

Handbuch | DE

TX1200

TwinCAT 2 | PLC-Bibliothek: TcDALIV2



Inhaltsverzeichnis

1	Vorwort.....	11
1.1	Hinweise zur Dokumentation	11
1.2	Sicherheitshinweise	12
1.3	Hinweise zur Informationssicherheit	13
2	Zielgruppen.....	14
3	DALI.....	15
4	Integration in TwinCAT	18
4.1	KL6811 Integration in TwinCAT (CX9020).....	18
4.2	KL6811 Integration in TwinCAT (BC9191).....	20
4.3	KL6821 Integration in TwinCAT (CX9020).....	24
4.4	KL6821 Integration in TwinCAT (BC9191).....	28
5	Programmierung	32
5.1	Bausteine	32
5.1.1	Notbeleuchtung - Funktionsbausteine.....	32
5.1.2	Farb-/Farbtemperatursteuerung - Funktionsbausteine	34
5.1.3	Entladungslampen - Funktionsbausteine	36
5.1.4	LED-Module - Funktionsbausteine.....	36
5.1.5	Push button - Funktionsbausteine.....	37
5.1.6	Präsenzmelder - Funktionsbausteine.....	38
5.1.7	Interior Automation - Funktionsbausteine	38
5.1.8	Helligkeitssensoren - Funktionsbausteine.....	39
5.1.9	B.E.G. - Funktionsbausteine	39
5.1.10	Osram - Funktionsbausteine	39
5.1.11	Philips - Funktionsbausteine	40
5.1.12	Steinel - Funktionsbausteine.....	41
5.1.13	Tridonic - Funktionsbausteine	42
5.1.14	Theben HTS - Funktionsbausteine	42
5.1.15	FB_DALIV2AddressingIntRandomAddressing.....	42
5.1.16	FB_DALIV2AddressingPhysicalSelection	44
5.1.17	FB_DALIV2AddressingRandomAddressing.....	45
5.1.18	FB_DALIV2ChangeAddressList.....	47
5.1.19	FB_DALIV2SwapShortAddress	49
5.1.20	FB_DALIV2SwapShortAddressList.....	49
5.1.21	FB_DALIV2xAddressingIntRandomAddressing.....	51
5.1.22	FB_DALIV2xChangeAddressList	52
5.1.23	FB_DALIV2ConstantLightControlEco	54
5.1.24	FB_DALIV2Dimmer1Switch	56
5.1.25	FB_DALIV2Dimmer1SwitchEco.....	59
5.1.26	FB_DALIV2Dimmer1SwitchMultiple.....	61
5.1.27	FB_DALIV2Dimmer2Switch	63
5.1.28	FB_DALIV2Dimmer2SwitchEco.....	65
5.1.29	FB_DALIV2Light	67
5.1.30	FB_DALIV2LightControl.....	69

5.1.31	FB_DALIV2Ramp.....	71
5.1.32	FB_DALIV2Sequencer.....	74
5.1.33	FB_DALIV2StairwellDimmer.....	78
5.1.34	FB_DALIV2GetSettings.....	80
5.1.35	FB_DALIV2GetSettingsSingleDevice.....	82
5.1.36	FB_DALIV2SetSettings.....	84
5.1.37	FB_DALIV2EmergencyLightingDT.....	86
5.1.38	FB_DALIV2EmergencyLightingFT.....	88
5.1.39	FB_DALIV2FileLogging.....	90
5.1.40	FB_DALIV2GetSettingsType01.....	91
5.1.41	FB_DALIV2SetSettingsType01.....	94
5.1.42	FB_DALIV2Communication.....	95
5.1.43	FB_DALIV2GetEventData.....	97
5.1.44	FB_DALIV2SendDALICommand.....	98
5.1.45	FB_DALIV2xSendDALICommand.....	100
5.1.46	FB_KL6811Config.....	102
5.1.47	FB_KL6821Communication.....	104
5.1.48	FB_KL6821Config.....	106
5.1.49	FB_DSIDirectArcPowerControl.....	109
5.1.50	FB_DALIV2AddToGroup.....	110
5.1.51	FB_DALIV2EnableWriteMemory.....	110
5.1.52	FB_DALIV2RemoveFromGroup.....	111
5.1.53	FB_DALIV2RemoveFromScene.....	112
5.1.54	FB_DALIV2Reset.....	113
5.1.55	FB_DALIV2SetShortAddress.....	114
5.1.56	FB_DALIV2StoreActualLevelInDTR.....	115
5.1.57	FB_DALIV2StoreDTRAsFadeRate.....	116
5.1.58	FB_DALIV2StoreDTRAsFadeTime.....	116
5.1.59	FB_DALIV2StoreDTRAsMaxLevel.....	117
5.1.60	FB_DALIV2StoreDTRAsMinLevel.....	118
5.1.61	FB_DALIV2StoreDTRAsPowerOnLevel.....	119
5.1.62	FB_DALIV2StoreDTRAsScene.....	120
5.1.63	FB_DALIV2StoreDTRAsShortAddress.....	120
5.1.64	FB_DALIV2StoreDTRAsSystemFailureLevel.....	121
5.1.65	FB_DALIV2DirectArcPowerControl.....	122
5.1.66	FB_DALIV2Down.....	123
5.1.67	FB_DALIV2EnableDAPCSequence.....	124
5.1.68	FB_DALIV2GoToScene.....	125
5.1.69	FB_DALIV2Off.....	126
5.1.70	FB_DALIV2OnAndStepUp.....	126
5.1.71	FB_DALIV2RecallMaxLevel.....	127
5.1.72	FB_DALIV2RecallMinLevel.....	128
5.1.73	FB_DALIV2StepDown.....	129
5.1.74	FB_DALIV2StepDownAndOff.....	130
5.1.75	FB_DALIV2StepUp.....	130
5.1.76	FB_DALIV2Up.....	131

5.1.77	FB_DALIV2QueryActualLevel.....	132
5.1.78	FB_DALIV2QueryBallast.....	133
5.1.79	FB_DALIV2QueryContentDTR	134
5.1.80	FB_DALIV2QueryContentDTR1	135
5.1.81	FB_DALIV2QueryContentDTR2	136
5.1.82	FB_DALIV2QueryDeviceType	137
5.1.83	FB_DALIV2QueryFadeTimeFadeRate	138
5.1.84	FB_DALIV2QueryGroups.....	139
5.1.85	FB_DALIV2QueryGroups0UpTo7.....	139
5.1.86	FB_DALIV2QueryGroups8UpTo15.....	140
5.1.87	FB_DALIV2QueryLampFailure	141
5.1.88	FB_DALIV2QueryLampPowerOn	142
5.1.89	FB_DALIV2QueryLimitError.....	143
5.1.90	FB_DALIV2QueryMaxLevel.....	144
5.1.91	FB_DALIV2QueryMinLevel.....	145
5.1.92	FB_DALIV2QueryMissingShortAddress	145
5.1.93	FB_DALIV2QueryPhysicalMinLevel.....	146
5.1.94	FB_DALIV2QueryPowerFailure	147
5.1.95	FB_DALIV2QueryPowerOnLevel.....	148
5.1.96	FB_DALIV2QueryRandomAddress.....	149
5.1.97	FB_DALIV2QueryRandomAddressH.....	150
5.1.98	FB_DALIV2QueryRandomAddressL.....	151
5.1.99	FB_DALIV2QueryRandomAddressM.....	151
5.1.100	FB_DALIV2QueryResetState.....	152
5.1.101	FB_DALIV2QuerySceneLevel.....	153
5.1.102	FB_DALIV2QueryStatus	154
5.1.103	FB_DALIV2QuerySystemFailureLevel.....	155
5.1.104	FB_DALIV2QueryVersionNumber	156
5.1.105	FB_DALIV2ReadMemoryLocation	157
5.1.106	FB_DALIV2Compare	158
5.1.107	FB_DALIV2Initialise	159
5.1.108	FB_DALIV2PhysicalSelection	160
5.1.109	FB_DALIV2ProgramShortAddress.....	160
5.1.110	FB_DALIV2QueryShortAddress.....	161
5.1.111	FB_DALIV2Randomise	162
5.1.112	FB_DALIV2SearchAddr	163
5.1.113	FB_DALIV2SearchAddrH.....	163
5.1.114	FB_DALIV2SearchAddrL	164
5.1.115	FB_DALIV2SearchAddrM	165
5.1.116	FB_DALIV2SetDTR	166
5.1.117	FB_DALIV2SetDTR1	166
5.1.118	FB_DALIV2SetDTR2	167
5.1.119	FB_DALIV2Terminate	168
5.1.120	FB_DALIV2VerifyShortAddress	169
5.1.121	FB_DALIV2Withdraw	169
5.1.122	FB_DALIV2WriteMemoryLocation	170

5.1.123	FB_DALIV2xAddToDeviceGroups	171
5.1.124	FB_DALIV2xDisableInstance	172
5.1.125	FB_DALIV2xDisablePowerCycleNotification	173
5.1.126	FB_DALIV2xEnableInstance	174
5.1.127	FB_DALIV2xEnablePowerCycleNotification	175
5.1.128	FB_DALIV2xIdentifyDevice	176
5.1.129	FB_DALIV2xRemoveFromDeviceGroups	177
5.1.130	FB_DALIV2xReset	178
5.1.131	FB_DALIV2xSetEventFilter	179
5.1.132	FB_DALIV2xSetEventScheme	180
5.1.133	FB_DALIV2xSetOperatingMode	181
5.1.134	FB_DALIV2xSetShortAddress	182
5.1.135	FB_DALIV2xStartQuiescentMode	183
5.1.136	FB_DALIV2xStopQuiescentMode	184
5.1.137	FB_DALIV2xQueryContentDTR0	185
5.1.138	FB_DALIV2xQueryContentDTR1	186
5.1.139	FB_DALIV2xQueryContentDTR2	187
5.1.140	FB_DALIV2xQueryDeviceGroups	188
5.1.141	FB_DALIV2xQueryDeviceStatus	189
5.1.142	FB_DALIV2xQueryEventFilter	190
5.1.143	FB_DALIV2xQueryEventScheme	191
5.1.144	FB_DALIV2xQueryInputDeviceError	193
5.1.145	FB_DALIV2xQueryInputValue	194
5.1.146	FB_DALIV2xQueryInputValueLatch	195
5.1.147	FB_DALIV2xQueryInstanceEnabled	196
5.1.148	FB_DALIV2xQueryInstanceError	197
5.1.149	FB_DALIV2xQueryInstanceStatus	198
5.1.150	FB_DALIV2xQueryMissingShortAddress	199
5.1.151	FB_DALIV2xQueryNumberOfInstances	200
5.1.152	FB_DALIV2xQueryOperatingMode	201
5.1.153	FB_DALIV2xQueryPowerCycleNotification	202
5.1.154	FB_DALIV2xQueryRandomAddressH	203
5.1.155	FB_DALIV2xQueryRandomAddressL	204
5.1.156	FB_DALIV2xQueryRandomAddressM	205
5.1.157	FB_DALIV2xQueryResetState	206
5.1.158	FB_DALIV2xQueryResolution	207
5.1.159	FB_DALIV2xQueryVersionNumber	208
5.1.160	FB_DALIV2xReadMemoryLocation	209
5.1.161	FB_DALIV2xCompare	210
5.1.162	FB_DALIV2xDTR0	211
5.1.163	FB_DALIV2xDTR1	212
5.1.164	FB_DALIV2xDTR2	213
5.1.165	FB_DALIV2xInitialise	214
5.1.166	FB_DALIV2xProgramShortAddress	215
5.1.167	FB_DALIV2xQueryShortAddress	216
5.1.168	FB_DALIV2xRandomise	217

5.1.169	FB_DALIV2xSearchAddrH.....	218
5.1.170	FB_DALIV2xSearchAddrL	219
5.1.171	FB_DALIV2xSearchAddrM	220
5.1.172	FB_DALIV2xTerminate	221
5.1.173	FB_DALIV2xVerifyShortAddress	221
5.1.174	FB_DALIV2xWithdraw	222
5.1.175	FB_DALIV2xWriteMemoryLocation	223
5.1.176	FB_DALIV2Inhibit.....	224
5.1.177	FB_DALIV2QueryBatteryCharge	225
5.1.178	FB_DALIV2QueryDurationTestResult.....	226
5.1.179	FB_DALIV2QueryEmergencyLevel.....	227
5.1.180	FB_DALIV2QueryEmergencyMaxLevel.....	228
5.1.181	FB_DALIV2QueryEmergencyMinLevel.....	229
5.1.182	FB_DALIV2QueryEmergencyMode	230
5.1.183	FB_DALIV2QueryEmergencyStatus	232
5.1.184	FB_DALIV2QueryFailureStatus	233
5.1.185	FB_DALIV2QueryFeatures	234
5.1.186	FB_DALIV2QueryLampEmergencyTime	235
5.1.187	FB_DALIV2QueryLampTotalOperationTime.....	236
5.1.188	FB_DALIV2QueryRatedDuration	237
5.1.189	FB_DALIV2QueryTestTiming.....	238
5.1.190	FB_DALIV2ReLightResetInhibit.....	240
5.1.191	FB_DALIV2ResetDurationTestDoneFlag.....	241
5.1.192	FB_DALIV2ResetFunctionTestDoneFlag	242
5.1.193	FB_DALIV2ResetLampTime.....	243
5.1.194	FB_DALIV2Rest.....	244
5.1.195	FB_DALIV2StartDurationTest	245
5.1.196	FB_DALIV2StartFunctionTest.....	246
5.1.197	FB_DALIV2StopTest.....	247
5.1.198	FB_DALIV2StoreDTRAsDurationTestInterval.....	248
5.1.199	FB_DALIV2StoreDTRAsEmergencyLevel	249
5.1.200	FB_DALIV2StoreDTRAsFunctionTestInterval	250
5.1.201	FB_DALIV2StoreDTRAsProlongTime.....	251
5.1.202	FB_DALIV2StoreDTRAsTestDelayTimeHighByte	252
5.1.203	FB_DALIV2StoreDTRAsTestDelayTimeLowByte	253
5.1.204	FB_DALIV2StoreDTRAsTestExecutionTimeout	254
5.1.205	FB_DALIV2QueryActualHIDFailure	255
5.1.206	FB_DALIV2QueryHIDFeatures	256
5.1.207	FB_DALIV2QueryHIDStatus	257
5.1.208	FB_DALIV2QueryStoredHIDFailure.....	258
5.1.209	FB_DALIV2QueryThermalLoad	259
5.1.210	FB_DALIV2QueryThermalOverloadTime.....	260
5.1.211	FB_DALIV2QueryThermalOverloadTimeHB.....	261
5.1.212	FB_DALIV2QueryThermalOverloadTimeLB	262
5.1.213	FB_DALIV2ResetStoredHIDFailure	263
5.1.214	FB_DALIV2DisableCurrentProtector.....	264

5.1.215	FB_DALIV2EnableCurrentProtector	265
5.1.216	FB_DALIV2QueryCurrentProtectorActive	266
5.1.217	FB_DALIV2QueryCurrentProtectorEnabled.....	267
5.1.218	FB_DALIV2QueryDimmingCurve.....	268
5.1.219	FB_DALIV2QueryFastFadeTime	269
5.1.220	FB_DALIV2QueryGearType	270
5.1.221	FB_DALIV2QueryLedFailureStatus	271
5.1.222	FB_DALIV2QueryLedFeatures	272
5.1.223	FB_DALIV2QueryLoadDecrease	273
5.1.224	FB_DALIV2QueryLoadIncrease.....	274
5.1.225	FB_DALIV2QueryMinFastFadeTime	275
5.1.226	FB_DALIV2QueryOpenCircuit	277
5.1.227	FB_DALIV2QueryOperatingMode.....	278
5.1.228	FB_DALIV2QueryPossibleOperatingModes	279
5.1.229	FB_DALIV2QueryReferenceMeasurementFailed	280
5.1.230	FB_DALIV2QueryReferenceRunning	281
5.1.231	FB_DALIV2QueryShortCircuit.....	282
5.1.232	FB_DALIV2QueryThermalOverload.....	283
5.1.233	FB_DALIV2QueryThermalShutDown.....	284
5.1.234	FB_DALIV2ReferenceSystemPower	285
5.1.235	FB_DALIV2SelectDimmingCurve	286
5.1.236	FB_DALIV2SetFastFadeTime.....	287
5.1.237	FB_DALIV2Activate	288
5.1.238	FB_DALIV2AssignColourToLinkedChannel.....	289
5.1.239	FB_DALIV2ColourTemperatureTcStepCooler	291
5.1.240	FB_DALIV2ColourTemperatureTcStepWarmer	292
5.1.241	FB_DALIV2CopyReportToTemporary	293
5.1.242	FB_DALIV2QueryAssignedColour	294
5.1.243	FB_DALIV2QueryColourStatus.....	296
5.1.244	FB_DALIV2QueryColourTypeFeatures.....	297
5.1.245	FB_DALIV2QueryColourValue.....	298
5.1.246	FB_DALIV2QueryGearFeaturesStatus	301
5.1.247	FB_DALIV2QueryRGBWAFControl	303
5.1.248	FB_DALIV2SetTemporaryColourTemperatureTc	304
5.1.249	FB_DALIV2SetTemporaryPrimaryNDimlevel.....	306
5.1.250	FB_DALIV2SetTemporaryRGBDimlevel.....	307
5.1.251	FB_DALIV2SetTemporaryRGBWAFControl	308
5.1.252	FB_DALIV2SetTemporaryWAFDimlevel.....	310
5.1.253	FB_DALIV2SetTemporaryXCoordinate	311
5.1.254	FB_DALIV2SetTemporaryYCoordinate	312
5.1.255	FB_DALIV2StartAutoCalibration	314
5.1.256	FB_DALIV2StoreColourTemperatureTcLimit.....	315
5.1.257	FB_DALIV2StoreGearFeaturesStatus	317
5.1.258	FB_DALIV2StoreTYPrimaryN	318
5.1.259	FB_DALIV2StoreXyCoordinatePrimaryN.....	320
5.1.260	FB_DALIV2XCoordinateStepDown.....	321

5.1.261	FB_DALIV2XCoordinateStepUp	322
5.1.262	FB_DALIV2YCoordinateStepDown.....	323
5.1.263	FB_DALIV2YCoordinateStepUp	324
5.1.264	KELVIN_TO_MIREK	326
5.1.265	MIREK_TO_KELVIN	326
5.1.266	FB_DALIV2x301QueryDoubleTimer	327
5.1.267	FB_DALIV2x301QueryDoubleTimerMin	328
5.1.268	FB_DALIV2x301QueryRepeatTimer	329
5.1.269	FB_DALIV2x301QueryShortTimer.....	330
5.1.270	FB_DALIV2x301QueryShortTimerMin	331
5.1.271	FB_DALIV2x301QueryStuckTimer	332
5.1.272	FB_DALIV2x301SetDoubleTimer	333
5.1.273	FB_DALIV2x301SetRepeatTimer	335
5.1.274	FB_DALIV2x301SetShortTimer	336
5.1.275	FB_DALIV2x301SetStuckTimer	337
5.1.276	FB_DALIV2x303CancelHoldTimer.....	338
5.1.277	FB_DALIV2x303CatchMovement	339
5.1.278	FB_DALIV2x303QueryCatching	340
5.1.279	FB_DALIV2x303QueryDeadtimeTimer	341
5.1.280	FB_DALIV2x303QueryHoldTimer	342
5.1.281	FB_DALIV2x303QueryReportTimer.....	343
5.1.282	FB_DALIV2x303SetDeadtimeTimer	344
5.1.283	FB_DALIV2x303SetHoldTimer	345
5.1.284	FB_DALIV2x303SetReportTimer	347
5.1.285	FB_DALIV2x304QueryDeadtimeTimer	348
5.1.286	FB_DALIV2x304QueryHysteresis.....	349
5.1.287	FB_DALIV2x304QueryHysteresisMin	350
5.1.288	FB_DALIV2x304QueryReportTimer.....	351
5.1.289	FB_DALIV2x304SetDeadtimeTimer	352
5.1.290	FB_DALIV2x304SetHysteresis	353
5.1.291	FB_DALIV2x304SetHysteresisMin	354
5.1.292	FB_DALIV2x304SetReportTimer	355
5.1.293	FB_DALIV2EnableDeviceType	356
5.1.294	FB_DALIV2QueryExtendedVersionNumber	357
5.1.295	FB_DALIV2IAPIR.....	358
5.1.296	FB_DALIV2xBEGLuxomat.....	360
5.1.297	FB_DALIV2xOsramProfPushbuttonCoupler	361
5.1.298	FB_DALIV2xOsramProfSensorCoupler	363
5.1.299	FB_DALIV2xSteinelLiveLinkMotionSensor	364
5.1.300	FB_DALIV2SmartSPOT	366
5.1.301	FB_DALIV2xThebenPlanoSpot360.....	367
5.2	Variablen	369
5.2.1	Vorschaltgeräte - Variablen.....	369
5.2.2	Notbeleuchtung - Variablen.....	373
5.2.3	Entladungslampen - Variablen	378
5.2.4	Farb-/Farbtemperatursteuerung - Variablen	380

5.2.5	Philips - Variablen	386
5.3	Datentypen	387
5.3.1	E_DALIV2AddrType	387
5.3.2	E_DALIV2CommandPriority	387
5.3.3	E_DALIV2ConfigurationCommands	387
5.3.4	E_DALIV2CurrentAddressingState	388
5.3.5	E_DALIV2DataFrameType	388
5.3.6	E_DALIV2DimmingCurve	388
5.3.7	E_DALIV2EventScheme	388
5.3.8	E_DALIV2InstAddrType	388
5.3.9	E_DALIV2OperationMode	389
5.3.10	E_DALIV2PowerSupplyMode	389
5.3.11	ST_DALIV2ChangeAddressList	389
5.3.12	ST_DALIV2ControlTable	389
5.3.13	ST_DALIV2DeviceSettings	389
5.3.14	ST_DALIV2DeviceSettingsType01	390
5.3.15	ST_DALIV2FileLogging	390
5.3.16	ST_DALIV2InData	390
5.3.17	ST_DALIV2OutData	391
5.3.18	ST_DALIV2SequenceTable	391
5.3.19	ST_DALIV2SwapShortAddressList	391
5.3.20	ST_KL6821InData	391
5.3.21	ST_KL6821OutData	391
5.4	Fehlercodes	391
6	Support und Service	395

1 Vorwort

1.1 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, für jede Installation und Inbetriebnahme die zu dem betreffenden Zeitpunkt veröffentlichte Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiter entwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Marken

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.

EtherCAT®

EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizenziert durch die Beckhoff Automation GmbH, Deutschland

Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zu widerhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

1.2 Sicherheitshinweise

Sicherheitsbestimmungen

Beachten Sie die folgenden Sicherheitshinweise und Erklärungen!
Produktspezifische Sicherheitshinweise finden Sie auf den folgenden Seiten oder in den Bereichen Montage, Verdrahtung, Inbetriebnahme usw.

Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

Erklärung der Symbole

In der vorliegenden Dokumentation werden die folgenden Symbole mit einem nebenstehenden Sicherheitshinweis oder Hinweistext verwendet. Die Sicherheitshinweise sind aufmerksam zu lesen und unbedingt zu befolgen!

GEFAHR

Akute Verletzungsgefahr!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht unmittelbare Gefahr für Leben und Gesundheit von Personen!

WARNUNG

Verletzungsgefahr!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht Gefahr für Leben und Gesundheit von Personen!

VORSICHT

Schädigung von Personen!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, können Personen geschädigt werden!

HINWEIS

Schädigung von Umwelt oder Geräten

Wenn der Hinweis neben diesem Symbol nicht beachtet wird, können Umwelt oder Geräte geschädigt werden.



Tipp oder Fingerzeig

Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

1.3 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

2 Zielgruppen

Für den Nutzer dieser Bibliothek werden folgende Grundkenntnisse vorausgesetzt:

- TwinCAT PLC Control
- TwinCAT System Manager
- PC und Netzwerkkennnisse
- Aufbau und Eigenschaften der Beckhoff Embedded-PC und deren Busklemmensystem
- Technologie von DALI Geräten
- Einschlägige Sicherheitsvorschriften der technischen Gebäudeausrüstung

Diese Softwarebibliothek ist für Gebäudeautomation-Systempartner der Beckhoff Automation GmbH & Co. KG. Die Systempartner sind tätig in dem Bereich Gebäudeautomation und beschäftigen sich mit Errichtung, Inbetriebsetzung, Erweiterung, Wartung und Service von mess-, steuer- und regelungstechnischen Anlagen der technischen Gebäudeausrüstung.

3 DALI

DALI (Digital Addressable Lighting Interface) ist eine Definition zur Standardisierung digitaler Schnittstellen von Vorschaltgeräten (Lampen) und Steuergeräten (Sensoren). Mit diesem Standard (IEC 62386) sind die Hersteller von Beleuchtungskomponenten in der Lage, komplexe Beleuchtungsaufgaben einfach und komfortabel zu lösen.

Die Busklemmen [KL6811](#) (DALI/DSI-Master) und [KL6821](#) (DALI2-Master) werden als normale Busklemme in das Busklemmensystem integriert und sind somit feldbusunabhängig. Über den jeweiligen Buskoppler werden die DALI-Daten an die DALI-Geräte weitergeleitet. Buscontroller bieten zusätzlich die Möglichkeit SPS-Programme in IEC 61131-3 dezentral auszuführen.

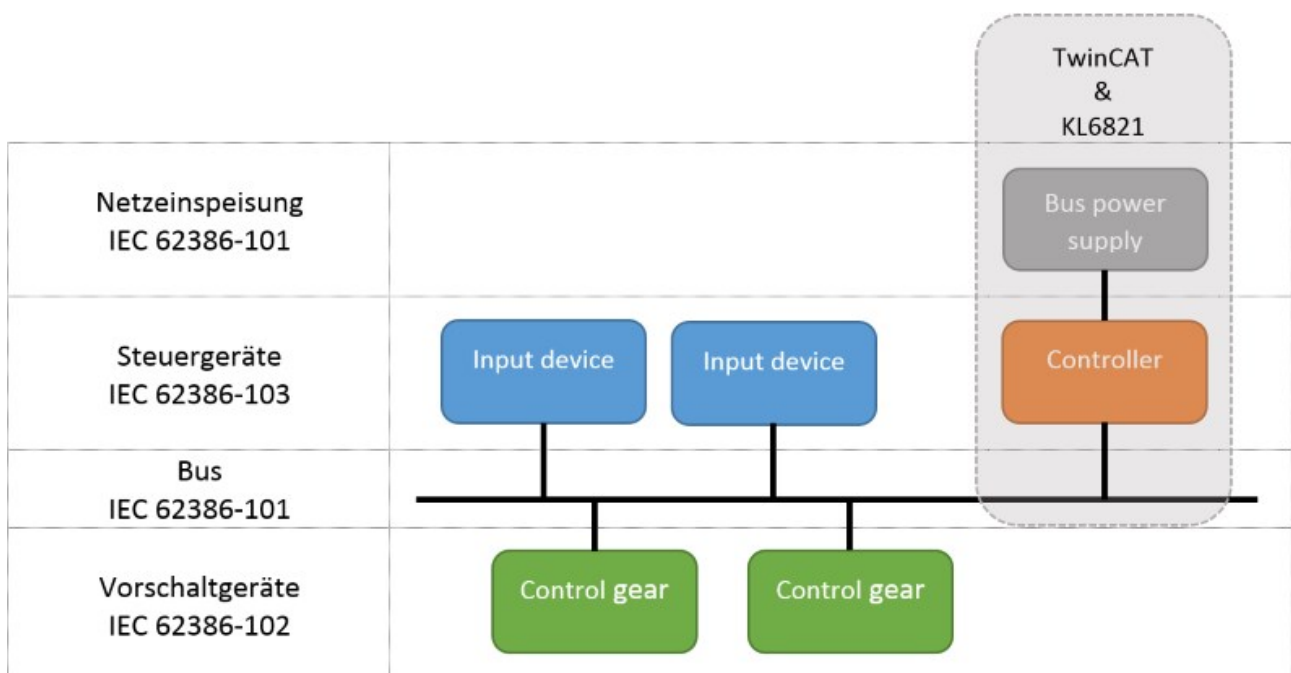
IEC 62386

DALI wird im Standard IEC 62386 spezifiziert und bietet Vorteile wie Flexibilität, Einfachheit, Benutzerfreundlichkeit und Robustheit. Die IEC 62386 wurde mehrfach überarbeitet und im November 2014 mit der Veröffentlichung der zweiten Revision deutlich erweitert. Während in der ersten Revision ausschließlich Vorschaltgeräte (Lampen) berücksichtigt wurden, werden ab der zweiten Revision auch Steuergeräte (Sensoren) mit einbezogen. Beschrieben werden diese in den jeweiligen Bereichen der IEC 62386:

IEC 62386-101	allgemeine Systemeigenschaften wie Verkabelung, Netzeinspeisung und Telegrammaufbau.
IEC 62386-102	allgemeine Eigenschaften der Vorschaltgeräte (Control gears).
	IEC 62386-201: Leuchtstofflampen (Gerätetyp 0) IEC 62386-202: Notbeleuchtung (Gerätetyp 1) IEC 62386-203: Entladungslampen (Gerätetyp 2) ...
IEC 62386-103	allgemeine Eigenschaften der Steuergeräte (Control devices).
	IEC 62386-301: Taster IEC 62386-303: Präsenzmelder IEC 62386-304: Helligkeitssensor ...

Die IEC 62386-101, IEC 62386-102 und IEC 62386-103 beschreiben allgemeine Eigenschaften, während IEC 62386-2xx und IEC 62386-3xx die einzelnen Gerätetypen spezifizieren.

IEC 62386-103 und IEC 62386-3xx wurden mit der Revision 2 in den DALI-Standard aufgenommen.

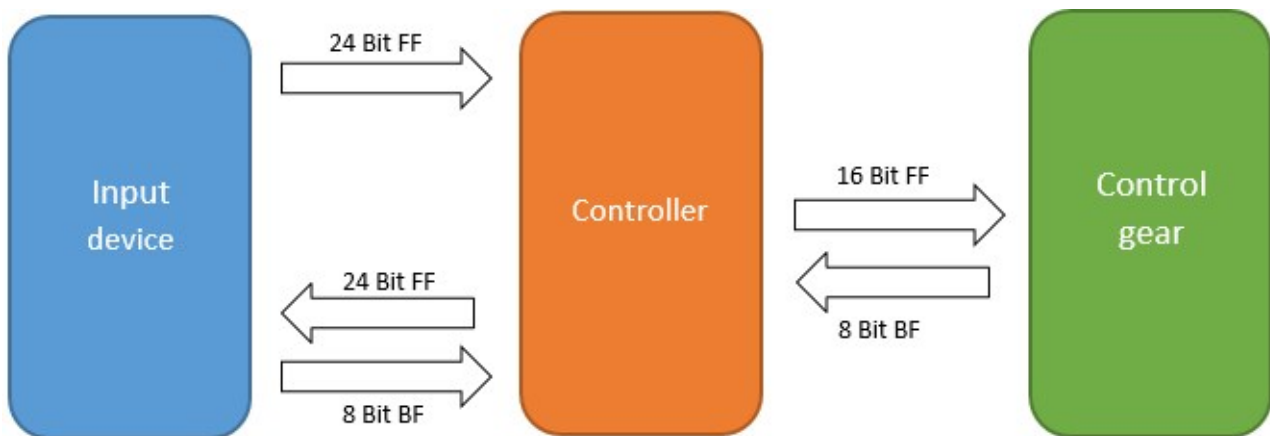


Pro DALI-Linie können bis zu 64 Vorschaltgeräte (Control gear) und bis zu 64 Eingangsgeräte (Input device) angeschlossen werden. Die KL6821 stellt den DALI-Controller da und ist pro DALI-Linie einmal vorhanden. Pro TwinCAT-Controller können beliebig viele DALI-Linien (KL6821) betrieben werden.

Kommunikation

Bei der Kommunikation wird zwischen drei Arten von Telegrammen unterschieden:

- 16-Bit Abfrage, Konfiguration- und Steuer-Telegramm.
- 24-Bit Abfrage, Konfiguration- und Steuer-Telegramm.
- 24-Bit Ereignis-Telegramm.



BF: Backward frame
FF: Forward frame

16-Bit Telegramme

16-Bit Telegramme werden immer von einem DALI-Controller an ein DALI-Vorschaltgerät (Control gear) gesendet. Diese dienen dazu die Geräte zu konfigurieren, Parameter abzufragen oder um Steuerbefehle zu versenden. Bei bestimmten DALI-Befehlen schickt das DALI-Vorschaltgerät eine 8-Bit Rückantwort. DALI-Vorschaltgeräte versenden nur auf Anforderung ein 8-Bit Telegramm.

In der DALI-Bibliothek werden diese Befehle durch die SPS-Bausteine mit dem Präfix *FB_DALIV2* bereitgestellt, also z.B. *FB_DALIV2QueryActualLevel()*.

24-Bit Telegramme

24-Bit Telegramme werden immer von einem DALI-Controller an ein DALI-Eingangsgerät (Input device) gesendet. Diese dienen auch dazu die Geräte zu konfigurieren, Parameter abzufragen oder um Steuerbefehle zu versenden. Bei bestimmten DALI-Befehlen schickt das DALI-Eingangsgerät eine 8-Bit Rückantwort.

In der DALI-Bibliothek werden diese Befehle durch die SPS-Bausteine mit dem Präfix *FB_DALIV2x* bereitgestellt, also z.B. *FB_DALIV2xQueryOperatingMode()*.

24-Bit Ereignisse

DALI-Eingangsgeräte sind in der Lage Ereignisse zu versenden. Diese werden immer von dem DALI-Controller ausgewertet und haben auch eine Länge von 24 Bit.

Einzelne Ereignisse können mit dem Baustein *FB_DALIV2xGetEventData()* [► 97] ausgefiltert und somit weiterverarbeitet werden.

Weitere Informationen zu DALI finden Sie auf der Homepage der DALI Activity Group (<http://www.dali-ag.org>) oder der Digital Illumination Interface Alliance (<https://www.digitalilluminationinterface.org>) und in der Norm IEC 62386.

Die hier vorliegende Version der DALI-Bibliothek ersetzt die bisherige Version der DALI-Bibliothek. Die ältere DALI-Library finden Sie mit Anleitung unter <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019285387.zip>. Sie sollten bei neueren Projekten nur noch die aktuelle Version einsetzen, da die ältere Version nicht weiterentwickelt wird.

Die KL6811 unterstützt nur die erste Revision des DALI-Standards. Der Betrieb von Steuergeräten (Sensoren) ist mit der KL6811 nicht möglich. Die KL6821 ist abwärtskompatibel zur KL6811, unterstützt aber kein DSI.

4 Integration in TwinCAT

4.1 KL6811 Integration in TwinCAT (CX9020)

Dieses Beispiel beschreibt, wie ein einfaches SPS-Programm für DALI in TwinCAT geschrieben werden kann und wie es mit der Hardware verknüpft wird. Es soll eine einzelne dimmbare Lampe angesteuert und per Taster verändert werden.

<https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019286795.zip> <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019286795.zip>

Hardware

Einrichtung der Komponenten

Es wird folgende Hardware benötigt:

- 1x Embedded-PC [CX9020](#)
- 1x Digitale 2-Kanal-Eingangsklemme KL1002 (für die Dimm und Reset Funktion)
- 1x [DALI-Klemme KL6811](#)
- 1x Endklemme KL9010

Richten Sie die Hardware sowie die DALI-Komponenten wie in den entsprechenden Dokumentationen beschrieben ein.

Dieses Beispiel geht davon aus, dass ein Dimm-Taster auf den ersten und ein Reset-Taster auf den zweiten Eingang der KL1002 gelegt wurde und sich an der DALI-Adresse 0 eine dimmbare Lampe befindet. Stellen Sie die Fade-Rate des Vorschaltgerätes zunächst auf 7 um ein ansprechendes Dimmen zu erzielen.

Software

Erstellung des SPS-Programms

Erstellen Sie ein neues SPS-Projekt für PC-basierte Systeme (ARM) und fügen die Bibliothek *TcDALIV2.lib* hinzu.

Erzeugen Sie als Nächstes die folgenden globalen Variablen:

```
VAR_GLOBAL
  bSwitch      AT %I*      : BOOL;
  bReset       AT %I*      : BOOL;
  stDALIInData AT %I*      : ST_DALIV2InData;
  stDALIOutData AT %Q*     : ST_DALIV2OutData;
  stCommandBuffer : ST_DALIV2CommandBuffer;
END_VAR
```

bSwitch: Eingangsvariable für den Dimm-Taster.

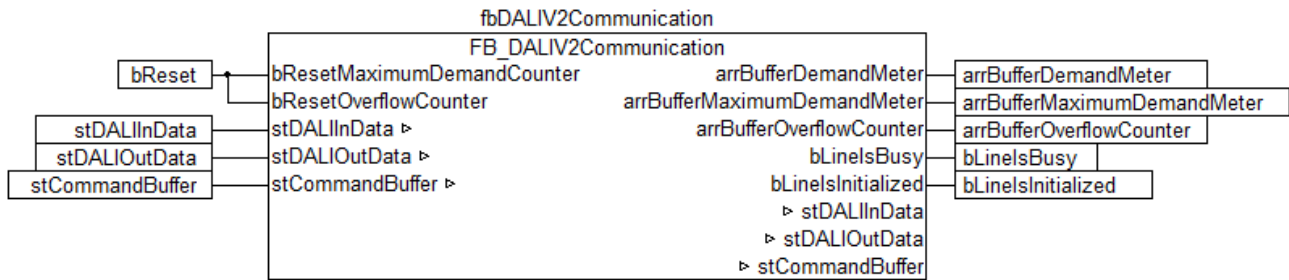
bReset: Eingangsvariable für den Reset-Taster.

stDALIInData: Eingangsvariable für die DALI-Klemme.

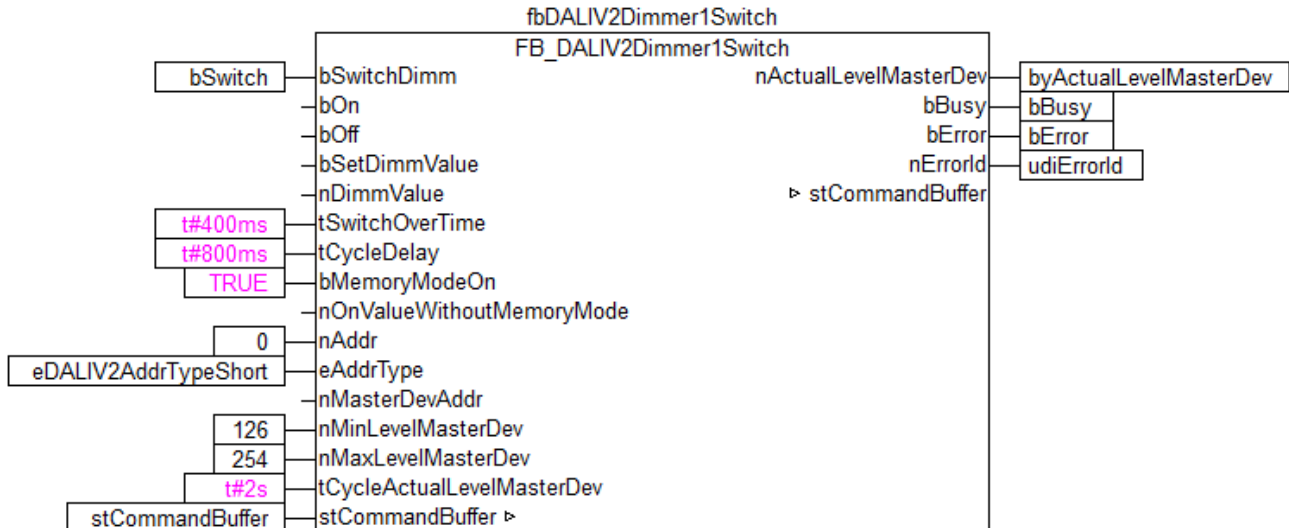
stDALIOutData: Ausgangsvariable für die DALI-Klemme.

stCommandBuffer: Wird für die Kommunikation mit DALI benötigt.

Legen Sie anschließend ein Programm (CFC) für die Hintergrundkommunikation mit DALI an. In diesem wird der Baustein [FB_DALIV2Communication\(\)](#) [► 95] aufgerufen. Achten Sie beim Kommunikationsbaustein darauf, mit *stDALIInData*, *stDALIOutData* und *stCommandBuffer* zu verknüpfen.



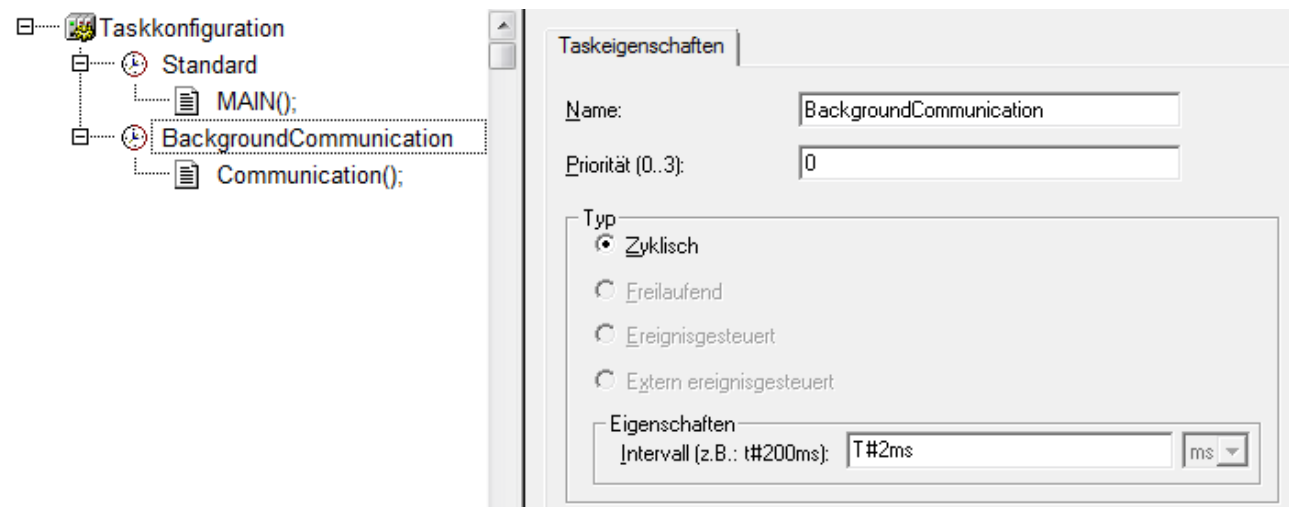
Legen Sie ein MAIN-Programm (CFC) an in dem der Baustein FB_DALIV2Dimmer1Switch() [► 56] aufgerufen wird. Der Eingang *bSwitchDimm* des Dimmer-Bausteins wird mit der globalen Variable *bSwitch* verknüpft und *stCommandBuffer* mit der globalen Variable *stCommandBuffer*.



Parameter

Die eingetragenen Parameter *nMinlevelMasterDevice* und *nMaxLevelMasterDevice* müssen unbedingt dem Minimal- und Maximalwert des angesprochenen Gerätes entsprechen, da es sonst zu Fehlfunktionen kommen kann.

Gehen Sie in die Taskkonfiguration und legen Sie eine neue Task für die Hintergrundkommunikation an. Fügen Sie dieser Task das Programm für die Kommunikation hinzu. Geben Sie dieser Task eine höhere Priorität (kleinere Zahl) und eine niedrigere Intervall-Zeit als der Standard-Task. Genauere Informationen dazu finden Sie in der Beschreibung des Bausteins FB_DALIV2Communication() [► 95].

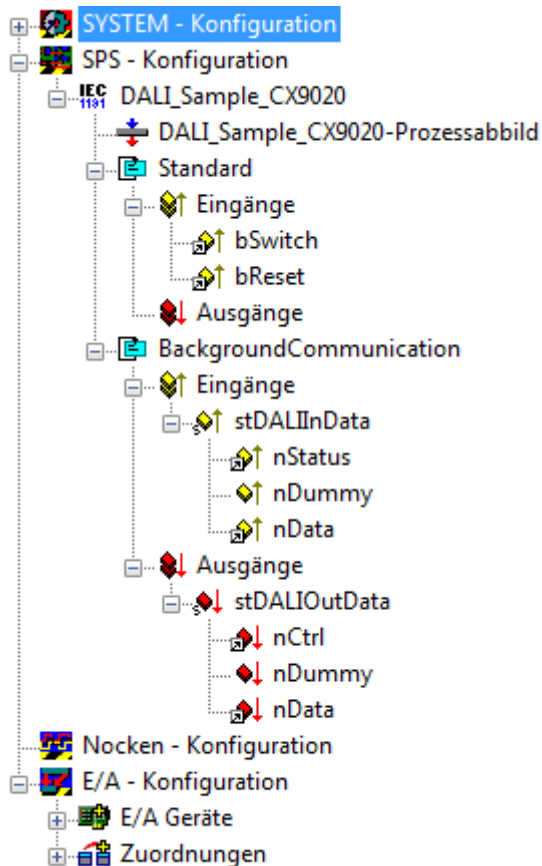


Laden Sie das Projekt als Bootprojekt auf den CX und speichern Sie es ab.

Konfiguration im System Manager

Legen Sie ein neues System-Manager-Projekt an, wählen Sie als Zielsystem den CX und lassen Sie nach dessen Hardware suchen.

Fügen Sie das oben angelegte SPS-Programm unter SPS-Konfiguration hinzu. Beim Aufklappen des SPS-Projekts in der Baumansicht werden die beiden Tasks aufgelistet. Beim Erweitern der Tasks ist sicherzustellen, dass alle globalen Ein- und Ausgangsvariablen der Standard-Task zugeordnet sind. Da die Variablen *stDALIInData* und *stDALIOutData* jedoch schneller abgearbeitet werden sollen, müssen diese mit Drag&Drop auf die Hintergrund-Kommunikations-Task verschoben werden.



Verknüpfen Sie die globalen Variablen des SPS-Programms nun mit den Ein- und Ausgängen der Busklemmen, erzeugen Sie die Zuordnungen und aktivieren Sie die Konfiguration. Starten Sie dann das Gerät im Run-Modus.

Ihr CX ist jetzt einsatzbereit.

Durch kurzes oder längeres Drücken auf den Dimm-Taster kann nun die Lampe geregelt werden. Mit dem Reset-Taster können Sie die Einträge in *arrBufferMaximumDemandMeter* und *arrBufferOverflowCounter* zurücksetzen.

Sehen Sie dazu auch

- 📄 [ST_DALIV2InData \[▶ 390\]](#)
- 📄 [ST_DALIV2OutData \[▶ 391\]](#)

4.2 KL6811 Integration in TwinCAT (BC9191)

Dieses Beispiel beschreibt, wie ein einfaches SPS-Programm für DALI in TwinCAT geschrieben werden kann und wie es mit der Hardware verknüpft wird. Es soll eine einzelne dimmbare Lampe angesteuert und per Taster verändert werden.

<https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019288203.zip> <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019288203.zip>

Hardware

Einrichtung der Komponenten

Es wird folgende Hardware benötigt:

- 1x Busklemmen Controller [BC9191](#)
- 1x Potenzialeinspeiseklemme 24V DC
- 1x Digitale 2-Kanal-Eingangsklemme KL1002 (für die Dimm und Reset Funktion)
- 1x [DALI-Klemme KL6811](#)
- 1x Endklemme KL9010

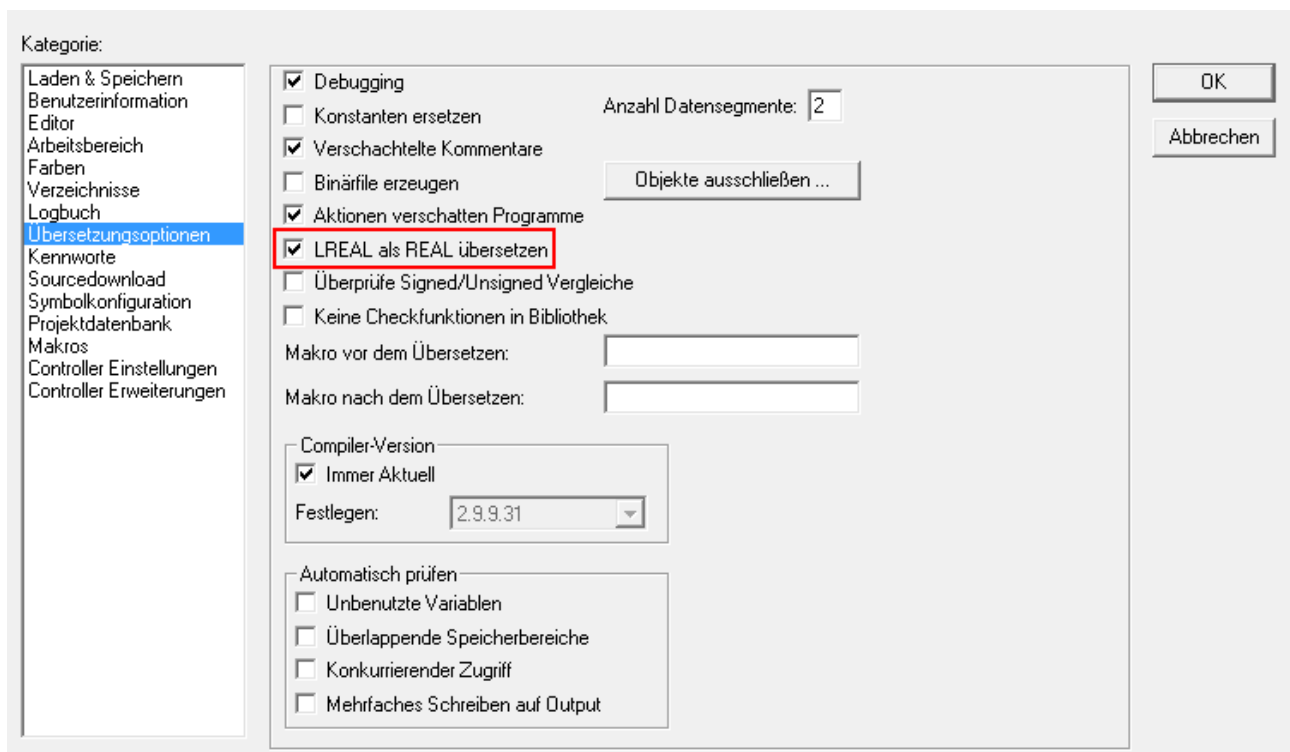
Richten Sie die Hardware sowie die DALI-Komponenten wie in den entsprechenden Dokumentationen beschrieben ein.

Dieses Beispiel geht davon aus, dass ein Dimm-Taster auf den ersten und ein Reset-Taster auf den zweiten Eingang der KL1002 gelegt wurde und sich an der DALI-Adresse 0 eine dimmbare Lampe befindet. Stellen Sie die Fade-Rate des Vorschaltgerätes zunächst auf 7 um ein ansprechendes Dimmen zu erzielen.

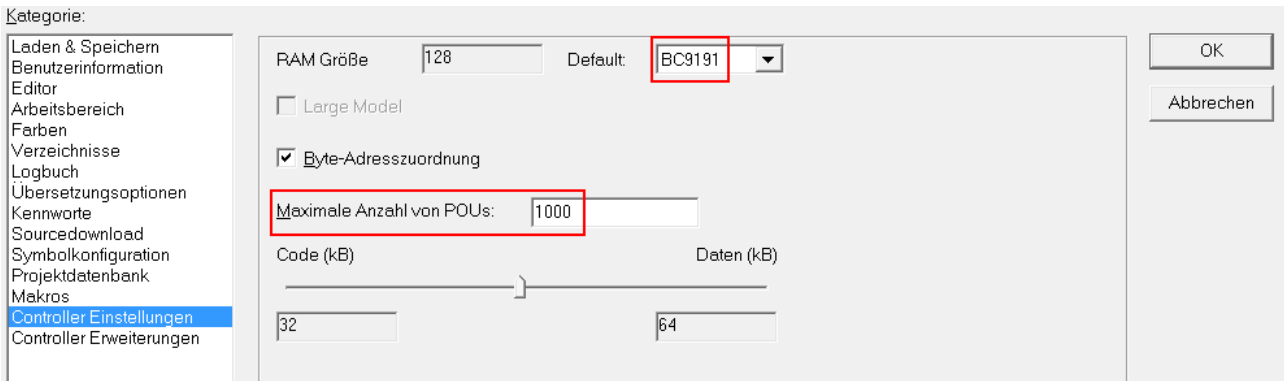
Software

Erstellung des SPS-Programms

Erstellen Sie ein neues SPS-Projekt für BC-basierte Systeme (BCxx50 über AMS) und fügen die Bibliotheken *TcDALIV2.lbx* und *TcSystemBCxx50.lbx* hinzu. Gehen Sie danach im Menü auf *Projekt* → *Optionen...* → *Übersetzungsoptionen* und wählen *LREAL als REAL übersetzen* an.



Unter *Controller Einstellungen* wählen Sie den *BC9191* und stellen Sie die *Maximale Anzahl der POU's* auf 1000



Erzeugen Sie als Nächstes die folgenden globalen Variablen:

```
VAR_GLOBAL
  bSwitch      AT %I*    : BOOL;
  bReset       AT %I*    : BOOL;
  stDALIInData AT %I*    : ST_DALIV2InData;
  stDALIOutData AT %Q*   : ST_DALIV2OutData;
  stCommandBuffer : ST_DALIV2CommandBuffer;
END_VAR
```

bSwitch: Eingangsvariable für den Dimm-Taster.

bReset: Eingangsvariable für den Reset-Taster.

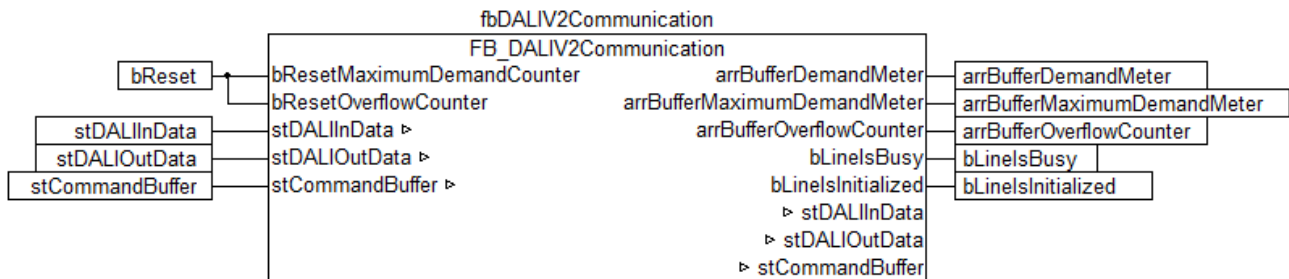
stDALIInData: Eingangsvariable für die DALI-Klemme.

stDALIOutData: Ausgangsvariable für die DALI-Klemme.

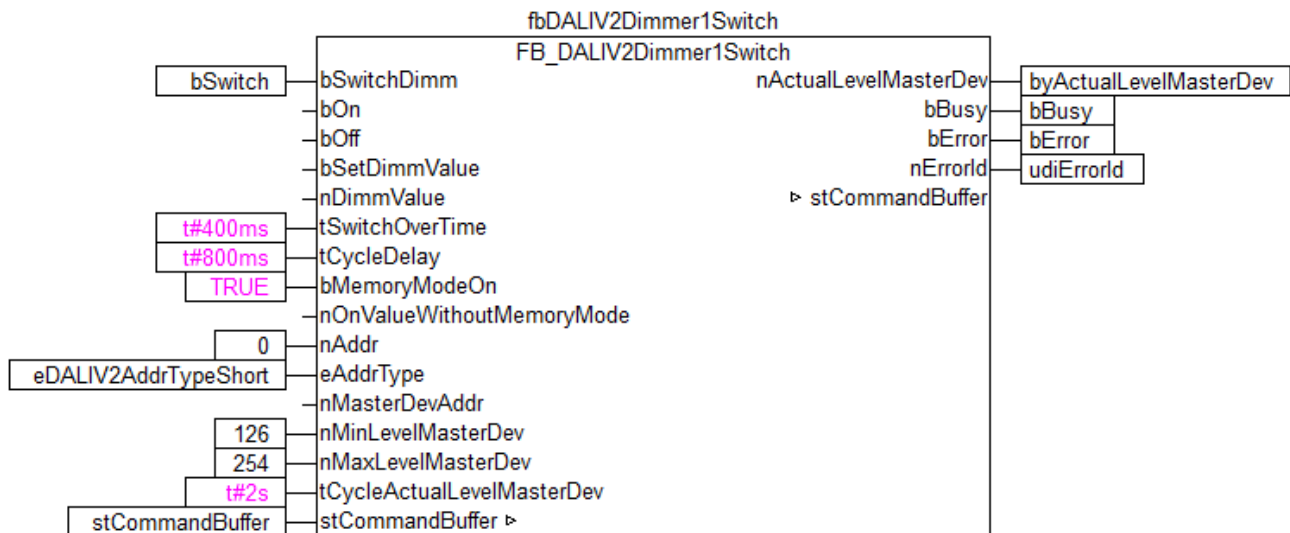
stCommandBuffer: Wird für die Kommunikation mit DALI benötigt.

Da BC-Geräte nur über eine Task verfügen, kann die Kommunikation mit DALI nicht separat ausgeführt werden.

Legen Sie daher ein MAIN-Programm (CFC) an in dem die Bausteine `FB_DALIV2Communication()` [► 95] und `FB_DALIV2Dimmer1Switch()` [► 56] aufgerufen werden. Achten Sie beim Kommunikationsbaustein darauf, mit `stDALIInData`, `stDALIOutData` und `stCommandBuffer` zu verknüpfen.



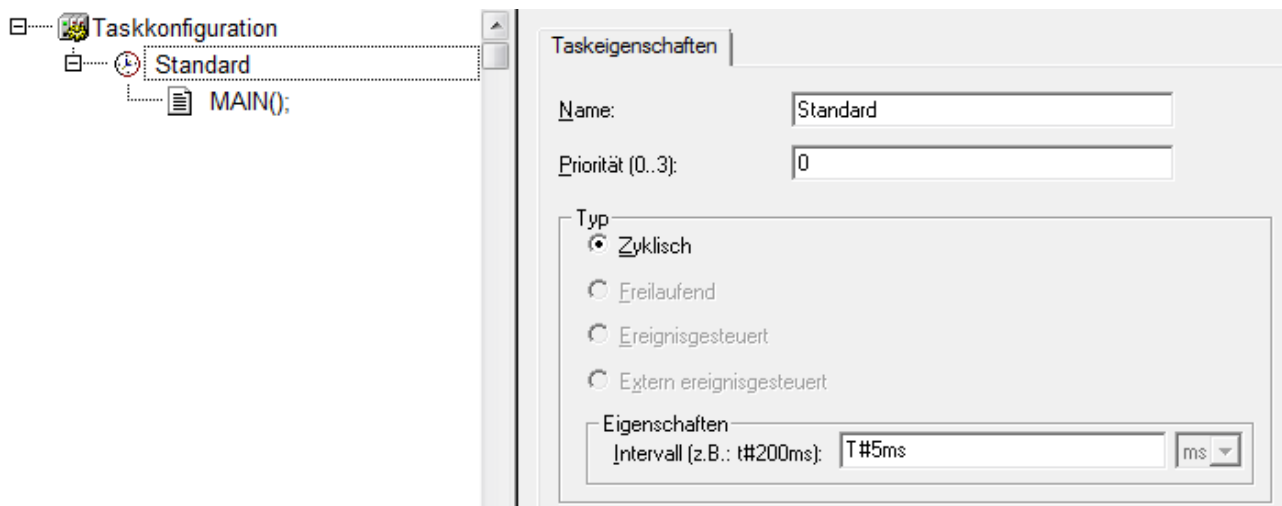
Der Eingang `bSwitchDimm` des Dimmer-Bausteins wird mit der globalen Variable `bSwitch` verknüpft und `stCommandBuffer` mit der globalen Variable `stCommandBuffer`.



i **Parameter**

Die eingetragenen Parameter nMinlevelMasterDevice und nMaxLevelMasterDevice müssen unbedingt dem Minimal- und Maximalwert des angesprochenen Gerätes entsprechen, da es sonst zu Fehlfunktionen kommen kann.

Gehen Sie in die Taskkonfiguration und geben Sie der Task eine niedrigere Intervall-Zeit. Genauere Informationen dazu finden Sie in der Beschreibung des Bausteins `FB_DALIV2Communication()` [► 95].

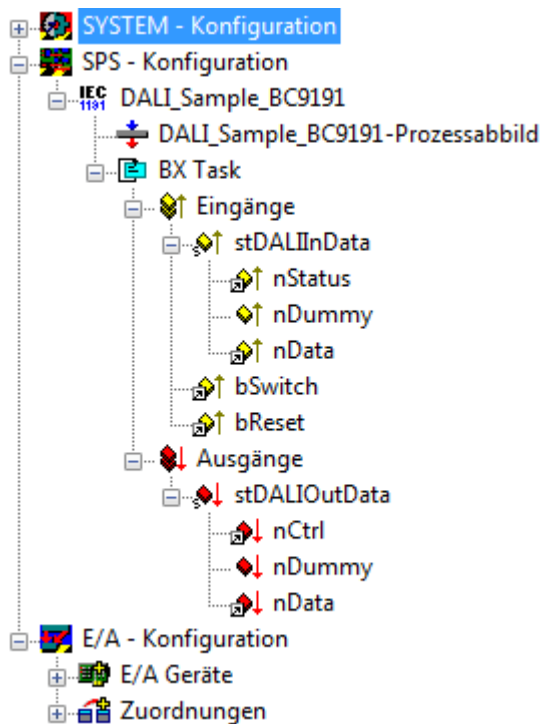


Laden Sie das Projekt als Bootprojekt auf den BC und speichern Sie es ab.

Konfiguration im System Manager

Legen Sie ein neues TwinCAT System-Manager-Projekt an, wählen Sie als Zielsystem den BC und lassen Sie nach dessen Hardware suchen.

Fügen Sie das oben angelegte SPS-Programm unter SPS-Konfiguration hinzu.



Verknüpfen Sie die globalen Variablen des SPS-Programms nun mit den Ein- und Ausgängen der Busklemmen, erzeugen Sie die Zuordnungen und aktivieren Sie die Konfiguration. Starten Sie dann das Gerät im Run-Modus.

Ihr BC ist jetzt einsatzbereit.

Durch kurzes oder längeres Drücken auf den Dimm-Taster kann nun die Lampe geregelt werden. Mit dem Reset-Taster können Sie die Einträge in *arrBufferMaximumDemandMeter* und *arrBufferOverflowCounter* zurücksetzen.

Sehen Sie dazu auch

- 📄 ST_DALIV2InData [▶ 390]
- 📄 ST_DALIV2OutData [▶ 391]

4.3 KL6821 Integration in TwinCAT (CX9020)

Dieses Beispiel beschreibt, wie ein einfaches SPS-Programm für DALI in TwinCAT geschrieben werden kann und wie es mit der Hardware verknüpft wird. Es soll eine einzelne dimmbare Lampe angesteuert und per Taster verändert werden.

<https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019289611.zip> <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019289611.zip>

Hardware

Einrichtung der Komponenten

Es wird folgende Hardware benötigt:

- 1x Embedded-PC [CX9020](#)
- 1x Digitale 4-Kanal-Eingangsklemme KL1104 (für die Dimm und Reset Funktion)
- 1x DALI-Klemme [KL6821](#)
- 1x Endklemme KL9010

Richten Sie die Hardware sowie die DALI-Komponenten wie in den entsprechenden Dokumentationen beschrieben ein.

Dieses Beispiel geht davon aus, dass ein Dimm-Taster auf den ersten und ein Reset-Taster auf den zweiten Eingang der KL1104 gelegt wurde und sich an der DALI-Adresse 0 eine dimmbare Lampe befindet. Stellen Sie die Fade-Rate des Vorschaltgerätes zunächst auf 7 um ein ansprechendes Dimmen zu erzielen.

Software

Erstellung des SPS-Programms

Erstellen Sie ein neues SPS-Projekt für PC-basierte Systeme (ARM) und fügen die Bibliothek *TcDALIV2.lib* hinzu.

Erzeugen Sie als Nächstes die folgenden globalen Variablen:

```
VAR_GLOBAL
  bSwitch      AT %I*      : BOOL;
  bReset       AT %I*      : BOOL;
  stKL6821InData  AT %I*   : ST_KL6821InData;
  stKL6821OutData AT %Q*   : ST_KL6821OutData;
  stCommandBuffer : ST_DALIV2CommandBuffer;
END_VAR
```

bSwitch: Eingangsvariable für den Dimm-Taster.

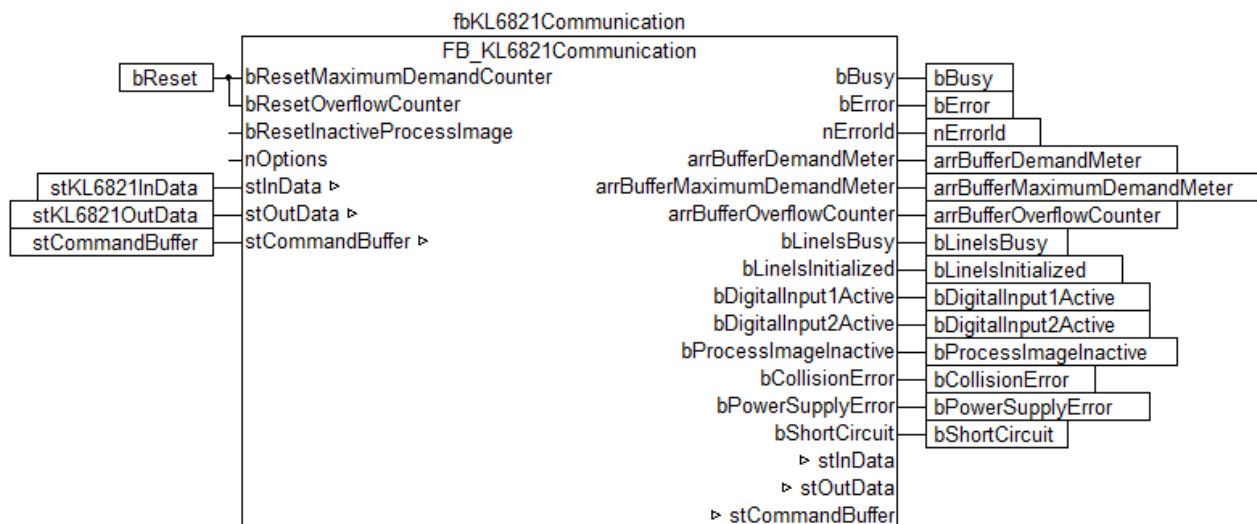
bReset: Eingangsvariable für den Reset-Taster.

ST_KL6821InData: [Eingangsvariable \[▶ 391\]](#) für die DALI-Klemme.

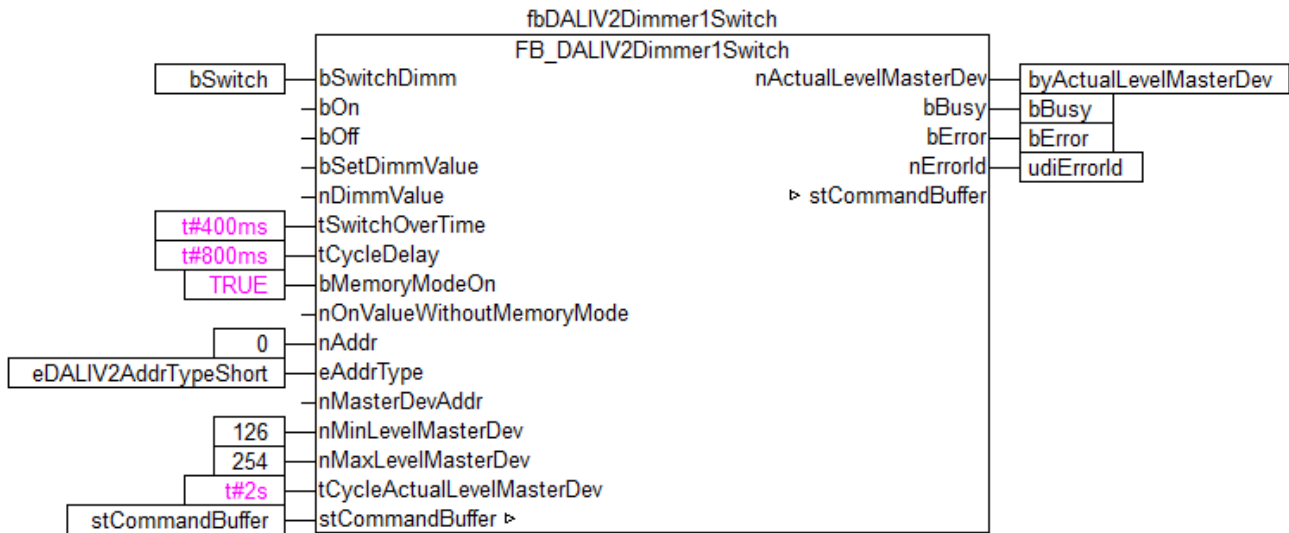
ST_KL6821OutData: [Ausgangsvariable \[▶ 391\]](#) für die DALI-Klemme.

stCommandBuffer: Wird für die Kommunikation mit DALI benötigt.

Legen Sie anschließend ein Programm (CFC) für die Hintergrundkommunikation mit DALI an. In diesem wird der Baustein `FB_KL6821Communication()` [\[▶ 104\]](#) aufgerufen. Achten Sie beim Kommunikationsbaustein darauf, mit *stInData*, *stOutData* und *stCommandBuffer* zu verknüpfen.



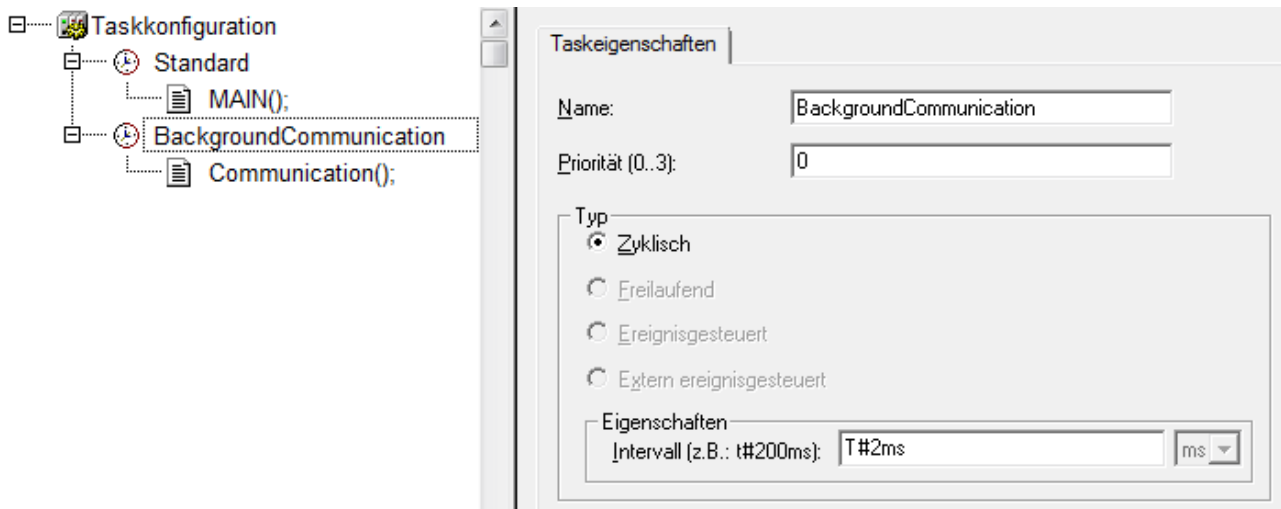
Legen Sie ein MAIN-Programm (CFC) an in dem der Baustein `FB_DALIV2Dimmer1Switch()` [\[▶ 56\]](#) aufgerufen wird. Der Eingang *bSwitchDimm* des Dimmer-Bausteins wird mit der globalen Variable *bSwitch* verknüpft und *stCommandBuffer* mit der globalen Variable *stCommandBuffer*.



