape and MDR the option “Comments reference” allows to reference comments and values from Project settings. The pairs of names and values contained in the Project settings will be written as header block common properties. For more information regarding “Project settings” please refer to (→5).

To reference comments, select one of the three supported Header profiles, click the “Select” button next to the field “Comments reference” and select the “Project settings” file, that you wish to reference.



### 14.7.2.2 Overview of header profiles and their differences

#### HD-Block

Field	Standard	CANape	MDA	EDR
hd_md_comments: <common_properties>	Comments reference (→14.7.2.1)	Comments reference (→14.7.2.1)	Comments reference (→14.7.2.1)	-

#### CG-Block

Field	Standard	CANape	MDA	EDR
si_tx_name	-	-	-	“CAETEC ARCOS” / “CAETEC ARCOS 1.5” / “CAETEC μCros” / “CAETEC μCros 1.1”
si_tx_path	-	-	-	Frontnumber
si_md_comment: <common_properties>	-	-	-	“StorageGroup”, “StorageRate”, “TargetFile”

*CN-Block (time channel)*

Field	Standard	CANape	MDA	EDR
cn_tx_name	"time"	"t"	"time"	"time"
si_tx_name	-	-	-	name of signal group
si_tx_path	-	-	-	Intervall

*CN-Block (data channel)*

Field	Standard	CANape	MDA	EDR
cn_md_comment: <names><display>	-	-	Bus: "signal name/bus name" SO-MEIP/OBD/UDS/CCP/XCP: "signal name/station name"	-
si_tx_name	station-/bus name or "CA-ETEC dataLog (signal type)"	Namespace	Namespace	ECU from source
si_tx_path	Namespace	station-/bus name or "CA-ETEC dataLog (signal type)"	station-/bus name or "CA-ETEC dataLog (signal type)"	Intervall
si_md_comment: <path><name>	-	-	-	Interface type, PN, SN, relative channel number
si_md_comment: <names> <description>	-	-	-	Signal type description according to table "EDR Use Cases Naming"
si_md_comment: <protocol>	"CCP" / "XCP" / "KWP" / "UDS"	"CCP" / "XCP" / "KWP" / "UDS"	"CCP" / "XCP" / "KWP" / "UDS"	"CCP" / "XCP" / "KWP" / "UDS" / "FreeRunning" / "NMEA"
si_md_comment: <common_properties>	-	-	-	Bus/OBD/UDS: "file", "MessageID"; SomelP: "BusFileName", "Identifier"; CCP/XCP: "file", "daq"

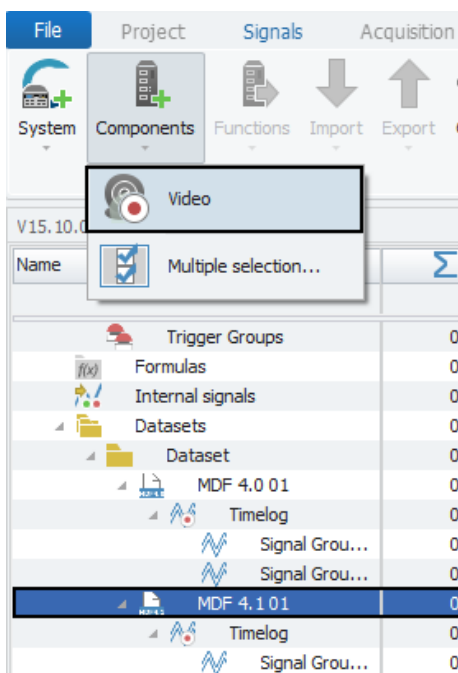
### 14.7.3 Video attachments in MDF 4.1

When working with an MDF 4.1 filesystem, you can attach "Video Streams" to your timelog. As the video is only an attachment, it can not have its own triggers nor duration settings. The video-file will be attached in the \*.avi format and have the same filename as the MDF file it belongs to. The MDF file knows about the existence and automatically synchronizes the timelog and video.

#### 14.7.3.1 Attaching a video

In order to attach a video, you will first need to add the "Video" component to your MDF 4.1 file.

Select the tree element "MDF 4.1 xx", click the "Components" button in the Ribbon and then choose "Video".



Once the video component has been added to your MDF file, you can choose a video signal, that you wish to attach.

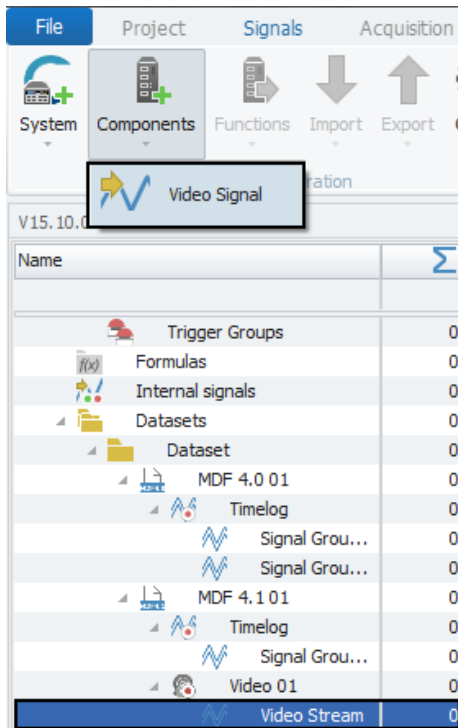
Select the new tree element "Video stream", click the "Components" button in the Ribbon and then choose "Video signal".

In the resulting window you will be presented with an overview of all the available video signals. Choose the one, you wish to attach, and confirm with "OK".



It is possible to attach multiple videos by adding an additional "Video Stream". To do so, select the "Video" tree element, click on the "Components" button in the ribbon and choose "Video Stream".



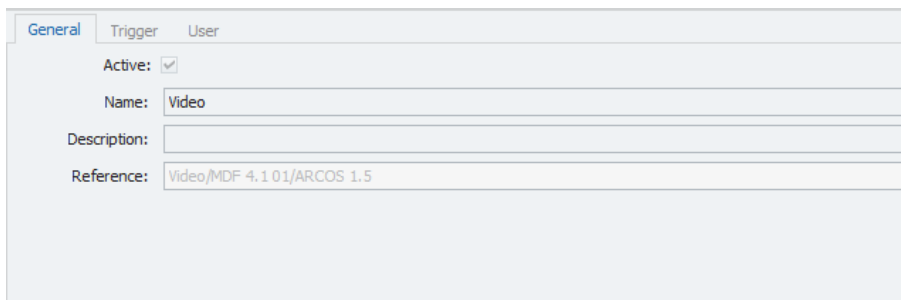


### 14.7.3.2 Details area for video in MDF 4.1

This section will explain the relevant settings for video attachments in MDF 4.1

#### Video xx General

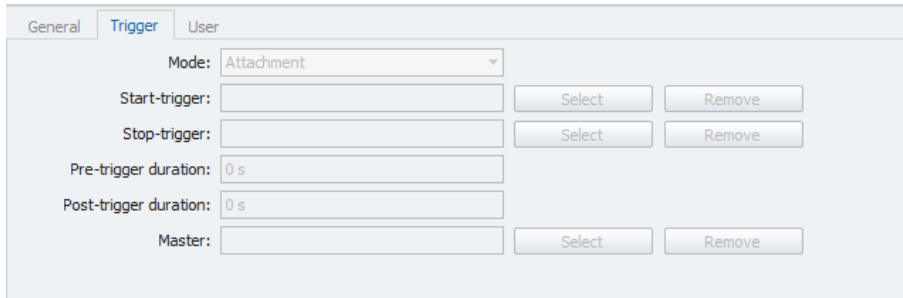
The tab “General” for the tree element “Video xx” allows you to activate or deactivate the video and give a user specific name.



In the field “Name” project parameters can be used as variables. For more information please refer to (→5.6).

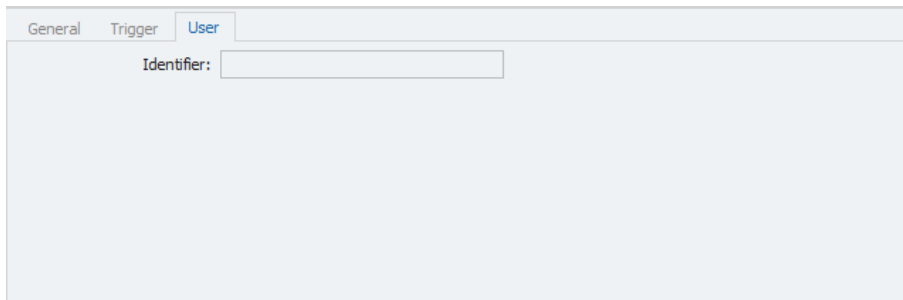
### Video xx Trigger

The tab "Trigger" for the tree element "Video xx" can mainly be neglected. The only relevant field of this tabsheet is the "Mode"-field. It is read-only and shows you that your video is an attachment to the MDF-file.



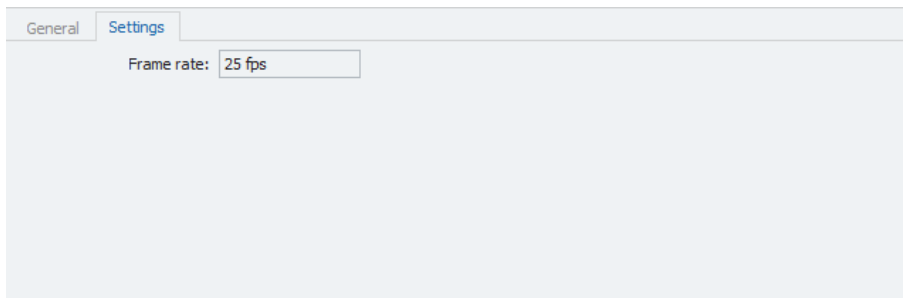
### Video xx User

The tab "User" for the tree element "Video xx" allows you to define a user-specific identifier to the video attachment. This identifier helps the user to later identify a specific video attachment.



### Video Stream Settings

The tab "Settings" for the tree element "Video Stream" allows you to set the framerate for the video.




## 14.8 Vector BLF / Vector ASCII / Vector ASCII compressed

These three filetypes are equal in functionality and differ only in the final exported file. The trace method records all the messages that arrive on the input bus (CAN, LIN, FlexRay). Regardless of the signals defined, all the messages are recorded. Filter rules can be defined to reduce the data volume. A typical trace application is the acquisition of all raw data in order to later evaluate the total traffic on the channel. Unlike most of the other methods, traces are event-oriented. This means the messages are not retrieved from the channels according to a set time pattern, but are recorded as soon as they arrive on the channel. This method accordingly has no parameter for sampling rate.



Including a bus tracing filetype in your dataset will produce a "Warning" message, saying that "at least one channel must be set active".

Symbol	Time	Type	Source	Message
	22.02.2018 10:39:04,991	WARNING	Bus trace	At least one channel must be set active

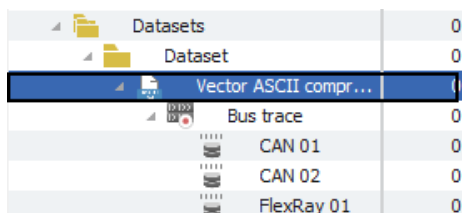
Please refer to the section "Bus trace" (→ [14.8.3.2](#)), in order to activate a channel for tracing.

### 14.8.1 Tree elements for bus tracing

Including a bus tracing filetype in your dataset will add various new child elements to your tree element "Dataset":

- **Vector BLF / Vector ASCII / Vector ASCII compressed xx**

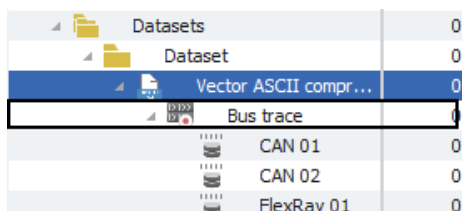
This element represents the bus tracing file, which will later be included in your exported dataset. You may add multiple files of the same filetype.



Datasets	0
Dataset	0
Vector ASCII compr...	0
Bus trace	0
CAN 01	0
CAN 02	0
FlexRay 01	0

- **Bus trace**

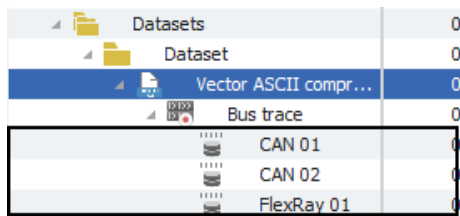
This element represents the "Bus trace" for recording all the traffic on a selected bus channel.



Datasets	0
Dataset	0
Vector ASCII compr...	0
Bus trace	0
CAN 01	0
CAN 02	0
FlexRay 01	0

- **Bus channels available for tracing**

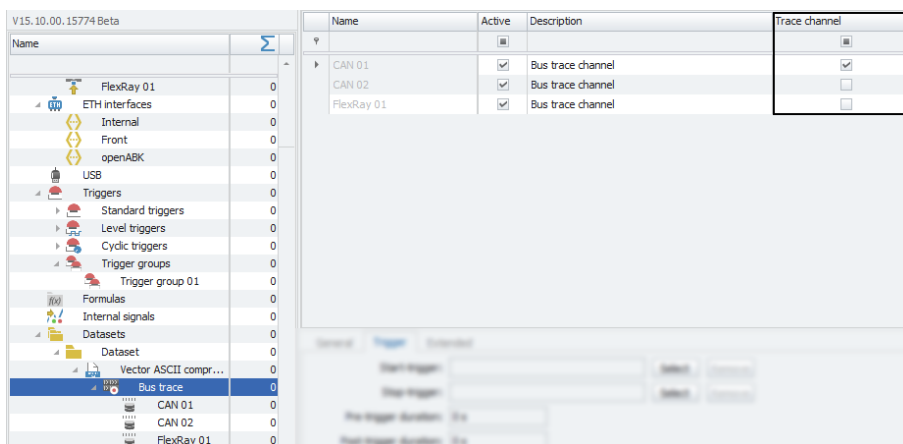
As child elements to the tree element “Bus trace” will appear all the Bus channels which are currently available for tracing.



### 14.8.2 Grid area for bus tracing

If the “Bus trace” is selected in the Measurement task tree, the grid area will show an overview of the Bus channels available for tracing.

At least one channel must be marked active for tracing, by ticking the checkbox labeled “Trace channel”.



If an ID Filter has been added for a Bus trace channel, selecting this Bus trace channel in the tree will show an overview of the existing ID Filters in the grid area.

For information on working with ID Filters for Bus trace, please refer to the chapter **Bus trace ID Filter** (→ [14.8.4](#)).

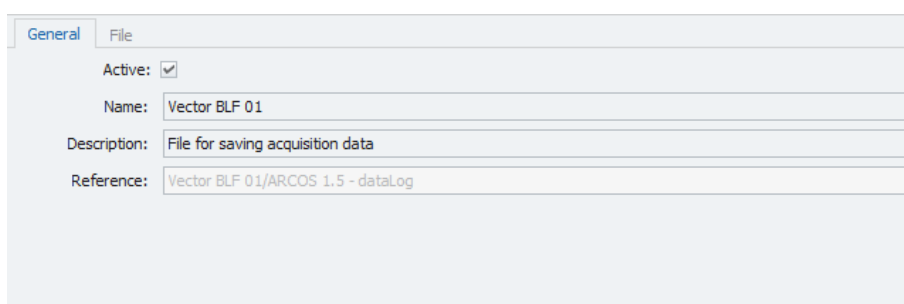
### 14.8.3 Details area for bus tracing

The details area contains settings for the behaviour of your bus tracing file (Vector BLF / Vector ASCII / Vector ASCII compressed), the “Bus trace” component or a traceable Bus channel.

#### 14.8.3.1 Bus tracing file

##### General

This tab provides general settings for the selected bus tracing file.



General	File
Active:	<input checked="" type="checkbox"/>
Name:	Vector BLF 01
Description:	File for saving acquisition data
Reference:	Vector BLF 01/ARCOS 1.5 - dataLog

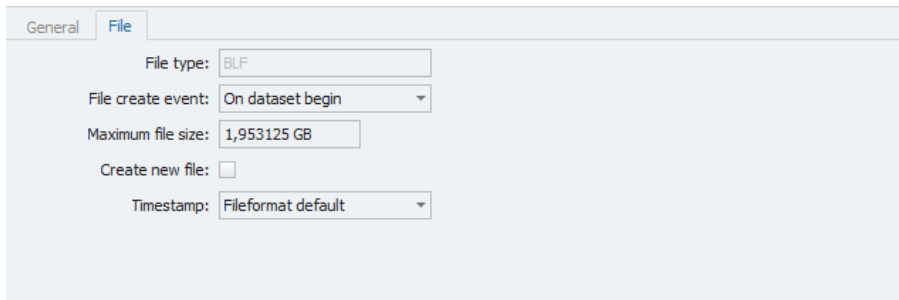
- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.



In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## File

This tab provides settings regarding the creation of the file.



- **File type**  
Tells you the type of the created file.
- **File create event**  
Define, when the bus tracing file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset ist started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.8.3.2</a> ). This will result in a splitting of the current ATFX file into multiple files, as a new file is created for each timethe trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.8.3.2</a> ). Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.8.3.2</a> ). Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party evaluation tools cannot handle files, that are larger.

- **Create new file**

If this box is marked active, a new file will be created, if the current file exceeds the maximum file size.

- **Timestamp**

This dropdown menu allows you to set the format of the timestamp for the file.

### 14.8.3.2 Bus trace

#### General

This tab provides general settings for the selected "Bus trace".



General	Trigger	Extended
Active:	<input checked="" type="checkbox"/>	
Name:	Bus trace	
Description:	Contains one/multiple bus trace(s)	
Reference:	Bus trace/Vector BLF 01/ARCOS 1.5 - dataLog	

- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.

- **Description**

Give a user-defined description to the selected file.

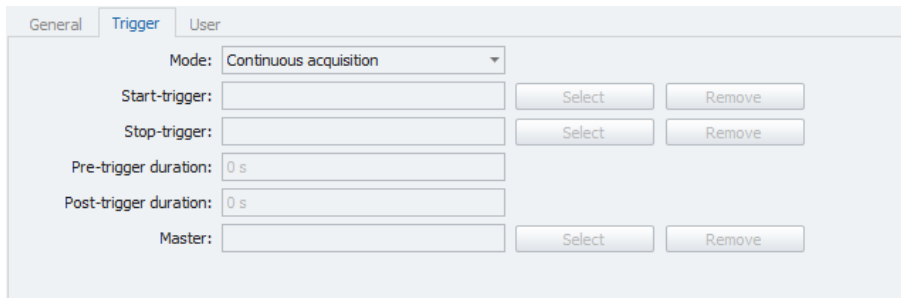
- **Reference**

This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## Trigger

This tab provides settings regarding the trigger for the start and stop of the “Bus trace”. When the “Bus trace” is started, all traffic on the channel will be stored to the bus tracing file.

Furthermore will this trigger provoke the creation of the bus tracing file, if you have chosen either “On trigger” or “On first trigger” as “File create event”.



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the “Bus trace”.

- **Stop-trigger**

Define a trigger, that will stop the “Bus trace”.

- **Pre-trigger duration**

Pre-trigger duration allows you to define, how long before the start trigger was set, the “Bus trace” will start.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, the “Bus trace” will stop.

## Extended

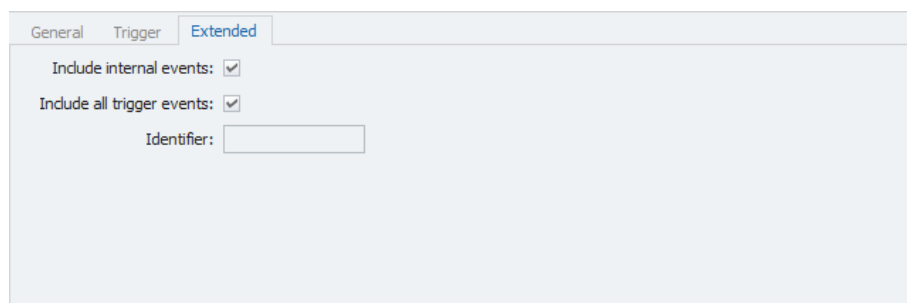
This tab provides extended settings for “Bus trace”.

- **Include internal events**

Define whether the occurrence of internal events should be stored in the trace data.

Changes and errors excepted.





General Trigger **Extended**

Include internal events:

Include all trigger events:

Identifier:

- **Include all trigger events**

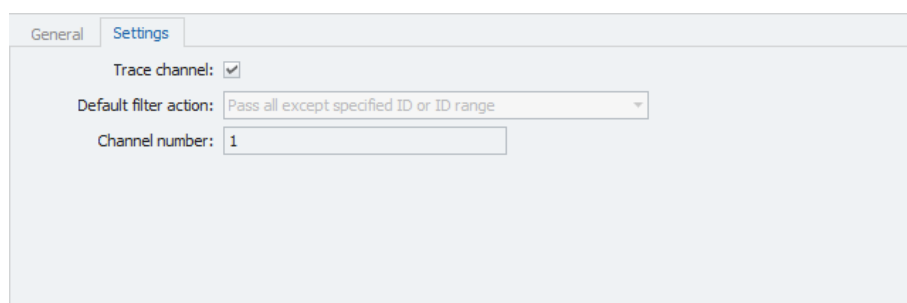
Define whether all trigger events should be stored in the trace data.

- **Identifier**

The field `Identifier` allows you to give a user identifier to the Bus trace. It does not have any effect other than helping the user identify a specific Bus trace.

### 14.8.3.3 Traceable Bus channel

#### Settings



General **Settings**

Trace channel:

Default filter action:

Channel number:

- **Trace channel**

Mark this box active in order to trace this channel.

- **Default filter action**

Define the default action for an ID Filter if it has been defined. For information on working with ID Filters for Bus trace, please refer to the chapter **Bus trace ID Filter** (→ [14.8.4](#)).

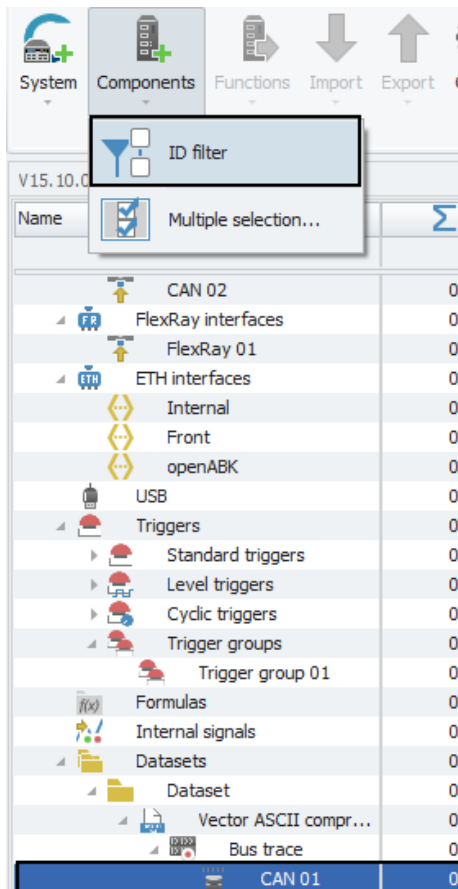
- **Channel number**

Define an alternative channel number that will be written in the datafile instead of the physical channel number.

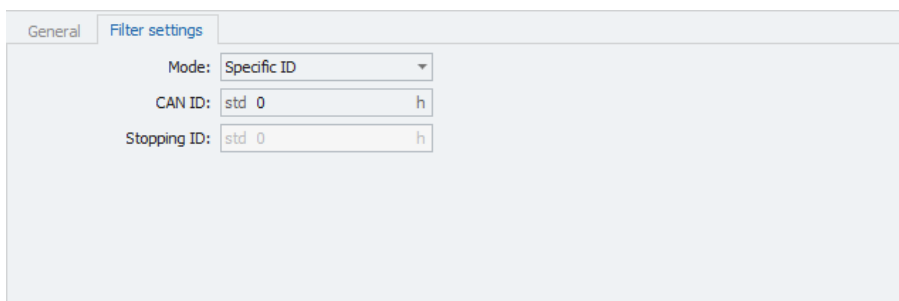
#### 14.8.4 Bus trace ID Filter

For “Bus trace” it is possible to specify one or more “ID Filters”. Such it is possible to specify a single ID or an ID range for a Bus and filter incoming traffic on that bus accordingly. It is possible to either block all traffic except the specified ID/ID range or to pass all traffic except the specified ID/ID range.

In order to specify an “ID Filter”, you will first need to add the “ID Filter” component to the desired Bus channel. To do so, select the desired Bus trace channel in the tree, click on the “Components” button in the Ribbon and then choose “ID Filter”. The new “ID Filter”



will appear in the grid area of the channel it belongs to. Select the filter and navigate to the “Filter settings” tab in the details area. Here you will be able to specify the settings for the filter.



- **Mode**

Specify whether a specific ID or a range of IDs should be used for the filter.

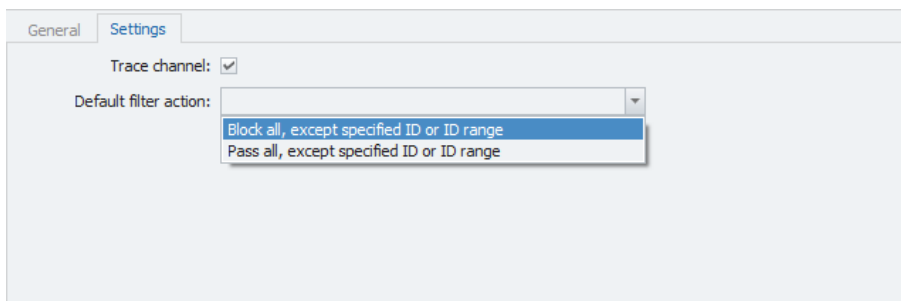
Changes and errors excepted.

- **CAN/LIN ID (For CAN and LIN channels only)**  
Allows you to define the specific/starting ID of your filter.
- **Frame ID (For FlexRay channels only)**  
Allows you to define the specific/starting ID of your filter.
- **Stopping ID**  
If using a range of IDs, this field allows you to define the stopping ID.
- **Cycle repetition (For FlexRay channels only)**  
Allows you to define cycle repetition of the frame.
- **Base cycle (For FlexRay channels only)**  
Allows you to define Base cycle of the frame.



All numbers in this tab can be entered in binary, decimal or hexadecimal format. Furthermore can the ID mode for the CAN ID switched from standard (0-7FF) to extended (0-1FFFFFFF).

Now the "ID Filter" has been specified and you will be able to define what the default action for filters on the traced Bus channel should be. To do so, select the respective Bus trace channel in the tree and navigate to the "Settings" tab in the details area. Here you will be able to choose between to default actions for filters on this channel.



- **Block all, except specified ID or ID range** will block out and ignore all incoming traffic on the channel except the specified ID/ID range.
- **Pass all, except specified ID or ID range** will store all incoming traffic on this channel except the spdecified ID/ID range, which will be blocked and ignored.



It is possible to add and specify multiple ID Filters for one Bus trace channel.

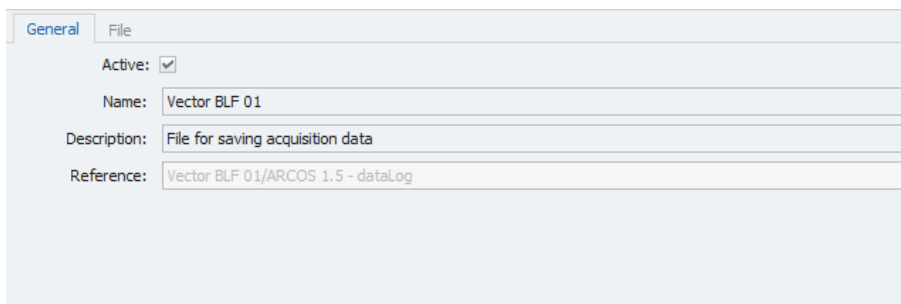
### 14.8.5 Details area for bus tracing (Ring buffer)

The details area contains settings for the behaviour of your bus tracing file (Vector BLF / Vector ASCII / Vector ASCII compressed), the “Bus trace” component or a traceable Bus channel.

#### 14.8.5.1 Bus tracing file

##### General

This tab provides general settings for the selected bus tracing file.



General	File
Active:	<input checked="" type="checkbox"/>
Name:	Vector BLF 01
Description:	File for saving acquisition data
Reference:	Vector BLF 01/ARCOS 1.5 - dataLog

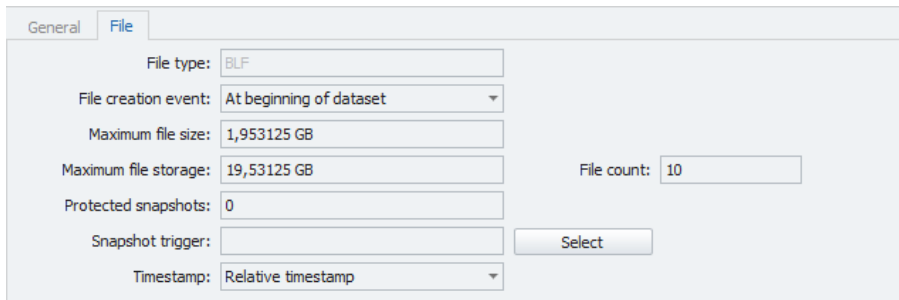
- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.



In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## File

This tab provides settings regarding the creation of the file.