Parameter

i Die eingetragenen Parameter nMinlevelMasterDevice und nMaxLevelMasterDevice müssen unbedingt dem Minimal- und Maximalwert des angesprochenen Gerätes entsprechen, da es sonst zu Fehlfunktionen kommen kann.

Gehen Sie in die Taskkonfiguration und legen Sie eine neue Task für die Hintergrundkommunikation an. Fügen Sie dieser Task das Programm für die Kommunikation hinzu. Geben Sie dieser Task eine höhere Priorität (kleinere Zahl) und eine niedrigere Intervall-Zeit als der Standard-Task. Genauere Informationen dazu finden Sie in der Beschreibung des Bausteins FB_KL6821Communication() [►_104].

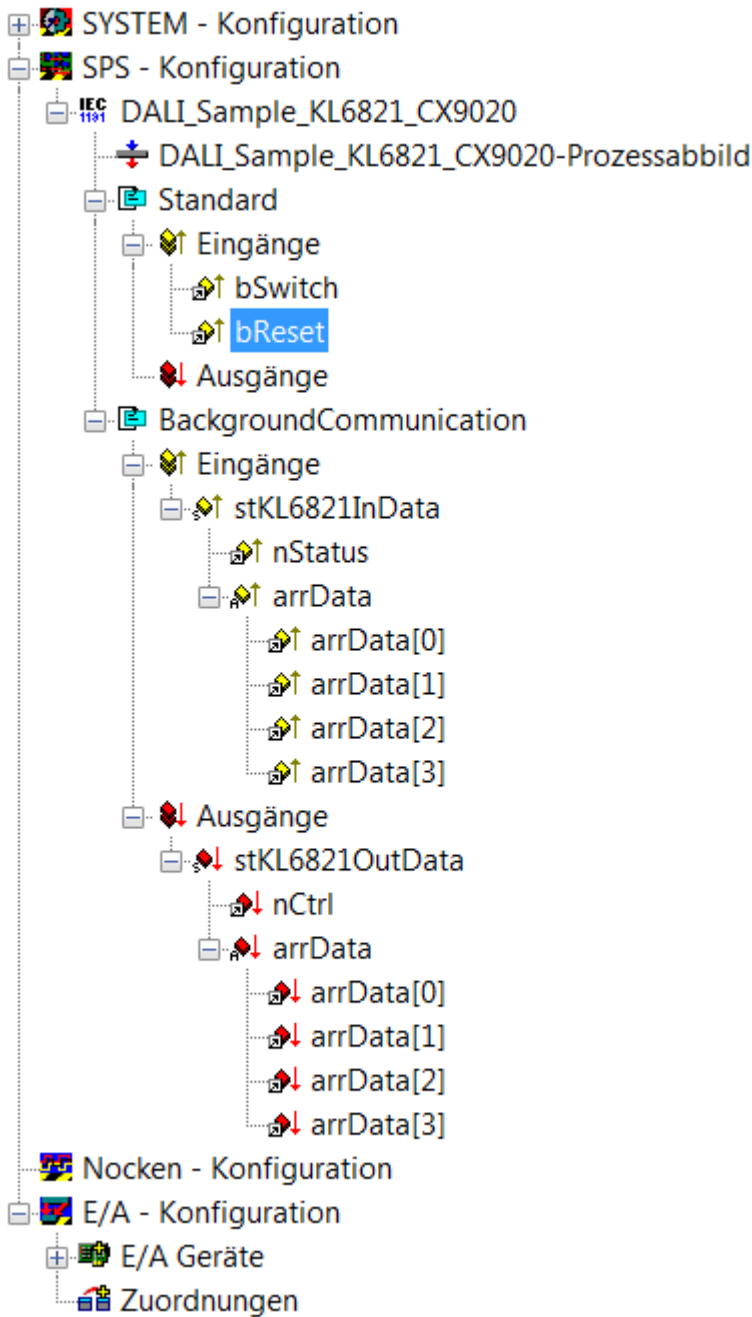


Laden Sie das Projekt als Bootprojekt auf den CX und speichern Sie es ab.

Konfiguration im System Manager

Legen Sie ein neues System-Manager-Projekt an, wählen Sie als Zielsystem den CX und lassen Sie nach dessen Hardware suchen.

Fügen Sie das oben angelegte SPS-Programm unter SPS-Konfiguration hinzu. Beim Aufklappen des SPS-Projekts in der Baumansicht werden die beiden Tasks aufgelistet. Beim Erweitern der Tasks ist sicherzustellen, dass alle globalen Ein- und Ausgangsvariablen der Standard-Task zugeordnet sind. Da die Variablen *stDALIInData* und *stDALIOutData* jedoch schneller abgearbeitet werden sollen, müssen diese mit Drag&Drop auf die Hintergrund-Kommunikations-Task verschoben werden.



Verknüpfen Sie die globalen Variablen des SPS-Programms nun mit den Ein- und Ausgängen der Busklemmen, erzeugen Sie die Zuordnungen und aktivieren Sie die Konfiguration. Starten Sie dann das Gerät im Run-Modus.

Ihr CX ist jetzt einsatzbereit.

Durch kurzes oder längeres Drücken auf den Dimm-Taster kann nun die Lampe geregelt werden. Mit dem Reset-Taster können Sie die Einträge in *arrBufferMaximumDemandMeter* und *arrBufferOverflowCounter* zurücksetzen.

Sehen Sie dazu auch

- FB_DALIV2Communication [► 95]

4.4 KL6821 Integration in TwinCAT (BC9191)

Dieses Beispiel beschreibt, wie ein einfaches SPS-Programm für DALI in TwinCAT geschrieben werden kann und wie es mit der Hardware verknüpft wird. Es soll eine einzelne dimmbare Lampe angesteuert und per Taster verändert werden.

<https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019291019.zip> <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019291019.zip>

Hardware

Einrichtung der Komponenten

Es wird folgende Hardware benötigt:

- 1x Busklemmen Controller [BC9191](#)
- 1x Potenzialeinspeiseklemme 24V DC, z.B. KL9190
- 1x Digitale 4-Kanal-Eingangsklemme KL1104 (für die Dimm und Reset-Funktion)
- 1x DALI-Klemme [KL6821](#)
- 1x Endklemme KL9010

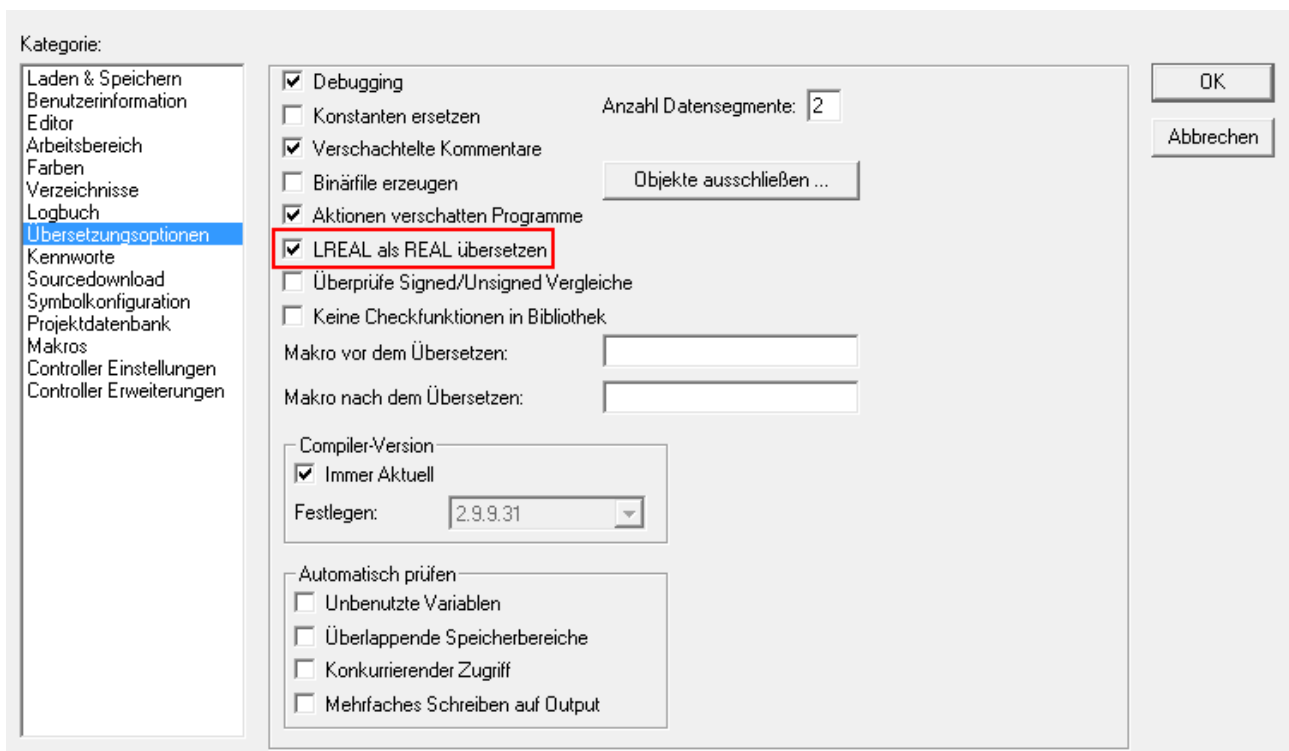
Richten Sie die Hardware sowie die DALI-Komponenten wie in den entsprechenden Dokumentationen beschrieben ein.

Dieses Beispiel geht davon aus, dass ein Dimm-Taster auf den ersten und ein Reset-Taster auf den zweiten Eingang der KL1104 gelegt wurde und sich an der DALI-Adresse 0 eine dimmbare Lampe befindet. Stellen Sie die Fade-Rate des Vorschaltgerätes zunächst auf 7 um ein ansprechendes Dimmen zu erzielen.

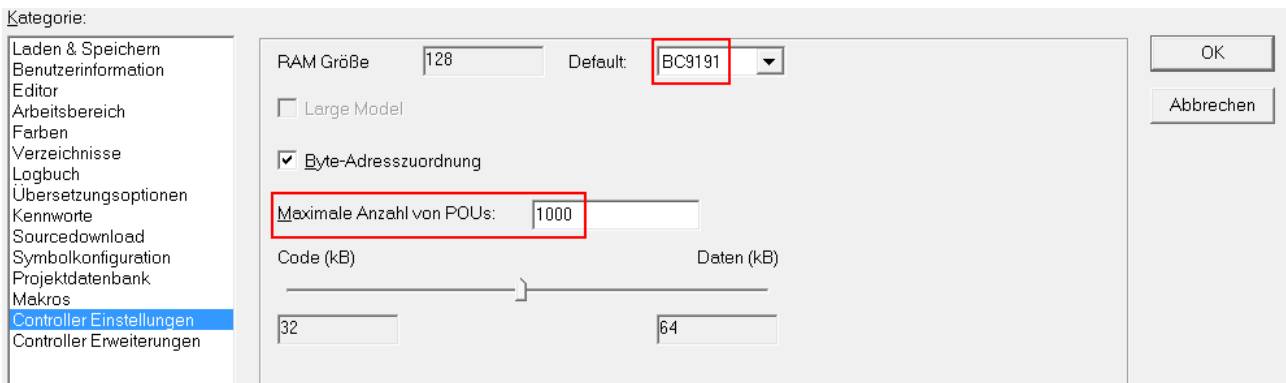
Software

Erstellung des SPS-Programms

Erstellen Sie ein neues SPS-Projekt für BC-basierte Systeme (BCxx50 über AMS) und fügen die Bibliotheken *TcDALIV2.lbx* und *TcSystemBCxx50.lbx* hinzu. Gehen Sie danach im Menü auf *Projekt* → *Optionen...* → *Übersetzungsoptionen* und wählen *LREAL als REAL übersetzen* an.



Unter *Controller Einstellungen* wählen Sie den *BC9191* und stellen Sie die *Maximale Anzahl der POU's* auf 1000



Erzeugen Sie als Nächstes die folgenden globalen Variablen:

```
VAR_GLOBAL
  bSwitch      AT %I*      : BOOL;
  bReset       AT %I*      : BOOL;
  stKL6821InData  AT %I*    : ST_KL6821InData;
  stKL6821OutData AT %Q*    : ST_KL6821OutData;
  stCommandBuffer : ST_DALIV2CommandBuffer;
END_VAR
```

bSwitch: Eingangsvariable für den Dimm-Taster.

bReset: Eingangsvariable für den Reset-Taster.

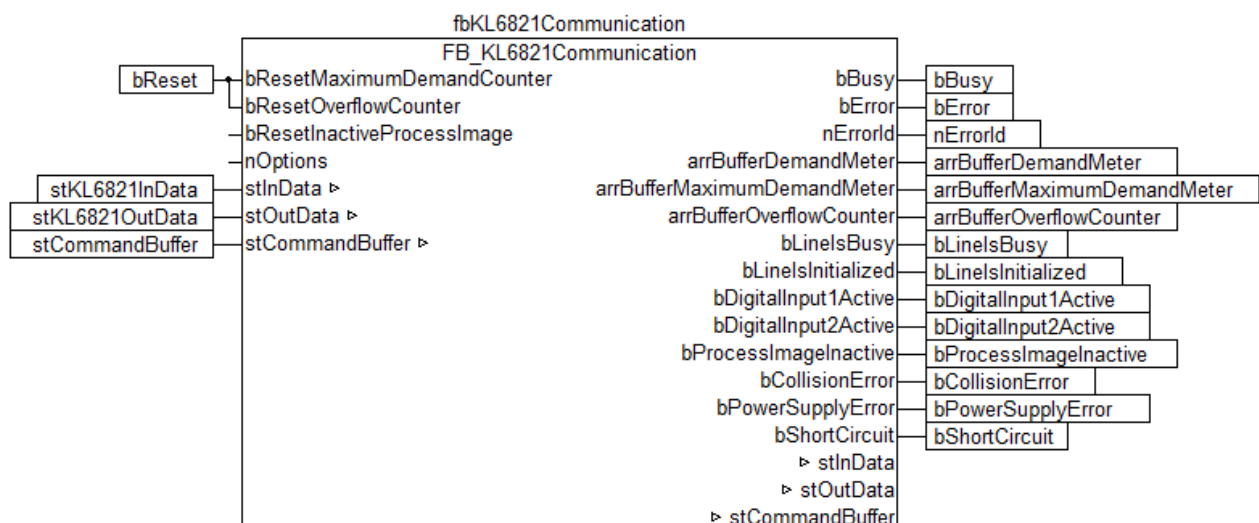
stKL6821InData: Eingangsvariable [▶ 391] für die DALI-Klemme.

stKL6821OutData: Ausgangsvariable [▶ 391] für die DALI-Klemme.

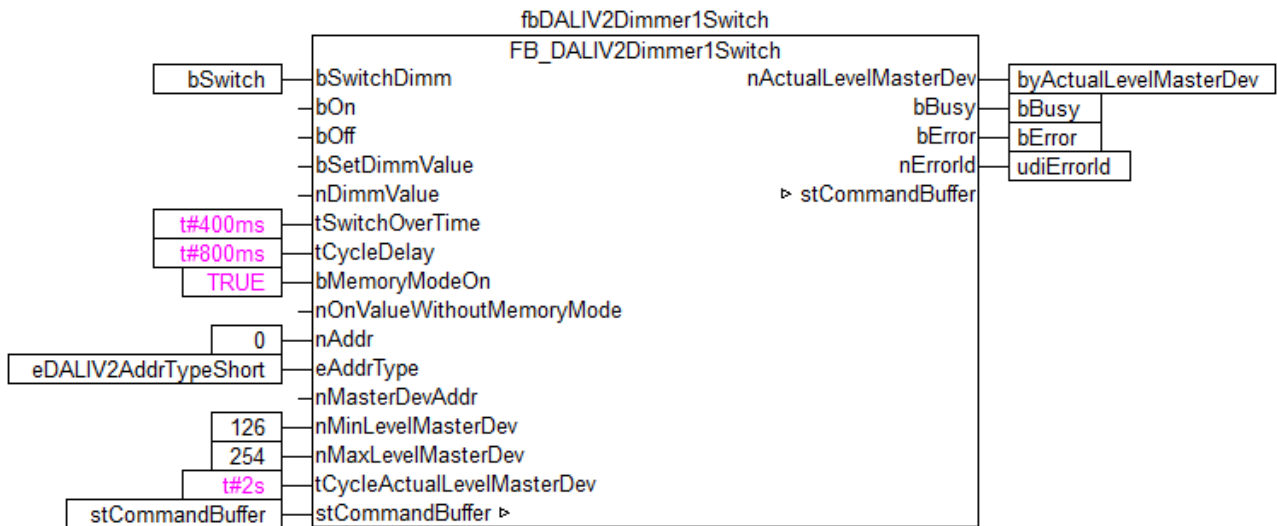
stCommandBuffer: Wird für die Kommunikation mit DALI benötigt.

Da BC-Geräte nur über eine Task verfügen, kann die Kommunikation mit DALI nicht separat ausgeführt werden.

Legen Sie daher ein MAIN-Programm (CFC) an in dem die Bausteine FB_KL6821Communication() [▶ 104] und FB_DALIV2Dimmer1Switch() [▶ 56] aufgerufen werden. Achten Sie beim Kommunikationsbaustein darauf, mit *stKL6821InData*, *stKL6821OutData* und *stCommandBuffer* zu verknüpfen.



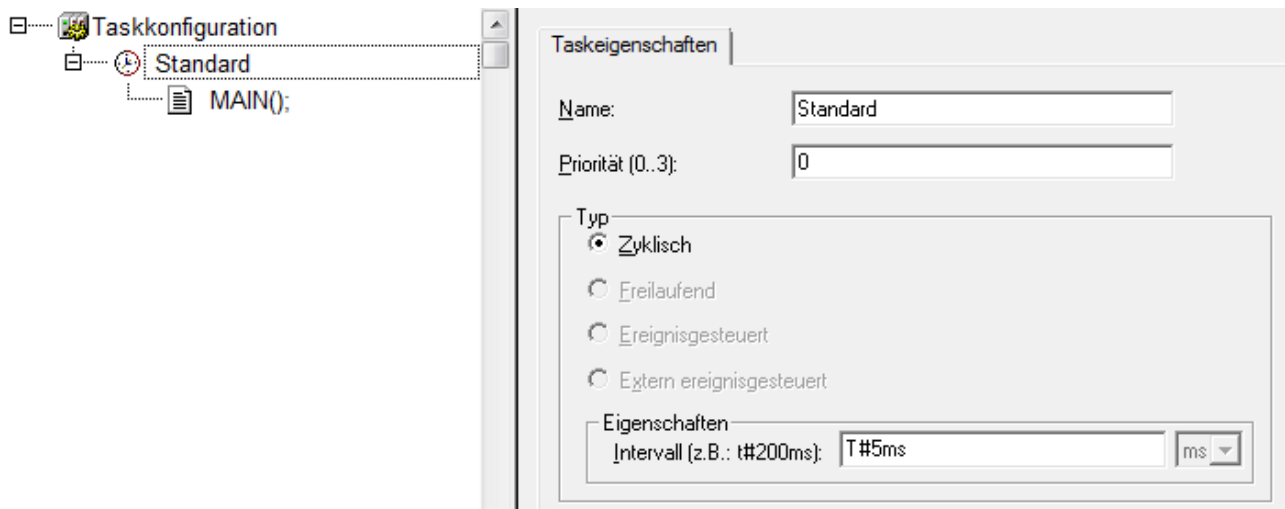
Der Eingang *bSwitchDimm* des Dimmer-Bausteins wird mit der globalen Variable *bSwitch* verknüpft und *stCommandBuffer* mit der globalen Variable *stCommandBuffer*.



i **Parameter**

Die eingetragenen Parameter nMinlevelMasterDevice und nMaxLevelMasterDevice müssen unbedingt dem Minimal- und Maximalwert des angesprochenen Gerätes entsprechen, da es sonst zu Fehlfunktionen kommen kann.

Gehen Sie in die Taskkonfiguration und geben Sie der Task eine niedrigere Intervall-Zeit. Genauere Informationen dazu finden Sie in der Beschreibung des Bausteins `FB_DALIV2Communication()` [► 95].

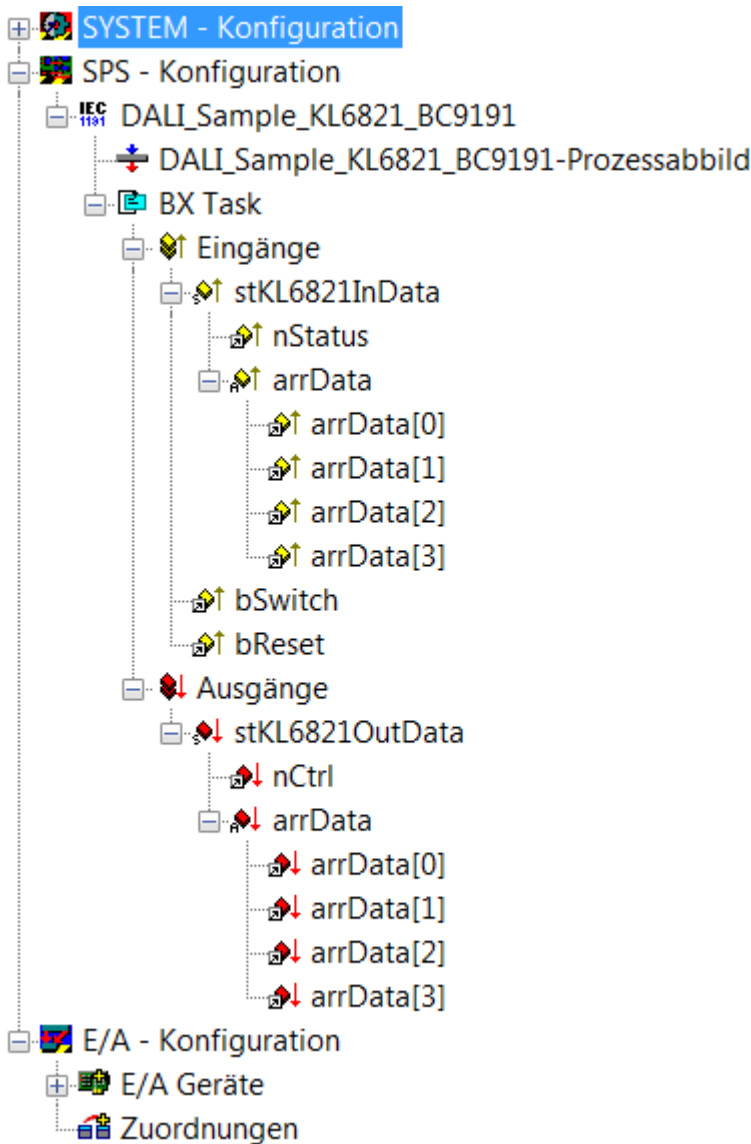


Laden Sie das Projekt als Bootprojekt auf den BC und speichern Sie es ab.

Konfiguration im System Manager

Legen Sie ein neues TwinCAT System-Manager-Projekt an, wählen Sie als Zielsystem den BC und lassen Sie nach dessen Hardware suchen.

Fügen Sie das oben angelegte SPS-Programm unter SPS-Konfiguration hinzu.



Verknüpfen Sie die globalen Variablen des SPS-Programms nun mit den Ein- und Ausgängen der Busklemmen, erzeugen Sie die Zuordnungen und aktivieren Sie die Konfiguration. Starten Sie dann das Gerät im Run-Modus.

Ihr BC ist jetzt einsatzbereit.

Durch kurzes oder längeres Drücken auf den Dimm-Taster kann nun die Lampe geregelt werden. Mit dem Reset-Taster können Sie die Einträge in *arrBufferMaximumDemandMeter* und *arrBufferOverflowCounter* zurücksetzen.

5 Programmierung

● Installation

i Ab TwinCAT 2.11 Build 2229 (R3 und x64 Engineering) werden die Bibliotheken "TcDA-LIV2.lib/.lb6/.lbox" standardmäßig mitinstalliert.

Hardware-Dokumentation im Beckhoff Information System: [KL6811 - Masterklemme für DALI/DSI](#)

Hardware-Dokumentation im Beckhoff Information System: [KL6821 - Masterklemme für DALI2](#)

Weitere erforderliche Bibliotheken

Für PC-Systeme (x86) und Embedded-PCs (CXxxxx):

- Standard.lib
- TcBase.lib
- TcSystem.lib

Für Busklemmen-Controller der Serie BCxx00:

- Standard.lb6
- TcPlcUtilitiesBC.lb6
- PlcHelperBC.lb6
- PlcSystemBC.lb6

Für Busklemmen-Controller der Serien BCxx50, BCxx20 und BC9191:

- Standard.lbx
- TcBaseBCxx50.lbx
- TcSystemBCxx50.lbx

Für Busklemmen-Controller der Serie BXxx00:

- Standard.lbx
- TcBaseBX.lbx
- TcSystemBX.lbx

● Speicherauslastung

i Durch Einbinden der Bibliothek wird bereits SPS-Programmspeicher verbraucht. Abhängig vom Applikationsprogramm kann daher der verbleibende Speicher nicht ausreichend sein.

5.1 Bausteine

5.1.1 Notbeleuchtung - Funktionsbausteine

Im Folgenden werden die Funktionsbausteine und Variablen für Notlichtversorgungsgeräte mit DALI-Schnittstelle beschrieben. Alle folgenden Funktionsbausteine rufen 'anwendungsbezogene Erweiterungsbefehle' auf. Laut DALI-Normung (IEC 62386) liegen diese Befehle im Bereich 224 bis 255. Da es verschiedene anwendungsbezogene Erweiterungsbefehle gibt, muss mit dem Funktionsbaustein `FB_DALIV2EnableDeviceType()` [▶ 356] festgelegt werden, welche Art von Vorschaltgeräten (Notbeleuchtung, Entladungslampen, ...) auf die Erweiterungsbefehle reagieren sollen. Eine genaue Beschreibung der einzelnen DALI-Kommandos und der Variablen für Notlichtversorgungsgeräte finden Sie in der Norm IEC 62386 Part 202.



Name	Beschreibung
FB_DALIV2Inhibit [▶ 224]	Es wird für 15 Minuten verhindert, dass das Vorschaltgerät in den Notbetrieb umschaltet.
FB_DALIV2QueryBatteryCharge [▶ 225]	Die Variable BATTERY CHARGE [▶ 376]
FB_DALIV2QueryDurationTestResult [▶ 226]	Die Variable DURATION TEST RESULT [▶ 376]
FB_DALIV2QueryEmergencyLevel [▶ 227]	Die Variable EMERGENCY LEVEL [▶ 375]
FB_DALIV2QueryEmergencyMaxLevel [▶ 228]	Die Variable EMERGENCY MAX LEVEL [▶ 375]
FB_DALIV2QueryEmergencyMinLevel [▶ 229]	Die Variable EMERGENCY MIN LEVEL [▶ 375]
FB_DALIV2QueryEmergencyMode [▶ 230]	Die Variable EMERGENCY MODE [▶ 377]
FB_DALIV2QueryEmergencyStatus [▶ 232]	Die Variable EMERGENCY STATUS [▶ 378]
FB_DALIV2QueryFailureStatus [▶ 233]	Die Variable FAILURE STATUS [▶ 377]
FB_DALIV2QueryFeatures [▶ 234]	Die Variable FEATURES [▶ 377] (Leistungsmerkmale) wird aus dem Vorschaltgerät ausgelesen.
FB_DALIV2QueryLampEmergencyTime [▶ 235]	Die Variable LAMP EMERGENCY TIME [▶ 377]
FB_DALIV2QueryLampTotalOperationTime [▶ 236]	Die Variable LAMP TOTAL OPERATION TIME [▶ 377]
FB_DALIV2QueryRatedDuration [▶ 237]	Die Variable RATED DURATION [▶ 377]
FB_DALIV2QueryTestTiming [▶ 238]	Je nach Inhalt des DTR (Data Transfer Register) werden die Variablen FUNCTION TEST DELAY TIME [▶ 376], DURATION TEST DELAY TIME [▶ 376], FUNCTION TEST INTERVAL [▶ 376], DURATION TEST INTERVAL [▶ 376], TEST EXECUTION TIMEOUT [▶ 376], PROLONG TIME [▶ 375]
FB_DALIV2RelightResetInhibit [▶ 240]	Das Vorschaltgerät wird zurück in den Notbetrieb geschaltet (wenn die Netzspannung fehlt).
FB_DALIV2ResetDurationTestDoneFlag [▶ 241]	Das Flag 'Betriebsdauertest abgeschlossen und Ergebnis ist gültig' wird zurückgesetzt.
FB_DALIV2ResetFunctionTestDoneFlag [▶ 242]	Das Flag 'Funktionstest abgeschlossen und Ergebnis ist gültig' wird zurückgesetzt.
FB_DALIV2ResetLampTime [▶ 243]	Die Variablen LAMP EMERGENCY TIME [▶ 377] und LAMP TOTAL OPERATION TIME [▶ 377]
FB_DALIV2Rest [▶ 244]	Schaltet die Lampe aus, wenn der Notbetrieb aktiv ist.
FB_DALIV2StartDurationTest [▶ 245]	Der Betriebsdauertest wird gestartet.
FB_DALIV2StartFunctionTest [▶ 246]	Der Funktionstest wird gestartet.
FB_DALIV2StopTest [▶ 247]	Stoppt jede Art von Funktionstest oder Betriebsdauertest.
FB_DALIV2StoreDTRAsDurationTestInterval [▶ 248]	Schreibt den Wert des DTR (Data Transfer Register) in die Variable DURATION TEST INTERVAL [▶ 376]
FB_DALIV2StoreDTRAsEmergencyLevel [▶ 249]	Schreibt den Wert des DTR (Data Transfer Register) in die Variable EMERGENCY LEVEL [▶ 375]
FB_DALIV2StoreDTRAsFunctionTestInterval [▶ 250]	Schreibt den Wert des DTR (Data Transfer Register) in die Variable FUNCTION TEST INTERVAL [▶ 376]
FB_DALIV2StoreDTRAsProlongTime [▶ 251]	Schreibt den Wert des DTR (Data Transfer Register) in die Variable PROLONG TIME [▶ 375]
FB_DALIV2StoreDTRAsTestDelayTimeHighByte [▶ 252]	Schreibt den Wert des DTR (Data Transfer Register) in das höherwertige Byte der Variablen TEST DELAY TIME [▶ 375]

Name	Beschreibung
FB_DALIV2StoreDTRAsTestDelayTimeLowByte [▶ 253]	Schreibt den Wert des DTR (Data Transfer Register) in das niederwertige Byte der Variablen TEST DELAY TIME
FB_DALIV2StoreDTRAsTestExecutionTimeout [▶ 254]	Schreibt den Wert des DTR (Data Transfer Register) in die Variable TEST EXECUTION TIMEOUT [▶ 376]



Diese Befehle gehören zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

5.1.2 Farb-/Farbtemperatursteuerung - Funktionsbausteine

Im Folgenden werden die Funktionsbausteine und Variablen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle beschrieben. Alle folgenden Funktionsbausteine rufen 'anwendungsbezogene Erweiterungsbefehle' auf. Laut DALI-Normung (IEC 62386) liegen diese Befehle im Bereich 224 bis 255. Da es verschiedene anwendungsbezogene Erweiterungsbefehle gibt, muss mit dem Funktionsbaustein `FB_DALIV2EnableDeviceType()` [▶ 356] festgelegt werden, welche Art von Vorschaltgeräten (Notbeleuchtung, Entladungslampen, ...) auf die Erweiterungsbefehle reagieren sollen. Eine genaue Beschreibung der einzelnen DALI-Kommandos und der Variablen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle finden Sie in der Norm IEC 62386 Part 209.



Name	Beschreibung
FB_DALIV2Activate [▶ 288]	Startet eine neue Überblendung. Eine laufende Überblendung wird zuvor beendet.
FB_DALIV2AssignColourToLinkedChannel [▶ 289]	Verknüpfte Ausgangskanäle werden der festgelegten Farbe zugeordnet.
FB_DALIV2ColourTemperatureTcStepCooler [▶ 291]	Der Wert <code>COLOUR TEMPERATURE Tc</code> [▶ 380] wird um 1 Mirek verringert.
FB_DALIV2ColourTemperatureTcStepWarmer [▶ 292]	Der Wert <code>COLOUR TEMPERATURE Tc</code> [▶ 380] wird um 1 Mirek erhöht.
FB_DALIV2CopyReportToTemporary [▶ 293]	Die Meldung Farbeinstellungen wird zu den temporären Farbeinstellungen kopiert.
FB_DALIV2QueryAssignedColour [▶ 294]	Liest die zugewiesene Farbe des vorgegebenen Ausgangskanals aus.
FB_DALIV2QueryColourStatus [▶ 296]	Die Variable <code>COLOUR STATUS</code> [▶ 385] wird aus dem Vorschaltgerät ausgelesen.
FB_DALIV2QueryColourTypeFeatures [▶ 297]	Die vom Vorschaltgerät unterstützten Farbdarstellungen werden ausgelesen.
FB_DALIV2QueryColourValue [▶ 298]	Der vorgegebene Farbwert wird aus dem Vorschaltgerät ausgelesen.
FB_DALIV2QueryGearFeaturesStatus [▶ 301]	Die Variable <code>GEAR FEATURES/STATUS</code> [▶ 385] wird aus dem Vorschaltgerät ausgelesen.
FB_DALIV2QueryRGBWAFControl [▶ 303]	Die Variable <code>RGBWAF CONTROL</code> [▶ 386] wird aus dem Vorschaltgerät ausgelesen.
FB_DALIV2SetTemporaryColourTemperatureTc [▶ 304]	Speichert den Wert in die Variable <code>TEMPORARY COLOUR TEMPERATURE Tc</code> [▶ 380] des Vorschaltgerätes.

Name	Beschreibung
FB_DALIV2SetTemporaryPrimaryNDimlevel [▶ 306]	Speichert den Wert in die Variable <u>TEMPORARY PRIMARY N DIMLEVEL</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2SetTemporaryRGBDimlevel [▶ 307]	Speichert die Werte in die Variablen <u>TEMPORARY RED DIMLEVEL</u> [▶ 380], <u>TEMPORARY GREEN DIMLEVEL</u> [▶ 380] und <u>TEMPORARY BLUE DIMLEVEL</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2SetTemporaryRGBWAFControl [▶ 308]	Speichert den Wert in die Variable <u>TEMPORARY RGBWAF CONTROL</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2SetTemporaryWAFDimlevel [▶ 310]	Speichert die Werte in die Variablen <u>TEMPORARY WHITE DIMLEVEL</u> [▶ 380], <u>TEMPORARY AMBER DIMLEVEL</u> [▶ 380] und <u>TEMPORARY FREECOLOUR DIMLEVEL</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2SetTemporaryXCoordinate [▶ 311]	Speichert den Wert in die Variable <u>TEMPORARY x-COORDINATE</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2SetTemporaryYCoordinate [▶ 312]	Speichert den Wert in die Variable <u>TEMPORARY y-COORDINATE</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2StartAutoCalibration [▶ 314]	Das Kalibrierungsverfahren wird gestartet um die x-Koordinate, die y-Koordinate und den TY-Wert aller unterstützen Primärfarben zu messen.
FB_DALIV2StoreColourTemperatureTcLimit [▶ 315]	Speichert den Wert in die Variable <u>COLOUR TEMPERATURE Tc COOLEST</u> [▶ 380], <u>COLOUR TEMPERATURE Tc WARMEST</u> [▶ 380], <u>COLOUR TEMPERATURE Tc PHYSICAL COOLEST</u> [▶ 380] oder <u>COLOUR TEMPERATURE Tc PHYSICAL WARMEST</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2StoreGearFeaturesStatus [▶ 317]	Speichert den Wert in die Variable <u>GEAR FEATURES/STATUS</u> [▶ 385] des Vorschaltgerätes.
FB_DALIV2StoreTYPrimaryN [▶ 318]	Speichert den Wert in die Variable <u>TY PRIMARY N</u> [▶ 380] des Vorschaltgerätes.
FB_DALIV2StoreXyCoordinatePrimaryN [▶ 320]	Kopiert den Wert aus den Variablen <u>TEMPORARY x-COORDINATE</u> [▶ 380] und <u>TEMPORARY y-COORDINATE</u> [▶ 380] in die Variablen <u>x-COORDINATE PRIMARY N</u> [▶ 380] und <u>y-COORDINATE PRIMARY N</u> [▶ 380].
FB_DALIV2XCoordinateStepDown [▶ 321]	Die Variable x-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung reduziert.
FB_DALIV2XCoordinateStepUp [▶ 322]	Die Variable x-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung erhöht.
FB_DALIV2YCoordinateStepDown [▶ 323]	Die Variable y-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung reduziert.
FB_DALIV2YCoordinateStepUp [▶ 324]	Die Variable y-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung erhöht.



Diese Befehle gehören zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl Enable Device Type 8 vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() gesendet werden kann. Ab Version 2.6.0 der Bibliotheken TcDALIV2 und TcDALIV2AppExtCmds jedoch wird der Befehl Enable Device Type 8 allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

5.1.3 Entladungslampen - Funktionsbausteine

Im Folgenden werden die Funktionsbausteine und Variablen für Entladungslampen mit DALI-Schnittstelle beschrieben. Alle folgenden Funktionsbausteine rufen 'anwendungsbezogene Erweiterungsbefehle' auf. Laut DALI-Normung (IEC 62386) liegen diese Befehle im Bereich 224 bis 255. Da es verschiedene anwendungsbezogene Erweiterungsbefehle gibt, muss mit dem Funktionsbaustein [FB_DALIV2EnableDeviceType\(\) \[▶ 356\]](#) festgelegt werden, welche Art von Vorschaltgeräten (Notbeleuchtung, Entladungslampen, ...) auf die Erweiterungsbefehle reagieren sollen. Eine genaue Beschreibung der einzelnen DALI-Kommandos und der Variablen für Entladungslampen mit DALI-Schnittstelle finden Sie in der Norm IEC 62386 Part 203.



Name	Beschreibung
FB_DALIV2QueryActualHIDFailure [▶ 255]	Die Variable ACTUAL HID FAILURE [▶ 379]
FB_DALIV2QueryHIDFeatures [▶ 256]	Die Variable HID FEATURES [▶ 380]
FB_DALIV2QueryHIDStatus [▶ 257]	Die Variable HID STATUS [▶ 379]
FB_DALIV2QueryStoredHIDFailure [▶ 258]	Die Variable STORED HID FAILURE [▶ 379]
FB_DALIV2QueryThermalLoad [▶ 259]	Die Variable THERMAL LOAD [▶ 380]
FB_DALIV2QueryThermalOverloadTime [▶ 260]	Die 16-Bit-Variable THERMAL OVERLOAD TIME [▶ 380]
FB_DALIV2QueryThermalOverloadTimeHB [▶ 261]	Das höherwertige Byte der 16-Bit-Variable THERMAL OVERLOAD TIME
FB_DALIV2QueryThermalOverloadTimeLB [▶ 262]	Das niederwertige Byte der 16-Bit-Variable THERMAL OVERLOAD TIME
FB_DALIV2ResetStoredHIDFailure [▶ 263]	Die Variable STORED HID FAILURE

i Diese Befehle gehören zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungslampen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\) \[▶ 356\]](#) gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungslampen intern automatisch vorangestellt.

5.1.4 LED-Module - Funktionsbausteine

Im Folgenden werden die Funktionsbausteine und Variablen für LED-Module mit DALI-Schnittstelle beschrieben. Alle folgenden Funktionsbausteine rufen 'anwendungsbezogene Erweiterungsbefehle' auf. Laut DALI-Normung (IEC 62386) liegen diese Befehle im Bereich 224 bis 255. Da es verschiedene anwendungsbezogene Erweiterungsbefehle gibt, muss mit dem Funktionsbaustein [FB_DALIV2EnableDeviceType\(\) \[▶ 356\]](#) festgelegt werden, welche Art von Vorschaltgeräten (Notbeleuchtung, Entladungslampen, ...) auf die Erweiterungsbefehle reagieren sollen. Eine genaue Beschreibung der einzelnen DALI-Kommandos und der Variablen für LED-Module mit DALI-Schnittstelle finden Sie in der Norm IEC 62386 Part 207.

Name	Beschreibung
FB_DALIV2DisableCurrentProtector [▶ 264]	Der Befehl sperrt die Stromschutzeinrichtung des Vorschaltgerätes.
FB_DALIV2EnableCurrentProtector [▶ 265]	Der Befehl gibt die Stromschutzeinrichtung des Vorschaltgerätes frei.
FB_DALIV2QueryCurrentProtectorActive [▶ 266]	Es wird abgefragt, ob die Stromschutzeinrichtung aktiv ist.
FB_DALIV2QueryCurrentProtectorEnabled [▶ 267]	Es wird abgefragt, ob die Stromschutzeinrichtung freigegeben ist.

Name	Beschreibung
FB_DALIV2QueryDimmingCurve [▶ 268]	Die Dimmkurve des Vorschaltgerätes wird ausgelesen.
FB_DALIV2QueryFastFadeTime [▶ 269]	Fragt den Wert von FAST FADE TIME ab.
FB_DALIV2QueryGearType [▶ 270]	Fragt den Wert von GEAR TYPE ab.
FB_DALIV2QueryLedFailureStatus [▶ 271]	Fragt den Wert von FAILURE STATUS ab.
FB_DALIV2QueryLedFeatures [▶ 272]	Fragt den Wert von FEATURES ab.
FB_DALIV2QueryLoadDecrease [▶ 273]	Es wird abgefragt, ob eine signifikante Lastabnahme (im Vergleich zur Referenzleistung des Systems) erkannt wurde.
FB_DALIV2QueryLoadIncrease [▶ 274]	Es wird abgefragt, ob ein signifikanter Lastanstieg (im Vergleich zur Referenzleistung des Systems) erkannt wurde.
FB_DALIV2QueryMinFastFadeTime [▶ 275]	Fragt den Wert von MIN FAST FADE TIME ab.
FB_DALIV2QueryOpenCircuit [▶ 277]	Es wird abgefragt, ob ein Leerlauf erkannt wurde.
FB_DALIV2QueryOperatingMode [▶ 278]	Fragt den Wert von OPERATING MODE ab.
FB_DALIV2QueryPossibleOperatingModes [▶ 279]	Fragt den Wert von POSSIBLE OPERATING MODE ab.
FB_DALIV2QueryReferenceMeasurementFailed [▶ 280]	Es wird abgefragt, ob eine gestartete Referenzmessung fehlgeschlagen ist.
FB_DALIV2QueryReferenceRunning [▶ 281]	Es wird abgefragt ob eine Referenzmessung der Systemleistung aktiv ist.
FB_DALIV2QueryShortCircuit [▶ 282]	Es wird abgefragt, ob ein Kurzschluss erkannt wurde.
FB_DALIV2QueryThermalOverload [▶ 283]	Es wird abgefragt ob eine thermische Überlast mit Reduzierung des Lichtstroms vorliegt.
FB_DALIV2QueryThermalShutDown [▶ 284]	Es wird abgefragt, ob eine thermische Abschaltung erfolgte.
FB_DALIV2ReferenceSystemPower [▶ 285]	Das Vorschaltgerät misst das Leistungsniveaus des Systems um Lastanstieg und Lastabnahme zu erkennen und speichert dieses ab.
FB_DALIV2SelectDimmingCurve [▶ 286]	Die Dimmkurve des Vorschaltgerätes wird ausgewählt.
FB_DALIV2SetFastFadeTime [▶ 287]	Setzt die FAST FADE TIME im Vorschaltgerät.



Diese Befehle gehören zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

5.1.5 Push button - Funktionsbausteine

Eine genaue Beschreibung der einzelnen DALI-Kommandos und der Variablen finden Sie in der Norm IEC 62386 Part 301.

Name	Beschreibung
FB_DALIV2x301QueryDoubleTimer [▶ 327]	Fragt den Wert des DOUBLE TIMER ab.
FB_DALIV2x301QueryDoubleTimerMin [▶ 328]	Fragt den minimalen Wert des DOUBLE TIMER ab.
FB_DALIV2x301QueryRepeatTimer [▶ 329]	Fragt den Wert des REPEAT TIMER ab.

Name	Beschreibung
FB_DALIV2x301QueryShortTimer [▶ 330]	Fragt den Wert des SHORT TIMER ab.
FB_DALIV2x301QueryShortTimerMin [▶ 331]	Fragt den minimalen Wert des SHORT TIMER ab.
FB_DALIV2x301QueryStuckTimer [▶ 332]	Fragt den Wert des STUCK TIMER ab.
FB_DALIV2x301SetDoubleTimer [▶ 333]	Setzt den Wert des DOUBLE TIMER.
FB_DALIV2x301SetRepeatTimer [▶ 335]	Setzt den Wert des REPEAT TIMER.
FB_DALIV2x301SetShortTimer [▶ 336]	Setzt den Wert des SHORT TIMER.
FB_DALIV2x301SetStuckTimer [▶ 337]	Setzt den Wert des STUCK TIMER.

5.1.6 Präsenzmelder - Funktionsbausteine

Eine genaue Beschreibung der einzelnen DALI-Kommandos und der Variablen finden Sie in der Norm IEC 62386 Part 303.

Name	Beschreibung
FB_DALIV2x303CancelHoldTimer [▶ 338]	Beendet vorzeitig den HOLD TIMER.
FB_DALIV2x303CatchMovement [▶ 339]	Nach Aufruf dieses Befehls wird nur einmalig ein Ereignis versendet, falls eine Bewegung erkannt wird.
FB_DALIV2x303QueryCatching [▶ 340]	Fragt ab, ob auf die Erkennung einer Bewegung gewartet wird.
FB_DALIV2x303QueryDeadtimeTimer [▶ 341]	Fragt den Wert des DEADTIME TIMER ab.
FB_DALIV2x303QueryHoldTimer [▶ 342]	Fragt den Wert des HOLD TIMER ab.
FB_DALIV2x303QueryReportTimer [▶ 343]	Fragt den Wert des REPORT TIMER ab.
FB_DALIV2x303SetDeadtimeTimer [▶ 344]	Setzt den Wert des DEADTIME TIMER.
FB_DALIV2x303SetHoldTimer [▶ 345]	Setzt den Wert des HOLD TIMER.
FB_DALIV2x303SetReportTimer [▶ 347]	Setzt den Wert des REPORT TIMER.

5.1.7 Interior Automation - Funktionsbausteine

Die Firma Interior Automation hat eigene DALI-Befehle für PIR Sensoren definiert. Diese Befehle gehen über die Möglichkeiten der DALI-Befehle nach Norm IEC 62386 hinaus. Für eine genauere Beschreibung der Befehle wenden Sie sich bitte an die Firma Interior Automation.

Name	Beschreibung
FB_DALIV2IAPIR [▶ 358]	Dieser Baustein liest zyklisch den Zustand eines IA PIR-Sensors aus und skaliert aus dem empfangenen Wert die gemessene Helligkeit und die erkannte Präsenz.
FB_DALIV2IAPIRPhysicalIndicatorOff	Deaktiviert die rote LED.
FB_DALIV2IAPIRPhysicalIndicatorOn	Aktiviert die rote LED.
FB_DALIV2IAPIRQueryExtendedVersion	Liest die Software-Versionsnummer aus.
FB_DALIV2IAPIRQueryFlags	Liest die Eigenschaften aus.
FB_DALIV2IAPIRQuerySensitivity	Liest die Sensorempfindlichkeit aus.
FB_DALIV2IAPIRQueryTimeout	Liest die Auszeit aus.
FB_DALIV2IAPIRStartIdentification	Lässt die grüne LED 10 Sekunden blinken.
FB_DALIV2IAPIRStoreDTRAsFlags	Speichert die Daten im Data Transfer Register (DTR) als Eigenschaften.
FB_DALIV2IAPIRStoreDTRAsSensitivity	Speichert die Daten im Data Transfer Register (DTR) als Sensorempfindlichkeit.

Name	Beschreibung
FB_DALIV2IAPIRStoreDTRAsTimeout	Speichert die Daten im Data Transfer Register (DTR) als Auszeit.



Diese Befehle gehören zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Geräte. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 128* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 128* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Geräte intern automatisch vorangestellt.

5.1.8 Helligkeitssensoren - Funktionsbausteine

Eine genaue Beschreibung der einzelnen DALI-Kommandos und der Variablen finden Sie in der Norm IEC 62386 Part 304.

Name	Beschreibung
<code>FB_DALIV2x304QueryDeadtimeTimer</code> [▶ 348]	Fragt den Wert des DEADTIME TIMER ab.
<code>FB_DALIV2x304QueryHysteresis</code> [▶ 349]	Fragt den Wert für die Hysterese ab.
<code>FB_DALIV2x304QueryHysteresisMin</code> [▶ 350]	Fragt den Wert für die minimal mögliche Hysterese ab.
<code>FB_DALIV2x304QueryReportTimer</code> [▶ 351]	Fragt den Wert des REPORT TIMER ab.
<code>FB_DALIV2x304SetDeadtimeTimer</code> [▶ 352]	Setzt den Wert des DEADTIME TIMER.
<code>FB_DALIV2x304SetHysteresis</code> [▶ 353]	Setzt den Wert für die Hysterese.
<code>FB_DALIV2x304SetHysteresisMin</code> [▶ 354]	Setzt den Wert für die minimal mögliche Hysterese.
<code>FB_DALIV2x304SetReportTimer</code> [▶ 355]	Setzt den Wert des REPORT TIMER.

5.1.9 B.E.G. - Funktionsbausteine

Die Firma B.E.G. hat eigene spezielle DALI-Befehle definiert. Diese Befehle gehen über die Möglichkeiten der DALI-Befehle nach Norm IEC 62386 hinaus. Allerdings sind diese Befehle nur für bestimmte Geräte der Firma B.E.G. anwendbar. Für eine genauere Beschreibung der Befehle wenden Sie sich bitte an die Firma B.E.G.. Des Weiteren bietet die Firma B.E.G. auch DALI-Multisensoren an, welche nach IEC 62386 Teil 103, 303 und 304 entwickelt worden sind. Die Geräte dieser Produktfamilie sind im Produktnamen mit „DALI-2-BMS“ gekennzeichnet.

Name	Beschreibung
<code>FB_DALIV2xBEGLuxomat</code> [▶ 360]	Dieser Baustein wertet die gemessene Helligkeit und die Präsenz des B.E.G. DALI-Steuergerätes aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergeräte zu initialisieren.

5.1.10 Osram - Funktionsbausteine

Die Firma Osram hat eigene spezielle DALI-Befehle definiert. Diese Befehle gehen über die Möglichkeiten der DALI-Befehle nach Norm IEC 62386 hinaus. Allerdings sind diese Befehle nur für bestimmte Geräte der Firma Osram anwendbar. Für eine genauere Beschreibung der Befehle wenden Sie sich bitte an die Firma Osram.

Name	Beschreibung
FB_DALIV2xOsramProfSensorCoupler [▶ 363]	Dieser Baustein wertet die gemessene Helligkeit und die Präsenz des Osram DALI Professional Sensor Couplers aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.
FB_DALIV2xOsramProfPushbuttonCoupler [▶ 361]	Dieser Baustein wertet den Zustand des digitalen Eingangs des Osram DALI Professional Pushbutton Couplers aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.
FB_DALIV2xOsramDisableSignalMode	Deaktiviert für einen Kanal den <i>Input Signal Mode</i> .
FB_DALIV2xOsramEnableSignalMode	Aktiviert für einen Kanal den <i>Input Signal Mode</i> .
FB_DALIV2xOsramQueryConfigurationId	Auslesen der Konfiguration für einen Kanal.
FB_DALIV2xOsramQueryInputDeviceType	Auslesen des Gerätetyps.
FB_DALIV2xOsramQueryInputValue	Auslesen des Eingangswertes eines Kanals.
FB_DALIV2xOsramQueryResolution	Fragt die Auflösung der Eingangswerte des Steuergerätes ab.
FB_DALIV2xOsramStoreConfigurationId	Schreibt die Konfiguration für einen Kanal.
FB_DALIV2xOsramIdentifySelectedDevice	Startet die Identifizierungs Routine für das selektierte (Zufallsadresse und Suchadresse sind gleich) Steuergerät.
FB_DALIV2xOsramQueryChannelSize	Liest die Anzahl der Kanäle aus, die das Steuergerät unterstützt.
FB_DALIV2xOsramQueryConfigurationFeature	Liest die möglichen Konfigurationswerte für einen Kanal aus.
FB_DALIV2xOsramQueryDeviceError	Fragt ab, ob das Steuergerät einen Fehler erkannt hat.
FB_DALIV2xOsramQueryStatus	Liest den Gerätestatus aus.
FB_DALIV2xOsramReadMemoryLocation	Es wird ein Byte aus dem Speicher des Steuergerätes ausgelesen.

5.1.11 Philips - Funktionsbausteine

Die Firma Philips hat eigene DALI-Befehle für spezielle Entladungslampen definiert. Diese Befehle gehen über die Möglichkeiten der DALI-Befehle nach Norm IEC 62386 Part 203 hinaus. Allerdings sind diese Befehle nur für bestimmte Vorschaltgeräte der Firma Philips anwendbar. Für eine genauere Beschreibung der Befehle wenden Sie sich bitte an die Firma Philips.

Name	Beschreibung
FB_DALIV2PhilipsChangePAEC	Aktiviert oder deaktiviert das "Application Extended Command Set".
FB_DALIV2PhilipsQueryCtrlGearOperationTime	Liest die Ausführungszeit des Vorschaltgerätes aus.
FB_DALIV2PhilipsQueryCtrlGearOvertempLevel	Liest den Übertemperatur Schwellwert des Vorschaltgerätes aus.
FB_DALIV2PhilipsQueryCtrlGearOvertempTime	Liest die Übertemperatur Zeit des Vorschaltgerätes aus.
FB_DALIV2PhilipsQueryCtrlGearTemperature	Liest die Temperatur des Vorschaltgerätes aus.
FB_DALIV2PhilipsQueryFailureStatus	Liest den Fehlerstatus aus.
FB_DALIV2PhilipsQueryHIDLampLevel	Liest den Lampenstatus aus.
FB_DALIV2PhilipsQueryHIDMaxFadeDownRate	Liest die maximale Ausblendrate aus.
FB_DALIV2PhilipsQueryHIDMaxFadeUpRate	Liest die maximale Einblendrate aus.
FB_DALIV2PhilipsQueryLampType	Liest den Lampentyp aus.
FB_DALIV2PhilipsQueryLampVoltage	Liest die Lampenspannung aus.
FB_DALIV2PhilipsQueryMainsVoltage	Liest die Hauptspannung aus.
FB_DALIV2PhilipsQueryTimeout	Liest die Auszeit aus.

Name	Beschreibung
FB_DALIV2PhilipsQueryUICByte	Liest ein Byte der UIC aus.
FB_DALIV2PhilipsSetDTRAsSegmentAddress	Speichert die Daten, die im Data Transfer Register (DTR) gehalten werden in das Segment Address Register.
FB_DALIV2PhilipsSetTestMode	Setzt das Vorschaltgerät in den Testmodus.
FB_DALIV2PhilipsStoreDTRAsLampType	Speichert die Daten im Data Transfer Register (DTR) als Lampentyp.
FB_DALIV2PhilipsStoreDTRAsOvertempLevel	Speichert die Daten im Data Transfer Register (DTR) als Übertemperatur Schwellwert.
FB_DALIV2PhilipsStoreDTRAsSegmentAddress	Speichert die Daten im Data Transfer Register (DTR) als Segment Address Register.
FB_DALIV2PhilipsStoreDTRAsTimeout	Speichert die Daten im Data Transfer Register (DTR) als Auszeit.



Diese Befehle gehören zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

5.1.12 Steinel - Funktionsbausteine

Die Firma Steinel hat eigene spezielle DALI-Befehle definiert. Diese Befehle gehen über die Möglichkeiten der DALI-Befehle nach Norm IEC 62386 hinaus. Allerdings sind diese Befehle nur für bestimmte Geräte der Firma Steinel anwendbar. Für eine genauere Beschreibung der Befehle wenden Sie sich bitte an die Firma Steinel.

Name	Beschreibung
FB_DALIV2xSteinelLiveLinkMotionSensor [▶ 364]	Dieser Baustein wertet die gemessene Helligkeit und die Präsenz des Steinel LiveLink DALI-Steuergerätes aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.
FB_DALIV2xSteinelSetBrightnessChangeLevel	Setzt den Wert für den <i>Brightness Change Level</i> .
FB_DALIV2xSteinelSetBrightnessChangeTimer	Setzt die Zeit für den <i>Brightness Change Timer</i> .
FB_DALIV2xSteinelSetEventFilter	Dieser Baustein setzt den Ereignis-Filter für die jeweilige Instanz des Steuergerätes.
FB_DALIV2xSteinelSetMotionDetectionRange	Stellt die Größe des Erfassungsbereichs ein.
FB_DALIV2xSteinelSetMotionTimer	Setzt die Zeit für den <i>Motion Timer</i> .
FB_DALIV2xSteinelSetMotionTimerRepeat	Setzt die Zeit für den <i>Motion Timer Repeat</i> .
FB_DALIV2xSteinelSetMotionVerificationLevel	Setzt die Empfindlichkeit für den Bewegungsmelder.
FB_DALIV2xSteinelSetSignalLedStatus	Schaltet die LEDs im Steuergerät.
FB_DALIV2xSteinelQueryBrightnessChangeLevel	Fragt den Wert für den <i>Brightness Change Level</i> ab.
FB_DALIV2xSteinelQueryBrightnessChangeTimer	Fragt die Zeit für den <i>Brightness Change Timer</i> ab.
FB_DALIV2xSteinelQueryEventFilter	Fragt den Ereignis-Filter für die jeweilige Instanz des Steuergerätes ab.

Name	Beschreibung
FB_DALIV2xSteinelQueryMotionDetectionRange	Fragt die Größe des Erfassungsbereichs ab.
FB_DALIV2xSteinelQueryMotionTimer	Fragt die Zeit des <i>Motion Timer</i> ab.
FB_DALIV2xSteinelQueryMotionTimerRepeat	Fragt die Zeit des <i>Motion Timer Repeat</i> ab.
FB_DALIV2xSteinelQueryMotionVerificationLevel	Fragt die Empfindlichkeit für den Bewegungsmelder ab.
FB_DALIV2xSteinelQuerySensorType	Fragt den Typ des Sensors ab.
FB_DALIV2xSteinelQuerySignalLedStatus	Fragt die LEDs im Steuergerät ab.

5.1.13 Tridonic - Funktionsbausteine

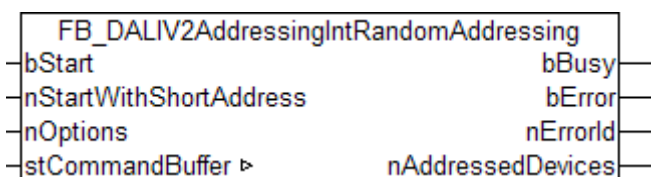
Name	Beschreibung
FB_DALIV2SmartSPOT [▶ 366]	Dieser Baustein liest zyklisch den Zustand eines smartSPOT-Sensors oder MSensor 02 aus und skaliert aus dem empfangenen Wert die gemessene Helligkeit und die erkannte Präsenz.

5.1.14 Theben HTS - Funktionsbausteine

Die Firma Theben hat eigene spezielle DALI-Befehle definiert. Diese Befehle gehen über die Möglichkeiten der DALI-Befehle nach Norm IEC 62386 hinaus. Allerdings sind diese Befehle nur für bestimmte Geräte der Firma Theben anwendbar. Für eine genauere Beschreibung der Befehle wenden Sie sich bitte an die Firma Theben.

Name	Beschreibung
FB_DALIV2xThebenPlanoSpot360 [▶ 367]	Dieser Baustein wertet die 3 gemessene Helligkeit und die Präsenz des ThebenHTS PlanoSpot DALI-Steuergerätes aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.

5.1.15 FB_DALIV2AddressingIntRandomAddressing



Dieser Funktionsbaustein adressiert die Vorschaltgeräte nach dem Zufallsprinzip. Der Anwender hat keinen Einfluss darauf, welches Vorschaltgerät welche Kurzadresse zugewiesen bekommt. Die Vergabe der Kurzadressen erfolgt aufsteigend.

Durch eine positive Flanke an dem Eingang `bStart` wird der Baustein gestartet und der Ausgang `bBusy` geht auf TRUE. Abhängig von den gewählten Optionen (Parameter `nOptions`) werden anschließend die Gruppenzugehörigkeit und die Szenen gelöscht. Die Klemme adressiert jetzt selbständig alle Vorschaltgeräte. Sind alle Vorschaltgeräte adressiert, so geht der Ausgang `bBusy` wieder auf FALSE. Die Ausgangsvariable `nAddressedDevices` gibt Auskunft darüber, wie viele Vorschaltgeräte eine Kurzadresse erhalten haben. Abhängig davon, wie viele Vorschaltgeräte angeschlossen sind, kann die Abarbeitung dieses Bausteines mehrere Minuten dauern. Dadurch, dass die Adressierung von der Klemme direkt ausgeführt wird, ist diese Methode etwas schneller als der Baustein

`FB_DALIV2AddressingRandomAddressing()`. [▶ 45] Allerdings liefert dieser Baustein keine Rückmeldungen während des Adressierens. Auch kann das Adressieren nicht vorzeitig abgebrochen werden.

Dieser Baustein kann nur ausgeführt werden, wenn die Firmware der Klemme den Stand 2A oder neuer hat.

VAR_INPUT

```
bStart          : BOOL;
nStartWithShortAddress : BYTE;
nOptions       : DWORD := DALIV2_OPTION_OPTICAL_FEEDBACK;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert und somit die Adressierung gestartet.

nStartWithShortAddress: Kurzadresse, die dem ersten Vorschaltgerät zugewiesen wird (0 ... 63).

nOptions: Optionen für die Adressierung der Vorschaltgeräte (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_COMPLETE_NEW_INSTALLATION	Es werden alle Vorschaltgeräte neu adressiert, auch die, die schon eine Kurzadresse haben.
DALIV2_OPTION_DELETE_ALL_GROUP_ASSIGNMENTS	Vor dem Adressieren werden bei allen Vorschaltgeräten, auch bei denen die evtl. nicht adressiert werden, die Gruppenzugehörigkeiten (siehe Variablen GROUP 0-7 und GROUP 8-15 [▶ 372])
DALIV2_OPTION_DELETE_ALL_SCENE_ASSIGNMENTS	Vor dem Adressieren werden bei allen Vorschaltgeräten, auch bei denen die evtl. nicht adressiert werden, die Szenen (siehe Variablen SCENE 0 [▶ 372] bis SCENE 15)
DALIV2_OPTION_OPTICAL_FEEDBACK	Vor dem Adressieren werden alle Vorschaltgeräte auf MIN LEVEL [▶ 371] gesetzt. Neu adressierte Vorschaltgeräte erhalten nach der Zuweisung der Kurzadresse die Helligkeit MAX LEVEL [▶ 371]

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nAddressedDevices : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

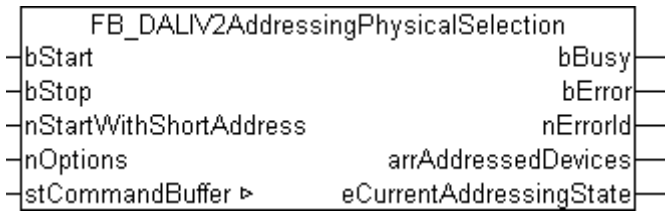
nAddressedDevices: Ist das Adressieren abgeschlossen (*bBusy* ist FALSE), so wird an diesem Ausgang die Anzahl der adressierten Vorschaltgeräte angezeigt.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [[▶ 95](#)] (KL6811) oder [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

5.1.16 FB_DALIV2AddressingPhysicalSelection



Dieser Funktionsbaustein adressiert die Vorschaltgeräte nach dem Adressierungsverfahren durch 'physikalische Auswahl'. Damit ist gemeint, dass die einzelnen Vorschaltgeräte durch Entfernen der Leuchtmittel selektiert und somit adressiert werden. Die Vergabe der Kurzadressen erfolgt aufsteigend, in der gleichen Reihenfolge wie die Leuchtmittel entfernt werden.

Durch eine positive Flanke an dem Eingang `bStart` wird der Baustein gestartet und der Ausgang `bBusy` geht auf TRUE. Abhängig von den gewählten Optionen (Parameter `nOptions`) werden anschließend die Gruppenzugehörigkeit und die Szenen gelöscht. Der Ausgang `eCurrentAddressingState` gibt dem Anwender den nächsten notwendigen Arbeitsschritt vor. Hierdurch wird z.B. vorgegeben ob bei dem nächsten Vorschaltgerät das Leuchtmittel entfernt, oder ob es wieder eingesetzt werden soll. Die Ausgangsvariable `arrAddressedDevices` gibt Auskunft darüber, welche Vorschaltgeräte schon eine Kurzadresse erhalten haben. Sind alle Vorschaltgeräte adressiert, so wird durch eine positive Flanke an dem Eingang `bStop` die Adressierung beendet und der Ausgang `bBusy` geht wieder auf FALSE.

VAR_INPUT

```

bStart          : BOOL;
bStop           : BOOL;
nStartWithShortAddress : BYTE := 0;
nOptions        : DWORD := DALIV2_OPTION_OPTICAL_FEEDBACK;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert und somit die Adressierung gestartet.

bStop: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit die Adressierung gestoppt.

nStartWithShortAddress: Kurzadresse, die dem ersten Vorschaltgerät zugewiesen wird (0 ... 63).

nOptions: Optionen für die Adressierung der Vorschaltgeräte (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_COMPLETE_NEW_INSTALLATION	Es werden alle Vorschaltgeräte neu adressiert, auch die, die schon eine Kurzadresse haben.
DALIV2_OPTION_DELETE_ALL_GROUP_ASSIGNMENTS	Vor dem Adressieren werden bei allen Vorschaltgeräten, auch bei denen die evtl. nicht adressiert werden, die Gruppenzugehörigkeiten (siehe Variablen GROUP 0-7 und GROUP 8-15 [▶ 372])
DALIV2_OPTION_DELETE_ALL_SCENE_ASSIGNMENTS	Vor dem Adressieren werden bei allen Vorschaltgeräten, auch bei denen die evtl. nicht adressiert werden, die Szenen (siehe Variablen SCENE 0 bis SCENE 15 [▶ 372])
DALIV2_OPTION_OPTICAL_FEEDBACK	Vor dem Adressieren werden alle Vorschaltgeräte auf MIN LEVEL [▶ 371] gesetzt. Neu adressierte Vorschaltgeräte erhalten nach der Zuweisung der Kurzadresse die Helligkeit MAX LEVEL [▶ 371]

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
arrAddressedDevices : ARRAY [0..63] OF BOOL;
eCurrentAddressingState : E_DALIV2CurrentAddressingState;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

arrAddressedDevices: Wird einem Vorschaltgerät eine Kurzadresse zugewiesen, so wird in dem Array das entsprechende Element auf TRUE gesetzt. Der Index spiegelt dabei die Kurzadresse des Vorschaltgerätes wider.

eCurrentAddressingState: Die Ausgangsvariable gibt den aktuellen Arbeitsschritt an (siehe Tabelle).

Element	Beschreibung
eDALIV2AddrStateIdle	Es findet keine Adressierung statt.
eDALIV2AddrStateRemoveLamp	Der Baustein wartet darauf, dass an einem Vorschaltgerät das Leuchtmittel entfernt wird.
eDALIV2AddrStateReinsertLamp	Der Baustein hat das Vorschaltgerät erkannt an dem das Leuchtmittel entfernt wurde (das Vorschaltgerät ist selektiert) und wartet jetzt darauf, dass es wieder eingesetzt wird.
eDALIV2AddrStateAddressingLamp	Das selektierte Vorschaltgerät wird adressiert.

VAR_IN_OUT

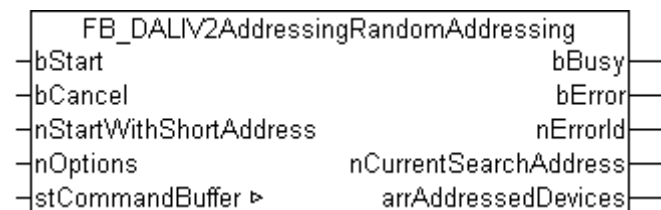
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Sehen Sie dazu auch

- 📖 [E_DALIV2CurrentAddressingState](#) [▶ 388]

5.1.17 FB_DALIV2AddressingRandomAddressing



Dieser Funktionsbaustein adressiert die Vorschaltgeräte nach dem Zufallsprinzip. Der Anwender hat keinen Einfluss darauf, welches Vorschaltgerät welche Kurzadresse zugewiesen bekommt. Die Vergabe der Kurzadressen erfolgt aufsteigend.

Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Abhängig von den gewählten Optionen (Parameter *nOptions*) werden anschließend die Gruppenzugehörigkeit und die Szenen gelöscht. Der Baustein adressiert jetzt selbständig alle Vorschaltgeräte. Die Ausgangsvariable *arrAddressedDevices* gibt Auskunft darüber, welche Vorschaltgerätes schon eine Kurzadresse erhalten haben. Sind alle Vorschaltgeräte adressiert, so geht der Ausgang *bBusy* wieder auf FALSE. Die Adressierung kann vorzeitig durch eine positive Flanke am Eingang

bCancel abgebrochen werden. Abhängig davon, wie viele Vorschaltgeräte angeschlossen sind, kann die Abarbeitung dieses Bausteines mehrere Minuten dauern.