- **File type**  
Tells you the type of the created file.
- **File create event**  
Define, when the bus tracing file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset is started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the Bus trace settings (→ <a href="#">14.8.3.2</a> ). This will result in a splitting of the current ATFX file into multiple files, as a new file is created for each time the trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the Bus trace settings (→ <a href="#">14.8.3.2</a> ). Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the Bus trace settings (→ <a href="#">14.8.3.2</a> ). Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party evaluation tools cannot handle files, that are larger.

- **Maximum file storage**

Define the maximum file storage space. If the maximum file storage space has been reached, older files will be deleted to make room for newer files.

- **File count**

Tells you how many files can be hold with the current combination of "Maximum file size" and "Maximum file storage". You may also define here, how many files you wish to be saved, and then the "Maximum file storage" will be filled in automatically.

- **Protected snapshots**

Define how many files before the trigger should be secured. The current and the following file will be secured automatically.

- **Snapshot trigger**

It is possible to mark certain datafile via a trigger, in order for these files to be secured. These files will not be deleted, when the "Maximum file storage" of the ring buffer is reached.

This field allows you to select the trigger event, that will mark a datafile.

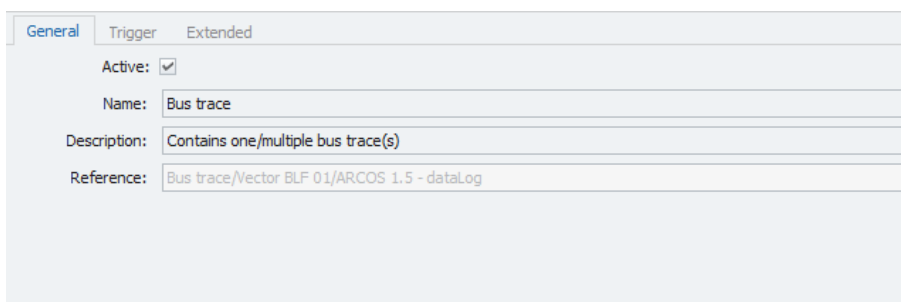
- **Timestamp**

This dropdown menu allows you to set the format of the timestamp for the file.

### 14.8.5.2 Bus trace

#### General

This tab provides general settings for the selected "Bus trace".



General	Trigger	Extended
Active:	<input checked="" type="checkbox"/>	
Name:	Bus trace	
Description:	Contains one/multiple bus trace(s)	
Reference:	Bus trace/Vector BLF 01/ARCOS 1.5 - dataLog	

- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.

- **Description**

Give a user-defined description to the selected file.

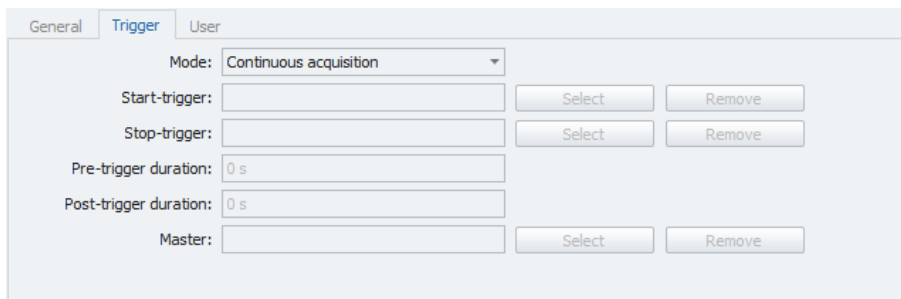
- **Reference**

This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## Trigger

This tab provides settings regarding the trigger for the start and stop of the "Bus trace". When the "Bus trace" is started, all traffic on the channel will be stored to the bus tracing file.

Furthermore will this trigger provoke the creation of the bus tracing file, if you have chosen either "On trigger" or "On first trigger" as "File create event".



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the "Bus trace".

- **Stop-trigger**

Define a trigger, that will stop the "Bus trace".

- **Pre-trigger duration**

Pre-trigger duration allows you to define, how long before the start trigger was set, the "Bus trace" will start.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, the "Bus trace" will stop.

- **Master**

Allows you to apply the trigger settings from another datafile of the current dataset.

### Extended

This tab allows you to choose whether you wish to store internal events in the trace data.



- **Include internal events**

Define whether the occurrence of internal events should be stored in the trace data.

- **Include all trigger events**

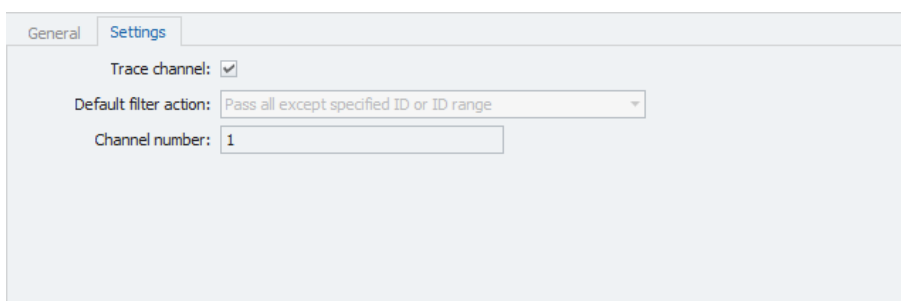
Define whether all trigger events should be stored in the trace data.

- **Identifier**

The field Identifier allows you to give a user identifier to the Bus trace. It does not have any effect other than helping the user identify a specific Bus trace.

#### 14.8.5.3 Traceable Bus channel

### Settings



- **Trace channel**

Mark this box active in order to trace this channel.

- **Default filter action**

Define the default action for an ID Filter if it has been defined. For information on working with ID Filters for Bus trace, please refer to the chapter **Bus trace ID Filter** (→ [14.8.4](#)).



- **Channel number**


Define an alternative channel number that will be written in the datafile instead of the physical channel number).

## 14.9 PCAP

PCAP is a filetype for message orientated rawdata recordings on ethernet channels. The trace method records all the messages that arrive on the ethernet channel. Regardless of the signals defined, all the messages are recorded. Filter rules can be defined to reduce the data volume. A typical trace application is the acquisition of all raw data in order to later evaluate the total traffic on the channel. Unlike most of the other methods, traces are event-oriented. This means the messages are not retrieved from the channels according to a set time pattern, but are recorded as soon as they arrive on the channel. This method accordingly has no parameter for sampling rate.



Including PCAP in your dataset will produce a “Warning” message, saying that “at least one channel must be set active”.

Symbol	Time	Type	Source	Message
	22.02.2018 11:41:50,708	WARNING	ETH trace	At least one channel must be set active

Please refer to the section “ETH trace” (→ [14.9.3.2](#)), in order to activate a channel for tracing.

### 14.9.1 Tree elements for PCAP

Including PCAP in your dataset will add various new child elements to your tree element “Dataset”:

- **PCAP**

This element represents the PCAP file, which will later be included in your exported dataset. You may add multiple files of the same filetype.

└─ Datasets	0
└─ Dataset	0
└─ PCAP	0
└─ ETH trace	0
└─ Internal	0
└─ Front	0
└─ openABK	0

- **ETH trace**

This element represents the “ETH trace” for recording all the traffic on a selected ethernet channel.

└─ Datasets	0
└─ Dataset	0
└─ PCAP	0
└─ ETH trace	0
└─ Internal	0
└─ Front	0
└─ openABK	0

- **ETH channels available for tracing**

As child elements to the tree element "ETH trace" will appear all the ETH channels which are currently available for tracing.

└─ Datasets	0
└─ Dataset	0
└─ PCAP	0
└─ ETH trace	0
└─ Internal	0
└─ Front	0
└─ openABK	0

### 14.9.2 Grid area for PCAP

If the "ETH trace" is selected in the Measurement task tree, the grid area will show an overview of the ethernet channels available for tracing.

At least one channel must be marked active for tracing, by ticking the checkbox labeled "Trace channel".

Name	Active	Description	Trace channel
Internal	<input checked="" type="checkbox"/>	ETH trace channel	<input type="checkbox"/>
Front	<input checked="" type="checkbox"/>	ETH trace channel	<input checked="" type="checkbox"/>
openABK	<input checked="" type="checkbox"/>	ETH trace channel	<input type="checkbox"/>

### 14.9.3 Details area for PCAP

The details area contains settings for the behaviour of your PCAP file or "ETH trace".

#### 14.9.3.1 PCAP file

##### General

This tab provides general settings for the selected Eth tracing file.

General	File
Active:	<input checked="" type="checkbox"/>
Name:	PCAP
Description:	File for saving ETH trace data
Reference:	PCAP/ARCOS 1.5 - dataLog

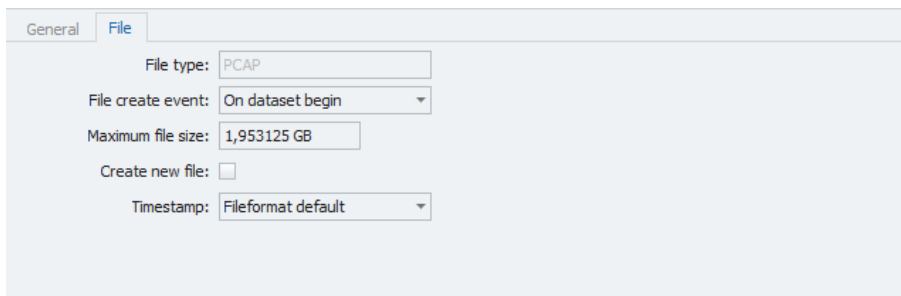
- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.



In the field "Name" project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## File

This tab provides settings regarding the creation of the file.



The screenshot shows a software interface with two tabs: 'General' and 'File'. The 'File' tab is selected. It contains the following settings:

- File type: PCAP
- File create event: On dataset begin
- Maximum file size: 1,953125 GB
- Create new file:
- Timestamp: Fileformat default

- **File type**  
Tells you the type of the created file.
- **File create event**  
Define, when the PCAP file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset ist started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.9.3.2</a> ). This will result in a splitting of the current ATFX file into multiple files, as a new file is created for each timethe trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.9.3.2</a> ). Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.9.3.2</a> ). Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party analysis tools cannot handle files, that are larger.

- **Create new file**

If this box is marked active, a new file will be created, if the current file exceeds the maximum file size.

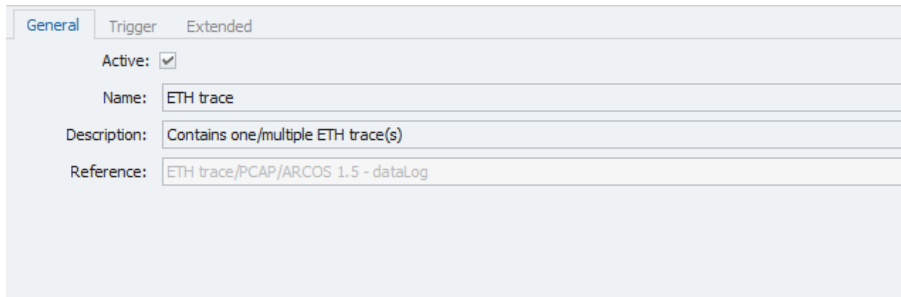
- **Timestamp**

This dropdown menu allows you to set the format of the timestamp for the file.

### 14.9.3.2 Eth trace

#### General

This tab provides general settings for the selected “ETH trace”.



The screenshot shows the 'General' tab of a configuration window. It contains the following fields:

- Active:** A checked checkbox.
- Name:** A text box containing 'ETH trace'.
- Description:** A text box containing 'Contains one/multiple ETH trace(s)'.
- Reference:** A text box containing 'ETH trace/PCAP/ARCOS 1.5 - dataLog'.

- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.

- **Description**

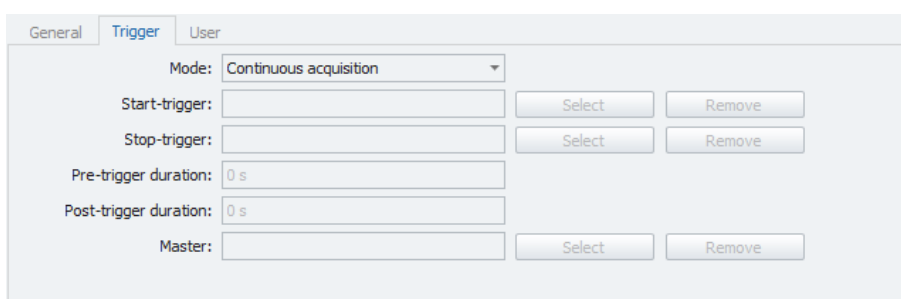
Give a user-defined description to the selected file.

- **Reference**

This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

#### Trigger

This tab provides settings regarding the trigger for the start and stop of the “ETH trace”. When the “ETH trace” is started, all traffic on the channel will be stored to the PCAP file. Furthermore will this trigger provoke the creation of the PCAP file, if you have chosen either “On trigger” or “On first trigger” as “File create event”.



The screenshot shows the 'Trigger' tab of a configuration window. It contains the following fields and controls:

- Mode:** A dropdown menu set to 'Continuous acquisition'.
- Start-trigger:** A text box with 'Select' and 'Remove' buttons.
- Stop-trigger:** A text box with 'Select' and 'Remove' buttons.
- Pre-trigger duration:** A text box containing '0 s'.
- Post-trigger duration:** A text box containing '0 s'.
- Master:** A text box with 'Select' and 'Remove' buttons.

- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the "ETH trace".

- **Stop-trigger**

Define a trigger, that will stop the "ETH trace".

- **Pre-trigger duration**

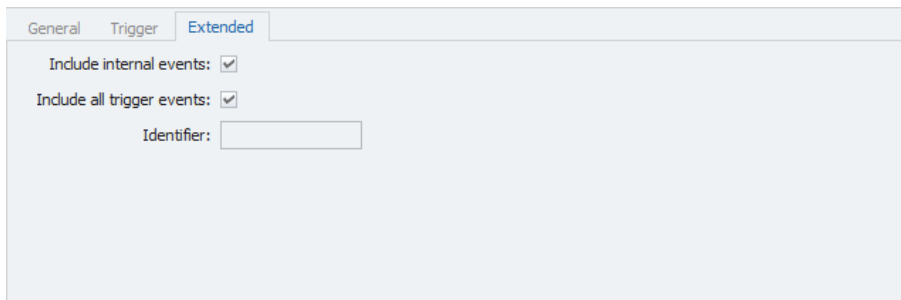
Pre-trigger duration allows you to define, how long before the start trigger was set, the "ETH trace" will start.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, the "ETH trace" will stop.

## Extended

This tab provides extended settings for “ETH trace”.



General Trigger **Extended**

Include internal events:

Include all trigger events:

Identifier:

- **Include internal events**

Define whether the occurrence of internal events should be stored in the trace data.

- **Include all trigger events**

Define whether all trigger events should be stored in the trace data.

- **Identifier**

The field `Identifier` allows you to give a user identifier to the ETH trace. It does not have any effect other than helping the user identify a specific ETH trace.



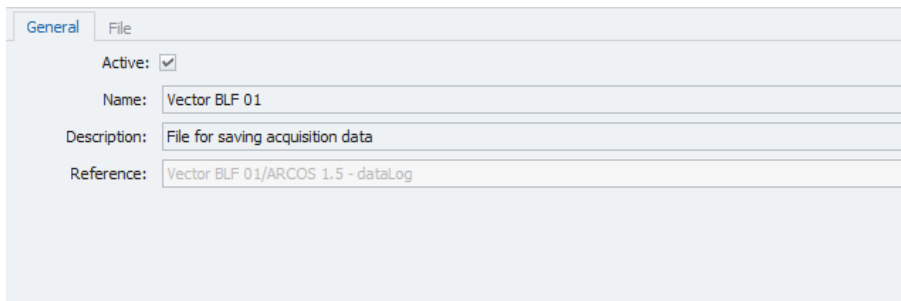
### 14.9.4 Details area for PCAP (Ring buffer)

The details area contains settings for the behaviour of your PCAP file, the “ETH trace” component or a traceable Bus channel.

#### 14.9.4.1 PCAP file

##### General

This tab provides general settings for the selected PCAP file.



The screenshot shows a configuration window with two tabs: 'General' and 'File'. The 'General' tab is active. It contains the following fields:

- Active:** A checked checkbox.
- Name:** A text field containing 'Vector BLF 01'.
- Description:** A text field containing 'File for saving acquisition data'.
- Reference:** A text field containing 'Vector BLF 01/ARCOS 1.5 - dataLog'.

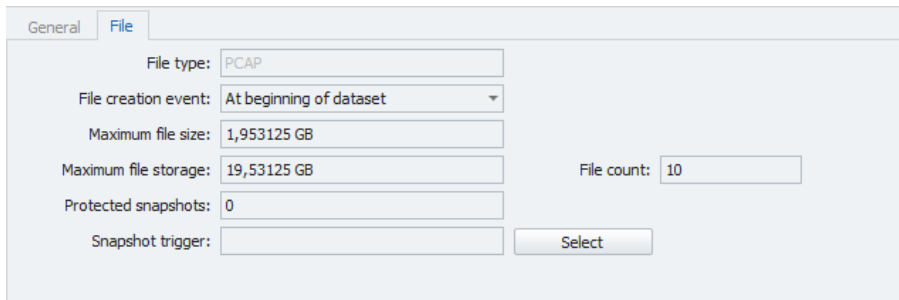
- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.



In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## File

This tab provides settings regarding the creation of the file.



- **File type**  
Tells you the type of the created file.
- **File create event**  
Define, when the PCAP file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset is started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the ETH trace settings (→ <a href="#">14.9.4.2</a> ). This will result in a splitting of the current ATFX file into multiple files, as a new file is created for each time the trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the ETH trace settings (→ <a href="#">14.9.4.2</a> ). Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the ETH trace settings (→ <a href="#">14.9.4.2</a> ). Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party evaluation tools cannot handle files, that are larger.

- **Maximum file storage**

Define the maximum file storage space. If the maximum file storage space has been reached, older files will be deleted to make room for newer files.

- **File count**

Tells you how many files can be hold with the current combination of "Maximum file size" and "Maximum file storage". You may also define here, how many files you wish to be saved, and then the "Maximum file storage" will be filled in automatically.

- **Protected snapshots**

Define how many files before the trigger should be secured. The current and the following file will be secured automatically.

- **Snapshot trigger**

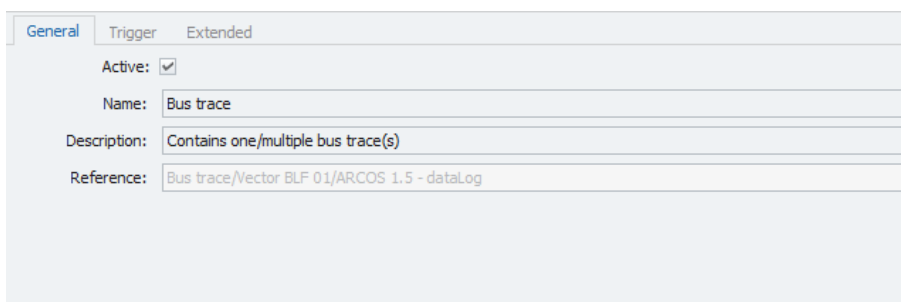
It is possible to mark certain datafile via a trigger, in order for these files to be secured. These files will not be deleted, when the "Maximum file storage" of the ring buffer is reached.

This field allows you to select the trigger event, that will mark a datafile.

#### 14.9.4.2 ETH trace

##### General

This tab provides general settings for the selected "ETH trace".



General	Trigger	Extended
Active:	<input checked="" type="checkbox"/>	
Name:	Bus trace	
Description:	Contains one/multiple bus trace(s)	
Reference:	Bus trace/Vector BLF 01/ARCOS 1.5 - dataLog	

- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.

- **Description**

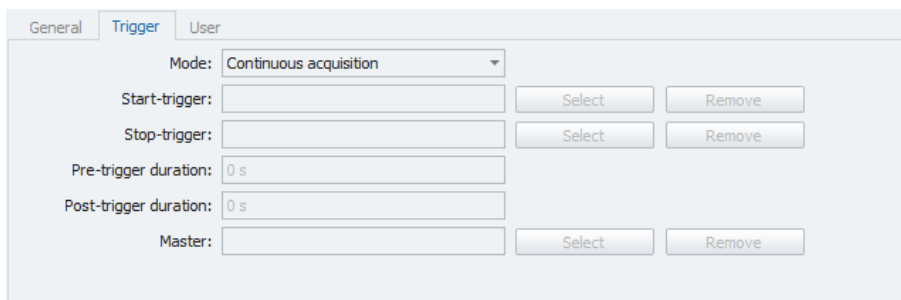
Give a user-defined description to the selected file.

- **Reference**

This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## Trigger

This tab provides settings regarding the trigger for the start and stop of the "ETH trace". When the "ETH trace" is started, all traffic on the channel will be stored to the PCAP file. Furthermore will this trigger provoke the creation of the PCAP file, if you have chosen either "On trigger" or "On first trigger" as "File create event".



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the "ETH trace".

- **Stop-trigger**

Define a trigger, that will stop the "ETH trace".

- **Pre-trigger duration**

Pre-trigger duration allows you to define, how long before the start trigger was set, the "ETH trace" will start.

- **Post-trigger duration**

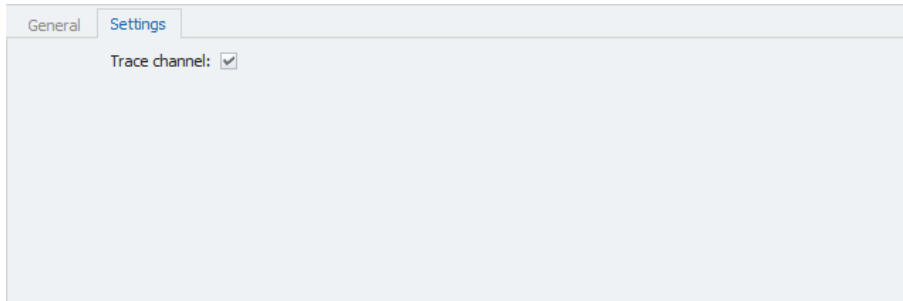
Post-trigger duration allows you to define, how long after the start trigger was set, the "ETH trace" will stop.

- **Master**

Allows you to apply the trigger settings from another datafile of the current dataset.

### 14.9.4.3 Traceable ETH channel

#### Settings



The screenshot shows a software interface with two tabs: 'General' and 'Settings'. The 'Settings' tab is active. Below the tabs, there is a checkbox labeled 'Trace channel:' which is checked.

#### Trace channel

Mark this box active in order to trace this channel.

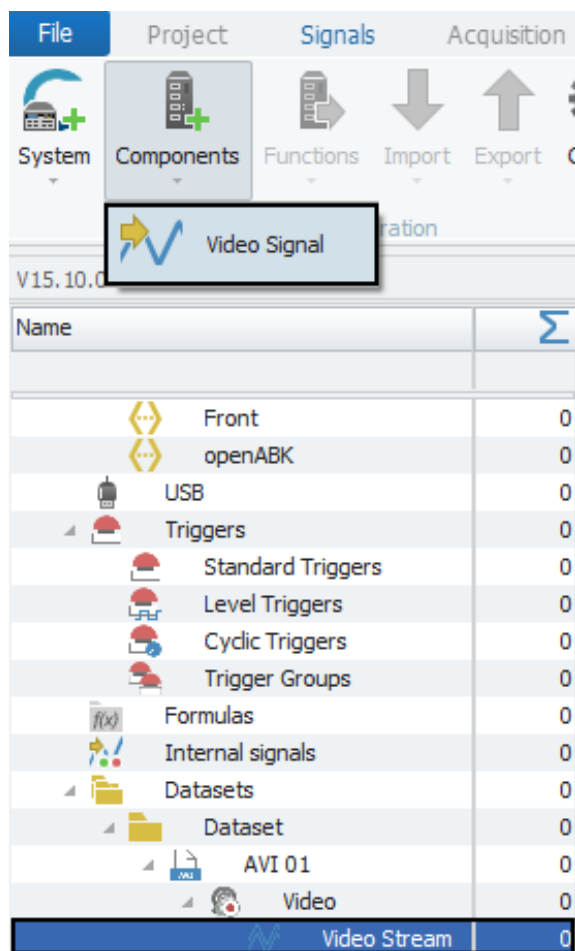
## 14.10 AVI

The "AVI" filetype is meant for recording of video streams. For each AVI file, that you add to your dataset you can only record one video stream.

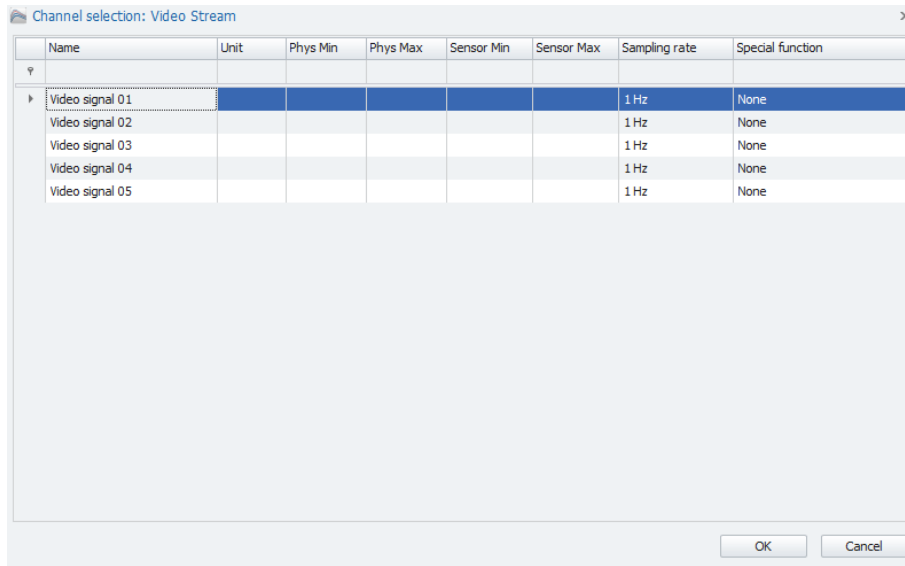
### 14.10.1 Including a video signal in the Video Stream

In order to store a "Video Stream", you will need to include a video signal in your "Video Stream".

To do so, select the tree element "Video Stream", click the "Components" button in the Ribbon and choose "Video Signal".



In the resulting window you will be presented with an overview of all the available video signals. You can choose one signal and confirm by clicking “OK”.

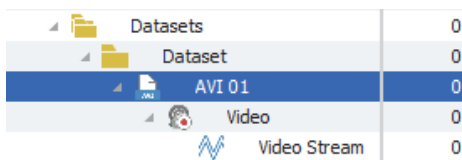


### 14.10.2 Tree elements for AVI

Including the “AVI” filetype in your dataset will add three new child elements to your tree element “Dataset”:

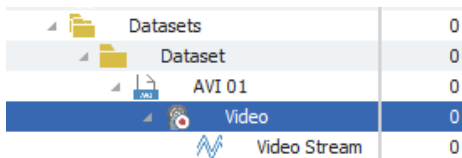
- **AVI xx**

This element represents the AVI file, which will later be included in your exported dataset. You may add multiple files of the same filetype.



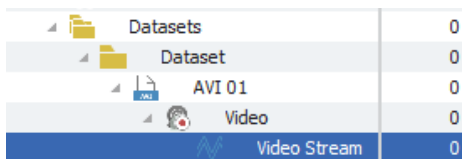
- **Video**

Represents the video element in your AVI file. It tells you, that the type of signal, which can be stored, is a video signal.



- **Video Stream**

This element represents the video signal which will be stored.



### 14.10.3 Grid area for AVI

If the "Video Stream" is selected in the Measurement task tree, and a video signal has already been included in the "Video Stream", the grid area will show the video signal which has been included in the "Video Stream".

Channel	Index	Active	Unit	Sampling rate
Video signal 01	1	<input checked="" type="checkbox"/>		1 Hz

### 14.10.4 Details area for AVI

The details area contains settings for the behaviour of your AVI file, Video element or Video Stream.

#### 14.10.4.1 AVI File

##### General

This tab provides general settings for the selected AVI file.

General	File
Active:	<input checked="" type="checkbox"/>
Name:	AVI 01
Description:	File for saving video data
Reference:	AVI 01/ARCOS 1.5 - dataLog

- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.



- **Reference**

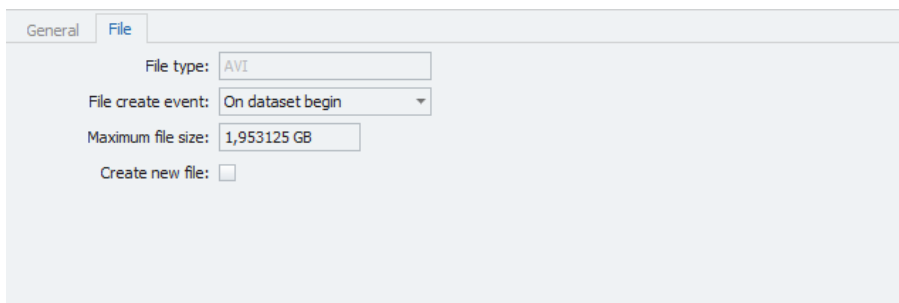
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.



In the field "Name" project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## File

This tab provides settings regarding the creation of the file.



General **File**

File type: AVI

File create event: On dataset begin

Maximum file size: 1,953125 GB

Create new file:

- **File type**

Tells you the type of the created file.

- **File create event**

Define, when the AVI file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset is started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.10.4.2</a> ). This will result in a splitting of the current ATFX file into multiple files, as a new file is created for each time the trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.10.4.2</a> ). Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the timelog settings (→ <a href="#">14.10.4.2</a> ). Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party evaluation tools cannot handle files, that are larger.

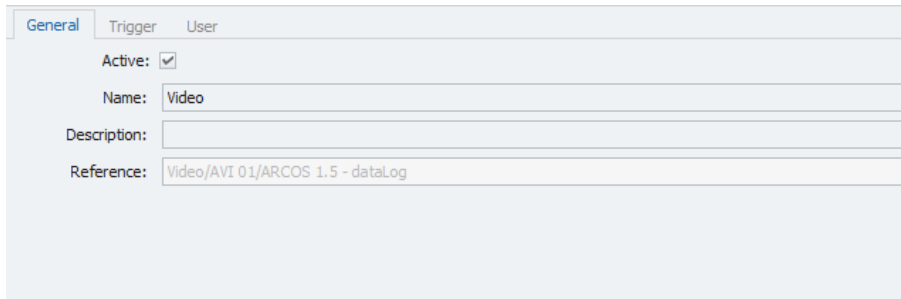
- **Create new file**

If this box is marked active, a new file will be created, if the current file exceeds the maximum file size.

## 14.10.4.2 Video

### General

This tab provides general settings for the selected Video element.



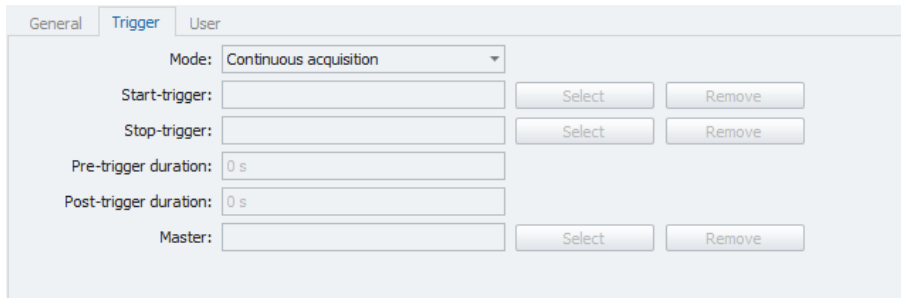
The screenshot shows a configuration dialog with three tabs: 'General', 'Trigger', and 'User'. The 'General' tab is active. It contains the following fields:

- Active:** A checked checkbox.
- Name:** A text field containing the value 'Video'.
- Description:** An empty text field.
- Reference:** A text field containing the value 'Video/AVI 01/ARCOS 1.5 - dataLog'.

- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## Trigger

This tab provides settings regarding the trigger for the start and stop of the Video element. When the Video element is started, the "Video Stream" will be stored to the AVI file. Furthermore will this trigger provoke the creation of the AVI file, if you have chosen either "On trigger" or "On first trigger" as "File create event".



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the Video element.

- **Stop-trigger**

Define a trigger, that will stop the Video element.

- **Pre-trigger duration**

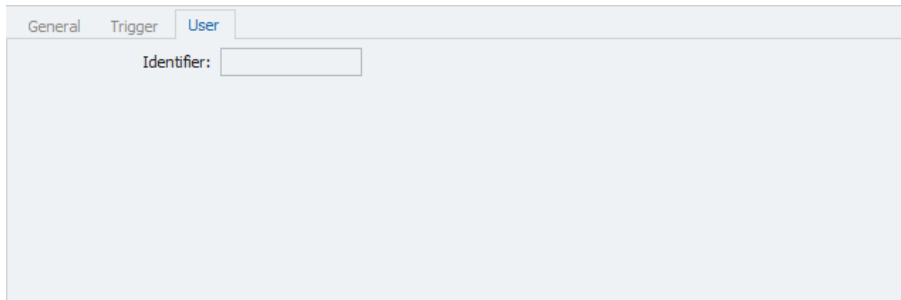
Pre-trigger duration allows you to define, how long before the start trigger was set, the Video element will start.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, the Video element will stop.

## User

The field "Identifier" allows you to give a user identifier to the timelog. It does not have any effect other than helping the user identify a specific timelog.



### 14.10.4.3 Video Stream

#### General

This tab provides general settings for the selected "Video Stream".



- **Active**  
Allows you to activate or deactivate the "Video Stream".
- **Name**  
Give a user-defined Name to the selected "Video Stream".
- **Description**  
Give a user-defined description to the selected "Video Stream".
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

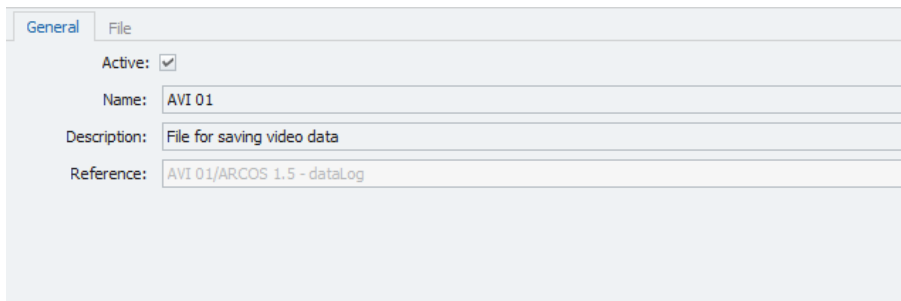
## 14.10.5 Details area for AVI (Ring buffer)

The details area contains settings for the behaviour of your AVI file, Video element or Video Stream.

### 14.10.5.1 AVI File

#### General

This tab provides general settings for the selected AVI file.



The screenshot shows a dialog box with two tabs: 'General' (selected) and 'File'. The 'General' tab contains the following fields:

- Active:
- Name: AVI 01
- Description: File for saving video data
- Reference: AVI 01/ARCOS 1.5 - dataLog

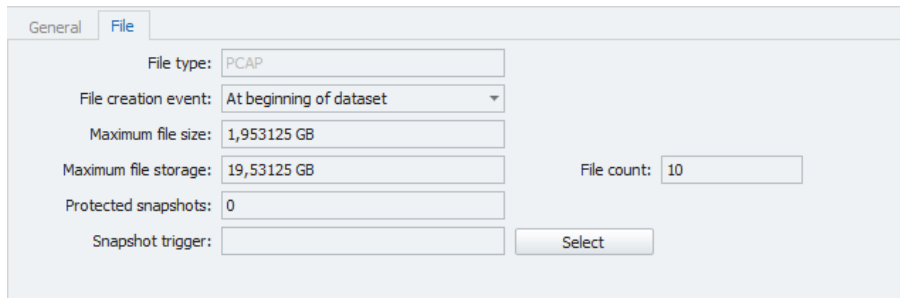
- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.



In the field "Name" project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## File

This tab provides settings regarding the creation of the file.



- **File type**  
Tells you the type of the created file.
- **File create event**  
Define, when the AVI file should be created. There are four possibilities:

File create event	Characteristics
On dataset begin	The file will be created once at logger start.
On recording start	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset is started.
On trigger	The file will be created on a trigger and record for a user defined duration. These settings can be defined in the Video settings (→ <a href="#">14.10.5.2</a> ). This will result in a splitting of the current ATFX file into multiple files, as a new file is created for each time the trigger is set.
On first trigger (per dataset)	The file will be created once, when the defined trigger is set for the first time since the beginning of the dataset and record for a user defined duration. These settings can be defined in the Video settings (→ <a href="#">14.10.5.2</a> ). Each following time the trigger is set, the data will be written in the same previously created file. Therefore there will only be one file.
On first trigger per recording	The file will be created once, when the defined trigger is set for the first time during a recording and save data for a user defined duration. These settings can be defined in the Video settings (→ <a href="#">14.10.5.2</a> ). Each following time the trigger is set during the same period of recording, the data will be written in the same previously created file. Therefore there will only be one file per recording.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party evaluation tools cannot handle files, that are larger.

- **Maximum file storage**

Define the maximum file storage space. If the maximum file storage space has been reached, older files will be deleted to make room for newer files.

- **File count**

Tells you how many files can be hold with the current combination of "Maximum file size" and "Maximum file storage". You may also define here, how many files you wish to be saved, and then the "Maximum file storage" will be filled in automatically.

- **Protected snapshots**

Define how many files before the trigger should be secured. The current and the following file will be secured automatically.

- **Snapshot trigger**

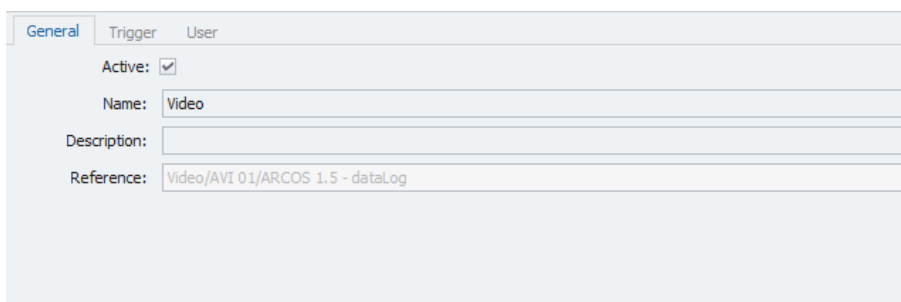
It is possible to mark certain datafile via a trigger, in order for these files to be secured. These files will not be deleted, when the "Maximum file storage" of the ring buffer is reached.

This field allows you to select the trigger event, that will mark a datafile.

## 14.10.5.2 Video

### General

This tab provides general settings for the selected Video element.



General	Trigger	User
Active: <input checked="" type="checkbox"/>		
Name: Video		
Description:		
Reference: Video/AVI 01/ARCOS 1.5 - dataLog		

- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.



- **Description**

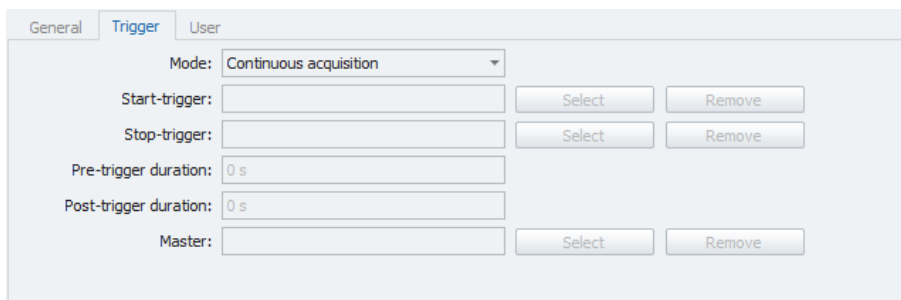
Give a user-defined description to the selected file.

- **Reference**

This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## Trigger

This tab provides settings regarding the trigger for the start and stop of the Video element. When the Video element is started, the "Video Stream" will be stored to the AVI file. Furthermore will this trigger provoke the creation of the AVI file, if you have chosen either "On trigger" or "On first trigger" as "File create event".



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start the Video element.

- **Stop-trigger**

Define a trigger, that will stop the Video element.

- **Pre-trigger duration**

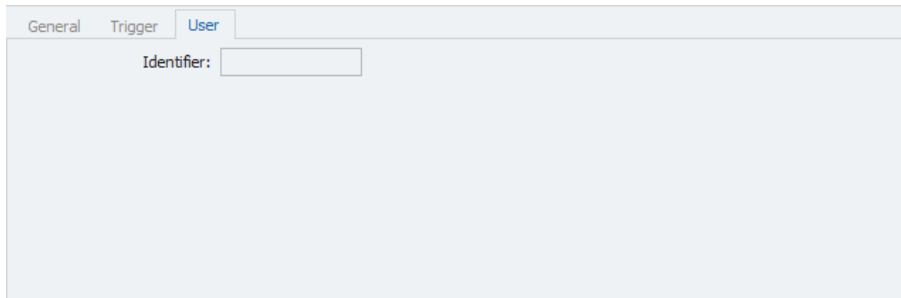
Pre-trigger duration allows you to define, how long before the start trigger was set, the Video element will start.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, the Video element will stop.

## User

The field "Identifier" allows you to give a user identifier to the timelog. It does not have any effect other than helping the user identify a specific timelog.



### 14.10.5.3 Video Stream

#### General

This tab provides general settings for the selected "Video Stream".



- **Active**  
Allows you to activate or deactivate the "Video Stream".
- **Name**  
Give a user-defined Name to the selected "Video Stream".
- **Description**  
Give a user-defined description to the selected "Video Stream".
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

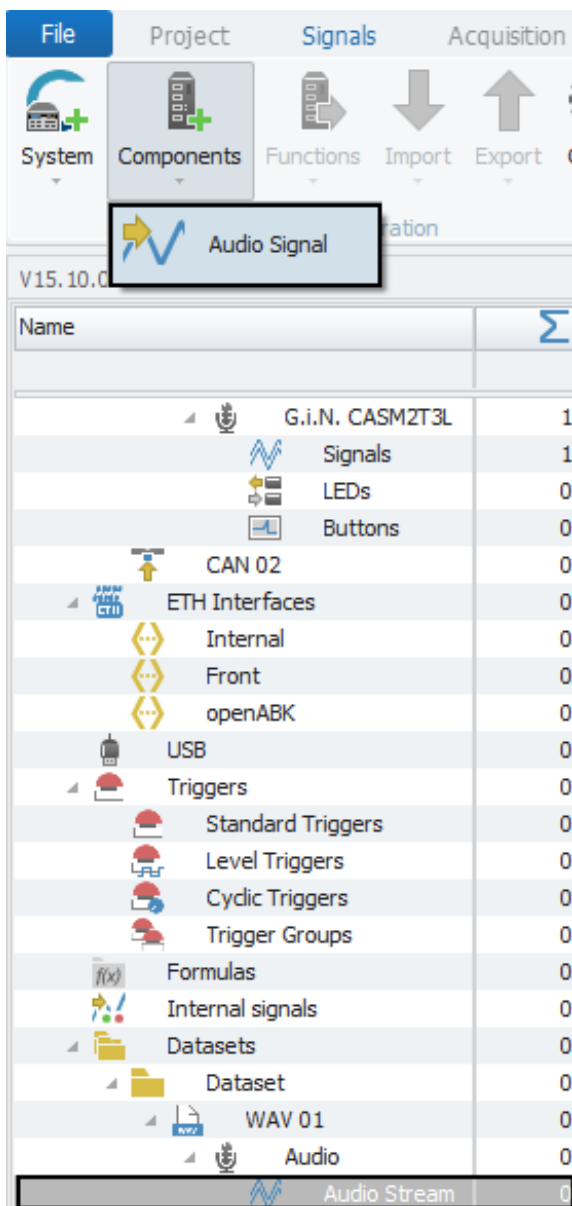
## 14.11 WAV

The “WAV” filetype is meant for recording of audio streams. For each WAV file, that you add to your dataset you can only record one audio stream.

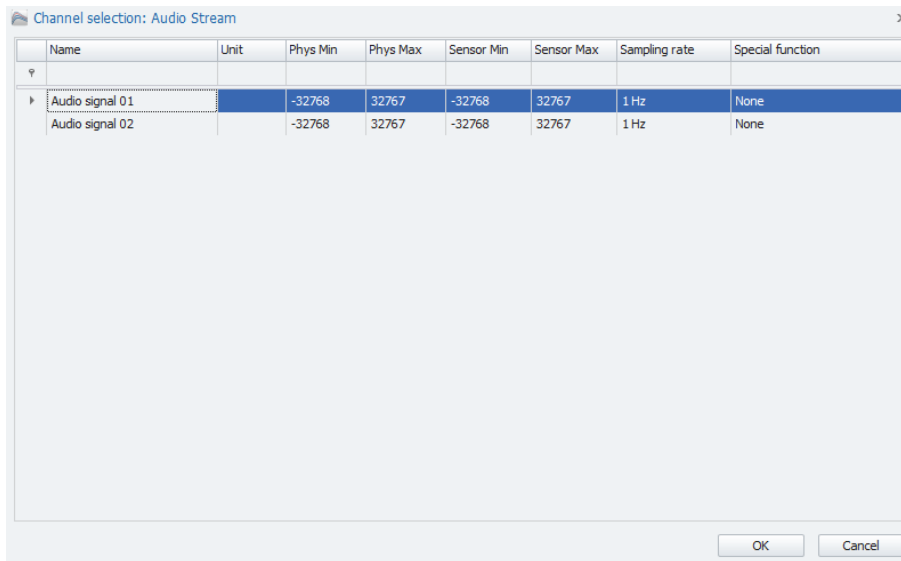
### 14.11.1 Including an audio signal in the audio Stream

In order to store an “Audio Stream”, you will need to include a audio signal in your “Audio Stream”.

To do so, select the tree element “Audio Stream”, click the “Components” button in the Ribbon and choose “Audio Signal”.



In the resulting window you will be presented with an overview of all the available audio signals. You can choose one signal and confirm by clicking “OK”.



### 14.11.2 Tree elements for WAV

Including the “WAV” filetype in your dataset will add three new child elements to your tree element “Dataset”:

- **WAV xx**

This element represents the WAV file, which will later be included in your exported dataset. You may add multiple files of the same filetype.

Datasets	0
Dataset	0
WAV 01	0
Audio	0
Audio Stream	1

- **Audio**

Represents the audio element in your WAV file. It tells you, that the type of signal, which can be stored, is an audio signal.

Datasets	0
Dataset	0
WAV 01	0
Audio	0
Audio Stream	1

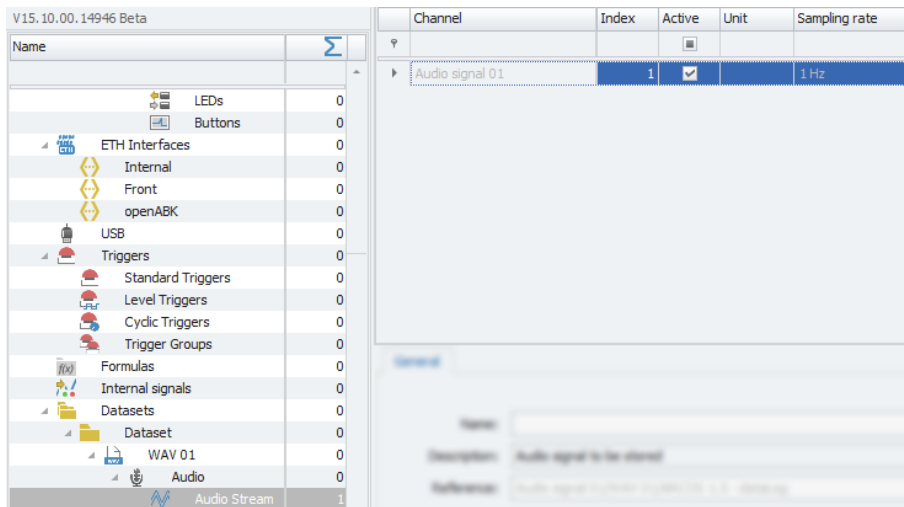
- **Audio Stream**

This element represents the audio signal which will be stored.

Datasets	0
Dataset	0
WAV 01	0
Audio	0
Audio Stream	1

### 14.11.3 Grid area for WAV

If the "audio Stream" is selected in the Measurement task tree, and a audio signal has already been included in the "audio Stream", the grid area will show the audio signal which has been included in the "audio Stream".

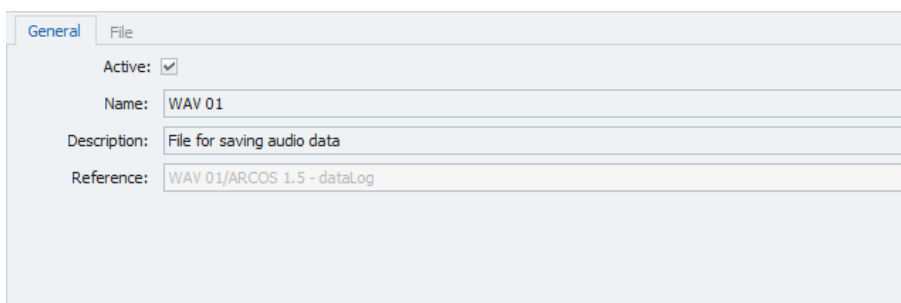


### 14.11.4 Details area for WAV

The details area contains settings for the behaviour of your WAV file.

#### General

This tab provides general settings for the selected WAV file.



- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.

- **Reference**

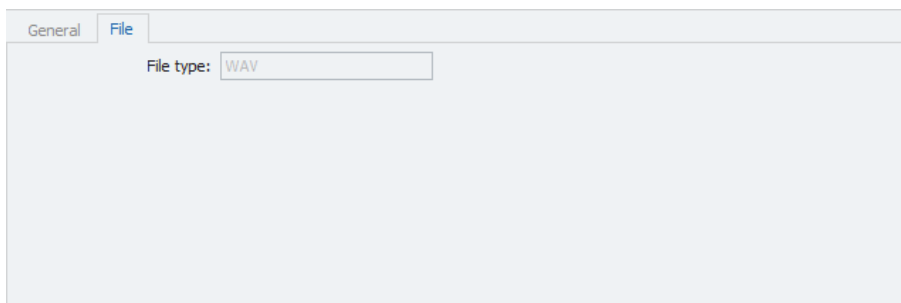
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.



In the field "Name" project parameters can be used as variables. For more information please refer to (→[5.6](#)).

## File

This tab tells you the filetype of your file.



## 14.12 GPX

The “GPX” filetype is meant for GPS Tracking.

### 14.12.1 Assigning GPS signals

Other than for other filetypes, when working with GPX, you will not be able to choose the signals you wish to store from a list of available signals, but you will need to assign a GPS task (latitude, longitude or altitude) to a signal. That means, that technically any signal can be used as GPS signal.

For instructions on assignig GPS tasks please refer to (→ [8.14.2.1](#)).



One GPS task may only be assigned to one signal. Use the “**Check**” button in the Ribbon to verify, that GPS tasks are uniquely assigned. If a GPS task is multiply assigned, navigate to the “Format” tab of the wrongly assigned signal and deassign the GPS task.

Exporting a configuration will also check your system for validity and inform you, if you have multiply assigned GPS tasks.

### 14.12.2 Tree elements for GPX

Including the “GPX” filetype in your dataset will add two new child elements to your tree element “Dataset”:

- **GPX**

This element represents the GPX file, which will later be included in your exported dataset.

└ Datasets	0
└ Dataset	0
└ GPX	0
└ GPS Tracking	1

- **GPS Tracking**

Represents the GPS signals you are tracking

└ Datasets	0
└ Dataset	0
└ GPX	0
└ GPS Tracking	1

### 14.12.3 Grid area for GPX

The grid area is not used for configuration of GPX. Instead signal selection in this case works via assignation of GPS tasks. Please refer to (→ [8.14.2.1](#)).

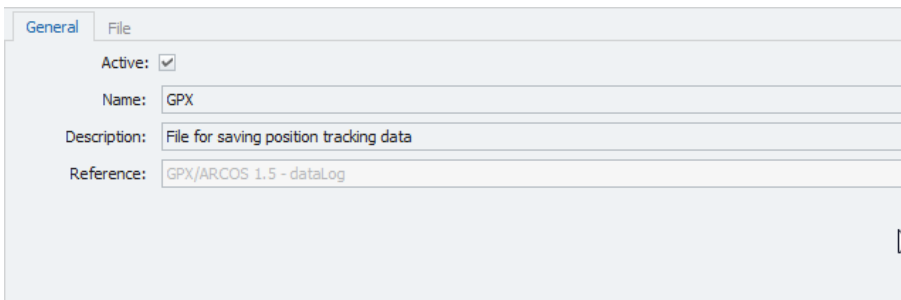
### 14.12.4 Details area for GPX

The details area contains settings for the behaviour of your GPX file and “GPS Tracking”.

#### 14.12.4.1 GPX File

##### General

This tab provides general settings for the selected GPX file.



- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected file.

- **Description**

Give a user-defined description to the selected file.

- **Reference**

This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

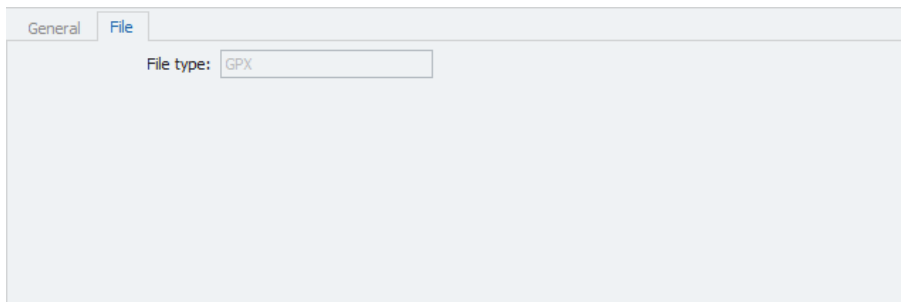


In the field “Name” project parameters can be used as variables. For more information please refer to (→[5.6](#)).



## File

This tab tells you the filetype of your file.

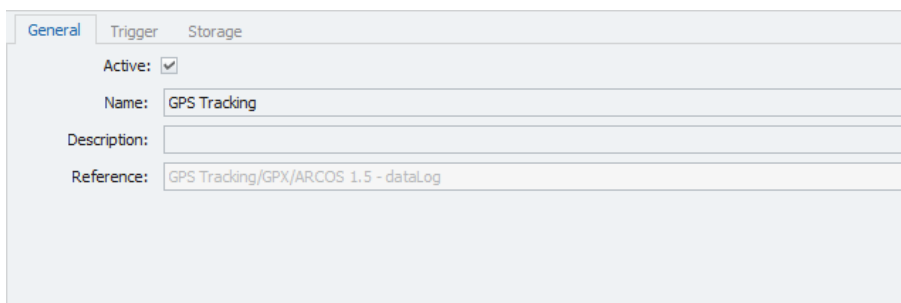


The screenshot shows a software interface with two tabs: 'General' and 'File'. The 'File' tab is active. Below the tabs, there is a label 'File type:' followed by a text input field containing the value 'GPX'.

### 14.12.4.2 GPS Tracking

#### General

This tab provides general settings for GPS Tracking.



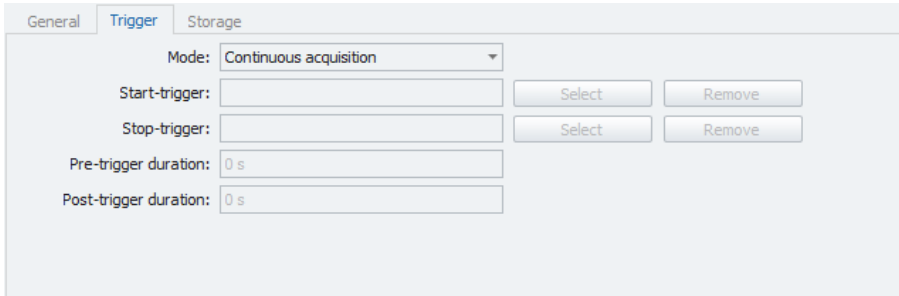
The screenshot shows a software interface with three tabs: 'General', 'Trigger', and 'Storage'. The 'General' tab is active. Below the tabs, there are several settings:

- Active:** A checkbox that is checked.
- Name:** A text input field containing 'GPS Tracking'.
- Description:** An empty text input field.
- Reference:** A text input field containing 'GPS Tracking/GPX/ARCOS 1.5 - dataLog'.

- **Active**  
Allows you to activate or deactivate "GPS Tracking".
- **Name**  
Give a user-defined Name to your "GPS Tracking".
- **Description**  
Give a user-defined description to your "GPS Tracking".
- **Reference**  
This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

## Trigger

This tab provides settings regarding the trigger for the start and stop of the “GPS Tracking”. When the “GPS Tracking” is started, all activated signals with GPS task will be stored to the GPX file.



- **Mode**

Define whether you wish to continuously store data or if you want to start/stop data storage via a trigger. There are two modes to control data storage via trigger:

**Start and stop trigger** allows you to set any previously defined trigger as start and/or stop condition.

**Stop is inverted start** will store data as long as the start trigger condition is met. Once it is no longer met and a possibly set **Post-trigger duration** has run out, data storage will stop.

- **Start-trigger**

Define a trigger, that will start “GPS Tracking”.

- **Stop-trigger**

Define a trigger, that will stop “GPS Tracking”.

- **Pre-trigger duration**

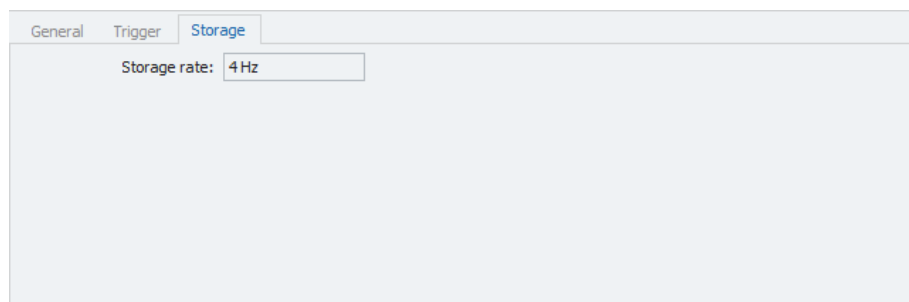
Pre-trigger duration allows you to define, how long before the start trigger was set, “GPS Tracking” will start.