VAR_INPUT

```
bStart          : BOOL;
bCancel         : BOOL;
nStartWithShortAddress : BYTE := 0;
nOptions       : DWORD := DALIV2_OPTION_OPTICAL_FEEDBACK;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert und somit die Adressierung gestartet.

bCancel: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit die Adressierung abgebrochen.

nStartWithShortAddress: Kurzadresse, die dem ersten Vorschaltgerät zugewiesen wird (0 ... 63).

nOptions: Optionen für die Adressierung der Vorschaltgeräte (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_COMPLETE_NEW_INSTALLATION	Es werden alle Vorschaltgeräte neu adressiert, auch die, die schon eine Kurzadresse haben.
DALIV2_OPTION_DELETE_ALL_GROUP_ASSIGNMENTS	Vor dem Adressieren werden bei allen Vorschaltgeräten, auch bei denen die evtl. nicht adressiert werden, die Gruppenzugehörigkeiten (siehe Variablen <u>GROUP 0-7</u> und <u>GROUP 8-15</u> [▶ 372])
DALIV2_OPTION_DELETE_ALL_SCENE_ASSIGNMENTS	Vor dem Adressieren werden bei allen Vorschaltgeräten, auch bei denen die evtl. nicht adressiert werden, die Szenen (siehe Variablen <u>SCENE 0 bis SCENE 15</u> [▶ 372])
DALIV2_OPTION_OPTICAL_FEEDBACK	Vor dem Adressieren werden alle Vorschaltgeräte auf <u>MIN LEVEL</u> [▶ 371] gesetzt. Neu adressierte Vorschaltgeräte erhalten nach der Zuweisung der Kurzadresse die Helligkeit <u>MAX LEVEL</u> [▶ 371]
DALIV2_OPTION_WITHOUT_RANDOMISE	Der Befehl <u>RANDOMISE</u> wird vor der Adressierung nicht aufgerufen. Hierdurch behalten alle Vorschaltgeräte ihre bisherige Zufallsadresse (<u>RANDOM ADDRESS</u> [▶ 372]) bei. Benutzen Sie diese Option nur, wenn es zwingend notwendig ist.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nCurrentSearchAddress : UDINT;
arrAddressedDevices : ARRAY [0..63] OF BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

nCurrentSearchAddress: Aktuelle Suchadresse (SEARCH ADDRESS [[▶ 372](#)]).

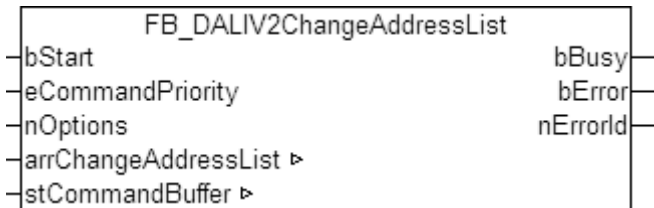
arrAddressedDevices: Wird einem Vorschaltgerät eine Kurzadresse zugewiesen, so wird in dem Array das entsprechende Element auf TRUE gesetzt. Der Index spiegelt dabei die Kurzadresse des Vorschaltgerätes wider.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.18 FB_DALIV2ChangeAddressList



Mit diesem Funktionsbaustein können die Kurzadressen mehrerer Vorschaltgeräte geändert werden. Im Gegensatz zum Baustein *FB_DALIV2SwapShortAddressList* ist es nicht notwendig, dass eine freie, unbenutzte Kurzadresse in der DALI-Line vorhanden ist.

In dem Array *arrChangeAddressList* vom Typ ST_DALIV2ChangeAddressList [▶ 389] wird eine Liste mit den Vorschaltgeräten übergeben, bei denen die Kurzadresse geändert werden soll. Die Liste hat 64 Einträge von 0 bis 63. Jeder Eintrag enthält eine Variable *nOldAddress* und *nNewAddress* mit dem die Adresszuordnung parametrisiert wird. Das Listen-Ende wird mit einem 255-Eintrag bei *nOldAddress* programmiert, so dass nicht unbedingt die ganze Liste ausgefüllt werden muss. Fehlt dieser Eintrag jedoch, so werden alle Einträge angenommen. Beim Starten des Bausteines (positive Flanke an *bStart*) wird zunächst das Listen-Ende anhand des beschriebenen Eintrages ermittelt und danach der gültige Listenbereich auf folgende Fehleinträge hin untersucht:

- Adresseinträge > 63
- doppelter Adress-Eintrag auf der Ursprungsseite *nOldAddress* (würde keinen Sinn machen)
- doppelter Adress-Eintrag auf der Zielseite *nNewAddress* (führt zu Doppelvergabe einer Adresse und damit zu Fehlern)

Der Baustein ermittelt dann anhand der Kurzadressen die internen Langadressen der DALI-Geräte und trägt diese jeweils in die Parameter *nRandomAddressHigh*, *nRandomAddressMiddle* und *nRandomAddressLow* der Liste ein. Tritt bei diesen Abfragen ein Fehler auf, so führt das zu einem Fehlereintrag für das jeweilige Gerät in dem Listenelement *nErrors* (siehe ST_DALIV2ChangeAddressList [▶ 389]). Der weitere Ablauf im Baustein hängt nun von der Option *DALIV2_OPTION_SAFE_ADDRESSING* (Eingang *nOptions*) ab. Ist sie gesetzt, so erfolgt eine sichere Neuadressierung: zunächst werden alle Kurzadressen der gewählten DALI-Geräte gelöscht. Danach werden an alle gewünschte Neuadressen Statusanfragen in die DALI-Linie geschickt. 2 Fälle sind nun möglich:

- Antwortet ein Gerät auf diese Anfrage, so ist diese gewünschte Neu-Adresse schon anderweitig vergeben. Die zuvor "gelöschten" DALI-Geräte werden mit ihren alten Adressen programmiert und eine Fehlermeldung ausgegeben.
- Antwortet kein Gerät auf diese Statusanfrage, so werden die zuvor "gelöschten" DALI-Geräte mit den gewünschten neuen Adressen belegt.

In beiden Fällen wird die Neuprogrammierung hinterher überprüft. Treten beim Löschen, bei der Statusabfrage oder bei der Neuprogrammierung Fehler auf, so führt dies zu einem Fehlereintrag für das jeweilige Gerät in dem Listenelement *nErrors* (siehe ST_DALIV2ChangeAddressList [▶ 389]).

Ist die Option `DALIV2_OPTION_SAFE_ADDRESSING` (Eingang `nOptions`) **nicht** gesetzt, so wird auf das Löschen der Kurzadressen sowie auf die Statusabfrage zum Vorhandensein gewünschter Neuadressen verzichtet und die neuen Adressen direkt programmiert. Dies ist dadurch möglich, dass die Programmierung über die zuvor ermittelte Langadresse erfolgt. Die Neuprogrammierung wird in diesem Falle nicht verifiziert.

Die einzelnen Bits im Listenelement `nErrors` haben folgende Bedeutung:

Bit	Fehler
0	Fehler beim Lesen des High-Bytes der Langadresse (<code>nRandomAddressHigh</code>).
1	Fehler beim Lesen des Middle-Bytes der Langadresse (<code>nRandomAddressMiddle</code>).
2	Fehler beim Lesen des Low-Bytes der Langadresse (<code>nRandomAddressLow</code>).
3	Fehler beim Löschen einer Kurzadresse.
4	Fehler beim Verifizieren einer Kurzadresse.
5	Fehler beim Programmieren einer Kurzadresse.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityHigh;
nOptions        : DWORD := 0;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOptions: Optionen für das Schreiben der Variablen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
<code>DALIV2_OPTION_SAFE_ADDRESSING</code>	Sichere Adressierung: Alte Kurz-Adressen werden gelöscht, die neuen auf schon Vorhandensein hin überprüft und die Neu-Programmierung verifiziert.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

VAR_IN_OUT

```
arrChangeAddressList : ARRAY [0.. 63] OF ST_DALIV2ChangeAddressList;
stCommandBuffer      : ST_DALIV2CommandBuffer;
```

arrChangeAddressList: Verweis auf die Liste [► 389] mit den zu ändernden Kurzadressen.

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

5.1.19 FB_DALIV2SwapShortAddress



Mit diesem Funktionsbaustein können die Kurzadressen zweier Vorschaltgeräte vertauscht werden. Hierzu ist es allerdings notwendig, dass in der DALI-Line eine freie, unbenutzte Kurzadresse vorhanden ist.

VAR_INPUT

```

bStart          : BOOL;
nShortAddress01 : BYTE;
nShortAddress02 : BYTE;
nFreeShortAddress : BYTE;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nShortAddress01: Kurzadresse vom ersten Vorschaltgerät (0 ... 63).

nShortAddress02: Kurzadresse vom zweiten Vorschaltgerät (0 ... 63).

nFreeShortAddress: Freie Kurzadresse (0 ... 63).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

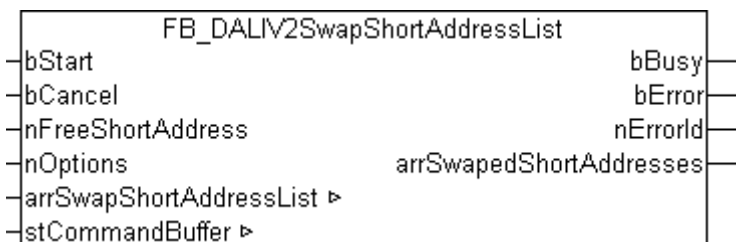
VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.20 FB_DALIV2SwapShortAddressList



Mit diesem Funktionsbaustein können die Kurzadressen mehrerer Vorschaltgeräte vertauscht werden. Hierzu ist es allerdings notwendig, dass in der DALI-Line eine freie, unbenutzte Kurzadresse vorhanden ist.

In dem Parameter *arrSwapShortAddressList* wird eine Liste mit den Vorschaltgeräten übergeben, bei denen die Kurzadresse geändert werden soll. Der Index der Struktur entspricht dabei der Kurzadresse des Vorschaltgerätes. Das Element *nNewShortAddress* enthält die neue Kurzadresse. *bShortAddressValid* muss auf TRUE gestestet werden, damit bei dem entsprechenden Vorschaltgerät die Kurzadresse geändert wird. Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Abhängig von den gewählten Optionen (Parameter *nOptions*) werden alle Lampen auf den Wert MIN LEVEL gesetzt. Die Elemente vom Ausgang *arrSwapedShortAddresses* werden zurückgesetzt. Wird bei einem Vorschaltgerät die neue Kurzadresse gesetzt, so wird bei dem Ausgang *arrSwapedShortAddresses* das entsprechende Element auf TRUE gesetzt. Die die Option DALIV2_OPTION_OPTICAL_FEEDBACK aktiv, so wird zusätzlich noch die Lampe auf den Wert MAX LEVEL geschaltet.

VAR_INPUT

```
bStart          : BOOL;
bCancel         : BOOL;
nFreeShortAddress : BYTE;
nOptions        : DWORD := DALIV2_OPTION_OPTICAL_FEEDBACK;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

bCancel: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit das Tauschen der Kurzadressen abgebrochen.

nFreeShortAddress: Freie Kurzadresse (0 ... 63).

nOptions: Optionen für das Vertauschen der Kurzadressen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_OPTICAL_FEEDBACK	Vor dem Vertauschen der Kurzadressen werden alle Vorschaltgeräte auf MIN LEVEL [▶ 371] gesetzt. Nach dem Zuweisen der neuen Kurzadresse wird die Helligkeit vom jeweiligen Vorschaltgerät auf MAX LEVEL

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
arrSwapedShortAddresses : ARRAY [0..63] OF BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

arrSwapedShortAddresses: Wird einem Vorschaltgerät eine Kurzadresse zugewiesen, so wird in dem Array das entsprechende Element auf TRUE gesetzt. Der Index spiegelt dabei die Kurzadresse des Vorschaltgerätes wider.

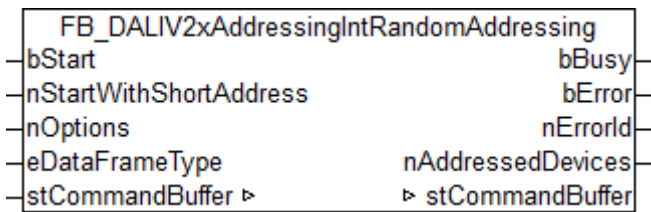
VAR_IN_OUT

```
arrSwapShortAddressList : ARRAY [0.. 63] OF ST_DALIV2SwapShortAddressList;
stCommandBuffer         : ST_DALIV2CommandBuffer;
```

arrSwapShortAddressList: Verweis auf die Liste [▶ 391] mit den zu vertauschenden Kurzadressen.

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.21 FB_DALIV2xAddressingIntRandomAddressing



Dieser Funktionsbaustein adressiert die Steuergeräte (Sensoren) nach dem Zufallsprinzip. Der Anwender hat keinen Einfluss darauf, welches Steuergerät welche Kurzadresse zugewiesen bekommt. Die Vergabe der Kurzadressen erfolgt aufsteigend.

Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Die Klemme adressiert jetzt selbständig alle Steuergeräte. Sind alle Steuergeräte adressiert, so geht der Ausgang *bBusy* wieder auf FALSE. Die Ausgangsvariable *nAddressedDevices* gibt Auskunft darüber, wie viele Steuergeräte eine Kurzadresse erhalten haben. Abhängig von der Anzahl der angeschlossenen Steuergeräte, kann die Abarbeitung dieses Bausteines mehrere Minuten dauern.

INPUT

```
bStart          : BOOL;
nStartWithShortAddress : BYTE := 0;
nOptions        : DWORD := DALIV2_OPTION_OPTICAL_FEEDBACK;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert und somit die Adressierung gestartet.

nStartWithShortAddress: Kurzadresse, die dem ersten Steuergerät zugewiesen wird (0 ... 63).

nOptions: Optionen für die Adressierung der Steuergeräte (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

Konstante	Beschreibung
DALIV2_OPTION_COMPLETE_NEW_INSTALLATION	Es werden alle Steuergeräte neu adressiert, auch die, die schon eine Kurzadresse haben.
DALIV2_OPTION_OPTICAL_FEEDBACK	Vor dem Adressieren werden alle Geräte auf <u>MIN LEVEL</u> [▶ 371] gesetzt. Neu adressierte Vorschaltgeräte erhalten nach der Zuweisung der Kurzadresse die Helligkeit MAX LEVEL

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nAddressedDevices : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nAddressedDevices: Ist das Adressieren abgeschlossen (*bBusy* ist FALSE), so wird an diesem Ausgang die Anzahl der adressierten Steuergeräte angezeigt.

VAR_IN_OUT

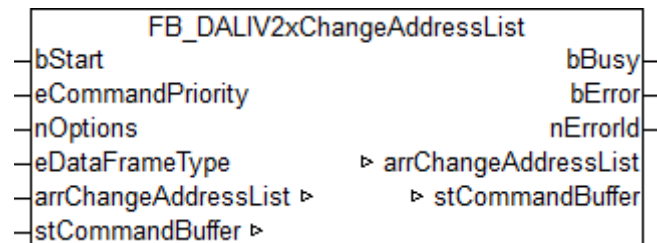
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [[104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.22 FB_DALIV2xChangeAddressList



Mit diesem Funktionsbaustein können die Kurzadressen mehrerer Steuergeräte geändert werden.

In dem Array `arrChangeAddressList` vom Typ `ST_DALIV2ChangeAddressList` [[389](#)] wird eine Liste mit den Steuergeräten übergeben, bei denen die Kurzadresse geändert werden soll. Die Liste hat 64 Einträge von 0 bis 63. Jeder Eintrag enthält eine Variable `nOldAddress` und `nNewAddress` mit dem die Adresszuordnung parametrisiert wird. Das Listen-Ende wird mit einem 255-Eintrag bei `nOldAddress` programmiert, so dass nicht unbedingt die ganze Liste ausgefüllt werden muss. Fehlt dieser Eintrag jedoch, so werden alle Einträge angenommen. Beim Start des Bausteines (positive Flanke an `bStart`) wird zunächst das Listen-Ende anhand des beschriebenen Eintrages ermittelt und danach der gültige Listenbereich auf folgende Fehleinträge hin untersucht:

- Adresseinträge > 63
- doppelter Adress-Eintrag auf der Ursprungsseite `nOldAddress` (würde keinen Sinn machen)
- doppelter Adress-Eintrag auf der Zielseite `nNewAddress` (führt zu Doppelvergabe einer Adresse und damit zu Fehlern)

Der Baustein ermittelt anhand der Kurzadressen die internen Langadressen der DALI-Geräte und trägt diese jeweils in die Parameter `nRandomAddressHigh`, `nRandomAddressMiddle` und `nRandomAddressLow` der Liste ein. Tritt bei diesen Abfragen ein Fehler auf, so führt das zu einem Fehlereintrag für das jeweilige Gerät in dem Listenelement `nErrors` (siehe `ST_DALIV2ChangeAddressList` [[389](#)]). Der weitere Ablauf im Baustein hängt nun von der Option `DALIV2_OPTION_SAFE_ADDRESSING` (Eingang `nOptions`) ab. Ist sie gesetzt, so erfolgt eine sichere Neuadressierung: zunächst werden alle Kurzadressen der gewählten DALI-Geräte gelöscht. Danach werden an alle gewünschte Neuadressen Statusanfragen in die DALI-Linie geschickt. 2 Fälle sind nun möglich:

- Antwortet ein Gerät auf diese Anfrage, so ist diese gewünschte Neu-Adresse schon anderweitig vergeben. Die zuvor "gelöschten" DALI-Geräte werden mit ihren alten Adressen programmiert und eine Fehlermeldung ausgegeben.
- Antwortet kein Gerät auf diese Statusanfrage, so werden die zuvor "gelöschten" DALI-Geräte mit den gewünschten neuen Adressen belegt.

In beiden Fällen wird die Neuprogrammierung hinterher überprüft. Treten beim Löschen, bei der Statusabfrage oder bei der Neuprogrammierung Fehler auf, so führt dies zu einem Fehlereintrag für das jeweilige Gerät in dem Listenelement `nErrors` (siehe `ST_DALIV2ChangeAddressList` [[389](#)]).

Ist die Option DALIV2_OPTION_SAFE_ADDRESSING (Eingang *nOptions*) **nicht** gesetzt, so wird auf das Löschen der Kurzadressen sowie auf die Statusabfrage zum Vorhandensein gewünschter Neuprogrammierungsadressen verzichtet und die neuen Adressen direkt programmiert. Dies ist dadurch möglich, dass die Programmierung über die zuvor ermittelte Langadresse erfolgt. Die Neuprogrammierung wird in diesem Falle nicht verifiziert.

Die einzelnen Bits im Listenelement *nErrors* haben folgende Bedeutung:

Bit	Fehler
0	Fehler beim Lesen des High-Bytes der Langadresse (<i>nRandomAddressHigh</i>).
1	Fehler beim Lesen des Middle-Bytes der Langadresse (<i>nRandomAddressMiddle</i>).
2	Fehler beim Lesen des Low-Bytes der Langadresse (<i>nRandomAddressLow</i>).
3	Fehler beim Löschen einer Kurzadresse.
4	Fehler beim Verifizieren einer Kurzadresse.
5	Fehler beim Programmieren einer Kurzadresse.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityHigh;
nOptions        : DWORD := 0;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOptions: Optionen für das Schreiben der Variablen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

Konstante	Beschreibung
DALIV2_OPTION_SAFE_ADDRESSING	Sichere Adressierung: Alte Kurz-Adressen werden gelöscht, die neuen auf schon Vorhandensein hin überprüft und die Neu-Programmierung verifiziert.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
arrChangeAddressList : ARRAY [0..63] OF ST_DALIV2ChangeAddressList;
stCommandBuffer       : ST_DALIV2CommandBuffer;
```

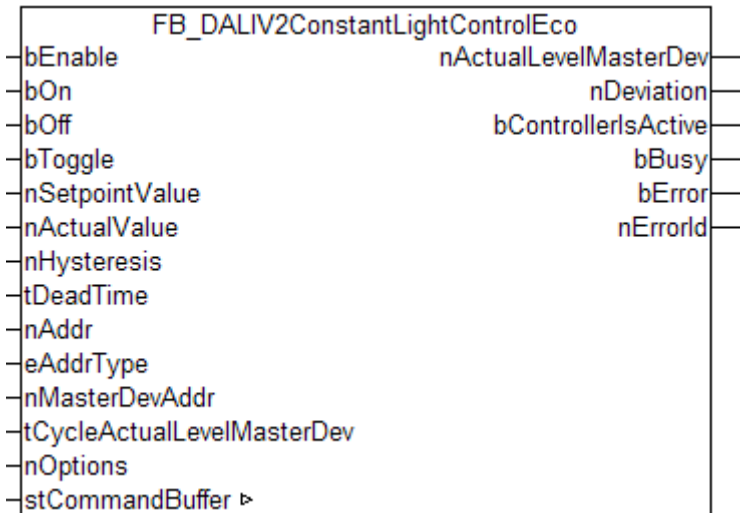
stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

arrChangeAddressList: Verweis auf die Liste [▶ 389] mit den zu ändernden Kurzadressen.

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.23 FB_DALIV2ConstantLightControlEco



Der Baustein `FB_DALIV2ConstantLightControlEco()` dient zur Konstantlichtregelung mit DALI-Vorschaltgeräten.

Durch zyklisches Auf- und Abdimmen wird versucht, auf einen vorgegebenen Sollwert zu regeln. Die Dynamik der Regelung wird durch eine Totzeit (`tDeadTime`) bestimmt. Die Totzeit legt die Wartezeit zwischen den einzelnen Befehlen zur Veränderung der Stellgröße fest. Je kleiner die Totzeit, desto schneller wird nachgeregelt. Eine frei definierbare Hysterese (`nHysteresis`) verhindert ein ständiges Schwingen um den Sollwert. Liegt der Istwert im Bereich der Hysterese um den Sollwert, so wird die Helligkeit der Lampen nicht verändert. Per Option kann festgelegt werden, ob die Lampen automatisch ein- und ausgeschaltet werden sollen (siehe Tabelle unten).

Parameter `nMasterDevAddr`

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Da die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, wird jeder Gruppe ein Referenzgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, `eAddrType = eAddrTypeShort`, so braucht kein Master-Device angegeben werden. Der Parameter `nMasterDevAddr` ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```

bEnable           : BOOL := TRUE;
bOn               : BOOL;
bOff              : BOOL;
bToggle          : BOOL;
nSetpointValue    : UINT := 500;
nActualValue      : UINT;
nHysteresis       : UINT := 50;
tDeadTime        : TIME := t#10s;
nAddr             : BYTE := 0;
eAddrType        : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr    : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
nOptions         : DWORD := 0;
    
```

bEnable: Schaltet den Baustein frei. Ist dieser Eingang auf FALSE, so sind die Eingänge *bOn*, *bOff* und *bToggle* gesperrt. Außerdem werden keine Stellgrößen ausgegeben.

bOn: Schaltet die angesprochenen Geräte auf MAX LEVEL [▶ 371] und aktiviert die Konstantlichtregelung.

bOff: Schaltet die angesprochenen Geräte aus und deaktiviert die Konstantlichtregelung.

bToggle: Je nach Zustand des Referenzgerätes wird die Beleuchtung ein- oder ausgeschaltet.

nSetpointValue: An diesem Eingang wird der Sollwert vorgegeben.

nActualValue: An diesem Eingang wird der Istwert angelegt.

nHysteresis: Regelhysterese um den Sollwert. Liegt der Istwert innerhalb dieses Bereiches, so werden die Stellgrößen der Lampen nicht verändert.

tDeadTime: Totzeit zwischen den einzelnen Befehlen, mit denen die Stellgröße der DALI-Lampen verändert wird.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nMasterDevAddr: Adresse des Master-Gerätes (Referenzgerät) bei Gruppen- und Sammelschaltungen.

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert (ACTUAL DIM LEVEL [▶ 370]) der Referenzlampe ausgelesen wird. Damit das Regeln der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt. Der ausgelesene Wert wird an dem Ausgang *nActualLevelMasterDev* ausgegeben.

nOptions: Optionen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_SWITCH_ON_AND_OFF	Es werden zum Verändern der Stellgröße die DALI-Befehle <u>ON AND STEP UP</u> [▶ 126] und <u>STEP DOWN AND OFF</u> [▶ 130] verwendet. Hierdurch werden die Lampen, nach dem Erreichen vom <u>MIN_LEVEL</u> <u>STEP UP</u> [▶ 130] und <u>STEP DOWN</u> [▶ 129] eingesetzt. Die Lampen bleiben dadurch immer eingeschaltet.
DALIV2_OPTION_SWITCH_ON_WITH_MIN_LEVEL	Wird durch die Konstandlichtregelung das Licht wieder eingeschaltet, so wird durch diese Option immer der Befehl <u>MIN_LEVEL</u> [▶ 371] verwendet. Ist die Option nicht gesetzt wird <u>MAX LEVEL</u> [▶ 371] aufgerufen. Diese Option steht ab V2.9.3 der SPS-Bibliothek zur Verfügung.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
nDeviation             : INT;
bControllerIsActive   : BOOL;
bBusy                  : BOOL;
bError                 : BOOL;
nErrorId               : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType = eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

nDeviation: Aktuelle Regelabweichung (Sollwert - Istwert).

bControllerIsActive: Dieser Ausgang wird gesetzt, sobald die Regelung aktiviert wurde.

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

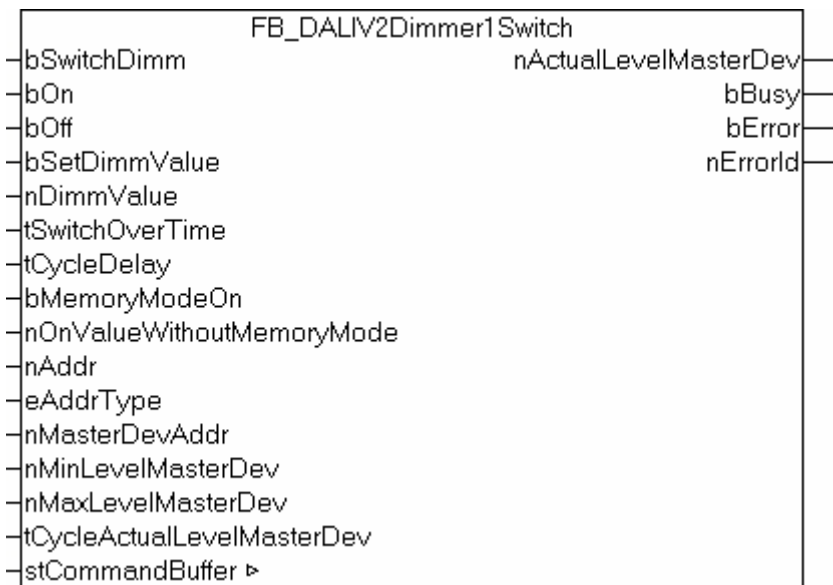
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.24 FB_DALIV2Dimmer1Switch



Eine einzelne DALI-Lampe, eine DALI-Gruppe oder eine komplette DALI-Line kann mit diesem Baustein über einen Taster geschaltet und gedimmt werden.

Bedienung über den Eingang bSwitchDimm

Durch ein kurzes Signal am Eingang *bSwitchDimm*, wird das Licht ein- oder ausgeschaltet. Liegt das Signal länger als *tSwitchOverTime* (empfohlener Richtwert: 200ms) an, so wird in den Dimmermodus umgeschaltet. Das Ausgangssignal fährt zyklisch zwischen *nMinLevelMasterDev* und *nMaxLevelMasterDev*. Um den maximalen oder minimalen Wert besser einstellen zu können, verweilt das Ausgangssignal für die Zeit *tCycleDelay* auf dem minimalen bzw. maximalen Wert. Wird das Signal wieder weggenommen, so bleibt das aktuelle Ausgangssignal anstehen. Durch einen erneuten Impuls auf den Eingang, wird der Ausgang auf 0 gesetzt. Wird im Dimmermodus das Signal *bSwitchDimm* kurzzeitig weggenommen, so ändert der Baustein seine Dimmrichtung.

Bedienung über die Eingänge bOn und bOff

Wird eine positive Flanke an den Eingängen *bOn* oder *bOff* angelegt, so wird das Licht direkt ein- oder ausgeschaltet. Beim Ausschalten wird der Ausgangswert auf 0 gesetzt. Das Verhalten beim Einschalten kann durch die Memoryfunktion beeinflusst werden (siehe unten).

Bedienung über die Eingänge bSetDimmValue und nDimmValue

Ändert sich der Wert *nDimmValue*, so werden die angesprochenen Geräte direkt auf diesen Helligkeitswert gesteuert. Wichtig ist hierbei, dass sich der Wert ändert. Durch eine Änderung auf den Wert 0, wird die Beleuchtung ausgeschaltet. Durch eine positive Flanke an den Eingang *bSetDimmValue* wird der Wert

nDimmValue direkt an den Ausgang ausgegeben. Das direkte Ändern des Ausgangs kann durch ein statisches 1-Signal am Eingang *bSetDimmValue* unterdrückt werden. Hierdurch kann ein Wert am Eingang *nDimmValue* angelegt werden, der erst bei der nächsten positiven Flanke, von *bSetDimmValue* an den Ausgang übergeben wird.

Mit Hilfe der Eingänge *bSetDimmValue* und *nDimmValue* können z.B. verschiedene Beleuchtungsszenarien realisiert werden. Das direkte Setzen des Ausgangs, mit Hilfe von *nDimmValue*, kann dazu benutzt werden, um bestimmte Helligkeiten anzufahren. Entweder direkt oder durch kontinuierliches Verändern des Wertes. *nDimmValue* muss einen Wert zwischen *nMinLevelMasterDev* und *nMaxLevelMasterDev* haben. Ausnahme ist der Wert 0. Liegt der Wert außerhalb des Bereichs, so wird der Ausgangswert auf die obere, bzw. untere Grenze begrenzt.

Memoryfunktion

Beim Einschalten muss unterschieden werden, ob die Memoryfunktion (Eingang *bMemoryModeOn*) aktiv ist oder nicht. Ist die Memoryfunktion aktiv, so wird beim Einschalten der zuletzt eingestellte Wert als Helligkeitswert übernommen. Ist die Memoryfunktion nicht aktiv, so wird den angesprochenen Geräten ein Helligkeitswert zugewiesen, der über den Parameter *nOnValueWithoutMemoryMode* vorgegeben wird. Hierbei ist es gleichgültig, ob das Licht über den Eingang *bOn* oder über den Eingang *bSwitchDimm* geschaltet wird. Zu beachten ist, dass der Parameter *nOnValueWithoutMemoryMode* zwischen *nMinLevelMasterDev* und *nMaxLevelMasterDev* liegen muss. Ist dieses nicht der Fall, so wird der Ausgangswert auf die untere bzw. obere Grenze angepasst.

Parameter *tSwitchOverTime*

Wird für den Parameter *tSwitchOverTime* eine Dauer von 0 vorgegeben, so kann mit dem Eingang *bSwitchDimm* das Licht nur gedimmt werden. Ein Ein-/Ausschalten ist nur mit den Eingängen *bOn* und *bOff* möglich.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Dadurch dass die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, werden jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType = eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```

bSwitchDimm      : BOOL;
bOn              : BOOL;
bOff             : BOOL;
bSetDimmValue    : BOOL;
nDimmValue       : BYTE;
tSwitchOverTime  : TIME := t#400ms;
tCycleDelay      : TIME := t#500ms;
bMemoryModeOn    : BOOL := FALSE;
nOnValueWithoutMemoryMode : BYTE := 254;
nAddr            : BYTE := 0;
eAddrType        : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr   : BYTE := 0;
nMinLevelMasterDev : BYTE := 126;
nMaxLevelMasterDev : BYTE := 254;
tCycleActualLevelMasterDev : TIME := t#0s;

```

bSwitchDimm: Schaltet oder dimmt die angesprochenen Geräte.

bOn: Schaltet die angesprochenen Geräte auf den letzten Ausgangswert oder auf den Wert *nOnValueWithoutMemoryMode*.

bOff: Schaltet die angesprochenen Geräte aus (Wert 0).

bSetDimmValue: Eine positive Flanke an diesem Eingang setzt die angesprochenen Geräte direkt auf den Helligkeitswert, der am Eingang *nDimmValue* ansteht. Ändert sich der Wert *nDimmValue*, so wird der Helligkeitswert direkt auf den geänderten Wert eingestellt, wenn der Eingang *bSetDimmValue* auf FALSE steht.

nDimmValue: siehe *bSetDimmValue*.

tSwitchOverTime: Umschaltzeit zwischen Licht ein/aus und Licht dimmen für den Eingang *bSwitchDimm*.

tCycleDelay: Wartezeit, wenn der Min- bzw. Max-Wert erreicht ist.

bMemoryModeOn: Schaltet auf Memoryfunktion um, so dass beim Einschalten der vorherige Wert an den Ausgang geschrieben wird.

nOnValueWithoutMemoryMode: Einschaltwert, wenn die Memoryfunktion nicht eingeschaltet ist.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

nMinLevelMasterDev: Minimalwert des Master-Gerätes.

nMaxLevelMasterDev: Maximalwert des Master-Gerätes.

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert ([ACTUAL DIM LEVEL](#) [► 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
bBusy                  : BOOL;
bError                 : BOOL;
nErrorId               : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType* = *eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer       : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.25 FB_DALIV2Dimmer1SwitchEco



Der Baustein FB_DALIDimmer1SwitchEco() stellt die speicherplatzsparende Variante des FB_DALIV2Dimmer1Switch() [► 56] dar. Er ist ohne die Sonderfunktion "Memoryfunktion ausschalten" ausgestattet.

Bedienung über den Eingang *bSwitchDimm*

Durch ein kurzes Signal am Eingang *bSwitchDimm*, wird das Licht ein- oder ausgeschaltet. Liegt das Signal länger als *tSwitchOverTime* (empfohlener Richtwert: 200ms) an, so wird in den Dimmermodus umgeschaltet, die Helligkeit erhöht, bzw. verringert sich gleichmäßig. Die Dimmrichtung wird durch kurzzeitiges Wegnehmen des Signals *bSwitchDimm* geändert.

Bedienung über die Eingänge *bOn* und *bOff*

Wird eine positive Flanke an den Eingängen *bOn* oder *bOff* angelegt, so wird das Licht direkt ein- oder ausgeschaltet. Beim Ausschalten wird der Ausgangswert auf 0 gesetzt.

Bedienung über die Eingänge *bSetDimmValue* und *nDimmValue*

Ändert sich der Wert *nDimmValue*, so werden die angesprochenen Geräte direkt auf diesen Helligkeitswert gesteuert. Wichtig ist hierbei, dass sich der Wert ändert. Durch eine Änderung auf den Wert 0, wird die Beleuchtung ausgeschaltet. Durch eine positive Flanke an den Eingang *bSetDimmValue* wird der Wert *nDimmValue* direkt an den Ausgang ausgegeben. Das direkte Ändern des Ausgangs kann durch ein statisches 1-Signal am Eingang *bSetDimmValue* unterdrückt werden. Hierdurch kann ein Wert am Eingang *nDimmValue* angelegt werden, der erst bei der nächsten positiven Flanke, von *bSetDimmValue* an den Ausgang übergeben wird.

Mit Hilfe der Eingänge *bSetDimmValue* und *nDimmValue* können z.B. verschiedene Beleuchtungsszenarien realisiert werden. Das direkte Setzen des Ausgangs, mit Hilfe von *nDimmValue*, kann dazu benutzt werden, um bestimmte Helligkeiten anzufahren. Entweder direkt oder durch kontinuierliches Verändern des Wertes.

Memoryfunktion

Im Gegensatz zum FB_DALIV2Dimmer1Switch() [► 56], bei der die Memoryfunktion über den Eingang *bMemoryModeOn* aktiviert oder ausgeschaltet werden kann, ist bei dieser speicherplatzsparenden Version die Memoryfunktion immer aktiv. Das bedeutet, dass beim Einschalten der zuletzt eingestellte Wert als Helligkeitswert übernommen wird. Hierbei ist es gleichgültig, ob das Licht über den Eingang *bOn* oder über den Eingang *bSwitchDimm* geschaltet wird.

Parameter *tSwitchOverTime*

Wird für den Parameter *tSwitchOverTime* eine Dauer von 0 vorgegeben, so kann mit dem Eingang *bSwitchDimm* das Licht nur gedimmt werden. Ein Ein-/Ausschalten ist nur mit den Eingängen *bOn* und *bOff* möglich.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Dadurch das die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, werden jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType = eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```
bSwitchDimm      : BOOL;
bOn              : BOOL;
bOff             : BOOL;
bSetDimmValue   : BOOL;
nDimmValue      : BYTE;
tSwitchOverTime : TIME := t#400ms;
nAddr           : BYTE := 0;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr  : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
```

bSwitchDimm: Schaltet oder dimmt die angesprochenen Geräte.

bOn: Schaltet die angesprochenen Geräte auf den letzten Ausgangswert.

bOff: Schaltet die angesprochenen Geräte aus (Wert 0).

bSetDimmValue: Eine positive Flanke an diesem Eingang setzt die angesprochenen Geräte direkt auf den Helligkeitswert, der am Eingang *nDimmValue* ansteht. Ändert sich der Wert *nDimmValue*, so wird der Helligkeitswert direkt auf den geänderten Wert eingestellt, wenn der Eingang *bSetDimmValue* auf FALSE steht.

nDimmValue: siehe *bSetDimmValue*.

tSwitchOverTime: Umschaltzeit zwischen Licht ein/aus und Licht dimmen für den Eingang *bSwitchDimm*.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert ([ACTUAL DIM LEVEL](#) [► 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
bBusy                 : BOOL;
bError                : BOOL;
nErrorId              : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType = eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

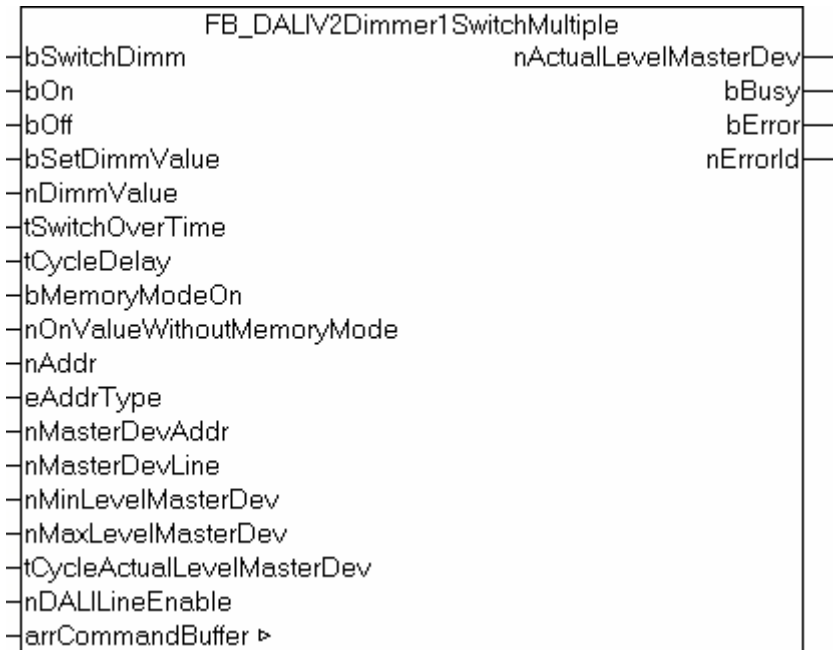
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

5.1.26 FB_DALIV2Dimmer1SwitchMultiple



Funktionsblock zum Schalten und Dimmen von DALI-Geräten mit einem Taster. Für Anwendungen bei denen bis zu fünf DALI-Linien (0..4) installiert sein können. Die Grundfunktion dieses Funktionsbausteines ist aus der Beschreibung des `FB_DALIV2Dimmer1Switch()` [► 56] zu entnehmen.

VAR_INPUT

```
bSwitchDimm : BOOL;
bOn : BOOL;
bOff : BOOL;
bSetDimmValue : BOOL;
nDimmValue : BYTE;
tSwitchOverTime : TIME := t#400ms;
tCycleDelay : TIME := t#500ms;
bMemoryModeOn : BOOL := FALSE;
nOnValueWithoutMemoryMode : BYTE := 254;
nAddr : BYTE := 0;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr : BYTE := 0;
nMasterDevLine : BYTE := 0;
nMinLevelMasterDev : BYTE := 126;
nMaxLevelMasterDev : BYTE := 254;
tCycleActualLevelMasterDev : TIME := t#0s;
nDALILineEnable : BYTE := 2#0000_0001;
```

bSwitchDimm: Schaltet oder dimmt die angesprochenen Geräte aller aktivierten DALI-Linien.

bOn: Schaltet die angesprochenen Geräte aller aktivierten DALI-Linien auf den letzten Ausgangswert oder auf den Wert `nOnValueWithoutMemoryMode`.

bOff: Schaltet die angesprochenen Geräte aller aktivierten DALI-Linien aus (Wert 0).

bSetDimmValue: Eine positive Flanke an diesem Eingang setzt die angesprochenen Geräte aller aktivierten DALI-Linien direkt auf den Helligkeitswert, der am Eingang `nDimmValue` ansteht. Ändert sich der Wert `nDimmValue`, so wird der Helligkeitswert direkt auf den geänderten Wert eingestellt, wenn der Eingang `bSetDimmValue` auf FALSE steht.

nDimmValue: siehe *bSetDimmValue*.

tSwitchOverTime: Umschaltzeit zwischen Licht ein/aus und Licht dimmen für den Eingang *bSwitchDimm*.

tCycleDelay: Wartezeit, wenn der Min- bzw. Max-Wert erreicht ist.

bMemoryModeOn: Schaltet auf Memoryfunktion um, so dass beim Einschalten der vorherige Wert an den Ausgang geschrieben wird.

nOnValueWithoutMemoryMode: Einschaltwert, wenn die Memoryfunktion nicht eingeschaltet ist.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType : Kurzadresse [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen (0 - 63).

nMasterDevLine: Linie, in der sich das Master-Gerät für Gruppen- und Sammelkonfiguration befindet. Je Funktionsbaustein gibt es nur ein Master-Gerät, das aus einer der aktivierten DALI-Linien zu wählen ist. Es gibt die Linien 0 bis 4.

nMinLevelMasterDev: Minimalwert des Master-Gerätes.

nMaxLevelMasterDev: Maximalwert des Master-Gerätes.

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert (ACTUAL DIM LEVEL [[▶ 370](#)]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

nDALILineEnable: Eingangsvariable in Form eines Bitmusters. Eine 1 im Bitmuster sagt aus, dass die DALI-Linie aktiviert ist. **Beispiel: 2#01001** dies bedeutet, dass die DALI-Linien **0** und **3** aktiviert sind.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
bBusy                 : BOOL;
bError                : BOOL;
nErrorId              : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType = eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

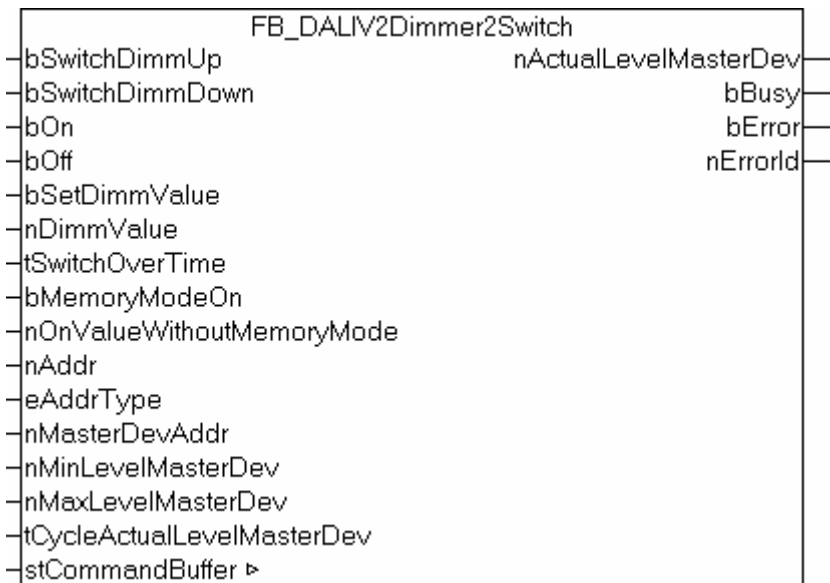
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

VAR_IN_OUT

```
arrCommandBuffer      : ARRAY [0..4] OF ST_DALIV2CommandBuffer;
```

arrCommandBuffer: Verweis auf die interne Strukturen zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [[▶ 95](#)] (KL6811) oder FB_KL6821Communication() [[▶ 104](#)] (KL6821).

5.1.27 FB_DALIV2Dimmer2Switch



Der Funktionsbaustein FB_DALIDimmer2Switch() entspricht im Funktionsumfang dem Funktionsbaustein FB_DALIV2Dimmer1Switch() [► 56]. Der Unterschied besteht darin, dass bei dem Funktionsbaustein FB_DALIDimmer2Switch() zwei Schalter angeschlossen werden. Hierdurch kann der Bediener gezielt auf- oder abdimmern.

Bedienung über die Eingänge *bSwitchDimmUp* und *bSwitchDimmDown*

Durch ein kurzes Signal am Eingang *bSwitchDimmUp* bzw. *bSwitchDimmDown*, wird das Licht ein- oder ausgeschaltet. Liegt das Signal länger als *tSwitchOverTime* (empfohlener Richtwert: 200ms) an, so wird in den Dimmermodus umgeschaltet. Die angesteuerten Geräte werden nun auf *nMaxLevelMasterDev* bzw. *nMinLevelMasterDev* gedimmt. Wird das Signal wieder weggenommen, so bleibt das aktuelle Ausgangssignal anstehen. Durch einen erneuten Impuls auf einen der Eingänge, wird der Ausgang auf 0 gesetzt.

Bedienung über die Eingänge *bOn* und *bOff*

Wird eine positive Flanke an den Eingängen *bOn* oder *bOff* angelegt, so wird das Licht direkt ein- oder ausgeschaltet. Beim Ausschalten wird der Ausgangswert auf 0 gesetzt. Das Verhalten beim Einschalten kann durch die Memoryfunktion beeinflusst werden (siehe unten).

Bedienung über die Eingänge *bSetDimmValue* und *nDimmValue*

Ändert sich der Wert *nDimmValue*, so werden die angesprochenen Geräte direkt auf diesen Helligkeitswert gesteuert. Wichtig ist hierbei, dass sich der Wert ändert. Durch eine Änderung auf den Wert 0, wird die Beleuchtung ausgeschaltet. Durch eine positive Flanke an den Eingang *bSetDimmValue* wird der Wert *nDimmValue* direkt an den Ausgang ausgegeben. Das direkte Ändern des Ausgangs kann durch ein statisches 1-Signal am Eingang *bSetDimmValue* unterdrückt werden. Hierdurch kann ein Wert am Eingang *nDimmValue* angelegt werden, der erst bei der nächsten positiven Flanke, von *bSetDimmValue* an den Ausgang übergeben wird.

Mit Hilfe der Eingänge *bSetDimmValue* und *nDimmValue* können z.B. verschiedene Beleuchtungsszenarien realisiert werden. Das direkte Setzen des Ausgangs, mit Hilfe von *nDimmValue*, kann dazu benutzt werden, um bestimmte Helligkeiten anzufahren. Entweder direkt oder durch kontinuierliches Verändern des Wertes. *nDimmValue* muss einen Wert zwischen *nMinLevelMasterDev* und *nMaxLevelMasterDev* haben. Ausnahme ist der Wert 0. Liegt der Wert außerhalb des Bereichs, so wird der Ausgangswert auf die obere, bzw. untere Grenze begrenzt.

Memoryfunktion

Beim Einschalten muss unterschieden werden, ob die Memoryfunktion (Eingang *bMemoryModeOn*) aktiv ist oder nicht. Ist die Memoryfunktion aktiv, so wird beim Einschalten der zuletzt eingestellte Wert als Helligkeitswert übernommen. Ist die Memoryfunktion nicht aktiv, so wird den angesprochenen Geräten ein Helligkeitswert zugewiesen, der über den Parameter *nOnValueWithoutMemoryMode* vorgegeben wird. Hierbei ist es gleichgültig, ob das Licht über den Eingang *bOn* oder über den Eingang *bSwitchDimmUp / bSwitchDimmDown* geschaltet wird. Zu beachten ist, dass der Parameter *nOnValueWithoutMemoryMode* zwischen *nMinLevelMasterDev* und *nMaxLevelMasterDev* liegen muss. Ist dieses nicht der Fall, so wird der Ausgangswert auf die untere bzw. obere Grenze angepasst.

Parameter *tSwitchOverTime*

Wird für den Parameter *tSwitchOverTime* eine Dauer von 0 vorgegeben, so kann mit dem Eingang *bSwitchDimmUp* bzw. *bSwitchDimmDown* das Licht nur gedimmt werden. Ein Ein-/Aussschalten ist nur mit den Eingängen *bOn* und *bOff* möglich.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Dadurch das die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, wird jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType = eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```

bSwitchDimmUp           : BOOL;
bSwitchDimmDown        : BOOL;
bOn                     : BOOL;
bOff                    : BOOL;
bSetDimmValue           : BOOL;
nDimmValue              : BYTE;
tSwitchOverTime        : TIME := t#400ms;
bMemoryModeOn          : BOOL := FALSE;
nOnValueWithoutMemoryMode : BYTE := 254;
nAddr                   : BYTE := 0;
eAddrType               : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr          : BYTE := 0;
nMinLevelMasterDev     : BYTE := 126;
nMaxLevelMasterDev     : BYTE := 254;
tCycleActualLevelMasterDev : TIME := t#0s;
    
```

bSwitchDimmUp: Schaltet oder dimmt die angesprochenen Geräte auf.

bSwitchDimmDown: Schaltet oder dimmt die angesprochenen Geräte ab.

bOn: Schaltet die angesprochenen Geräte auf den letzten Ausgangswert oder auf den Wert *nOnValueWithoutMemoryMode*.

bOff: Schaltet die angesprochenen Geräte aus (Wert 0).

bSetDimmValue: Eine positive Flanke an diesem Eingang setzt die angesprochenen Geräte direkt auf den Helligkeitswert, der am Eingang *nDimmValue* ansteht. Ändert sich der Wert *nDimmValue*, so wird der Helligkeitswert direkt auf den geänderten Wert eingestellt, wenn der Eingang *bSetDimmValue* auf FALSE steht.

nDimmValue: siehe *bSetDimmValue*.

tSwitchOverTime: Umschaltzeit zwischen Licht ein/aus und Licht dimmen für die Eingänge *bSwitchDimmUp* und *bSwitchDimDown*.

bMemoryModeOn: Schaltet auf Memoryfunktion um, so das beim Einschalten der vorherige Wert an den Ausgang geschrieben wird.

nOnValueWithoutMemoryMode: Einschaltwert, wenn die Memoryfunktion nicht eingeschaltet ist.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

nMinLevelMasterDev: Minimalwert des Master-Gerätes.

nMaxLevelMasterDev: Maximalwert des Master-Gerätes.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert ([ACTUAL DIM LEVEL](#) [► 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
bBusy                 : BOOL;
bError                : BOOL;
nErrorId              : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType* = *eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.28 FB_DALIV2Dimmer2SwitchEco



Der Baustein [FB_DALIDimmer2SwitchEco\(\)](#) stellt die speicherplatzsparende Variante des [FB_DALIV2Dimmer2Switch\(\)](#) [► 63] dar. Er ist ohne die Sonderfunktion "Memoryfunktion ausschalten" ausgestattet.

Bedienung über die Eingänge *bSwitchDimmUp* und *bSwitchDimmDown*

Durch ein kurzes Signal am Eingang *bSwitchDimmUp* bzw. *bSwitchDimmDown*, wird das Licht ein- oder ausgeschaltet. Liegt das Signal länger als *tSwitchOverTime* (empfohlener Richtwert: 200ms) an, so wird in den Dimmermodus umgeschaltet. Die angesteuerten Geräte werden nun gedimmt. Wird das Signal wieder weggenommen, so bleibt das aktuelle Ausgangssignal anstehen. Durch einen erneuten Impuls auf einen der Eingänge, wird der Ausgang auf 0 gesetzt.

Bedienung über die Eingänge *bOn* und *bOff*

Wird eine positive Flanke an den Eingängen *bOn* oder *bOff* angelegt, so wird das Licht direkt ein- oder ausgeschaltet. Beim Ausschalten wird der Ausgangswert auf 0 gesetzt.

Memoryfunktion

Im Gegensatz zum `FB_DALIV2Dimmer2Switch()` [► 63], bei der die Memoryfunktion über den Eingang *bMemoryModeOn* aktiviert oder ausgeschaltet werden kann, ist bei dieser speicherplatzsparenden Version die Memoryfunktion immer aktiv. Das bedeutet, dass beim Einschalten der zuletzt eingestellte Wert als Helligkeitswert übernommen wird. Hierbei ist es gleichgültig, ob das Licht über den Eingang *bOn* oder über den Eingang *bSwitchDimm* geschaltet wird.

Parameter *tSwitchOverTime*

Wird für den Parameter *tSwitchOverTime* eine Dauer von 0 vorgegeben, so kann mit dem Eingang *bSwitchDimmUp* bzw. *bSwitchDimmDown* das Licht nur gedimmt werden. Ein Ein-/Ausschalten ist nur mit den Eingängen *bOn* und *bOff* möglich.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Da die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, wird jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType = eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```

bSwitchDimmUp      : BOOL;
bSwitchDimmDown    : BOOL;
bOn                : BOOL;
bOff               : BOOL;
bSetDimmValue      : BOOL;
nDimmValue         : BYTE;
tSwitchOverTime    : TIME := t#400ms;
nAddr              : BYTE := 0;
eAddrType          : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr     : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
    
```

bSwitchDimmUp: Schaltet oder dimmt die angesprochenen Geräte auf.

bSwitchDimmDown: Schaltet oder dimmt die angesprochenen Geräte ab.

bOn: Schaltet die angesprochenen Geräte auf den letzten Ausgangswert.

bOff: Schaltet die angesprochenen Geräte aus (Wert 0).

bSetDimmValue: Eine positive Flanke an diesem Eingang setzt die angesprochenen Geräte direkt auf den Helligkeitswert, der am Eingang *nDimmValue* ansteht. Ändert sich der Wert *nDimmValue*, so wird der Helligkeitswert direkt auf den geänderten Wert eingestellt, wenn der Eingang *bSetDimmValue* auf FALSE steht.

nDimmValue: siehe *bSetDimmValue*.

tSwitchOverTime: Umschaltzeit zwischen Licht ein/aus und Licht dimmen für den Eingang *bSwitchDimm*.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert (ACTUAL DIM LEVEL [▶ 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
bBusy                  : BOOL;
bError                 : BOOL;
nErrorId               : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType = eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

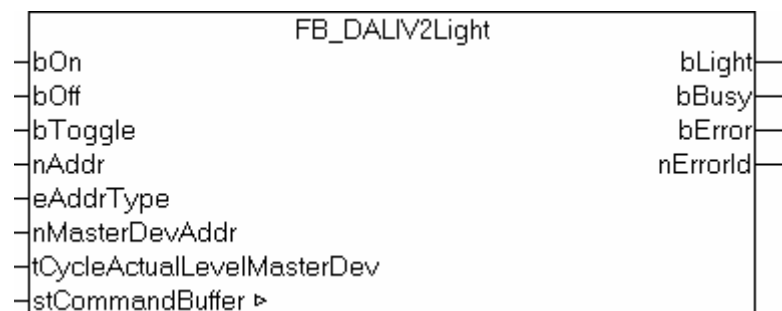
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer       : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.29 FB_DALIV2Light



Der Baustein FB_DALIV2Light() ist ein einfacher Baustein zum Ein- und Ausschalten von DALI-Lampen.

Bedienung

Durch eine positive Flanke an den Eingang *bOn* wird das Licht auf den Maximalwert (MAX LEVEL [▶ 371]) des Vorschaltgerätes gesetzt. Nach fehlerfreier Ausführung des Funktionsbausteines erhält der Ausgang *bLight* den Zustand TRUE. Durch eine positive Flanke an dem Eingang *bOff* wird das Licht ausgeschaltet und der Ausgang *bLight* wird auf FALSE gesetzt. Wird an *bToggle* eine positive Flanke angelegt, so liest der

Funktionsbaustein zunächst den aktuellen Lichtwert des Master-Gerätes aus und beurteilt dann, ob sich das Licht im Ein- oder Auszustand befindet. Je nach getroffener Aussage wird das Licht in den jeweiligen anderen Zustand versetzt, also von An nach Aus, bzw. von Aus nach An umgeschaltet.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Da die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände besitzen. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, wird jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType* = *eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```
bOn           : BOOL;
bOff          : BOOL;
bToggle      : BOOL;
nAddr        : BYTE := 0;
eAddrType    : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
```

bOn: Schaltet die angesprochenen Geräte ein (auf den Wert MAX LEVEL).

bOff: Schaltet die angesprochenen Geräte aus (auf den Wert 0).

bToggle: Negiert den Zustand der angesprochenen Geräte.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert (ACTUAL DIM LEVEL [▶ 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
bLight       : BOOL;
bBusy        : BOOL;
bError       : BOOL;
nErrorId     : UDINT;
```

bLight: Zustand der Lampe oder Gruppe nach dem Aufrufen des Bausteins.

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

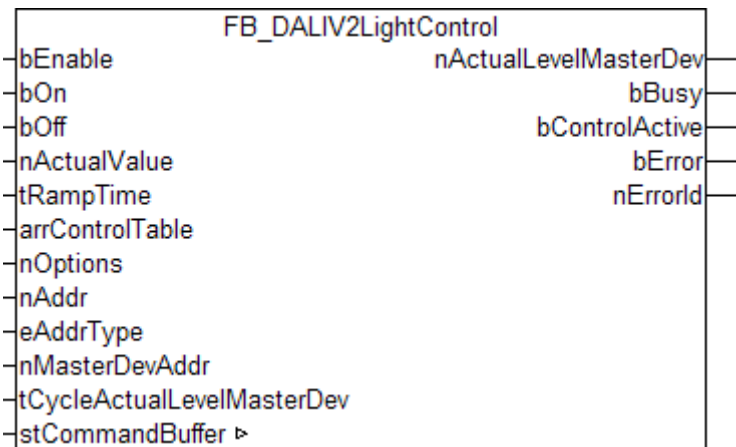
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

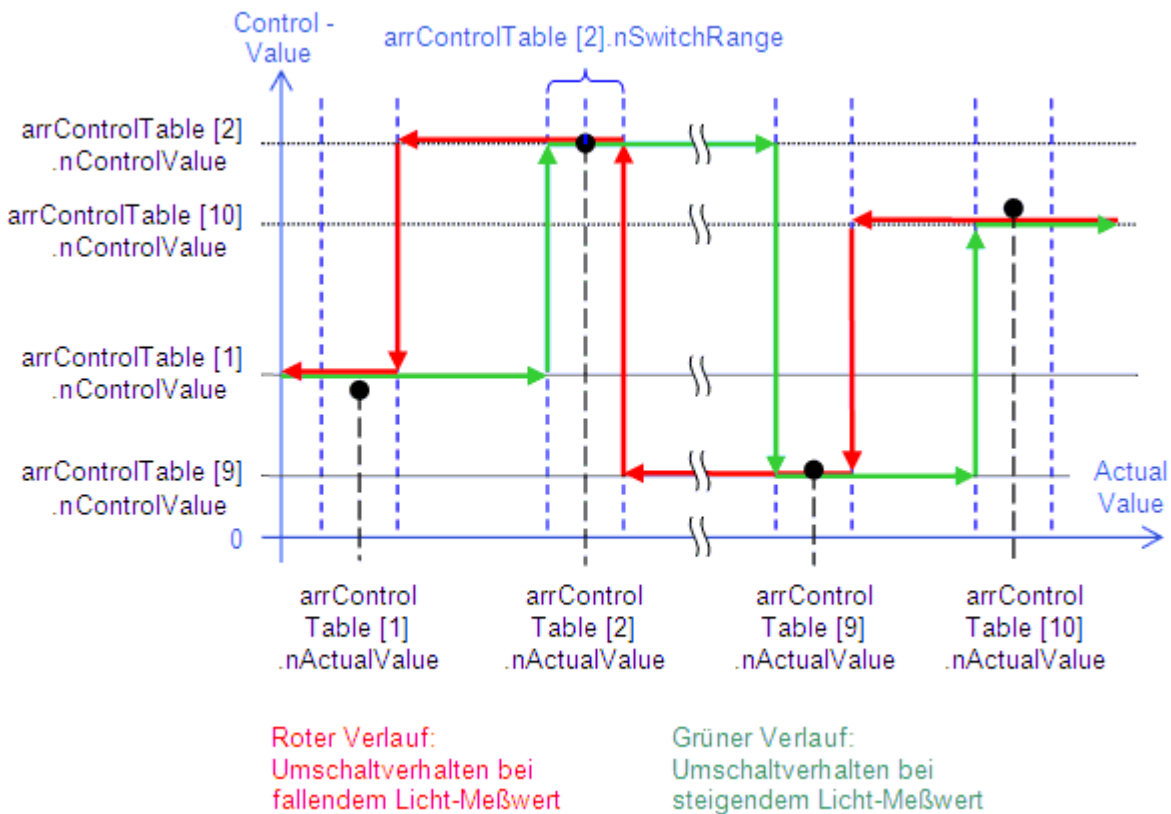
stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.30 FB_DALIV2LightControl



Funktionsblock zur tageslichtabhängigen Lichtsteuerung mit bis zu 30 Stützpunkten.

Kernstück dieses Funktionsblockes ist eine Eingangs-/Stellgrößen-Tabelle von 30 Elementen mit einer Schwellwertumschaltung. Erreicht der Istwert der Helligkeit den Bereich eines Stützpunktes ($arrControlTable[n].nActualValue - arrControlTable[n].nSwitchRange/2 \dots arrControlTable[n].nActualValue + arrControlTable[n].nSwitchRange/2$) so springt die Stellgröße auf den entsprechenden Wert $arrControlTable[n].nControlValue$ (siehe Diagramm). Daran angekoppelt ist ein Rampenbaustein, der die Stellgröße in der Zeit $tRampTime$ anfährt. Beim Einschalten mit einer positiven Flanke an bOn wird das Licht jedoch zunächst auf die nächstgelegene Stellgröße direkt geschaltet. Erst dann wird die Steuerung aktiviert. Während die Steuerung aktiv ist, kann jederzeit mit einer positiven Flanke an bOn "nachgestartet" und damit das Licht auf die nächstgelegene Stellgröße direkt gesteuert werden. Eine positive Flanke an $bOff$ schaltet alle angesteuerten Lampen direkt ab.



Es muss nicht der ganze Bereich der Tabelle genutzt werden. Das erste Tabellenelement, welches als Tabellenende $nSwitchRange$ eine 0 hat, wird als Beginn des unbenutzten Bereiches angesehen.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Da die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, wird jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType = eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```

bEnable          : BOOL := TRUE;
bOn              : BOOL;
bOff             : BOOL;
nActualValue     : UINT;
tRampTime        : TIME := t#30s;
arrControlTable  : ARRAY [1..30] OF ST_DALIV2ControlTable;
nOptions         : DWORD := 0;
nAddr            : BYTE := 0;
eAddrType        : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr   : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
    
```

bEnable: Solange dieser Eingang auf TRUE steht, sind die Eingänge *bOn* und *bOff* aktiv. Ein negativer Zustand deaktiviert die Eingänge und setzt den Baustein nach Abarbeitung der letzten notwendigen DALI-Befehle zurück. Es werden dann keine DALI-Befehle außer die zyklische Abfrage der Helligkeit des Mastergerätes ausgegeben.

bOn: Eine steigende Flanke schaltet die angesteuerten Lampen direkt auf die nächst gelegene Stellgröße.

bOff: Eine steigende Flanke schaltet die angesteuerten Lampen unmittelbar aus.

nActualValue: Iswert der Helligkeit.

tRampTime: Zeitdauer, in der die Stellgröße auf den nächsten Wert gesteuert wird (voreingestellter Wert: 30s).

arrControlTable: [Eingangs-/Stellgrößen-Tabelle](#) [▶ 389]. *arrControlTable[1]* bis *arrControlTable[30]* des Typen *ST_DALIV2ControlTable*.

nOptions: Reserviert für zukünftige Entwicklungen.

nAddr: Adresse des Einzelgerätes bei Einzelansteuerung oder der Gruppe bei Gruppenansteuerung.

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund die aktuelle Stellgröße ([ACTUAL DIM LEVEL](#) [▶ 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```

nActualLevelMasterDev : BYTE;
bBusy                 : BOOL;
bControlActive        : BOOL;
bError                : BOOL;
nErrorId              : UDINT;
    
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType = eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Dieser Ausgang ist immer dann aktiv, solange eine Befehlsabarbeitung (*bOn*, *bOff*, *bToggle* oder Rampenfahrt) aktiv ist.

bControlActive: Dieser Ausgang wird gesetzt, sobald die Lichtsteuerung aktiviert wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

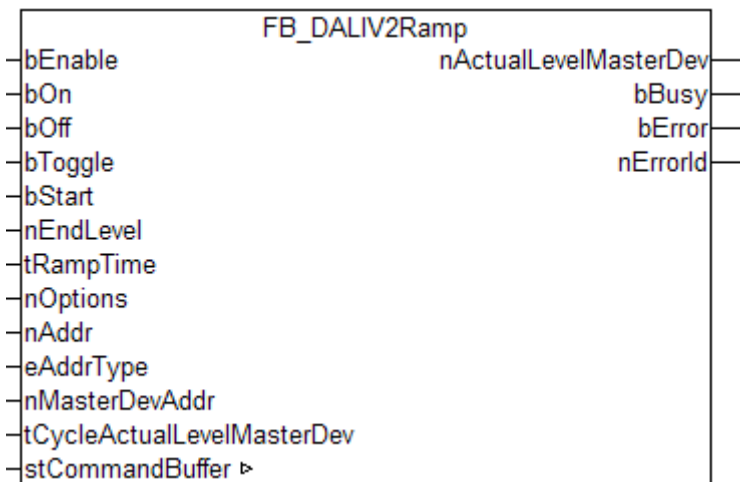
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.31 FB_DALIV2Ramp



Funktionsblock zur Realisierung einer Lichtrampe.

Durch eine steigende Flanke am Eingang *bOn* wird das Licht auf den Maximalwert der Master-Lampe geschaltet - eine steigende Flanke am Eingang *bOff* schaltet das Licht wieder aus. Steigende Flanken am Eingang *bToggle* invertieren den jeweiligen Lichtzustand. Eine positive Flanke am Eingang *bStart* lässt den Baustein das Licht vom aktuellen Wert auf *nEndLevel* dimmen. Die dazu benötigte Zeit wird durch *tRampTime* festgelegt. Alle Eingänge sind nur aktiv, solange *bEnable* auf *TRUE* steht, ansonsten wird der Baustein intern zurückgesetzt und keine weiteren DALI-Befehle mehr ausgegeben.

Bei jedem Start der Dimm-Rampe findet eine Überprüfung statt, ob der Wert *nEndLevel* innerhalb der zulässigen Grenzen ([MIN LEVEL \[► 371\]](#) ... [MAX LEVEL](#)) des Master-Gerätes liegt. Der Wert "0" ist ebenfalls erlaubt.

Rampenrealisierung

Grundsätzlich ist der Baustein so konzipiert, dass er innerhalb der vorgegebenen Rampenzeit die nötige Anzahl an *OnAndStepUp*- bzw. *StepDownAndOff*-Befehlen zeitlich gleichmäßig verteilt ausgibt.

Die Abarbeitung dieser Step- und Query-Befehle benötigt jedoch ihrerseits Zeit. Je weiter die Rampenzeit verringert wird, desto wahrscheinlicher wird die Tatsache, dass die intern berechnete Zeit für einen StepUp- bzw StepDown- Befehl nicht mehr ausreichend ist; die tatsächliche Rampenzeit wird im Vergleich zur eingestellten Zeit immer größer.

Um auch kleine Rampen realisieren zu können wird unterhalb eines intern voreingestellten Grenzwert von 11s Rampenzeit die Funktionsweise des Bausteines vom Schritt-Betrieb auf *DirectArcPower*-Betrieb umgeschaltet. Der DALI-Befehl *DirectArcPowerControl* steuert die entsprechenden Lampen von ihrem aktuellen Wert innerhalb der [FADE TIME \[► 371\]](#) auf den eingestellten Endwert. Insgesamt gibt es 16 verschiedene Fade-Time-Werte, die jeweils im AnVorschaltgerät einer jeden Lampe hinterlegt sind und folgende Werte annehmen können:

nFadeTime	tFadeTime (s)
0	<0,0707
1	0,707
2	1,000
3	1,414
4	2,000
5	2,828
6	4,000
7	5,657
8	8,000
9	11,314
10	16,000
11	22,627
12	32,000
13	45,255
14	64,000
15	90,510

Für die am Baustein eingestellte Rampe wird der nächste Wert aus der Tabelle genommen und in alle anzusteuern den Vorschaltgeräte einprogrammiert. Bei 6s wäre das beispielsweise Stufe 7 = 5,657s. Derselbe Wert würde ebenfalls für eine Rampenzeit von 5s genommen werden. Ein genaues Zeitverhalten ist somit leider nicht möglich. Nach erfolgreicher Beendigung der Rampenansteuerung wird der Wert in alle Vorschaltgeräte zurück übertragen, den das Mastergerät vor der Zeiteinstellung innehatte.

Während der Rampenfahrt wird der Ausgabewert *nActualLevelMasterDev* als intern berechneter Wert ausgegeben, um den DALI-Bus nicht mit ständigen Abfrage-Befehlen zu belasten. Dieser Berechnung liegen die Start-Endwert-Differenz sowie die gewählte Rampenzeit zugrunde. Da es sich um einen berechneten Wert handelt, kann er durch Rundungen und Befehlsverzögerungen mit Fehlern behaftet sein und soll nur zur Orientierung dienen. Nach Beendigung der Rampe jedoch wird die Helligkeit direkt abgefragt und der Ausgabewert ist wieder exakt.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Da die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, wird jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType = eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.

VAR_INPUT

```

bEnable           : BOOL := TRUE;
bOn               : BOOL;
bOff              : BOOL;
bToggle           : BOOL;
bStart            : BOOL;
nEndLevel         : BYTE := 254;
tRampTime         : TIME := t#8s;
nOptions          : DWORD := 0;
nAddr             : BYTE := 0;
eAddrType         : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr    : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
    
```

bEnable: Solange dieser Eingang auf TRUE steht, sind die Eingänge *bOn*, *bOff*, *bToggle* und *bStart* aktiv. Ein negativer Zustand deaktiviert die Eingänge und setzt den Baustein nach Abarbeitung der letzten notwendigen DALI-Befehle zurück. Es werden dann keine DALI-Befehle außer die zyklische Abfrage der Helligkeit des Mastergerätes ausgegeben.

bOn: Eine steigende Flanke schaltet die angesteuerten Lampen direkt auf den Maximalwert der Master-Lampe.

bOff: Eine steigende Flanke schaltet die angesteuerten Lampen unmittelbar aus.

bToggle: Eine steigende Flanke schaltet die angesteuerten Lampen unmittelbar aus, wenn die Master-Lampe **nicht** ausgeschaltet ist und umgekehrt ein, wenn die Master-Lampe aus ist.

bStart: Liegt an diesem Eingang eine steigende Flanke an, so wird das Licht vom derzeitigen Wert (ausschlaggebend ist hier das Master-Gerät) auf *nEndLevel* herauf bzw. herabgedimmt. Die dafür benötigte Zeit wird mit *tRampTime* festgelegt. Der Dimmvorgang kann jederzeit durch *bOn*, *bOff* oder *bToggle* unterbrochen werden.

nEndLevel: Zielwert des Dimmvorgangs. (Gültiger Wertebereich: 0 oder MIN LEVEL ... MAX LEVEL).

tRampTime: Rampenzeit, siehe *bStart*. (Voreingestellter Wert: 8 Sekunden).

nOptions: Reserviert für zukünftige Entwicklungen.

nAddr: Adresse des Einzelgerätes bei Einzelansteuerung oder der Gruppe bei Gruppenansteuerung.

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert (ACTUAL DIM LEVEL [► 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
bBusy                 : BOOL;
bError                : BOOL;
nErrorId              : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType* = *eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Dieser Ausgang ist immer dann aktiv, solange eine Befehlsabarbeitung (*bOn*, *bOff*, *bToggle* oder Rampenfahrt) aktiv ist.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

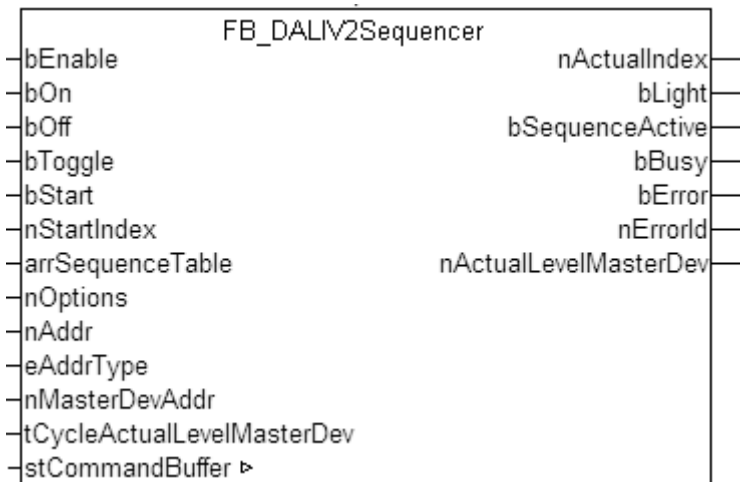
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
```

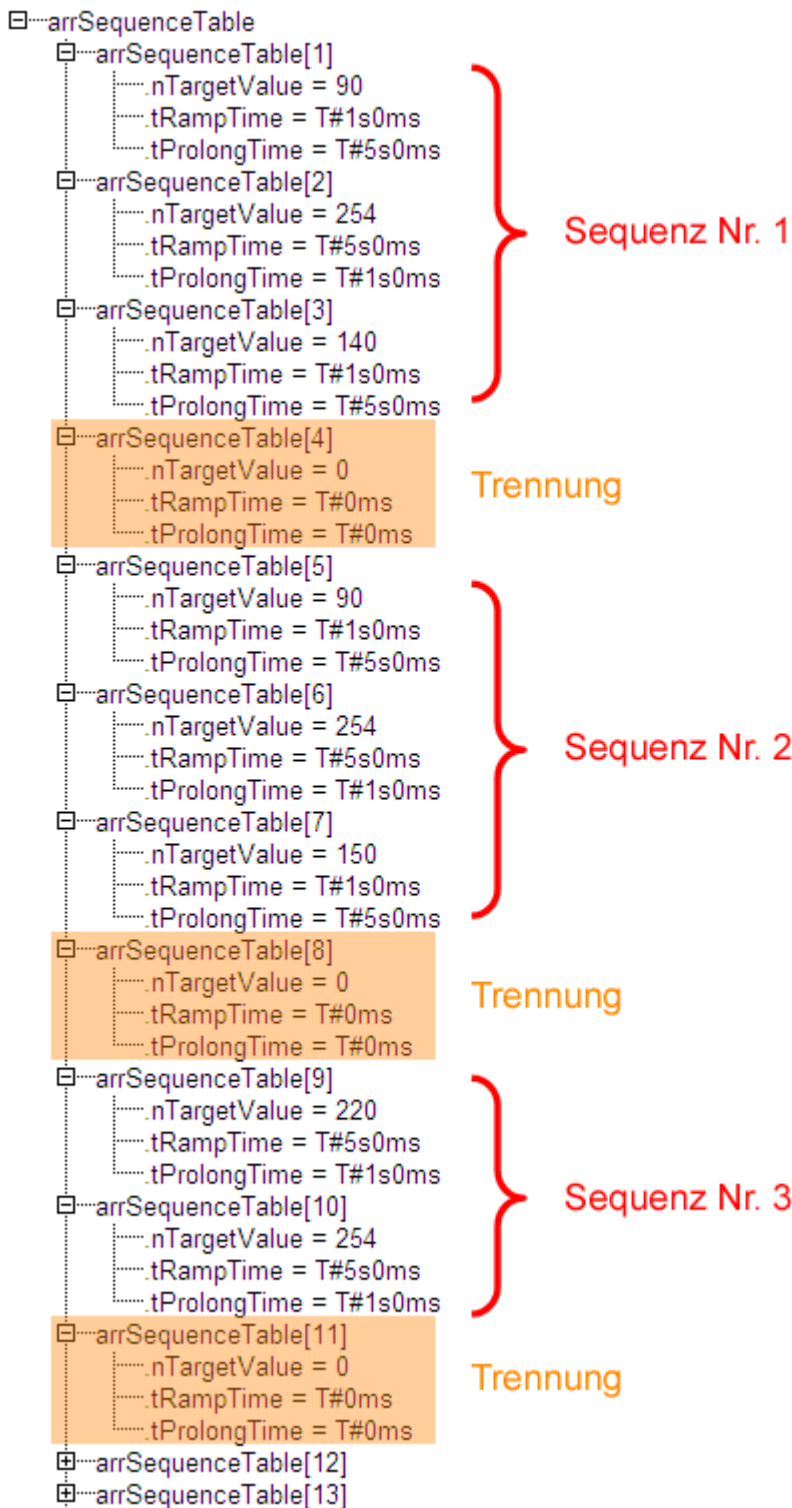
stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [► 95] (KL6811) oder FB_KL6821Communication() [► 104] (KL6821).