- **Post-trigger duration**

Post-trigger duration allows you to define, how long after the start trigger was set, “GPS Tracking” will stop.

## Storage

This tab allows you to set the storage rate for “GPS Tracking”



Storage rates with decimal places will be rounded to three decimal places.

## 14.13 CAETEC binary (Classings)

The “CAETEC binary” filetype allows for classing of signals. Classings are processes for counting values or sequences (such as cycles) of signals. The standard procedure is to take the range in which the counted events are expected and divide this into so-called classes. Any values lying above the highest or below the lowest class are generally ignored (no open-border classes). Please refer to the chapter “**Classing methods**” (→ [14.15](#)).

The methods fundamentally differ in their counting strategies (for example, when levels are crossed, when a class is reached, cycle amplitudes and so on), and they are standardized. Which particular standard was applied in the design of each method is explained in the chapters on the different classing methods.

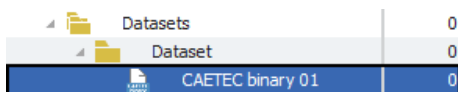
A signal can be used in several classing processes simultaneously.

Some methods can be applied with different numbers of signals. This is what is referred to as the dimension of the particular classing. 1D, 2D, 3D specify how many signals are joined in a class (joint classing). When joined, the classes of the signals form a matrix. Counting is always performed in the matrix element in which the counting conditions for all the involved signals are met by the same sampling instance.

### 14.13.1 Tree elements for CAETEC binary

Including a “CAETEC binary” file in your dataset will add one new child element called “CAETEC binary xx” to your tree element “Dataset”.

Multiple “CAETEC binary” files can be added to your dataset. They will be labeled equally and the counter will be raised by one for each new file. This may be necessary, if multiple classings should be included in the dataset, as each “CAETEC binary” file can only contain one classing.



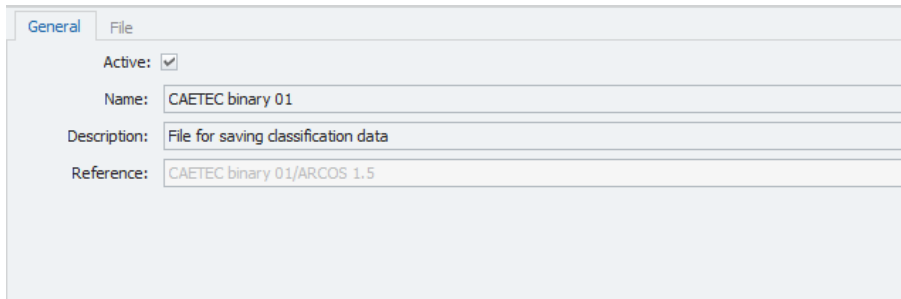
Datasets	0
Dataset	0
CAETEC binary 01	0

### 14.13.2 Details area for CAETEC binary

The details area contains settings regarding the “CAETEC binary” file.

#### General

This tab provides general settings for the selected CAETEC binary file.

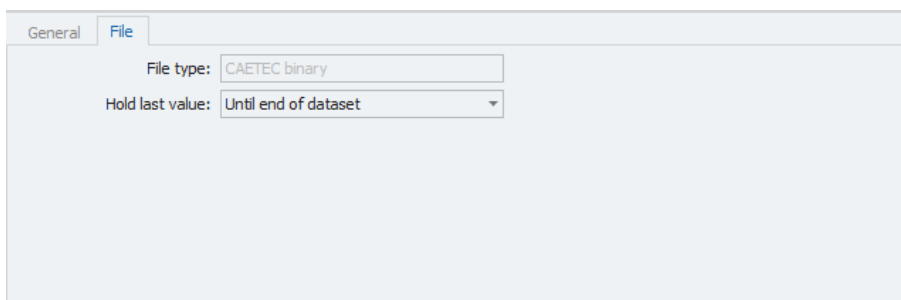


Active:	<input checked="" type="checkbox"/>
Name:	CAETEC binary 01
Description:	File for saving classification data
Reference:	CAETEC binary 01/ARCOS 1.5

- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected formula/signal.
- **Description**  
Give a user-defined description to the selected formula/signal.
- **Reference**  
This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

#### File

This tab provides settings regarding the creation of the file.

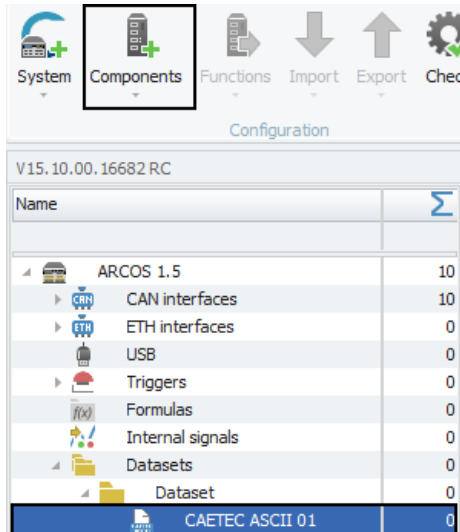


File type:	CAETEC binary
Hold last value:	Until end of dataset

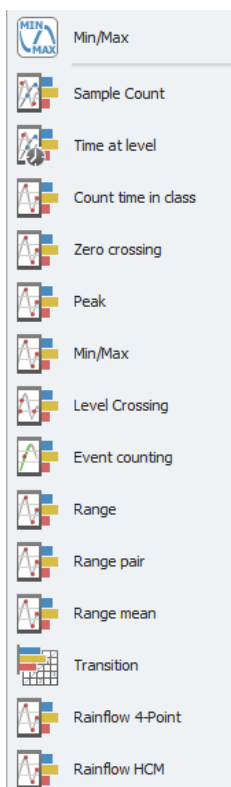
- **File type**  
Tells you the type of the created file.
- **Hold last value**  
Specify, for how long the last value of the signal will be hold.

### 14.13.3 Adding a classing

In order to add a classing, select the “CAETEC binary” or “CAETEC ASCII” file, which should contain the classing, in the measurement task tree.



Then click the “Components” button in the Ribbon and choose the desired classing method.



Once you have added a classing, you may now proceed to configure this classing. Please refer to the chapter “**Classing methods**” (→ [14.15](#)) for instructions.

## 14.14 CAETEC ASCII (Classings)

The “CAETEC ASCII” filetype allows for classing of signals. Classings are processes for counting values or sequences (such as cycles) of signals. The standard procedure is to take the range in which the counted events are expected and divide this into so-called classes. Any values lying above the highest or below the lowest class are generally ignored (no open-border classes).

The methods fundamentally differ in their counting strategies (for example, when levels are crossed, when a class is reached, cycle amplitudes and so on), and they are standardized. Which particular standard was applied in the design of each method is explained in the chapters on the different classing methods. Please refer to the chapter “**Classing methods**” (→ 14.15).

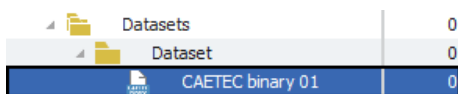
A signal can be used in several classing processes simultaneously.

Some methods can be applied with different numbers of signals. This is what is referred to as the dimension of the particular classing. 1D, 2D, 3D specify how many signals are joined in a class (joint classing). When joined, the classes of the signals form a matrix. Counting is always performed in the matrix element in which the counting conditions for all the involved signals are met by the same sampling instance.

### 14.14.1 Tree elements for CAETEC ASCII

Including a “CAETEC ASCII” file in your dataset will add one new child element called “CAETEC ASCII xx” to your tree element “Dataset”.

Multiple “CAETEC ASCII” files can be added to your dataset. They will be labeled equally and the counter will be raised by one for each new file. This may be necessary, if multiple classings should be included in the dataset, as each “CAETEC ASCII” file can only contain one classing.



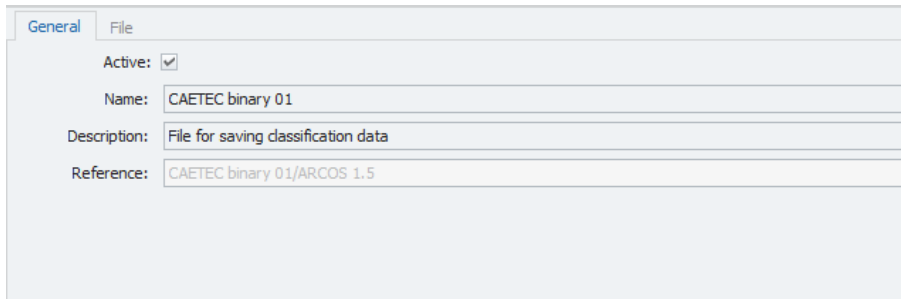
▲	Datasets	0
▲	Dataset	0
▲	CAETEC binary 01	0

### 14.14.2 Details area for CAETEC ASCII

The details area contains settings regarding the "CAETEC ASCII" file.

#### General

This tab provides general settings for the selected CAETEC ASCII file.



Active:	<input checked="" type="checkbox"/>
Name:	CAETEC binary 01
Description:	File for saving classification data
Reference:	CAETEC binary 01/ARCOS 1.5

- **Active**

Allows you to activate or deactivate the selected file.

- **Name**

Give a user-defined Name to the selected formula/signal.

- **Description**

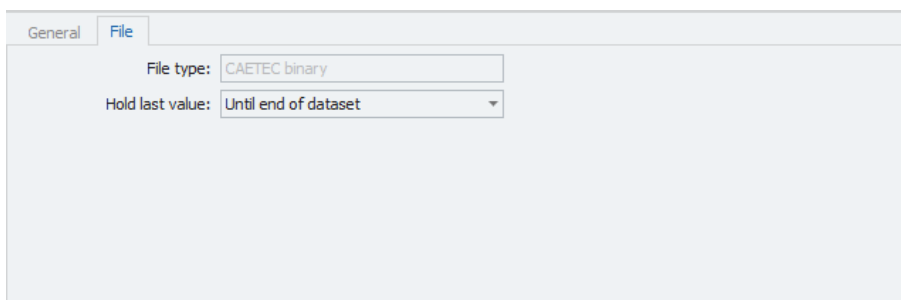
Give a user-defined description to the selected formula/signal.

- **Reference**

This field serves as the tree element's unique identifier inside the measurement task tree. It cannot be changed.

#### File

This tab provides settings regarding the creation of the file.



File type:	CAETEC binary
Hold last value:	Until end of dataset

- **File type**

Tells you the type of the created file.

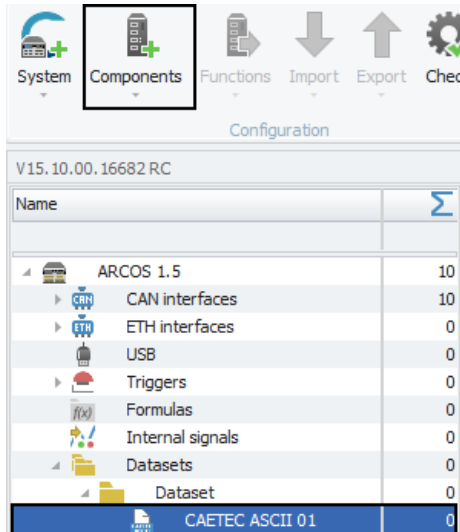
- **Hold last value**

Specify, for how long the last value of the signal will be hold.

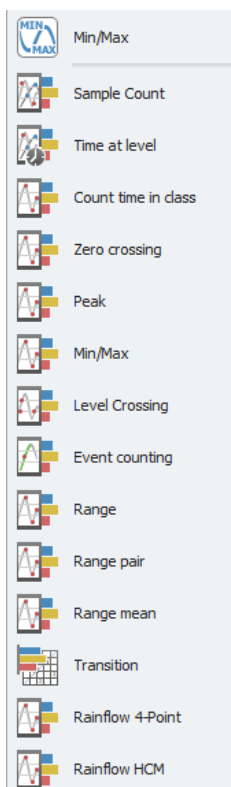


### 14.14.3 Adding a classing

In order to add a classing, select the “CAETEC binary” or “CAETEC ASCII” file, which should contain the classing, in the measurement task tree.



Then click the “Components” button in the Ribbon and choose the desired classing method.



Once you have added a classing, you may now proceed to configure this classing. Please refer to the chapter “Classing methods” (→ [14.15](#)) for instructions.

## 14.15 Classing methods

This chapter will only explain the different classing methods. A classing is always a part of either a CAETEC binary or a CAETEC ASCII file for your dataset. So in order to work with a classing, you will need to add one of these filetypes for each classing, that you wish to define.

For instructions on how to do this, please refer to the chapters “**Setting up a dataset**” (→ [14.1.4](#)) and “**CAETEC binary**” (→ [14.13](#)) or “**CAETEC ASCII**” (→ [14.14](#)).

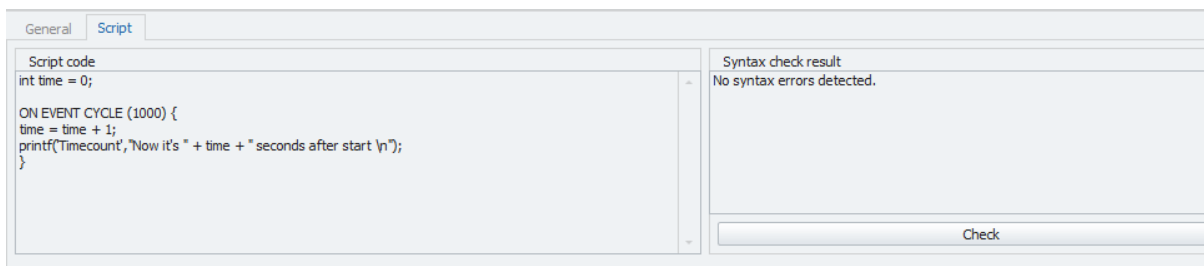
## 14.16 Script file

The “Script file” filetype is meant for recording script logging messages. The “Script file” will only contain information if it is explicitly defined as a target by a script. Otherwise it will remain empty.

### 14.16.1 Including a Script file as a target in a script

To include a “Script file” as a target in a script, you will first need to create a script. For details on scripts please refer to (→ [10](#)).

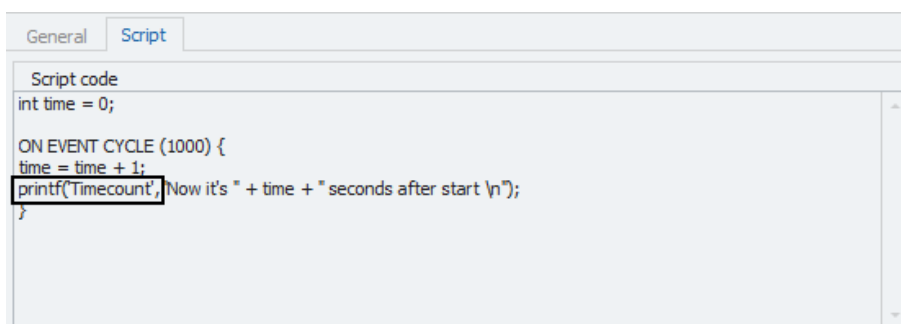
Then you will need to navigate to the tab labeled “Script” in the details area of the script.



The script you can see in the figure above will write a timecounter saying “Now it is x seconds after start” in the “Scriptfile”, where the value x starts at 0 and increases by 1 every second.

It is crucial here, that the target, to which to print the value, equals the name of the “Script file” to which the information should be written.

The first variable in parenthesis following the “printf” command defines the target. In this case the target is ‘Timecount’.



Now the last step is to navigate to the “General” tab in the details area of our “Script file” and name it exactly as the target in the script, so the script can find the target to which it is supposed to write.

General File

Active:

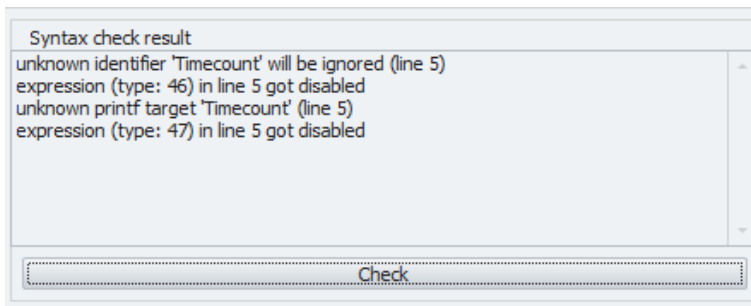
Name: Timecount

Description: File for saving script logging messages

Reference: Timecount/ARCOS 1.5

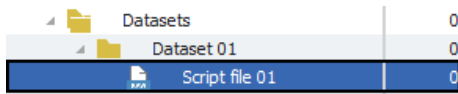


If the target in the script and the “Script file” name are not equal, at syntax check in the script you will get a syntax error looking like this.



### 14.16.2 Tree elements for Script file

Including the “Script file” filetype in your dataset will add one new child element labeled “Script file xx” to your tree element “Dataset”.

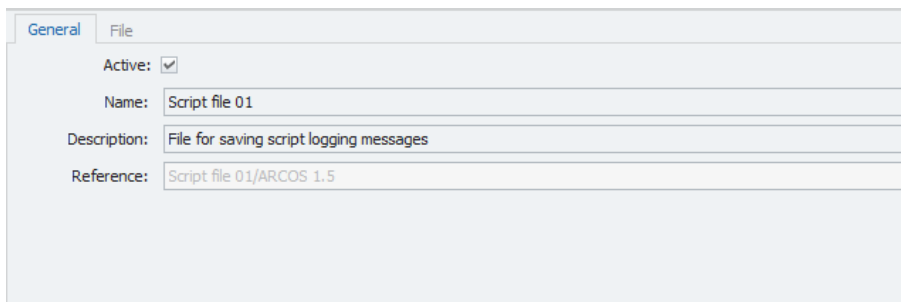


### 14.16.3 Details area for Script file

The details area contains settings for the behaviour of your Script file.

#### General

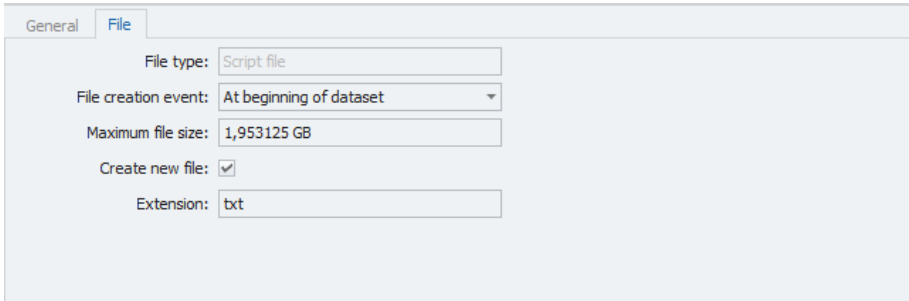
This tab provides general settings for the selected Script file.



- **Active**  
Allows you to activate or deactivate the selected file.
- **Name**  
Give a user-defined Name to the selected file.
- **Description**  
Give a user-defined description to the selected file.
- **Reference**  
This field serves as the tree element’s unique identifier inside the measurement task tree. It cannot be changed.

## File

This tab provides settings regarding the creation of the file.



The screenshot shows the 'File' tab settings. The 'File type' is set to 'Script file'. The 'File creation event' is set to 'At beginning of dataset'. The 'Maximum file size' is set to '1,953125 GB'. The 'Create new file' checkbox is checked. The 'Extension' is set to 'txt'.

- **File type**

Tells you the type of the created file.

- **File create event**

Define, when the Script file file should be created. There are two possibilities:

File create event	Characteristics
At beginning of dataset	The file will be created once at logger start.
At start of recording	The file will be created everytime, recording via the dataset is started or restarted after a pause. Starting the recording may happen at the beginning of the dataset (mode: Continuous acquisition) or via a trigger (modes: Start and pause trigger; Stop is inverted start). This may result in a splitting of the current dataset file into multiple files, as a new file is created for each time the dataset ist started.

- **Maximum file size**

Define the maximum file size. It is recommended not to raise the maximum file size above 2GB, as some third party analysis tools cannot handle files, that are larger.

- **Create new file**

If this box is marked active, a new file will be created, if the current file exceeds the maximum file size.

- **Extension**

Define the filename extension without a leading dot.

## 15 Datatransfer

Stored measurement data can later be transferred from the logger and thus made available for further analysis and processing. To do so, it is necessary, to configure one or more “Transfer events”, that will trigger the transfer, and the desired connection method used for the transfer. The first chapter of this section will explain “Transfer events” and in the following chapter will be explained configuration of the transfer connection method.

Furthermore the logger will check whenever a data transfer is happening, if a newer version of the current logger-configuration (datalog.ccmc) and firmware are available. Instructions on how to set define the correct path will be explained for each type of transfer connection in the respective chapter.

### 15.1 Transfer events

In order for data transfer to take place, it must be triggered by an event. These events can be system events, such as starting or shutting down the logger; trigger events, i.e. reactions to defined triggers; or time events, such as the arrival of a particular point in time. Multiple events can also be defined, even of the same type.

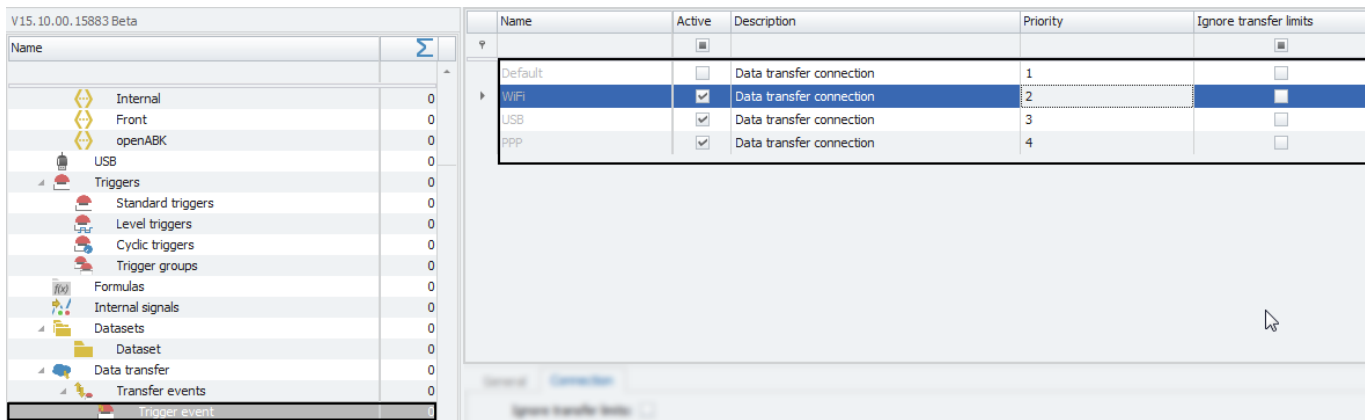
In the event that data transfer limits have been defined, these can, if the need arises, be over-ridden by specific transfer events.

### 15.1.1 General Information about transfer events

#### Multiple targets for transfer events

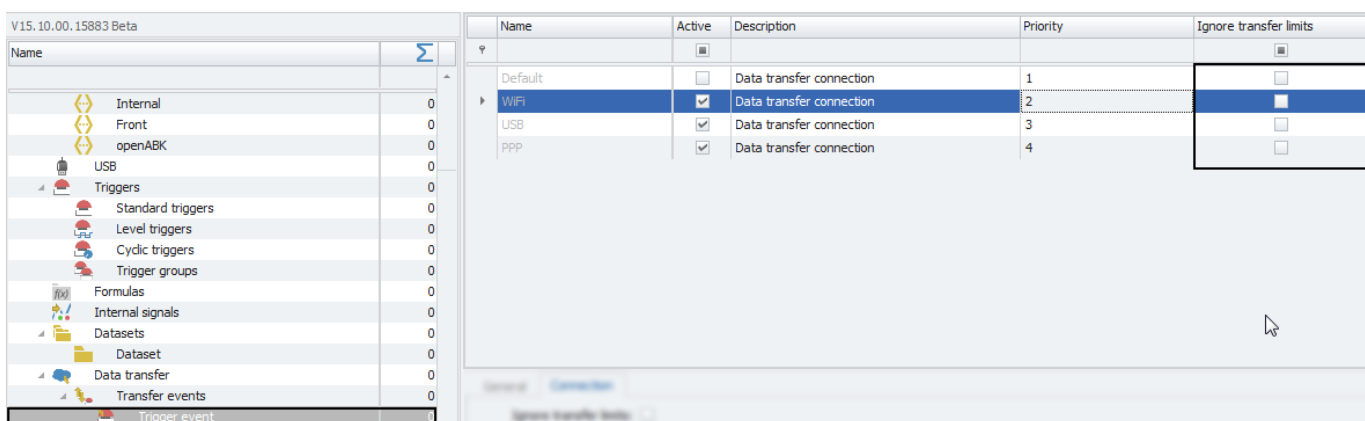
Each transfer event can use multiple connections to different targets, in order to ensure successful data transfer. In order to do so, activate every connection you want to use for this transfer event in the transfer events grid area. For instructions on transfer connections please refer to (→ 15.3).

By setting the priority you define, which connection will be used first, starting with priority 1. As soon as data transfer via one connection has been successfully completed, the other connections will be skipped and the dataset will be erased from the logger.



#### Ignore transfer limits

In the grid area of every transfer event you can choose to “Ignore transfer limits” for every transfer connection. This means, it will ignore any defined restrictions for that transfer connection. These restrictions can be set in the “Basic” settings tab in the details area of each transfer connection. Please refer to (→ 15.3) and then to the respective connection type’s section.

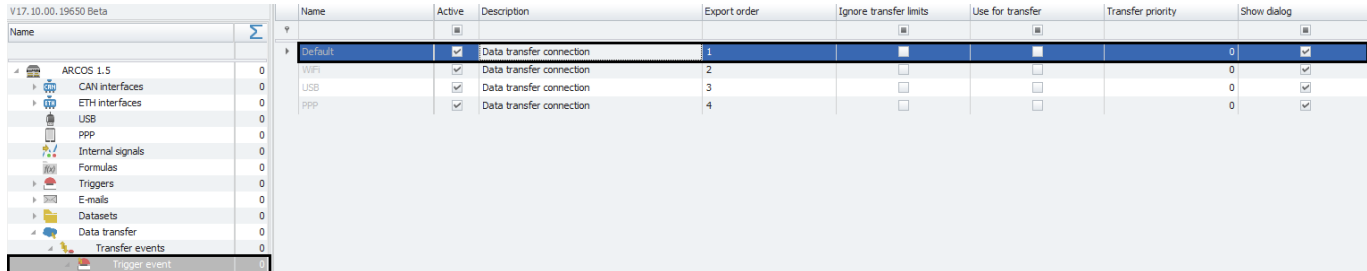


#### Default data transfer connection

Changes and errors excepted.



In the grid area of every transfer event you can choose the default data transfer connection for data transfer. This means, the connection, that has been defined as default data transfer connection will be used. This setting can be set in the “Basic” settings tab in the details area of each transfer connection. Please refer to (→ 15.3) and then to the respective connection type’s section.



Name	Active	Description	Export order	Ignore transfer limits	Use for transfer	Transfer priority	Show dialog
default	<input checked="" type="checkbox"/>	Data transfer connection	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0	<input checked="" type="checkbox"/>
WIFI	<input checked="" type="checkbox"/>	Data transfer connection	2	<input type="checkbox"/>	<input type="checkbox"/>	0	<input checked="" type="checkbox"/>
USB	<input checked="" type="checkbox"/>	Data transfer connection	3	<input type="checkbox"/>	<input type="checkbox"/>	0	<input checked="" type="checkbox"/>
PPP	<input checked="" type="checkbox"/>	Data transfer connection	4	<input type="checkbox"/>	<input type="checkbox"/>	0	<input checked="" type="checkbox"/>

### 15.1.2 Export order

The export order of a transfer event is a setting that applies only to the respective transfer event and has no effect on other transfer events. The export order determines, for a given transfer event, the order, in which the transfer event should try to use a certain connection for data transfer.

If the defined default connection is set to USB and, for example, the set order is

1. Default
2. WIFI
3. USB
4. PPP

The logger will try to start a transfer for this event to Default (USB). If no USB device is connected, it will then continue the list (WIFI,...) until the next available transfer connection and start transfer using that connection.

### 15.1.3 Transfer priority

The transfer priority of a transfer event is, unlike the export order, a global setting. It regulates the order of data transfer between all defined transfer events in your configuration. Transfer priorities do not apply to only a transfer event (Trigger event or system event) but they always apply to a combination of a transfer event and a transfer connection. The following example will try to explain how that works.

*In the example, there are three transfer events, the “Startup” event and the “Shutdown” event and a trigger event. The “Startup” event in combination with the “PPP” transfer connection has a priority of 100, so that as soon as the logger starts, it will, whenever possible, start sending data via UMTS.*

## 15.1 TRANSFER EVENTS

Name	Active	Description	Export order	Ignore transfer limits	Use for transfer	Transfer priority
Default	<input checked="" type="checkbox"/>	Data transfer connection	1	<input type="checkbox"/>	<input type="checkbox"/>	0
WiFi	<input checked="" type="checkbox"/>	Data transfer connection	2	<input type="checkbox"/>	<input type="checkbox"/>	0
PPP	<input checked="" type="checkbox"/>	Data transfer connection	3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	80
USB	<input checked="" type="checkbox"/>	Data transfer connection	4	<input type="checkbox"/>	<input type="checkbox"/>	0

The trigger event is also using a “PPP” transfer connection, because when the trigger is fired, it should also start transferring data as soon as possible. However, it is not of the highest priority to start sending right away, so it is set to a priority of 80. That means, when the trigger is fired, the combination of the trigger event has a lower priority than the combination of the “Startup” event and the “PPP” connections and will therefore wait for the “Startup” event to finish transfer before it can start a new datatransfer.

Name	Active	Description	Export order	Ignore transfer limits	Use for transfer	Transfer priority
Default	<input checked="" type="checkbox"/>	Data transfer connection	1	<input type="checkbox"/>	<input type="checkbox"/>	0
WiFi	<input checked="" type="checkbox"/>	Data transfer connection	2	<input type="checkbox"/>	<input type="checkbox"/>	255
PPP	<input checked="" type="checkbox"/>	Data transfer connection	3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	80
USB	<input checked="" type="checkbox"/>	Data transfer connection	4	<input type="checkbox"/>	<input type="checkbox"/>	0

However, as soon as the vehicle returns to the garage and is turned off, we might have a quicker transfer connection available, such as WIFI or ETH, and want datatransfer to happen using that connection. For that reason, the combination of the “Shutdown” event and the WIFI connection have the highest possible priority setting (255). That means, as soon as a shutdown condition has occurred, all current data transfer will be stopped, the logger will try to establish a WIFI connection and start datatransfer using that WIFI connection.

### 15.1.4 Trigger events

A “Trigger event” will trigger data transfer whenever the selected trigger is set. Any previously configured trigger of your system can be used to trigger data transfer.

In order to configure a “Trigger event”, you will first need to add the “Trigger event” as a “Transfer event”. To do so, select the tree element “Transfer events”, click on the “Components” button in the Ribbon and choose “Trigger event”.

# 15.1 TRANSFER EVENTS

V17.10.00.19650 Beta

Name	Active	Description	Export order	Ignore transfer limits	Use for transfer	Transfer priority
Default	<input checked="" type="checkbox"/>	Data transfer connection	1	<input type="checkbox"/>	<input type="checkbox"/>	0
WiFi	<input checked="" type="checkbox"/>	Data transfer connection	2	<input type="checkbox"/>	<input type="checkbox"/>	258
PPP	<input checked="" type="checkbox"/>	Data transfer connection	3	<input type="checkbox"/>	<input type="checkbox"/>	0
USB	<input checked="" type="checkbox"/>	Data transfer connection	4	<input type="checkbox"/>	<input type="checkbox"/>	0

File Project Signals Acquisition View

System Components Functions Import Export Check Adjust

V15.10.0

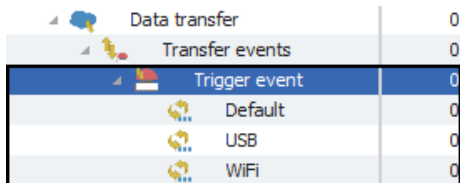
Name

- Trigger event
- Time event
- System event
- Multiple selection...
- Internal
- Front
- openABK
- USB
- Triggers
  - Standard triggers
  - Standard trigger 01
  - Level triggers
  - Cyclic triggers
  - Trigger groups
- Formulas
- Internal signals
- Datasets
- Dataset
- Data transfer
- Transfer events
- Connections

### 15.1.4.1 Tree elements for Trigger events

Adding a “Trigger event” for data transfer will add one new child element to the tree element “Transfer events”, called “Trigger event”. Multiple “Trigger events” may be added, which will each result in an extra tree element. The name of these elements can be changed in the details area of each tree element.

Each “Trigger event” will also possess various child elements representing the possible targets (transfer connections) for this trigger event.



Data transfer	0
Transfer events	0
Trigger event	0
Default	0
USB	0
WiFi	0

### 15.1.4.2 Grid area for Trigger events

The grid area for a “Trigger event” for data transfer will present you with an overview of the available connections for data transfer.

You can choose the desired connection for data transfer by ticking the “Use for transfer” tickbox, and you can override transfer limits by ticking the tickbox labeled “Ignore transfer limits”.

You may also choose multiple connections for transfer and prioritize them.

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

	Name	Active	Description	Priority	Ignore transfer limits	Use for transfer
☿		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
I	Default	<input checked="" type="checkbox"/>	Data transfer connection	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	USB	<input checked="" type="checkbox"/>	Data transfer connection	2	<input type="checkbox"/>	<input type="checkbox"/>
	PPP	<input checked="" type="checkbox"/>	Data transfer connection	3	<input type="checkbox"/>	<input type="checkbox"/>
	WiFi	<input checked="" type="checkbox"/>	Data transfer connection	4	<input type="checkbox"/>	<input type="checkbox"/>

### 15.1.4.3 Details area for Trigger events

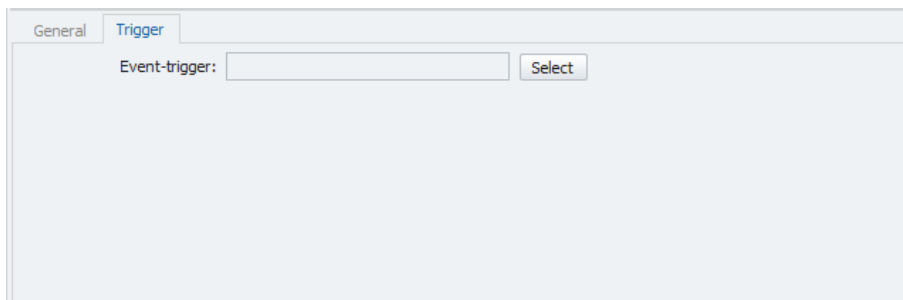
The Details area shows settings for the “Trigger event”, that has been selected in the tree. The different tabs of the details area will be explained in the following.

#### General

Please refer to (→[4.2.2](#)).

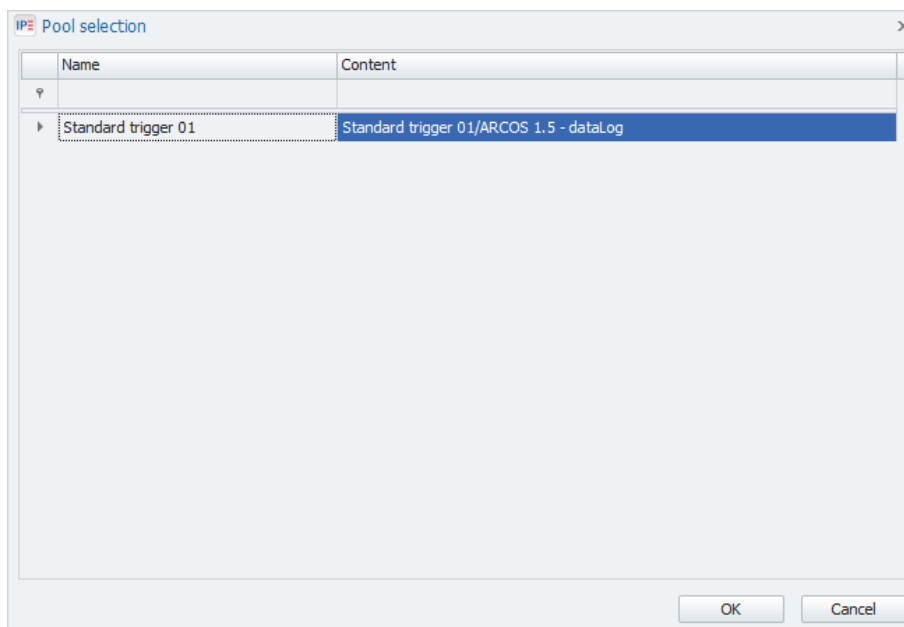
#### Trigger

This tab contains only the setting “Event trigger”, which allows you to select which trigger should be used to start data transfer.



Clicking the “Select” button will open a window, which allows you, to choose the desired trigger. Confirm with “OK”.

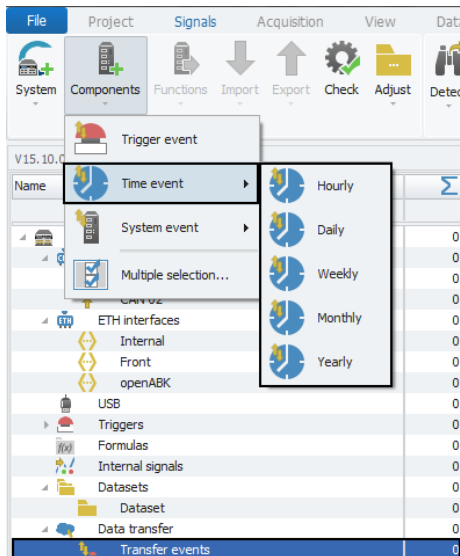
Once a trigger has been chosen, the “Trigger event” will automatically named after this event.



### 15.1.5 Time events

A "Time event" will trigger data transfer according to a set time intervall.

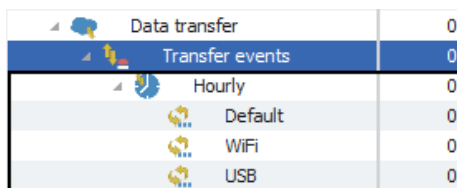
In order to configure a "Time event", you will first need to add the "Time event" as a "Transfer event". To do so, select the tree element "Transfer events", click on the "Components" button in the Ribbon and choose the desired time intervall from the menu "Time event".



#### 15.1.5.1 Tree elements for Time events

Adding a "Time event" for data transfer will add one new child element to the tree element "Transfer events", called "Time event". Multiple "Time events" may be added, which will each result in an extra tree element. The name of these elements can be changed in the details area of each tree element.

Each "Time event" will also possess various child elements representing the possible targets (transfer connections) for this trigger event.



### 15.1.5.2 Grid area for Time events

The grid area for a “Time event” for data transfer will present you with an overview of the available connections for data transfer.

You can choose the desired connection for data transfer by ticking the “Use for transfer” tickbox, and you can override transfer limits by ticking the tickbox labeled “Ignore transfer limits”.

You may also choose multiple connections for transfer and prioritize them.

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

Name	Active	Description	Priority	Ignore transfer limits	Use for transfer
☺	<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
I Default	<input checked="" type="checkbox"/>	Data transfer connection	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
USB	<input checked="" type="checkbox"/>	Data transfer connection	2	<input type="checkbox"/>	<input type="checkbox"/>
PPP	<input checked="" type="checkbox"/>	Data transfer connection	3	<input type="checkbox"/>	<input type="checkbox"/>
WiFi	<input checked="" type="checkbox"/>	Data transfer connection	4	<input type="checkbox"/>	<input type="checkbox"/>

### 15.1.5.3 Details area for Time events

The Details area shows settings for the “Time event”, that has been selected in the tree. The different tabs of the details area will be explained in the following.

#### General

Please refer to (→4.2.2).

#### Time

This tab allows you to specify the time intervall for the transfer event.

General
Event

Minute:

Hour:

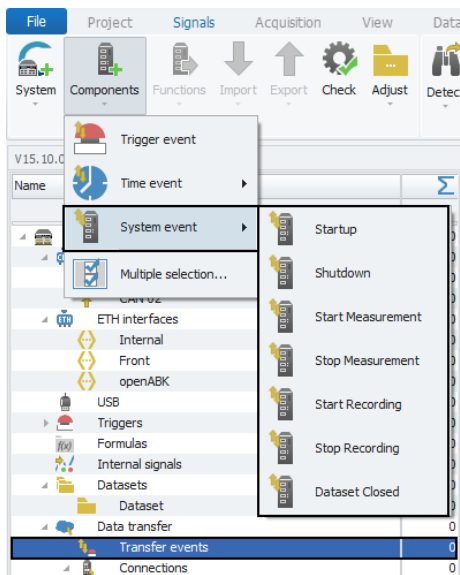
Day:

Month:

### 15.1.6 System events

A "System event" will trigger according to a set "System event" such as "Startup", "Shutdown", "Dataset closed" and others.

In order to configure a "System event", you will first need to add the "System event" as a "Transfer event". To do so, select the tree element "Transfer events", click on the "Components" button in the Ribbon and choose the desired event from the menu "System event".



#### Available System events

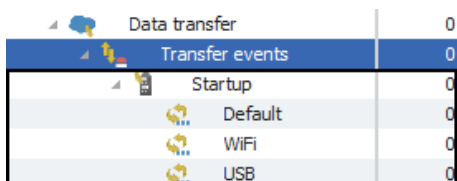
System event	Characteristics
Startup	Data transfer will start as soon as startup is finished.
Shutdown	Data transfer will start when shutdown is prompted. Shutdown will not occur until data transfer has finished.
Start Measurement	Data transfer will start as soon as measurement has started.
Stop Measurement	Data transfer will start as soon as measurement has stopped.
Start Recording	Data transfer will start as soon as recording of measurement data in a dataset has started.
Stop Recording	Data transfer will start as soon as recording of measurement data in a dataset has stopped.
Datset closed	Data transfer will start as soon as the current dataset is closed.



### 15.1.6.1 Tree elements for System events

Adding a "System event" for data transfer will add one new child element to the tree element "Transfer events", called according to the type of "System event" you have chosen. Multiple "System events" may be added, which will each result in an extra tree element.

Each "System event" will also possess various child elements representing the possible targets (transfer connections) for this trigger event.



### 15.1.6.2 Grid area for System events

The grid area for a "System event" for data transfer will present you with an overview of the available connections for data transfer.

You can choose the desired connection for data transfer by ticking the "Use for transfer" tickbox, and you can override transfer limits by ticking the tickbox labeled "Ignore transfer limits".

You may also choose multiple connections for transfer and prioritize them.

Also you can find here two important functions, which are the "Column chooser" (→4.3.1) and the "Filter editor" (→4.3.2).

Name	Active	Description	Priority	Ignore transfer limits	Use for transfer
☺	<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
I Default	<input checked="" type="checkbox"/>	Data transfer connection	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>
USB	<input checked="" type="checkbox"/>	Data transfer connection	2	<input type="checkbox"/>	<input type="checkbox"/>
PPP	<input checked="" type="checkbox"/>	Data transfer connection	3	<input type="checkbox"/>	<input type="checkbox"/>
WiFi	<input checked="" type="checkbox"/>	Data transfer connection	4	<input type="checkbox"/>	<input type="checkbox"/>

### 15.1.6.3 Details area for System events

As "System events" are definite predefined events, there are no further settings for these events available in the details area. It contains only the "General" tab, which will allow you to give a user specific description to the event. Please refer to (→4.2.2).

## 15.2 Transfer event targets

Transfer events need to have at least one target defined, in order for transfer to happen. Transfer event targets are transfer connections, which get assigned to a transfer event. For instructions regarding the transfer connections please refer to (→ [15.3](#)).

### 15.2.1 Tree elements for transfer event targets

Each transfer event in the measurement task tree has a number of child elements, corresponding the transfer connections that have been configured so far. These child elements are the transfer event's targets.

▲	Data transfer	0
▲	Transfer events	0
▲	Trigger event	0
▲	Default	0
▲	USB	0
▲	WiFi	0
▲	LAN	0

### 15.2.2 Grid area for transfer event targets

The grid area for a transfer event target will present you with an overview of all the currently defined datasets, that can be set for data transfer.

Also if the transfer event targets transmit mode has been set to "Custom selection" (→ [15.2.3](#)), the tickbox "Transmit" allows you to set which datasets will be transmitted.

Also you can find here two important functions, which are the "Column chooser" (→ [4.3.1](#)) and the "Filter editor" (→ [4.3.2](#)).

	Name	Active	Description	Transmit
♀		<input type="checkbox"/>		<input type="checkbox"/>
▶	Dataset 01	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
	Dataset 02	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
	Ring buffer 01	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>

### 15.2.3 Details area for transfer event targets

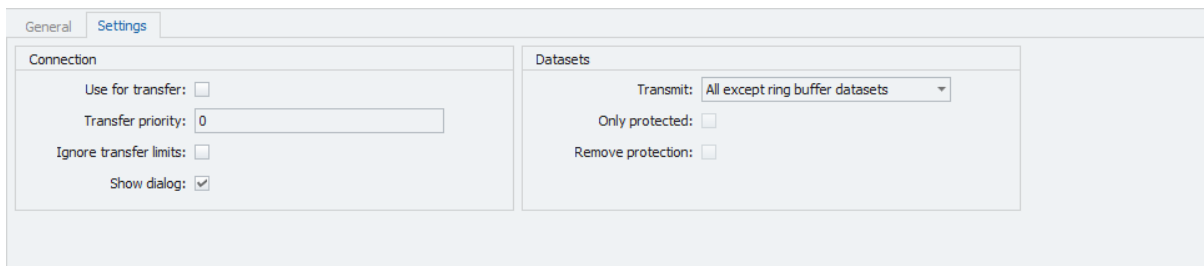
The details area for transfer event targets provides settings regarding the target.

#### General

Please refer to (→[4.2.2](#)).

#### Settings

Transfer event target specific settings.



- **Use for transfer**  
Set if you want to use this connection for datatransfer.
- **Transfer priority**  
Set the transfer priority for this Transfer event/Transfer connection combination. For further details please see (→ [15.1.3](#)).
- **Ignore transfer limits**  
Allows you to override transfer limits.
- **Show dialog**  
If marked active, a dialog will inform the user about active datatransfer for this connection on connected displays.
- **Transmit**  
Choose which datasets you wish to transmit via this connection. If you set the mode to "Custom selection", you may set the datasets you want to transmit in the grid area, using the tickbox "Transmit".
- **Only protected (Ringbuffer datasets only)**  
If set, the ringbuffer datasets will only be transmitted if they are protected.
- **Remove protection (Ringbuffer datasets only)**  
If set, the ringbuffer datasets' protection will be removed once transmitted.

## 15.3 Transfer connections

In order for data to be transferred, a transfer connection has to be set up. There are four different types of connections, that can be set up:

- USB (→[15.3.1](#))
- WIFI (→[15.3.2](#))
- LAN (→[15.3.3](#))
- PPP/UMTS (→[15.3.4](#))

### 15.3.1 Data transfer via USB

USB is the predefined default connection for data transfer and therefore it is included in the measurement task tree by default. Via USB it is possible to transfer data to and from an external storage device.

Configuration of a USB connection happens exclusively inside the details area of the tree element "USB".

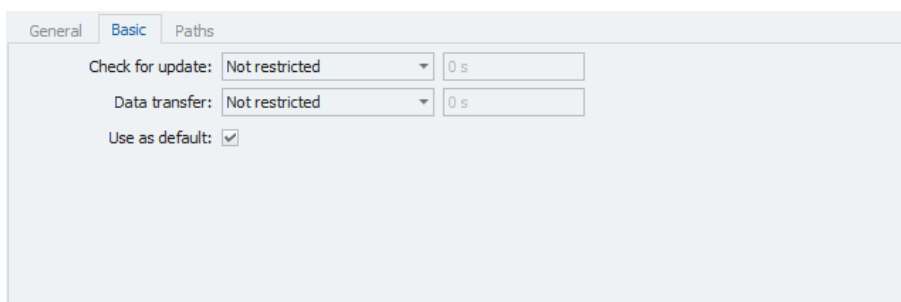
#### 15.3.1.1 Details area for USB

##### General

Please refer to (→[4.2.2](#)).

##### Basic

This tab contains basic settings for the connection.



Setting	Value
Check for update:	Not restricted (dropdown)   0 s (text field)
Data transfer:	Not restricted (dropdown)   0 s (text field)
Use as default:	<input checked="" type="checkbox"/>

- **Check for update**

Allows you to define, how often and when the logger should check for updated configuration or firmware.

If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

- **Data transfer**

Allows you to restrict, how often data transfer from the logger may occur.

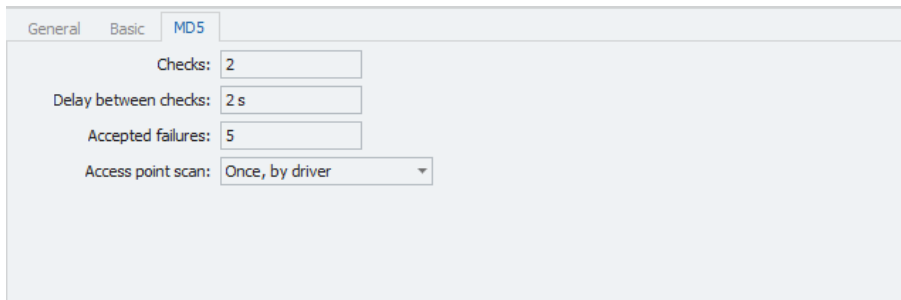
If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

- **Use as default**

Tick or untick this box in order to make this your default connection for data transfer.

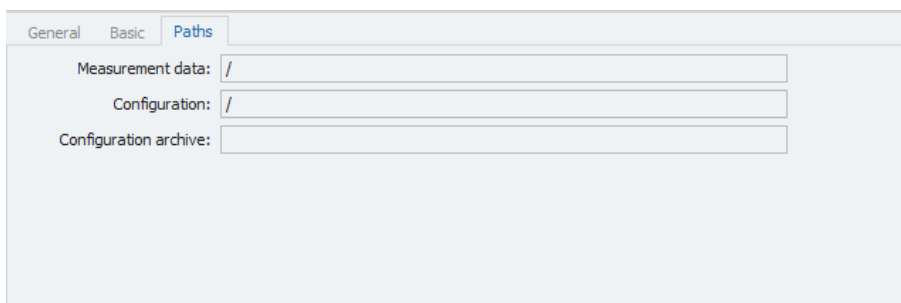
## MD5

This tab provides settings regarding the MD5 check of transferred files.



## Paths

This tab allows you to set the paths for data transfer.



- **Measurement data**

Define the path, where measurement data should be stored.

- **Configuration (cfgdir)**

Define the path, where the logger will check for a newer version of the current configuration (datalog.ccmc) and firmware.

If there is a newer version, the logger will download it, append the current timestamp in the filename, and apply/install it at the next possible moment.

- **Configuration archive (optional)**

Define the subpath for previous logger configurations and firmwares. If this subpath has been defined, the logger will copy the previously used configuration/firmware here, when he receives a newer version. If this subpath is not defined, the logger will leave the file in the "Configuration" path, that has been defined before.

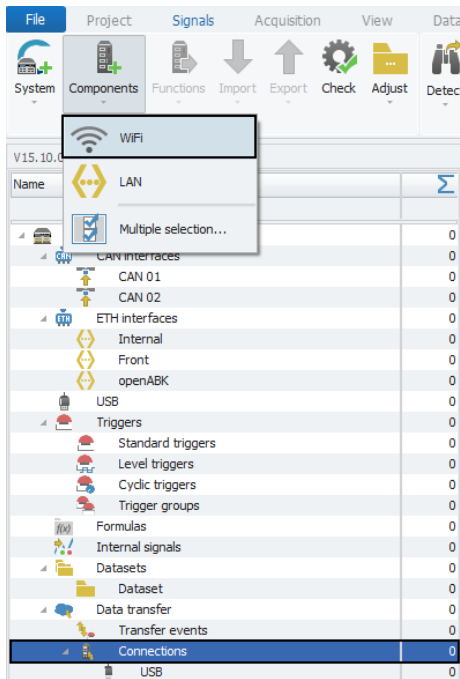


In the field "Measurement data" and "Configuration" project parameters can be used as variables. For more information please refer to (→5.6).

### 15.3.2 Data transfer via WIFI

In order to transfer data via WIFI, you will first need to add a WIFI connection to your system. To do so, select the tree element "Connections", click on the "Components" button in the Ribbon and then choose "WIFI".

After having set up the WIFI connection, you will need to set up a fileserver to which to transfer data. For instructions on how to set up a fileserver please refer to the section "Fileserver" (→ [15.3.6](#)).



### 15.3.2.1 Tree elements for WIFI connections

Adding a WIFI connection for data transfer will add two new child elements to the tree element "Connections":

- **WIFI**  
Represents the WIFI connection itself.
- **Access point xx**  
Represents the WIFI access point. A WIFI connection can have various access points, in order to connect to different networks. To add a new access point, select the tree element WIFI, then click the "Components" button in the Ribbon and select "Access point".

Data transfer	0
Transfer events	0
Connections	0
USB	0
WIFI	0
Access point 01	0

### 15.3.2.2 Grid area for WIFI connections

The grid area for a "WIFI connection" for data transfer will present you with an overview of all the currently defined access points for WIFI connections.

Also you can find here two important functions, which are the "Column chooser" (→4.3.1) and the "Filter editor" (→4.3.2).

Name	Active	Description	Priority
Access point 01	<input checked="" type="checkbox"/>	WIFI access point	1

### 15.3.2.3 Details area for WIFI

The details area provides settings either for the WIFI connection in general or for a specific access point, depending on which element has been selected in the tree.

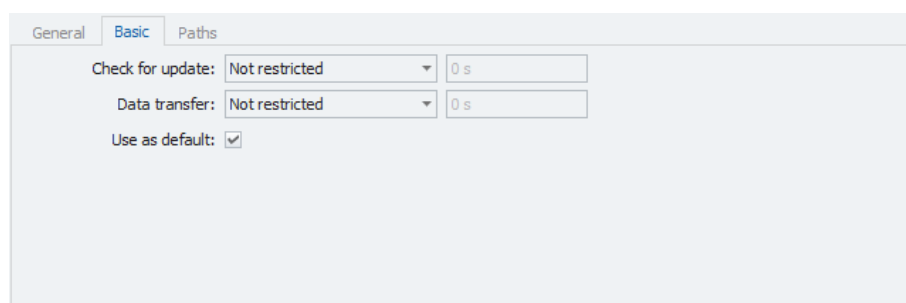
#### WIFI Settings

##### General

Please refer to (→[4.2.2](#)).

##### Basic

This tab contains basic settings for the connection.



General	Basic	Paths
Check for update: Not restricted (dropdown) 0 s (input)		
Data transfer: Not restricted (dropdown) 0 s (input)		
Use as default: <input checked="" type="checkbox"/>		

- **Check for update**

Allows you to define, how often and when the logger should check for updated configuration or firmware.

If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

- **Data transfer**

Allows you to restrict, how often data transfer from the logger may occur.

If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

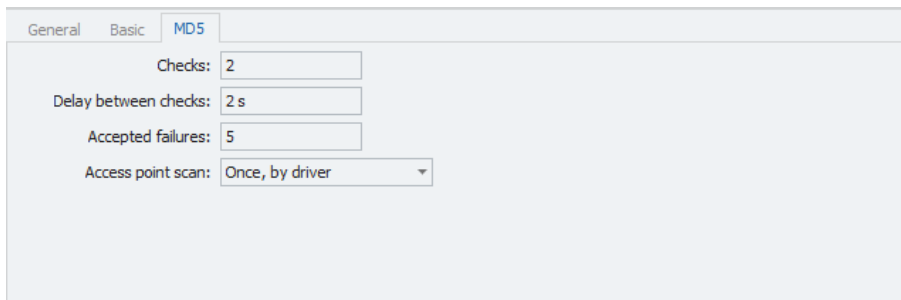
- **Use as default**

Tick or untick this box in order to make this your default connection for data transfer.



## MD5

This tab provides settings regarding the MD5 check of transferred files.



Setting	Value
Checks	2
Delay between checks	2 s
Accepted failures	5
Access point scan	Once, by driver

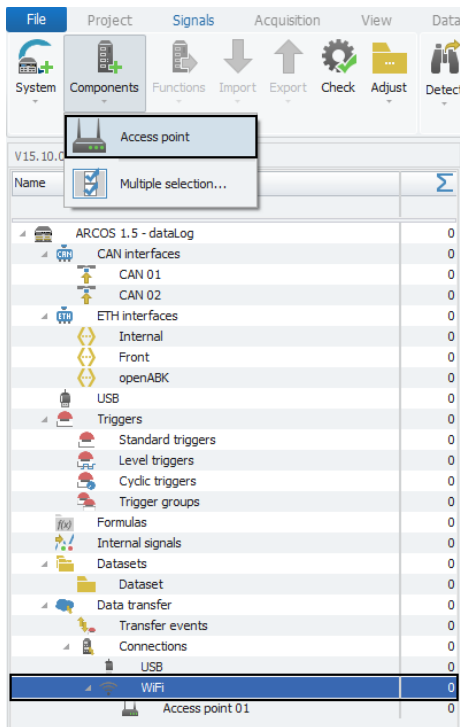
- **Checks**  
Define how often the check should be executed.
- **Delay between checks**  
Define the delay between two checks.
- **Accepted failures**  
Define the number of accepted failures before aborting the data transfer.
- **Access point scan**  
Specify the method how to search for WIFI access points.

## Access point Settings

Each WIFI connection can have multiple access points, in order to be able to connect to different wireless networks. By setting the priority of the access points in the grid area, you define, which access point will be used first, starting with priority 1. As soon as data transfer via one access point has been successfully completed, the other access points will be skipped and the dataset will be erased from the logger.

In order to add an extra access point, select the tree element "WIFI", click on the "Components" button in the Ribbon and the choose "Access point".

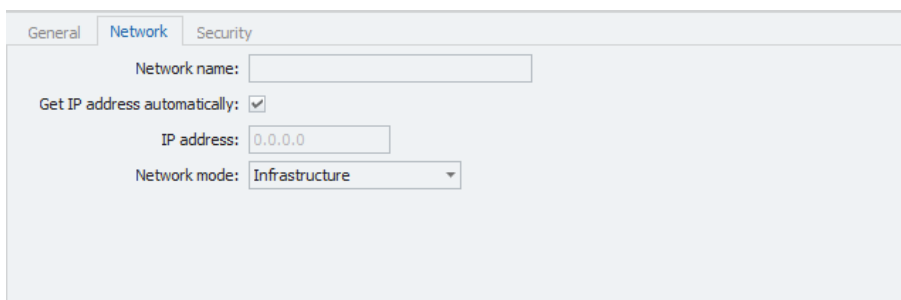
### General



Please refer to (→[4.2.2](#)).