5.1.32 FB_DALIV2Sequencer

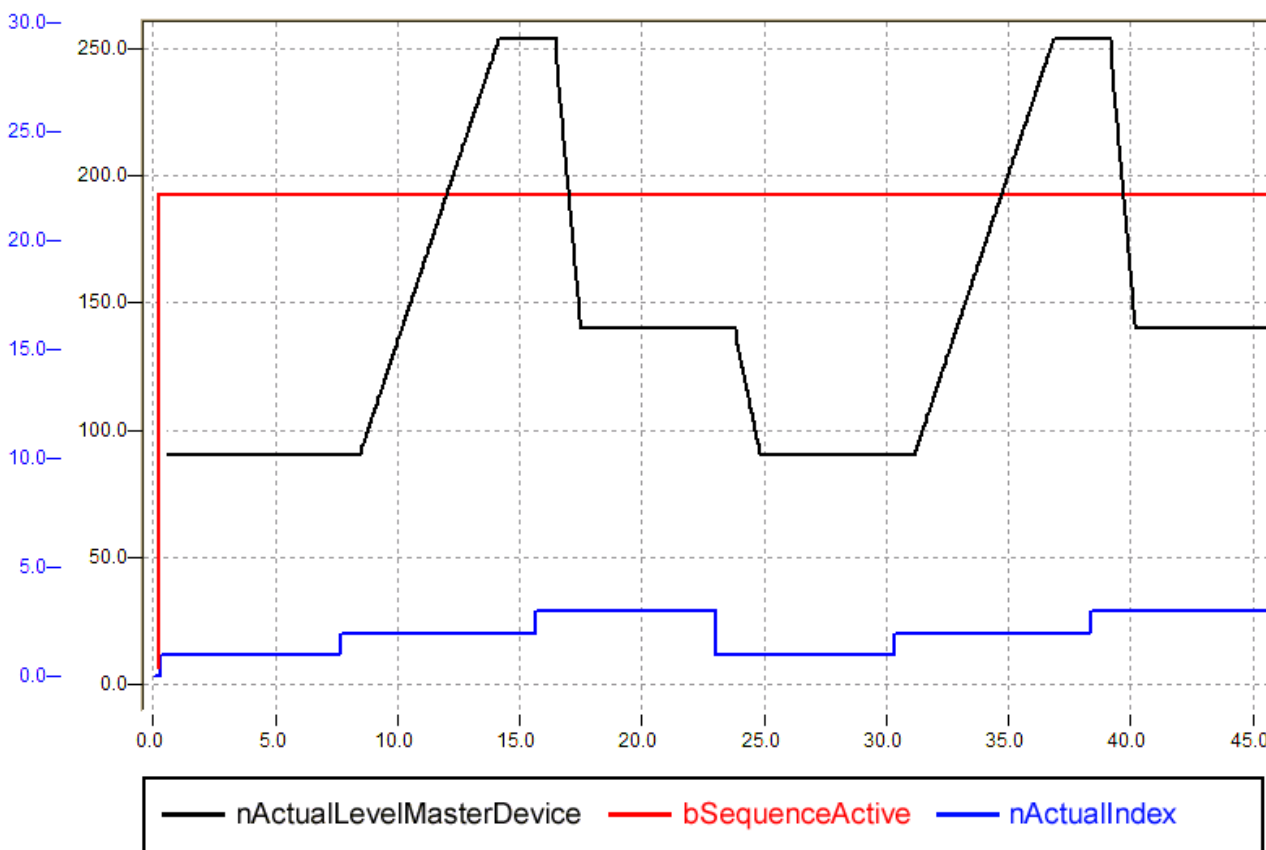


Funktionsblock zur Realisierung von Lichtsequenzen mit bis zu 50 Stützpunkten.

Kernstück dieses Funktionsblockes ist ein Rampenbaustein, der einzelne, in einer Tabelle definierte Helligkeitswerte in einer einstellbaren Zeit anfährt und auf diesem Helligkeitswert dann eine ebenfalls definierbare Zeit verweilt. Nach dem Verweilen wird dann der nächste Wert angefahren. Die Tabelle *arrSequenceTable* besteht, wie bereits erwähnt, aus 50 Einträgen mit den Werten für *nTargetValue* (Zielwert), *tRampTime* (Zeit zum Erreichen des Zielwertes) und *tProlongTime* (Verweilzeit auf dem Zielwert). Es ist nicht zwingend notwendig, alle 50 Werte zu nutzen. Ein 0-Eintrag aller 3 Werte markiert das Ende einer Sequenz. Darüber hinaus ist es mit dem Eingang *nStartIndex* möglich, eine Lichtsequenz an jeder beliebigen Stelle der Tabelle beginnen zu lassen. Dadurch lassen sich auch innerhalb der 50 Einträge mehrere verschiedene Lichtsequenzen programmieren, die untereinander jeweils durch 0-Eintrags-Elemente getrennt sind:



Sequenz 1 beispielsweise sieht im zeitlichen Verlauf folgendermaßen aus (*nStartIndex=1*, *nOptions.bit0=TRUE*, Erläuterung siehe unten):



Darüber hinaus lässt sich der Baustein "normal" ein- und ausschalten (Ein: Maximalwert der Lampen, Aus: 0) sowie über den Eingang *bToggle* zwischen "Ein" und "Aus" hin- und herschalten. Alle Befehlseingänge sind jedoch nur dann aktiv, wenn der Eingang *bEnable* auf *TRUE* steht. Wird er zurück auf *FALSE* gesetzt, so werden keine Befehle mehr angenommen und der Lichtwert behält seinen aktuellen Zustand - auch aus einer Rampenfahrt heraus.

i Dieser Baustein basiert, wie eingangs erwähnt, auf dem [FB_DALIV2Ramp](#) [► 71]. Der Rampenbaustein versucht die eingestellte Rampenzeit so genau wie möglich abzubilden. Dennoch ist es notwendig einmalig, wie zyklisch Daten aus dem DALI-Vorschaltgeräten abzufragen, was je nach eingestellter PLC-Zykluszeit unterschiedlich dauert. Daher sind Ungenauigkeiten in der Rampenzeit nicht ausgeschlossen.

VAR_INPUT

```

bEnable      : BOOL := TRUE;
bOn          : BOOL;
bOff         : BOOL;
bToggle     : BOOL;
bStart      : BOOL;
nStartIndex  : USINT := 0;
arrSequenceTable : ARRAY [1..nMaxSequenceValues] OF ST_DALIV2SequenceTable;
nOptions    : DWORD := 0;
nAddr       : BYTE := 0;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
    
```

bEnable: Solange dieser Eingang auf *TRUE* steht, sind die Eingänge *bOn*, *bOff*, *bToggle* und *bStart* aktiv. Ein negativer Zustand deaktiviert die Eingänge und setzt den Baustein zurück.

bOn: Eine steigende Flanke schaltet *nActualLevelMasterDevice* direkt auf den Maximalwert .

bOff: Eine steigende Flanke schaltet *nActualLevelMasterDevice* unmittelbar auf "0".

bToggle: Schaltet den Lichtzustand zwischen An (Maximalwert) und Aus (0) jeweils hin und her.

bStart: Eine positive Flanke startet eine Lichtsequenz ab dem unter *nStartIndex* definierten Anfang.

nStartIndex: Siehe *bStart*.

arrSequenceTable: Lichtwert-Tabelle [[▶ 391](#)] mit den dazugehörigen Rampen- und Verweilzeiten.

nOptions: Parametriereingang. Die Setzen (bzw. Nicht-Setzen) der einzelnen Bits dieser Variablen vom Typ *DWORD* hat folgende Wirkung:

Bit	Beschreibung
0	nicht gesetzt: Der Baustein stoppt nach Ablauf einer Sequenz seine Aktivität. Eine erneute positive Flanke an <i>bStart</i> wäre für einen Sequenz-Neustart nötig. gesetzt: Nach Ablauf einer Sequenz läuft der Baustein automatisch an der an <i>nStartIndex</i> definierten Stelle weiter.
1..31	-- reserviert für zukünftige Optionen --

nAddr: Adresse des Einzelgerätes bei Einzelansteuerung oder der Gruppe bei Gruppenansteuerung.

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

eAddrType: Kurzadresse [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert (ACTUAL DIM LEVEL [[▶ 370](#)]) ausgelesen wird. Damit das Ansteuern der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
nActualIndex          : USINT;
bLight                : BOOL;
bSequenceActive       : BOOL;
bBusy                  : BOOL;
bError                : BOOL;
nErrorId              : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType = eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

nActualIndex : Verweis auf das aktuelle Element in der Sequenz-Tabelle. Ist eine Sequenz beendet (*bSequenceActive = FALSE*, s.u.) wird dieser Ausgang zu "0".

bLight: Dieser Ausgang ist solange gesetzt, wie *nActualLevelMasterDev* größer als "0" ist.

bSequenceActive : Bei Abarbeitung einer Sequenz wird dieser Ausgang auf *TRUE* gesetzt.

bBusy: Dieser Ausgang ist immer dann aktiv, solange eine Befehlsabarbeitung (*bOn*, *bOff*, *bToggle* oder Rampenfahrt) aktiv ist.

bError: Dieser Ausgang wird auf *TRUE* geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf *FALSE* zurückgesetzt.

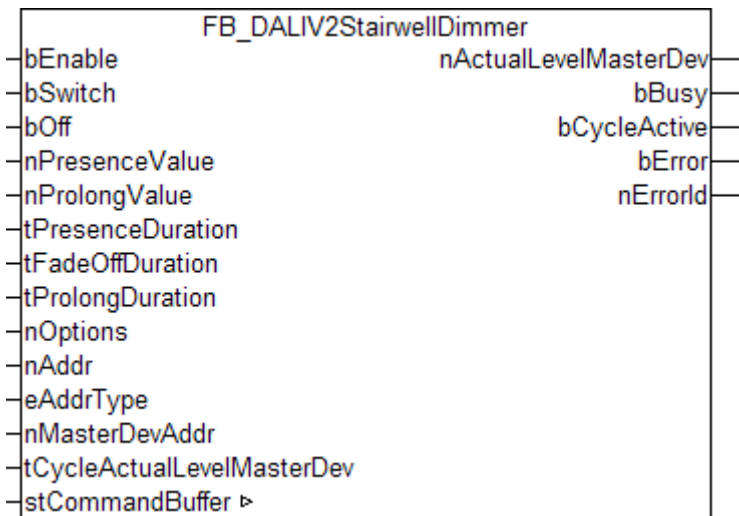
nErrorId: Enthält den spezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf "0" zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [[▶ 95](#)] (KL6811) oder FB_KL6821Communication() [[▶ 104](#)] (KL6821).

5.1.33 FB_DALIV2StairwellDimmer



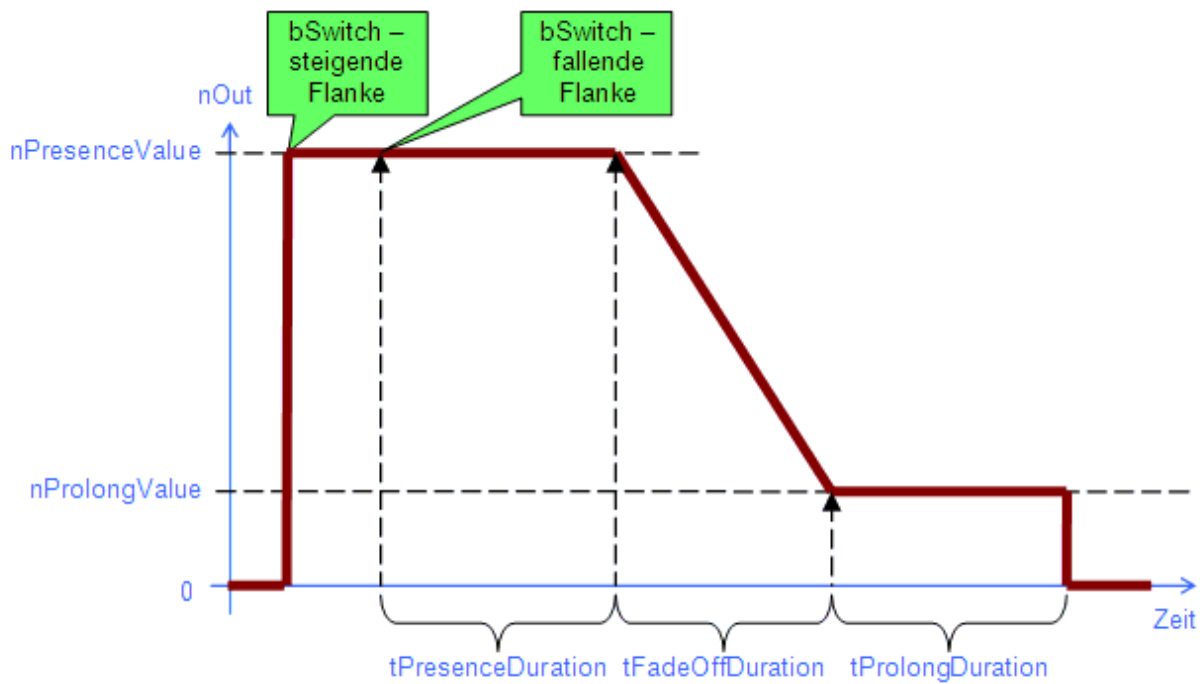
Funktionsblock zur Steuerung von Treppenhausbeleuchtungen.

Durch eine steigende Flanke am Eingang *bSwitch* wird das Licht auf den Wert *nPresenceValue* geschaltet. Durch eine fallende Flanke an *bSwitch* wird ein Timer mit der Laufzeit von *tPresenceDuration* gestartet. Nach Ablauf dieses Timers werden die angesteuerten Lampen innerhalb der Zeit *tFadeTime* auf den Wert *nProlongValue* gedimmt. Dieser Wert wird über den Zeitraum *tProlongDuration* beibehalten. Danach wird das Licht ausgeschaltet. Eine positive Flanke am Eingang *bOff* schaltet das Licht sofort aus, eine erneute steigende Flanke am Eingang *bSwitch* schaltet das Licht jederzeit - auch während der Dimm- und Wartezeiten - wieder ein.

Bei jedem Einschalten des Lichtes findet eine Überprüfung statt, ob die Werte *nPresenceValue* und *nProlongValue* innerhalb der zulässigen Grenzen (MIN LEVEL [▶ 371] ... MAX LEVEL) des Master-Gerätes liegen. Der Wert "0" ist ebenfalls erlaubt. Darüber hinaus ist es möglich, dass *nPresenceValue* kleiner als *nProlongValue* ist oder beide Werte identisch sind.

Parameter *nMasterDevAddr*

Das DALI-System bietet die Möglichkeit, Lampen nicht nur einzeln zu steuern, sondern diese auch gruppenweise oder mit Sammelbefehlen anzusprechen. Da die einzelnen Geräte Teilnehmer von verschiedenen Gruppen sein können, kann die Situation entstehen, dass vor einer Gruppen- bzw. Sammelansteuerung die einzelnen Lampen verschiedene Helligkeitszustände innehaben. Um dennoch eine Aussage darüber treffen zu können, ob die Lampen nun ein- oder ausgeschaltet werden sollen, wird jeder Gruppe ein Führungsgerät (Master-Device) zugeordnet, nach dessen Zustand sich die restlichen Geräte richten. Soll mit dem Baustein eine einzelne Lampe angesteuert werden, *eAddrType* = *eAddrTypeShort*, so braucht kein Master-Device angegeben werden. Der Parameter *nMasterDevAddr* ist in diesem Fall ohne Bedeutung.



VAR_INPUT

```

bEnable      : BOOL := TRUE;
bSwitch      : BOOL;
bOff         : BOOL;
nPresenceValue : BYTE := 254;
nProlongValue  : BYTE := 200;
tPresenceDuration : TIME := t#30s;
tFadeOffDuration  : TIME := t#10s;
tProlongDuration  : TIME := t#20s;
nOptions       : DWORD := 0;
nAddr          : BYTE := 0;
eAddrType      : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nMasterDevAddr : BYTE := 0;
tCycleActualLevelMasterDev : TIME := t#0s;
    
```

bEnable: Solange dieser Eingang auf TRUE steht, sind die Eingänge *bOn* und *bOff* aktiv. Ein negativer Zustand deaktiviert die Eingänge und setzt den Baustein nach Abarbeitung der letzten notwendigen DALI-Befehle zurück. Es werden dann keine DALI-Befehle außer die zyklische Abfrage der Helligkeit des Mastergerätes mehr ausgegeben.

bSwitch: Bei steigender Flanke: Die angesteuerten Lampen werden direkt auf *nPresenceValue* geschaltet. Bei fallender Flanke: Starten der Präsenzzeit (siehe Grafik).

bOff: Schaltet die angesteuerten Lampen unmittelbar aus.

nPresenceValue: Wert, auf den die angesteuerten Lampen während der Präsenzzeit geschaltet werden sollen. (Gültiger Wertebereich: 0 oder MIN LEVEL ... MAX LEVEL -voreingestellter Wert: 254).

nProlongValue: Wert, auf den die angesteuerten Lampen während der Verweilzeit geschaltet werden sollen. (Gültiger Wertebereich: 0 oder MIN LEVEL ... MAX LEVEL -voreingestellter Wert: 200).

tPresenceDuration: Dauer der Präsenzzeit, bei der die angesteuerten Lampen nach einer fallenden Flanke an *bSwitch* auf *nPresenceValue* geschaltet werden. (Voreingestellter Wert: 30 Sekunden).

tFadeOffDuration: Zeitdauer, in der der Helligkeitswert von *nPresenceValue* auf *nProlongValue* gesteuert wird. (Voreingestellter Wert: 10 Sekunden).

tProlongDuration: Dauer der Verweilzeit. (Voreingestellter Wert: 20 Sekunden).

nOptions: Reserviert für zukünftige Entwicklungen.

nAddr: Adresse des Einzelgerätes bei Einzelansteuerung oder der Gruppe bei Gruppenansteuerung.

nMasterDevAddr: Adresse des Master-Gerätes bei Gruppen- und Sammelschaltungen.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

tCycleActualLevelMasterDev: Zykluszeit, mit der im Hintergrund der aktuelle Istwert (ACTUAL DIM LEVEL [▶ 370]) ausgelesen wird. Damit das Dimmen der Lampen nicht gestört wird, wird das Auslesen immer in der niedrigsten Priorität durchgeführt. Wird der Wert auf 0 gesetzt, so wird das Auslesen gesperrt.

VAR_OUTPUT

```
nActualLevelMasterDev : BYTE;
bBusy                 : BOOL;
bCycleActive          : BOOL;
bError                : BOOL;
nErrorId              : UDINT;
```

nActualLevelMasterDev: Aktueller Ausgabewert des Master-Gerätes (bei *eAddrType* = *eAddrTypeShort* immer des jeweilig angesprochenen Gerätes).

bBusy: Dieser Ausgang ist immer dann gesetzt, wenn eine Lichtänderung aktiv ist, also Ein, Aus und Rampenfahrt. Dabei spielen Start- und Zielwert keine Rolle. Auch bei einer Rampenfahrt von z.B. 100 nach 100 in 10s würde dieser Ausgang gesetzt sein.

bCycleActive: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Zyklus abgearbeitet bzw. die Lampen ausgeschaltet wurden.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

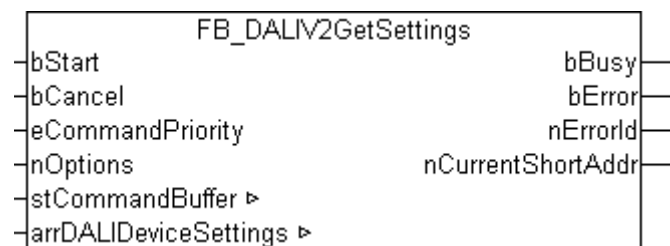
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.34 FB_DALIV2GetSettings



Dieser Baustein liest von allen Vorschaltgeräten einer DALI-Line die Variablen (MIN LEVEL, MAX LEVEL, FADE TIME, ...) aus, und speichert diese in einer Struktur vom Typ ST_DALIV2DeviceSettings [▶ 389].

Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Zunächst wird geprüft, ob ein Vorschaltgerät überhaupt vorhanden ist. Ist dies der Fall, so wird in der jeweiligen Struktur (siehe ST_DALIV2DeviceSettings [▶ 389]) das Bit *bPresent* gesetzt und anschließend nacheinander die Einstellungen des Vorschaltgerätes ausgelesen und in die betreffenden Variablen der Struktur geschrieben. Wird ein Gerät als nicht vorhanden erkannt, so wird das Auslesen übersprungen und mit dem nächsten Gerät begonnen. Der Index der Struktur spiegelt dabei die Adresse des Gerätes wider, d.h. vom Gerät mit der Kurzadresse 0 liegen die Daten in *arrDALIDeviceSettings[0]* bis hin zu dem Gerät mit der Kurzadresse 63 in *arrDALIDeviceSettings[63]*. Tritt während des Auslesevorganges ein Lesefehler bei einem Gerät auf, so wird das entsprechende Bit in *nErrors* der jeweiligen Struktur gesetzt, ohne dass jedoch der Funktionsbaustein an sich in Störung geht. In der folgenden Tabelle wird aufgelistet, welches Bit in der Variablen *nErrors* gesetzt wird, wenn beim Auslesen einer Variablen aus einem Vorschaltgerät ein Fehler aufgetreten ist:

Bit	Fehler
0	Bei dem Versuch das Vorschaltgerät zu suchen, ist ein Fehler aufgetreten.
1	Fehler beim Auslesen der Variable ACTUAL DIM LEVEL [▶ 370]
2	Fehler beim Auslesen der Variable POWER ON LEVEL [▶ 370]
3	Fehler beim Auslesen der Variable SYSTEM FAILURE LEVEL [▶ 370]
4	Fehler beim Auslesen der Variable MIN LEVEL [▶ 371]
5	Fehler beim Auslesen der Variable MAX LEVEL [▶ 371]
6	Fehler beim Auslesen der Variable FADE RATE [▶ 371]
7	Fehler beim Auslesen der Variable FADE TIME [▶ 371]
8	Fehler beim Auslesen der Variable RANDOM ADDRESS [▶ 372]
9	Fehler beim Auslesen der Variablen GROUP 0-7, GROUP 8-15 [▶ 372]
10	Fehler beim Auslesen der Variablen SCENE 0 bis SCENE 15 [▶ 372]
11	Fehler beim Auslesen der Variable STATUS INFORMATION [▶ 372]
12	Fehler beim Auslesen der Variable VERSION NUMBER [▶ 373]
13	Fehler beim Auslesen der Variable DEVICE TYPE [▶ 373]
14	Fehler beim Auslesen der Variable PHYSICAL MIN LEVEL [▶ 373]

Nach Abarbeitung des Bausteines wechselt der Ausgang *bBusy* von TRUE nach FALSE. Abhängig davon, wie viele Vorschaltgeräte angeschlossen sind, kann die Abarbeitung dieses Bausteines mehrere Sekunden dauern.

VAR_INPUT

```
bStart          : BOOL;
bCancel         : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nOptions        : DWORD;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

bCancel: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit das Auslesen der Variablen abgebrochen.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOptions: Optionen für das Auslesen der Variablen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_ACTUAL_DIM_LEVEL	Die Variable ACTUAL DIM LEVEL [▶ 370]
DALIV2_OPTION_POWER_ON_LEVEL	Die Variable POWER ON LEVEL [▶ 370]
DALIV2_OPTION_SYSTEM_FAILURE_LEVEL	Die Variable SYSTEM FAILURE LEVEL [▶ 370]
DALIV2_OPTION_MIN_LEVEL	Die Variable MIN LEVEL [▶ 371]
DALIV2_OPTION_MAX_LEVEL	Die Variable MAX LEVEL [▶ 371]
DALIV2_OPTION_FADE_RATE_FADE_TIME	Die Variablen FADE RATE [▶ 371]
DALIV2_OPTION_RANDOM_ADDRESS	Die Variable RANDOM ADDRESS [▶ 372]
DALIV2_OPTION_GROUPS	Die Variablen GROUP 0-7 und GROUP 8-15 [▶ 372]
DALIV2_OPTION_SCENE_LEVELS	Die Variablen SCENE 0 bis SCENE 15 [▶ 372]
DALIV2_OPTION_STATUS_INFORMATION	Die Variable STATUS INFORMATION [▶ 372]
DALIV2_OPTION_VERSION_NUMBER	Die Variable VERSION NUMBER [▶ 373]
DALIV2_OPTION_DEVICE_TYPE	Die Variable DEVICE TYPE [▶ 373]
DALIV2_OPTION_PHYSICAL_MIN_LEVEL	Die Variable PHYSICAL MIN LEVEL [▶ 373]

Konstante	Beschreibung
DALIV2_OPTION_DONT_CLEAR_DEVICE_SETTINGS	Vor dem Lesen wird die Variable <i>arrDALIDeviceSettings</i> nicht gelöscht.
DALIV2_OPTION_ALL	Alle Variablen werden ausgelesen.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nCurrentShortAddr : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nCurrentShortAddr: Kurzadresse vom aktuellem Vorschaltgerät von dem gerade versucht wird die Variablen auszulesen.

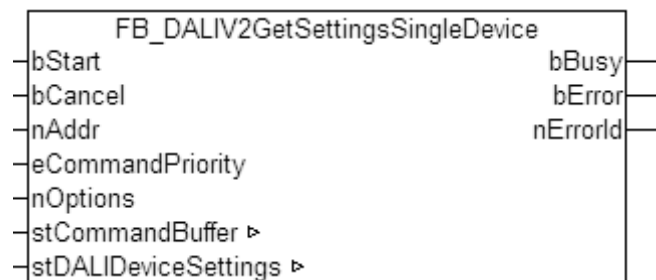
VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
arrDALIDeviceSettings : ARRAY [0..63] OF ST_DALIV2DeviceSettings;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

arrDALIDeviceSettings: Verweis auf ein Array mit 64 Elementen vom Typ [ST_DALIV2DeviceSettings](#) [▶ 389]. In dieser Variablen werden die Einstellungen von jedem einzelnen DALI-Vorschaltgerät abgelegt.

5.1.35 FB_DALIV2GetSettingsSingleDevice



Im Gegensatz zum Baustein [FB_DALIV2GetSettings\(\)](#) [▶ 80] liest dieser Baustein die Variablen (MIN LEVEL, MAX LEVEL, FADE TIME, ...) von nur einem bestimmten Gerät aus und speichert diese in einer Struktur vom Typ [ST_DALIV2DeviceSettings](#) [▶ 389].

Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Anschließend werden nacheinander die Einstellungen des Vorschaltgerätes ausgelesen und in die betreffenden Variablen der Struktur geschrieben. Tritt während des Auslesevorganges ein Lesefehler bei einem Gerät auf, so wird das Element *nErrors* der jeweiligen Struktur gesetzt, ohne dass jedoch der Funktionsbaustein an sich in Störung geht. In der folgenden Tabelle wird aufgelistet, welches Bit in der Variablen *nErrors* gesetzt wird, wenn beim Auslesen einer Variablen aus einem Vorschaltgerät ein Fehler aufgetreten ist:

Bit	Fehler
0	Bei dem Versuch das Vorschaltgerät zu suchen, ist ein Fehler aufgetreten.

Bit	Fehler
1	Fehler beim Auslesen der Variable ACTUAL DIM LEVEL [▶ 370]
2	Fehler beim Auslesen der Variable POWER ON LEVEL [▶ 370]
3	Fehler beim Auslesen der Variable SYSTEM FAILURE LEVEL [▶ 370]
4	Fehler beim Auslesen der Variable MIN LEVEL [▶ 371]
5	Fehler beim Auslesen der Variable MAX LEVEL [▶ 371]
6	Fehler beim Auslesen der Variable FADE RATE [▶ 371]
7	Fehler beim Auslesen der Variable FADE TIME [▶ 371]
8	Fehler beim Auslesen der Variable RANDOM ADDRESS [▶ 372]
9	Fehler beim Auslesen der Variablen GROUP 0-7
10	Fehler beim Auslesen der Variablen SCENE 0 bis SCENE 15 [▶ 372]
11	Fehler beim Auslesen der Variable STATUS INFORMATION [▶ 372]
12	Fehler beim Auslesen der Variable VERSION NUMBER [▶ 373]
13	Fehler beim Auslesen der Variable DEVICE TYPE [▶ 373]
14	Fehler beim Auslesen der Variable PHYSICAL MIN LEVEL [▶ 373]

Nach Abarbeitung des Bausteines wechselt der Ausgang *bBusy* von TRUE nach FALSE.

VAR_INPUT

```

bStart          : BOOL;
bCancel        : BOOL;
nAddr          : BYTE;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nOptions       : DWORD;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

bCancel: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit das Auslesen der Variablen abgebrochen.

nAddr: Die Adresse des Gerätes, dessen Werte gelesen werden sollen.

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOptions: Optionen für das Auslesen der Variablen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_ACTUAL_DIM_LEVEL	Die Variable ACTUAL DIM LEVEL [▶ 370]
DALIV2_OPTION_POWER_ON_LEVEL	Die Variable POWER ON LEVEL [▶ 370]
DALIV2_OPTION_SYSTEM_FAILURE_LEVEL	Die Variable SYSTEM FAILURE LEVEL [▶ 370]
DALIV2_OPTION_MIN_LEVEL	Die Variable MIN LEVEL [▶ 371]
DALIV2_OPTION_MAX_LEVEL	Die Variable MAX LEVEL [▶ 371]
DALIV2_OPTION_FADE_RATE_FADE_TIME	Die Variablen FADE RATE [▶ 371]
DALIV2_OPTION_RANDOM_ADDRESS	Die Variable RANDOM ADDRESS [▶ 372]
DALIV2_OPTION_GROUPS	Die Variablen GROUP 0-7 und GROUP 8-15 [▶ 372]
DALIV2_OPTION_SCENE_LEVELS	Die Variablen SCENE 0 bis SCENE 15 [▶ 372]
DALIV2_OPTION_STATUS_INFORMATION	Die Variable STATUS INFORMATION
DALIV2_OPTION_VERSION_NUMBER	Die Variable VERSION NUMBER [▶ 373]
DALIV2_OPTION_DEVICE_TYPE	Die Variable DEVICE TYPE [▶ 373]
DALIV2_OPTION_PHYSICAL_MIN_LEVEL	Die Variable PHYSICAL MIN LEVEL [▶ 373]

Konstante	Beschreibung
DALIV2_OPTION_DONT_CLEAR_DEVICE_SETTINGS	Vor dem Lesen wird die Variable <i>stDALIDeviceSettings</i> nicht gelöscht.
DALIV2_OPTION_ALL	Alle Variablen werden ausgelesen.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
stDALIDeviceSettings : ST_DALIV2DeviceSettings;
```

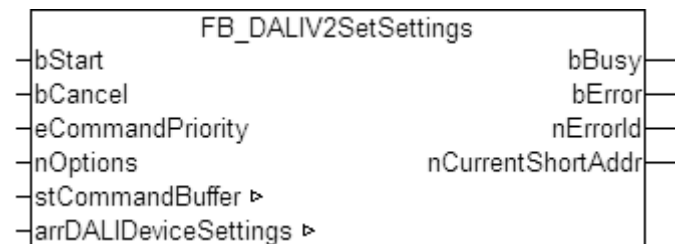
stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

stDALIDeviceSettings: Verweis auf eine Struktur vom Typ [ST_DALIV2DeviceSettings](#) [▶ 389]. In dieser Variablen werden die Einstellungen vom DALI-Vorschaltgerät abgelegt.

Sehen Sie dazu auch

- ☰ Vorschaltgeräte - Variablen [▶ 372]

5.1.36 FB_DALIV2SetSettings



Dieser Baustein initialisiert die Variablen (MIN LEVEL, MAX LEVEL, FADE TIME, ...) aller Vorschaltgeräte einer DALI-Line mit den Werten, die in einer Struktur vom Typ [ST_DALIV2DeviceSettings](#) [▶ 389] abgelegt sind.

Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Zunächst wird geprüft, ob in der jeweiligen Struktur (siehe [ST_DALIV2DeviceSettings](#) [▶ 389]) das Bit *bPresent* gesetzt ist. Ist dieses der Fall, so werden alle Variablen vom Vorschaltgerät, die nicht schreibgeschützt sind, mit den jeweiligen Werten der Struktur initialisiert. Der Index der Struktur spiegelt dabei die Adresse des Vorschaltgerätes wider, d.h. vom Gerät mit der Kurzadresse 0 liegen die Daten in *arrDALIDeviceSettings[0]* bis hin zu dem Vorschaltgerät mit der Kurzadresse 63 in *arrDALIDeviceSettings[63]*. Tritt während des Schreibvorganges ein Fehler bei einem Gerät auf, so wird das Element *nErrors* der jeweiligen Struktur gesetzt, ohne dass jedoch der Funktionsbaustein an sich in Störung geht. In der folgenden Tabelle wird aufgelistet, welches Bit in der Variablen *nErrors* gesetzt wird, wenn beim Schreiben einer Variablen in einem Vorschaltgerät ein Fehler aufgetreten ist:

Bit	Fehler
2	Fehler beim Schreiben der Variable <u>POWER ON LEVEL</u> [▶ 370]
3	Fehler beim Schreiben der Variable <u>SYSTEM FAILURE LEVEL</u> [▶ 370]
4	Fehler beim Schreiben der Variable <u>MIN LEVEL</u> [▶ 371]
5	Fehler beim Schreiben der Variable <u>MAX LEVEL</u> [▶ 371]
6	Fehler beim Schreiben der Variable <u>FADE RATE</u> [▶ 371]
7	Fehler beim Schreiben der Variable <u>FADE TIME</u> [▶ 371]
9	Fehler beim Schreiben der Variablen <u>GROUP 0-7</u> und <u>GROUP 8-15</u> [▶ 372]
10	Fehler beim Schreiben der Variablen <u>SCENE 0</u> bis <u>SCENE 15</u> [▶ 372]

Nach Abarbeitung des Bausteines wechselt der Ausgang *bBusy* von TRUE nach FALSE. Abhängig davon, wie viele Vorschaltgeräte angeschlossen sind, kann die Abarbeitung dieses Bausteines mehrere Sekunden dauern.

VAR_INPUT

```
bStart          : BOOL;
bCancel         : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nOptions        : DWORD;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

bCancel: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit das Initialisieren der Variablen abgebrochen.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOptions: Optionen für das Schreiben der Variablen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_POWER_ON_LEVEL	Die Variable POWER ON LEVEL
DALIV2_OPTION_SYSTEM_FAILURE_LEVEL	Die Variable SYSTEM FAILURE LEVEL
DALIV2_OPTION_MIN_LEVEL	Die Variable MIN LEVEL
DALIV2_OPTION_MAX_LEVEL	Die Variable MAX LEVEL
DALIV2_OPTION_FADE_RATE	Die Variable FADE RATE
DALIV2_OPTION_FADE_TIME	Die Variable FADE TIME
DALIV2_OPTION_GROUPS	Die Variablen GROUP 0-7
DALIV2_OPTION_SCENE_LEVELS	Die Variablen SCENE 0
DALIV2_OPTION_ALL	Alle Variablen werden initialisiert.
DALIV2_OPTION_PUSH_DALI_COMMANDS	Der Puffer, der die Rückmeldungen von den Vorschaltgeräten enthält, wird nicht ausgelesen. Dadurch wird das Schreiben schneller, Fehler werden jedoch nicht erkannt.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nCurrentShortAddr : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nCurrentShortAddr: Kurzadresse vom aktuellem Vorschaltgerät bei dem gerade versucht wird, die Variablen zu initialisieren.

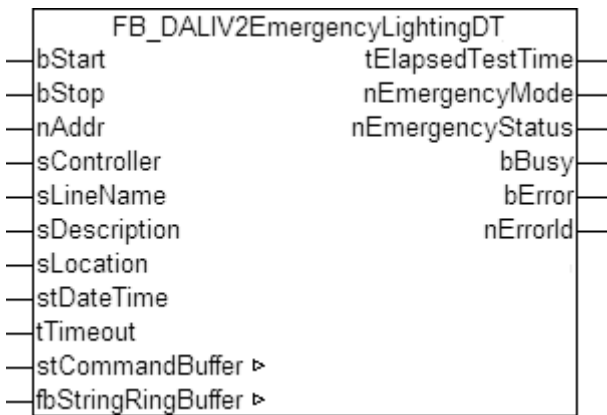
VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
arrDALIDeviceSettings : ARRAY [0..63] OF ST_DALIV2DeviceSettings;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

arrDALIDeviceSettings: Verweis auf ein Array mit 64 Elementen vom Typ `ST_DALIV2DeviceSettings` [▶ 389]. In dieser Variablen werden die Einstellungen von jedem einzelnen DALI-Vorschaltgerät abgelegt.

5.1.37 FB_DALIV2EmergencyLightingDT



Dieser Baustein dient zur Dauer-Prüfung (Duration-Test) eines DALI-Notbeleuchtungs-Gerätes. Die Testergebnisse werden zum Test-Ende über einen FIFO-Buffer (IN-OUT-Variable *fbStringRingBuffer*) geschrieben, welcher seinerseits durch den Baustein `FB_DALIV2FileLogging()` [▶ 90] in eine Datei ausgelesen wird. Ereignisse, die den Test verhindern oder unterbrechen werden zusätzlich zur Ergebnis-Meldung an den Ausgängen *bError* und *nErrorID* angezeigt.

Folgende Ereignisse verhindern die Ausführung eines Dauer-Tests:

- Das Gerät läuft im automatischen Dauertest-Modus, d.h. ein Testintervall ist im Gerät programmiert.
- Das Gerät führt gerade einen Test aus bzw. ein Test steht automatisch bevor (Funktionstest oder Dauertest)
- Die Notfall-Batterie ist nicht vollständig geladen
- Das Gerät befindet sich zu Testbeginn nicht in der Notfall-Bereitschaft ("normal-mode")

Zu den Ereignissen, die einen begonnenen Dauer-Test unterbrechen gehören diese:

- Das Gerät hat nach dem Teststart den Dauer-Test-Modus nicht erreicht
- Der Test ist nicht richtig beendet worden, d.h. das Gerät befindet sich nach dem Teststart und einer gewissen Wartezeit irgendwann weder im Test-Modus noch (wieder zurück) im Notfall-Bereitschafts-Modus ("normal-mode")
- Ein DALI-Kommando wurde fehlerhaft abgearbeitet
- Die Timeout-Zeit ist abgelaufen

i Dieser Baustein ist nur in der PC-Version der DALIV2-Bibliothek verfügbar.



Während eine Datei geöffnet ist, ist es dem Log-Baustein nicht möglich, Daten in diese Datei zu schreiben!

VAR_INPUT

```
bStart      : BOOL;
bStop       : BOOL;
nAddr       : BYTE;
sController : STRING(20);
sLineName   : STRING(10);
sDescription : STRING(20);
sLocation   : STRING(20);
stDateTime  : TIMESTRUCT;
tTimeout    : TIME := t#120m;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

bStop: Ist ein Dauertest erfolgreich gestartet worden, so wird intern der [Emergency-Mode \[▶ 377\]](#) und der [Emergency-Status \[▶ 378\]](#) zyklisch abgefragt, um zu beurteilen, wann der Test beendet ist und ob Fehler während des Tests auftreten. Eine positive Flanke am *bStop*-Eingang bricht den Dauertest in genau dieser Phase ab und gibt eine entsprechende Meldung in der Log-Datei. Dies ist dann sinnvoll, wenn lediglich beurteilt werden soll ob die Notbeleuchtung eine bestimmte Zeit lang (bis zum Zeitpunkt des Stoppens) läuft.

nAddr: Adresse des zu testenden Gerätes.

sController: Controller-Zugehörigkeit des DALI-Gerätes. Dient zur Beschreibung in der Log-Datei.

sLineName: Linien-Zugehörigkeit des DALI-Gerätes. Dient zur Beschreibung in der Log-Datei.

sDescription: Weitere ergänzende Beschreibung des Gerätes in der Log-Datei.

sLocation: Beschreibung des Einbauortes.

stDateTime: Eingang für aktuelles Datum und Zeit.

tTimeout: Zeit in welcher der Test abgeschlossen sein muss.

VAR_OUTPUT

```
tElapsedTestTime : TIME;
nEmergencyMode   : BYTE;
nEmergencyStatus : BYTE;
bBusy            : BOOL;
bError           : BOOL;
nErrorId         : UDINT;
```

tElapsedTestTime: Testdauer. Bei steigender Flanke an *bStart* wird die Zeitausgabe auf 0 gesetzt und läuft dann solange hoch, wie der Baustein aktiv ist. Mit einer fallenden Flanke an *bBusy* behält der Ausgang seinen derzeitigen Wert, sodass auch nach der Bausteinbearbeitung die Testdauer noch zur Verfügung steht.

nEmergencyMode: Während des Tests wird an diesem Ausgang der derzeit intern abgefragte Emergency-Mode des DALI-Gerätes ausgegeben.

nEmergencyStatus: dto. für Emergency-Status.

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung des Tests ein Fehler, wie oben beschrieben, aufgetreten ist. Durch das erneute Ausführen eines Tests wird dieser Ausgang zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines neuen Tests auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
fbStringRingBuffer : FB_MemRingBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

fbStringRingBuffer: Verweis auf den FIFO-Buffer, in denen die Log-Einträge abgelegt werden.

Ein <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019292427.zip> zeigt die richtige Belegung der Eingänge sowie die Verknüpfung zum Logger-Baustein `FB_DALIV2FileLogging()`. Dabei werden 3 Vorschaltgeräte und dessen Lampen jeweils am 8. jeden Monats getestet.

Die Log-Datei würde dann für die 3 adressierten Geräte beispielsweise folgendes anzeigen:

	A	B	C	D	E	F	G	H
1	Date/Time	Controller	Line	Address	Description	Location	Duration	Result
2								
3	08.07.2009 08:40	CX-Floor1	Line01	1	EM-14	Entrance A	00:00:02	Error while executing DALI-Command.
4								
5	08.07.2009 09:45	CX-Floor1	Line01	2	EM-15	Entrance B	00:00:02	Error while executing DALI-Command.
6								
7	08.07.2009 10:50	CX-Floor1	Line01	3	EM-16	Exit C	00:00:02	Error while executing DALI-Command.
8								
9	08.08.2009 08:40	CX-Floor1	Line01	1	EM-14	Entrance A	00:00:07	Test not executable: Battery not fully charged.
10								
11	08.08.2009 09:45	CX-Floor1	Line01	2	EM-15	Entrance B	00:00:02	Error while executing DALI-Command.
12								
13	08.08.2009 10:50	CX-Floor1	Line01	3	EM-16	Exit C	00:00:02	Error while executing DALI-Command.
14								
15	08.09.2009 08:40	CX-Floor1	Line01	1	EM-14	Entrance A	01:00:07	PASS
16								
17	08.09.2009 09:45	CX-Floor1	Line01	2	EM-15	Entrance B	01:00:07	PASS
18								
19	08.09.2009 10:50	CX-Floor1	Line01	3	EM-16	Exit C	01:00:07	PASS
20								

Am 8. Juli sind noch keine der Vorschaltgeräte eingebaut, was zu DALI-Kommando-Fehlern bei allen Geräten führt.

Am 8. August ist immerhin schon die erste Lampe angeschlossen, nur die Batterie ist noch nicht voll geladen.

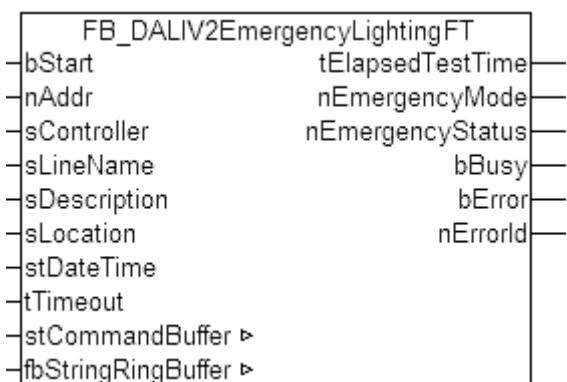
Am 8. September schließlich sind alle Geräte betriebsbereit und haben ihren ersten Dauertest erfolgreich abgeschlossen.

Es ist zu beachten, dass beim Öffnen der Datei ggf. die Spaltenbreite zur Sichtbarkeit der Daten erweitert werden muss.

Sehen Sie dazu auch

📖 [FB_DALIV2FileLogging](#) [► 90]

5.1.38 FB_DALIV2EmergencyLightingFT



Dieser Baustein dient zur Funktions-Prüfung (Function-Test) eines DALI-Notbeleuchtungs-Gerätes. Die Testergebnisse werden zum Test-Ende über einen FIFO-Buffer (IN-OUT-Variable `fbStringRingBuffer`) geschrieben, welcher seinerseits durch den Baustein `FB_DALIV2FileLogging()` [► 90] in eine Datei ausgelesen wird. Ereignisse die den Test verhindern oder unterbrechen werden zusätzlich zur Ergebnis-Meldung an den Ausgängen `bError` und `nErrorID` angezeigt.

Folgende Ereignisse verhindern die Ausführung eines Funktionstest:

- Das Gerät läuft im automatischen Funktionstest-Modus, d.h. ein Testintervall ist im Gerät programmiert.
- Das Gerät führt gerade einen Test aus bzw. ein Test steht automatisch bevor (Funktionstest oder Dauertest)
- Das Gerät befindet sich zu Testbeginn nicht in der Notfall-Bereitschaft ("normal-mode")

Zu den Ereignissen, die einen begonnenen Funktionstest unterbrechen gehören diese:

- Das Gerät hat nach dem Teststart den Funktionstest-Modus nicht erreicht
- Der Test ist nicht richtig beendet worden, d.h. das Gerät befindet sich nach dem Teststart und einer gewissen Wartezeit weder im Test-Modus noch (wieder zurück) im Notfall-Bereitschafts-Modus ("normal-mode")
- Ein DALI-Kommando wurde fehlerhaft abgearbeitet
- Die Timeout-Zeit ist abgelaufen



Dieser Baustein ist nur in der PC-Version der DALIV2-Bibliothek verfügbar.



Während eine Datei geöffnet ist, ist es dem Log-Baustein nicht möglich, Daten in diese Datei zu schreiben!

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
sController : STRING(20);
sLineName   : STRING(10);
sDescription : STRING(20);
sLocation   : STRING(20);
stDateTime  : TIMESTRUCT;
tTimeout    : TIME := t#120m;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse des zu testenden Gerätes.

sController: Controller-Zugehörigkeit des DALI-Gerätes. Dient zur Beschreibung in der Log-Datei.

sLineName: Linien-Zugehörigkeit des DALI-Gerätes. Dient zur Beschreibung in der Log-Datei.

sDescription: Weitere ergänzende Beschreibung des Gerätes in der Log-Datei.

sLocation: Beschreibung des Einbauortes.

stDateTime: Eingang für aktuelles Datum und Zeit.

tTimeout: Zeit in der der Test abgeschlossen sein muss.

VAR_OUTPUT

```
tElapsedTestTime : TIME;
nEmergencyMode   : BYTE;
nEmergencyStatus : BYTE;
bBusy            : BOOL;
bError           : BOOL;
nErrorId         : UDINT;
```

tElapsedTestTime: Testdauer. Bei steigender Flanke an *bStart* wird die Zeitausgabe auf 0 gesetzt und läuft dann solange hoch, wie der Baustein aktiv ist. Mit einer fallenden Flanke an *bBusy* behält der Ausgang seinen derzeitigen Wert, sodass auch nach der Bausteinabarbeitung die Testdauer noch zur Verfügung steht.

nEmergencyMode: Während des Tests wird an diesem Ausgang der derzeit intern abgefragte Emergency-Mode [► 377] des DALI-Gerätes ausgegeben.

nEmergencyStatus: dto. für [Emergency-Status](#) [[▶ 378](#)].

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung des Tests ein Fehler, wie oben beschrieben, aufgetreten ist. Durch das erneute Ausführen eines Tests wird dieser Ausgang zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines neuen Tests auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
fbStringRingBuffer   : FB_MemRingBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [[▶ 95](#)] (KL6811) oder [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

fbStringRingBuffer: Verweis auf den FIFO-Buffer, in denen die Log-Einträge abgelegt werden.

Ein <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019292427.zip> zeigt die richtige Belegung der Eingänge sowie die Verknüpfung zum Logger-Baustein [FB_DALIV2FileLogging\(\)](#) [[▶ 90](#)]. Dabei werden 3 Vorschaltgeräte und dessen Lampen jeweils immer am Dienstag getestet.

Die Log-Datei würde dann für die 3 adressierten Geräte beispielsweise folgendes anzeigen:

	A	B	C	D	E	F	G	H
1	Date/Time	Controller	Line	Address	Description	Location	Duration	Result
2								
3	25.08.2009 15:00	CX-Floor1	Line01	1	EM-14	Entrance A	00:00:01	Error while executing DALI-Command.
4								
5	25.08.2009 15:02	CX-Floor1	Line01	2	EM-15	Entrance B	00:00:01	Error while executing DALI-Command.
6								
7	25.08.2009 15:04	CX-Floor1	Line01	3	EM-16	Exit C	00:00:02	Error while executing DALI-Command.
8								
9	01.09.2009 15:00	CX-Floor1	Line01	1	EM-14	Entrance A	00:00:01	Device in auto-test-mode! Test not executed
10								
11	01.09.2009 15:02	CX-Floor1	Line01	2	EM-15	Entrance B	00:00:01	Error while executing DALI-Command.
12								
13	01.09.2009 15:04	CX-Floor1	Line01	3	EM-16	Exit C	00:00:02	Error while executing DALI-Command.
14								
15	08.09.2009 15:00	CX-Floor1	Line01	1	EM-14	Entrance A	00:00:19	PASS
16								
17	08.09.2009 15:02	CX-Floor1	Line01	2	EM-15	Entrance B	00:00:41	PASS
18								
19	08.09.2009 15:04	CX-Floor1	Line01	3	EM-16	Exit C	00:00:41	PASS
20								

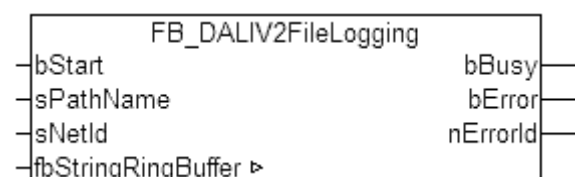
Am 25. August sind noch keine der Vorschaltgeräte eingebaut, was zu DALI-Kommando-Fehlern bei allen Geräten führt.

Am 1. September ist immerhin schon die erste Lampe angeschlossen, jedoch ist der automatische Test im Vorschaltgerät noch aktiviert, was zu einem Testabbruch führt.

Am 8. September schließlich sind alle Geräte betriebsbereit und haben ihren ersten Funktionstest erfolgreich abgeschlossen.

Es ist zu beachten, dass beim Öffnen der Datei ggf. die Spaltenbreite zur Sichtbarkeit der Daten erweitert werden muss.

5.1.39 FB_DALIV2FileLogging



Dieser Baustein liest die in den Bausteinen [FB_DALIV2EmergencyLightingFT\(\)](#) [► 88] und [FB_DALIV2EmergencyLightingDT\(\)](#) [► 86] jeweils beschriebenen FIFO-Puffer (IN-OUT-Variable *fbStringRingBuffer*) aus und schreibt die Inhalte in eine Log-Datei.



Dieser Baustein ist nur in der PC-Version der DALIV2-Bibliothek verfügbar.

VAR_INPUT

```
bStart      : BOOL;
sPathName   : STRING;
sNetId      : STRING;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

sPathName: Enthält den Pfad- und Dateinamen der zu öffnenden Pufferdatei.

Der Pfad kann nur auf das lokale File System des Rechners zeigen! Das bedeutet, Netzwerkpfade können hier nicht angegeben werden!

sNetId: Hier kann ein String mit der Netzwerkadresse des TwinCAT-Rechners angegeben werden, auf dem die Pufferdatei geschrieben/gelesen werden soll. Für den lokalen Rechner kann auch ein Leerstring angegeben werden.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis der Log-Puffer geleert ist.

bError: Sollte ein ADS-Fehler bei der Übertragung in die Log-Datei erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

nErrId: Liefert bei einem gesetzten *bError*-Ausgang die ADS-Fehlernummer oder den Befehlsspezifischen Fehlercode (Tabelle).

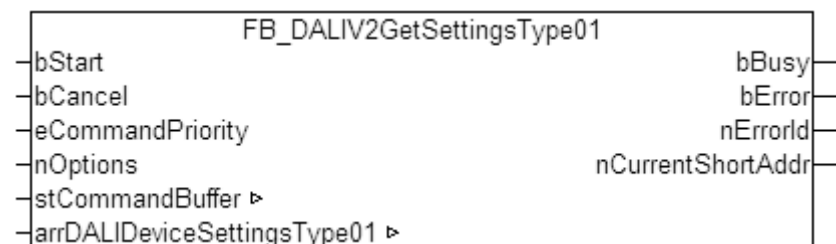
VAR_IN_OUT

```
fbStringRingBuffer : FB_MemRingBuffer;
```

fbStringRingBuffer: Verweis auf den FIFO-Buffer, in denen die Log-Einträge abgelegt werden.

Ein <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019292427.zip> zeigt die richtige Belegung der Eingänge sowie die Verknüpfung zum Logger-Baustein *FB_DALIV2FileLogging*. Das Erscheinungsbild der jeweiligen Log-Datei ist unter den Funktionsbausteinen [FB_DALIV2EmergencyLightingDT\(\)](#) [► 86] und [FB_DALIV2EmergencyLightingFT\(\)](#) [► 88] dargestellt

5.1.40 FB_DALIV2GetSettingsType01



Dieser Baustein liest von allen Notbeleuchtungs-Vorschaltgeräten einer DALI-Line die Variablen (BATTERY CHARGE, DURATION TEST RESULT, LAMP EMERGENCY TIME ...) aus, und speichert diese in einer Struktur vom Typ [ST_DALIV2DeviceSettingsType01](#) [► 390].

Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Zunächst wird geprüft, ob ein Vorschaltgerät überhaupt vorhanden ist. Ist dies der Fall, so wird in der jeweiligen Struktur (siehe [ST_DALIV2DeviceSettingsType01](#) [▶ 390]) das Bit *bPresent* gesetzt und anschließend nacheinander die Einstellungen des Vorschaltgerätes ausgelesen und in die betreffenden Variablen der Struktur geschrieben. Wird ein Gerät als nicht vorhanden erkannt, so wird das Auslesen übersprungen und mit dem nächsten Gerät begonnen. Der Index der Struktur spiegelt dabei die Adresse des Gerätes wider, d.h. vom Gerät mit der Kurzadresse 0 liegen die Daten in *arrDALIDeviceSettingsType01[0]* bis hin zu dem Gerät mit der Kurzadresse 63 in *arrDALIDeviceSettingsType01[63]*. Tritt während des Auslesevorganges ein Lesefehler bei einem Gerät auf, so wird das entsprechende Bit in *nErrors* der jeweiligen Struktur gesetzt, ohne dass jedoch der Funktionsbaustein an sich in Störung geht. In der folgenden Tabelle wird aufgelistet, welches Bit in der Variablen *nErrors* gesetzt wird, wenn beim Auslesen einer Variablen aus einem Vorschaltgerät ein Fehler aufgetreten ist:

Bit	Fehler
0	Bei dem Versuch das Vorschaltgerät zu suchen, ist ein Fehler aufgetreten.
1	Fehler beim Auslesen der Variable BATTERY CHARGE [▶ 376]
2	Fehler beim Auslesen der Variable DURATION TEST RESULT [▶ 376]
3	Fehler beim Auslesen der Variable LAMP EMERGENCY TIME [▶ 377]
4	Fehler beim Auslesen der Variable LAMP TOTAL OPERATION TIME [▶ 377]
5	Fehler beim Auslesen der Variable EMERGENCY LEVEL [▶ 375]
6	Fehler beim Auslesen der Variable EMERGENCY MIN LEVEL [▶ 375]
7	Fehler beim Auslesen der Variable EMERGENCY MAX LEVEL [▶ 375]
8	Fehler beim Auslesen der Variable RATED DURATION [▶ 377]
9	Fehler beim Auslesen der Variable FUNCTION TEST DELAY TIME [▶ 376]
10	Fehler beim Auslesen der Variable DURATION TEST DELAY TIME [▶ 376]
11	Fehler beim Auslesen der Variable FUNCTION TEST INTERVAL [▶ 376]
12	Fehler beim Auslesen der Variable DURATION TEST INTERVAL [▶ 376]
13	Fehler beim Auslesen der Variable TEST EXECUTION TIMEOUT [▶ 376]
14	Fehler beim Auslesen der Variable PROLONG TIME [▶ 375]
15	Fehler beim Auslesen der Variable EMERGENCY MODE [▶ 377]
16	Fehler beim Auslesen der Variable FEATURES [▶ 377]
17	Fehler beim Auslesen der Variable FAILURE STATUS [▶ 377]
18	Fehler beim Auslesen der Variable EMERGENCY STATUS [▶ 378]

Nach Abarbeitung des Bausteines wechselt der Ausgang *bBusy* von TRUE nach FALSE. Abhängig davon, wie viele Vorschaltgeräte angeschlossen sind, kann die Abarbeitung dieses Bausteines mehrere Sekunden dauern.

VAR_INPUT

```
bStart          : BOOL;
bCancel         : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityHigh;
nOptions       : DWORD;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

bCancel: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit das Auslesen der Variablen abgebrochen.

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOptions: Optionen für das Auslesen der Variablen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_BATTERY_CHARGE	Die Variable BATTERY CHARGE
DALIV2_OPTION_DURATION_TEST_RESULT	Die Variable DURATION TEST RESULT
DALIV2_OPTION_LAMP_EMERGENCY_TIME	Die Variable LAMP EMERGENCY TIME
DALIV2_OPTION_LAMP_TOTAL_OPERATION_TIME	Die Variable LAMP TOTAL OPERATION TIME
DALIV2_OPTION_EMERGENCY_LEVEL	Die Variable EMERGENCY LEVEL
DALIV2_OPTION_EMERGENCY_MIN_LEVEL	Die Variable EMERGENCY MIN LEVEL
DALIV2_OPTION_EMERGENCY_MAX_LEVEL	Die Variable EMERGENCY MAX LEVEL
DALIV2_OPTION_RATED_DURATION	Die Variable RATED DURATION
DALIV2_OPTION_NEXT_FUNCTION_TEST	Die Variable FUNCTION TEST DELAY TIME
DALIV2_OPTION_NEXT_DURATION_TEST	Die Variable DURATION TEST DELAY TIME
DALIV2_OPTION_FUNCTION_TEST_INTERVAL	Die Variable FUNCTION TEST INTERVAL
DALIV2_OPTION_DURATION_TEST_INTERVAL	Die Variable DURATION TEST INTERVAL
DALIV2_OPTION_TEST_EXECUTION_TIMEOUT	Die Variable TEST EXECUTION TIMEOUT
DALIV2_OPTION_PROLONG_TIME	Die Variable PROLONG TIME
DALIV2_OPTION_EMERGENCY_MODE	Die Variable EMERGENCY MODE
DALIV2_OPTION_FEATURES	Die Variable FEATURES [► 377] wird ausgelesen.
DALIV2_OPTION_FAILURE_STATUS	Die Variable FAILURE STATUS
DALIV2_OPTION_EMERGENCY_STATUS	Die Variable EMERGENCY STATUS
DALIV2_OPTION_ALL	Alle Variablen werden ausgelesen.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId      : UDINT;
nCurrentShortAddr : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

nCurrentShortAddr: Kurzadresse vom aktuellen Vorschaltgerät von dem gerade versucht wird die Variablen auszulesen.

VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
arrDALIDeviceSettingsType01 : ARRAY [0..63] OF ST_DALIV2DeviceSettingsType01;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

arrDALIDeviceSettingsType01: Verweis auf ein Array mit 64 Elementen vom Typ [ST_DALIV2DeviceSettingsType01 \[► 390\]](#). In dieser Variablen werden die Einstellungen von jedem einzelnen DALI-Vorschaltgerät abgelegt.

5.1.41 FB_DALIV2SetSettingsType01



Dieser Baustein initialisiert die Variablen (EMERGENCY LEVEL, FUNCTION TEST DELAY TIME, DURATION TEST DELAY TIME ...) aller Notbeleuchtungs-Vorschaltgeräten einer DALI-Line mit den Werten, die in einer Struktur vom Typ [ST_DALIV2DeviceSettingsType01](#) [[390](#)] abgelegt sind.

Durch eine positive Flanke an dem Eingang *bStart* wird der Baustein gestartet und der Ausgang *bBusy* geht auf TRUE. Zunächst wird geprüft, ob in der jeweiligen Struktur (siehe [ST_DALIV2DeviceSettingsType01](#) [[390](#)]) das Bit *bPresent* gesetzt ist. Ist dieses der Fall, so werden alle Variablen vom Vorschaltgerät, die nicht schreibgeschützt sind, mit den jeweiligen Werten der Struktur initialisiert. Der Index der Struktur spiegelt dabei die Adresse des Vorschaltgerätes wider, d.h. vom Gerät mit der Kurzadresse 0 liegen die Daten in *arrDALIDeviceSettingsType01*[0] bis hin zu dem Vorschaltgerät mit der Kurzadresse 63 in *arrDALIDeviceSettingsType01*[63]. Tritt während des Schreibvorganges ein Fehler bei einem Gerät auf, so wird das Element *nErrors* der jeweiligen Struktur gesetzt, ohne dass jedoch der Funktionsbaustein an sich in Störung geht. In der folgenden Tabelle wird aufgelistet, welches Bit in der Variablen *nErrors* gesetzt wird, wenn beim Schreiben einer Variablen in einem Vorschaltgerät ein Fehler aufgetreten ist:

Bit	Fehler
5	Fehler beim Schreiben der Variable EMERGENCY LEVEL [375]
9	Fehler beim Schreiben der Variable FUNCTION TEST DELAY TIME [376]
10	Fehler beim Schreiben der Variable DURATION TEST DELAY TIME [376]
11	Fehler beim Schreiben der Variable FUNCTION TEST INTERVAL [376]
12	Fehler beim Schreiben der Variable DURATION TEST INTERVAL [376]
13	Fehler beim Schreiben der Variable TEST EXECUTION TIMEOUT [376]
14	Fehler beim Schreiben der Variable PROLONG TIME [375]

Nach Abarbeitung des Bausteines wechselt der Ausgang *bBusy* von TRUE nach FALSE. Abhängig davon, wie viele Vorschaltgeräte angeschlossen sind, kann die Abarbeitung dieses Bausteines mehrere Sekunden dauern.

VAR_INPUT

```

bStart          : BOOL;
bCancel         : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityHigh;
nOptions        : DWORD;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

bCancel: Über eine positive Flanke an diesem Eingang wird der Baustein deaktiviert und somit das Auslesen der Variablen abgebrochen.

eCommandPriority: [Priorität](#) [[387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOptions: Optionen für das Auslesen der Variablen (siehe Tabelle). Die einzelnen Konstanten müssen miteinander ODER verknüpft werden.

Konstante	Beschreibung
DALIV2_OPTION_EMERGENCY_LEVEL	Die Variable EMERGENCY LEVEL
DALIV2_OPTION_NEXT_FUNCTION_TEST	Die Variable FUNCTION TEST DELAY TIME

Konstante	Beschreibung
DALIV2_OPTION_NEXT_DURATION_TEST	Die Variable DURATION TEST DELAY TIME
DALIV2_OPTION_FUNCTION_TEST_INTERVAL	Die Variable FUNCTION TEST INTERVAL
DALIV2_OPTION_DURATION_TEST_INTERVAL	Die Variable DURATION TEST INTERVAL
DALIV2_OPTION_TEST_EXECUTION_TIMEOUT	Die Variable TEST EXECUTION TIMEOUT
DALIV2_OPTION_PROLONG_TIME	Die Variable PROLONG TIME
DALIV2_OPTION_ALL	Alle Variablen werden initialisiert.
DALIV2_OPTION_PUSH_DALI_COMMANDS	Der Puffer, der die Rückmeldungen von den Vorschaltgeräten enthält, wird nicht ausgelesen. Dadurch wird das Schreiben schneller, Fehler werden jedoch nicht erkannt.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId      : UDINT;
nCurrentShortAddr : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nCurrentShortAddr: Kurzadresse vom aktuellen Vorschaltgerät bei dem gerade versucht wird, die Variablen zu initialisieren.