### Network

This tab contains settings regarding the network to which the logger will connect.



- **Network name**

Define the name of the WIFI network (SSID).

Changes and errors excepted.

- **Get IP address automatically**

Define whether the logger will expect the allocation of a valid IP address automatically by a DHCP server, or whether you want to manually set an IP address.

- **IP address**

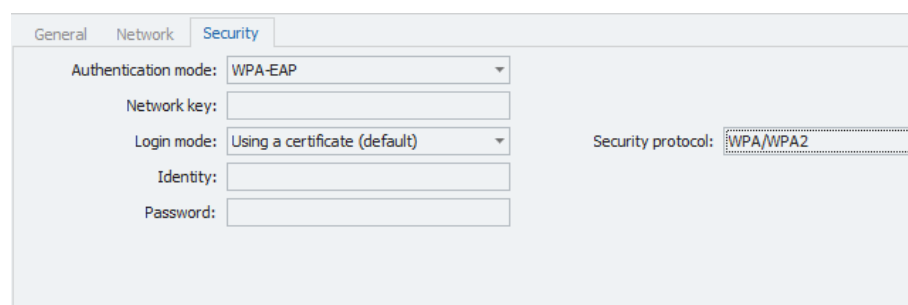
If DHCP is disabled, this field allows you to manually enter an IP address.

- **Network mode**

Allows you to choose the mode of the network you wish to connect to.

## Security

This tab provides security settings regarding the access point and the network you wish to connect to.



The screenshot shows a configuration window with three tabs: 'General', 'Network', and 'Security'. The 'Security' tab is active. It contains the following fields:

- Authentication mode:** A dropdown menu set to 'WPA-EAP'.
- Network key:** An empty text input field.
- Login mode:** A dropdown menu set to 'Using a certificate (default)'.
- Identity:** An empty text input field.
- Password:** An empty text input field.
- Security protocol:** A dropdown menu set to 'WPA/WPA2'.

- **Authentication mode**

Select the authentication mode of the network.

- **Network key**

Type in the authentication key of the network.

- **Login mode**

If the authentication mode of the network is WPA-EAP, then you may here set whether to login with a certificate or not.

- **Security protocol**

If the authentication mode of the network is WPA-EAP, then you may here set the security protocol. The default setting "WPA/WPA2" will automatically choose the correct protocol.

- **Identity**

If the authentication mode of the network is WPA-EAP, then you may here enter your identity/username.

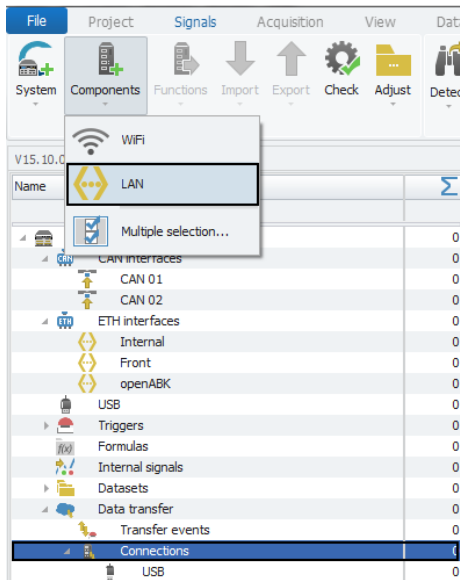
- **Password**

If the authentication mode of the network is WPA-EAP, then you may here enter your password.

### 15.3.3 Data transfer via LAN

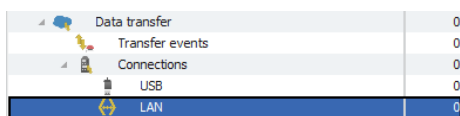
In order to transfer data via LAN, you will first need to add a LAN connection to your system. To do so, select the tree element "Connections", click on the "Components" button in the Ribbon and then choose "LAN".

After you have set up the LAN connection, you will need to additionally set up a fileserver, where the transferred data will be stored. Please refer to the section "Fileserver" (→ [15.3.6](#)).



#### 15.3.3.1 Tree elements for LAN connections

Adding a LAN connection for data transfer will add one new child element called "LAN" to the tree element "Connections"



#### 15.3.3.2 Details area for LAN

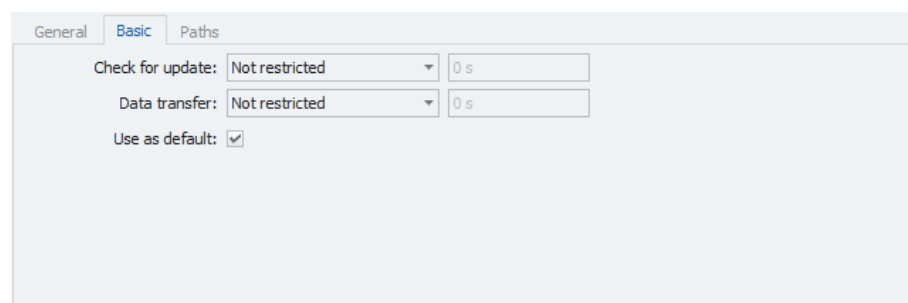
The details area provides settings for the LAN connection.

##### General

Please refer to (→ [4.2.2](#)).

## Basic

This tab contains basic settings for the connection.



The screenshot shows the 'Basic' tab selected in a settings window. It contains three rows of controls: 'Check for update' with a dropdown menu set to 'Not restricted' and a text field set to '0 s'; 'Data transfer' with a dropdown menu set to 'Not restricted' and a text field set to '0 s'; and 'Use as default' with a checked checkbox.

- **Check for update**

Allows you to define, how often and when the logger should check for updated configuration or firmware.

If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

- **Data transfer**

Allows you to restrict, how often data transfer from the logger may occur.

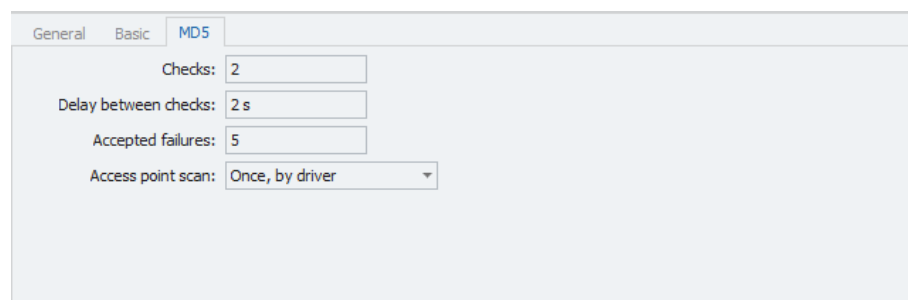
If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

- **Use as default**

Tick or untick this box in order to make this your default connection for data transfer.

## MD5

This tab provides settings regarding the MD5 check of transferred files.



The screenshot shows the 'MD5' tab selected in a settings window. It contains four rows of controls: 'Checks' with a text field set to '2'; 'Delay between checks' with a text field set to '2 s'; 'Accepted failures' with a text field set to '5'; and 'Access point scan' with a dropdown menu set to 'Once, by driver'.

- **Checks**

Define how often the check should be executed.

- **Delay between checks**

Define the delay between two checks.

- **Accepted failures**

Define the number of accepted failures before aborting the data transfer.

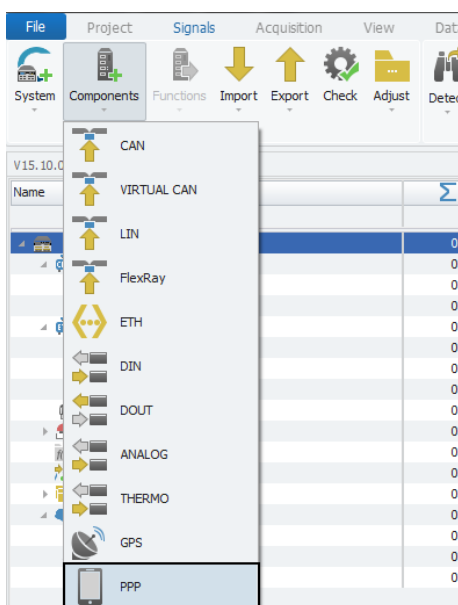
### 15.3.4 Data transfer via PPP/UMTS

In order to transfer data via PPP/UMTS, you will first need to configure a PPP/UMTS connection. To do so, please refer to (→ [15.3.4.1](#)). Once a PPP/UMTS connection has been set up, you will need so set the basic settings for PPP data transfer and you will also need to set up a fileserver. This chapter treats the basic settings for data transfer via PPP.

For instructions on how to set up a fileserver please refer to the section "Fileserver" (→ [15.3.6](#)).

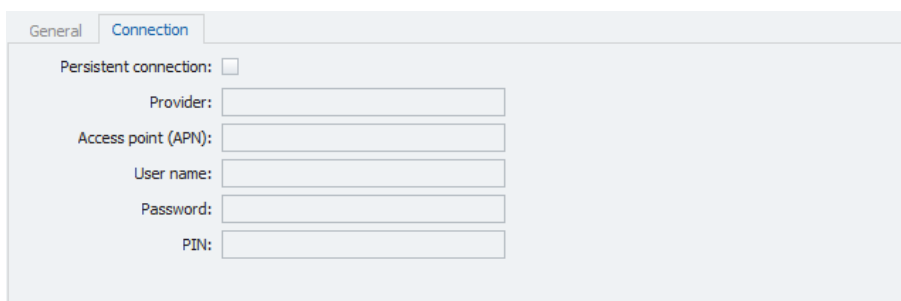
#### 15.3.4.1 Setting up a PPP/UMTS connection

To set up a PPP/UMTS connection, select your system (Arcos,  $\mu$ Cros,  $\mu$ Cros XL) in the tree, click the "Components" button in the Ribbon and choose "PPP".



This will add two elements called "PPP" to the measurement task tree. One as a child element to the main system (Arcos,  $\mu$ Cros,  $\mu$ Cros XL) and one as a child element to the tree element "Connections". Select the first of the two, navigate to the tab "Connection" in the details area and fill in the access data. This data can be obtained from your simcard provider.

The option "Persistent connection" allows you to maintain a connection not only during data transfer but also during measurement.



The screenshot shows the 'Connection' tab in the software interface. It contains the following fields and options:

- Persistent connection:**
- Provider:**
- Access point (APN):**
- User name:**
- Password:**
- PIN:**



From V17.10.00 on it is possible to fill in a PIN code to unlock the SIM card. It is therefore not longer necessary to remove the PIN code from the SIM card.

#### 15.3.4.2 Details area for PPP/UMTS

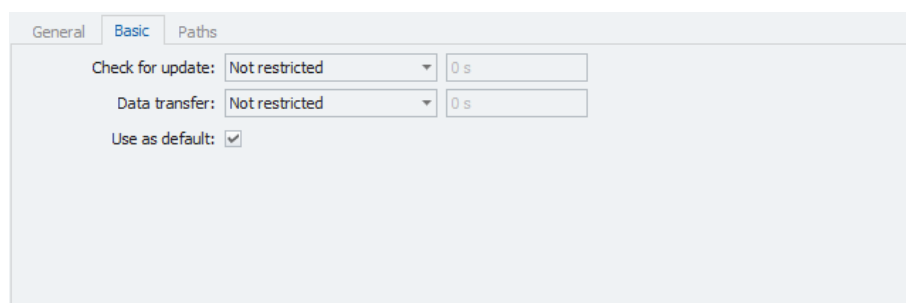
Configuration of a USB connection happens exclusively inside the details area of the tree element "USB".

##### General

Please refer to (→[4.2.2](#)).

##### Basic

This tab contains basic settings for the connection.



General	Basic	Paths
Check for update: Not restricted (dropdown) 0 s (input field)		
Data transfer: Not restricted (dropdown) 0 s (input field)		
Use as default: <input checked="" type="checkbox"/>		

- **Check for update**

Allows you to define, how often and when the logger should check for updated configuration or firmware.

If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

- **Data transfer**

Allows you to restrict, how often data transfer from the logger may occur.

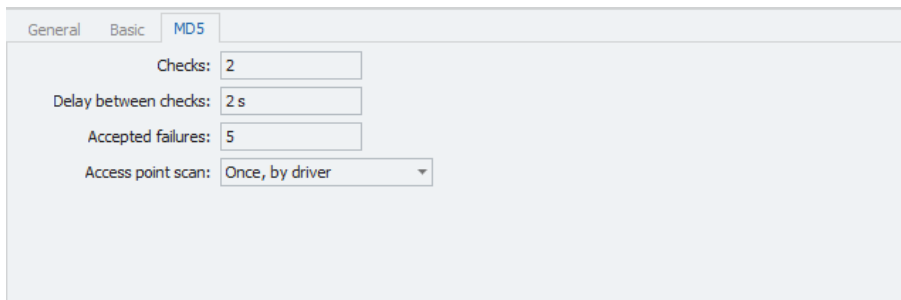
If the option "Interval" is selected, the time interval may be set in the field on the right of the dropdown menu.

- **Use as default**

Tick or untick this box in order to make this your default connection for data transfer.

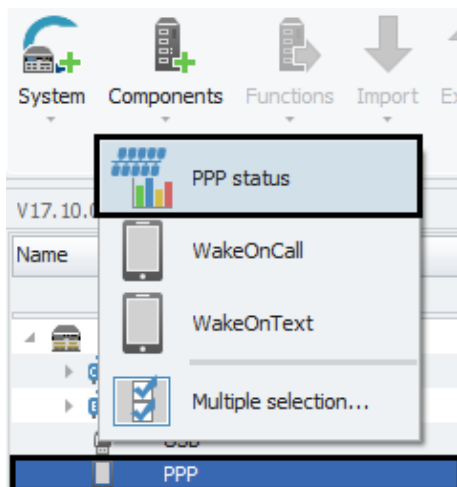
## MD5

This tab provides settings regarding the MD5 check of transferred files.



### 15.3.4.3 PPP status signals

The PPP module of the logger provides a list of status signals regarding the PPP connection. To access these signals, the "PPP status" component needs to be added to the configuration. To do so, select the "PPP" interface in the tree, click the "Components" button in the Ribbon and then choose "PPP status".



The grid area of the new tree element "PPP status" provides you with an overview of the available "PPP status" signals. These signals are of the type "Internal signal", for further information regarding these signals please refer to the chapter "Internal signals properties" (→[8.22.3](#)).



### Overview of PPP status signals

Signal	Meaning	Unit
Connection status	0 = not registered (not searching) 1 = registered (home network) 2 = not registered (searching) 3 = registration denied 5 = registered (roaming)	-
Connection type	1 = GSM 2 = UMTS 3 = GSM (with EDGE) 4 = UMTS (with HSDPA) 5 = UMTS (with HSUPA) 6 = UMTS (with HSDPA and HSUPA) 7 = LTE	-
Network operator	Information about your SIM card's network operator	-
Signal strength	Information about the quality of the signals	(dBm)
Bit error rate	0 = <0,2% 1 = 0.2%-0.4% 2 = 0.4%-0.8% 3 = 0.8%-1.6% 4 = 1.6%-3.2% 5 = 3.2%-6.4% 6 0 6.4%-12.8% 7 = >12.8%	(error bits/total bits)

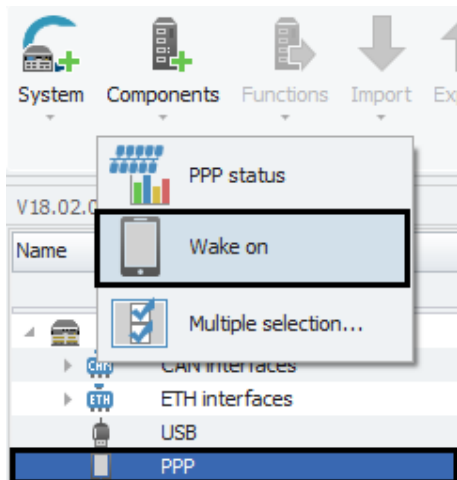
### 15.3.5 Wake on Call/Text

The two functions “Wake on Call” and “Wake on Text” are functions of the “PPP” interface and both allow to wake up the logger remotely via a mobilephone.

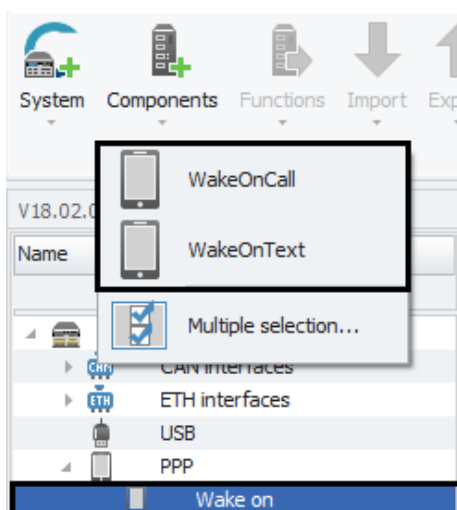
In the case of “Wake on Call”, a phone number will be defined in the configuration and if the logger receives a call from the configured number, it will wake up.

In the case of “Wake on Text” a passphrase will be defined in the configuration and if the logger receives a text containing the configured passphrase, it will wake up.

To add “Wake on Call/Text” you will first need to add the “Wake on” interface. To do so select the “PPP” element in the measurement task tree, click on the “Components” button in the Ribbon and then select “Wake on”.

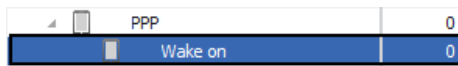


In the next step select the “Wake on” interface in the measurement task tree, click the “Components” button in the Ribbon and then choose “WakeOnCall” or “WakeOnText”.



### 15.3.5.1 Tree elements for Wake on Call/Text

Adding the “Wake on” interface to your system will add one new element labeled “Wake on” to your measurement task tree.



### 15.3.5.2 Grid area for Wake on Call/Text

The grid area for a “Wake on” will present you with an overview of all the currently defined “Wake on Call/Text” modules.

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).

	Name	Active	Description	Number	Passphrase
▼		<input type="checkbox"/>			
▶	Wake on Call 01	<input checked="" type="checkbox"/>	PPP Wake on Call		
	Wake on Call 02	<input checked="" type="checkbox"/>	PPP Wake on Call		
	Wake on Text 01	<input checked="" type="checkbox"/>	PPP Wake on Text		

### 15.3.5.3 Details area for Wake on Call/Text

The details area provides settings for a “Wake on Call/Text” module. Choose the module you wish to configure in the grid area and then navigate to the details area.

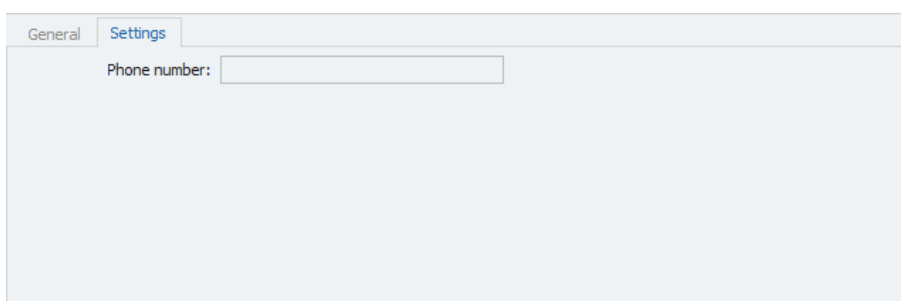
#### General

Please refer to (→4.2.2).

#### Settings

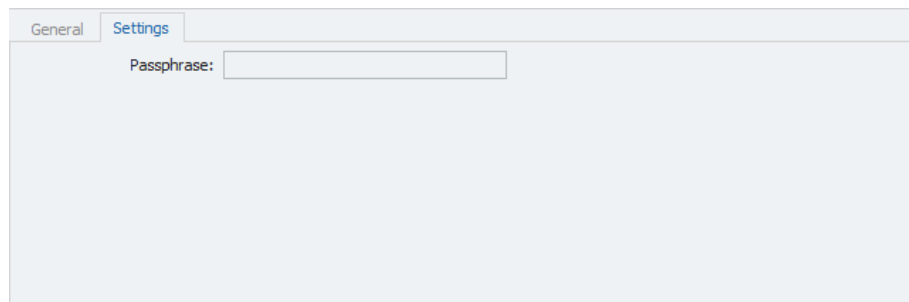
This tab contains settings for the connection.

In case of “Wake on Call” you can here define the phone number, that will wake up the logger.



The screenshot shows the 'Settings' tab selected. It contains a single text input field labeled 'Phone number:'.

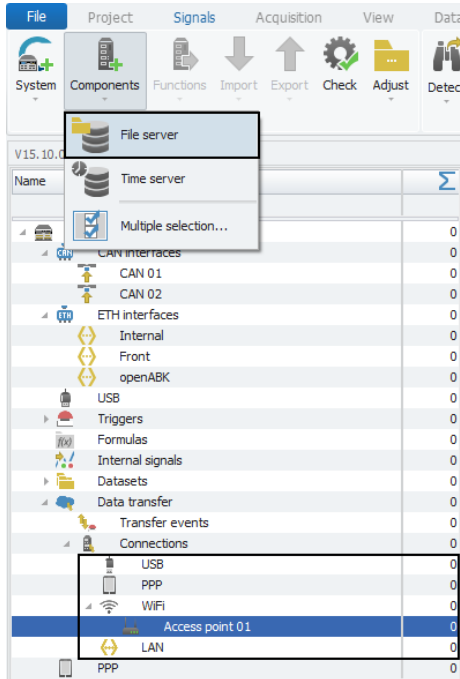
In case of "Wake on Text" you can here define the passphrase, that will wake up the logger.



The image shows a screenshot of a software settings window. At the top, there are two tabs: 'General' and 'Settings', with 'Settings' being the active tab. Below the tabs, there is a label 'Passphrase:' followed by a rectangular text input field. The rest of the window is empty.

### 15.3.6 Setting up a Fileserver

Transferring data via WIFI/LAN/PPP/UMTS requires the configuration of a fileserver to which the data will be transferred. To do so, select the tree element for the connection you wish to configure: "PPP", "LAN", "Access point xx" (for WIFI connections the access point has to be selected instead of the WIFI connection), click the "Components" button in the Ribbon and choose "Fileserver".



### 15.3.6.1 Multiple File servers

Each data transfer connection can have multiple file servers, in order to ensure successful data transfer. By setting the priority of the file servers in the grid area, you define, which file server will be used first, starting with priority 1. As soon as data transfer via one file server has been successfully completed, the other file servers will be skipped and the dataset will be erased from the logger.

Name	Active	Description	Priority
File server 01	<input checked="" type="checkbox"/>	SFTP or SCP File Server	1
File server 02	<input checked="" type="checkbox"/>	SFTP or SCP File Server	2
File server 03	<input checked="" type="checkbox"/>	SFTP or SCP File Server	3

### 15.3.6.2 Tree elements for File servers

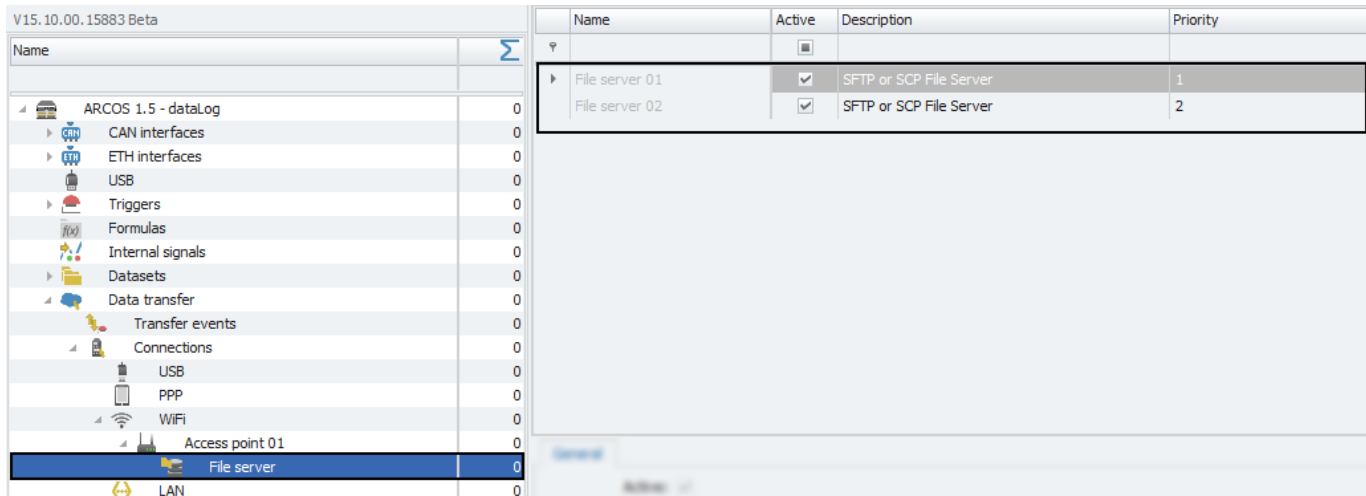
Adding a Fileserver will add one new element called "Fileserver" to the tree. It is possible to add multiple file servers, to one connection, but they will all be found under the same tree element. An overview of the file servers can be seen in the grid area.

Connections	0
USB	0
PPP	0
WIFI	0
Access point 01	0
File server	0
LAN	0

### 15.3.6.3 Grid area for File servers

The grid area for a “Fileserver” will present you with an overview of all the currently defined added file servers. It also allows you to prioritize the various file servers.

Also you can find here two important functions, which are the “Column chooser” (→4.3.1) and the “Filter editor” (→4.3.2).



Name	Active	Description	Priority
File server 01	<input checked="" type="checkbox"/>	SFTP or SCP File Server	1
File server 02	<input checked="" type="checkbox"/>	SFTP or SCP File Server	2

### 15.3.6.4 Details area for File servers

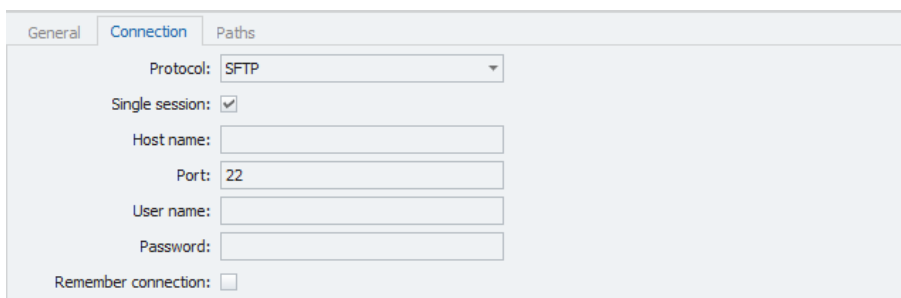
The details area provides settings for a file server. Choose the file server you wish to configure in the grid area of the tree element “File server” and then navigate to the details area.

#### General

Please refer to (→4.2.2).

#### Connection

This tab contains settings for the connection.



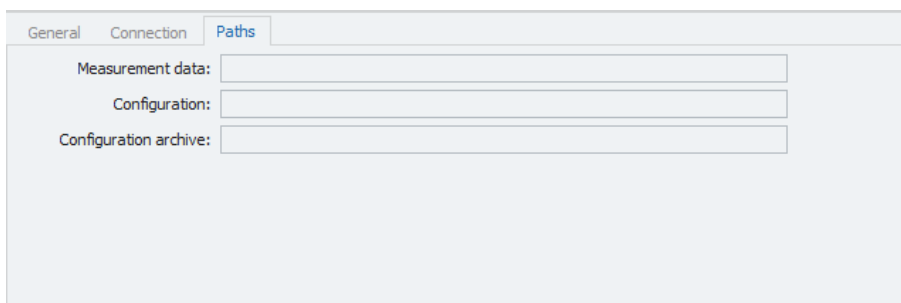
- **Protocol**  
Select the protocol used for data transfer to the file server.
- **Single session**  
If set, the SFTP single session mode will be used.

Changes and errors excepted.

- **Hostname**  
Define the IP address or hostname of the SSH file server.
- **Port**  
Define the port or hostname of the SSH file server.
- **User name**  
Fill in the user name to access the SSH file server.
- **Password**  
Fill in the password to access the SSH file server.
- **Remember connection**  
If set, after a successful transfer this server will be tried first in the future. If at any point unsuccessful, priority will be reset to 0.

## Paths

This tab allows you to set the paths for data transfer.



- **Measurement data**  
Define the path, where measurement data should be stored.
- **Configuration**  
Define the path, where the logger will check for a newer version of the current configuration (datalog.ccmc) and firmware. If there is a newer version, the logger will download it, append the current timestamp in the filename, and apply/install it at the next possible moment.
- **Configuration archive (optional)**  
Define the subpath for previous logger configurations and firmwares. If this subpath has been defined, the logger will copy the previously used configuration/firmware here, when he receives a newer version. If this subpath is not defined, the logger will leave the file in the "Configuration" path, that has been defined before.



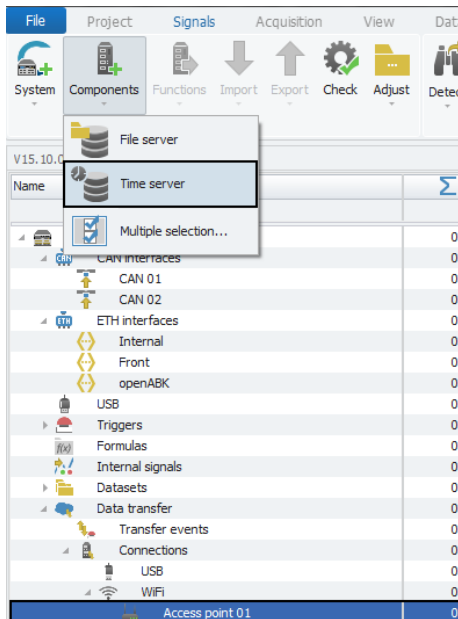
In the fields "Measurement data" and "Configuration" project parameters can be used as variables. For more information please refer to (→5.6).



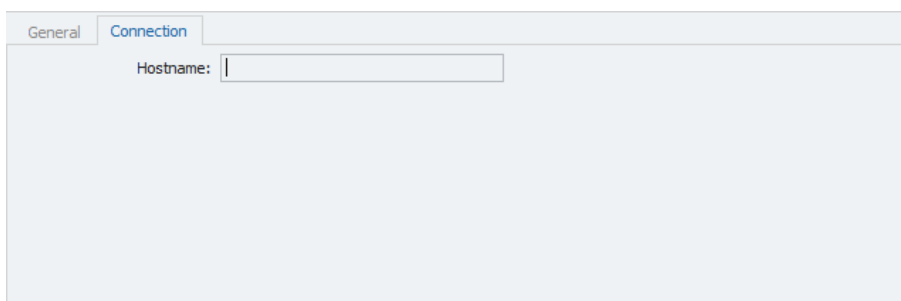
## 16 Setting up a time server

In order for your logger to always have the correct time, you will need to set up a time server to which the logger will connect and then synchronize its time with.

A time server can be configured for every LAN, WIFI or PPP/UMTS connection. To do so, select the tree element for the connection you wish to configure: "PPP", "LAN", "Access point xx" (for WIFI connections the access point has to be selected instead of the WIFI connection), click the "Components" button in the Ribbon and choose "Time server".



Then select the "Time server xx" in the grid area of the tree element "Time server xx" and navigate to the "Connections" tab in the details area and set the hostname/IP address in the field "Hostname".



## 17 Obtaining extended support

[support@caetec.de](mailto:support@caetec.de)

+49 8142 501365



# Manual

# ARCOS

# μCROS



## Hardware and Configuration

Version 1.0

Date of issue: December 4, 2018

© Copyright 2007 - 2015 by:

CAETEC GmbH  
Industriestrasse 1  
82140 Olching  
Germany

All rights reserved. Any reprinting, photocopying or translation of this manual, in whole or in part, requires advance written approval of CAETEC.

Pictures and sketches are for illustration purposes only and are not to be used as design drawings nor to serve as offer or assembly drawings.

All specifications are based on the technical status of 09/01/2014. We reserve the right to make any changes required to technically improve the equipment.

This manual has been produced with all due diligence.

CAETEC shall not be held liable for any damage resulting from the use of this manual, providing it is not due to gross negligence on our own part or the part of our legal representative or vicarious agent, and to the extent that the damage does not stem from personal injury, bodily harm or damage to health.

All related registered brands and trademarks are the property of the respective owners.

# Contents

<b>1</b>	<b>Foreword</b>	<b>3</b>
<b>2</b>	<b>Configuration</b>	<b>4</b>
2.1	Symbols	4
2.2	References	4
2.3	Notes	4
<b>3</b>	<b>Hardware</b>	<b>6</b>
3.1	Overview	6
3.2	ARCOS Basic Platform	8
3.2.1	About the Connections	9
3.2.2	Connection and LED Assignment	11
3.3	μCROS	12
3.3.1	About the connections	13
3.3.2	Connection and LED Assignment	13
3.4	Upper Expansion Chassis Unit (USB)	15
3.5	Lower Expansion Chassis Unit (ETH)	16
3.6	Interface Module for the Lower Expansion Chassis Unit	18
3.6.1	CAN	18
3.6.1.1	Input Buffer	18
3.6.1.2	Switching Output for External Hardware (not CAN interface v2)	19
3.6.1.3	Assigning the connections	19
3.6.1.4	Technical Data	21
3.6.2	LIN	21
3.6.2.1	Input Buffer	22
3.6.2.2	Power Supply on the LIN Channel	22
3.6.2.3	Setting the Baud Rate	22
3.6.2.4	Connection Assignment	22
3.6.2.5	Technical Data	23
3.6.3	FlexRay	24
3.6.3.1	Input Buffer	24
3.6.3.2	Timing Parameters	24
3.6.3.3	Connection Assignment	24
3.6.3.4	Technical Data	25
3.6.4	Analog/Digital I/O	26
3.6.4.1	Input Buffer	26
3.6.4.2	Sampling Rate	26
3.6.4.3	Connection Assignment	26
3.6.4.4	Technical Data	28
3.7	Interface Modules for the Upper Expansion Chassis Unit	29
3.7.1	WiFi	29
3.7.1.1	Applications	29
3.7.1.2	Module Placement	29
3.7.1.3	Configuration parameters	30
3.7.1.4	Technical Data	31

# 1 Foreword

## 2 Configuration

### 2.1 Symbols

Various paragraphs in this manual are marked with special symbols. These symbols have the following meanings:



This symbol highlights important information that, if ignored, may prevent successful use of the program.



This symbol refers to additional information supplementing this manual.

### 2.2 References

References to other sections of this manual are generally placed in brackets and are indicated by an arrow:

(→ [2.1](#)) refers to Section 2.1.

When this manual is read in digital form, a mouse-click on such a reference accesses the particular section of the book.

### 2.3 Notes

This manual explains the ARCOS and  $\mu$ CROS logger systems. It provides technical descriptions of the components and their technical data, along with a description of the configuration and the resulting application options.

The functionality of the systems depends on the particular firmware version in use. Therefore, all functions and parameters that are only available with a specific firmware version (or higher) have been marked with the respective version number in superscript (e.g., with 1.3<sup>1</sup>). Which particular firmware version is installed can be determined by checking the display, the Web interface or the log file.

In addition, you will find brief instructions for a variety of useful procedures and methods for working with the logger. This information is provided with no claim to completeness, and revisions shall be made as development progresses



Using the logger and display while operating a vehicle can distract the driver. Concentration on traffic should be the driver's top priority at all times.



Operating a data logger in transmission mode can interfere with the proper functioning of safety and control components, resulting in the cancellation of the respective vehicle operating permit. It is the sole responsibility of the installing party to supervise this. Moreover, it is that same party's obligation to instruct vehicle drivers on this matter, as well as to advise them on proper procedures.



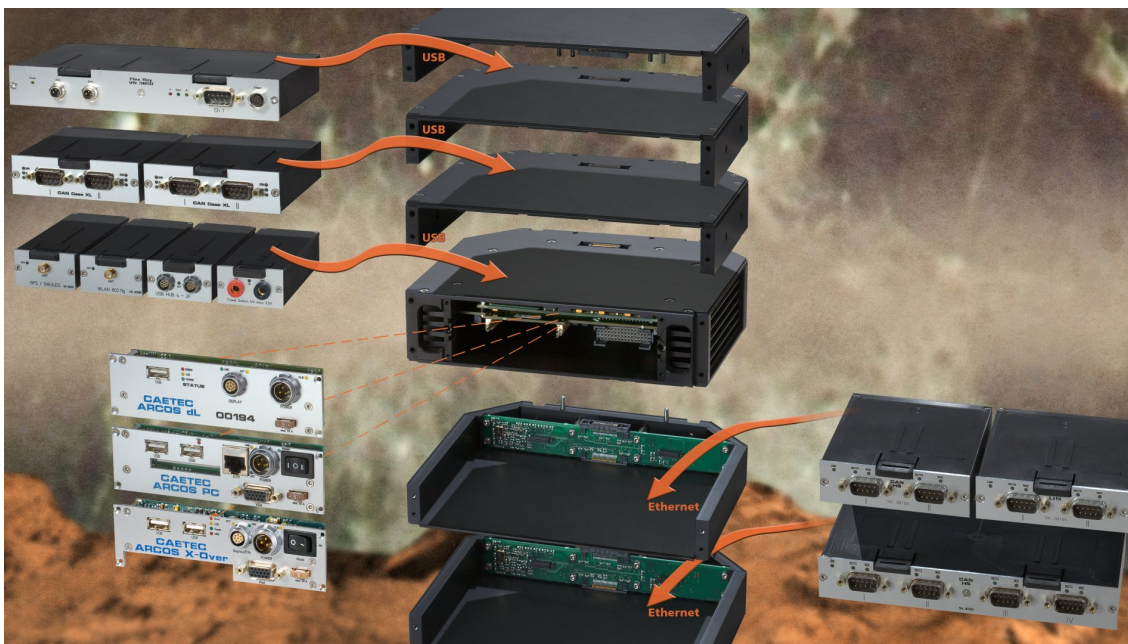
## 3 Hardware

### 3.1 Overview

The “ARCOS” hardware system comprises a number of modules connected to a basic platform unit. The functioning of the particular system (ARCOS dL, ARCOS PC or ARCOS X-Over) varies according to the types of components used.

The  $\mu$ CROS is the little “brother” of the ARCOS dL – while it is only equipped with a logger function and is less modular, it is compacter and water-tight.

The complete ARCOS system comprising the basic platform, upper an lower units and modules:



The complete logger unit, comprising the basic platform and the chassis units (without slide-in modules):



The basic platform consists of a housing containing the central processor assembly. The particular front employed determines the functionality of the system. The number of modules in the assembly is variable and depends on the interfaces required for a particular application.

A basic distinction is made between those modules that are designed to be mounted on top of the basic platform unit and those for assembly underneath. These two types (upper/lower units) are not interchangeable, if only due to their different connector channels (the so-called "backplane"). Separately, however, the chassis units for the lower (or upper) modules are interchangeable, irrespective of their order. There is no set limit to the number of chassis units assembled. The number is limited by the maximum size of the complete housing (installation space), the maximum permitted current draw, and similar peripheral parameters.

All modules are centrally connected to the supply voltage via the jack in the basic platform. Each chassis unit has its own power supply conditioning (DC/DC conversion), its own "power supply", so to speak.

The connecting sockets in the top or bottom chassis units can be covered with a protective plate. It is screwed onto the housing and is fitted with a rubber seal.

A display is auxiliary to the ARCOS dL and  $\mu$ CROS. Basically, the logger can be used without a display, as well, with no impairment of measurement. The significant advantage of the display is that it serves as the user interface for transferring the measured data at the end of a measurement (cf. Chapter "Display") or for documenting the driven track (track selection).

## 3.2 ARCOS Basic Platform

The basic platform unit comprises an x86 processor platform, to which all the information and signals are fed. The chassis units are connected to it via bus systems. All the connections between the basic platform unit and the external periphery (power supply, display,...) plug into the front plate.

The operating system with the data acquisition software, as well as the measured data, is stored on a compact flash memory card (CF card). Another option for data storage is a separate S-ATA hard drive (rotating or solid state). The removable front plate provides access to the storage media.

A so-called "heat spreader" connects the processor system to its housing. The heat produced by the processor is conveyed through the housing to the cooling fins. When required, the rear fans increase convection in the fins, but do not ventilate within the housing.

In the ARCOS dL, the USB connector is for transferring data (importing configurations or transferring measured data) and operating an external configuration provided. This port, therefore, is only used for inserting the same type of USB memory sticks or external USB hard drives as are used in PC technology (max. 500 mA). Expanding the periphery by using USB-CAN interfaces or comparable components via the USB connection is only supported for ARCOs as PC.

Along with the data lines, the ARCOS dL and X-Over display connection socket includes the power supply lines, i.e. the display receives its operating power from the basic unit. The connector is configured as an Ethernet port, connecting the basic platform unit via adapter cable with the Ethernet jack for diagnostic/webinterface with a PC. The ARCOS PC and X-Over are equipped with a standard VGA connector, while the ARCOS PC also has a standard LAN connector.

The power supply connector contains not only the pins for positive and ground (or, in automotive engineering terms, the "Clamps 30 and 31"), but also a pin for switch input (what automotive engineers often call a "Clamp 15"). A voltage level above approx. 3V on this pin starts the logger. This start capability exists in addition to start via bus messages (e.g., "Wake-on-CAN" = WoC). Regardless of whether WoX is activated, sufficient voltage at the control pin (CI 15) will start the logger.

This behavior also enables an emergency start-up, ensuring there's always a way to safely start the logger (following a configuration error, e.g., such as Start on a CAN message that never occurs).

To minimize standby current consumption, the Clamp 15 has a very high impedance (> 10 kilohm). In surroundings with high disturbance levels, one should make sure the potential is properly connected to the clamp connector and not leave the pin open – in order to avoid unexpected system start-ups.

The external fuse is a vehicle fuse, for a maximum current of 7.5 A.

As an indication of the total current draw of a typical system, we measured a current draw of approx. 3A with 12 V for the following assembly:

- basic platform

- incl. four-channel CAN modules (2 units)
- plus WLAN module



The upper limit for supply voltage is 50 V. Loggers manufactured after February 2008 are equipped with an integrated over-voltage trip in the supply line. This protects the device against voltage > 50 V DC by tripping the fuse in a controlled manner. For this purpose, the supply source must be able to supply current at least twice as high as the nominal strength of the fuse. Otherwise the over-voltage device won't trip safely, in which case, over-voltage may cause serious damage to the logger. (In a vehicle, this condition is generally met.)



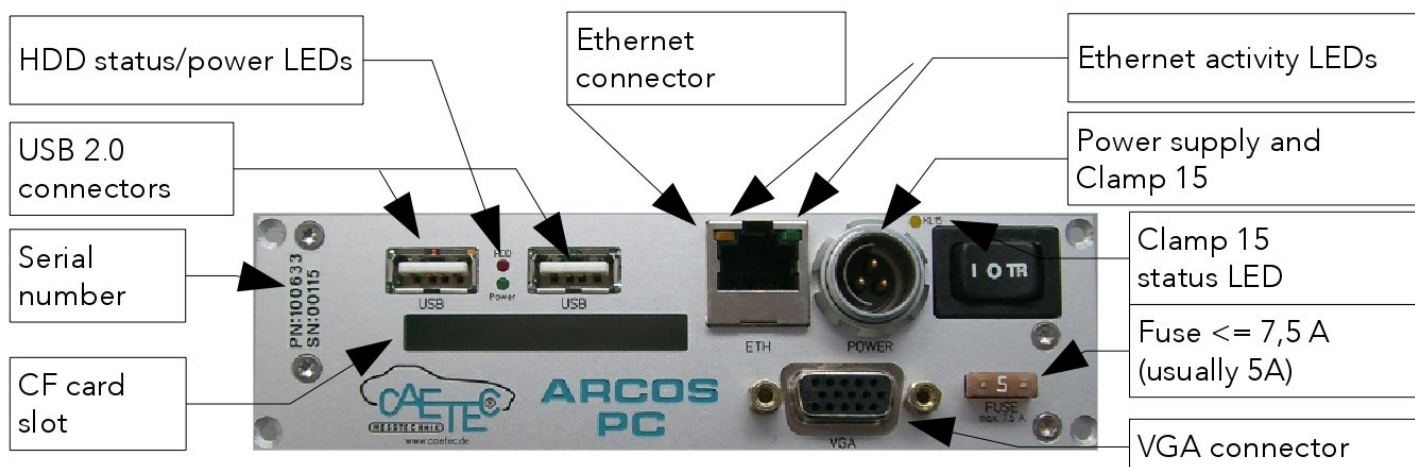
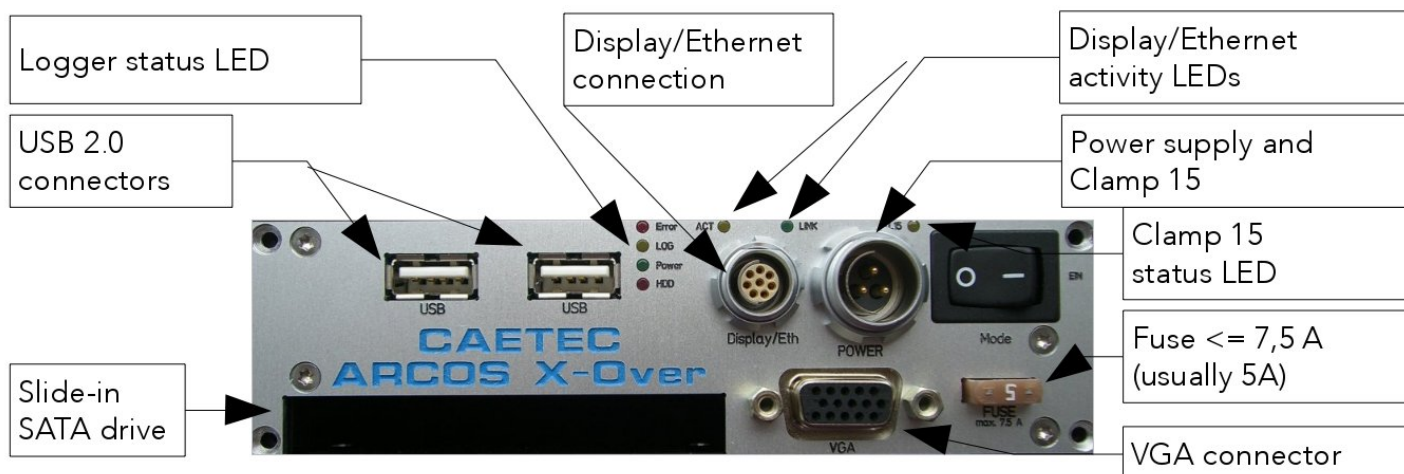
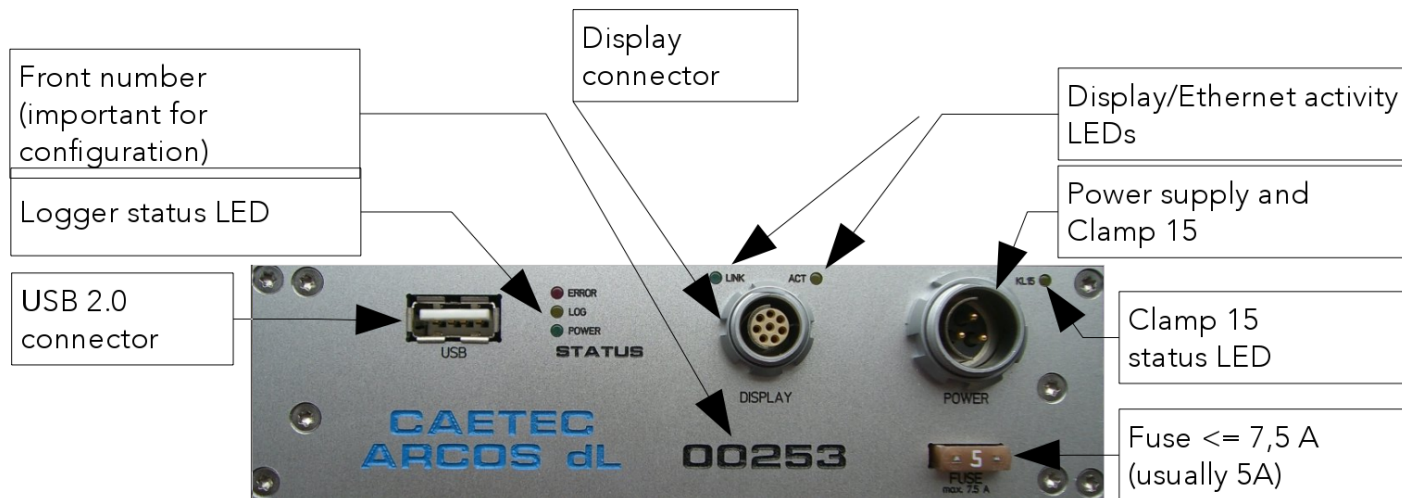
Loggers manufactured before February 2008 have no such over-voltage trip. Under no circumstances are the supply lines of these devices to be connected to voltages > 50 V DC.

The fuse may only be changed when the logger is disconnected from the power supply. (Pull the plug in the front plate)

### 3.2.1 About the Connections







### 3.2.2 Connection and LED Assignment

USB	
Standard assignment of PC technology.	Transfers configurations and measurement data (ARCOS dL), and (in the ARCOS PC) is available for other USB hardware

Logger Operating Status (ARCOS dL and X-Over)	
Error (red): Off	No error.
Briefly flashes	Firmware malfunction has occurred, fallback will start shortly.
Blinks (lit/dark: 50/50 )	Firmware running with fallback configuration
Lit (steady)	Firmware malfunction
Log (yellow): Off	No measurement
Blinks 1.3 (lit/dark: 50/50 )	Measurement running, but there is no method storing data
On	Measurement running, data are being stored.
Power (green): Off	Measurement system is off - no power supply connected
Short blinks (lit/dark: 25/75 )	System booting, due to met start condition (e.g. Clamp 15 of WoX)
Lit (steady)	Boot was successful, logger is operational. (Check the Log LED for the measurement mode.)
Long blinks (lit/dark: 75/25 )	System shutting down (takes approx. 10s), ends with the mode "short blinks"

Display Pin Assignment ARCOS dL and X-Over	
1	Supply -
2	Supply -
3	Supply +
4	Supply +
5	Data 1 (RX+, orange and white)
6	Data 2 (RX-, orange)
7	Data 3 (TX+, green and white)
8	Data 4 (TX-, green)

Display Activity LEDs (ARCOS dL and X-Over)	
Link	Lit when the cables are connected and a link has been established
Act	Blinks when data packets are being transported

Power Pin Assignment	
1	Supply +
2	Ground
3	Clamp 15

Clamp 15 Status Display	
LED on/off	Voltage level for logger start is present/not present (voltage above 3V)

VGA (ARCOS X-Over and PC)	
Standard assignment of PC technology	Connection to a standard VGA monitor

LAN (RJ45) (ARCOS PC)	
Standard assignment of PC technology	LAN connector

### 3.3 $\mu$ CROS

Unlike the ARCOS, the  $\mu$ CROS has a closed housing to which no expansion modules can be connected. Due to its hardware architecture, it runs the same  $\mu$ CROS software as an ARCOS dL; the configurations are also totally interchangeable – except that the  $\mu$ CROS has a lower computing power and supports fewer interfaces and bus interfaces.

In the  $\mu$ CROS LE version, the CAN channels are purely passive, no CAN Acknowledge is set and sending is not supported (which means there's no CCP/XCP!). However, an upgrade via software license is subsequently possible.

The  $\mu$ CROS comprises an x86 processor platform connected via internal Ethernet connector to the integrated two or four-channel CAN interface. The platform also provides two slots for any combination of GPS, WLAN or UMTS modules. Each module has an SMA jack on the front plate as antenna connector. All the  $\mu$ CROS connectors to external periphery (power supply, display, CAN, USB, the antenna connections of the custom options) are on the front plate.

The operating system with the data acquisition software and the measurement data are stored on a separate mSATA storage medium installed in the  $\mu$ CROS.

A so-called "heat spreader" connects the processor system to its housing. The heat produced by the processor is conveyed through the housing to the cooling fins. Since it produces less heat than the ARCOS, the housing is totally without fans. So the  $\mu$ CROS contains no moving parts.

The  $\mu$ CROS USB connector is for transferring data (importing configurations or transferring measured data) and operating an external configuration provided. This means this port is only used for inserting the same type of USB memory sticks or external USB hard drives (max. 500 mA) as are used in PC technology.

No provision has been made for expanding the periphery by means of USB-CAN interfaces or comparable components.

The  $\mu$ CROS display jack includes the data lines, as well as the power supply lines, i.e. the display receives its operating power from the logger. The  $\mu$ CROS connector is configured as an Ethernet port. This port is connected via adapter cable with the Ethernet jack for diagnostic/Webinterface with an PC.

The power supply jack contains not only the pins for positive and ground, but also a pin for switch input (a so-called "Clamp 15"). At this pin, a voltage level above approx. 3 V starts the logger. This start capability is provided in addition to start via bus messages ("Wake-on-CAN" = WoC). Regardless of whether WoX is activated, sufficient voltage at the control pin (CI 15) will start the logger.

To minimize standby current consumption, the Clamp 15 has a very high impedance (> 10 kilohm). In surroundings with high disturbance levels, one should make sure the potential

is properly connected to the clamp connector and not leave the pin open – in order to avoid unexpected system start-ups.

The fuse of the complete system is an SMD fuse for a maximum current of 3 A. The display power supply has been equipped with a self-healing 500 mA fuse.

With the  $\mu$ CROS, typical total current draw is virtually independent of the options selected (WLAN, UMTS, GPS):

Current draw: approx. 0.5 A with 12 V



The upper limit for the supply voltage is 36 V, without an integrated over-voltage trip in the supply line. The fuse for the total system can only be changed by CAETEC.

### 3.3.1 About the connections



### 3.3.2 Connection and LED Assignment

Power Pin Assignment	
1	Clamp 15
2	Supply +
3	Supply +
4	Supply +
5	Ground
6	Ground
7	Ground
8	n.c.



Pin Assignment of Signal Connectors	
1	CAN 1 High
2	CAN 1 Low
3	CAN 2 High
4	CAN 2 Low
5	CAN 3 High
6	CAN 3 Low
7	CAN 4 High
8	CAN 4 Low
9	Display Vcc
10	Display GND
11	n.c.
12	n.c.
13	n.c.
14	n.c.
15	Ethernet RX+
16	Ethernet RX-
17	Ethernet TX+
18	Ethernet TX-
19	USB Vcc
20	USB D-
21	USB D+
22	USB GND

Logger Operating Status	
Error (red): Off	No error.
Briefly flashes	Firmware malfunction has occurred, fallback will start shortly.
Blinks (lit/dark: 50/50 )	Firmware running with fallback configuration
Lit (steady)	Firmware malfunction
Log (yellow): Off	No measurement
Blinks 1.3 (lit/dark: 50/50 )	Measurement running, but there is no method storing data
On	Measurement running, data are being stored.
Power (green): Off	Measurement system is off - no power supply connected
Short blinks (lit/dark: 25/75 )	Measurement system is off - power supply is connected (approx. 900µA current draw)
Blinks (lit/dark: 50/50 )	System booting, due to met start condition (e.g., Clamp 15 or WoC)
Lit (steady)	Boot was successful, logger is operational. (Check the Log LED for the measurement mode.)
Long blinks (lit/dark: 75/25 )	System shutting down (takes approx. 10s), ends with the mode "short blinks"

CL15 Status Display	
LED on/off	Voltage for logger start is present/not present (voltage above 3V)

USB	
Standard assignment of PC technology	Transfer of configurations and measured data

Display/Ethernet Pin Assignment (identical to ARCOS dL and X-Over)	
1	Supply -
2	Supply -
3	Supply +
4	Supply +
5	Data 1 (RX+, orange and white)
6	Data 2 (RX-, orange)
7	Data 3 (TX+, green and white)
8	Data 4 (TX-, green)

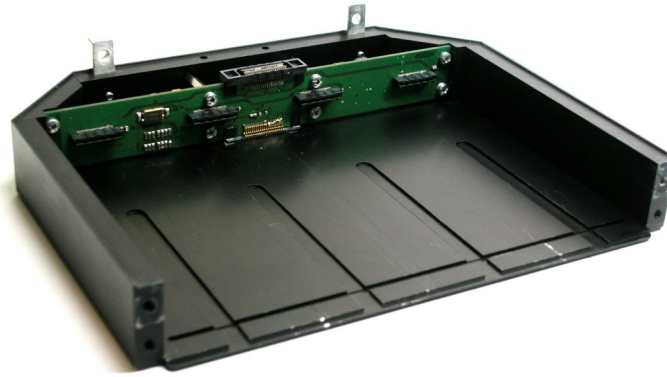
CAN Pin Assignment	
1	CAN 1 Low
2	CAN 2 Low
3	CAN 3 Low
4	CAN 4 Low
5	n.c.
6	CAN 1 High
7	CAN 2 High
8	CAN 3 High
9	CAN 4 High

A CAN bus with “standard” assignment (D-Sub socket with CAN Low Pin 2, CAN High Pin 7 and all other pins unassigned!) can also be directly connected to this plug, without an adapter/breakout box and received on the second CAN channel of the  $\mu$ CROS.

### 3.4 Upper Expansion Chassis Unit (USB)

This expansion chassis unit is mounted on top of the basic platform unit and houses slide-in modules with USB interfaces.

To avoid accidental use as a “lower chassis unit”, its connector has been keyed, with differently positioned alignment pins. In addition, the grooves cut in the lid of the chassis unit (cf. 2.4-1), which serve as assembly guides for the modules, have a different design. The lid serves as the bottom for the next chassis unit. This means that modules for the “upper chassis unit” cannot be inserted in the “lower chassis unit”, and vice versa.