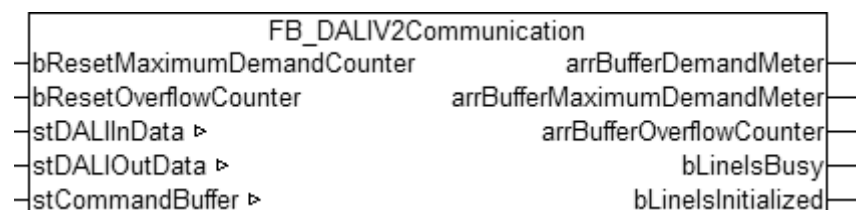
VAR_IN_OUT

```
stCommandBuffer      : ST_DALIV2CommandBuffer;
arrDALIDeviceSettingsType01 : ARRAY [0..63] OF ST_DALIV2DeviceSettingsType01;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

arrDALIDeviceSettingsType01: Verweis auf ein Array mit 64 Elementen vom Typ [ST_DALIV2DeviceSettingsType01](#) [▶ 390]. In dieser Variablen werden die Einstellungen von jedem einzelnen DALI-Vorschaltgerät abgelegt.

5.1.42 FB_DALIV2Communication



Die Bausteine für die DALI-Befehle greifen nicht direkt auf das Prozessabbild der DALI-Busklemme zu, sondern legen die einzelnen DALI-Befehle in drei verschiedene Puffer ab. Der Baustein [FB_DALIV2Communication\(\)](#) liest sequentiell die DALI-Befehle aus diesen drei Puffern aus und gibt die DALI-Befehle zu der KL6811 weiter. Hierdurch wird sichergestellt, dass nicht mehrere Bausteine gleichzeitig auf das Prozessabbild der KL6811 zugreifen. Jeder dieser drei Puffer wird mit einer anderen Priorität (hoch, mittel oder niedrig) abgearbeitet. Der Anwender der SPS-Bibliothek kann durch den Parameter *eCommandPriority*, den es bei den meisten Bausteinen gibt, beeinflussen mit welcher Priorität der jeweilige DALI-Befehl von dem Baustein [FB_DALIV2Communication\(\)](#) bearbeitet werden soll.

Die Puffer, in denen die DALI-Befehle abgelegt werden, sind alle in einer Variablen vom Typ `ST_DALIV2CommandBuffer` enthalten. Pro KL6811 gibt es eine Instanz vom Baustein `FB_DALIV2Communication()` und eine Variable vom Typ `ST_DALIV2CommandBuffer`. Der Baustein `FB_DALIV2Communication()` sollte, wenn möglich, in einer separaten, schnelleren Task aufgerufen werden.

Über die Ausgänge des Bausteins kann ermittelt werden, wie stark die Puffer ausgelastet sind. Hierzu werden drei Arrays ausgegeben, bei dem jedes Element (0, 1 oder 2) für einen der drei Buffer (hoch, mittel oder niedrig) steht. Sollten Sie feststellen, dass einer der drei Puffer regelmäßig überläuft, so sollten Sie folgende Maßnahmen in Betracht ziehen:

- Wie stark sind die einzelnen SPS-Tasks ausgelastet? Der TwinCAT System Manager bietet zur Analyse entsprechende Hilfsmittel an.
- Versuchen Sie die Zykluszeit der Task, in der der Baustein `FB_DALIV2Communication()` aufgerufen wird, zu verringern. Der Wert sollte nicht größer als 6 ms sein, optimal sind 2 ms.
- Überprüfen Sie die Zykluszeit der SPS-Task, in der die Bausteine für die einzelnen DALI-Befehle aufgerufen werden. Dieser Wert sollte zwischen 10 ms und 60 ms liegen.
- Vermeiden Sie möglichst das Pollen (regelmäßiges Auslesen) von Werten. Lesen Sie nur dann Werte aus, wenn diese auch benötigt werden.
- Verteilen Sie die einzelnen Vorschaltgeräte gleichmäßig auf mehrere DALI-Linien. Da pro SPS-Zyklus mehrere DALI-Linien gleichzeitig bearbeitet werden, erhöht sich hierdurch der Datendurchsatz insgesamt.

VAR_INPUT

```
bResetMaximumDemandCounter : BOOL;
bResetOverflowCounter       : BOOL;
```

bResetMaximumDemandCounter: Eine positive Flanke setzt den gespeicherten Wert der maximalen Befehlspeicher-Auslastung, `arrBufferMaximumDemandMeter` (0 - 100%, siehe `VAR_OUTPUT`), zurück.

bResetOverflowCounter: Eine positive Flanke setzt den gespeicherten Wert der Anzahl der Befehlspeicher-Überläufe, `arrBufferOverflowCounter` (siehe `VAR_OUTPUT`), zurück.

VAR_OUTPUT

```
arrBufferDemandMeter        : ARRAY [0..2] OF BYTE;
arrBufferMaximumDemandMeter : ARRAY [0..2] OF BYTE;
arrBufferOverflowCounter    : ARRAY [0..2] OF UINT;
bLineIsBusy                 : BOOL;
bLineIsInitialized          : BOOL;
```

arrBufferDemandMeter: Belegung des jeweiligen Puffers (0 - 100%).

arrBufferMaximumDemandMeter: Bisherige maximale Auslastung des jeweiligen Puffers (0 - 100%).

arrBufferOverflowCounter: Bisherige Anzahl der Pufferüberläufe.

bLineIsBusy: Solange der Baustein `FB_DALIV2Communication()` aktiv ist, ist dieser Ausgang gesetzt.

bLineIsInitialized: Wird der Baustein das erste Mal aufgerufen (z.B. beim Starten der Steuerung), so wird eine Initialisierung durchgeführt. Während dieser Zeit können keine DALI-Befehle bearbeitet werden.

VAR_IN_OUT

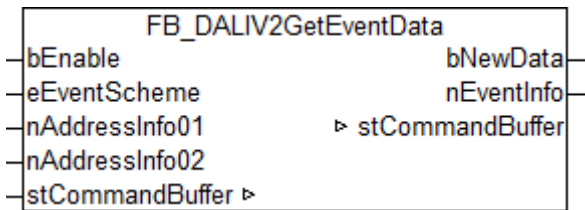
```
stDALIInData      : ST_DALIV2InData;
stDALIOutData     : ST_DALIV2OutData;
stCommandBuffer   : ST_DALIV2CommandBuffer;
```

stDALIInData: Struktur im Eingangsprozessabbild [[▶ 390](#)] der KL6811. Sie dient zur Kommunikation von der KL6811 zur SPS. Bei Verwendung von `FB_KL6811Config()` [[▶ 102](#)] wird diese Struktur mit dem Parameter `stInData` des Bausteins verbunden.

stDALIOutData: Struktur im Ausgangsprozessabbild [[▶ 391](#)] der KL6811. Sie dient zur Kommunikation von der SPS zur KL6811. Bei Verwendung von `FB_KL6811Config()` wird diese Struktur mit dem Parameter `stOutData` des Bausteins verbunden.

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den DALI-Bausteinen.

5.1.43 FB_DALIV2GetEventData



Filtert ein, durch das Ereignis-Schema spezifiziertes Ereignis aus.

Jedes Ereignis, das von einem DALI-Gerät versendet wird enthält zwei Felder die Auskunft über die Ereignisquelle liefern. Diese beiden Felder sind jeweils eine Kombination aus der Kurzadresse, Instanznummer, Instanztyp, Instanzgruppe oder der Gerätegruppe. Bei dem Empfänger der Ereignisse muss bekannt sein, nach welchem Adress-Schema die Daten versendet werden.

Pro Ereignis, welches empfangen und weiter verarbeitet werden soll, muss eine Instanz von FB_DALIV2GetEventData() angelegt und mit den richtigen Ereignis-Schema konfiguriert werden.

VAR_INPUT

```
bEnable      : BOOL;
eEventScheme : E_DALIV2EventScheme := eDALIV2EventSchemeDeviceInstance;
nAddressInfo01 : BYTE;
nAddressInfo02 : BYTE;
```

bEnable: Schaltet den Baustein frei. Ist dieser Eingang auf FALSE, so werden keine weiteren Ereignisse ausgegeben.

eEventScheme: Das Ereignis-Schema [▶ 388] definiert die Adress-Informationen, die für die Filterung des gewünschten Ereignis erwartet werden.

nAddressInfo01: (siehe Tabelle unten)

nAddressInfo02: (siehe Tabelle unten)

eEventScheme	nAddressInfo01	nAddressInfo02
eDALIV2EventSchemeInstance	Instanztyp (0-31)	Instanznummer (0-31)
eDALIV2EventSchemeDevice	Kurzadresse (0-63)	Instanztyp (0-31)
eDALIV2EventSchemeDeviceInstance	Kurzadresse (0-63)	Instanznummer (0-31)
eDALIV2EventSchemeDeviceGroup	Gerätegruppe (0-31)	Instanztyp (0-31)
eDALIV2EventSchemeInstanceGroup	Instanzgruppe (0-31)	Instanztyp (0-31)

VAR_OUTPUT

```
bNewData : BOOL;
nEventInfo : WORD;
```

bNewData: Wurde ein Ereignis empfangen, das dem Ereignis-Schema und den Adressinformationen entspricht, so geht dieser Ausgang für ein SPS-Zyklus auf TRUE.

nEventInfo: Ist der Ausgang *bNewData* TRUE, so liegen an diesem Ausgang weitere Informationen zu dem Ereignis an. Die genaue Bedeutung ist abhängig vom Gerätetyp und wird im jeweiligen Part 3xx der IEC 62386 beschrieben.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.44 FB_DALIV2SendDALICommand



Dieser Baustein dient zum allgemeinen Senden eines DALI-Kommandos, definiert per Befehlsnummer und, falls erforderlich, Übergabeparameter. Dabei kann weiterhin eingestellt werden, ob der Befehl zweimal wiederholt gesendet wird und ob auf eine Antwort gewartet werden soll. Letzteres kann beispielsweise dazu benutzt werden, eine schnelle Abfolge von Step-Up-Befehlen zu realisieren.

Gegenüber dem Baustein `FB_DALIV2xSendDALICommand()` [► 100] ist dieser Funktionsblock nur in der Lage DALI-Vorschaltgeräte (Lampen) anzusteuern.

VAR_INPUT

```

bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nCommand        : INT := 0;
nParameter      : BYTE := 0;
bWaitingForDALISlaveResponse : BOOL := FALSE;
bRepeatCommand  : BOOL := FALSE;
bSuppressResponseBuffer : BOOL := FALSE;
nDeviceType     : BYTE := 0;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nCommand: Nummer des zu sendenden DALI-Befehls.

nParameter: Parameter zur Wertübergabe.

bWaitingForDALISlaveResponse: Bei *FALSE* wird **nicht** auf die Antwort des DALI-Gerätes gewartet. Die Anwendung ist im Zusammenhang mit jeder Art von Abfragebefehlen sinnlos.

bRepeatCommand: Entscheidet, ob das Kommando zweimal hintereinander gesendet werden soll.

bSuppressResponseBuffer: Falls *TRUE* anliegt, wird der interne Puffer **nicht** mit der Antwort vom Baustein `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821) gefüllt.

nDeviceType: Kennung für den Gerätetyp.

Wert	Beschreibung
0	Standardgerät
1	Gerät für <u>Notbeleuchtung</u> [► 32].
2	Gerät für <u>Entladungslampen</u> [► 36].
3	Gerät für <u>Niedervolt-Halogenlampen</u> .
4	Gerät zum <u>Dimmen von Glühlampen</u> .
5	Gerät zur <u>Umwandelung von digitalen Signalen in Gleichspannungssignale</u> .
6	Gerät für <u>Licht emittierende Dioden (LED)</u> [► 36].
7	Gerät für <u>Schaltfunktionen</u> .
8	Gerät zur <u>Farb-/Farbtemperatursteuerung</u> [► 34].
9	Sequenz.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId      : UDINT;
nResponseData : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf *TRUE* geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf *FALSE* zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

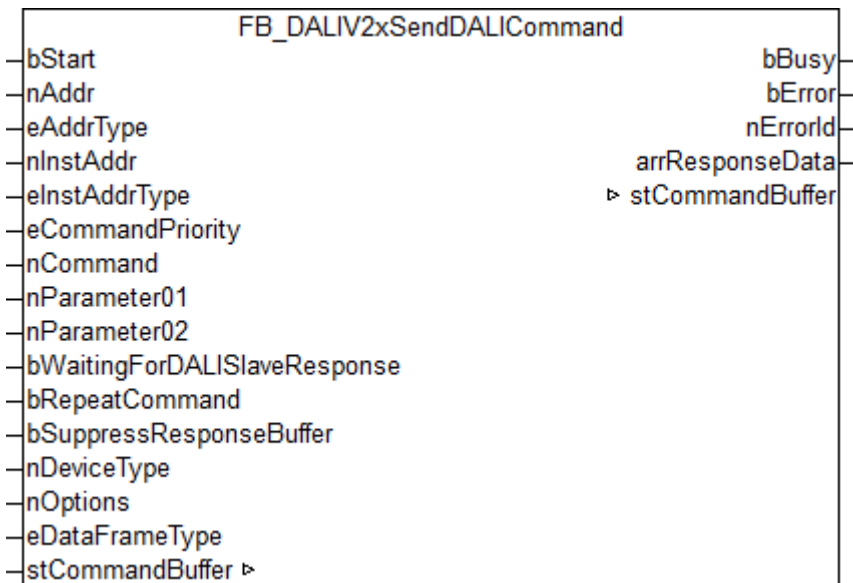
nResponseData: Der empfangende Wert vom DALI-Gerät, falls ein Abfragebefehl aufgerufen wurde.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

5.1.45 FB_DALIV2xSendDALICommand



Dieser Baustein dient zum allgemeinen Senden eines DALI-Kommandos, definiert per Befehlsnummer und -falls erforderlich- Übergabeparameter. Dabei kann weiterhin eingestellt werden, ob der Befehl zweimal wiederholt gesendet wird und ob auf eine Antwort gewartet werden soll. Letzteres kann beispielsweise dazu benutzt werden, eine schnelle Abfolge von Step-Up-Befehlen zu realisieren.

Gegenüber dem Baustein `FB_DALIV2SendDALICommand()` [▶ 98] ist dieser Funktionsblock auch in der Lage DALI-Steuergeräte (Sensoren) anzusteuern.

VAR_INPUT

```

bStart          : BOOL;
nAddr           : BYTE;
eAddrType      : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr      : BYTE := 0;
eInstAddrType  : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nCommand       : INT := 0;
nParameter01   : BYTE := 0;
nParameter02   : DINT := 0;
bWaitingForDALISlaveResponse : BOOL := FALSE;
bRepeatCommand : BOOL := FALSE;
bSuppressResponseBuffer : BOOL := FALSE;
nDeviceType    : BYTE := 0;
nOptions       : DWORD := 0;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType16Bit;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz innerhalb des Teilnehmers.

eInstAddrType: Definiert die Bedeutung der Variable [▶ 388] nInstAddr für das Ansprechen der Instanz (z.B. per Instanznummer, Instanztyp, ...)

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Bibliothek abgearbeitet wird.

nCommand: Nummer des zu sendenden DALI-Befehls.

nParameter01: Parameter zur Wertübergabe.

nParameter02: Parameter zur Wertübergabe.

bWaitingForDALISlaveResponse: Bei *FALSE* wird **nicht** auf die Antwort des DALI-Gerätes gewartet. Die Anwendung ist im Zusammenhang mit jeder Art von Abfragebefehlen sinnlos.

bRepeatCommand: Entscheidet, ob das Kommando zweimal hintereinander gesendet werden soll.

bSuppressResponseBuffer: Falls *TRUE* anliegt, wird der interne Puffer **nicht** mit der Antwort vom Baustein `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821) gefüllt.

nDeviceType: Kennung für den Gerätetyp.

Wert	Beschreibung
0	Standardgerät
1	Gerät für <u>Notbeleuchtung</u> [► 32].
2	Gerät für <u>Entladungslampen</u> [► 36].
3	Gerät für Niedervolt-Halogenlampen.
4	Gerät zum Dimmen von Glühlampen.
5	Gerät zur Umwandlung von digitalen Signalen in Gleichspannungssignale.
6	Gerät für <u>Licht emittierende Dioden (LED)</u> [► 36].
7	Gerät für Schaltfunktionen.
8	Gerät zur <u>Farb-/Farbtemperatursteuerung</u> [► 34].
9	Sequenzer.

nOptions: reserviert für zukünftige Erweiterungen.

eDataFrameType: Ausgabeformat [► 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit*, *eDALIV2DataFrameType16Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId      : UDINT;
arrResponseData : ARRAY [0..3] OF BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

arrResponseData: Der empfangende Wert vom DALI-Gerät, falls ein Abfragebefehl aufgerufen wurde.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

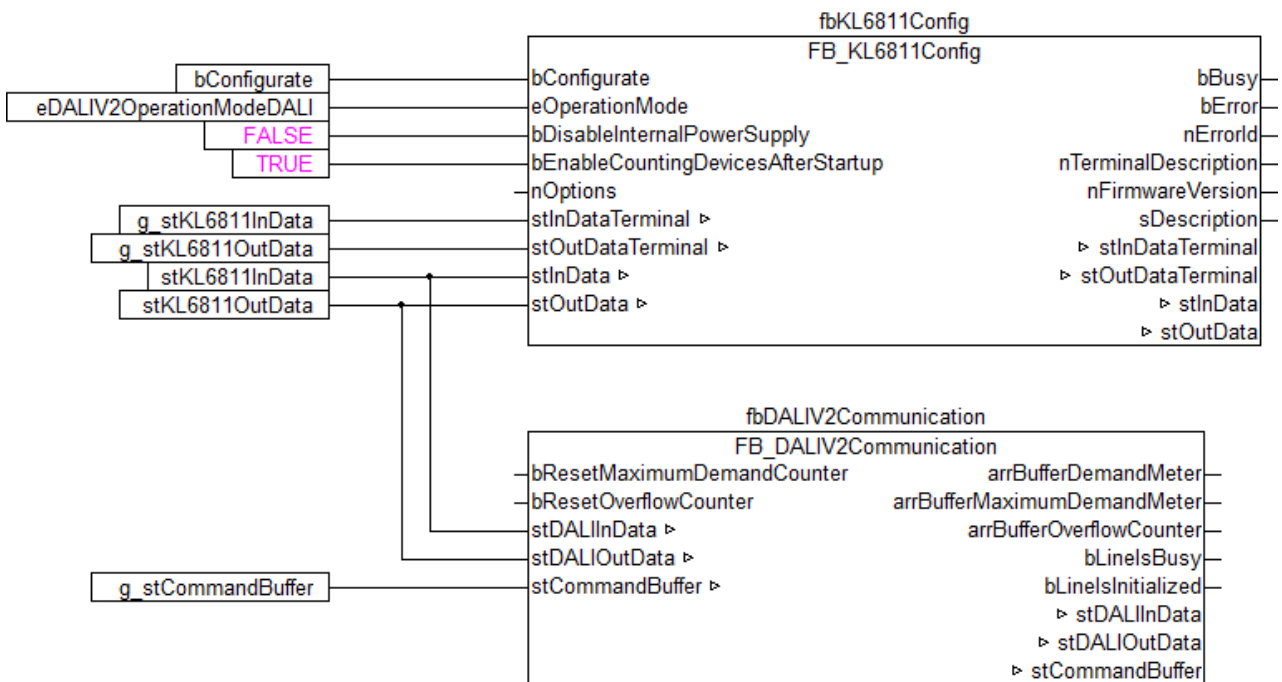
5.1.46 FB_KL6811Config



Dieser Baustein dient zum Konfigurieren der KL6811. Das Konfigurieren wird beim Aufstarten des SPS-Programms ausgeführt oder durch eine positive Flanke am Eingang *bConfigure*. Die Parameter werden in den jeweiligen Register der KL6811 spannungsausfallsicher abgespeichert. Des Weiteren werden aus der KL6811 einige allgemeine Informationen, wie die Version der Firmware, ausgelesen.

Beispiel

Der Baustein wird in der gleichen Task, wie der Baustein `FB_DALIV2Communication()` [► 95] aufgerufen.



Der Baustein `FB_KL6811Config()` ist mit dem Prozessabbild der KL6811 verbunden. Nach Abschluss der Konfiguration erhält der Baustein `FB_DALIV2Communication()` die Prozesswerte der KL6811. Während der Konfiguration können keine DALI-Befehle versendet werden.

Beispieldateien entpacken <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/>

12019293835.zip

VAR_INPUT

```

bConfigure           : BOOL := FALSE;
eOperationMode       : E_DALIV2OperationMode := eDALIV2OperationModeDALI;
bDisableInternalPowerSupply : BOOL := FALSE;
bEnableCountingDevicesAfterStartup : BOOL := FALSE;
nOptions             : DWORD := 0;
    
```

bConfigure: Durch eine positive Flanke an diesem Eingang wird das Konfigurieren der Busklemme gestartet.

eOperationMode: Definiert die Betriebsart der Klemme (DALI oder DSI). Entspricht Register 32 Bit 12 bis 15 der Busklemme.

bDisableInternalPowerSupply: Ist dieser Eingang TRUE, so wird durch die Konfiguration das interne DALI-Netzteil der Klemme deaktiviert. Entspricht Register 32 Bit 3 der Busklemme.

bEnableCountingDevicesAfterStartup: Ist dieser Eingang TRUE, so wird beim Starten der Klemme die Anzahl der DALI-Geräte gezählt. Entspricht Register 32 Bit 4 der Busklemme.

nOptions: reserviert für zukünftige Erweiterungen.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nTerminalDescription : WORD;
nFirmwareVersion : WORD;
sDescription   : STRING;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Durch das erneute Aktivieren des Bausteins über den Eingang *bConfigure* wird der Ausgang wieder auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das erneute Aktivieren des Bausteins über den Eingang *bConfigure* wieder auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nTerminalDescription: Enthält die Klemmenbezeichnung (z.B. 6811). Entspricht Register 8 der Busklemme.

nFirmwareVersion: Enthält die Version der Firmware. Entspricht Register 9 der Busklemme.

sDescription: Klemmenbezeichnung und die Version der Firmware als String (z.B. 'Terminal KL6811 / Firmware 2H').

VAR_IN_OUT

```
stInDataTerminal : ST_DALIV2InData;
stOutDataTerminal : ST_DALIV2OutData;
stInData         : ST_DALIV2InData;
stOutData        : ST_DALIV2OutData;
```

stInDataTerminal: Verweis auf die Struktur [▶ 390] zur Kommunikation mit der KL6811.

stOutDataTerminal: Verweis auf die Struktur [▶ 391] zur Kommunikation mit der KL6811.

stInData: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_DALIV2Communication() [▶ 95] (KL6811).

stOutData: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_DALIV2Communication() [▶ 95] (KL6811).

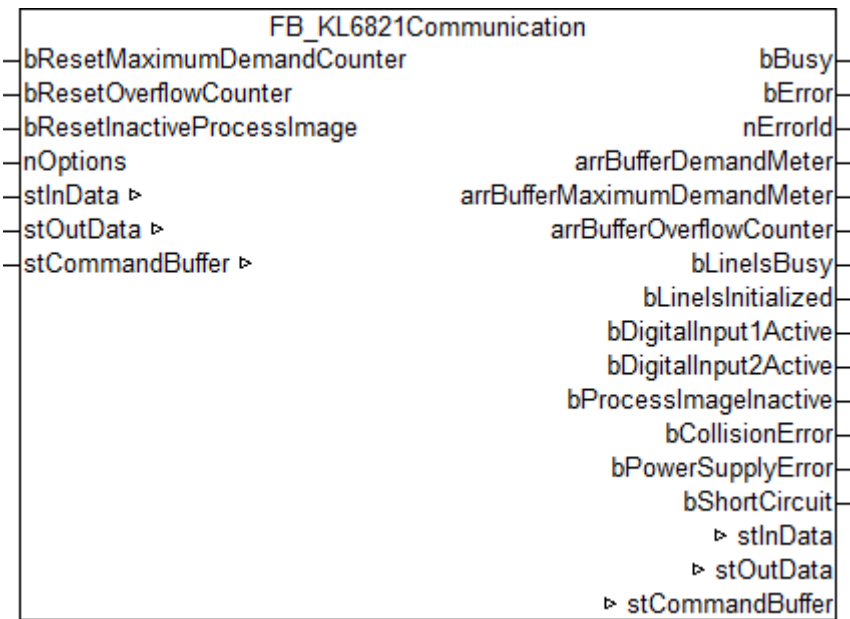
Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2253	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.11.1

Sehen Sie dazu auch

- 📖 E_DALIV2OperationMode [▶ 389]

5.1.47 FB_KL6821Communication



Die Bausteine für die DALI-Befehle greifen nicht direkt auf das Prozessabbild der DALI-Busklemme zu, sondern legen die einzelnen DALI-Befehle in drei verschiedene Puffer ab. Der Baustein FB_KL6821Communication() liest sequentiell die DALI-Befehle aus diesen drei Puffern aus und gibt die DALI-Befehle zur KL6821 weiter. Hierdurch wird sichergestellt, dass nicht mehrere Bausteine gleichzeitig auf das Prozessabbild der Busklemme zugreifen. Jeder dieser drei Puffer wird mit einer anderen Priorität (hoch, mittel oder niedrig) abgearbeitet. Der Anwender der SPS-Bibliothek kann durch den Parameter *eCommandPriority*, den es bei den meisten Bausteinen gibt, beeinflussen mit welcher Priorität der jeweilige DALI-Befehl von dem Baustein FB_KL6821Communication() bearbeitet werden soll.

Die Puffer, in denen die DALI-Befehle abgelegt werden, sind alle in einer Variablen vom Typ ST_DALIV2CommandBuffer enthalten. Pro KL6821 gibt es eine Instanz vom Baustein FB_KL6821Communication() und eine Variable vom Typ ST_DALIV2CommandBuffer. Der Baustein FB_KL6821Communication() sollte, wenn möglich, in einer separaten, schnelleren Task aufgerufen werden.

Über die Ausgänge des Bausteins kann ermittelt werden, wie stark die Puffer ausgelastet sind. Hierzu werden drei Arrays ausgegeben, bei dem jedes Element (0, 1 oder 2) für einen der drei Buffer (hoch, mittel oder niedrig) steht. Wenn Sie feststellen, dass einer der drei Puffer regelmäßig überläuft, sollten Sie folgende Maßnahmen in Betracht ziehen:

- Wie stark sind die einzelnen SPS-Task ausgelastet? Der TwinCAT System Manager bietet zur Analyse entsprechende Hilfsmittel an.
- Versuchen Sie die Zykluszeit der Task, in der der Baustein FB_KL6821Communication() aufgerufen wird, zu verringern. Der Wert sollte nicht größer als 6 ms sein, optimal sind 2 ms.
- Überprüfen Sie die Zykluszeit der SPS-Task, in der die Bausteine für die einzelnen DALI-Befehle aufgerufen werden. Dieser Wert sollte zwischen 10 ms und 60 ms liegen.
- Vermeiden Sie möglichst das Pollen (regelmäßiges Auslesen) von Werten. Lesen Sie nur dann Werte aus, wenn diese auch benötigt werden.
- Verteilen Sie die einzelnen Vorschaltgeräte gleichmäßig auf mehrere DALI-Linien. Da pro SPS-Zyklus mehrere DALI-Linien gleichzeitig bearbeitet werden, erhöht sich hierdurch der Datendurchsatz insgesamt.

VAR_INPUT

```
bResetMaximumDemandCounter : BOOL;
bResetOverflowCounter       : BOOL;
bResetInactiveProcessImage  : BOOL;
nOptions                     : DWORD := 0;
```

bResetMaximumDemandCounter: Eine positive Flanke setzt den gespeicherten Wert der maximalen Befehlspeicher-Auslastung, *arrBufferMaximumDemandMeter* (0 - 100%, siehe VAR_OUTPUT), zurück.

bResetOverflowCounter: Eine positive Flanke setzt den gespeicherten Wert der Anzahl der Befehlspeicher-Überläufe, *arrBufferOverflowCounter* (siehe VAR_OUTPUT), zurück.

bResetInactiveProcessImage: Eine positive Flanke hebt die Sperrung des Prozessabbildes der Klemme wieder auf. Die Ausgänge *bProcessImageInactive*, *bDigitalInput1Active* und *bDigitalInput2Active* werden wieder auf FALSE gesetzt. Die Sperrung wird aktiviert, sobald einer der beiden digitalen Eingänge an der Klemme betätigt wurde.

nOptions: reserviert für zukünftige Erweiterungen.

VAR_OUTPUT

```

bBusy           : BOOL;
bError          : BOOL;
nErrorId        : UDINT;
arrBufferDemandMeter      : ARRAY [0..2] OF BYTE;
arrBufferMaximumDemandMeter : ARRAY [0..2] OF BYTE;
arrBufferOverflowCounter  : ARRAY [0..2] OF UINT;
bLineIsBusy       : BOOL;
bLineIsInitialized : BOOL;
bDigitalInput1Active : BOOL;
bDigitalInput2Active : BOOL;
bProcessImageInactive : BOOL;
bCollisionError    : BOOL;
bPowerSupplyError  : BOOL;
bShortCircuit      : BOOL;

```

bBusy: Der Ausgang wird gesetzt, sobald der Baustein einen Befehl verarbeitet und bleibt so lange aktiv, bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung des Bausteins ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Siehe [Fehlercodes](#) [► 391].

arrBufferDemandMeter: Belegung des jeweiligen Puffers (0 - 100%).

arrBufferMaximumDemandMeter: Bisherige maximale Auslastung des jeweiligen Puffers (0 - 100%).

arrBufferOverflowCounter: Bisherige Anzahl der Pufferüberläufe.

bLineIsBusy: Solange der Baustein *FB_KL6821Communication()* aktiv ist, ist dieser Ausgang gesetzt.

bLineIsInitialized: Wird der Baustein das erste Mal aufgerufen (z.B. beim Starten der Steuerung), so wird eine Initialisierung durchgeführt. Während dieser Zeit können keine DALI-Befehle bearbeitet werden.

bDigitalInput1Active: Der digitale Eingang 1 an der Klemme wurde betätigt oder ist betätigt (siehe auch Klemmendokumentation). Der Ausgang *bProcessImageInactive* wird gesetzt und es können keine weiteren DALI-Befehle über die Steuerung bearbeitet werden.

bDigitalInput2Active: Der digitale Eingang 2 an der Klemme wurde betätigt oder ist betätigt (siehe auch Klemmendokumentation). Der Ausgang *bProcessImageInactive* wird gesetzt und es können keine weiteren DALI-Befehle über die Steuerung bearbeitet werden.

bProcessImageInactive: Einer der beiden digitalen Eingänge wurde an der Klemme betätigt. Es können keine weiteren DALI-Befehle über die Steuerung bearbeitet werden. Über den Eingang *bResetInactiveProcessImage* muss die Sperrung wieder freigeschaltet werden.

bCollisionError: Es wurde eine Datenkollision auf dem DALI-Bus erkannt, während ein Befehl versendet wurde.

bPowerSupplyError: Die KL6821 hat einen Fehler am internen DALI-Netzteil erkannt.

bShortCircuit: Kurzschluss auf dem DALI-Bus.

VAR_IN_OUT

```
stInData      : ST_KL6821InData;
stOutData     : ST_KL6821OutData;
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stInData: Struktur [► 390] im Eingangsprozessabbild der KL6821. Sie dient zur Kommunikation von der KL6821 zur SPS. Bei Verwendung des [FB_KL6821Config](#) [► 106] ist diese Struktur mit dem Parameter *stInData* verbunden.

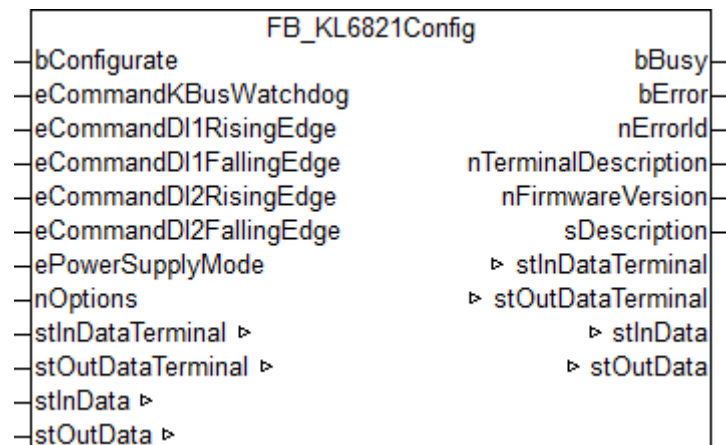
stOutData: Struktur [► 391] im Ausgangsprozessabbild der KL6821. Sie dient zur Kommunikation von der SPS zur KL6821. Bei Verwendung des [FB_KL6821Config](#) [► 106] ist diese Struktur mit dem Parameter *stOutData* verbunden.

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den DALI-Bausteinen.

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

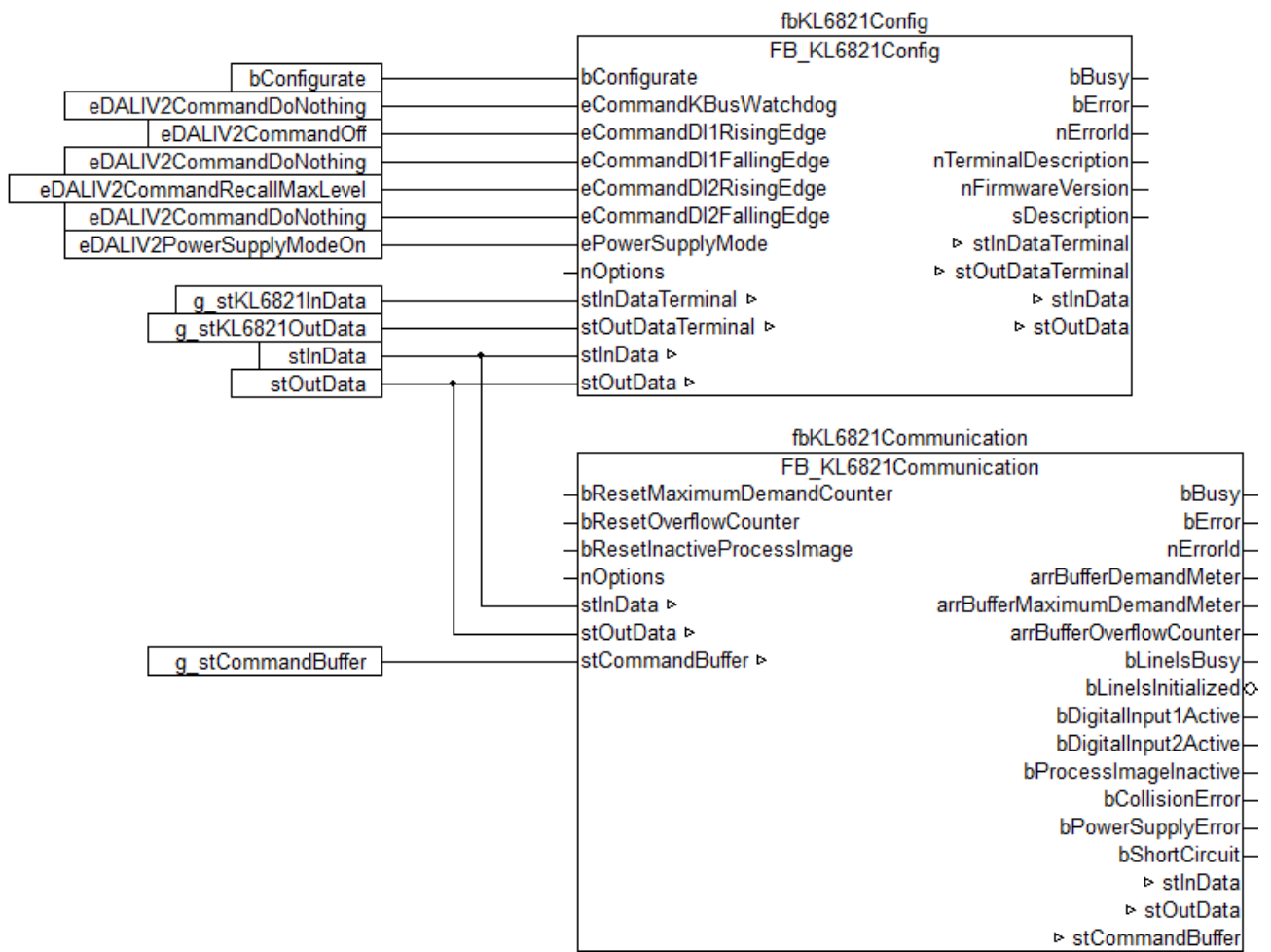
5.1.48 FB_KL6821Config



Dieser Baustein dient zum Konfigurieren der KL6821. Das Konfigurieren wird beim Aufstarten des SPS-Programms ausgeführt oder durch eine positive Flanke am Eingang *bConfigure*. Die Parameter werden in dem jeweiligen Register der KL6821 spannungsausfallsicher abgespeichert. Des Weiteren werden aus der KL6821 einige allgemeine Informationen, wie die Version der Firmware, ausgelesen.

Beispiel

Der Baustein wird in der gleichen Task, wie der Baustein [FB_KL6821Communication\(\)](#) [► 104] aufgerufen.



Der Baustein FB_KL6821Config() ist mit dem Prozessabbild der KL6821 verbunden. Nach Abschluss der Konfiguration erhält der Baustein FB_KL6821Communication() die Prozesswerte der KL6821. Während der Konfiguration können keine DALI-Befehle versendet werden.

Beispieldateien entpacken <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019295243.zip>

12019295243.zip

VAR_INPUT

```
bConfigure : BOOL := FALSE;
eCommandKBusWatchdog : E_DALIV2ConfigurationCommands := eDALIV2CommandDoNothing;
eCommandDI1RisingEdge : E_DALIV2ConfigurationCommands := eDALIV2CommandOff;
eCommandDI1FallingEdge : E_DALIV2ConfigurationCommands := eDALIV2CommandDoNothing;
eCommandDI2RisingEdge : E_DALIV2ConfigurationCommands := eDALIV2CommandRecallMaxLevel;
eCommandDI2FallingEdge : E_DALIV2ConfigurationCommands := eDALIV2CommandDoNothing;
ePowerSupplyMode : E_DALIV2PowerSupplyMode := eDALIV2PowerSupplyModeOn;
nOptions : DWORD := 0;
```

bConfigure: Durch eine positive Flanke an diesem Eingang wird das Konfigurieren der Busklemme gestartet.

eCommandKBusWatchdog: Definiert den DALI-Befehl [▶ 387], der versendet wird, sobald die Busklemme über den K-Bus nicht mehr angesprochen wird.

eCommandDI1RisingEdge: Definiert den DALI-Befehl, der versendet wird, sobald am Eingang 1 der Busklemme eine steigende Flanke erkannt wird.

eCommandDI1FallingEdge: Definiert den DALI-Befehl, der versendet wird, sobald am Eingang 1 der Busklemme eine fallende Flanke erkannt wird.

eCommandDI2RisingEdge: Definiert den DALI-Befehl, der versendet wird, sobald am Eingang 2 der Busklemme eine steigende Flanke erkannt wird.

eCommandDI2FallingEdge: Definiert den DALI-Befehl, der versendet wird, sobald am Eingang 2 der Busklemme eine fallende Flanke erkannt wird..

ePowerSupplyMode: Definiert die Betriebsart [▶ 389] des internen DALI-Netzteils.

nOptions: reserviert für zukünftige Erweiterungen.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nTerminalDescription : WORD;
nFirmwareVersion : WORD;
sDescription    : STRING;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Durch das erneute Aktivieren des Bausteins über den Eingang *bConfigure* wird der Ausgang wieder auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das erneute Aktivieren des Bausteins über den Eingang *bConfigure* wieder auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nTerminalDescription: Enthält die Klemmenbezeichnung (z.B. 6821). Entspricht Register 8 der Busklemme.

nFirmwareVersion: Enthält die Version der Firmware. Entspricht Register 9 der Busklemme.

sDescription: Klemmenbezeichnung und die Version der Firmware als String (z.B. 'Terminal KL6821 / Firmware 2H').

VAR_IN_OUT

```
stInDataTerminal : ST_KL6821InData;
stOutDataTerminal : ST_KL6821OutData;
stInData         : ST_KL6821InData;
stOutData        : ST_KL6821OutData;
```

stInDataTerminal: Verweis auf die Struktur [▶ 391] zur Kommunikation mit der KL6821.

stOutDataTerminal: Verweis auf die Struktur [▶ 391] zur Kommunikation mit der KL6821.

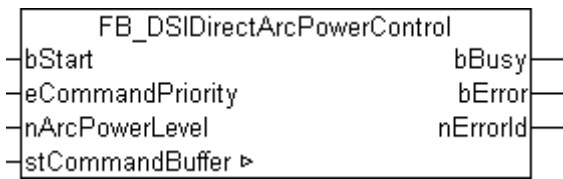
stInData: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

stOutData: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.49 FB_DSIDirectArcPowerControl



Der Parameter *nArcPowerLevel* gibt die Helligkeit vor, auf welche die Lampe geschaltet wird.

nArcPowerLevel	Bemerkung
0	Lampe wird ausgeschaltet
1	Lampe wird auf die minimale Helligkeit geschaltet
255	Lampe wird auf die maximale Helligkeit geschaltet

Vorschaltgeräte mit DSI-Schnittstelle haben keine Kurzadresse. Alle Vorschaltgeräte an einer DSI-Linie erhalten den gleichen Wert.

Beachten Sie bitte, dass die KL6811 in den DSI-Mode umgeschaltet werden muss. Einzelheiten hierzu finden Sie in der Anleitung zur KL6811. Die KL6821 unterstützt kein DSI.

DSI-Vorschaltgeräte und DALI-Vorschaltgeräte können nicht an einer Linie gemischt werden. Es ist aber möglich, mehrere KL6811 an einem Controller in je unterschiedlichen Betriebsarten (DSI/DALI) zu betreiben.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nArcPowerLevel  : BYTE;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nArcPowerLevel: Lampenleistungswert (0 ... 255).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```

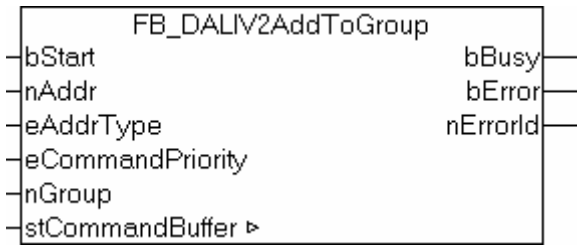
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_DALIV2Communication()` [▶ 95] (KL6811).

Sehen Sie dazu auch

- FB_KL6821Communication [▶ 104]

5.1.50 FB_DALIV2AddToGroup



Die adressierten Vorschaltgeräte werden zu der entsprechenden Gruppe (*nGroup*) hinzugefügt. Eine gültige Gruppennummer liegt im Wertebereich von 0 bis 15.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nGroup      : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nGroup: Gruppennummer (0-15).

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

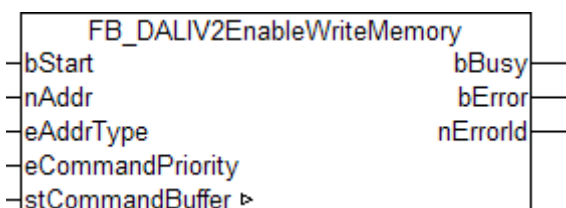
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.51 FB_DALIV2EnableWriteMemory



Schaltet den Schreibzugriff per `FB_DALIV2WriteMemoryLocation()` [► 170] auf den internen Speicher des Vorschaltgerätes frei.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

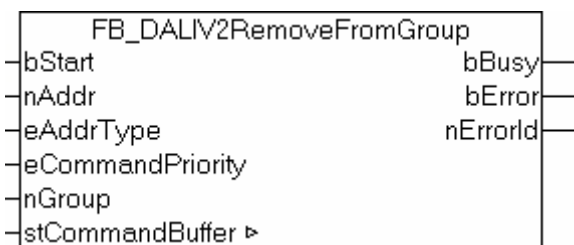
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

5.1.52 FB_DALIV2RemoveFromGroup



Ein oder mehrere Vorschaltgeräte werden aus einer Gruppe entfernt. Eine gültige Gruppennummer liegt im Wertebereich von 0 bis 15.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nGroup     : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nGroup: Gruppe, aus der die Vorschaltgeräte entfernt werden sollen.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

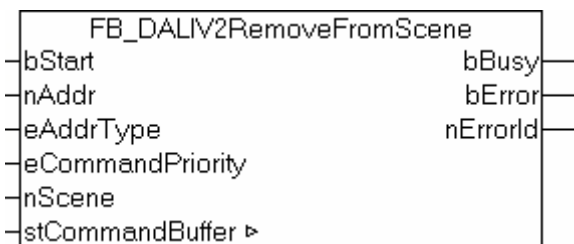
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.53 FB_DALIV2RemoveFromScene



Ein oder mehrere Vorschaltgeräte werden aus einer Szene entfernt. Eine gültige Szenennummer liegt im Wertebereich von 0 bis 15.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nScene     : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

nScene: Szene, aus der die Vorschaltgeräte entfernt werden sollen.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.54 FB_DALIV2Reset



Mit diesem Baustein werden alle Variablen des Vorschaltgerätes auf die Standardwerte zurückgesetzt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.55 FB_DALIV2SetShortAddress



Mit Hilfe dieses Bausteins erhalten ein oder mehrere Vorschaltgeräte eine (neue) Kurzadresse. Gültige Kurzadressen liegen im Wertebereich von 0 bis 63. Wird als Kurzadresse 255 übergeben, so wird die Kurzadresse im Vorschaltgerät gelöscht.

Wollen Sie einem Gerät eine Kurzadresse geben, welches noch keine Kurzadresse hat, so müssen Sie den Befehl als Broadcast versenden (*eAddrType* = *eDALIV2AddrTypeBroadcast*). Hierdurch bekommen alle Vorschaltgeräte, die an der DALI-Klemme angeschlossen sind, die Kurzadresse *nAddr*. Also auch die Vorschaltgeräte, die bisher noch keine Kurzadresse hatten.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nNewShortAddress : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nNewShortAddress: Neue Kurzadresse (0-15) oder Maske (255).

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.56 FB_DALIV2StoreActualLevelInDTR



Der Baustein schreibt den aktuellen Lampenleistungswert in das DTR. Hierbei wird der aktuelle Lampenleistungswert nicht verändert.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[▶ 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.57 FB_DALIV2StoreDTRAsFadeRate



Schreibt den Wert des DTR in die Variable FADE RATE [▶ 371]. Der mögliche Wertebereich beträgt 1 bis 15.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.58 FB_DALIV2StoreDTRAsFadeTime



Der Baustein schreibt den Wert des DTR in die Variable FADE TIME [▶ 371] (Stufenzeit). Der mögliche Wertebereich beträgt 0 bis 15.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.59 FB_DALIV2StoreDTRAsMaxLevel



Dieser Baustein schreibt den Wert des DTR in die Variable MAX LEVEL [▶ 371]. Wird ein Wert kleiner als MIN LEVEL angegeben, so wird der Wert auf MIN LEVEL gesetzt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.60 FB_DALIV2StoreDTRAsMinLevel



Dieser Baustein schreibt den Wert des DTR in die Variable MIN LEVEL [▶ 371] (minimal erlaubte Lampenleistung). Wird ein Wert größer als MAX LEVEL angegeben, so wird der Wert auf MAX LEVEL gesetzt. Ist der Wert kleiner als PHYSICAL MIN LEVEL [▶ 373], so wird MIN LEVEL auf PHYSICAL MIN LEVEL gesetzt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.61 FB_DALIV2StoreDTRAsPowerOnLevel



Dieser Baustein schreibt den Wert des DTR in die Variable [POWER ON LEVEL](#) [► 370].

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.62 FB_DALIV2StoreDTRAsScene



Der Inhalt des DTR wird zu der angegebenen Szene als Lampenleistungswert gespeichert. Der Wertebereich der Szenennummer geht von 0 bis 15.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nScene      : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nScene: Szene, dessen Lampenleistungswert geändert werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

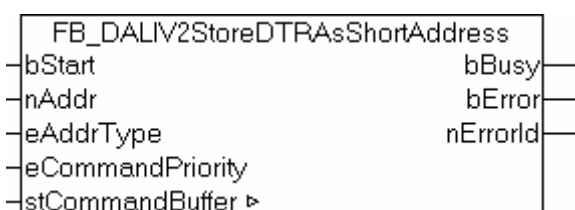
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.63 FB_DALIV2StoreDTRAsShortAddress



Der Inhalt des DTR (Data Transfer Register) wird bei dem entsprechenden Vorschaltgerät als Kurzadresse abgespeichert. Die Struktur des DTR ist 0AAA AAA1 (A: Signifikantes Adressbit) oder 1111 1111 (Maske). Steht 1111 1111 im DTR, so wird die Kurzadresse aus dem Vorschaltgerät entfernt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

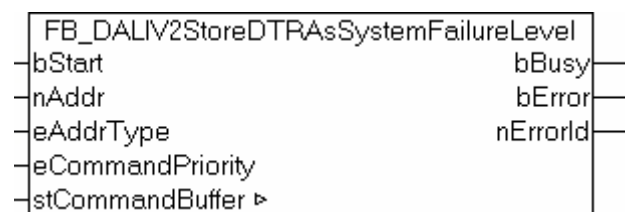
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.64 FB_DALIV2StoreDTRAsSystemFailureLevel



Dieser Baustein schreibt den Wert des DTR in die Variable SYSTEM FAILURE LEVEL [▶ 370].

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

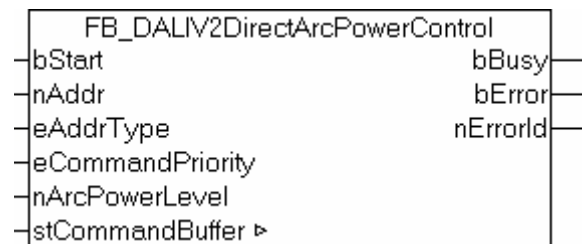
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.65 FB_DALIV2DirectArcPowerControl



Liegt der Parameter *nArcPowerLevel* außerhalb von MAX VALUE [▶ 371] oder MIN VALUE, so wird die Lampe auf den entsprechend kleinsten oder größten Wert geschaltet. Ist die Lampe ausgeschaltet, so wird diese durch den Befehl eingeschaltet.

Die Geschwindigkeit Limit welcher der angegebene Sollwert erreicht werden soll, wird durch die Variable FADE TIME [▶ 371] angegeben.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nArcPowerLevel : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nArcPowerLevel: Lampenleistungswert.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.66 FB_DALIV2Down



Die Lampe wird für 200ms dunkler gedimmt. Ist die Lampenleistung bereits auf den Wert [MIN LEVEL](#) [▶ 371], so wird die Helligkeit nicht verändert. Die Lampe wird mit diesem Befehl nicht ausgeschaltet.

Die Geschwindigkeit, mit der innerhalb der 200 ms gedimmt werden soll, wird durch die Variable [FADE RATE](#) [▶ 371] angegeben.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

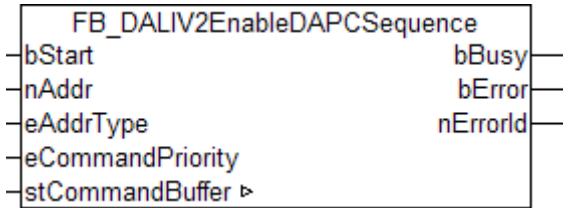
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.67 FB_DALIV2EnableDAPCSequence



Mit diesem Befehl wird eine *Direct Arc Power Control* (DAPC) Sequenz gestartet. Nach diesem Befehl müssen DirectArcPowerControl-Befehle mit dem Baustein [FB_DALIV2DirectArcPowerControl\(\)](#) [► 122] versendet werden. Zwischen den einzelnen Befehlen dürfen nicht mehr als 200 ms liegen, andernfalls wird die Sequenz beendet.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

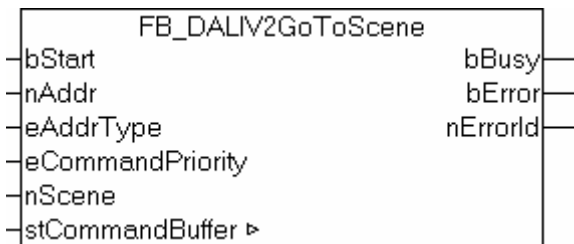
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```


stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.68 FB_DALIV2GoToScene



Dieser Baustein setzt den Lampenleistungswert auf den für die Szene *nScene* gespeicherten Wert. Wenn das Vorschaltgerät nicht zur Szene gehört, dann bleibt der Lampenleistungswert unverändert. Ist die Lampe ausgeschaltet, so wird diese mit dem Befehl eingeschaltet.

Die Geschwindigkeit, mit der der Lampenleistungswert erreicht werden soll, wird durch die Variable [FADE TIME \[► 371\]](#) angegeben.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nScene      : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[► 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[► 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nScene: Szene, die aufgerufen werden soll (0 - 15).

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

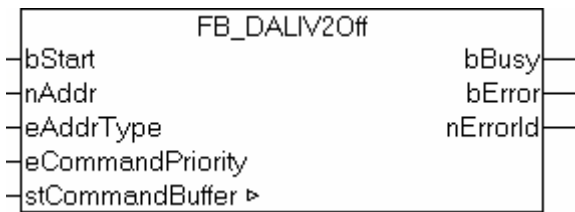
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.69 FB_DALIV2Off



Die DALI-Lampen werden unmittelbar ausgeschaltet.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

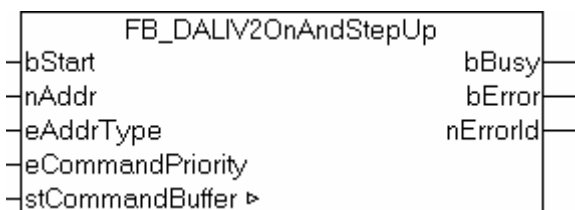
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.70 FB_DALIV2OnAndStepUp



Dieser Baustein setzt den aktuellen Lampenleistungswert unmittelbar um eine Stufe höher. Ist die Lampe ausgeschaltet, so wird diese eingeschaltet und auf MIN_LEVEL [▶ 371] gesetzt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

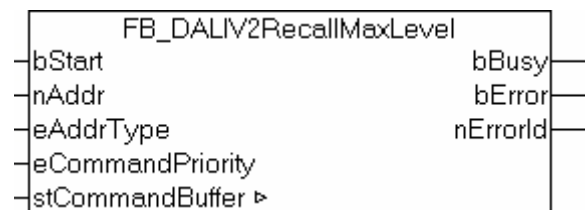
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.71 FB_DALIV2RecallMaxLevel



Dieser Baustein setzt den aktuellen Lampenleistungswert stufenlos auf MAX LEVEL [▶ 371]. Wenn die Lampe ausgeschaltet ist, wird sie mit diesem Befehl eingeschaltet.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

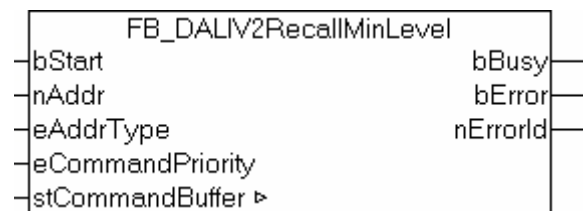
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.72 FB_DALIV2RecallMinLevel



Dieser Baustein setzt den aktuellen Lampenleistungswert stufenlos auf MIN LEVEL [▶ 371]. Wenn die Lampe ausgeschaltet ist, wird sie mit diesem Befehl eingeschaltet.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.73 FB_DALIV2StepDown



Dieser Baustein setzt den aktuellen Lampenleistungswert unmittelbar um eine Stufe tiefer. Mit diesem Befehl werden die Lampen nicht ausgeschaltet. Hat der Lampenleistungswert bereits [MIN LEVEL \[► 371\]](#) erreicht, so wird der Wert nicht weiter verringert.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[► 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[► 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.74 FB_DALIV2StepDownAndOff



Dieser Baustein setzt den aktuellen Lampenleistungswert unmittelbar um eine Stufe tiefer. Hat der Lampenleistungswert bereits [MIN LEVEL](#) [▶ 371] erreicht, so wird die Lampe ausgeschaltet.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

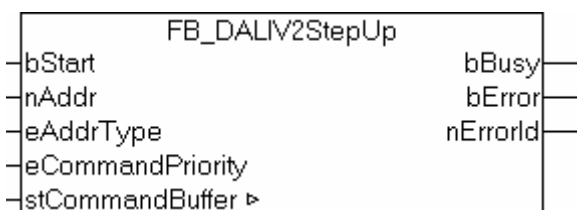
VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.75 FB_DALIV2StepUp



Dieser Baustein setzt den aktuellen Lampenleistungswert unmittelbar um eine Stufe höher. Mit diesem Befehl werden die Lampen nicht eingeschaltet. Hat der Lampenleistungswert bereits [MAX LEVEL \[► 371\]](#) erreicht, so wird der Wert nicht weiter erhöht.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[► 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[► 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

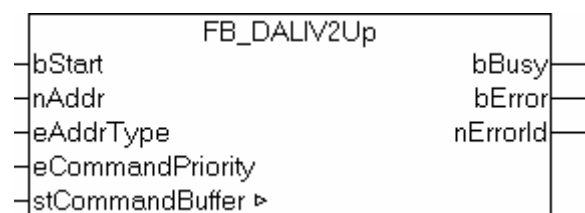
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.76 FB_DALIV2Up



Die Lampe wird für 200ms heller gedimmt. Ist die Lampenleistung bereits auf den Wert [MAX LEVEL \[► 371\]](#), so wird die Helligkeit nicht verändert. Die Lampe wird mit diesem Befehl nicht eingeschaltet.

Die Geschwindigkeit, mit der innerhalb der 200 ms gedimmt werden soll, wird durch die Variable [FADE RATE \[► 371\]](#) angegeben.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```


bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

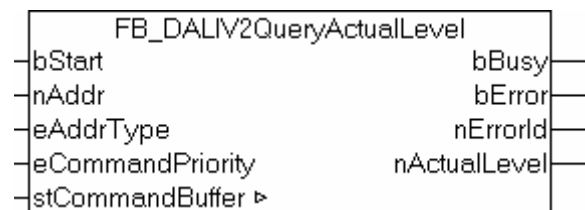
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.77 FB_DALIV2QueryActualLevel



Die Variable ACTUAL DIM LEVEL [▶ 370] (aktuelle Lampenleistung) wird vom Vorschaltgerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nActualLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

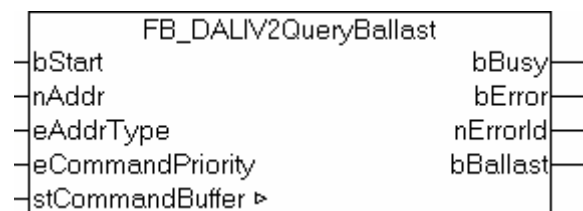
nActualLevel: Lampenleistung (0-254).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.78 FB_DALIV2QueryBallast



Der Baustein gibt Auskunft, ob ein bestimmtes Vorschaltgerät betriebsbereit ist.

Mit diesem Befehl kann sehr einfach festgestellt werden, ob sich an einer DALI-Linie überhaupt Vorschaltgeräte befinden. Hierzu wird der Baustein mit dem Parameter *eAddrType = eDALIV2AddrTypeBroadcast* aufgerufen. Sind die Ausgänge *bBallast* FALSE und *nError* 0, so befindet sich an der DALI-Linie kein Vorschaltgerät. Ist der Ausgang *nError* 0 und der Ausgang *bBallast* TRUE, so befindet sich genau ein Vorschaltgerät an der DALI-Linie. Wenn mehrere Vorschaltgeräte angeschlossen sind, so wird an *nError* eine 5 (mehrere Vorschaltgeräte haben geantwortet) zurückgegeben. Hierbei ist es gleichgültig ob die Vorschaltgeräte Kurzadressen haben oder nicht.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bBallast    : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

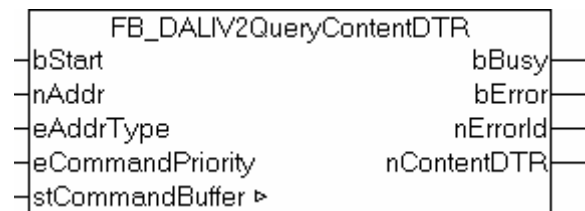
bBallast: Ist der Ausgang aktiv, so ist das entsprechende Vorschaltgerät betriebsbereit.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.79 FB_DALIV2QueryContentDTR



Der Inhalt des DTR (Data Transfer Register) wird aus dem Vorschaltgerät ausgelesen.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[▶ 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nContentDTR : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

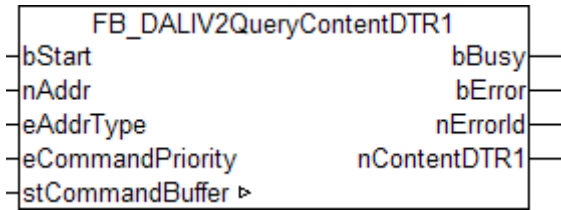
nContentDTR: Inhalt des DTR (Data Transfer Register).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.80 FB_DALIV2QueryContentDTR1



Der Inhalt des DTR1 (Data Transfer Register 1) wird aus dem Vorschaltgerät ausgelesen.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[▶ 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nContentDTR1 : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

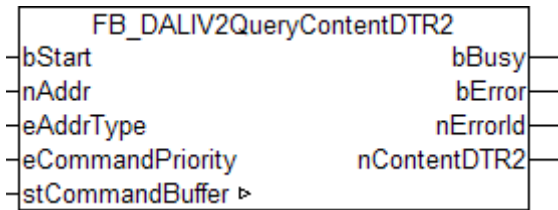
nContentDTR1: Inhalt des DTR1 (Data Transfer Register 1).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.81 FB_DALIV2QueryContentDTR2



Der Inhalt des DTR2 (Data Transfer Register 2) wird aus dem Vorschaltgerät ausgelesen.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nContentDTR2 : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

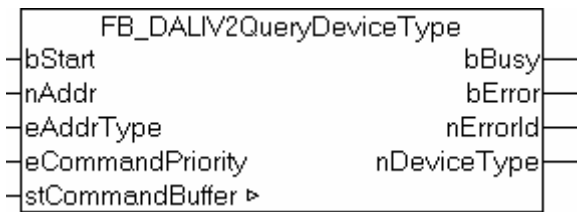
nContentDTR2: Inhalt des DTR2 (Data Transfer Register 2).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.82 FB_DALIV2QueryDeviceType



Es wird aus dem Vorschaltgerät der Gerätetyp ([DEVICE TYPE](#) [▶ 373]) ausgelesen. Folgende Gerätetypen sind laut Norm IEC 62386 definiert:

Wert	Beschreibung
0	Standardgerät
1	Gerät für Notbeleuchtung [▶ 32]
2	Gerät für Entladungslampen [▶ 36]
3	Gerät für Niedervolt-Halogenlampen
4	Gerät zum Dimmen von Glühlampen
5	Gerät zur Umwandlung von digitalen Signalen in Gleichspannungssignale
6	Gerät für Licht emittierende Dioden (LED) [▶ 36]
7	Gerät für Schaltfunktionen
8	Gerät zur Farb-/Farbtemperatursteuerung [▶ 34]
9	Sequenzer

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nDeviceType : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

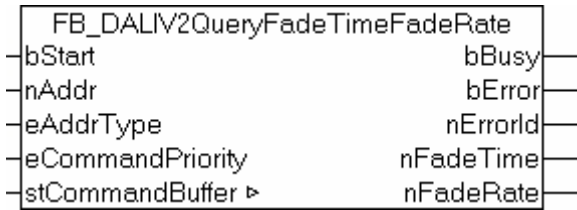
nDeviceType: Kennung für den Gerätetyp (siehe Tabelle oben).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.83 FB_DALIV2QueryFadeTimeFadeRate



Die Variablen [FADE TIME \[► 371\]](#) (Stufenzeit) und [FADE RATE \[► 371\]](#) (Stufengeschwindigkeit) werden aus dem Vorschaltgerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse, Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nFadeTime  : BYTE;
nFadeRate  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

nFadeTime: Stufenzeit (0 bis 15).

nFadeRate: Stufengeschwindigkeit (1 bis 15).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

Sehen Sie dazu auch

- 📖 [E_DALIV2AddrType \[► 387\]](#)
- 📖 [E_DALIV2CommandPriority \[► 387\]](#)

5.1.84 FB_DALIV2QueryGroups



Die Variablen GROUP 0-7 und GROUP 8-15 [▶ 372] werden aus dem Vorschaltgerät ausgelesen und zu einen 16-Bit-Wert verknüpft. Jedes Bit repräsentiert eine Gruppe. Bit 0 Gruppe 0 und Bit 15 Gruppe 15. Ist das Bit gesetzt, so gehört das Vorschaltgerät der entsprechenden Gruppe an.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nGroups     : WORD;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

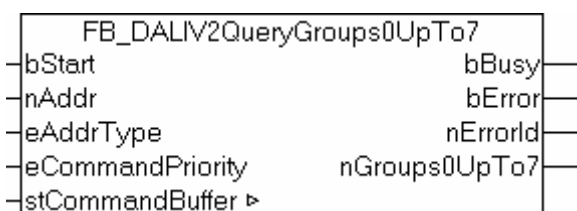
nGroups: Gruppenzugehörigkeit.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.85 FB_DALIV2QueryGroups0UpTo7



Die Variable GROUP 0-7 [▶ 372] wird aus dem Vorschaltgerät ausgelesen und zu einen 8-Bit-Wert verknüpft. Jedes Bit repräsentiert eine Gruppe. Bit 0 Gruppe 0 und Bit 7 Gruppe 7. Ist das Bit gesetzt, so gehört das Vorschaltgerät der entsprechenden Gruppe an.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nGroups0UpTo7 : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

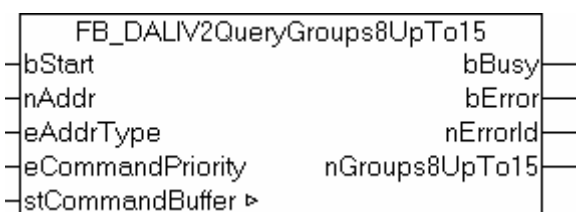
nGroups0UpTo7: Gruppenzugehörigkeit.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.86 FB_DALIV2QueryGroups8UpTo15



Die Variable GROUP 8-15 [▶ 372] wird aus dem Vorschaltgerät ausgelesen und zu einen 8-Bit-Wert verknüpft. Jedes Bit repräsentiert eine Gruppe. Bit 0 Gruppe 8 und Bit 7 Gruppe 15. Ist das Bit gesetzt, so gehört das Vorschaltgerät der entsprechenden Gruppe an.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nGroups8UpTo15 : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

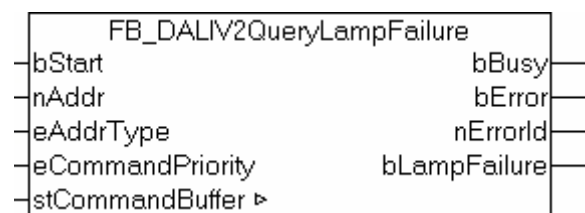
nGroups8UpTo15: Gruppenzugehörigkeit.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.87 FB_DALIV2QueryLampFailure



Der Baustein gibt Auskunft, ob ein bestimmtes Vorschaltgerät ein Lampenproblem hat.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bLampFailure : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

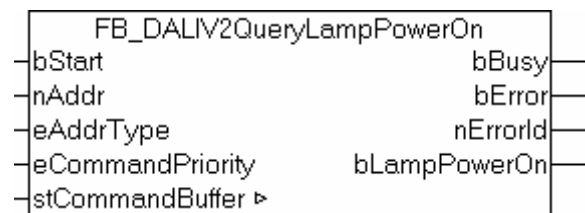
bLampFailure: Ist der Ausgang aktiv, so hat das entsprechende Vorschaltgerät einen Lampenausfall.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [► 95] (KL6811) oder FB_KL6821Communication() [► 104] (KL6821).

5.1.88 FB_DALIV2QueryLampPowerOn



Der Baustein gibt Auskunft, ob die Lampe von einem bestimmten Vorschaltgerät eingeschaltet ist.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bLampPowerOn : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

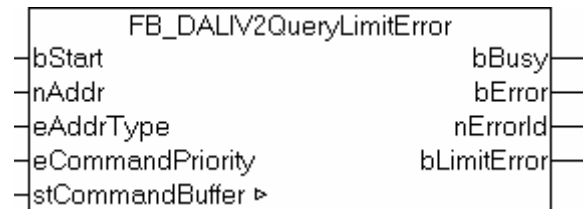
bLampPowerOn: Ist der Ausgang aktiv, so ist die Lampe des entsprechenden Vorschaltgerätes eingeschaltet.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.89 FB_DALIV2QueryLimitError



Der Baustein gibt Auskunft, ob der letzte Lampenleistungswert an einem bestimmten Vorschaltgerät nicht eingehalten wurde, da er über [MAX LEVEL](#) [▶ 371] oder unter [MIN LEVEL](#) lag.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
bLimitError : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

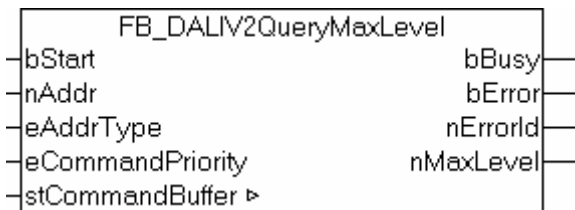
bLimitError: Ist der Ausgang aktiv, so wurde der letzte Lampenleistungswert nicht eingehalten.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.90 FB_DALIV2QueryMaxLevel



Die Variable [MAX LEVEL \[► 371\]](#) (maximal erlaubte Lampenleistung) wird aus dem Vorschaltgerät ausgelesen. Dieser Wert legt die Grenze der Lampenleistungsbefehle nach oben fest.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[► 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[► 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nMaxLevel  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

nMaxLevel: maximal erlaubte Lampenleistung (0 - 254).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.91 FB_DALIV2QueryMinLevel



Die Variable MIN LEVEL [▶ 371] (minimal erlaubte Lampenleistung) wird aus dem Vorschaltgerät ausgelesen. Dieser Wert legt die Grenze der Lampenleistungsbefehle nach unten fest.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nMinLevel   : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

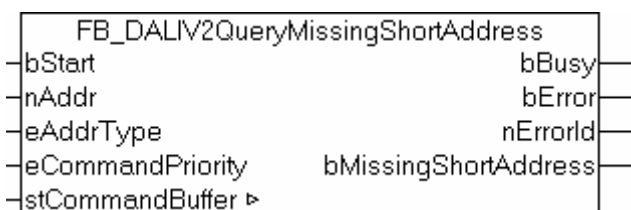
nMinLevel: minimal erlaubte Lampenleistung (0 - 254).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.92 FB_DALIV2QueryMissingShortAddress



Der Baustein gibt Auskunft, ob ein bestimmtes Vorschaltgerät keine Kurzadresse besitzt.

Mit diesem Befehl kann festgestellt werden, ob sich an einer DALI-Linie Vorschaltgeräte ohne Kurzadresse befinden. Hierzu wird der Baustein mit dem Parameter `eAddrType = eDALIV2AddrTypeBroadcast` aufgerufen. Sind die Ausgänge `bMissingShortAddress` FALSE und `nError` 0, so haben alle Vorschaltgeräte eine gültige Kurzadresse. Ist der Ausgang `nError` 0 und der Ausgang `bMissingShortAddress` TRUE, so hat genau ein Vorschaltgerät keine Kurzadresse. Wenn mehrere Vorschaltgeräte keine Kurzadresse haben, so wird an `nError` eine 5 (mehrere Vorschaltgeräte haben geantwortet) zurückgegeben.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bMissingShortAddress : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

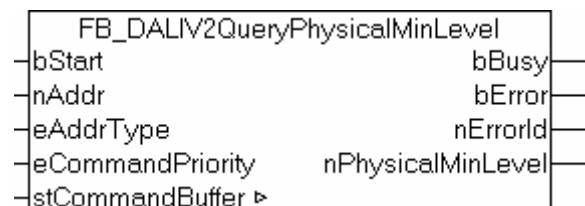
bMissingShortAddress: Ist der Ausgang aktiv, so besitzt das entsprechende Vorschaltgerät keine Kurzadresse.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.93 FB_DALIV2QueryPhysicalMinLevel



Die Variable PHYSICAL MIN LEVEL [▶ 373] (physikalisch kleinstmögliche Lampenleistung) wird aus dem Vorschaltgerät ausgelesen. Der Wert kann nur gelesen werden und wird vom Hersteller fest vorgegeben.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nPhysicalMinLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

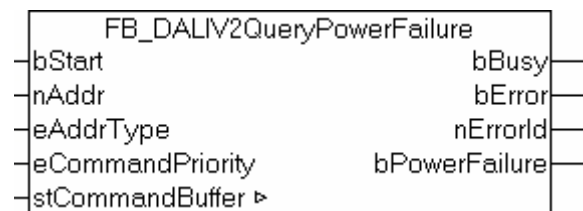
nPhysicalMinLevel: physikalisch kleinstmögliche Lampenleistung (0 - 254).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.94 FB_DALIV2QueryPowerFailure



Abfrage, ob das Vorschaltgerät seit dem Einschalten keinen Reset oder keinen Lampenleistungssteuerbefehl empfangen hat.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ [387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bPowerFailure : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ [391](#)].

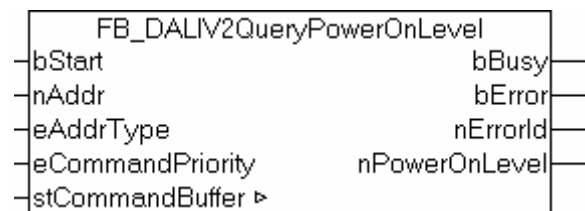
bPowerFailure: Ist der Ausgang aktiv, so wurde noch kein Lampenleistungssteuerbefehl an das Vorschaltgerät gesendet.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ [95](#)] (KL6811) oder FB_KL6821Communication() [▶ [104](#)] (KL6821).

5.1.95 FB_DALIV2QueryPowerOnLevel



Die Variable POWER ON LEVEL [▶ [370](#)] (Einschaltlampenleistung) wird aus dem Vorschaltgerät ausgelesen. Direkt nach dem Anlegen der Stromversorgung schaltet das Vorschaltgerät die Lampe auf diesen Lampenleistungswert.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ [387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ [387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nPowerOnLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

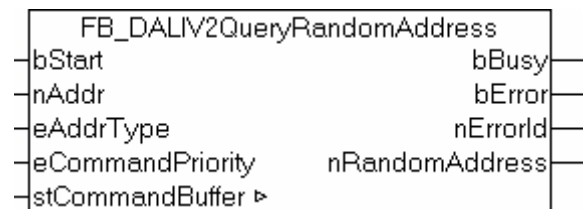
nPowerOnLevel: Einschaltlampenleistung (0 - 254).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.96 FB_DALIV2QueryRandomAddress



Die Variable [RANDOM ADDRESS](#) [▶ 372] (Zufallsadresse) wird aus dem Vorschaltgerät ausgelesen.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nRandomAddress : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

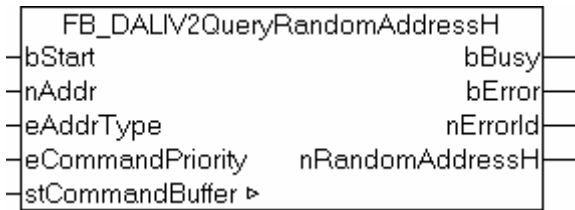
nRandomAddress: Zufallsadresse / Langadresse des Vorschaltgerätes.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.97 FB_DALIV2QueryRandomAddressH



Das höherwertige Byte der Variable [RANDOM ADDRESS \[▶ 372\]](#) (Zufallsadresse) wird aus dem Vorschaltgerät ausgelesen.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[▶ 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nRandomAddressH : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

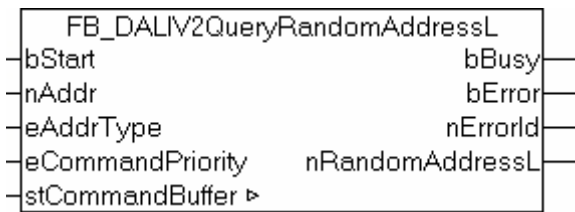
nRandomAddressH: Das höherwertige Byte der Zufallsadresse / Langadresse.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.98 FB_DALIV2QueryRandomAddressL



Das niederwertige Byte der Variable RANDOM ADDRESS [▶ 372] (Zufallsadresse) wird aus dem Vorschaltgerät ausgelesen.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nRandomAddressL : BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nRandomAddressL: Das niederwertige Byte der Zufallsadresse / Langadresse.

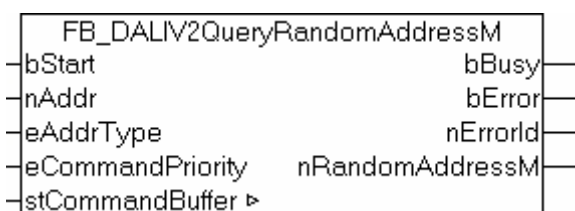
VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.99 FB_DALIV2QueryRandomAddressM



Das mittlere Byte der Variable RANDOM ADDRESS [▶ 372] (Zufallsadresse) wird aus dem Vorschaltgerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nRandomAddressM : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

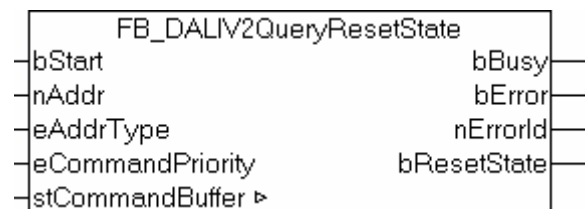
nRandomAddressM: Das mittlere Byte der Zufallsadresse / Langadresse.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.100 FB_DALIV2QueryResetState



Der Baustein gibt Auskunft, ob sich ein bestimmtes Vorschaltgerät im Rücksetzzustand befindet.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bResetState : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

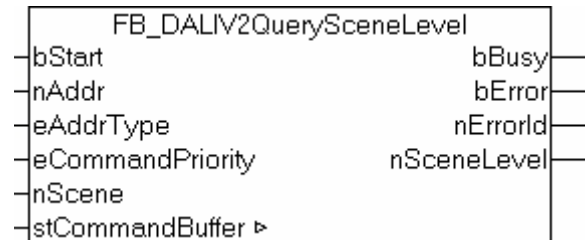
bResetState: Ist der Ausgang aktiv, so befindet sich das entsprechende Vorschaltgerät im Rücksetzzustand.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.101 FB_DALIV2QuerySceneLevel



Der Lampenleistungswert der entsprechenden Szene wird aus dem Vorschaltgerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nScene      : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nScene: Szene, von dem der Lampenleistungswert ausgelesen werden soll (0 - 15).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nSceneLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

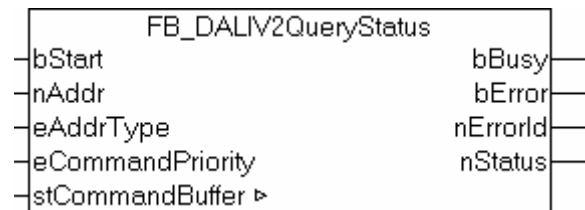
nSceneLevel: Lampenleistungswert der Szene.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.102 FB_DALIV2QueryStatus



Die Variable [STATUS INFORMATION](#) [▶ 372] (Statusinformationen) wird aus dem Vorschaltgerät ausgelesen. Die Statusinformation enthält die acht wichtigsten Zustandsmeldungen eines Vorschaltgerätes. Hierbei ist die Bedeutung der einzelnen Bit wie folgt definiert:

Bit	Beschreibung
0	Status des Vorschaltgerätes. 0: OK.
1	Lampenausfall. 0: OK.
2	Lampenleistung ein. 0: AUS.
3	Grenzwertfehler. 0: letzter angeforderter Lampenleistungswert liegt zwischen MIN LEVEL und MAX LEVEL oder AUS.
4	Stufung fertig: 0: Stufung beendet. 1: Stufung aktiv.
5	Rücksetzzustand. 0: Nein.
6	Kurzadresse fehlt. 0: Nein.
7	Störung Stromversorgung. 0: Nein, Rücksetzen oder Lampenleistungssteuerbefehl wurde nach dem letzten Einschalten empfangen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nStatus    : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

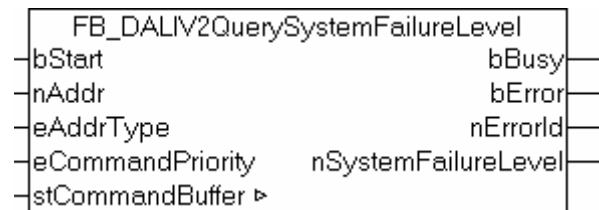
nStatus: Statusinformationen (siehe Tabelle oben).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.103 FB_DALIV2QuerySystemFailureLevel



Die Variable SYSTEM FAILURE LEVEL [▶ 370] (Lampenleistungswert bei Systemfehler) wird aus dem Vorschaltgerät ausgelesen. Wird am DALI-Bus ein Fehler (z.B. das Fehlen der Versorgungsspannung) erkannt, so schaltet das Vorschaltgerät die Lampe auf diesen Lampenleistungswert.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse, Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nSystemFailureLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nSystemFailureLevel: Lampenleistungswert bei einem Systemfehler.

VAR_IN_OUT

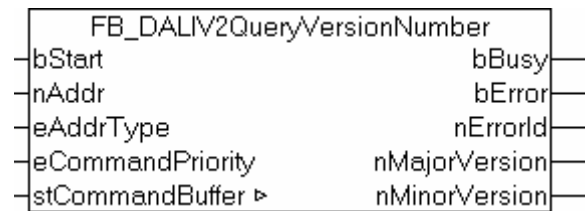
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Sehen Sie dazu auch

- ▣ [E_DALIV2AddrType](#) [▶ 387]
- ▣ [E_DALIV2CommandPriority](#) [▶ 387]

5.1.104 FB_DALIV2QueryVersionNumber



Die Variable [VERSION NUMBER](#) [▶ 373] (Versionsnummer) wird aus dem Vorschaltgerät ausgelesen. Die Versionsnummer entspricht der Versionsnummer der IEC-Norm, nach der die Software und die Hardware des Vorschaltgerätes entwickelt und hergestellt wurde. Die Versionsnummer kann nur gelesen werden und wird vom Hersteller vorgegeben. Die Hauptversion (*nMajorVersion*) und die Unterversion (*nMinorVersion*) können einen Wertebereich von je 0 bis 15 (4-Bit) annehmen.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse, Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nMajorVersion : BYTE;
nMinorVersion : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nMajorVersion: Hauptversionsnummer.

nMinorVersion: Unterversionsnummer.

VAR_IN_OUT

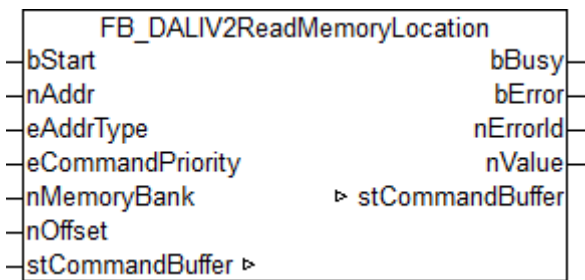
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Sehen Sie dazu auch

- [E_DALIV2AddrType](#) [▶ 387]
- [E_DALIV2CommandPriority](#) [▶ 387]

5.1.105 FB_DALIV2ReadMemoryLocation



Es wird ein Byte aus dem Speicher des Vorschaltgerätes ausgelesen. Die genaue Speicherbank wird durch den Parameter *nMemoryBank* angegeben und die Adresse innerhalb der Speicherbank durch den Parameter *nOffset*.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nMemoryBank : BYTE;
nOffset : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nMemoryBank: Die Speicherbank, auf die zugegriffen werden soll.

nOffset: Die Adresse innerhalb der Speicherbank, auf die zugegriffen werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nValue     : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

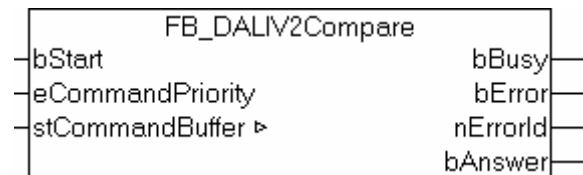
nValue: gelesenes Byte aus der Speicherbank des Vorschaltgerätes.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

5.1.106 FB_DALIV2Compare



Das Vorschaltgerät vergleicht seine Zufallsadresse ([RANDOM ADDRESS](#) [► 372]) mit der Suchadresse ([SEARCH ADDRESS](#) [► 372]). Ist die Zufallsadresse kleiner oder gleich der Suchadresse und ist das Vorschaltgerät nicht ausgeschlossen, dann wird der Ausgang *bAnswer* auf TRUE gesetzt.

VAR_INPUT

```
bStart      : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bAnswer    : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

bAnswer: Die Zufallsadresse ist kleiner oder gleich der Suchadresse.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▸ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▸ 104\]](#) (KL6821).

5.1.107 FB_DALIV2Initialise



Mit diesem Befehl wird die Adressierung der Vorschaltgeräte gestartet. Durch den Baustein [FB_DALIV2Terminate\(\) \[▸ 168\]](#) muss die Adressierung wieder beendet werden. Die maximale Dauer ist auf 15 min begrenzt. Jedes Vorschaltgerät beendet selbstständig, nach Ablauf der 15 min, die Adressierung. Die Reaktion der Vorschaltgeräte, die diesen Befehl empfangen, ist von dem Parameter *nParameter* abhängig:

Wert (binär)	Beschreibung
0000 0000	Alle Vorschaltgeräte reagieren.
0AAA AAA1	Vorschaltgeräte mit der Adresse AAA AAA reagieren.
1111 1111	Vorschaltgeräte ohne Kurzadresse reagieren.

VAR_INPUT

```
bStart : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nParameter : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▸ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nParameter: Gibt vor, welche Vorschaltgeräte auf diesen Befehl reagieren sollen (siehe Tabelle oben).

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

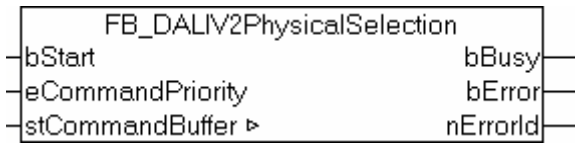
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▸ 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▸ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▸ 104\]](#) (KL6821).

5.1.108 FB_DALIV2PhysicalSelection



Alle Vorschaltgeräte, die diesen Befehl empfangen, gehen in die Betriebsart *physikalische Auswahl*. In dieser Betriebsart wird das Vergleichen der Zufallsadresse ([RANDOM ADDRESS \[▶ 372\]](#)) mit der Suchadresse ([SEARCH ADDRESS \[▶ 372\]](#)) gesperrt.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

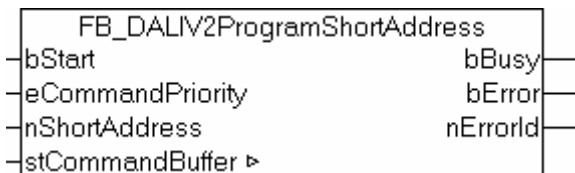
VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[▶ 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

5.1.109 FB_DALIV2ProgramShortAddress



Alle ausgewählten Vorschaltgeräte speichern den Wert *nShortAddress* ([SHORT ADDRESS \[▶ 372\]](#)) als Kurzadresse ab.

Ausgewählt bedeutet:

- Die Zufallsadresse ([RANDOM ADDRESS \[▶ 372\]](#)) des Vorschaltgerätes ist gleich der Suchadresse ([SEARCH ADDRESS \[▶ 372\]](#)).
- Die physikalische Auswahl wurde vom Vorschaltgerät festgestellt, da die Lampe vom Vorschaltgerät getrennt wurde (nach dem Empfang des Befehls [FB_DALIV2PhysicalSelection\(\) \[▶ 160\]](#)).

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nShortAddress   : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nShortAddress: Kurzadresse, die den ausgewählten Vorschaltgeräten zugewiesen werden soll (0 - 63).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

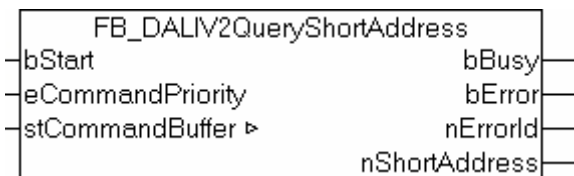
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.110 FB_DALIV2QueryShortAddress



Ist das Vorschaltgerät ausgewählt, so sendet es seine Kurzadresse (SHORT ADDRESS [▶ 372]).

Ausgewählt bedeutet:

- Die Zufallsadresse (RANDOM ADDRESS [▶ 372]) des Vorschaltgerätes ist gleich der Suchadresse (SEARCH ADDRESS [▶ 372]).
- Die physikalische Auswahl wurde vom Vorschaltgerät festgestellt, da die Lampe vom Vorschaltgerät getrennt wurde (nach dem Empfang des Befehls FB_DALIV2PhysicalSelection() [▶ 160]).

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nShortAddress : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nShortAddress: Kurzadresse vom Vorschaltgerät (0 - 63).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.111 FB_DALIV2Randomise



Die Vorschaltgeräte erzeugen eine neue Zufallsadresse (RANDOM ADDRESS [▶ 372]).

VAR_INPUT

```
bStart      : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.112 FB_DALIV2SearchAddr



Dieser Baustein setzt die Suchadresse ([SEARCH ADDRESS \[► 372\]](#)).

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nSearchAddr     : UDINT;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: [Priorität \[► 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nSearchAddr: Suchadresse.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

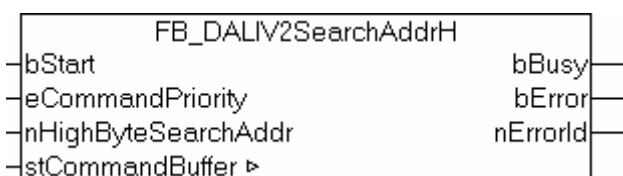
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.113 FB_DALIV2SearchAddrH



Dieser Baustein setzt die oberen 8 Bit der 24-Bit Suchadresse ([SEARCH ADDRESS \[► 372\]](#)).

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nHighByteSearchAddr : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nHighByteSearchAddr: oberen 8 Bit der 24-Bit Suchadresse.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

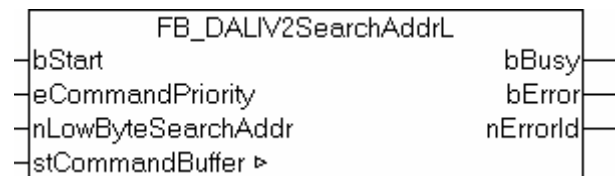
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.114 FB_DALIV2SearchAddrL



Dieser Baustein setzt die unteren 8 Bit der 24-Bit Suchadresse (SEARCH ADDRESS [▶ 372]).

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nLowByteSearchAddr : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nLowByteSearchAddr: unteren 8 Bit der 24-Bit Suchadresse.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

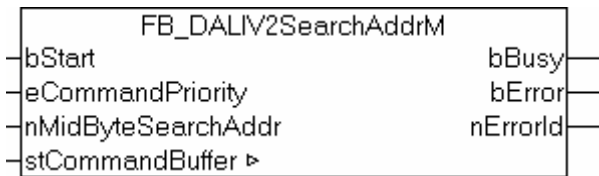
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

5.1.115 FB_DALIV2SearchAddrM



Dieser Baustein setzt die mittleren 8 Bit der 24-Bit Suchadresse ([SEARCH ADDRESS \[► 372\]](#)).

VAR_INPUT

```
bStart : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nMidByteSearchAddr : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nMidByteSearchAddr: mittleren 8 Bit der 24-Bit Suchadresse.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

Sehen Sie dazu auch

- ▣ [E_DALIV2CommandPriority \[► 387\]](#)

5.1.116 FB_DALIV2SetDTR



Der Befehl steht nur als Sammelruf (Broadcast) zur Verfügung. Es wird das DTR aller Vorschaltgeräte beschrieben.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDTR            : BYTE;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDTR: Wert, der in das DTR geschrieben werden soll.

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

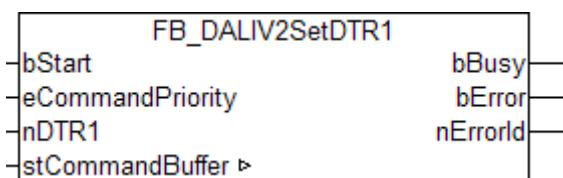
VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [[▶ 95](#)] (KL6811) oder `FB_KL6821Communication()` [[▶ 104](#)] (KL6821).

5.1.117 FB_DALIV2SetDTR1



Der Befehl steht nur als Sammelruf (Broadcast) zur Verfügung. Es wird das DTR1 aller Vorschaltgeräte beschrieben.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart      : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDTR1      : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDTR1: Wert, der in das DTR1 geschrieben werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

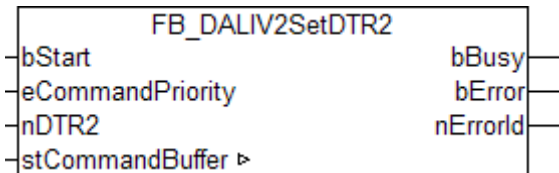
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.118 FB_DALIV2SetDTR2



Der Befehl steht nur als Sammelruf (Broadcast) zur Verfügung. Es wird das DTR2 aller Vorschaltgeräte beschrieben.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart      : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDTR2      : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDTR2: Wert, der in das DTR2 geschrieben werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

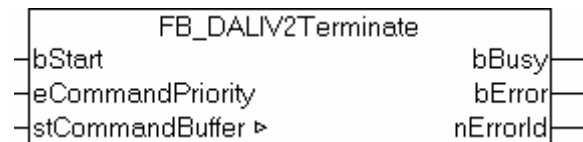
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.119 FB_DALIV2Terminate



Bei allen Vorschaltgeräten wird die Adressierung beendet.

VAR_INPUT

```
bStart      : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.120 FB_DALIV2VerifyShortAddress



Ist bei dem Vorschaltgerät die Kurzadresse gleich dem Parameter *nShortAddress*, so wird der Ausgang *bAnswer* auf TRUE gesetzt.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nShortAddress   : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nShortAddress: Kurzadresse [▶ 387] mit der die eigene Kurzadresse verglichen werden soll.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
bAnswer        : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

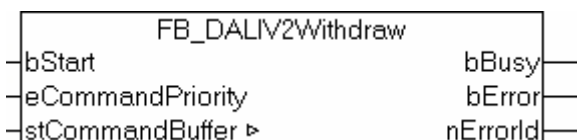
bAnswer: Parameter *nShortAddress* ist gleich der eigenen Kurzadresse.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.121 FB_DALIV2Withdraw



Vorschaltgeräte, bei denen die Zufallsadresse (RANDOM ADDRESS [▶ 372]) gleich der Suchadresse (SEARCH ADDRESS [▶ 372]) ist, dürfen nicht länger auf den Befehl FB_DALIV2Compare() [▶ 158] reagieren.

VAR_INPUT

```
bStart      : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

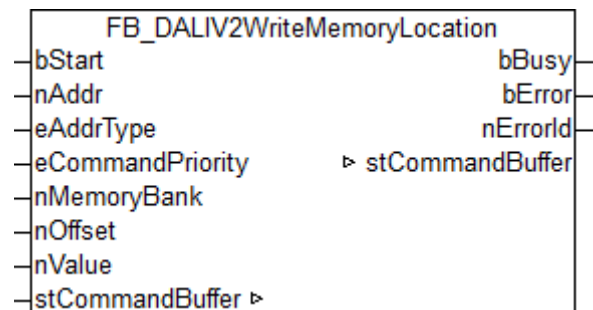
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.122 FB_DALIV2WriteMemoryLocation



Der Wert *nValue* wird in die Speicherbank des Vorschaltgerätes geschrieben. Die genaue Speicherbank wird durch *nMemoryBank* angegeben und die Adresse innerhalb der Speicherbank durch *nOffset*.



Dieser Befehl kann nur von DALI-Geräten, die der Norm IEC 62386 entsprechen, ausgeführt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nMemoryBank : BYTE;
nOffset     : BYTE;
nValue      : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nMemoryBank: Die Speicherbank, auf die zugegriffen werden soll.

nOffset: Die Adresse innerhalb der Speicherbank, auf die zugegriffen werden soll.

nValue: Wert, der in die Speicherbank des Vorschaltgerätes geschrieben werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

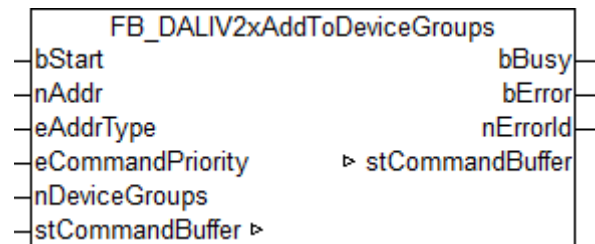
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [[▶ 95](#)] (KL6811) oder [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

Sehen Sie dazu auch

- [FB_DALIV2EnableWriteMemory](#) [[▶ 110](#)]

5.1.123 FB_DALIV2xAddToDeviceGroups



Ordnet das Steuergerät einer oder mehreren Gruppen zu.

Es stehen insgesamt 32 Gruppen zur Verfügung, denen ein Steuergerät zugeordnet werden kann. Jedes Bit der Variablen *nDeviceGroups* entspricht einer dieser Gruppen. Ist das Bit gesetzt, so wird das Steuergerät der jeweiligen Gruppe zugeordnet. Dabei entspricht Bit 0 der Gruppe 0 und Bit 31 der Gruppe 31.

Soll ein Steuergerät aus einer Gruppe entfernt werden, so kann hierzu der Baustein [FB_DALIV2xRemoveFromDeviceGroups\(\)](#) [[▶ 177](#)] genutzt werden.

VAR INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDeviceGroups : DWORD;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDeviceGroups: 32-Bit Variable, bei der jedes Bit die entsprechende Gruppe repräsentiert, der das Steuergerät zugeordnet werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

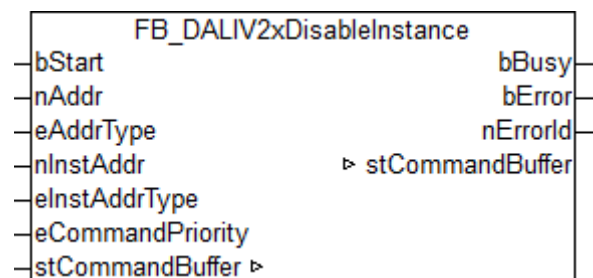
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.124 FB_DALIV2xDisableInstance



Die Instanz des Steuergerätes wird deaktiviert.

Mit dem Baustein [FB_DALIV2xEnableInstance\(\)](#) [[▶ 174](#)] kann die Instanz aktiviert werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

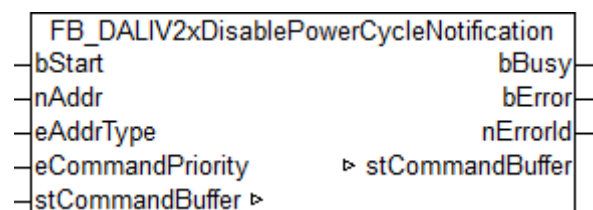
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.125 FB_DALIV2xDisablePowerCycleNotification



Dieser Baustein sperrt das *Power Cycle Notification* Ereignis.

Mit dem Baustein [FB_DALIV2xEnablePowerCycleNotification\(\)](#) [▶ 175] kann das Ereignis freigeschaltet werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.126 FB_DALIV2xEnableInstance



Die Instanz des Steuergerätes wird aktiviert.

Mit dem Baustein FB_DALIV2xDisableInstance() [▶ 172] kann die Instanz deaktiviert werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

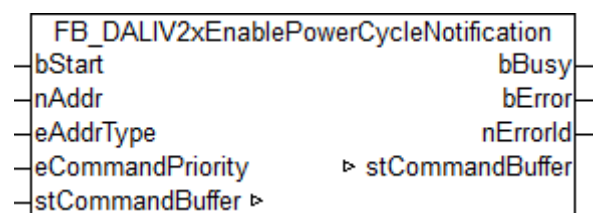
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.127 FB_DALIV2xEnablePowerCycleNotification



Dieser Baustein schaltet das *Power Cycle Notification* Ereignis frei.

Mit dem Baustein FB_DALIV2xDisablePowerCycleNotification() [▶ 173] kann das Ereignis gesperrt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

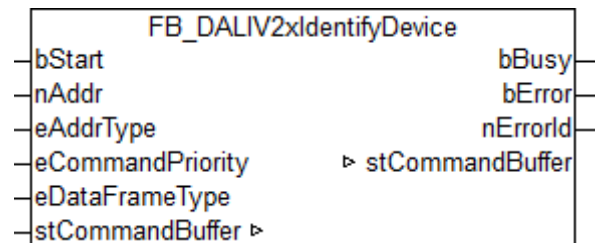
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.128 FB_DALIV2xIdentifyDevice



Startet die Identifizierungsroutine für das Steuergerät.

Die Dauer liegt bei ca. 10 sec und endet automatisch. Der genaue Umfang der Identifizierungsroutine ist vom Hersteller des Steuergerätes anhängig.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

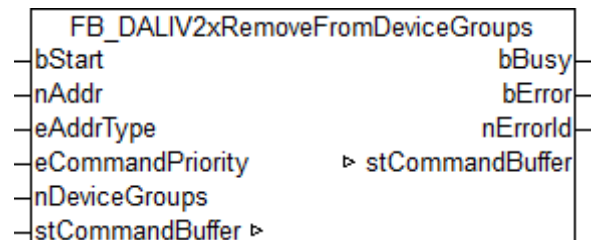
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.129 FB_DALIV2xRemoveFromDeviceGroups



Entfernt das Steuergerät aus einer oder mehreren Gruppen.

Es stehen insgesamt 32 Gruppen zur Verfügung, denen ein Steuergerät zugeordnet werden kann. Jedes Bit der Variablen *nDeviceGroups* entspricht einer dieser Gruppen. Ist das Bit gesetzt, so wird das Steuergerät aus der jeweiligen Gruppe entfernt. Dabei entspricht Bit 0 der Gruppe 0 und Bit 31 der Gruppe 31.

Soll ein Steuergerät einer Gruppe zugeordnet werden, so kann hierzu der Baustein *FB_DALIV2xAddToDeviceGroups()* [▶ 171] genutzt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDeviceGroups : DWORD;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDeviceGroups: 32-Bit Variable, bei der jedes Bit die entsprechende Gruppe repräsentiert, aus der das Steuergerät entfernt werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

VAR_IN_OUT

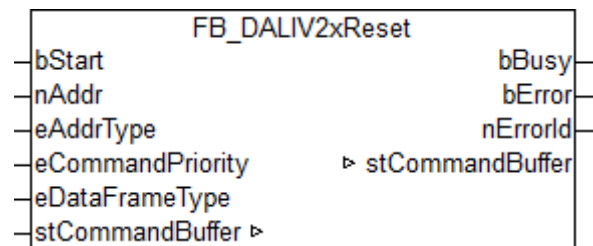
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.130 FB_DALIV2xReset



Dieser Baustein setzt alle Parameter auf die Standardwerte zurück.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [► 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.131 FB_DALIV2xSetEventFilter



Dieser Baustein setzt den Ereignis-Filter für die jeweilige Instanz des Steuergerätes.

Jedes Bit in *nEventFilter* repräsentiert ein Ereignis. Ist das Bit gesetzt, so ist auch das zugehörige Ereignis freigeschaltet. Das Ereignis ist gesperrt, wenn das Bit nicht gesetzt ist.

Die Bedeutung der einzelnen Bits ist der Dokumentation des jeweiligen Steuergerätes zu entnehmen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nEventFilter : DWORD := 0;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [[▶ 388](#)], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nEventFilter: Jedes Bit repräsentiert ein Ereignis das freigegeben oder gesperrt werden soll.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

VAR_IN_OUT

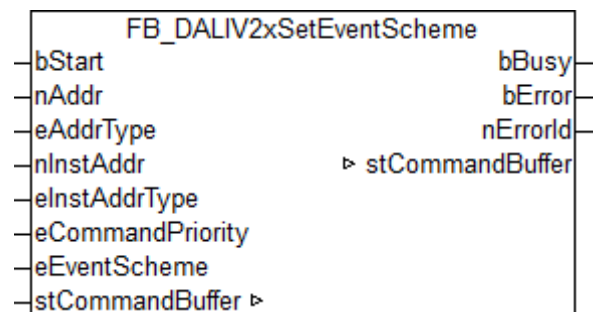
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.132 FB_DALIV2xSetEventScheme



Mit diesem Baustein kann das Adressierungs-Schema für die Ereignisse der jeweiligen Instanz des Steuergerätes festgelegt werden.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr       : BYTE := 0;
eInstAddrType   : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eEventScheme    : E_DALIV2EventScheme := eDALIV2EventSchemeInstance;
```


bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eEventScheme: Adressierungs-Schema [▶ 388] für die Ereignisse.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

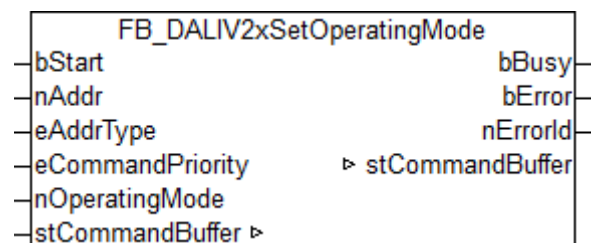
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.133 FB_DALIV2xSetOperatingMode



Setzt den *Operating Mode* für das Steuergerät.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nOperatingMode : BYTE := 0;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nOperatingMode: Neuer Operating Mode.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

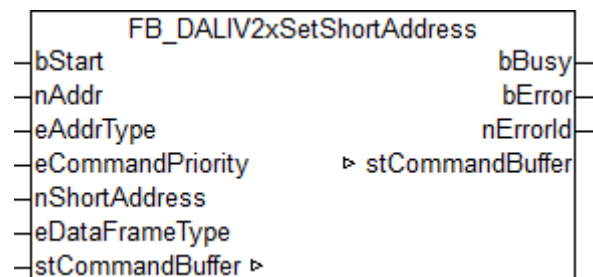
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.134 FB_DALIV2xSetShortAddress



Setzt die Kurzadresse des Steuergerätes.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nShortAddress : BYTE;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nShortAddress: Neue Kurzadresse (0...63, 255)

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

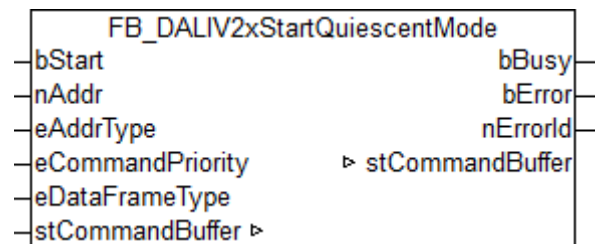
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.135 FB_DALIV2xStartQuiescentMode



Der *Quiescent Mode* des Steuergerätes wird gestartet.

Im *Quiescent Mode* sendet das Steuergerät keine Befehle oder Ereignisse. Der *Quiescent Mode* endet nach ca. 15 min oder durch *FB_DALIV2xStopQuiescentMode()* [▶ 184].

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: [Ausgabeformat](#) [[▶ 388](#)] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

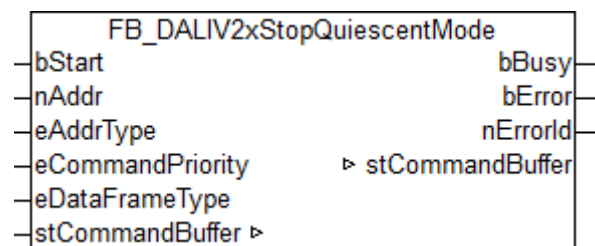
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.136 FB_DALIV2xStopQuiescentMode



Der *Quiescent Mode* des Steuergerätes wird beendet.

Im *Quiescent Mode* sendet das Steuergerät keine Befehle oder Ereignisse. Der *Quiescent Mode* wird durch `FB_DALIV2xStartQuiescentMode()` [[▶ 183](#)] gestartet.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: [Ausgabeformat](#) [[▶ 388](#)] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

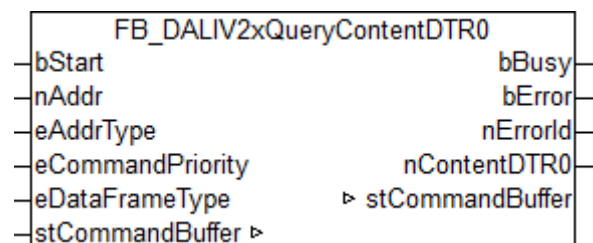
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.137 FB_DALIV2xQueryContentDTR0



Der Inhalt des DTR0 (Data Transfer Register) wird aus dem Steuergerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: [Ausgabeformat](#) [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nContentDTR0 : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nContentDTR0: Inhalt des DTR0 (Data Transfer Register)

VAR_IN_OUT

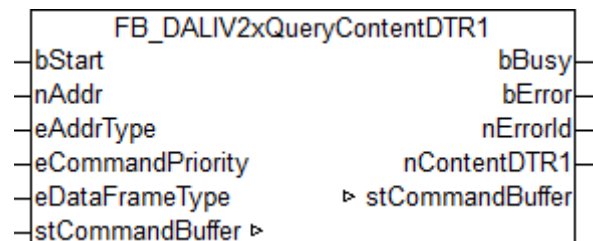
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.138 FB_DALIV2xQueryContentDTR1



Der Inhalt des DTR1 (Data Transfer Register) wird aus dem Steuergerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: [Ausgabeformat](#) [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nContentDTR1 : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nContentDTR1: Inhalt des DTR1 (Data Transfer Register)

VAR_IN_OUT

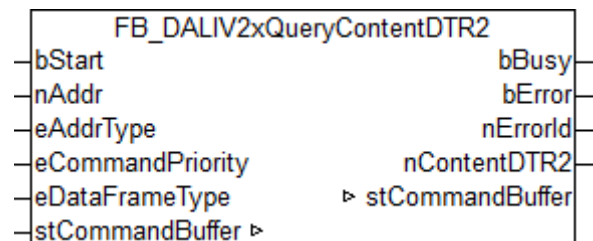
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.139 FB_DALIV2xQueryContentDTR2



Der Inhalt des DTR2 (Data Transfer Register) wird aus dem Steuergerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: [Ausgabeformat](#) [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nContentDTR2 : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nContentDTR2: Inhalt des DTR2 (Data Transfer Register)

VAR_IN_OUT

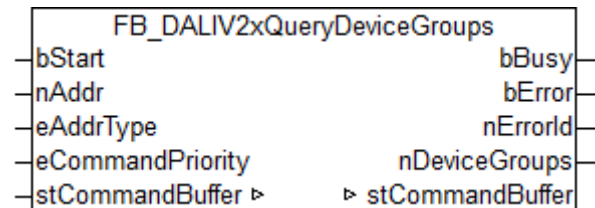
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.140 FB_DALIV2xQueryDeviceGroups



Fragt die Gruppenzuordnungen des Steuergerätes ab.

Es stehen insgesamt 32 Gruppen zur Verfügung, denen ein Steuergerät zugeordnet sein kann. Jedes Bit der Variablen *nDeviceGroups* entspricht einer dieser Gruppen. Ist das Bit gesetzt, so ist das Steuergerät der jeweiligen Gruppe zugeordnet. Dabei entspricht Bit 0 der Gruppe 0 und Bit 31 der Gruppe 31.

Soll ein Steuergerät einer Gruppe zugeordnet werden, so kann hierzu der Baustein [FB_DALIV2xAddToDeviceGroups\(\)](#) [▶ 171] genutzt werden.

Soll ein Steuergerät aus einer Gruppe entfernt werden, so kann hierzu der Baustein [FB_DALIV2xRemoveFromDeviceGroups\(\)](#) [▶ 177] genutzt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nDeviceGroups : DWORD;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nDeviceGroups: 32-Bit Variable, bei der jedes Bit die entsprechende Gruppe repräsentiert, der das Steuergerät zugeordnet wurde.

VAR_IN_OUT

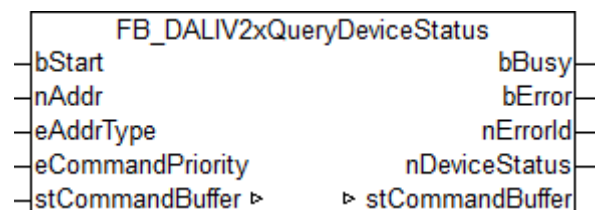
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.141 FB_DALIV2xQueryDeviceStatus



Der Baustein liest den *Device Status* des Steuergerätes aus.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nDeviceStatus : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nDeviceStatus: Der *Device Status* des Steuergerätes.

VAR_IN_OUT

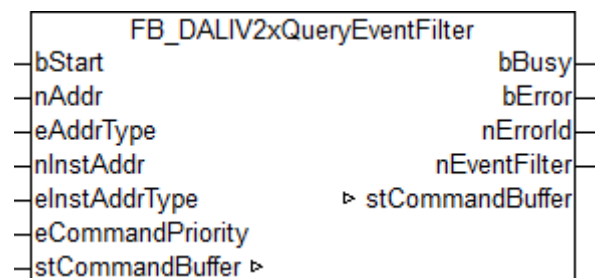
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.142 FB_DALIV2xQueryEventFilter



Dieser Baustein fragt den Ereignis-Filter für die jeweilige Instanz des Steuergerätes ab.

Jedes Bit in *nEventFilter* repräsentiert ein Ereignis. Ist das Bit gesetzt, so ist auch das zugehörige Ereignis freigeschaltet. Das Ereignis ist gesperrt, wenn das Bit nicht gesetzt ist.

Die Bedeutung der einzelnen Bits ist der Dokumentation des jeweiligen Steuergerätes zu entnehmen.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr  : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nEventFilter : DWORD;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nEventFilter: Jedes Bit repräsentiert ein Ereignis das freigegeben oder gesperrt wurde.

VAR_IN_OUT

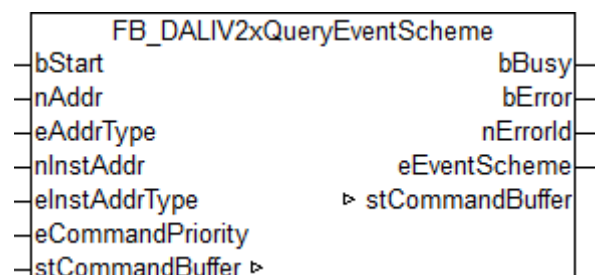
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.143 FB_DALIV2xQueryEventScheme



Mit diesem Baustein kann das Adressierungs-Schema für die Ereignisse der jeweiligen Instanz des Steuergerätes abgefragt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
eEventScheme : E_DALIV2EventScheme;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

eEventScheme: Adressierungs-Schema [▶ 388] für die Ereignisse.

VAR_IN_OUT

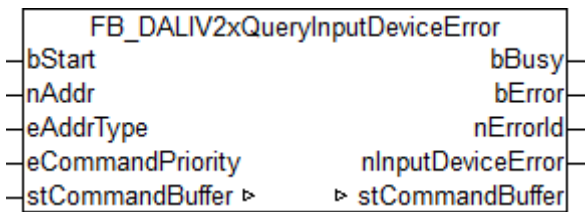
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.144 FB_DALIV2xQueryInputDeviceError



Der Baustein liest den *Input Device Error* des Steuergerätes aus.

Die Bedeutung des *Input Device Error* ist Abhängig vom Hersteller des Steuergerätes.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nInputDeviceError : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nInputDeviceError: Der *Input Device Error* des Steuergerätes.

VAR_IN_OUT

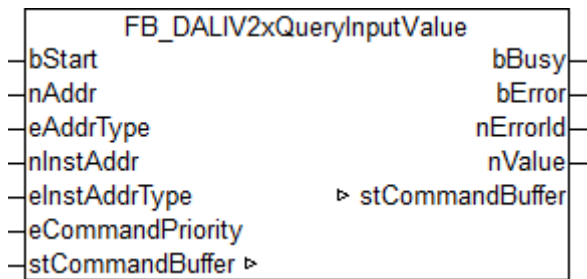
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.145 FB_DALIV2xQueryInputValue



Fragt das erste Byte des Eingangswertes der Instanz des Steuergerätes ab.

Der aktuelle Eingangswert wird zwischen gespeichert und das höchstwertige Byte (MSB) zurückgegeben.

Alle weiteren Bytes können mit dem Baustein [FB_DALIV2xQueryInputValueLatch\(\)](#) [► 195] ausgelesen werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [► 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nValue      : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

nValue: Das höchstwertige Byte (MSB) des Eingangswertes.

VAR_IN_OUT

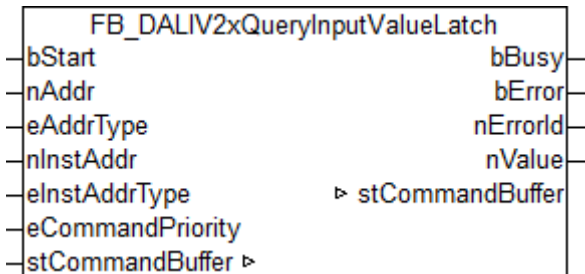
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.146 FB_DALIV2xQueryInputValueLatch



Fragt jeweils das folgende Byte des Eingangswertes der Instanz des Steuergerätes ab.

Das erste Byte wird mit dem Baustein FB_DALIV2xQueryInputValue() [► 194]ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [► 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nValue     : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

nValue: Das folgende Byte des Eingangswertes.

VAR_IN_OUT

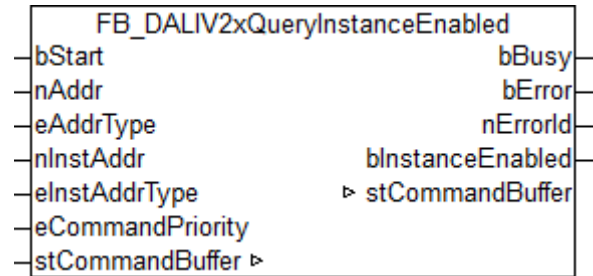
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [[104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.147 FB_DALIV2xQueryInstanceEnabled



Fragt ab, ob die Instanz des Steuergerätes aktiviert ist.

Mit dem Baustein `FB_DALIV2xDisableInstance()` [[172](#)] kann die Instanz deaktiviert werden.

Mit dem Baustein `FB_DALIV2xEnableInstance()` [[174](#)] kann die Instanz aktiviert werden.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [[387](#)], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [[388](#)], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [[387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bInstanceEnabled : BOOL;
    
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[391](#)].

blnInstanceEnabled: Ist TRUE, wenn die Instanz freigeschaltet wurde.

VAR_IN_OUT

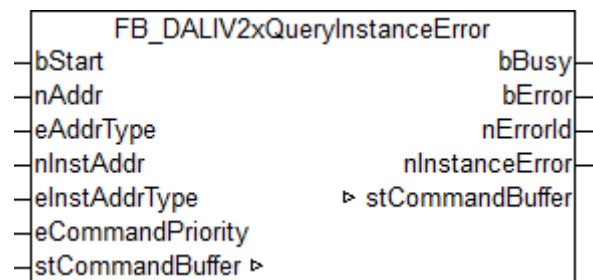
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.148 FB_DALIV2xQueryInstanceError



Der Baustein liest den *Instance Error* des Steuergerätes aus.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nInstanceError : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

nInstanceError: Der *Instance Error* des Steuergerätes.

VAR_IN_OUT

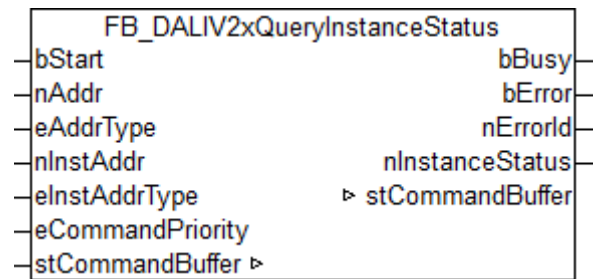
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.149 FB_DALIV2xQueryInstanceStatus



Der Baustein liest den *Instance Status* des Steuergerätes aus.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [► 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nInstanceStatus : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nInstanceStatus: Der *Instance Status* des Steuergerätes.

VAR_IN_OUT

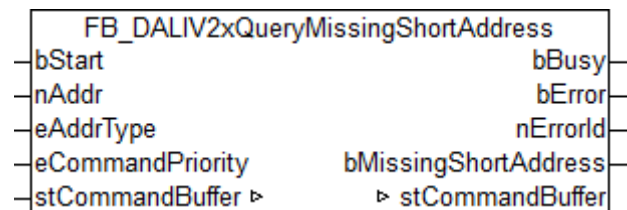
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.150 FB_DALIV2xQueryMissingShortAddress



Fragt ab, ob das Steuergerät keine gültige Kurzadresse besitzt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
bMissingShortAddress : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

bMissingShortAddress: Ist TRUE, falls das Steuergerät keine gültige Kurzadresse besitzt.

VAR_IN_OUT

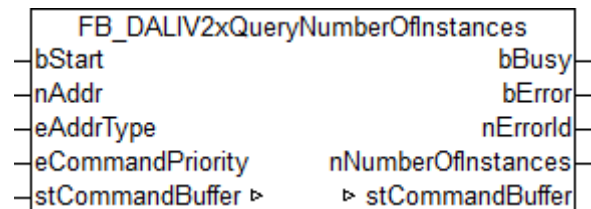
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.151 FB_DALIV2xQueryNumberOfInstances



Fragt die Anzahl der Instanzen ab, die das Steuergerät besitzt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nNumberOfInstances : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

nNumberOfInstances: Anzahl der Instanzen des Steuergerätes.

VAR_IN_OUT

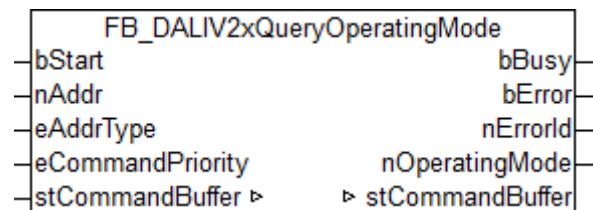
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.152 FB_DALIV2xQueryOperatingMode



Fragt den *Operating Mode* des Steuergerätes ab.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nOperatingMode : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nOperatingMode: Operating Mode.

VAR_IN_OUT

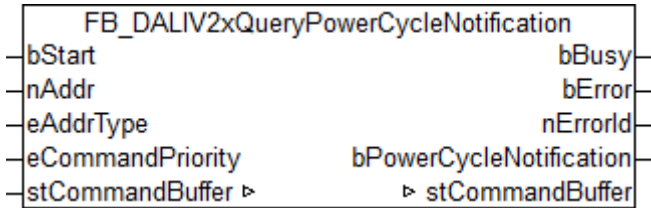
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.153 FB_DALIV2xQueryPowerCycleNotification



Fragt ab, ob das *Power Cycle Notification* Ereignis freigeschaltet ist.

Mit dem Baustein [FB_DALIV2xDisablePowerCycleNotification\(\)](#) [► 173] kann das Ereignis gesperrt werden.

Mit dem Baustein [FB_DALIV2xEnablePowerCycleNotification\(\)](#) [► 175] kann das Ereignis freigeschaltet werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bPowerCycleNotification : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

bPowerCycleNotification: Ist TRUE, wenn das *Power Cycle Notification* Ereignis freigeschaltet ist.

VAR_IN_OUT

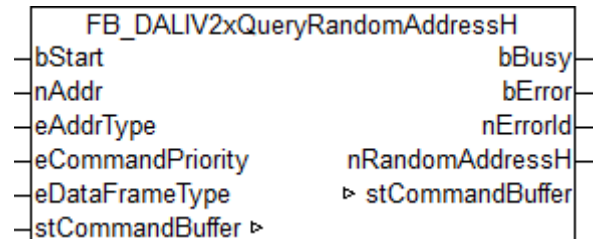
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.154 FB_DALIV2xQueryRandomAddressH



Das höherwertige Byte der Zufallsadresse wird aus dem Steuergerät ausgelesen.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nRandomAddressH : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nRandomAddressH: Das höherwertige Byte der Zufallsadresse / Langadresse.

VAR_IN_OUT

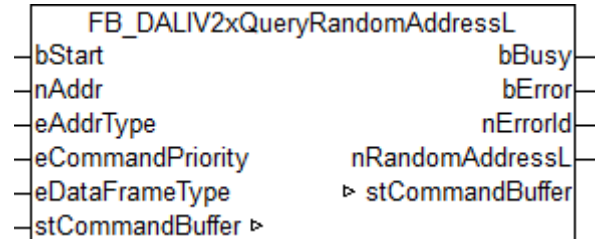
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.155 FB_DALIV2xQueryRandomAddressL



Das niederwertige Byte der Zufallsadresse wird aus dem Steuergerät ausgelesen.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;

```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nRandomAddressL : BYTE;

```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nRandomAddressL: Das niederwertige Byte der Zufallsadresse / Langadresse.

VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;

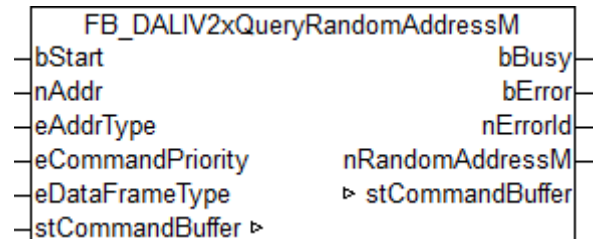
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.156 FB_DALIV2xQueryRandomAddressM



Das mittlere Byte der Zufallsadresse wird aus dem Steuergerät ausgelesen.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nRandomAddressM : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nRandomAddressM: Das mittlere Byte der Zufallsadresse / Langadresse.

VAR_IN_OUT

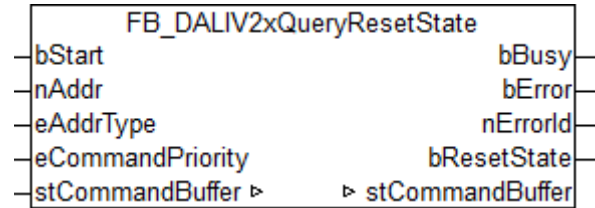
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.157 FB_DALIV2xQueryResetState



Fragt ab, ob die Parameter des Steuergerätes ihre Standardwerte besitzen.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bResetState : BOOL;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bResetState: Ist TRUE, wenn die Parameter des Steuergerätes die Standardwerte besitzen.

VAR_IN_OUT

```

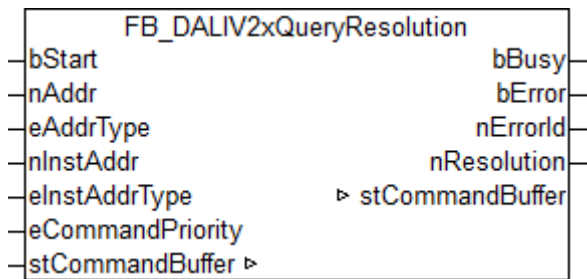
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.158 FB_DALIV2xQueryResolution



Fragt die Auflösung der Eingangswerte des Steuergerätes ab.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nResolution : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nResolution: Auflösung der Eingangswerte.

VAR_IN_OUT

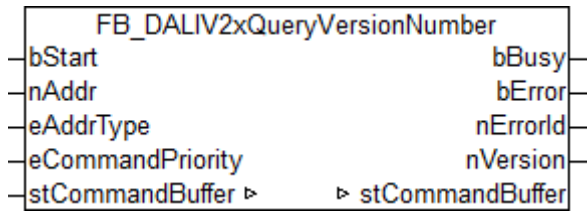
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.159 FB_DALIV2xQueryVersionNumber



Fragt die Versionsnummer des Steuergerätes ab.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nVersion    : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nVersion: Die Versionsnummer des Steuergerätes.

VAR_IN_OUT

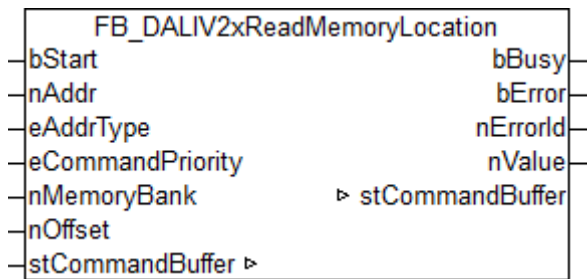
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.160 FB_DALIV2xReadMemoryLocation



Es wird ein Byte aus dem Speicher des Steuergerätes ausgelesen. Die genaue Speicherbank wird durch das den Parameter *nMemoryBank* angegeben und die Adresse innerhalb der Speicherbank durch den Parameter *nOffset*.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nMemoryBank : BYTE := 0;
nOffset     : BYTE := 0;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nMemoryBank: Die Speicherbank, auf die zugegriffen werden soll.

nOffset: Die Adresse innerhalb der Speicherbank, auf die zugegriffen werden soll.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nValue     : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nValue: gelesenes Byte aus der Speicherbank des Steuergerätes.

VAR_IN_OUT

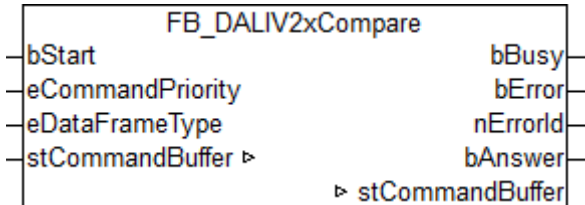
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.161 FB_DALIV2xCompare



Das Vorschaltgerät vergleicht seine Zufallsadresse mit der Suchadresse. Ist die Zufallsadresse kleiner oder gleich der Suchadresse und ist das Steuergerät nicht ausgeschlossen, dann wird der Ausgang *bAnswer* auf TRUE gesetzt.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
bAnswer        : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bAnswer: Die Zufallsadresse ist kleiner oder gleich der Suchadresse.

VAR_IN_OUT

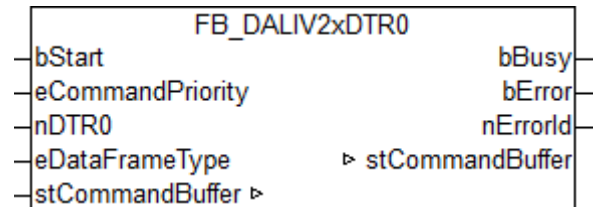
```
stCommandBuffer : ST_DALIV2CommandBuffer
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.162 FB_DALIV2xDTR0



Der Befehl steht nur als Sammelruf (Broadcast) zur Verfügung. Es wird das DTR0 aller Steuergeräte beschrieben.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDTR0          : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDTR0: Wert, der in das DTR0 geschrieben werden soll.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```

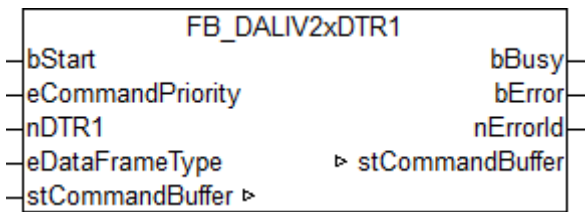
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.163 FB_DALIV2xDTR1



Der Befehl steht nur als Sammelruf (Broadcast) zur Verfügung. Es wird das DTR1 aller Steuergeräte beschrieben.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDTR1          : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDTR1: Wert, der in das DTR1 geschrieben werden soll.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```

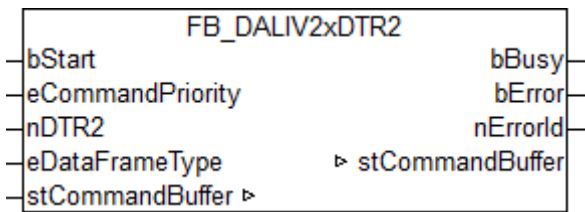
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.164 FB_DALIV2xDTR2



Der Befehl steht nur als Sammelruf (Broadcast) zur Verfügung. Es wird das DTR2 aller Steuergeräte beschrieben.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDTR2           : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDTR2: Wert, der in das DTR2 geschrieben werden soll.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```

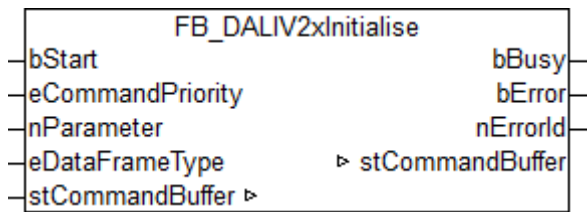
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.165 FB_DALIV2xInitialise



Mit diesem Befehl wird die Adressierung der Steuergeräte gestartet. Durch den Baustein [FB_DALIV2xTerminate\(\)](#) [[▶ 221](#)] muss die Adressierung wieder beendet werden. Die maximale Dauer ist auf 15 min begrenzt. Jedes Steuergerät beendet selbstständig, nach Ablauf der 15 min, die Adressierung. Die Reaktion der Steuergeräte, die diesen Befehl empfangen, ist von dem Parameter *nParameter* abhängig:

eDataFrameType = eDataFrameTypeOsram:

Wert (binär)	Beschreibung
0000 0000	Alle Steuergeräte reagieren.
0AAA AAA1	Steuergeräte mit der Adresse AAA AAA reagieren.
1111 1111	Steuergeräte ohne Kurzadresse reagieren.

eDataFrameType = eDataFrameType24Bit:

Wert (binär)	Beschreibung
0111 1111	Steuergeräte ohne Kurzadresse reagieren.
00AA AAAA	Steuergeräte mit der Adresse AAA AAA reagieren.
1111 1111	Alle Steuergeräte reagieren.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nParameter      : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nParameter: Gibt vor, welche Steuergeräte auf diesen Befehl reagieren sollen (siehe Tabelle oben).

eDataFrameType: Ausgabeformat [[▶ 388](#)] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

VAR_IN_OUT

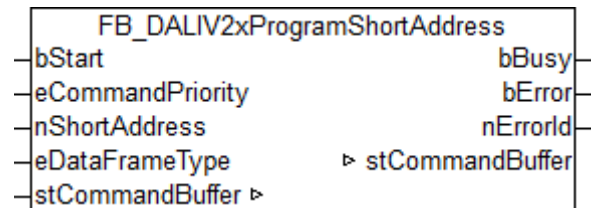
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [[104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.166 FB_DALIV2xProgramShortAddress



Alle ausgewählten Steuergeräte speichern den Wert *nShortAddress* als Kurzadresse ab.

Ausgewählt bedeutet:

- Die Zufallsadresse des Steuergerätes ist gleich der Suchadresse

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nShortAddress    : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nShortAddress: [Kurzadresse](#) [[387](#)], die den ausgewählten Steuergeräten zugewiesen werden soll (0 - 63).

eDataFrameType: [Ausgabeformat](#) [[388](#)] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[391](#)].

VAR_IN_OUT

```

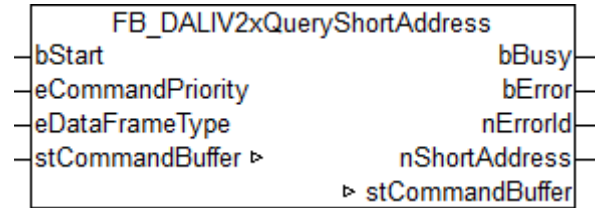
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [[104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.167 FB_DALIV2xQueryShortAddress



Ist das Steuergerät ausgewählt, so sendet es seine Kurzadresse.

Ausgewählt bedeutet:

- Die Zufallsadresse des Steuergerätes ist gleich der Suchadresse.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nShortAddress  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nShortAddress: Kurzadresse vom Steuergerät (0 - 63).

VAR_IN_OUT

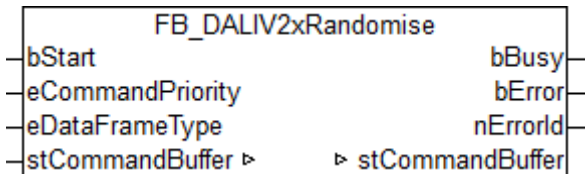
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.168 FB_DALIV2xRandomise



Die Steuergeräte erzeugen eine neue Zufallsadresse.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

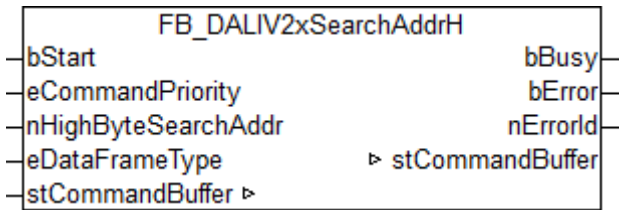
Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

Sehen Sie dazu auch

- 📖 [E_DALIV2CommandPriority](#) [▶ 387]
- 📖 [E_DALIV2DataFrameType](#) [▶ 388]

5.1.169 FB_DALIV2xSearchAddrH



Dieser Baustein setzt die oberen 8 Bit der 24-Bit Suchadresse.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nHighByteSearchAddr : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▷ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nHighByteSearchAddr: oberen 8 Bit der 24-Bit Suchadresse.

eDataFrameType: Ausgabeformat [▷ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▷ 391].

VAR_IN_OUT

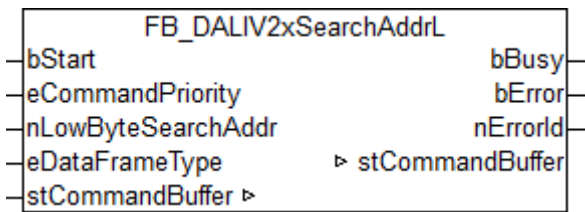
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▷ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.170 FB_DALIV2xSearchAddrL



Dieser Baustein setzt die unteren 8 Bit der 24-Bit Suchadresse.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nLowByteSearchAddr : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nLowByteSearchAddr: unteren 8 Bit der 24-Bit Suchadresse.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

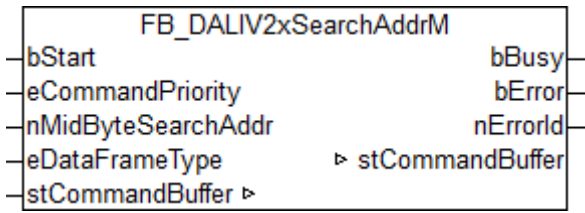
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.171 FB_DALIV2xSearchAddrM



Dieser Baustein setzt die mittleren 8 Bit der 24-Bit Suchadresse.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nMidByteSearchAddr : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nMidByteSearchAddr: mittleren 8 Bit der 24-Bit Suchadresse.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

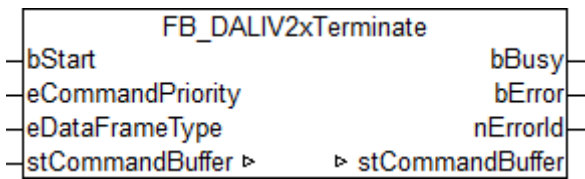
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.172 FB_DALIV2xTerminate



Bei allen Steuergeräten wird die Adressierung beendet.

VAR_INPUT

```

bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```

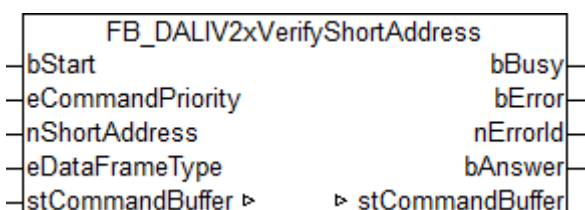
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.173 FB_DALIV2xVerifyShortAddress



Ist bei dem Steuergerät die Kurzadresse gleich dem Parameter *nShortAddress*, so wird der Ausgang *bAnswer* auf TRUE gesetzt.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nShortAddress   : BYTE;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nShortAddress: Kurzadresse mit der die eigene Kurzadresse verglichen werden soll.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
bAnswer        : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bAnswer: Parameter *nShortAddress* ist gleich der eigenen Kurzadresse.

VAR_IN_OUT

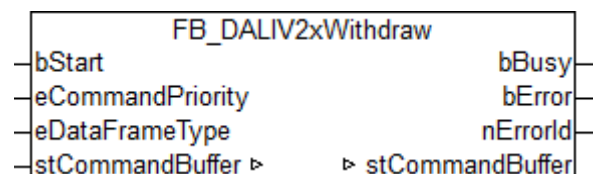
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.174 FB_DALIV2xWithdraw



Steuergeräte, bei denen die Zufallsadresse gleich der Suchadresse ist, dürfen nicht länger auf den Befehl FB_DALIV2xCompare() [▶ 210] reagieren.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDataFrameType  : E_DALIV2DataFrameType := eDALIV2DataFrameType24Bit;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDataFrameType: Ausgabeformat [▶ 388] des DALI-Befehls (*eDALIV2DataFrameType24Bit* oder *eDALIV2DataFrameTypeOsram*).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein *FB_KL6821Communication()* [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.175 FB_DALIV2xWriteMemoryLocation



Der Wert *nValue* wird in die Speicherbank des Steuergerätes geschrieben. Die genaue Speicherbank wird durch *nMemoryBank* angegeben und die Adresse innerhalb der Speicherbank durch *nOffset*.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

```
nMemoryBank : BYTE;
nOffset      : BYTE;
nValue       : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nMemoryBank: Die Speicherbank, auf die zugegriffen werden soll.

nOffset: Die Adresse innerhalb der Speicherbank, auf die zugegriffen werden soll.

nValue: Wert, der in die Speicherbank des Steuergerätes geschrieben werden soll.