The chassis units have been designed to house up to four slide-in modules per unit. The available space can be divided up between 1/4, 2/4 and 4/4 slide-in modules. The form of the slide-in modules depends on the interface type. The modules currently available for the upper chassis unit are:

- > WLAN
- > GPS
- > UMTS/GPRS
- > Power Switch
- > USB/Power Switch
- > USB hub
- > ETH 1G
- > USB HDD
- > Video
- > CANCase XL (ARCOS PC and ARCOS X-Over only)
- > VN1630 (ARCOS PC and ARCOS X-Over only)

A so-called "backplane" connects the interfaces to the basic unit. The backplane loops both the data signals and the supply through the chassis unit, which makes the chassis units cascadable. Each chassis unit has its own power supply conditioning, its own power supply unit (DC/DC conversion), so to speak. The slide-in modules are identified by their individual IDs, so they function independently of their slot position or the assembly order of the chassis units.

The chassis unit stacks are screwed to the basic platform unit by means of four connecting tabs (two at the front and two at the rear).

In the final expansion chassis unit, the opening above the contacts can be covered with a screw-down lid.

### 3.5 Lower Expansion Chassis Unit (ETH)

The lower expansion chassis unit is mounted under the basic platform unit and houses slide-in modules with Ethernet interfaces. To avoid accidentally substituting it for an "upper chassis unit", its connector has been keyed, with differently positioned alignment pins. In addition, the grooves cut in the lid of the chassis unit, which serve as assembly guides for the interfaces, have a different design. The bottom serves as the lid for the next chassis unit (i.e. on the bottom of the adjoining unit). This means that modules for the "lower chassis unit" cannot be used in the "upper chassis unit", and vice versa.



The chassis unit has been designed to house up to two modules per unit. The available space can be divided up between 1/2 and 2/2 slide-in modules. The currently available interfaces are CAN, LIN, FlexRay, MOST and Analog/Digital I/O.

A modified Ethernet interface in the so-called "backplane" connects the interfaces to the basic unit. The backplane loops both the interface and the supply through the chassis unit, which makes the chassis units cascable. Each chassis unit has its own power supply conditioning, its own power supply unit (DC/DC conversion), so to speak. In the final expansion chassis unit, the opening above the contacts can be covered with a screw-down lid.

The slide-in modules are each identified by an individual ID (product number/serial number), so they function independently of their slot position or the assembly order of the chassis units. When setting up the system, one should thus bear in mind that the node number for the bus channel is linked to the interface module and not the slot position. The assignment of each interface module to a node number is stored in the logger and configurable via the webinterface.



Any time an interface is replaced, please check the configuration of its node assignment (see 2.8.2 Initializing the Modules)!

The total number of chassis units that can be mounted under the basic platform is limited only by such peripheral parameters as the maximum current of the complete assembly, the overall data throughput or the assembly size.

## 3.6 Interface Module for the Lower Expansion Chassis Unit

The lower expansion chassis unit houses the modules with field bus channels. Currently available are slide-in modules for LIN, CAN, FlexRay, MOST and Analog/Digital I/O. Except for Analog/Digital I/O, all module types have quick-start functionality, i.e. signals on one or more channels start and stop the logger.

### 3.6.1 CAN

The models currently available are galvanically separate CAN slide-in modules – with two CAN channels (half-width), and four CAN channels (full-width) – and a not galvanically separate CAN slide-in with four CAN channels (half-width). This means that per expansion chassis unit up to eight CAN channels are available. These modules are equipped with Wake-on-CAN (WoC) and quick-start functionality.

In this context, Wake-on-CAN means that any CAN signal on one or more of the CAN channels can start the logger. You can even specify for the content of a specific message bit to be used for start-up. For details about starting and stopping with WoC, see and the following chapter.

Idle current draw needs to be minimized, but message loss should also be avoided. To optimize idle current draw with respect to these competing demands, two modes were introduced:

- > WoC: very little idle current, but the first messages are lost (max. 30 ms)
- > WoC with “No message lost”: higher idle current, but no messages are lost.

#### 3.6.1.1 Input Buffer

Logger booting time is currently approx. 20 seconds. During this time, any data in the CAN interface is buffered. Once booting procedure is completed, it is transferred to the logger.

The following table shows the buffer size, with the maximum number of buffered messages – for the complete slide-in module, regardless of whether it is a two-channel or four-channel model. A CAN message requires exactly 20 Bytes.

Buffer sizes		
CAN 2/4 Version 1	16MB	838,860 messages
CAN 4 Version 2	16MB	838,860 messages

The buffer time for a particular buffer size depends on the bus load and CAN bit rate. It is easy to compute using the above information (a 500 kbit CAN bus handles about 3600 messages per second).

*If, for example, we have*

- > 50 bus load, 500 kbit rate -> 1800 messages/s and
- > 4 CAN channels

*we can calculate the time  $t$  until the buffer is full:*

$$t = \frac{838860 \text{ messages}}{4 * 1800 \text{ messages/s}} = 116.5 \text{ s}$$

(Note: In a two-channel expansion module, the buffer is distributed on only two nodes, so the buffer available per node is twice as large and, as a result, the maximum buffer time is twice as long.)

The current implementation uses FIFO memory. In the event of buffer overflow, you lose any data arriving while the memory is full.

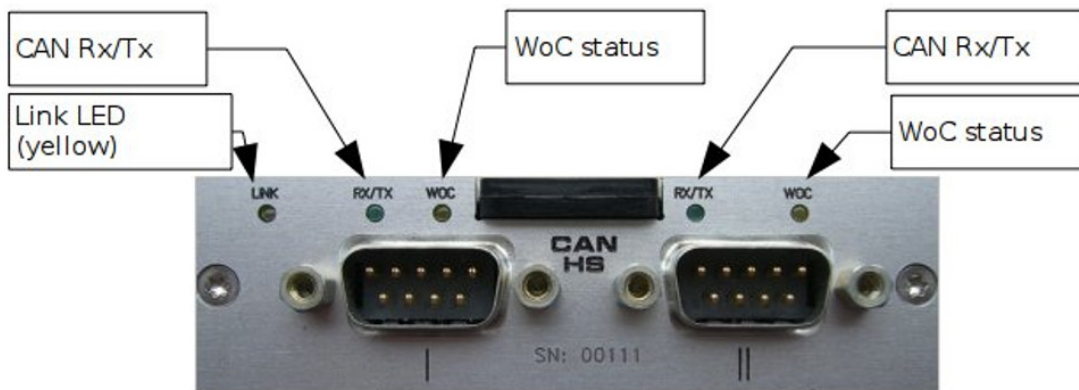
Overflow can occur if booting time is excessively long. This can occur if, at an unfavorable moment, the power supply drops, so the logger shuts off without shutdown. In certain cases, the operating system will then perform a disk consistency check, which can take a minute or longer. The logger will always boot, as the operating system is robust against sudden voltage drops. You must take this possible circumstance into consideration, though, when calculating the length of the booting phase.

### 3.6.1.2 Switching Output for External Hardware (not CAN interface v2)

On the one hand, an external voltage level can start the logger, as well as any external hardware such as CAN measurement modules (thermo, analog, ...). Conversely, CAN signals can start the logger (WoC). In that case, the logger must produce a control signal to initiate the power-up process in the external measurement hardware.

The switching voltage is provided by the logger to the nine-way D-Sub sockets of the CAN channels. The signal consists of a galvanically separate 5 V TTL voltage on pin 1 of the 9-pin D-Sub. The corresponding ground (galvanically separate from the system power supply) is on pin 6.

### 3.6.1.3 Assigning the connections



CAN interface - front plate of two-channel module (four-channel module is comparable)

CAN Interface LEDs	Comment
Link LED (yellow)	Data transfer from the interface to the logger
CAN Rx/Tx (green)	Data transfer on the CAN channel
WoC (yellow)	“Wake on CAN” status (for more on blink codes, see Table 2.6.1-3)

CAN channel LEDs



WoC status LED (blink codes)	Comment
Off	WoC not activated
Short, double	WoC activated and waiting for CAN signal (to wake the logger)
Lit (steady)(basic platform Log LED is off)	WoC with "No Message Loss" activated and waiting for CAN signal (to wake the logger)
Fast blinks	This CAN channel has woken the logger (logger is starting)
Lit (steady)	CAN interface is in operation and measuring

WoC LED Blink codes for the CAN channel

Pin	CAN D-Sub Assignment
1	TTL signal for the switch module (galvanically isolated from supply)
2	CAN Low (galvanically isolated)
3	GND
6	GND for TTL switch voltage (galvanically isolated from power supply)
7	CAN High (galvanically decoupled)

CAN channel D-Sub plug assignment



CAN interface - four-channel module front plate version 2

Pin	CAN LEMO Assignment
1	CAN High (not galvanically decoupled!)
2	CAN Low (not galvanically decoupled!)
3	n.c.

CAN channel LEMO jack assignment

The functions of the LEDs and blink codes are identical to those of the D-Sub version.

### 3.6.1.4 Technical Data

#### Four-channel module:

Dimensions (W x H x D):	168 mm x 25 mm x 119 mm
Weight:	approx. 260 g
Input voltage:	6 - 50 V DC
Operating temperature:	-40 °C to +85 °C
Connectors:	4 x 9-pin D-Sub male
Number of channels:	four channels
Current draw:	in standby, 0.145 mA (12V) in standby with WoC, 19.5 mA (12V) in standby with WoCnml, 106.5 mA (12V) in operation, 150 mA (12V)

#### Two-channel module:

Dimensions (W x H x D):	84 mm x 25 mm x 119 mm
Weight:	approx. 146 g
Input voltage:	6 - 50 V DC
Operating temperature:	-40 °C to +85 °C
Connectors:	2 x 9-pin D-Sub male
Number of channels:	two channels
Current draw:	in standby, 0.155 mA (12V) in standby with WoC, 10.5 mA (12V) in standby with WoCnml, 70.5 mA (12V) in operation, 110 mA (12V)

#### Four-channel module version 2:

Dimensions (W x H x D):	84 mm x 25 mm x 119 mm
Weight:	approx. 150 g
Input voltage:	6 - 50 V DC
Operating temperature:	-40 °C to +85 °C
Connectors:	LEMO EGG.0B.303.CLL mating plug: LEMO FGG.0B.303.CLAD
Number of channels:	four channels
Current draw:	in standby, 0.130 mA (12V) in standby with WoC, 3.1 mA (12V) in standby with WoCnml, 39.6 mA (12V) in operation, 80 mA (12V)

### 3.6.2 LIN

The models currently available are a galvanically separate LIN slide-in module with two LIN channels (half-width), and a not galvanically separate LIN slide-in with four LIN channels (half-width). This means that per expansion chassis unit up to eight LIN channels are available. These modules are equipped with Wake on LIN (WoL) functionality.



In this context, Wake on LIN means that any LIN signal on one or more of the LIN channels can start the logger. You can even specify that the content of a specific message bit be used to trigger start. (For details about starting and stopping with WoL, see Error: Reference source not found, Error: Reference source not found and the following chapter.)

The LIN modules operate in the logger only in “slave mode”.

### 3.6.2.1 Input Buffer

Logger booting time is currently approx. 20 seconds. During this time, any data in the LIN interface is buffered. Once booting procedure is completed, the data is transferred to the logger. Buffer size is 16 MB per slide-in module.

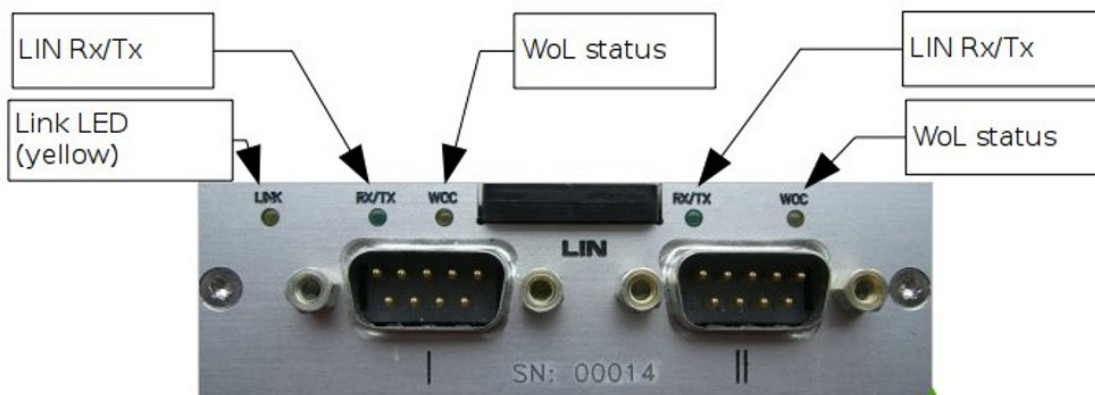
### 3.6.2.2 Power Supply on the LIN Channel

Supply voltage is typically provided through the LIN wiring set. If this voltage is missing or is lower than 10V, there is an auxiliary power supply to the LIN interface. In other words, a LIN connector supplying only signal GND and LIN data can operate at the interface.

### 3.6.2.3 Setting the Baud Rate

The LIN interface has automatic baud rate recognition. This means the baud rate need not be set in the configuration.

### 3.6.2.4 Connection Assignment



LIN interface - two-channel module front plate

LIN Interface LEDs	Comment
Link LED (yellow)	Data transfer from the interface to the logger
LIN Rx/Tx (green)	Data transfer on the LIN channel
WoL (yellow)	“Wake on LIN” status (for more on blink codes, see Table 2.6.2-2)

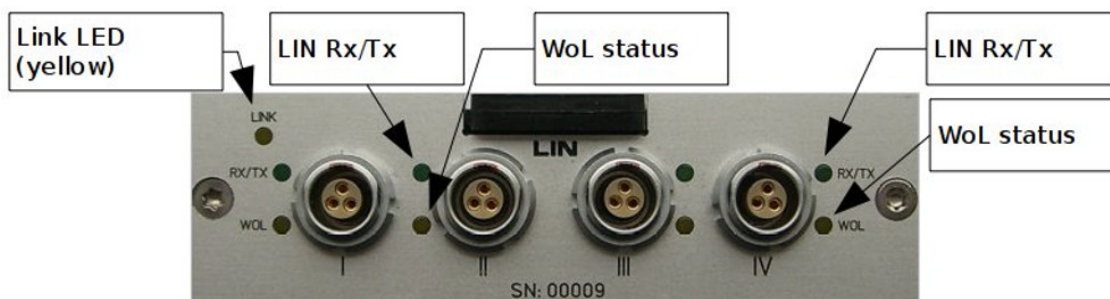
LIN channel LED functions

WoL status LED (blink codes)	Comment
Off	WoL not activated
Short, double	WoL activated and waiting for LIN signal (to wake the logger)
Fast blinks	This LIN channel has woken the logger (logger is starting)
Lit (steady)	LIN interface is in operation and measuring

LIN module WoL LED blink codes

Pin	LIN D-Sub Assignment
1	TTL switch voltage for switch module (galvanically isolated from supply)
3	GND
6	GND for TTL switch voltage (galvanically isolated from power supply)
7	DATA-LIN (galvanically isolated from power supply)
9	VCC-LIN

LIN module D-Sub plug assignment



LIN interface - four-channel module front plate version 2

Pin	LIN LEMO Assignment
1	DATA-LIN (not galvanically decoupled!)
2	n.c.
3	n.c.

LEMO jack assignment for the LIN channel

The functions of the LEDs and blink codes are identical to those of the D-Sub version.

### 3.6.2.5 Technical Data

#### Two-channel module:

Dimensions (W x H x D): 84 mm x 25 mm x 119 mm  
 Weight: approx. 145 g

Changes and errors excepted.

Input voltage:	6 - 50 V DC
Operating temperature:	-40 °C to +85 °C
Connectors:	2 x 9-pin D-Sub male
Number of channels:	two channels
Current draw:	in standby, 0.16 mA (12V) in operation, 120 mA (12V)

#### Four-channel module version 2:

Dimensions (W x H x D):	84 mm x 25 mm x 119 mm
Weight:	approx. 156 g
Input voltage:	6 - 50 V DC
Operating temperature:	-40 °C to +85 °C
Connectors:	LEMO EGG.0B.303.CLL mating plug: LEMO FGG.0B.303.CLAD
Number of channels:	four channels
Current draw:	in standby, 0.20 mA (12V) in standby with 4 x WoL, 50 mA (12V) in operation, 115 mA (12V)

### 3.6.3 FlexRay

Currently available are half-width FlexRay slide-in modules with two FlexRay channels, each configurable as an A or B channel. This means that per expansion chassis unit up to four FlexRay channels are available. These modules are equipped with Wake-on-FlexRay functionality (WoFR).

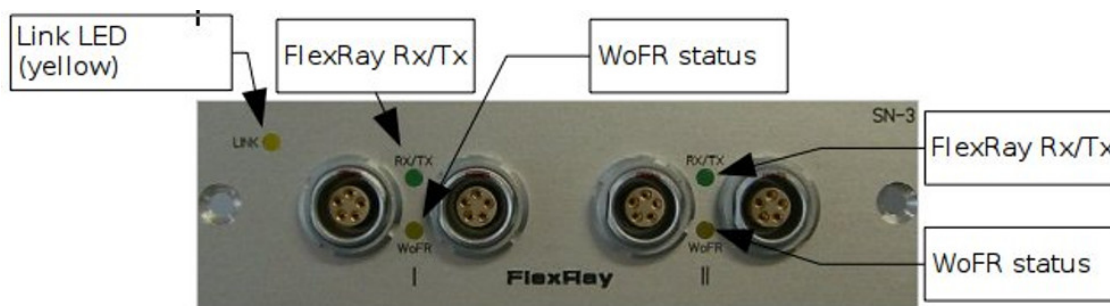
#### 3.6.3.1 Input Buffer

Logger booting time is currently approx. 20 seconds. During this time, any data in the FlexRay interface is buffered. Once booting procedure is completed, the data is transferred to the logger. Buffer size per slide-in module is 128 MB.

#### 3.6.3.2 Timing Parameters

Due to the method used for signal acquisition on the bus, the FlexRay interface needs no additional parameters to define the exact timing on the bus. Therefore, they require no specification in the configuration, providing messages or signals need only be received and not sent (see also Chapter Error: Reference source not found Error: Reference source not found).

#### 3.6.3.3 Connection Assignment



FlexRay interface - two channel module front plate

LED FlexRay Interface	Comment
Link LED (yellow)	Data transfer from the interface to the logger
FlexRay Rx/Tx (green)	Data transfer on the FlexRay channel
WoFR (yellow)	“Wake on FlexRay” status (for more on blink codes, see Table 2.6.2-2)

Flexray module LED functions

WoFR status LED (blink codes)	Comment
Off	No WoFR activated
Short, double	WoFR activated and waiting for FlexRay signal (to wake the logger)
Fast blinks	This FlexRay channel has woken the logger (logger is starting)
Lit (steady)	FlexRay interface is in operation and measuring

FlexRay module WoFR LED blink codes

Pin	FlexRay LEMO Assignment
1	Reserve (shield)
2	BP in
3	BM in
4	BP out
5	BM out

FlexRay module LEMO jack assignment

Per FlexRay channel there are two five-pin Lemo sockets, whose PINs are connected straight through. This offers a variety of connection possibilities, from a simple branch line to looping right into the bus.

### 3.6.3.4 Technical Data

Size (W x H x D): 84 mm x 25 mm x 119 mm  
 Weight: approx. 165 g

Changes and errors excepted.

Input voltage:	6 - 50 V DC
Operating temperature:	-40 °C to +85 °C
Connectors:	LEMO EGG.0B.305.CLN mating plug: LEMO FGG.0B.305.CLAD
Number of channels:	two (either A or B channel possible)
Transceivers:	2 x TJA 1080
Controllers:	2 x Fujitsu MB88121B
Memory:	128 MB
Current draw:	in standby, 0.26 mA (12V) in standby with WoFR, 5 mA (12V) in operation, 340 mA (12V)

### 3.6.4 Analog/Digital I/O

The currently available unit is a (half-width) Analog/Digital I/O (ADIO) slide-in module with eight analog channels, eight digital bit inputs and eight digital bit outputs.

#### 3.6.4.1 Input Buffer

Logger booting time is currently approx. 20 seconds. During this time, any data in the ADIO interface is buffered. Once booting procedure is completed, the data is transferred to the logger. During booting, digital outputs are not addressed and remain in a defined start state (no bit set). Once the logger firmware is fully started and backlog execution has been started, the digital outputs are also activated.

Buffer size is 8 MB per slide-in module.

#### 3.6.4.2 Sampling Rate

Each channel of the ADIO interface can be independently parameterized with a sampling rate in the range of 0.1...1000.0 Hz. The digital outputs are activated when needed, i.e. upon value change, with a maximum rate of 100 Hz (for more information, see Chapters Error: Reference source not found, Error: Reference source not found and Error: Reference source not found).

#### 3.6.4.3 Connection Assignment



## ADIO interface - front plate

LET ADIO Interface	Comment
Link LED (yellow)	Data transfer from the interface to the logger

Pin	ADIO LEMO Analog Input Assignment
1	Analog input channel 1 + or 5 +
2	Analog input channel 1 - or 5 -
3	Analog input channel 2 + or 6 +
4	Analog input channel 2 - or 6 -
5	Analog input channel 3 + or 7 +
6	Analog input channel 3 - or 7 -
7	Analog input channel 4 + or 8 +
8	Analog input channel 4 - or 8 -
9	GND
10	GND

## LEMO jack assignment at the analog input

For analog input, the ADIO provides two jacks, for the channels 1 - 4 and channels 5 - 8 respectively. Though these are differential inputs, it may be useful to connect the ADIO grounds (GND) with the ground of the voltage source to be measured, so as to create a common reference potential.

Pin	ADIO LEMO Digital Input Assignment
1	Digital input bit 1
2	Digital input bit 2
3	Digital input bit 3
4	Digital input bit 4
5	Digital input bit 5
6	Digital input bit 6
7	Digital input bit 7
8	Digital input bit 8
9	Digital GND
10	Digital GND

## LEMO jack assignment for digital input

Always connect the digital inputs to a fixed reference ground. The switching threshold between high (=1) and low (=0) level is approx. 1.5 V DC.

Pin	ADIO LEMO Digital Output Assignment
1	Digital output bit 1
2	Digital output bit 2
3	Digital output bit 3
4	Digital output bit 4
5	Digital output bit 5
6	Digital output bit 6
7	Digital output bit 7
8	Digital output bit 8
9	Digital GND
10	Digital GND

LEMO jack assignment for digital output

The digital outputs function as switches, for connecting the particular connection pin to ground. If a logical 0 is applied, they are open; they are closed if a logical 1 is applied. In combination with the appropriate external switches, it is thus possible to control external devices (e.g., relays, LEDs).

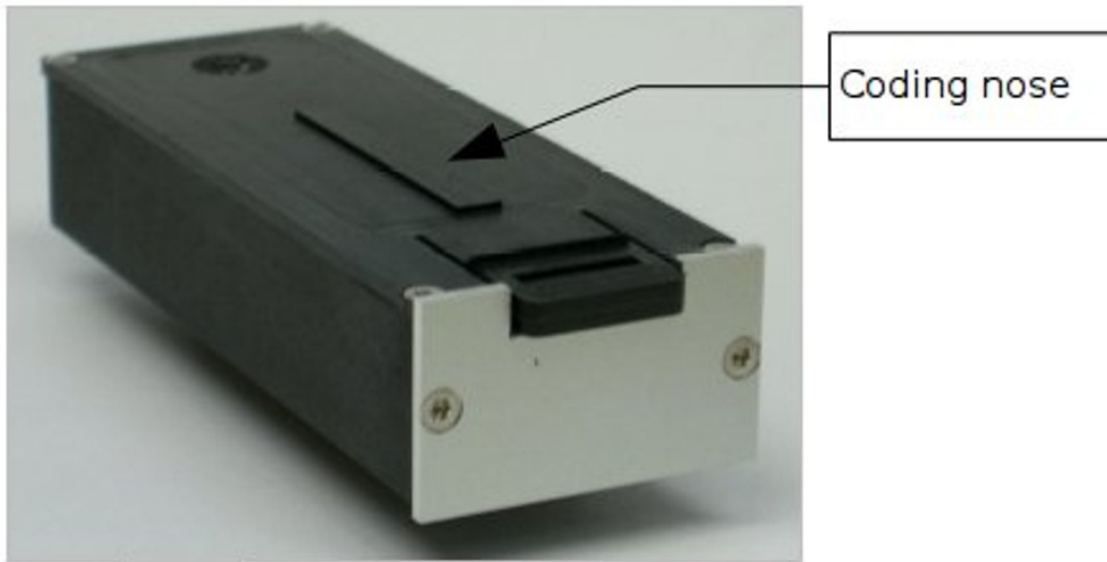
#### 3.6.4.4 Technical Data

Dimensions (W x H x D):	84 mm x 25 mm x 119 mm
Weight:	approx. 180 g
Input voltage:	6 - 50 V DC
Operating temperature:	-40 °C to +85 °C
Connectors:	Analog input: LEMO EEB.2B.310.CLN mating plug: LEMO FGB.2B.310.CLAD Digital input: LEMO EEG.2B.310.CLN mating plug: LEMO FGG.2B.310.CLAD Digital output: LEMO EEA.2B.310.CLN mating plug: LEMO FGA.2B.310.CLAD
Number of channels:	2 x four analog channels 8 digital bit inputs 8 digital bit outputs
Measuring range:	Analog input: $\pm 60$ V, 16-bit resolution (1.8 mV) Digital input: 0...60 V switching threshold at approx. 1.5 V input resistance: 1 MOhm Digital output switching voltage: 0...60 V Max. switching current: 250 mA
Memory:	8 MB
Current draw:	in standby, 0.13 mA (12V) in operation, 68 mA (12V)



## 3.7 Interface Modules for the Upper Expansion Chassis Unit

All the modules for the upper expansion chassis unit have a coding nose. This ensures that they are operated only in an upper expansion unit.



USB dummy module for the upper expansion chassis unit

### 3.7.1 WiFi

#### 3.7.1.1 Applications

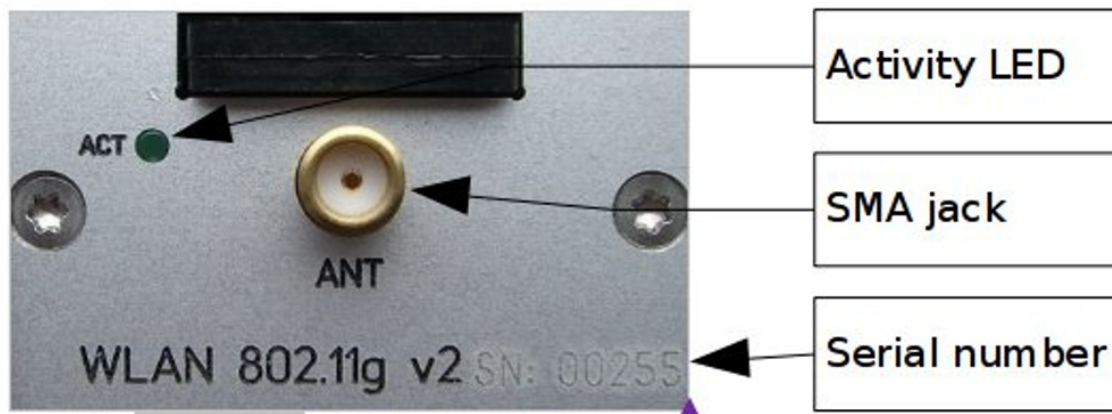
The WiFi module is another channel (like a memory stick, for example) for transmitting recorded data from the logger to an external computer, such as a server station, or for importing new configurations. Data transfer that is started when the logger is shut down, for example, can be directed to the WiFi module.

The module is only activated when data transfer is started, and is inactive during measurement operation. This increases data security – a WiFi that is inactive cannot be misused for unauthorized data transmission. Moreover, this reduces total current draw of the measurement system.

Conversely, it isn't possible to use the logger's WiFi port for continuous measurement-data transfer or for data display during measurement.

#### 3.7.1.2 Module Placement





V2 WiFi module

The module is a quarter-unit wide and can be operated in any slot within a given chassis unit. Neither its position within a chassis unit nor the chassis level it occupies requires specification during configuration.

### 3.7.1.3 Configuration parameters

Each module has a fixed MAC address. During module activation, the IP address is either, depending on its configuration, dynamically assigned (DHCP server) or set from the logger.



V2 WiFi module bottom

The MAC address is very often used by system administrators to protect networks against unauthorized use. When replacing modules, make sure the used module has authorized access to the required networks. For this purpose, the sticker on the bottom of the module has the MAC address printed under the product number (Illus. above).

In the case of the  $\mu$ CROS, you can obtain the MAC address from CAETEC (please specify the  $\mu$ CROS serial number). Due to lack of space, there is no sticker, so the address must be read by software from the module.

In the case of built-in modules, you can list MAC addresses according to the respective serial numbers printed on the front of the module.

With firmware Version 1.3 and higher, the option of transferring data via WLAN, DT\_WLAN, must be enabled in the logger. Without activation no data transfer is possible.

#### 3.7.1.4 Technical Data

Dimensions (W x H x D)	ARCOS: 42mm x 25mm x 114mm μCROS: 25mm x 8mm x 80mm
Weight	ARCOS: approx. 63 g μCROS: approx. 14 g
Operating temperature	-40°C bis +85°C
Input voltage	ARCOS: 6 50 V DC μCROS: 6 36 V DC
WLAN	Standard 802.11g/n, 2.4 Ghz Maximale Rate 54 Mbit/s Sicherheitseinstellungen nach WPA-PSK und WPA-EAP Zugriff nur über Infrastruktur (Access-Point), Adhoc-Verbindung nicht möglich für ARCOS und μCROS möglich
Current draw	approx. 40 mA