VAR_OUTPUT

```
bBusy        : BOOL;
bError       : BOOL;
nErrorId     : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.176 FB_DALIV2Inhibit



Es wird für 15 Minuten verhindert, dass das Vorschaltgerät in den Notbetrieb umschaltet. Durch den Baustein FB_DALIV2RelightResetInhibit() [▶ 240] kann das Unterdrücken des Notbetriebes wieder aufgehoben werden.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

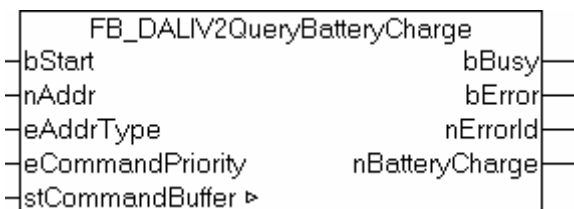
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.177 FB_DALIV2QueryBatteryCharge



Die Variable BATTERY CHARGE [▶ 376] (Akku Ladezustand) wird aus dem Vorschaltgerät ausgelesen. Wird 255 zurückgeliefert, so kann der Wert vom Vorschaltgerät nicht ermittelt werden.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nBatteryCharge : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

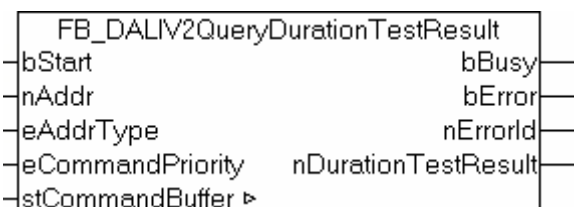
nBatteryCharge: Ladezustand des Akkus. 0: leer / 254: voll.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.178 FB_DALIV2QueryDurationTestResult



Die Variable DURATION TEST RESULT [▶ 376] (Ergebnis vom Betriebsdauertest) wird aus dem Vorschaltgerät ausgelesen. Die Einheit beträgt 2 Minuten pro Schritt. Ein Wert von 255 bedeutet den maximalen Wert von 8,5 Stunden oder länger. Der Wert ist nur dann gültig, wenn in der Variablen EMERGENCY STATUS [▶ 378] das Bit 2 gesetzt ist.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nDurationTestResult : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

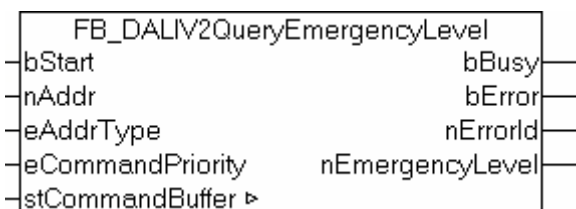
nDurationTestResult: Ergebnis vom Betriebsdauertest in 2 Minuten Schritten.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.179 FB_DALIV2QueryEmergencyLevel



Die Variable EMERGENCY LEVEL [▶ 375] (Notbeleuchtungsstärke) wird aus dem Vorschaltgerät ausgelesen. Wird 255 zurückgeliefert, so kann der Wert vom Vorschaltgerät nicht ermittelt werden.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nEmergencyLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

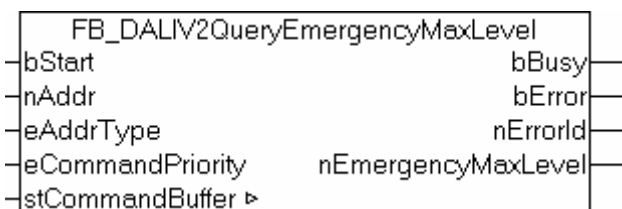
nEmergencyLevel: Notbeleuchtungsstärke des Vorschaltgerätes.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.180 FB_DALIV2QueryEmergencyMaxLevel



Die Variable EMERGENCY MAX LEVEL [▶ 375] (maximale Notbeleuchtungsstärke) wird aus dem Vorschaltgerät ausgelesen. Wird 255 zurückgeliefert, so kann der Wert vom Vorschaltgerät nicht ermittelt werden.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nEmergencyMaxLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

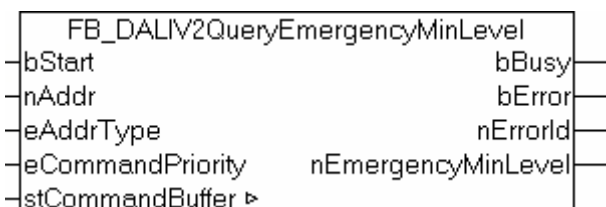
nEmergencyMaxLevel: maximale Notbeleuchtungsstärke des Vorschaltgerätes.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.181 FB_DALIV2QueryEmergencyMinLevel



Die Variable EMERGENCY MIN LEVEL [▶ 375] (minimale Notbeleuchtungsstärke) wird aus dem Vorschaltgerät ausgelesen. Wird 255 zurückgeliefert, so kann der Wert vom Vorschaltgerät nicht ermittelt werden.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nEmergencyMinLevel : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

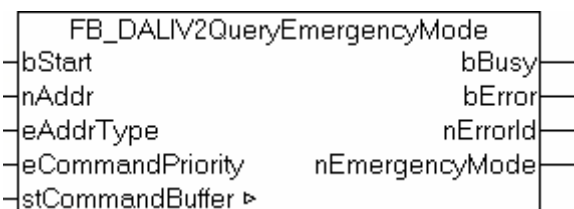
nEmergencyMinLevel: minimale Notbeleuchtungsstärke des Vorschaltgerätes.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.182 FB_DALIV2QueryEmergencyMode



Die Variable EMERGENCY MODE [▶ 377] (Modus Notbetrieb) wird aus dem Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	Rücksetz Modus. 0: Nein.
1	Bereitschaft Notbetrieb (Normalbetrieb). 0: Nein.

Bit	Beschreibung
2	Notbetrieb. 0: Nein.
3	verlängerter Notbetrieb nach Rückkehr der Netzspannung. 0: Nein.
4	Funktionstest aktiv. 0: Nein.
5	Betriebsdauerstest aktiv. 0: Nein.
6	angeschlossener Unterdrückungstaster aktiv. 0: nicht aktiv bzw. nicht vorhanden.
7	angeschlossene Netzspannung aktiv. 0: Aus.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nEmergencyMode: BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

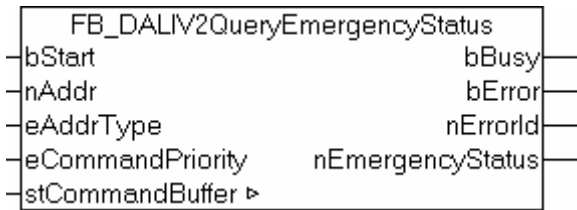
nEmergencyMode: Modus vom Notbetrieb (siehe Tabelle oben).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.183 FB_DALIV2QueryEmergencyStatus



Die Variable `EMERGENCY STATUS [▶ 377]` (Zustand Notbetrieb) wird aus dem Vorschaltgerät ausgelesen.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType() [▶ 356]` gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

Bit	Beschreibung
0	Unterdrückungsmodus. 0: Nein.
1	Funktionstest abgeschlossen und Ergebnis ist gültig. 0: Nein.
2	Betriebsdauertest abgeschlossen und Ergebnis ist gültig. 0: Nein.
3	Akkuladung betriebsbereit. 0: in Gang.
4	verzögerter Start vom Funktionstest. 0: Nein.
5	verzögerter Start vom Betriebsdauertest. 0: Nein.
6	Identifikation aktiv. 0: Nein.
7	ausgewählt. 0: Nein

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType` gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nEmergencyStatus : BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

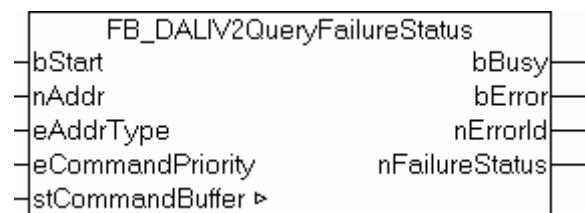
nEmergencyStatus: Zustand vom Notbetrieb (siehe Tabelle oben).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.184 FB_DALIV2QueryFailureStatus



Die Variable [FAILURE STATUS](#) [▶ 377] (Fehlerzustand) wird aus dem Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	Störung im Schaltkreis des Vorschaltgerätes. 0: Nein.
1	Störung Akkubetriebsdauer. 0: Nein.
2	Störung Akku. 0: Nein.
3	Störung am Leuchtmittel der Notbeleuchtung. 0: Nein.
4	Zeitüberschreitung beim Funktionstest aufgetreten. 0: Nein.
5	Zeitüberschreitung beim Betriebsdauertest aufgetreten. 0: Nein.
6	Funktionstest fehlgeschlagen. 0: Nein.
7	Betriebsdauertest fehlgeschlagen. 0: Nein

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nFailureStatus : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

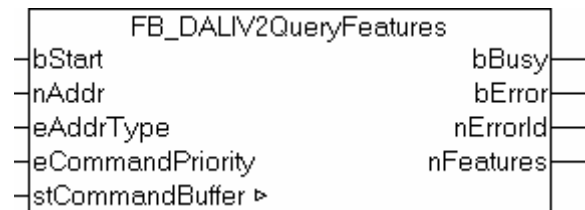
nFailureStatus: Der Fehlerzustand des Vorschaltgerätes (siehe Tabelle oben).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.185 FB_DALIV2QueryFeatures



Die Variable FEATURES [▶ 377] (Leistungsmerkmale) wird aus dem Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	integriertes Notlichtversorgungsgerät. 0: Nein.
1	Notlichtversorgungsgerät in Dauerschaltung. 0: Nein.
2	schaltbares Notlichtversorgungsgerät in Dauerschaltung. 0: Nein.
3	Autotestfähigkeit. 0: Nein.
4	Einstellbare Stärke der Notbeleuchtung. 0: Nein.
5	angeschlossener Unterdrückungstaster wird unterstützt. 0: Nein.
6	physikalische Auswahl wird unterstützt. 0: Nein.
7	Reserve



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nFeatures  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

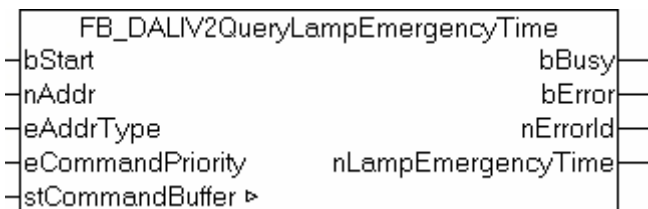
nFeatures: Leistungsmerkmale des Vorschaltgerätes (siehe Tabelle oben).

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.186 FB_DALIV2QueryLampEmergencyTime



Die Variable [LAMP EMERGENCY TIME](#) [▶ 377] (Notbetriebszeit der Lampe) wird aus dem Vorschaltgerät ausgelesen. Die Einheit beträgt 1 Stunde pro Schritt. Ein Wert von 255 bedeutet den maximalen Wert von 254 Stunden oder länger. Die Variable wird immer am Anfang des 1-Stunden-Intervalls erhöht. Wird der maximale Wert von 255 erreicht, so wird dieser nicht weiter erhöht. Durch den Befehl [FB_DALIV2ResetLampTime\(\)](#) [▶ 243] wird die Variable wieder zurückgesetzt.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nLampEmergencyTime : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

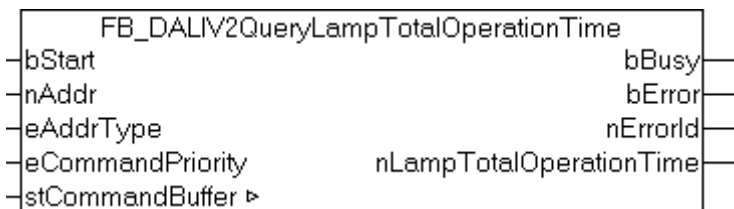
nLampEmergencyTime: Notbetriebszeit der Lampe vom Vorschaltgerät.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.187 FB_DALIV2QueryLampTotalOperationTime



Die Variable [LAMP TOTAL OPERATION TIME](#) [▶ 377] (gesamte Betriebszeit der Lampe) wird aus dem Vorschaltgerät ausgelesen. Die Einheit beträgt 4 Stunden pro Schritt. Ein Wert von 255 bedeutet den maximalen Wert von 1016 Stunden oder länger. Die Variable wird immer am Anfang des 4-Stunden-Intervalls erhöht. Wird der maximale Wert von 255 erreicht, so wird dieser nicht weiter erhöht. Durch den Befehl [FB_DALIV2ResetLampTime\(\)](#) [▶ 243] wird die Variable wieder zurückgesetzt.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nLampTotalOperationTime : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

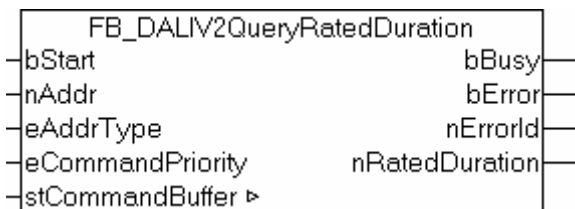
nLampTotalOperationTime: Gesamte Betriebszeit der Lampe vom Vorschaltgerät.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.188 FB_DALIV2QueryRatedDuration



Die Variable RATED_DURATION [▶ 377] (Nennbetriebsdauer) wird aus dem Vorschaltgerät ausgelesen. Die Einheit beträgt 2 Minuten pro Schritt.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nRatedDuration : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

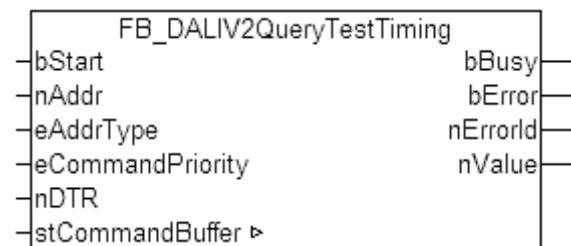
nRatedDuration: Nennbetriebsdauer vom Vorschaltgerät.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [► 95] (KL6811) oder FB_KL6821Communication() [► 104] (KL6821).

5.1.189 FB_DALIV2QueryTestTiming



Dieser Baustein liest die Prüfzeiten (Intervalle, Dauer bis zum nächsten Ereignis, etc.) aus dem Vorschaltgerät. Dies geschieht in Abhängigkeit vom Wert, der am Eingang *nDTR* anliegt. Der Hintergrund ist, dass das DALI-Grundkommando 242 "Query Test Timing" zusammen mit dem Inhalt des Data-Transfer-Register (DTR) funktioniert, welches vorher entsprechend beschrieben werden muss. Der Baustein *FB_DALIV2QueryTestTiming* fasst diese beiden Aktionen zusammen. Folgende Daten werden in Abhängigkeit vom Wert *nDTR* aus dem Gerät ausgelesen:

nDTR	ausgelesener Wert
0 (2#0000 0000)	Wenn Automatik Test aktiviert ist: Zeit bis zum nächsten Function-Test (high-byte) in Viertelstunden Wenn Automatiktest nicht aktiviert ist: MASK (255)
1 (2#0000 0001)	Wenn Automatik Test aktiviert ist: Zeit bis zum nächsten Function-Test (low-byte) in Viertelstunden Wenn Automatiktest nicht aktiviert ist: MASK (255)
2 (2#0000 0010)	Wenn Automatik Test aktiviert ist: Zeit bis zum nächsten Duration-Test (high-byte) in Viertelstunden Wenn Automatiktest nicht aktiviert ist: MASK (255)
3 (2#0000 0011)	Wenn Automatik Test aktiviert ist: Zeit bis zum nächsten Duration-Test (low-byte) in Viertelstunden Wenn Automatiktest nicht aktiviert ist: MASK (255)
4 (2#0000 0100)	Wenn Automatik Test aktiviert ist: Test-Intervall des Function-Test in Tagen Wenn Automatiktest nicht aktiviert ist: MASK (255)

nDTR	ausgelesener Wert
5 (2#0000 0101)	Wenn Automatik Test aktiviert ist: Test-Intervall des Duration-Test in Wochen Wenn Automatiktest nicht aktiviert ist: MASK (255)
6 (2#0000 0110)	Test-Ausführungs-Timeout in Tagen (maximaler Ausführungszeitraum für einen Test). Gilt nur für den Automatiktest!
7 (2#0000 0111)	Lampen-Ein-Verlängerungszeit nach Verlassen des Notfall-Betriebes, gemessen in 0,5 min-Schritten



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [► 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDTR        : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDTR: In Abhängigkeit dieses Eingangs werden entsprechende Werte gelesen, s.o. (Tabelle)

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nValue      : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

nValue: Ergebnis.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

5.1.190 FB_DALIV2ReLightResetInhibit



Das Vorschaltgerät wird zurück in den Notbetrieb geschaltet (wenn die Netzspannung fehlt). Hierdurch wird die Funktion des Bausteins `FB_DALIV2Inhibit()` [► 224] aufgehoben.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [► 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

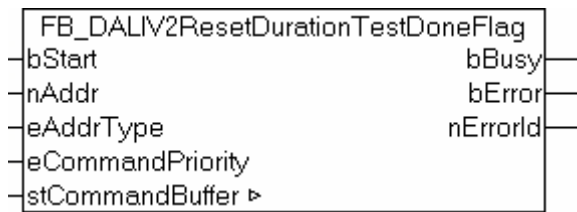
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

5.1.191 FB_DALIV2ResetDurationTestDoneFlag



Das Flag 'Betriebsdauertest abgeschlossen und Ergebnis ist gültig' wird zurückgesetzt. Das Flag ist in der Variable [EMERGENCY STATUS](#) [[▶ 378](#)] (Zustand Notbetrieb) Bit 2 und kann mit dem Baustein [FB_DALIQueryEmergencyStatus\(\)](#) [[▶ 232](#)] abgefragt werden.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [[▶ 356](#)] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

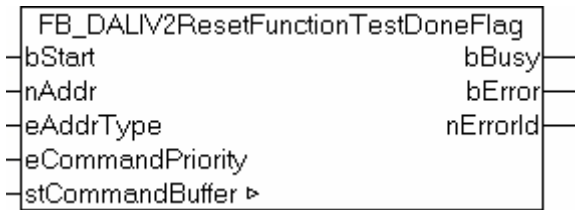
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [[▶ 95](#)] (KL6811) oder [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

5.1.192 FB_DALIV2ResetFunctionTestDoneFlag



Das Flag 'Funktionstest abgeschlossen und Ergebnis ist gültig' wird zurückgesetzt. Das Flag ist in der Variable [EMERGENCY STATUS](#) [[▶ 378](#)] (Zustand Notbetrieb) Bit 1 und kann mit dem Baustein [FB_DALIQueryEmergencyStatus\(\)](#) [[▶ 232](#)] abgefragt werden.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [[▶ 356](#)] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [[▶ 95](#)] (KL6811) oder [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

5.1.193 FB_DALIV2ResetLampTime



Die Variablen LAMP EMERGENCY TIME [▶ 377] (Notbetriebszeit der Lampe) und LAMP TOTAL OPERATION TIME [▶ 377] (gesamte Betriebszeit der Lampe) werden zurückgesetzt.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.194 FB_DALIV2Rest



Mit diesem Baustein wird die Lampe ausgeschaltet, wenn der Notbetrieb aktiv ist. Ansonsten wird auf Normalbetrieb umgeschaltet für den Fall das die Netzspannung wieder vorhanden ist oder der Baustein `FB_DALIV2ReLightResetInhibit()` [► 240] aufgerufen wurde.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [► 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [► 95] (KL6811) oder `FB_KL6821Communication()` [► 104] (KL6821).

5.1.195 FB_DALIV2StartDurationTest



Der Betriebsdauertest wird gestartet. Wird der Betriebsdauertest verzögert gestartet, so wird dieses in der Variable `EMERGENCY STATUS` [▶ 378] (Zustand Notbetrieb) Bit 5 angezeigt.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.196 FB_DALIV2StartFunctionTest



Der Funktionstest wird gestartet. Wird der Funktionstest verzögert gestartet, so wird dieses in der Variable EMERGENCY STATUS [▶ 378] (Zustand Notbetrieb) Bit 4 angezeigt.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.197 FB_DALIV2StopTest



Stoppt jede Art von Funktionstest oder Betriebsdauertest.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

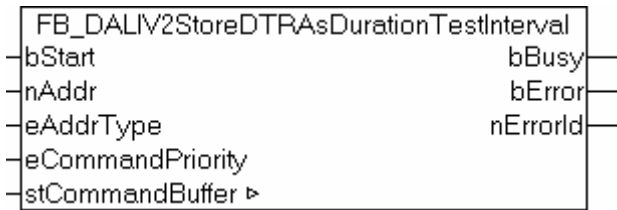
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.198 FB_DALIV2StoreDTRAsDurationTestInterval



Schreibt den Wert des DTR (Data Transfer Register) in die Variable DURATION TEST INTERVAL [▶ 376] (Intervallzeit Betriebsdauertest).



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.199 FB_DALIV2StoreDTRAsEmergencyLevel



Schreibt den Wert des DTR (Data Transfer Register) in die Variable EMERGENCY LEVEL [▶ 375] (Notbeleuchtungsstärke).



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

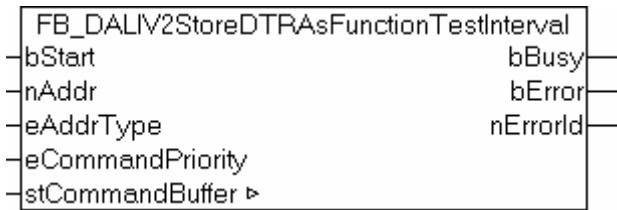
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.200 FB_DALIV2StoreDTRAsFunctionTestInterval



Schreibt den Wert des DTR (Data Transfer Register) in die Variable [FUNCTION TEST INTERVAL](#) [▶ 376] (Intervallzeit Funktionstest).



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.201 FB_DALIV2StoreDTRAsProlongTime



Schreibt den Wert des DTR (Data Transfer Register) in die Variable PROLONG TIME [▶ 375] (Verlängerungszeit).



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.202 FB_DALIV2StoreDTRAsTestDelayTimeHighByte



Schreibt den Wert des DTR (Data Transfer Register) in das höherwertige Byte der Variable TEST DELAY TIME [▶ 375].



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

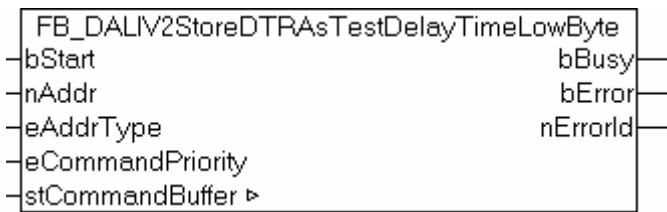
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.203 FB_DALIV2StoreDTRAsTestDelayTimeLowByte



Schreibt den Wert des DTR (Data Transfer Register) in das niederwertige Byte der Variable TEST DELAY TIME [▶ 375].



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

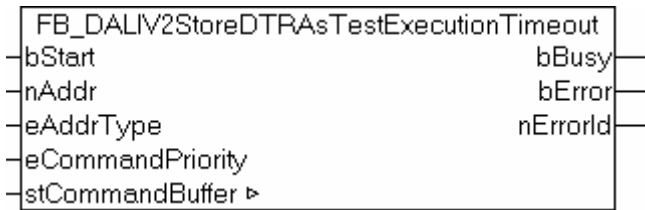
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.204 FB_DALIV2StoreDTRAsTestExecutionTimeout



Schreibt den Wert des DTR (Data Transfer Register) in die Variable [TEST EXECUTION TIMEOUT](#) [▶ 376] (maximale Ausführungszeit für einen Test).



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 1* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 1* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Notbeleuchtungen intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

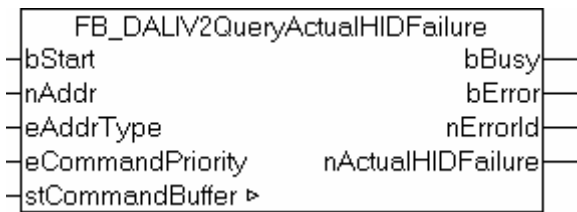
VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.205 FB_DALIV2QueryActualHIDFailure



Die Variable ACTUAL HID FAILURE [▶ 379] (aktueller Fehlerzustand) wird aus dem Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	Versorgungsspannung zu niedrig. 0: Nein.
1	Versorgungsspannung zu hoch. 0: Nein.
2	Übertemperatur am Umformer. 0: Nein.
3	Reserve
4	Zeitüberschreitung beim Zünden der Lampe. 0: Nein.
5	Reserve
6	Lampenspannung außerhalb der Spezifikation. 0: Nein.
7	Lampenzyklusfehler. 0: Nein.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nActualHIDFailure : BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

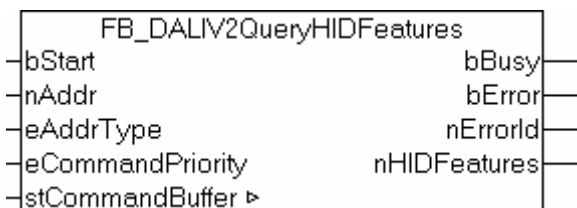
nActualHIDFailure: Inhalt der Variablen ACTUAL HID FAILURE.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.206 FB_DALIV2QueryHIDFeatures



Die Variable [HID FEATURES](#) [▶ 380] (Leistungsmerkmale) wird aus dem Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	'Versorgungsspannung zu niedrig' kann abgefragt werden. 0: Nein.
1	'Versorgungsspannung zu hoch' kann abgefragt werden. 0: Nein.
2	'Übertemperatur am Umformer' kann abgefragt werden. 0: Nein.
3	Reserve
4	Reserve
5	Reserve
6	'Lampenspannung außerhalb der Spezifikation' kann abgefragt werden. 0: Nein.
7	Physikalische Auswahl wird unterstützt. 0: Nein.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nHIDFeatures : BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

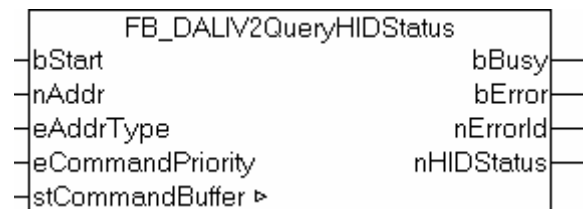
nHIDFeatures: Inhalt der Variable HID FEATURES.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.207 FB_DALIV2QueryHIDStatus



Die Variable [HID STATUS](#) [▶ 379] (Zustand) wird aus dem Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	Anlaufzeit betriebsbereit. 0: Nein.
1	Der Lampenleistungswert hat seinen angeforderten Sollwert erreicht. 0: Nein.
2	Vorschaltgerät wartet auf das Zünden der Lampe. 0: Nein.
3	Reserve
4	Reserve
5	Reserve
6	Identifikation aktiv. 0: Nein.
7	Reserve



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```

bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nHIDStatus : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

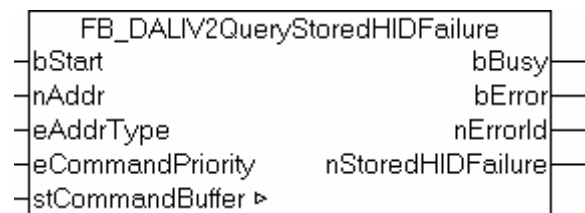
nHIDStatus: Inhalt der Variable HID STATUS.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.208 FB_DALIV2QueryStoredHIDFailure



Die Variable STORED HID FAILURE [▶ 379] (gespeicherter Fehlerzustand) wird aus dem Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	Versorgungsspannung zu niedrig. 0: Nein.
1	Versorgungsspannung zu hoch. 0: Nein.
2	Übertemperatur am Umformer. 0: Nein.
3	Reserve
4	Zeitüberschreitung beim Zünden der Lampe. 0: Nein.
5	Reserve
6	Lampenspannung außerhalb der Spezifikation. 0: Nein.
7	Lampenzyklusfehler. 0: Nein.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorgeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nStoredHIDFailure : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

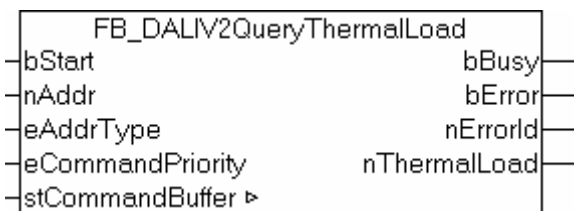
nStoredHIDFailure: Inhalt der Variable STORED HID FAILURE.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.209 FB_DALIV2QueryThermalLoad



Die Variable THERMAL_LOAD [▶ 380] wird ausgelesen. Diese beinhaltet einen Prozentwert im Bereich von 0% bis 127,5% in 0,5%-Schritten.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nThermalLoad : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

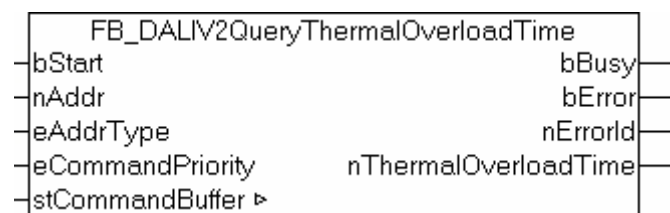
nThermalLoad: Inhalt der Variable THERMAL LOAD.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.210 FB_DALIV2QueryThermalOverloadTime



Die 16-Bit-Variable THERMAL OVERLOAD TIME [▶ 380] wird ausgelesen.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId      : UDINT;
nThermalOverloadTime : WORD;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

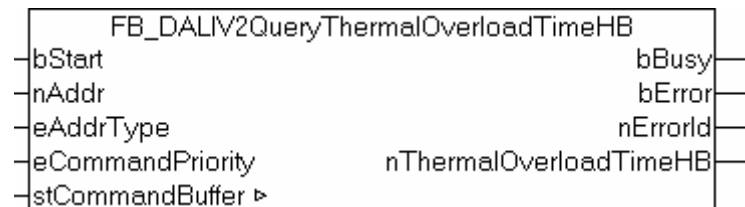
nThermalOverloadTime: Der Wert der 16-Bit-Variable THERMAL OVERLOAD TIME.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.211 FB_DALIV2QueryThermalOverloadTimeHB



Das höherwertige Byte der 16-Bit-Variable THERMAL OVERLOAD TIME [▶ 380] wird ausgelesen.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr          : BYTE;
eAddrType      : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId      : UDINT;
nThermalOverloadTimeHB : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

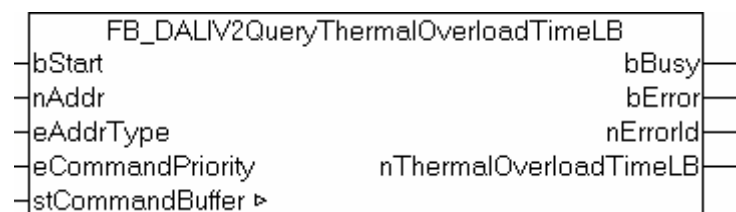
nThermalOverloadTimeHB: Das höherwertige Byte der Variable THERMAL OVERLOAD TIME.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.212 FB_DALIV2QueryThermalOverloadTimeLB



Das niederwertige Byte der 16-Bit-Variable THERMAL OVERLOAD TIME [▶ 380] wird ausgelesen.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nThermalOverloadTimeLB : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

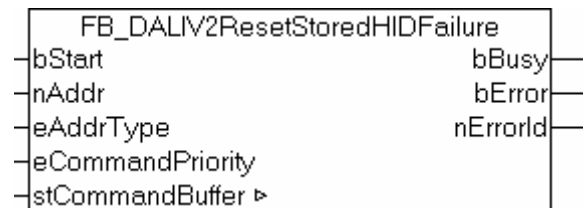
nThermalOverloadTimeLB: Das niederwertige Byte der Variable THERMAL OVERLOAD TIME.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.213 FB_DALIV2ResetStoredHIDFailure



Die Variable [STORED HID FAILURE](#) [▶ 379] (gespeicherte Lampenstörungen) wird zurückgesetzt.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 2* vorgeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 2* allen anwendungsbezogenen Erweiterungsbefehlen für DALI-Entladungsleuchten intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

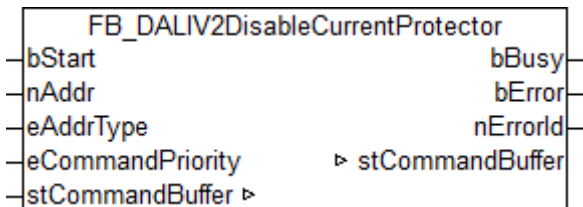
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

5.1.214 FB_DALIV2DisableCurrentProtector



Der Befehl sperrt die Stromsicherheit des Vorschaltgerätes.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Modulen mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

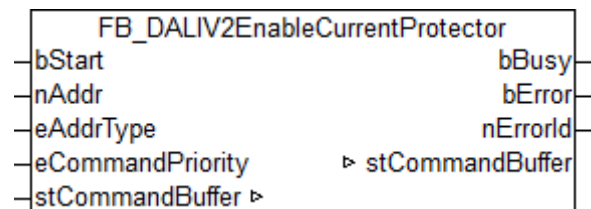
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.215 FB_DALIV2EnableCurrentProtector



Der Befehl gibt die Stromschutzeinrichtung des Vorschaltgerätes frei.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

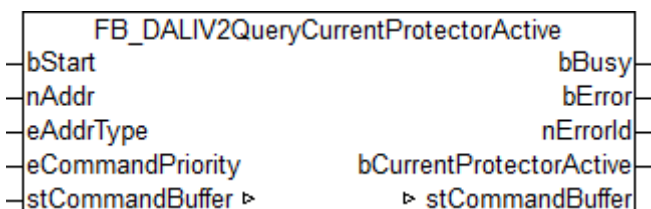
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.216 FB_DALIV2QueryCurrentProtectorActive



Es wird abgefragt, ob die Stromschutzeinrichtung aktiv ist.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
bCurrentProtectorActive : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

bCurrentProtectorActive: Stromschutzeinrichtung aktiv.

VAR_IN_OUT

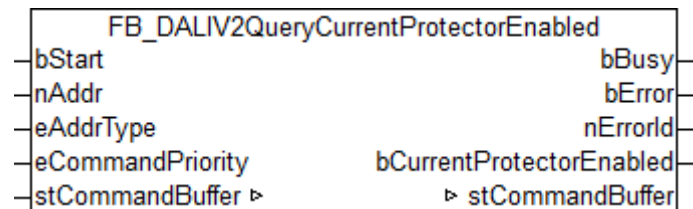
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.217 FB_DALIV2QueryCurrentProtectorEnabled



Es wird abgefragt, ob die Stromschutzeinrichtung freigegeben ist.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
bCurrentProtectorEnabled : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

bCurrentProtectorEnabled: Stromschutzeinrichtung freigegeben.

VAR_IN_OUT

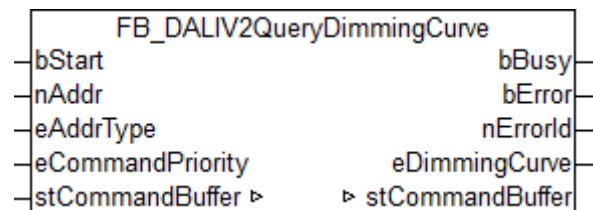
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.218 FB_DALIV2QueryDimmingCurve



Die Dimmkurve des Vorschaltgerätes wird ausgelesen.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
eDimmingCurve : E_DALIV2DimmingCurve;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

eDimmingCurve: [Dimmkurve](#) [▶ 388] (liniar oder logarithmisch).

VAR_IN_OUT

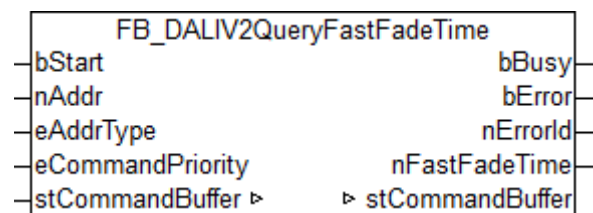
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.219 FB_DALIV2QueryFastFadeTime



Fragt den Wert von FAST FADE TIME ab.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nFastFadeTime : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nFastFadeTime: Wert von FAST FADE TIME.

VAR_IN_OUT

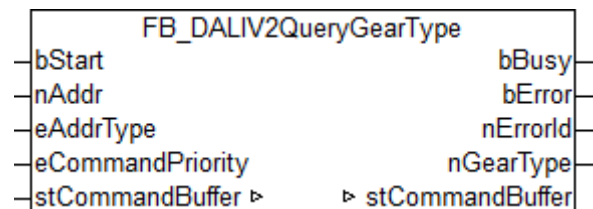
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.220 FB_DALIV2QueryGearType



Fragt den Wert von GEAR TYPE ab.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nGearType  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nGearType: Wert von GEAR TYPE.

VAR_IN_OUT

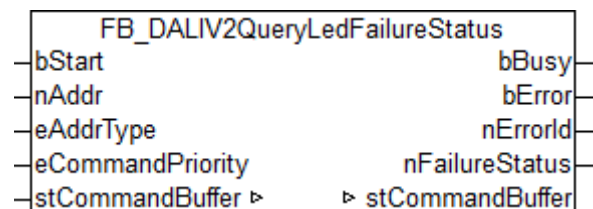
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.221 FB_DALIV2QueryLedFailureStatus



Fragt den Wert von FAILURE STATUS ab.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nFailureStatus: BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nFailureStatus: Wert von FAILURE STATUS.

VAR_IN_OUT

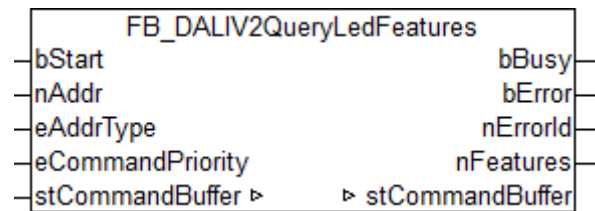
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.222 FB_DALIV2QueryLedFeatures



Fragt den Wert von FEATURES ab.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nFeatures  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nFeatures: Wert von FEATURES.

VAR_IN_OUT

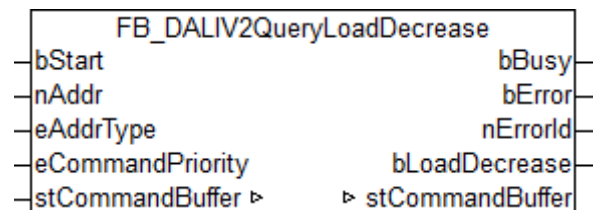
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.223 FB_DALIV2QueryLoadDecrease



Es wird abgefragt, ob eine signifikante Lastabnahme (im Vergleich zur Referenzleistung des Systems) erkannt wurde.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bLoadDecrease : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bLoadDecrease: Lastabnahme.

VAR_IN_OUT

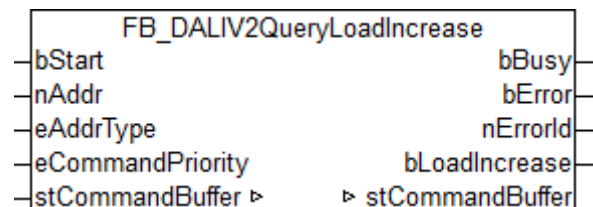
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.224 FB_DALIV2QueryLoadIncrease



Es wird abgefragt, ob ein signifikanter Lastanstieg (im Vergleich zur Referenzleistung des Systems) erkannt wurde.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bLoadIncrease : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bLoadIncrease: Lastanstieg.

VAR_IN_OUT

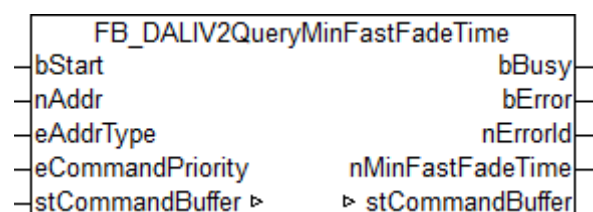
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.225 FB_DALIV2QueryMinFastFadeTime



Fragt den Wert von MIN FAST FADE TIME ab.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nMinFastFadeTime : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nMinFastFadeTime: Wert von MIN FAST FADE TIME.

VAR_IN_OUT

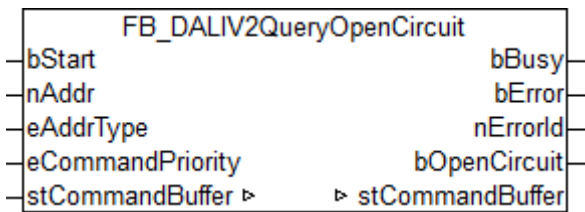
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.226 FB_DALIV2QueryOpenCircuit



Es wird abgefragt, ob ein Leerlauf erkannt wurde.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bOpenCircuit : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bOpenCircuit: Leerlauf.

VAR_IN_OUT

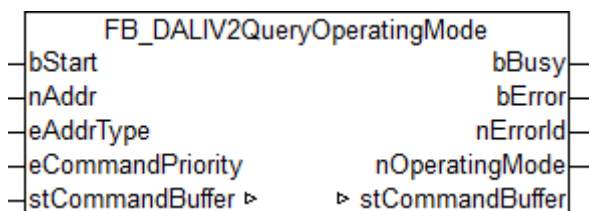
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.227 FB_DALIV2QueryOperatingMode



Fragt den Wert von OPERATING MODE ab.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nOperatingMode: BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nOperatingMode: Wert von OPERATING MODE.

VAR_IN_OUT

```

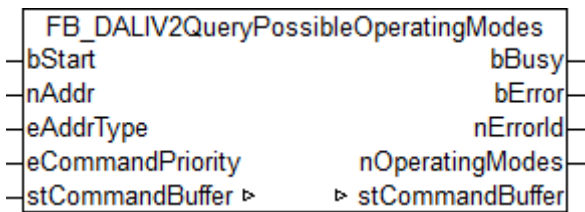
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.228 FB_DALIV2QueryPossibleOperatingModes



Fragt den Wert von OPERATING MODE ab.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
nOperatingModes : BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nOperatingModes: Wert von OPERATING MODE.

VAR_IN_OUT

```

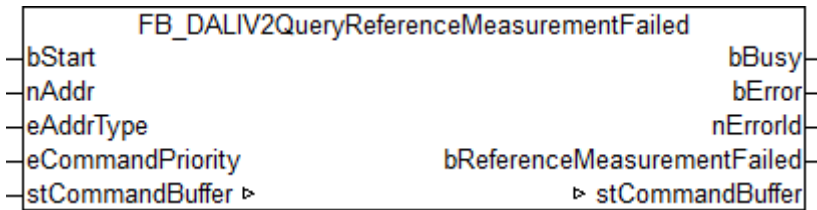
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.229 FB_DALIV2QueryReferenceMeasurementFailed



Es wird abgefragt, ob eine gestartete Referenzmessung fehlgeschlagen ist.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock **FB_DALIV2EnableDeviceType()** [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy           : BOOL;
bError          : BOOL;
nErrorId        : UDINT;
bReferenceMeasurementFailed : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bReferenceMeasurementFailed: Referenzmessung fehlgeschlagen.

VAR_IN_OUT

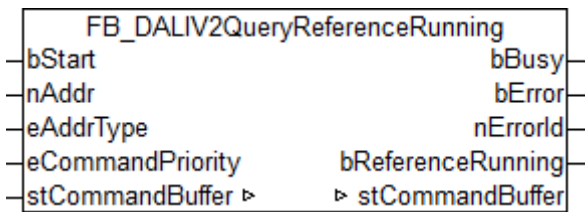
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen **FB_DALIV2Communication()** [▶ 95] (KL6811) oder **FB_KL6821Communication()** [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.230 FB_DALIV2QueryReferenceRunning



Es wird abgefragt eine Referenzmessung der Systemleistung aktiv ist.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse, Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy           : BOOL;
bError          : BOOL;
nErrorId        : UDINT;
bReferenceRunning : BOOL;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

bReferenceRunning: Referenzmessung der Systemleistung aktiv.

VAR_IN_OUT

```

stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

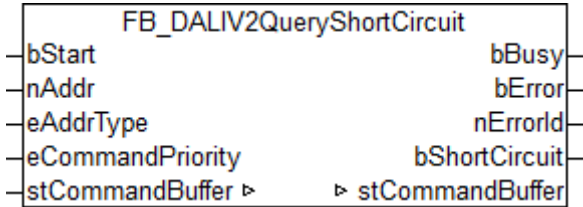
Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

Sehen Sie dazu auch

- 📄 E_DALIV2AddrType [▶ 387]
- 📄 E_DALIV2CommandPriority [▶ 387]

5.1.231 FB_DALIV2QueryShortCircuit



Es wird abgefragt, ob ein Kurzschluss erkannt wurde.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy           : BOOL;
bError          : BOOL;
nErrorId        : UDINT;
bShortCircuit   : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

bShortCircuit: Kurzschluss.

VAR_IN_OUT

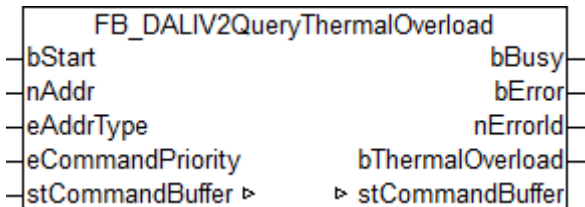
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.232 FB_DALIV2QueryThermalOverload



Es wird abgefragt ob eine thermische Überlast mit Reduzierung des Lichtstroms vorliegt.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bThermalOverload : BOOL;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bThermalOverload: thermische Überlast.

VAR_IN_OUT

```

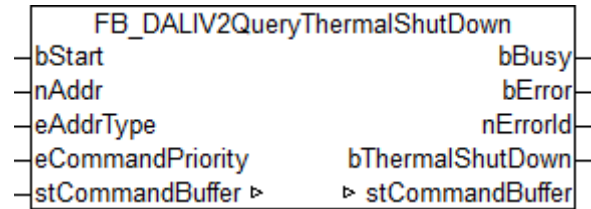
stCommandBuffer : ST_DALIV2CommandBuffer;
  
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen
 FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.233 FB_DALIV2QueryThermalShutDown



Es wird abgefragt, ob eine thermische Abschaltung erfolgte.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Modulen mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\) \[▶ 356\]](#) gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Modulen mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[▶ 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
bThermalShutDown : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

bThermalShutDown: thermische Abschaltung.

VAR_IN_OUT

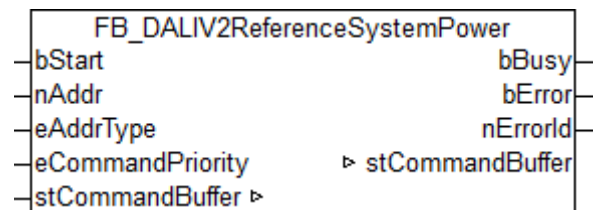
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.234 FB_DALIV2ReferenceSystemPower



Das Vorschaltgerät misst das Leistungsniveaus des Systems um Lastanstieg und Lastabnahme zu erkennen und speichert dieses ab.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

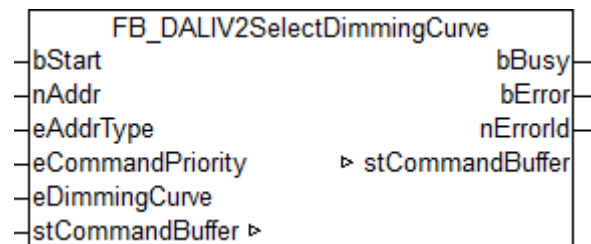
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.235 FB_DALIV2SelectDimmingCurve



Die Dimmkurve des Vorschaltgerätes wird ausgewählt.

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
eDimmingCurve : E_DALIV2DimmingCurve := eDALIV2DimmingCurveLogarithmic;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

eDimmingCurve: liniare oder logarithmische Dimmkurve [▶ 388].

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

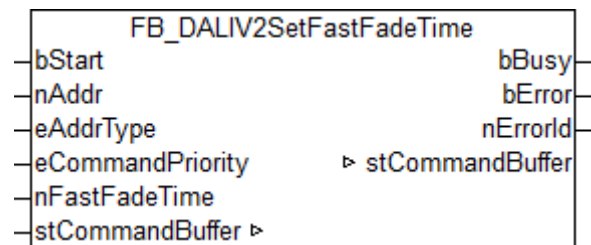
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.236 FB_DALIV2SetFastFadeTime



Setzt die FAST FADE TIME im Vorschaltgerät.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 6* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [► 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 6* allen anwendungsbezogenen Erweiterungsbefehlen für LED-Module mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nFastFadeTime : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nFastFadeTime: Der neue Wert für die FAST FADE TIME (0 - 27).

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

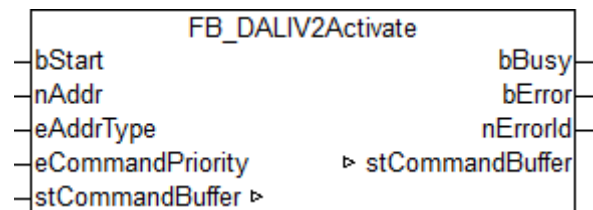
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.237 FB_DALIV2Activate



Eine laufende Überblendung wird beendet und eine neue Überblendung gestartet. Hierbei wird nur die Farbe geändert.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur Tc
- Primär(farbe) N
- RGBWAF

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

VAR_IN_OUT

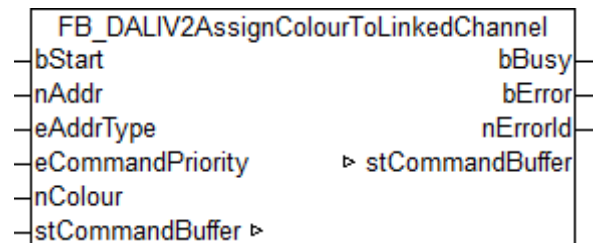
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [[▶ 95](#)] (KL6811) oder FB_KL6821Communication() [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.238 FB_DALIV2AssignColourToLinkedChannel



Verknüpfte Ausgangskanäle werden der festgelegten Farbe (siehe Tabelle) zugeordnet. Die verknüpften Kanäle werden durch Bit 0 bis Bit 5 der Variablen TEMPORARY RGBWAF CONTROL [[▶ 380](#)] angegeben. Die Kanalzuordnung wird nicht geändert, wenn TEMPORARY RGBWAF CONTROL den Wert 255 (MASK) enthält. Alle TEMPORARY COLOUR SETTINGS werden nach der Verwendung dieses Befehls auf MASK gesetzt.

Wert	Beschreibung
0	keine Farbe zugeordnet
1	Rot
2	Grün
3	Blau
4	Weiß
5	Bernsteingelb
6	frei wählbare Farbe

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) verändert.

Der Baustein unterstützt die Farbdarstellungen:

- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nColour     : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nColour: Farbe, die den Ausgangskanälen zugeordnet wird (siehe Tabelle).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

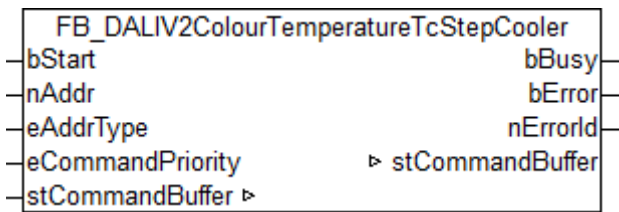
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.239 FB_DALIV2ColourTemperatureTcStepCooler



Der Wert COLOUR TEMPERATURE Tc [▶ 380] wird ohne Überblendung um 1 Mirek verringert. Falls der Wert COLOUR TEMPERATURE Tc [▶ 380] bereits den Wert COLOUR TEMPERATURE Tc COOLEST [▶ 380] hat, erfolgt keine Änderung.

Kann die Farbtemperatur von dem DALI-Vorschaltgerät nicht erreicht werden, so wird in COLOUR STATUS [▶ 380] Bit 1 (Colour temperature Tc out of range) gesetzt. Dieser Befehl wird vom DALI-Vorschaltgerät nur dann ausgeführt, wenn in der Variablen COLOUR STATUS [▶ 380] das Bit 5 (Colour type colour temperature Tc active) gesetzt ist.

Für die Umrechnung von oder nach Kelvin stehen die Funktionen KELVIN TO MIREK [▶ 326] und MIREK TO KELVIN [▶ 326] zur Verfügung.

Der Baustein unterstützt die Farbdarstellungen:

- Farbtemperatur Tc



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken TcDALIV2 und TcDALIV2AppExtCmds jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

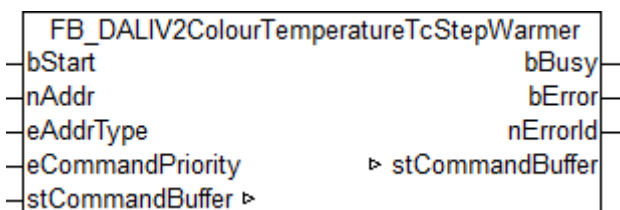
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.240 FB_DALIV2ColourTemperatureTcStepWarmer



Der Wert COLOUR TEMPERATURE Tc [▶ 380] wird ohne Überblendung um 1 Mirek erhöht. Falls der Wert COLOUR TEMPERATURE Tc [▶ 380] bereits den Wert COLOUR TEMPERATURE Tc WARMEST [▶ 380] hat, erfolgt keine Änderung.

Kann die Farbtemperatur von dem DALI-Vorschaltgerät nicht erreicht werden, so wird in COLOUR STATUS [▶ 380] Bit 1 (Colour temperature Tc out of range) gesetzt. Dieser Befehl wird vom DALI-Vorschaltgerät nur dann ausgeführt, wenn in der Variablen COLOUR STATUS [▶ 380] das Bit 5 (Colour type colour temperature Tc active) gesetzt ist.

Für die Umrechnung von oder nach Kelvin stehen die Funktionen KELVIN TO MIREK [▶ 326] und MIREK TO KELVIN [▶ 326] zur Verfügung.

Der Baustein unterstützt die Farbdarstellungen:

- Farbtemperatur Tc



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken TcDALIV2 und TcDALIV2AppExtCmds jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) | 391].

VAR_IN_OUT

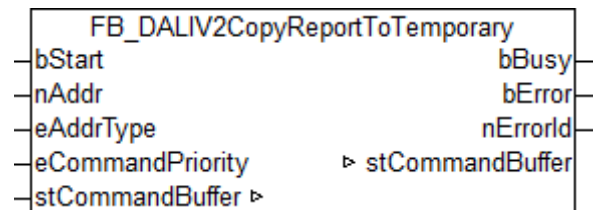
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) | 95] (KL6811) oder [FB_KL6821Communication\(\)](#) | 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.241 FB_DALIV2CopyReportToTemporary



Die Meldung Farbeinstellungen wird zu den temporären Farbeinstellungen kopiert.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur Tc
- Primär(farbe) N
- RGBWAF

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) | 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId  : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

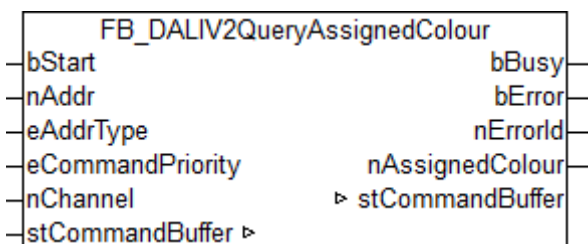
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.242 FB_DALIV2QueryAssignedColour



Die Variable ASSIGNED COLOUR [▶ 384] wird aus dem DALI-Vorschaltgerät ausgelesen. Diese enthält die zugewiesene Farbe (siehe Tabelle) des vorgegebenen Ausgangskanals (0 - 5) aus. Wird eine nicht existierende Kanalnummer angegeben, so wird 255 (MASK) zurückgeliefert.

Wert	Beschreibung
0	keine Farbe zugeordnet
1	Rot
2	Grün
3	Blau
4	Weiß
5	Bernsteingelb
6	frei wählbare Farbe

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) verändert.

Der Baustein unterstützt die Farbdarstellungen:

- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nChannel    : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nChannel: Kanalnummer (0 - 5).

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nAssignedColour : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nAssignedColour: zugewiesene Farbe des Kanals (siehe Tabelle).

VAR_IN_OUT

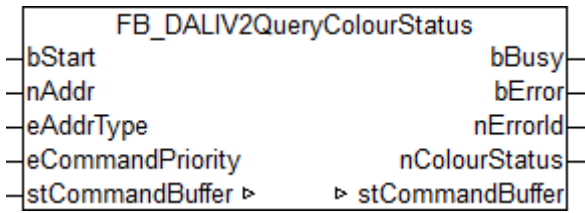
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.243 FB_DALIV2QueryColourStatus



Die Variable COLOUR STATUS [▶ 385] wird aus dem DALI-Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	xy-coordinate-Farbpunkt liegt außerhalb des gültigen Bereichs.
1	Farbtemperatur Tc liegt außerhalb des gültigen Bereichs.
2	automatische Kalibrierung ist aktiv.
3	automatische Kalibrierung war erfolgreich.
4	Farbdarstellung <i>xy-coordinate</i> aktiv.
5	Farbdarstellung <i>colour temperature Tc</i> aktiv.
6	Farbdarstellung <i>primary N</i> aktiv.
7	Farbdarstellung <i>RGBWAF</i> aktiv.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur Tc
- Primär(farbe) N
- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nColourStatus : BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nColourStatus: Farbstatus (siehe Tabelle oben).

VAR_IN_OUT

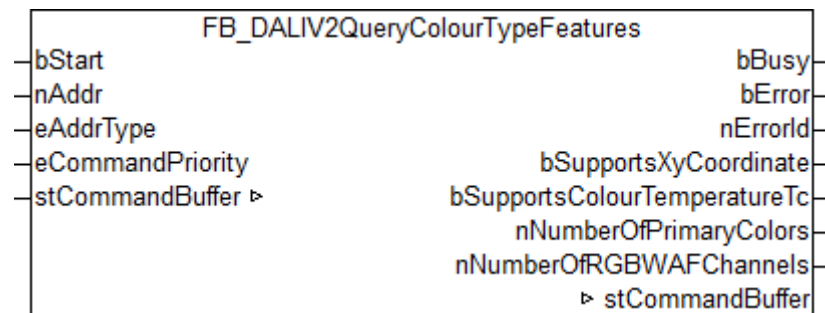
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.244 FB_DALIV2QueryColourTypeFeatures



Die Variable [COLOUR TYPE FEATURES](#) [▶ 386] wird aus dem DALI-Vorschaltgerät ausgelesen. Dieser enthält die vom DALI-Vorschaltgerät unterstützten Farbdarstellungen.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur Tc
- Primär(farbe) N
- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy           : BOOL;
bError          : BOOL;
nErrorId        : UDINT;
bSupportsXyCoordinate : BOOL;
bSupportsColourTemperatureTc : BOOL;
nNumberOfPrimaryColors : BYTE;
nNumberOfRGBWAFChannels : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bSupportsXyCoordinate: Das Vorschaltgerät ist *xy-coordinate* fähig.

bSupportsColourTemperatureTc: Das Vorschaltgerät ist *Colour temperature Tc* fähig.

nNumberOfPrimaryColors: Anzahl der vom Vorschaltgerät unterstützten Primärfarben. Ein Wert von 0 bedeutet, dass diese Farbdarstellung nicht unterstützt wird.

nNumberOfRGBWAFChannels: Anzahl der vom Vorschaltgerät unterstützten RGBWAF-Kanäle. Ein Wert von 0 bedeutet, dass diese Farbdarstellung nicht unterstützt wird.

VAR_IN_OUT

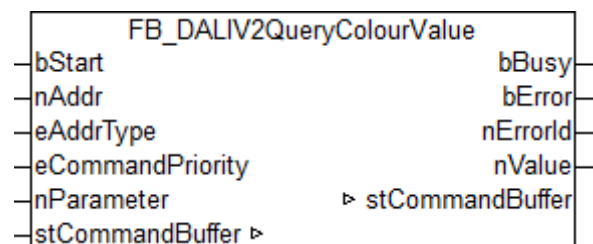
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.245 FB_DALIV2QueryColourValue



Die vorgegebene Variable (Farbwert) wird aus dem DALI-Vorschaltgerät ausgelesen. Der auszulesende Wert, wird durch *nParameter* definiert (siehe Tabelle unten).

Bestimmte Variablen können direkt über DALI-Befehle ausgelesen werden (z.B. `FB_QueryColourStatus()` [[▶ 296](#)] oder `FB_QueryRGBWAFControl()` [[▶ 303](#)]). Weitere Einzelheiten zu den Variablen finden sie [hier](#) [[▶ 380](#)].

Wert	Beschreibung
0	x-COORDINATE
1	y-COORDINATE
2	COLOUR TEMPERATURE T _c
3	PRIMARY N DIMLEVEL 0
4	PRIMARY N DIMLEVEL 1
5	PRIMARY N DIMLEVEL 2
6	PRIMARY N DIMLEVEL 3
7	PRIMARY N DIMLEVEL 4
8	PRIMARY N DIMLEVEL 5
9	RED DIMLEVEL
10	GREEN DIMLEVEL
11	BLUE DIMLEVEL
12	WHITE DIMLEVEL
13	AMBER DIMLEVEL
14	FREECOLOUR DIMLEVEL
15	RGBWAF CONTROL
64	x-COORDINATE PRIMARY N 0
65	y-COORDINATE PRIMARY N 0
66	TY PRIMARY N 0
67	x-COORDINATE PRIMARY N 1
68	y-COORDINATE PRIMARY N 1
69	TY PRIMARY N 1
70	x-COORDINATE PRIMARY N 2
71	y-COORDINATE PRIMARY N 2
72	TY PRIMARY N 2
73	x-COORDINATE PRIMARY N 3
74	y-COORDINATE PRIMARY N 3
75	TY PRIMARY N 3
76	x-COORDINATE PRIMARY N 4
77	y-COORDINATE PRIMARY N 4
78	TY PRIMARY N 4
79	x-COORDINATE PRIMARY N 5
80	y-COORDINATE PRIMARY N 5
81	TY PRIMARY N 5
82	NUMBER OF PRIMARIES
128	COLOUR TEMPERATURE T _c COOLEST
129	COLOUR TEMPERATURE T _c PHYSICAL COOLEST
130	COLOUR TEMPERATURE T _c WARMEST
131	COLOUR TEMPERATURE T _c PHYSICAL WARMEST
192	TEMPORARY x-COORDINATE
193	TEMPORARY y-COORDINATE
194	TEMPORARY COLOUR TEMPERATURE T _c
195	TEMPORARY PRIMARY N DIMLEVEL 0
196	TEMPORARY PRIMARY N DIMLEVEL 1
197	TEMPORARY PRIMARY N DIMLEVEL 2

Wert	Beschreibung
198	TEMPORARY PRIMARY N DIMLEVEL 3
199	TEMPORARY PRIMARY N DIMLEVEL 4
200	TEMPORARY PRIMARY N DIMLEVEL 5
201	TEMPORARY RED DIMLEVEL
202	TEMPORARY GREEN DIMLEVEL
203	TEMPORARY BLUE DIMLEVEL
204	TEMPORARY WHITE DIMLEVEL
205	TEMPORARY AMBER DIMLEVEL
206	TEMPORARY FREECOLOUR DIMLEVEL
207	TEMPORARY RGBWAF CONTROL
208	TEMPORARY COLOUR TYPE
224	REPORT x-COORDINATE
225	REPORT y-COORDINATE
226	REPORT COLOUR TEMPERATURE T _c
227	REPORT PRIMARY N DIMLEVEL 0
228	REPORT PRIMARY N DIMLEVEL 1
229	REPORT PRIMARY N DIMLEVEL 2
230	REPORT PRIMARY N DIMLEVEL 3
231	REPORT PRIMARY N DIMLEVEL 4
232	REPORT PRIMARY N DIMLEVEL 5
233	REPORT RED DIMLEVEL
234	REPORT GREEN DIMLEVEL
235	REPORT BLUE DIMLEVEL
236	REPORT WHITE DIMLEVEL
237	REPORT AMBER DIMLEVEL
238	REPORT FREECOLOUR DIMLEVEL
239	REPORT RGBWAF CONTROL
240	REPORT COLOUR TYPE

Antworten, die eine aktive Farbdarstellung betreffen, sind nur gültig, wenn die Farbdarstellung des angeforderten Farbwertes aktiv ist (siehe `FB_DALIV2QueryColourStatus()` [▶ 296]) oder das Vorschaltgerät in der Lage ist, eine Neuberechnung des geforderten Farbwertes von der aktiven Farbdarstellung in einen Farbwert einer anderen Farbdarstellung durchzuführen.

Falls das Vorschaltgerät die Koordinaten nicht kennt oder die Primärfarbe nicht vorliegt, muss die Antwort 255 (MASK) sein.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) und DTR1 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur T_c
- Primär(farbe) N
- RGBWAF

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr      : BYTE;
eAddrType  : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nParameter : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nParameter: Farbwert, der ausgelesen werden soll (siehe Tabelle oben).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nValue     : UINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nValue: Enthält den ausgelesenen Wert.

VAR_IN_OUT

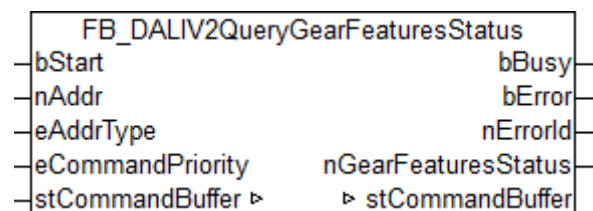
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.246 FB_DALIV2QueryGearFeaturesStatus



Die Variable GEAR FEATURES/STATUS [▶ 385] wird aus dem DALI-Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	automatische Aktivierung.

Bit	Beschreibung
1 - 5	reserviert.
6	automatische Kalibrierung wird unterstützt.
7	Wiederherstellung der automatischen Kalibrierung wird unterstützt.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur Tc
- Primär(farbe) N
- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nGearFeaturesStatus : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nGearFeaturesStatus: Statusinformationen (siehe Tabelle oben).

VAR_IN_OUT

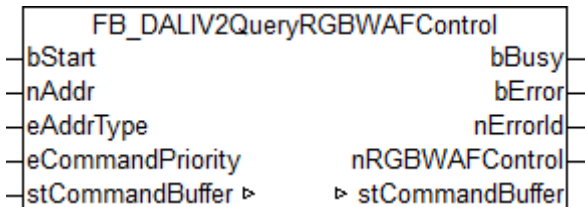
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.247 FB_DALIV2QueryRGBWAFControl



Die Variable `RGBWAF_CONTROL` [▶ 386] wird aus dem DALI-Vorschaltgerät ausgelesen.

Bit	Beschreibung
0	Ausgangskanal 0 / Rot
1	Ausgangskanal 1 / Grün
2	Ausgangskanal 2 / Blau
3	Ausgangskanal 3 / Weiß
4	Ausgangskanal 4 / Bernstein gelb
5	Ausgangskanal 5 / Frei wählbare Farbe
6 - 7	00 = Kanalsteuerung 01 = Farbsteuerung 10 = Standardisierte Farbsteuerung 11 = Reserviert

Falls ein Ausgangskanal bzw. eine Farbe nicht unterstützt wird, so ist das entsprechende Bit FALSE.

Der Baustein unterstützt die Farbdarstellungen:

- RGBWAF

i Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nRGBWAFControl: BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nRGBWAFControl: Informationen über die Kanalzuordnung (siehe Tabelle oben).

VAR_IN_OUT

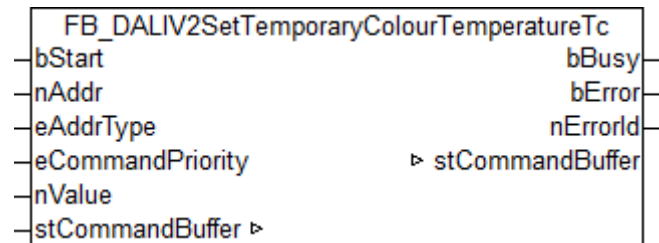
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.248 FB_DALIV2SetTemporaryColourTemperatureTc



Speichert den Wert in die Variable TEMPORARY COLOUR TEMPERATURE Tc [▶ 380] des DALI-Vorschaltgerätes. Der Wert kann mit dem Funktionsblock FB_DALIV2QueryColourValue() [▶ 298] ausgelesen werden.

Der Wert wird in Einheiten von 1 Mirek ausgedrückt. Ein Wert von 0 wird ignoriert und deshalb nicht gespeichert. Die Farbtemperatur Tc kann von 1 Mirek (1000000 K) bis 65534 Mirek (15,26 K) variieren. Für die Umrechnung von oder nach Kelvin stehen die Funktionen KELVIN TO MIREK [▶ 326] und MIREK TO KELVIN [▶ 326] zur Verfügung.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) und DTR1 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- Farbtemperatur Tc



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nValue      : UINT;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nValue: Der Wert, der in die Variable TEMPORARY COLOUR TEMPERATURE Tc geschrieben wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

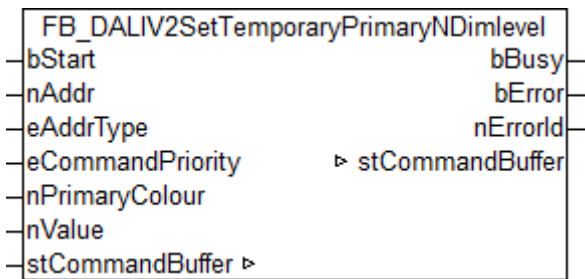
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.249 FB_DALIV2SetTemporaryPrimaryNDimlevel



Speichert den Wert in die Variable `TEMPORARY PRIMARY N DIMLEVEL` [▶ 380] des Vorschaltgerätes. Der Wert kann mit dem Funktionsblock `FB_DALIV2QueryColourValue()` [▶ 298] ausgelesen werden.

Der Wert wird in Einheiten von 1 / 65536 ausgedrückt. Der Höchstwert des PRIMARY N DIMLEVEL ist 0,99997 und wird auf einem linearen Maßstab interpretiert.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register), DTR1 und DTR2 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- Primär(farbe) N



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority: E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nPrimaryColour : BYTE;
nValue      : UINT;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nPrimaryColour: Primärfarbe (0 - 5).

nValue: Der Wert, der in die Variable `TEMPORARY PRIMARY N DIMLEVEL` geschrieben wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

VAR_IN_OUT

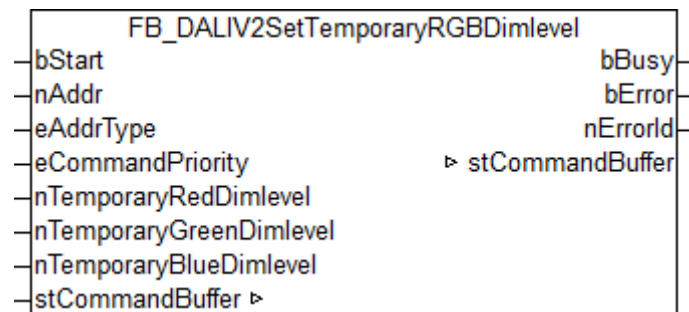
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [► 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.250 FB_DALIV2SetTemporaryRGBDimlevel



Speichert die Werte in die Variablen [TEMPORARY RED DIMLEVEL](#) [► 380], [TEMPORARY GREEN DIMLEVEL](#) [► 380] und [TEMPORARY BLUE DIMLEVEL](#) [► 380] des DALI-Vorschaltgerätes. Die Werte können mit dem Funktionsblock [FB_DALIV2QueryColourValue\(\)](#) [► 298] ausgelesen werden.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register), DTR1 und DTR2 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [► 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken [TcDALIV2](#) und [TcDALIV2AppExtCmds](#) jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nTemporaryRedDimlevel : BYTE;
nTemporaryGreenDimlevel : BYTE;
nTemporaryBlueDimlevel : BYTE;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ [387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nTemporaryRedDimlevel: Der neue Wert für TEMPORARY RED DIMLEVEL.

nTemporaryGreenDimlevel: Der neue Wert für TEMPORARY GREEN DIMLEVEL.

nTemporaryBlueDimlevel: Der neue Wert für TEMPORARY BLUE DIMLEVEL.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ [391](#)].

VAR_IN_OUT

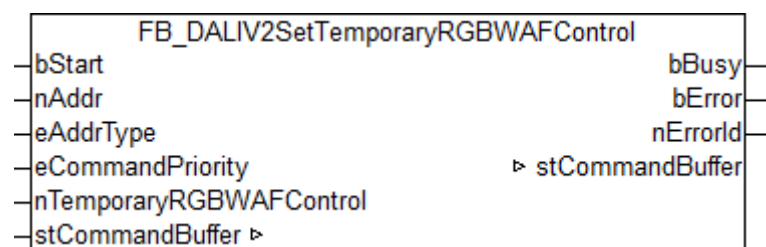
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ [95](#)] (KL6811) oder FB_KL6821Communication() [▶ [104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.251 FB_DALIV2SetTemporaryRGBWAFControl



Speichert den Wert in die Variable TEMPORARY RGBWAF CONTROL [▶ [380](#)] des DALI-Vorschaltgerätes. Der Wert kann mit dem Funktionsblock FB_DALIV2QueryColourValue() [▶ [298](#)] ausgelesen werden. Der Eingang *nTemporaryRGBWAFControl* enthält dabei die neue Zuordnung (siehe Tabelle).

Bit	Beschreibung
0	Ausgangskanal 0 / Rot
1	Ausgangskanal 1 / Grün
2	Ausgangskanal 2 / Blau
3	Ausgangskanal 3 / Weiß
4	Ausgangskanal 4 / Bernstein gelb
5	Ausgangskanal 5 / Frei wählbare Farbe

Bit	Beschreibung
6 - 7	00 = Kanalsteuerung 01 = Farbsteuerung 10 = Standardisierte Farbsteuerung 11 = Reserviert

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) verändert.

Der Baustein unterstützt die Farbdarstellungen:

- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nTemporaryRGBWAFControl : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nTemporaryRGBWAFControl: Enthält die Zuordnung (siehe Tabelle oben).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

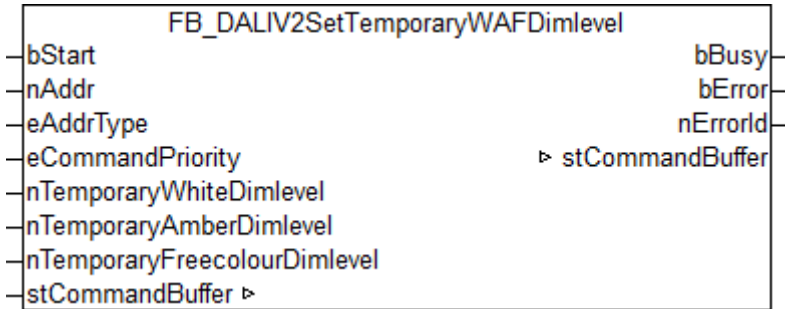
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.252 FB_DALIV2SetTemporaryWAFDimlevel



Speichert die Werte in die Variablen TEMPORARY WHITE DIMLEVEL [▷ 380], TEMPORARY AMBER DIMLEVEL [▷ 380] und TEMPORARY FREECOLOUR DIMLEVEL [▷ 380] des DALI-Vorschaltgerätes. Die Werte können mit dem Funktionsblock FB_DALIV2QueryColourValue() [▷ 298] ausgelesen werden.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register), DTR1 und DTR2 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▷ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nTemporaryWhiteDimlevel : BYTE;
nTemporaryAmberDimlevel : BYTE;
nTemporaryFreecolourDimlevel : BYTE;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▷ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▷ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nTemporaryWhiteDimlevel: Der neue Wert für TEMPORARY WHITE DIMLEVEL.

nTemporaryAmberDimlevel: Der neue Wert für TEMPORARY AMBER DIMLEVEL.

nTemporaryFreecolourDimlevel: Der neue Wert für TEMPORARY FREECOLOUR DIMLEVEL.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

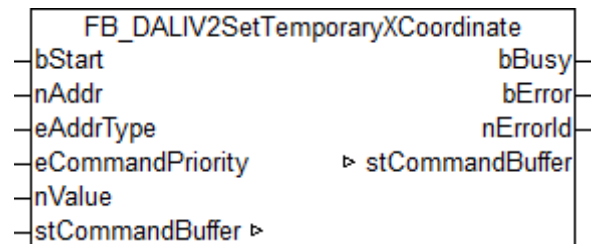
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.253 FB_DALIV2SetTemporaryXCoordinate



Speichert den Wert in die Variable [TEMPORARY x-COORDINATE](#) [▶ 380] des DALI-Vorschaltgerätes. Der Wert kann mit dem Funktionsblock [FB_DALIV2QueryColourValue\(\)](#) [▶ 298] ausgelesen werden.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) und DTR1 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Primär(farbe) N



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nValue      : UINT;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nValue: Der Wert, der in die Variable TEMPORARY x-COORDINATE geschrieben wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

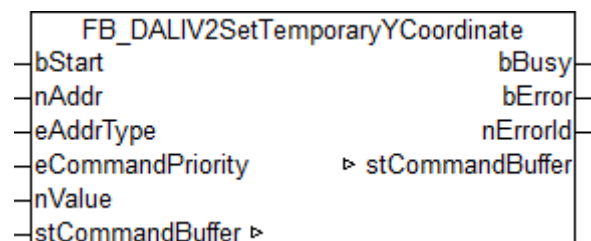
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.254 FB_DALIV2SetTemporaryYCoordinate



Speichert den Wert in die Variable TEMPORARY y-COORDINATE [▶ 380] des DALI-Vorschaltgerätes. Der Wert kann mit dem Funktionsblock FB_DALIV2QueryColourValue() [▶ 298] ausgelesen werden.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) und DTR1 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Primär(farbe) N



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nValue      : UINT;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nValue: Der Wert, der in die Variable TEMPORARY y-COORDINATE geschrieben wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

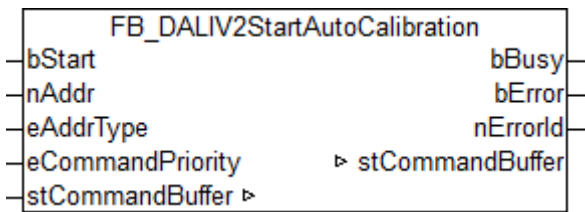
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.255 FB_DALIV2StartAutoCalibration



Das Kalibrierungsverfahren wird gestartet um die x-Koordinate, die y-Koordinate und den TY-Wert aller unterstützten Primärfarben zu messen.

Der Befehl startet einen 15-min-Zeitgeber oder stößt ihn erneut an. Bit 2 der Variablen `COLOUR STATUS` [▶ 385] ist 1, während der Zeitgeber aktiv ist (siehe `FB_DALIV2QueryColourStatus()` [▶ 296]). Nach Ablauf des Zeitgebers, werden die letzte Farbdarstellung, der letzte Farbwert und das letzte Lampenleistungsniveau direkt wieder gespeichert.

Während der Zeitgeberperiode führt das DALI-Vorschaltgerät ein Kalibrierungsverfahren durch, um die x-Koordinate, die y-Koordinate und den TY-Wert aller unterstützten Primärfarben zu messen. Während des laufenden Kalibrierungsverfahrens reagiert das DALI-Vorschaltgerät auf keinen Befehl, außer `TERMINATE` [▶ 168], `QUERY COLOUR STATUS` [▶ 296] und `START AUTO CALIBRATION`. Außerdem wird zu Beginn der Kalibrierung in der Variablen `COLOUR STATUS` [▶ 385] das Bit 3 auf 0 gesetzt. Der Befehl `TERMINATE` [▶ 168] beendet das Verfahren und hält den Zeitgeber an.

Wenn die Kalibrierung erfolgreich war, wird in `COLOUR STATUS` [▶ 385] Bit 3 auf 1 gesetzt und der Zeitgeber angehalten. Wenn die Kalibrierung nicht erfolgreich war, so werden die letzten erfolgreichen Kalibrierungsdaten wiederhergestellt wenn das DALI-Vorschaltgerät hierzu in der Lage ist. Anschließend wird Bit 3 von `COLOUR STATUS` auf 1 gesetzt. Die Fähigkeit, die letzten erfolgreichen Kalibrierungsdaten wiederherzustellen, ist ein Merkmal des Betriebsgerätes (siehe Befehl `QUERY GEAR FEATURES/STATUS` [▶ 301]).

Aufgrund der Tatsache, dass die Kalibrierung länger als 15 min dauern kann, sollte der Status der automatischen Kalibrierung mit dem Befehl `QUERY COLOUR STATUS` [▶ 296] periodisch überprüft und der Zeitgeber der Kalibrierung mit dem Befehl `START AUTO CALIBRATION` neu gestartet werden (wenn notwendig).

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur Tc
- Primär(farbe) N
- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl `Enable Device Type 8` vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl `Enable Device Type 8` allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

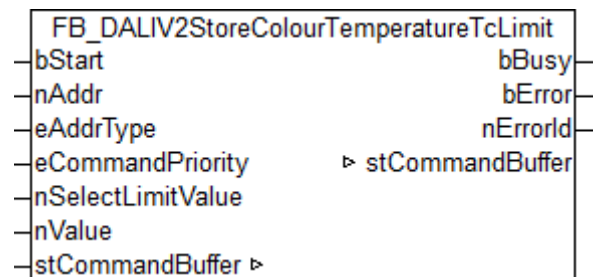
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.256 FB_DALIV2StoreColourTemperatureTcLimit



Speichert den Wert in die Variable [COLOUR TEMPERATURE Tc COOLEST](#) [▶ 380], [COLOUR TEMPERATURE Tc WARMEST](#) [▶ 380], [COLOUR TEMPERATURE Tc PHYSICAL COOLEST](#) [▶ 380] oder [COLOUR TEMPERATURE Tc PHYSICAL WARMEST](#) [▶ 380] des DALI-Vorschaltgerätes. Die Werte können mit dem Funktionsblock [FB_DALIV2QueryColourValue\(\)](#) [▶ 298] ausgelesen werden. Der Eingang *nSelectLimitValue* definiert den Grenzwert der neu gesetzt werden soll:

Wert	Grenzwert	Beschreibung
0	COLOUR TEMPERATURE Tc COOLEST	niedrigster möglicher Wert, aber immer gleich oder wärmer als der niedrigste mögliche physikalische Wert.
1	COLOUR TEMPERATURE Tc WARMEST	höchster möglicher Wert, aber immer gleich oder kälter als der höchste mögliche physikalische Wert.
2	COLOUR TEMPERATURE Tc PHYSICAL COOLEST	niedrigster möglicher physikalischer Wert.

Wert	Grenzwert	Beschreibung
3	COLOUR TEMPERATURE Tc PHYSICAL WARMEST	höchster möglicher physikalischer Wert.

Für die Umrechnung von oder nach Kelvin stehen die Funktionen [KELVIN TO MIREK](#) [▶ 326] und [MIREK TO KELVIN](#) [▶ 326] zur Verfügung.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register), DTR1 und DTR2 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- Farbtemperatur Tc



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nSelectLimitValue : BYTE;
nValue      : UINT;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nSelectLimitValue: Gibt den Grenzwert vor, der neu gesetzt werden soll (siehe Tabelle oben).

nValue: Der Wert, der in die ausgewählte Variable geschrieben wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

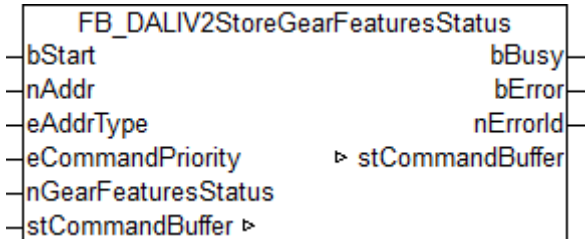
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.257 FB_DALIV2StoreGearFeaturesStatus



Speichert den Wert in die Variable GEAR FEATURES/STATUS [▶ 385] des DALI-Vorschaltgerätes. Mit dem Baustein FB_DALIV2QueryGearFeaturesStatus() [▶ 301] können die Werte ausgelesen werden.

Falls Bit 0 auf 1 gesetzt wird, müssen alle Befehle zur Steuerung der Lampenleistung, mit Ausnahme von ENABLE DAPC SEQUENCE [▶ 124], automatisch einen Farbübergang auslösen.

Bit	Beschreibung
0	automatische Aktivierung.
1 - 7	reserviert.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register) verändert.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten
- Farbtemperatur Tc
- Primär(farbe) N
- RGBWAF



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/ Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```

bStart          : BOOL;
nAddr           : BYTE;
eAddrType       : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nGearFeaturesStatus : BYTE;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nGearFeaturesStatus: Wert, der in die Variable GEAR FEATURES/STATUS geschrieben wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

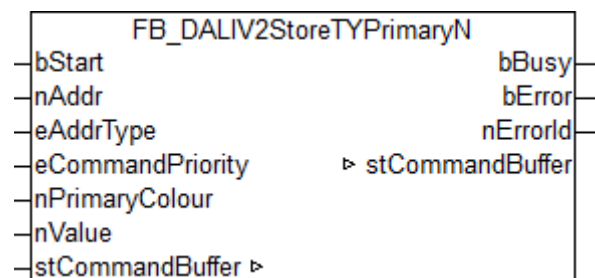
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.258 FB_DALIV2StoreTYPrimaryN



Speichert den Wert in die Variable [TY PRIMARY N](#) [▶ 380] des DALI-Vorschaltgerätes.

Der Wert wird in Einheiten von 0,5 lm ausgedrückt, was einen möglichen Bereich von $TY_{min} = 0$ lm bis $TY_{max} = 32767$ lm ergibt. Ein Wert von 65535 (MASK) bedeutet 'unbekannt'. Der Parameter *nPrimaryColour* gibt die Primärfarbe an und muss im Bereich von 0 bis 5 liegen, abhängig von der verfügbaren Anzahl von Primärfarben. Für jeden anderen Wert wird der Befehl ignoriert.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR (Data Transfer Register), DTR1 und DTR2 verändert.

Der Baustein unterstützt die Farbdarstellungen:

- Primär(farbe) N



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken `TcDALIV2` und `TcDALIV2AppExtCmds` jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nPrimaryColour : BYTE;
nValue      : UINT;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nPrimaryColour: Primärfarbe (0 - 5).

nValue: Der Wert, der in die Variable TY PRIMARY N geschrieben wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

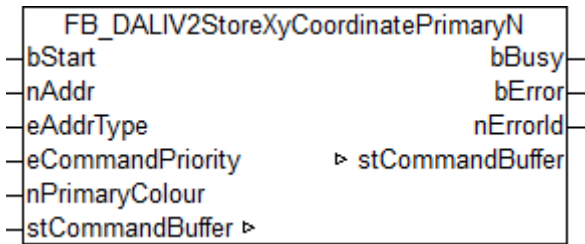
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.259 FB_DALIV2StoreXyCoordinatePrimaryN



Kopiert den Wert aus den Variablen TEMPORARY x-COORDINATE [▶ 380] und TEMPORARY y-COORDINATE [▶ 380] in die Variablen x-COORDINATE PRIMARY N [▶ 380] und y-COORDINATE PRIMARY N [▶ 380].

Der Parameter *nPrimaryColour* gibt die Primärfarbe an und muss im Bereich von 0 bis 5 liegen, abhängig von der verfügbaren Anzahl von Primärfarben. Für jeden anderen Wert wird der Befehl ignoriert.

Dieser Befehl ist dazu vorgesehen, die zur Primärfarbe dazugehörigen aktuellen xy-Koordinaten zu speichern. xy-Koordinaten außerhalb des Farbenraum-Chromatizitäts-Diagramms sind nicht sinnvoll und sollten deshalb vermieden werden.

Durch den Funktionsblock werden bei allen DALI-Vorschaltgeräten der DALI-Linie die DTR2 (Data Transfer Register) verändert.

Der Baustein unterstützt die Farbdarstellungen:

- Primär(farbe) N



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock FB_DALIV2EnableDeviceType() [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nPrimaryColour : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nPrimaryColour: Primärfarbe (0 - 5).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

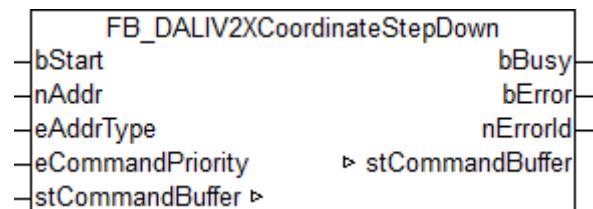
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.260 FB_DALIV2XCoordinateStepDown



Die Variable x-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung reduziert.

Falls der neue Farbwert nicht mit einer Farbe übereinstimmt, die vom DALI-Vorschaltgerät erreicht werden kann, muss dies durch das Bit 0 von [COLOUR STATUS](#) [▶ 385] (xy-coordinate-Farbpunkt liegt außerhalb des gültigen Bereichs), angezeigt werden. Der Befehl wird nur dann ausgeführt, wenn das Bit 4 vom [COLOUR STATUS](#) [▶ 385] (Farbdarstellung xy-coordinate aktiv) gesetzt ist.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

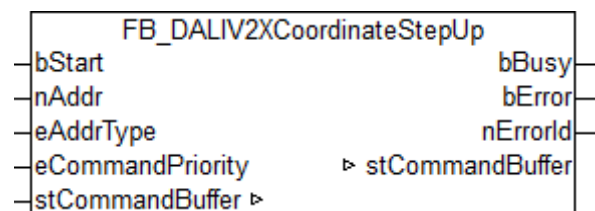
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [▶ 95] (KL6811) oder [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.261 FB_DALIV2XCoordinateStepUp



Die Variable x-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung erhöht.

Falls der neue Farbwert nicht mit einer Farbe übereinstimmt, die vom DALI-Vorschaltgerät erreicht werden kann, muss dies durch das Bit 0 von [COLOUR STATUS](#) [▶ 385] (xy-coordinate-Farbpunkt liegt außerhalb des gültigen Bereichs), angezeigt werden. Der Befehl wird nur dann ausgeführt, wenn das Bit 4 vom [COLOUR STATUS](#) [▶ 385] (Farbdarstellung xy-coordinate aktiv) gesetzt ist.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

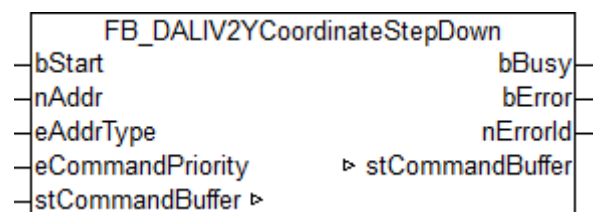
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.262 FB_DALIV2YCoordinateStepDown



Die Variable y-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung reduziert.

Falls der neue Farbwert nicht mit einer Farbe übereinstimmt, die vom DALI-Vorschaltgerät erreicht werden kann, muss dies durch das Bit 0 von COLOUR STATUS [▶ 385] (xy-coordinate-Farbpunkt liegt außerhalb des gültigen Bereichs) angezeigt werden. Der Befehl wird nur dann ausgeführt, wenn das Bit 4 vom COLOUR STATUS [▶ 385] (Farbdarstellung xy-coordinate aktiv) gesetzt ist.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

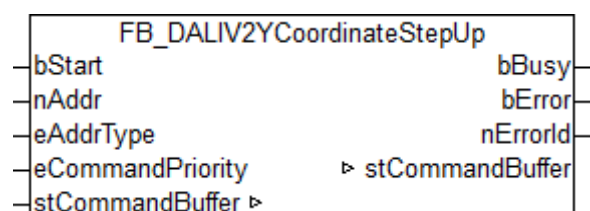
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.263 FB_DALIV2YCoordinateStepUp



Die Variable y-COORDINATE wird um 256 Schritte (256 / 65536) ohne Überblendung erhöht.

Falls der neue Farbwert nicht mit einer Farbe übereinstimmt, die vom DALI-Vorschaltgerät erreicht werden kann, muss dies durch das Bit 0 von [COLOUR STATUS \[► 385\]](#) (xy-coordinate-Farbpunkt liegt außerhalb des gültigen Bereichs) angezeigt werden. Der Befehl wird nur dann ausgeführt, wenn das Bit 4 vom [COLOUR STATUS \[► 385\]](#) (Farbdarstellung xy-coordinate aktiv) gesetzt ist.

Der Baustein unterstützt die Farbdarstellungen:

- xy-Koordinaten



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type 8* vorangeht, welcher mit dem Funktionsblock [FB_DALIV2EnableDeviceType\(\)](#) [\[► 356\]](#) gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type 8* allen anwendungsbezogenen Erweiterungsbefehlen für Lampen zur Farb-/Farbtemperatursteuerung mit DALI-Schnittstelle intern automatisch vorangestellt.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[► 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität \[► 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

VAR_IN_OUT

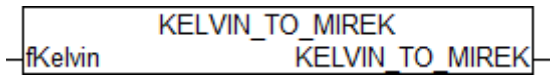
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\) \[► 95\]](#) (KL6811) oder [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.264 KELVIN_TO_MIREK



Umrechnung der Farbtemperatur von Kelvin in Mirek.

Mirek ist die Einheit, die bei den meisten DALI-Befehlen verwendet wird. Der Rückgabewert ist begrenzt und liegt im Bereich von 0 ... 65535 (siehe Tabelle).

Mirek = 1.000.000 / (Farbtemperatur in Kelvin).

Kelvin	Mirek
0	65535
15	65535
16	62500
1000	1000
10000	100
1000000	1
1000001	0

VAR_INPUT

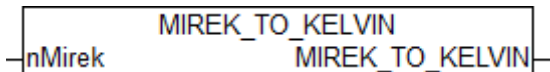
fKelvin : LREAL;

fKelvin: Farbtemperatur in Kelvin.

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.265 MIREK_TO_KELVIN



Umrechnung der Farbtemperatur von Mirek in Kelvin.

Mirek ist die Einheit, die bei den meisten DALI-Befehlen verwendet wird. Der Rückgabewert ist begrenzt und liegt im Bereich von ca. 15,259 ...1000001 (siehe Tabelle).

Mirek = 1.000.000 / (Farbtemperatur in Kelvin).

Mirek	Kelvin
0	1000001
1	1000000
100	10000
1000	1000
10000	100
65534	15,259
65535	15,259

VAR_INPUT

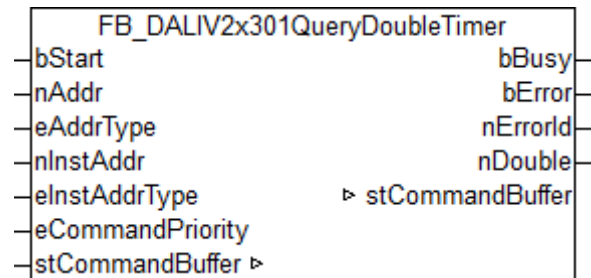
nMirek : UINT;

nMirek: Farbtemperatur in Mirek.

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2244	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.10.0

5.1.266 FB_DALIV2x301QueryDoubleTimer



Fragt den Wert des DOUBLE TIMER ab.

Die Einheit ist 20 ms. Der maximal zulässige Wert ist 2000 ms. Durch den Wert 0 wird der Timer deaktiviert. Der minimal zulässige Wert kann durch `FB_DALIV2x301QueryDoubleTimerMin()` [▶ 328] abgefragt werden.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
  
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nDouble    : BYTE;
  
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nDouble: Wert des DOUBLE TIMER [20 ms].

VAR_IN_OUT

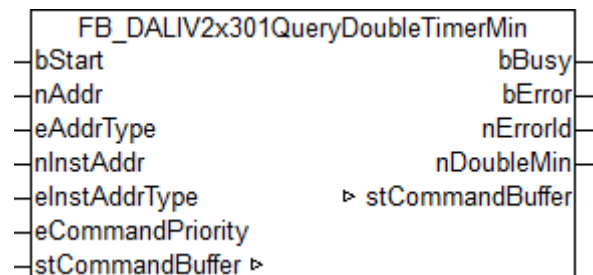
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.267 FB_DALIV2x301QueryDoubleTimerMin



Fragt den minimalen Wert des DOUBLE TIMER ab.

Die Einheit ist 20 ms.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nDoubleMin : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[► 391\]](#).

nDoubleMin: minimale Wert des DOUBLE TIMER [20 ms].

VAR_IN_OUT

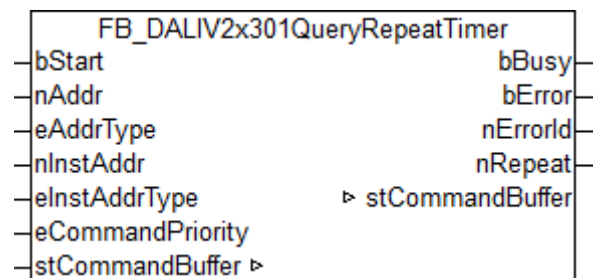
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\) \[► 104\]](#) (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.268 FB_DALIV2x301QueryRepeatTimer



Fragt den Wert des REPEAT TIMER ab.

Die Einheit ist 20 ms. Der zulässige Wertebereich ist von 100 ms bis 2000 ms; also von 5 bis 100.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[► 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz \[► 388\]](#), innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität \[► 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nRepeat : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nRepeat: Wert des REPEAT TIMER [20 ms].

VAR_IN_OUT

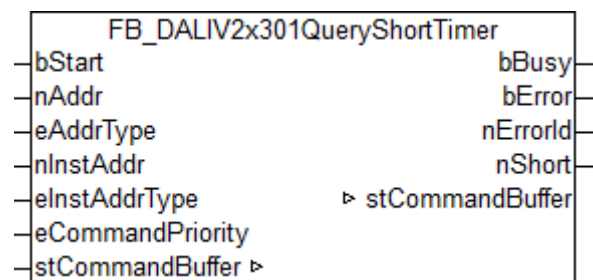
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.269 FB_DALIV2x301QueryShortTimer



Fragt den Wert des SHORT TIMER ab.

Die Einheit ist 20 ms. Der maximal zulässige Wert ist 5100 ms. Der minimal zulässige Wert kann durch [FB_DALIV2x301QueryShortTimerMin\(\)](#) [▶ 331] abgefragt werden.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nShort     : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

nShort: Wert des SHORT TIMER [20 ms].

VAR_IN_OUT

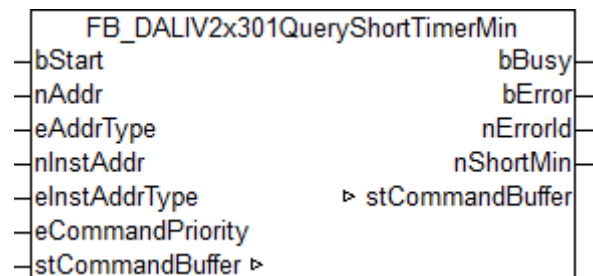
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.270 FB_DALIV2x301QueryShortTimerMin



Fragt den minimalen Wert des SHORT TIMER ab.

Die Einheit ist 20 ms.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nShortMin  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

nShortMin: minimale Wert des SHORT TIMER [20 ms].

VAR_IN_OUT

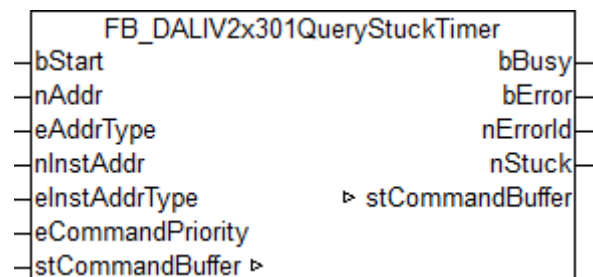
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.271 FB_DALIV2x301QueryStuckTimer



Fragt den Wert des STUCK TIMER ab.

Die Einheit ist 1 s. Der zulässige Wertebereich ist von 5 s bis 255 s.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [[▶ 388](#)], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nStuck     : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

nStuck: Wert des STUCK TIMER [s].

VAR_IN_OUT

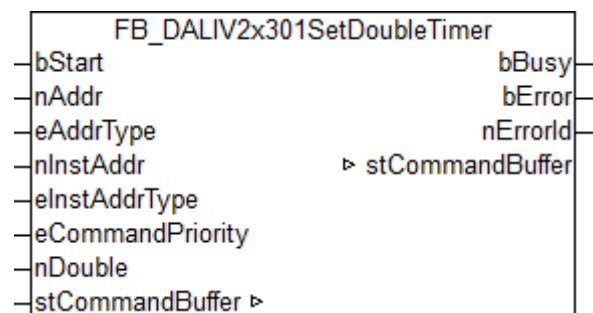
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein `FB_KL6821Communication()` [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.272 FB_DALIV2x301SetDoubleTimer



Setzt den Wert des DOUBLE TIMER.

Die Einheit ist 20 ms. Der maximal zulässige Wert ist 2000 ms. Durch den Wert 0 wird der Timer deaktiviert. Der minimal zulässige Wert kann durch `FB_DALIV2x301QueryDoubleTimerMin()` [[▶ 328](#)] abgefragt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
```

```
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDouble : BYTE := 0;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse, Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz, innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDouble: Wert des DOUBLE TIMER [20 ms].

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

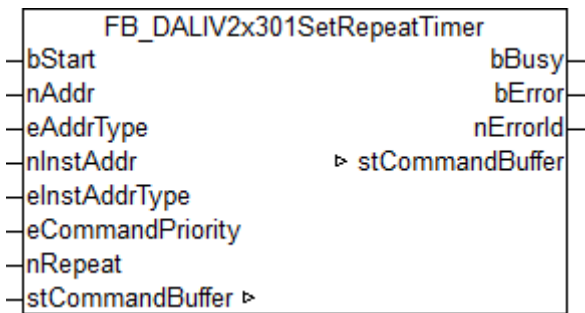
Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

Sehen Sie dazu auch

- 📖 [E_DALIV2AddrType](#) [▶ 387]
- 📖 [E_DALIV2InstAddrType](#) [▶ 388]
- 📖 [E_DALIV2CommandPriority](#) [▶ 387]

5.1.273 FB_DALIV2x301SetRepeatTimer



Setzt den Wert des REPEAT TIMER.

Die Einheit ist 20 ms. Der zulässige Wertebereich ist von 100 ms bis 2000 ms; also von 5 bis 100.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nRepeat     : BYTE := 8;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nRepeat: Wert des REPEAT TIMER [20 ms].

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

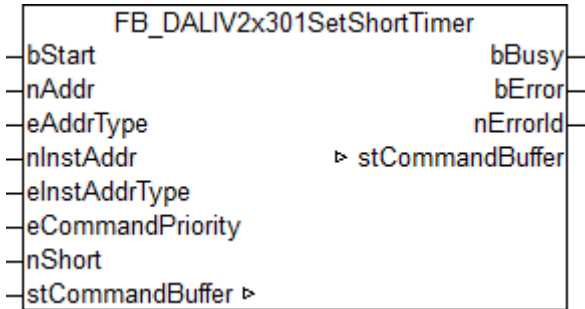
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.274 FB_DALIV2x301SetShortTimer



Setzt den Wert des SHORT TIMER.

Die Einheit ist 20 ms. Der maximal zulässige Wert ist 5100 ms. Der minimal zulässige Wert kann durch **FB_DALIV2x301QueryShortTimerMin()** [▶ 331] abgefragt werden.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nShort      : BYTE := 25;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nShort: Wert des SHORT TIMER [20 ms].

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
    
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

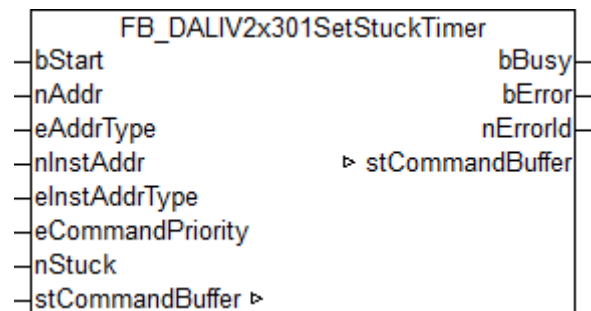
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.275 FB_DALIV2x301SetStuckTimer



Setzt den Wert des STUCK TIMER.

Die Einheit ist 1 s. Der zulässige Wertebereich ist von 5 s bis 255 s.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nStuck : BYTE := 20;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nStuck: Wert des STUCK TIMER [s].

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

VAR_IN_OUT

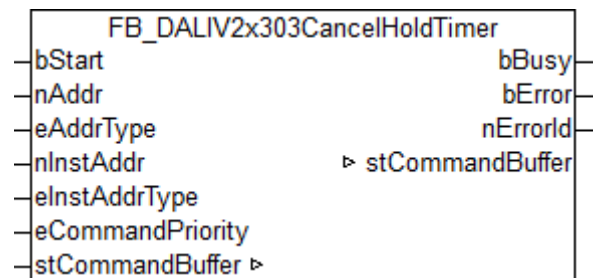
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.276 FB_DALIV2x303CancelHoldTimer



Beendet vorzeitig den HOLD TIMER.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[▶ 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz \[▶ 388\]](#), innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes \[▶ 391\]](#).

VAR_IN_OUT

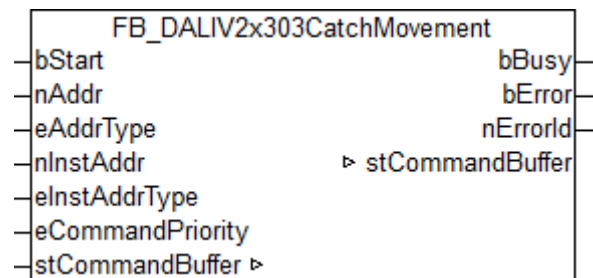
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\) \[▶ 104\]](#) (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.277 FB_DALIV2x303CatchMovement



Nach Aufruf dieses Befehls wird nur einmalig ein Ereignis versendet, falls eine Bewegung erkannt wird.

Für diese Funktion muss der Event Filter entsprechend konfiguriert werden.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse \[▶ 387\]](#), Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz \[▶ 388\]](#), innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität \[▶ 387\]](#) (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

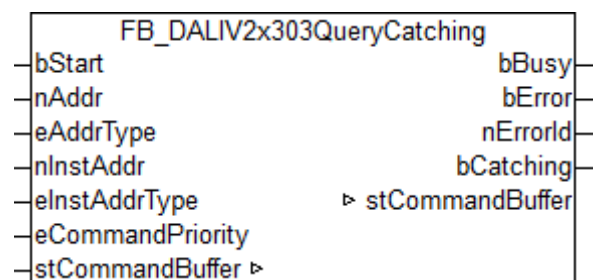
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.278 FB_DALIV2x303QueryCatching



Fragt ab, ob auf die Erkennung einer Bewegung gewartet wird. Diese Funktion kann durch [FB_DALIV2x303CatchMovement\(\)](#) [▶ 339] aktiviert werden.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
bCatching : BOOL;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

bCatching: Catching aktiv.

VAR_IN_OUT

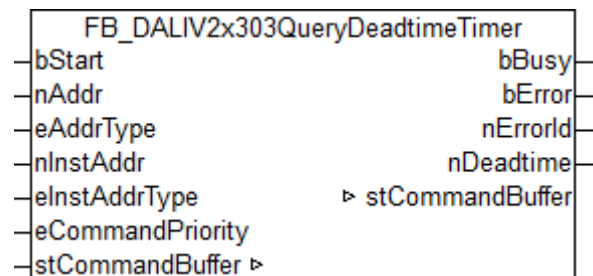
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.279 FB_DALIV2x303QueryDeadtimeTimer



Fragt den Wert des DEADTIME TIMER ab.

Die Einheit ist 50 ms. Der zulässige Wertebereich ist von 0 s bis 12,75 s.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nDeadtime  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nDeadtime: Wert des DEADTIME TIMER [50 ms].

VAR_IN_OUT

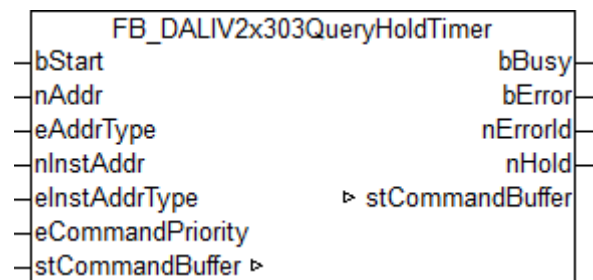
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.280 FB_DALIV2x303QueryHoldTimer



Fragt den Wert des HOLD TIMER ab.

Die Einheit ist 10 s. Der zulässige Wertebereich ist von 1 s bis 42,3 min.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [► 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nHold      : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [► 391].

nHold: Wert des HOLD TIMER [10 s].

VAR_IN_OUT

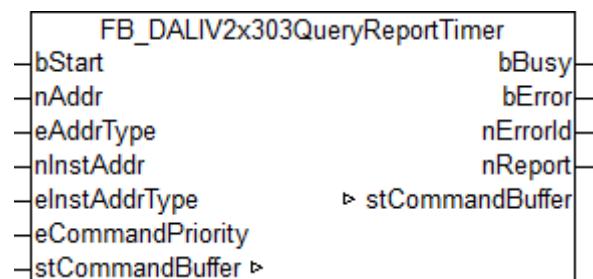
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.281 FB_DALIV2x303QueryReportTimer



Fragt den Wert des REPORT TIMER ab.

Die Einheit ist 1 s. Der zulässige Wertebereich ist von 1 s bis 4 min 15 s.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nReport    : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nReport: Wert des REPORT TIMER [s].

VAR_IN_OUT

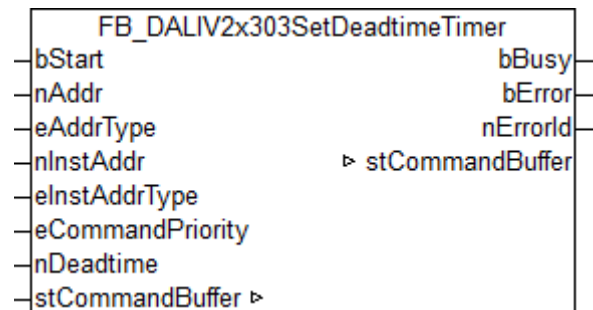
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.282 FB_DALIV2x303SetDeadtimeTimer



Setzt den Wert des DEADTIME TIMER.

Die Einheit ist 50 ms. Der zulässige Wertebereich ist von 0 s bis 12,75 s.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
```

```
nInstAddr      : BYTE := 0;
eInstAddrType  : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDeadtime     : BYTE := 2;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDeadtime: Wert des DEADTIME TIMER [50 ms].

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

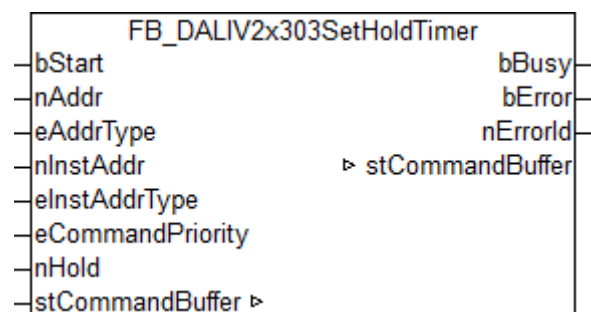
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.283 FB_DALIV2x303SetHoldTimer



Setzt den Wert des HOLD TIMER.

Wird der Wert 0 übergeben, so wird HOLD TIME auf 1 s gesetzt. Die Einheit ist 10 s. Der zulässige Wertebereich ist von 1 s bis 42,3 min.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nHold       : BYTE := 90;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nHold: Wert des HOLD TIMER [10 s]. Der Wert 0 entspricht 1 s.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.284 FB_DALIV2x303SetReportTimer



Setzt den Wert des REPORT TIMER.

Die Einheit ist 1 s. Der zulässige Wertebereich ist von 1 s bis 4 min 15 s.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nReport     : BYTE := 20;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nReport: Wert des REPORT TIMER [s].

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

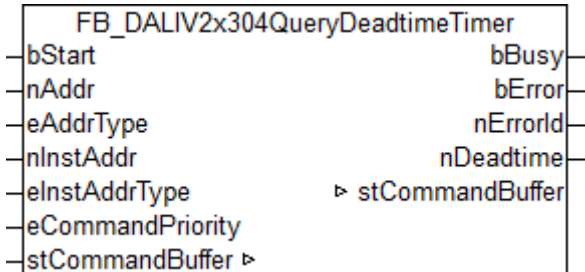
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.285 FB_DALIV2x304QueryDeadtimeTimer



Fragt den Wert des DEADTIME TIMER ab.

Die Einheit ist 50 ms. Der zulässige Wertebereich ist von 0 s bis 12,75 s.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nDeadtime  : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nDeadtime: Wert des DEADTIME TIMER [50 ms].

VAR_IN_OUT

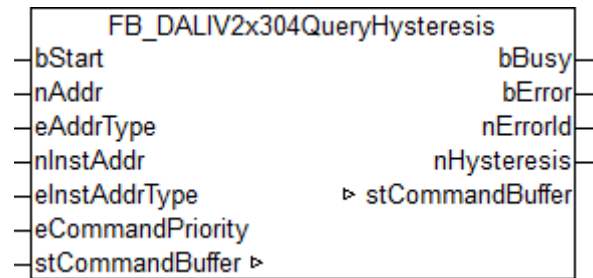
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```


stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein
 FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.286 FB_DALIV2x304QueryHysteresis



Fragt den Wert für die Hysterese ab.

Die Hysterese wird in % angegeben und liegt im Wertebereich 0% bis 25 %.

VAR_INPUT

```

bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
    
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```

bBusy       : BOOL;
bError      : BOOL;
nErrorId    : UDINT;
nHysteresis : BYTE;
    
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in nErrorId enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nHysteresis: Hysterese in %.

VAR_IN_OUT

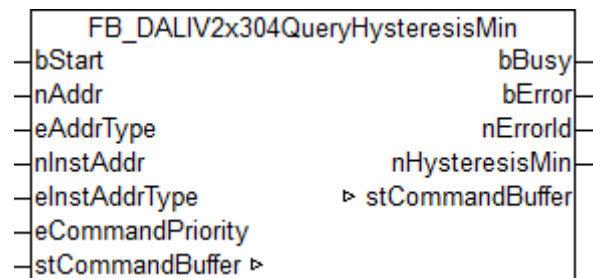
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.287 FB_DALIV2x304QueryHysteresisMin



Fragt den Wert für die minimal mögliche Hysterese ab.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nHysteresisMin: BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

nHysteresisMin: minimal mögliche Hysterese.

VAR_IN_OUT

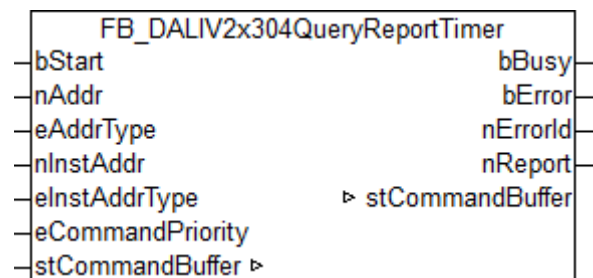
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.288 FB_DALIV2x304QueryReportTimer



Fragt den Wert des REPORT TIMER ab.

Die Einheit ist 1 s. Der zulässige Wertebereich ist von 1 s bis 4 min 15 s.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
nReport : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [► 391].

nReport: Wert des REPORT TIMER [s].

VAR_IN_OUT

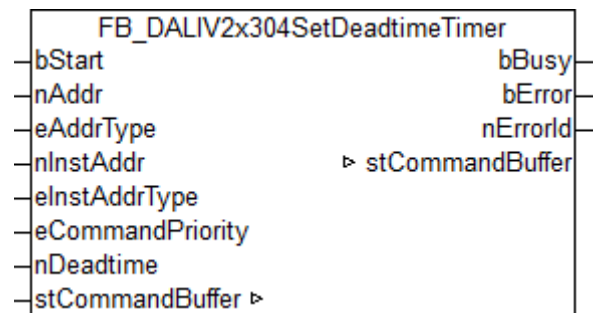
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.289 FB_DALIV2x304SetDeadtimeTimer



Setzt den Wert des DEADTIME TIMER.

Die Einheit ist 50 ms. Der zulässige Wertebereich ist von 0 s bis 12,75 s.

VAR_INPUT

```
bStart : BOOL;
nAddr : BYTE;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDeadtime : BYTE := 30;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [► 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [► 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [► 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDeadtime: Wert des DEADTIME TIMER [50 ms].

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

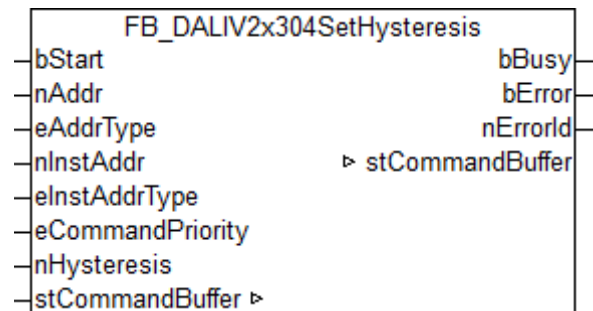
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.290 FB_DALIV2x304SetHysteresis



Setzt den Wert für die Hysterese.

Die Hysterese wird in % angegeben und liegt im Wertebereich 0 % bis 25 %.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nHysteresis : BYTE := 5;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nHysteresis: Hysterese in %.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [[▶ 391](#)].

VAR_IN_OUT

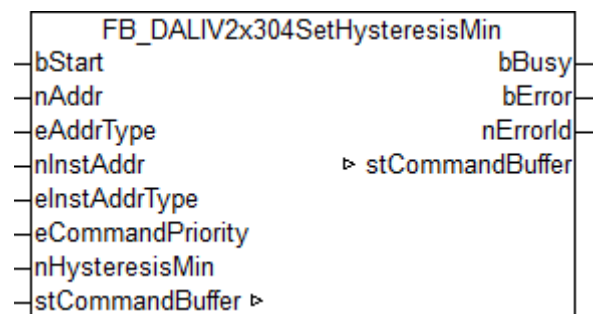
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.291 FB_DALIV2x304SetHysteresisMin



Setzt den Wert für die minimal mögliche Hysterese.

Da die Hysterese in Prozent angegeben wird, ist die absolute Größe der Hysterese abhängig vom aktuellen Messwert. Bei sehr kleinen Messwerten ist somit auch die Hysterese sehr klein. Dieses führt dazu, dass unnötige Ereignisse versendet werden. Aus diesem Grund kann eine minimal mögliche Hysterese eingestellt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nHysteresisMin : BYTE := 0;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: Adresse der Instanz [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nHysteresisMin: minimal mögliche Hysterese.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

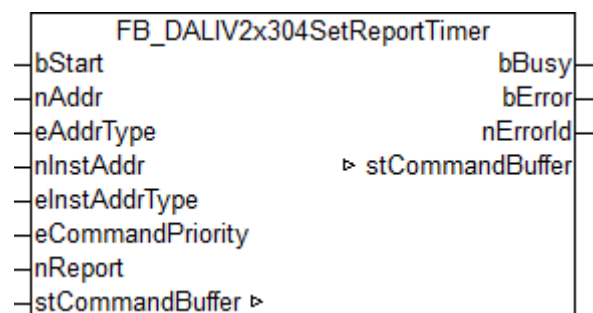
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.292 FB_DALIV2x304SetReportTimer



Setzt den Wert des REPORT TIMER.

Die Einheit ist 1 s. Der zulässige Wertebereich ist von 1 s bis 4 min 15 s.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
nInstAddr   : BYTE := 0;
```



```
eInstAddrType : E_DALIV2InstAddrType := eDALIV2InstAddrTypeNumber;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nReport : BYTE := 30;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

nInstAddr: [Adresse der Instanz](#) [▶ 388], innerhalb des DALI-Steuergerätes.

eInstAddrType: Definiert die Zugriffsart auf die gewünschte Instanz innerhalb des DALI-Steuergerätes.

eCommandPriority: [Priorität](#) [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nReport: Wert des REPORT TIMER [s].

VAR_OUTPUT

```
bBusy : BOOL;
bError : BOOL;
nErrorId : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [▶ 391].

VAR_IN_OUT

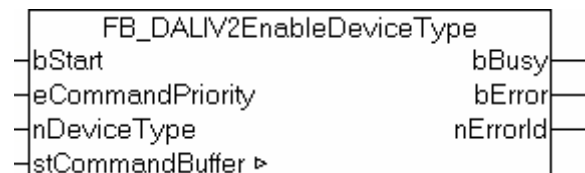
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.293 FB_DALIV2EnableDeviceType



Dieser Befehl muss jedes Mal vor einem anwendungsbezogenen Erweiterungsbefehl gesendet werden. Hierdurch reagieren nur die Vorschaltgeräte, die auch zu dem entsprechenden Gerätetyp ([DEVICE TYPE](#) [▶ 373]) gehören. Für den Gerätetyp 0 braucht der Befehl nicht angewendet werden. Folgende Gerätetypen sind laut Norm IEC 62386 definiert:

Wert	Beschreibung
0	Standardgerät

Wert	Beschreibung
1	Gerät für <u>Notbeleuchtung</u> [▶ 32].
2	Gerät für <u>Entladungslampen</u> [▶ 36].
3	Gerät für Niedervolt-Halogenlampen.
4	Gerät zum Dimmen von Glühlampen.
5	Gerät zur Umwandlung von digitalen Signalen in Gleichspannungssignale.
6	Gerät für <u>Licht emittierende Dioden (LED)</u> [▶ 36].
7	Gerät für Schalfunktionen.
8	Gerät zur <u>Farb-/Farbtemperatursteuerung</u> [▶ 34].
9	Sequenzner.

VAR_INPUT

```
bStart          : BOOL;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDeviceType     : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDeviceType: Kennung für den Gerätetyp (siehe Tabelle oben).

VAR_OUTPUT

```
bBusy          : BOOL;
bError         : BOOL;
nErrorId       : UDINT;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

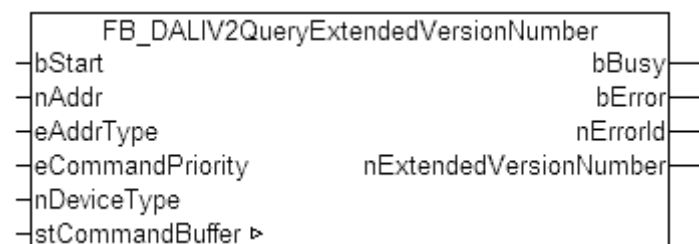
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.294 FB_DALIV2QueryExtendedVersionNumber



Die Variable EXTENDED VERSION NUMBER [▶ 378] (erweiterte Versionsnummer) wird aus dem Vorschaltgerät ausgelesen.



Dieser Befehl gehört zu den anwendungsbezogenen Erweiterungsbefehlen. Diese funktionieren nur dann, wenn ihnen der Befehl *Enable Device Type* vorangeht, welcher mit dem Funktionsblock `FB_DALIV2EnableDeviceType()` [▶ 356] gesendet werden kann. Ab Version 2.6.0 der Bibliotheken *TcDALIV2* und *TcDALIV2AppExtCmds* jedoch wird der Befehl *Enable Device Type* intern automatisch vorangestellt. Welcher *Device Type* aktiviert werden soll, kann am Eingang `nDeviceType` [▶ 356] eingestellt werden.

VAR_INPUT

```
bStart      : BOOL;
nAddr       : BYTE;
eAddrType   : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
nDeviceType : BYTE;
```

bStart: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

nDeviceType: Kennung für den Gerätetyp (siehe Tabelle unter `FB_DALIV2EnableDeviceType()` [▶ 356]).

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
nExtendedVersionNumber : BYTE;
```

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in `nErrorId` enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

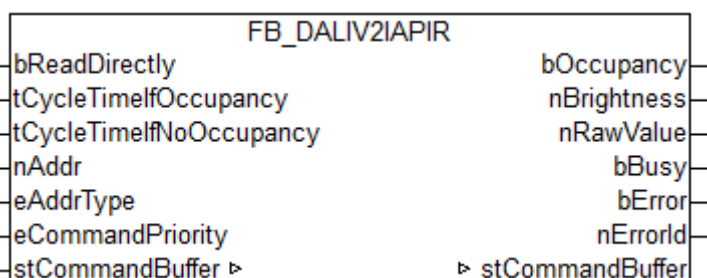
nExtendedVersionNumber: erweiterte Versionsnummer des Vorschaltgerätes.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen `FB_DALIV2Communication()` [▶ 95] (KL6811) oder `FB_KL6821Communication()` [▶ 104] (KL6821).

5.1.295 FB_DALIV2IAPIR



Dieser Baustein liest zyklisch den Zustand eines IA PIR-Sensors aus und skaliert aus dem empfangenen Wert die gemessene Helligkeit und die erkannte Präsenz.

Es können zwei verschiedene Zykluszeiten vorgegeben werden. Eine Zykluszeit, die benutzt wird wenn keine Präsenz erkannt wurde, und eine die bei aktiver Präsenz zum Einsatz kommt. Hierdurch besteht die Möglichkeit, die Zugriffe auf den DALI-Bus zu minimieren. Ist Präsenz erkannt, so kann in der Regel eine langsamere Zykluszeit (z.B. 20 s) gewählt werden, da die Lichtregelung und das Ausschalten der Beleuchtung nicht zeitkritisch sind. Ist keine Präsenz vorhanden, so sollte eine kleinere Zykluszeit (z.B. 2s) gewählt werden. Dadurch erfolgt das Einschalten der Beleuchtung beim Betreten des Raums mit einer möglichst geringen Reaktionszeit.

Weitere Informationen und eine Beschreibung der elektrischen und physikalischen Eigenschaften finden Sie in der Produktbeschreibung des IA PIR-Sensors.

VAR_INPUT

```
bReadDirectly      : BOOL := FALSE;
tCycleTimeIfOccupancy : TIME := t#20s;
tCycleTimeIfNoOccupancy : TIME := t#2s;
nAddr              : BYTE := 0;
eAddrType          : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority    : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bReadDirectly: Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

tCycleTimeIfOccupancy: Zykluszeit bei aktiver Präsenz.

tCycleTimeIfNoOccupancy: Zykluszeit bei nicht aktiver Präsenz.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: [Kurzadresse](#) [[▶ 387](#)], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: [Priorität](#) [[▶ 387](#)] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bOccupancy        : BOOL;
nBrightness        : INT;
nRawValue          : BYTE;
bBusy              : BOOL;
bError             : BOOL;
nErrorId           : UDINT;
```

bOccupancy: Der Ausgang ist gesetzt, wenn der Sensor Präsenz erkennt.

nBrightness: Gemessene Helligkeit in Lux.

nRawValue: Der ausgelesene Wert vom Sensor vor der Umrechnung.

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe [Fehlercodes](#) [[▶ 391](#)].

VAR_IN_OUT

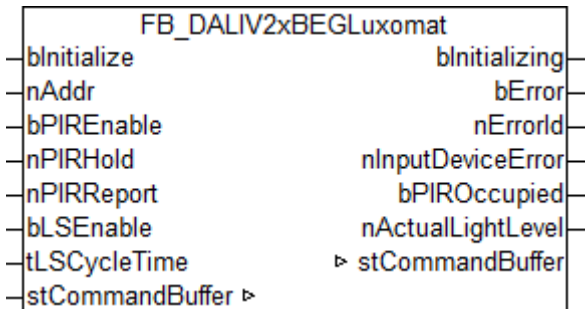
```
stCommandBuffer    : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen [FB_DALIV2Communication\(\)](#) [[▶ 95](#)] (KL6811) oder [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2234	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.9.0

5.1.296 FB_DALIV2xBEGLuxomat



Dieser Baustein dient als Beispiel und ist aus diesem Grund nicht in der Bibliothek enthalten. Der Baustein kann als Export-Datei heruntergeladen und in das gewünschte Projekt importiert werden. Bei Bedarf können somit beliebige Anpassungen durchgeführt werden.

Dieser Baustein wertet die gemessene Helligkeit und die Präsenz des B.E.G. Luxomat DALI-Steuergerätes aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergeräte zu initialisieren.

Das DALI-Steuergeräte belegt eine Kurzadresse und enthält zwei Instanzen. Instanz 0 ist der Bewegungsmelder und entspricht der Norm IEC 62386-303. Der Helligkeitssensor ist in der Instanz 1 abgelegt und entspricht der Norm IEC 62386-304.

Eine erkannte Bewegung wird als Ereignis versendet, während die Helligkeit zyklisch ausgelesen wird. Jede Instanz kann einzeln bei Bedarf deaktiviert werden.

Download der TwinCAT 2 PLC Control Export-Datei: <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019296651.zip>

VAR_INPUT

```

blInitialize      : BOOL := FALSE;
nAddr             : BYTE;
(* Occupancy Sensor Parameters *)
bPIREnable       : BOOL := TRUE;
nPIRHold         : BYTE := 1; (* 10 s *)
nPIRReport       : BYTE := 30; (* 30 s *)
(* Light Sensor Parameters *)
blSEnable        : BOOL := TRUE;
tLSCycleTime     : TIME := t#1m;
    
```

blInitialize: Durch eine positive Flanke an diesem Eingang wird das DALI-Steuergerät initialisiert. Die Parameter werden in den DALI-Steuergerät geschrieben welcher über die Kurzadresse *nAddr* erreichbar sein muss. Während der Initialisierung ist der Ausgang *blinitializing* TRUE.

nAddr: Kurzadresse des DALI-Steuergerätes.

bPIREnable: Parameter: Ist dieser Eingang TRUE, so wird der Bewegungsmelder (Instanz 0) freigegeben. Durch ein FALSE wird der Präsenzmelder deaktiviert.

nPIRHold: Parameter: Setzt den Wert für den *Hold Timer*. Wird keine Bewegung erkannt, so wird erst nach Ablauf des *Hold Timer* der Status des Bewegungsmelders geändert. Die Einheit ist 10 s. Somit sind Zeiten bis 42 min 20 s (Wert 254) möglich. Der Wert 0 entspricht 1 s während der Wert 255 ignoriert wird.

nPIRReport: Parameter: Setzt den Wert für den *Report Timer*. Der Status des Bewegungsmelders wird nach Ablauf des *Report Timer* erneut gesendet, auch dann, wenn der Status sich nicht geändert hat. Die Einheit ist 1 s. Durch den Wert 0 wird der *Report Timer* deaktiviert. Somit sind Zeiten bis 4 min 15 s (Wert 255) möglich.

blSEnable: Parameter: Ist dieser Eingang TRUE, so wird der Helligkeitssensor (Instanz 1) freigegeben. Durch ein FALSE wird der Helligkeitssensor deaktiviert.

tLSCycleTime: Zykluszeit, in der der aktuelle Istwert des Helligkeitssensors ausgelesen wird.

Weitere Einzelheiten zu den Parametern sind aus der Norm IEC 62386 und der Dokumentation des Herstellers zu entnehmen.

VAR_OUTPUT

```
bInitializing      : BOOL;
bError            : BOOL;
nErrorId         : UDINT;
nInputDeviceError : BYTE;
(* Occupancy Sensor *)
bPIROccupied     : BOOL;
(* Light Sensor *)
nActualLightLevel : UINT;
```

bInitializing: Dieser Ausgang ist während der Initialisierung TRUE.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Siehe [Fehlercodes](#) [► 391].

nInputDeviceError: Vor der Initialisierung wird der Status des DALI-Steuergerätes (INPUT DEVICE ERROR) abgefragt. 0 bedeutet kein Fehler. Die einzelnen Fehlernummern sind herstellerspezifisch.

bPIROccupied: Dieser Ausgang zeigt den Status des Bewegungsmelders an.

nActualLightLevel: Dieser Ausgang zeigt den Status des Helligkeitssensors an.

VAR_IN_OUT

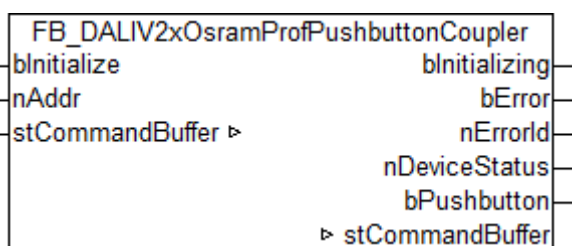
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [► 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.297 FB_DALIV2xOsramProfPushbuttonCoupler





Dieser Baustein dient als Beispiel und ist aus diesem Grund nicht in der Bibliothek enthalten. Der Baustein kann als Export-Datei heruntergeladen und in das gewünschte Projekt importiert werden. Bei Bedarf können somit beliebige Anpassungen durchgeführt werden.

Dieser Baustein wertet den Zustand des digitalen Eingangs des Osram DALI Professional Pushbutton Couplers aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.

Das DALI-Steuergeräte belegt eine Kurzadresse. Das Gerät entspricht nicht vollständig der Norm IEC 62386 und benutzt stattdessen ein firmenspezifisches Protokoll.

Download der TwinCAT 2 PLC Control Export-Datei: <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019298059.zip>

VAR_INPUT

```
bInitialize      : BOOL := FALSE;
nAddr           : BYTE;
```

bInitialize: Durch eine positiven Flanke an diesem Eingang wird das DALI-Steuergerät initialisiert. Die Parameter werden in den DALI-Steuergerät geschrieben welcher über die Kurzadressen *nPIRAddr* und *nLSAddr* erreichbar sein muss. Während der Initialisierung ist der Ausgang *bInitializing* TRUE.

nAddr: Kurzadresse des DALI-Steuergerätes.

Weitere Einzelheiten zu den Parametern sind aus der Dokumentation des Herstellers zu entnehmen.

VAR_OUTPUT

```
bInitializing    : BOOL;
bError           : BOOL;
nErrorId        : UDINT;
nDeviceStatus   : BYTE;
bPushbutton     : BOOL;
```

bInitializing: Dieser Ausgang ist während der Initialisierung TRUE.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Siehe [Fehlercodes](#) [[▶ 391](#)].

nDeviceStatus: Vor der Initialisierung wird der Status des DALI-Steuergerätes abgefragt und an diesem Ausgang ausgegeben.

bPushbutton: Dieser Ausgang zeigt den Istwert des digitalen Eingangs an.

VAR_IN_OUT

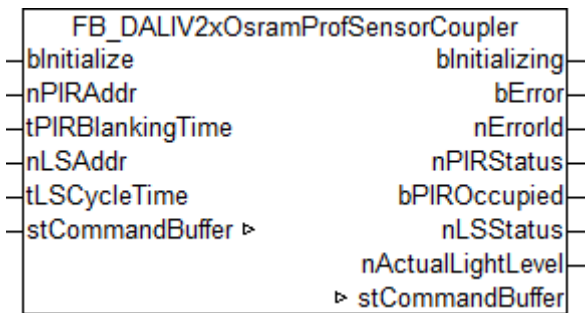
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.298 FB_DALIV2xOsramProfSensorCoupler



Dieser Baustein dient als Beispiel und ist aus diesem Grund nicht in der Bibliothek enthalten. Der Baustein kann als Export-Datei heruntergeladen und in das gewünschte Projekt importiert werden. Bei Bedarf können somit beliebige Anpassungen durchgeführt werden.

Dieser Baustein wertet die gemessene Helligkeit und die Präsenz des Osram DALI Professional Sensor Couplers aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.

Der Bewegungsmelder als auch der Helligkeitssensor belegen jeweils eine separate Kurzadresse. Somit belegt das DALI-Steuergerät zwei Kurzadressen. Das Gerät entspricht nicht vollständig der Norm IEC 62386 und benutzt stattdessen ein firmenspezifisches Protokoll.

Eine erkannte Bewegung wird als Ereignis versendet, während die Helligkeit zyklisch ausgelesen wird.

Download der TwinCAT 2 PLC Control Export-Datei: <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019299467.zip>

VAR_INPUT

```
blInitialize      : BOOL := FALSE;
(* Occupancy Sensor Parameters *)
nPIRAddr         : BYTE;
tPIRBlankingTime : TIME := T#1M;
(* Light Sensor Parameters *)
nLSAddr         : BYTE;
tLSCycleTime    : TIME := T#1M;
```

blInitialize: Durch eine positiven Flanke an diesem Eingang wird das DALI-Steuergerät initialisiert. Die Parameter werden in den DALI-Steuergerät geschrieben welcher über die Kurzadressen *nPIRAddr* und *nLSAddr* erreichbar sein muss. Während der Initialisierung ist der Ausgang *bInitializing* TRUE.

nPIRAddr: Kurzadresse des DALI-Steuergerätes für den Bewegungsmelder.

tPIRBlankingTime: Parameter: Wurde der Status des Bewegungsmelders gesendet, so wird für diese Zeit keine weitere Änderung übertragen. Die Einheit ist 1 s. Somit sind Zeiten bis 4 min 15 s (Wert 255) möglich.

nLSAddr: Kurzadresse des DALI-Steuergerätes für den Helligkeitssensor.

tLSCycleTime: Zykluszeit, in der der aktuelle Istwert des Helligkeitssensors ausgelesen wird.

Weitere Einzelheiten zu den Parametern sind aus der Dokumentation des Herstellers zu entnehmen.

VAR_OUTPUT

```
bInitializing    : BOOL;
bError           : BOOL;
nErrorId         : UDINT;
(* Occupancy Sensor *)
nPIRStatus       : BYTE;
bPIROccupied     : BOOL;
(* Light Level *)
nLSStatus        : BYTE;
nActualLightLevel : WORD;
```

bInitializing: Dieser Ausgang ist während der Initialisierung TRUE.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Siehe [Fehlercodes](#) [[▶ 391](#)].

nPIRStatus: Vor der Initialisierung wird der Status des DALI-Steuergerätes für den Bewegungsmelder abgefragt und an diesem Ausgang ausgegeben.

bPIROccupied: Dieser Ausgang zeigt den Istwert des Bewegungsmelder an.

nLSStatus: Vor der Initialisierung wird der Status des DALI-Steuergerätes für den Helligkeitssensor abgefragt und an diesem Ausgang ausgegeben.

nActualLightLevel: Dieser Ausgang zeigt den Istwert des Helligkeitssensors an.

VAR_IN_OUT

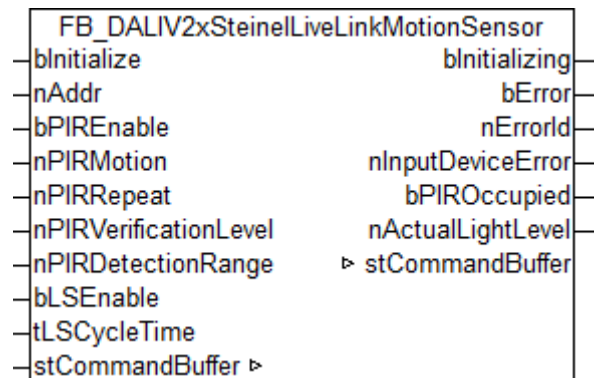
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein [FB_KL6821Communication\(\)](#) [[▶ 104](#)] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.299 FB_DALIV2xSteinelLiveLinkMotionSensor



i Dieser Baustein dient als Beispiel und ist aus diesem Grund nicht in der Bibliothek enthalten. Der Baustein kann als Export-Datei heruntergeladen und in das gewünschte Projekt importiert werden. Bei Bedarf können somit beliebige Anpassungen durchgeführt werden.

Dieser Baustein wertet die gemessene Helligkeit und die Präsenz des Steinel LiveLink DALI-Steuergerätes aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.

Das DALI-Steuergerät belegt eine Kurzadresse und enthält 2 Instanzen. In Instanz 1 ist der Bewegungsmelder und in Instanz 0 der Helligkeitssensor abgelegt. Das Gerät entspricht nicht vollständig der Norm IEC 62386 und benutzt stattdessen ein firmenspezifisches Protokoll.

Eine erkannte Bewegung wird als Ereignis versendet, während die Helligkeit zyklisch ausgelesen wird. Jede Instanz kann einzeln bei Bedarf deaktiviert werden.

Download TwinCAT 2 PLC Control Export-Datei: <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019300875.zip>

VAR_INPUT

```

bInitialize      : BOOL := FALSE;
nAddr           : BYTE;
(* Occupancy Sensor Parameters *)
bPIREnable      : BOOL := TRUE;
nPIRMotion      : BYTE := 4;      (* 5 s + (nPIRMotion * 5 s) *)
nPIRRepeat      : BYTE := 6;      (* 5 s + (nPIRRepeat * 5 s) *)
nPIRVerificationLevel : BYTE := 1;  (* standard *)
nPIRDetectionRange : BYTE := 255;  (* (only for HF sensors) 100 % *)
(* Light Sensor Parameters *)
bLSEnable       : BOOL := TRUE;
tLSCycleTime    : TIME := T#1M;

```

bInitialize: Durch eine positiven Flanke an diesem Eingang wird das DALI-Steuergerät initialisiert. Die Parameter werden in den DALI-Steuergerät geschrieben welcher über die Kurzadresse *nAddr* erreichbar sein muss. Während der Initialisierung ist der Ausgang *bInitializing* TRUE.

nAddr: Kurzadresse des DALI-Steuergerätes.

bPIREnable: Parameter: Ist dieser Eingang TRUE, so wird der Bewegungsmelder (Instanz 1) freigegeben. Durch ein FALSE wird der Präsenzmelder deaktiviert.

nPIRMotion: Parameter: Setzt den Wert für den *Motion Timer*. Wird keine Bewegung erkannt, so wird erst nach Ablauf des *Motion Timer* der Status des Bewegungsmelders geändert. Die Zeit berechnet sich aus der Formel $5\text{ s} + nPIRMotion * 5\text{ s}$. Somit sind Zeiten von 5 s bis 21 min 20 s möglich.

nPIRRepeat: Parameter: Setzt den Wert für den *Repeat Timer*. Der Status des Bewegungsmelders wird nach Ablauf des *Repeat Timer* erneut gesendet, auch dann, wenn der Status sich nicht geändert hat. Die Zeit berechnet sich aus der Formel $5\text{ s} + nPIRRepeat * 5\text{ s}$. Somit sind Zeiten von 5 s bis 21 min 20 s möglich.

nPIRVerificationLevel: Parameter: Die Erfassungsempfindlichkeit kann angepasst werden (0: Bewegungserkennung deaktiviert, 1: sehr empfindlich bis 15: sehr unempfindlich).

nPIRDetectionRange: Parameter: Kommt ein HF-Sensor zum Einsatz, so kann die Größe für den Erfassungsbereich eingestellt werden (0: 0 % bis 255: 100 %).

bLSEnable: Parameter: Ist dieser Eingang TRUE, so wird der Helligkeitssensor (Instanz 0) freigegeben. Durch ein FALSE wird der Helligkeitssensor deaktiviert.

tLSCycleTime: Zykluszeit, in der der aktuelle Istwert des Helligkeitssensors ausgelesen wird.

Weitere Einzelheiten zu den Parametern sind aus der Norm IEC 62386 und der Dokumentation des Herstellers zu entnehmen.

VAR_OUTPUT

```

bInitializing    : BOOL;
bError          : BOOL;
nErrorId        : UDINT;
nInputDeviceError : BYTE;
(* Occupancy Sensor *)
bPIROccupied    : BOOL;
(* Light Sensor *)
nActualLightLevel : UINT;

```

bInitializing: Dieser Ausgang ist während der Initialisierung TRUE.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Siehe [Fehlercodes](#) [► 391].

nInputDeviceError: Vor der Initialisierung wird der Status des DALI-Steuergerätes (INPUT DEVICE ERROR) abgefragt. 0 bedeutet kein Fehler. Die einzelnen Fehlernummern sind herstellerspezifisch.

bPIROccupied: Dieser Ausgang zeigt den Status des Bewegungsmelder an.

nActualLightLevel: Dieser Ausgang zeigt den Status des Helligkeitssensors an.

VAR_IN_OUT

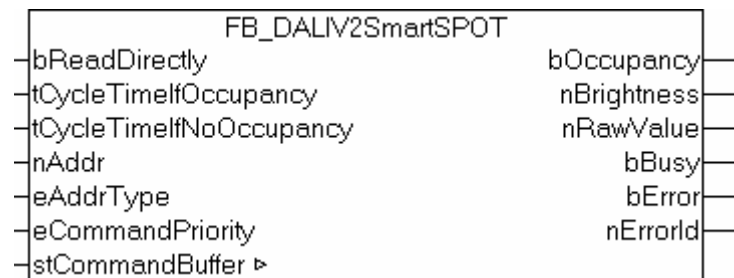
```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.1.300 FB_DALIV2SmartSPOT



Dieser Baustein liest zyklisch den Zustand eines smartSPOT-Sensors oder MSensor 02 aus und skaliert aus dem empfangenen Wert die gemessene Helligkeit und die erkannte Präsenz.



Soll der Baustein mit dem MSensor 02 eingesetzt werden, so muss sich dieser im Slave-Mode befinden.

Es können zwei verschiedene Zykluszeiten vorgegeben werden. Eine Zykluszeit, die benutzt wird, wenn keine Präsenz erkannt wurde, und eine die bei aktiver Präsenz zum Einsatz kommt. Hierdurch besteht die Möglichkeit, die Zugriffe auf den DALI-Bus zu minimieren. Ist Präsenz erkannt, so kann in der Regel eine langsamere Zykluszeit (z.B. 20 s) gewählt werden, da die Lichtregelung und das Ausschalten der Beleuchtung nicht zeitkritisch sind. Ist keine Präsenz vorhanden, so sollte eine kleinere Zykluszeit (z.B. 2s) gewählt werden. Dadurch erfolgt das Einschalten der Beleuchtung beim Betreten des Raums mit einer möglichst geringen Reaktionszeit.

Die gemessene Helligkeit des Sensors ist abhängig von der Reflektion der im Raum vorhandenen Möbel. Der gemessene Wert beträgt nur 20 % bis 40 % von der realen Helligkeit auf der gemessenen Fläche. Beispiel: 500 Lux auf dem Tisch mit einem Reflektionsgrad von 30 % ergibt einen Helligkeitswert von 150 Lux. Der Messbereich des Sensors ist so ausgelegt, dass die übliche Helligkeit am Arbeitsplatz von 200 Lux bis 1000 Lux geregelt werden kann.

Weitere Informationen und eine Beschreibung der elektrischen und physikalischen Eigenschaften finden Sie in der Produktbeschreibung des smartSPOT-Sensors oder des MSensor 02.

VAR_INPUT

```
bReadDirectly : BOOL := FALSE;
tCycleTimeIfOccupancy : TIME := t#20s;
tCycleTimeIfNoOccupancy : TIME := t#2s;
nAddr : BYTE := 0;
eAddrType : E_DALIV2AddrType := eDALIV2AddrTypeShort;
eCommandPriority : E_DALIV2CommandPriority := eDALIV2CommandPriorityMiddle;
```

bReadDirectly: Durch eine positive Flanke an diesem Eingang werden die Istwerte unmittelbar gelesen.

tCycleTimeIfOccupancy: Zykluszeit für das Auslesen der Istwerte bei aktiver Präsenz.

tCycleTimeIfNoOccupancy: Zykluszeit für das Auslesen der Istwerte bei nicht aktiver Präsenz.

nAddr: Adresse eines Teilnehmers oder einer Gruppe.

eAddrType: Kurzadresse [▶ 387], Gruppenadresse oder Sammelruf (Broadcast).

eCommandPriority: Priorität [▶ 387] (hoch, mittel oder niedrig), mit der der Befehl von der Library abgearbeitet wird.

VAR_OUTPUT

```
bOccupancy      : BOOL;
nBrightness     : INT;
nRawValue       : BYTE;
bBusy           : BOOL;
bError          : BOOL;
nErrorId        : UDINT;
```

bOccupancy: Der Ausgang ist gesetzt, wenn der Sensor Präsenz erkennt.

nBrightness: Gemessene Helligkeit in Lux.

nRawValue: Der ausgelesene Wert vom Sensor vor der Umrechnung.

bBusy: Bei der Aktivierung des Bausteins wird der Ausgang gesetzt und bleibt so lange aktiv bis der Befehl abgearbeitet wurde.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten. Wird durch das Ausführen eines Befehls an den Eingängen auf FALSE zurückgesetzt.

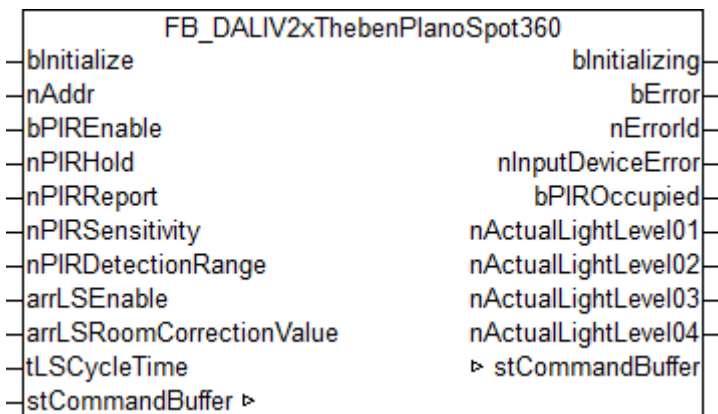
nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Wird durch das Ausführen eines Befehls an den Eingängen auf 0 zurückgesetzt. Siehe Fehlercodes [▶ 391].

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit den Bausteinen FB_DALIV2Communication() [▶ 95] (KL6811) oder FB_KL6821Communication() [▶ 104] (KL6821).

5.1.301 FB_DALIV2xThebenPlanoSpot360



Dieser Baustein dient als Beispiel und ist aus diesem Grund nicht in der Bibliothek enthalten. Der Baustein kann als Export-Datei heruntergeladen und in das gewünschte Projekt importiert werden. Bei Bedarf können somit beliebige Anpassungen durchgeführt werden.

Dieser Baustein wertet die 3 gemessene Helligkeit und die Präsenz des ThebenHTS PlanoSpot DALI-Steuergerätes aus. Ebenfalls ist es mit diesem Baustein möglich, das DALI-Steuergerät zu initialisieren.

Das DALI-Steuergerät belegt eine Kurzadresse und enthält 5 Instanzen. Instanz 0 ist der Bewegungsmelder und entspricht der Norm IEC 62386-303. Die Helligkeitssensoren sind in Instanz 1 bis 4 abgelegt und entsprechen der Norm IEC 62386-304.

Eine erkannte Bewegung wird als Ereignis versendet, während die Helligkeiten zyklisch ausgelesen werden. Jede Instanz kann einzeln bei Bedarf deaktiviert werden.

Download der TwinCAT 2 PLC Control Export-Datei: <https://infosys.beckhoff.com/content/1031/tcplclibdali/Resources/12019302283.zip>

VAR_INPUT

```
bInitialize          : BOOL := FALSE;
nAddr                : BYTE;
(* Occupancy Sensor Parameters *)
bPIREnable          : BOOL := TRUE;
nPIRHold             : BYTE := 1;      (* 10 s *)
nPIRReport           : BYTE := 30;    (* 30 s *)
nPIRSensitivity      : BYTE := 3;     (* standard *)
nPIRDetectionRange  : BYTE := 0;     (* standard *)
(* Light Sensor Parameters *)
arrLSEnable          : ARRAY [1..4] OF BOOL := TRUE, TRUE, TRUE, TRUE;
arrLSRoomCorrectionValue : ARRAY [1..4] OF BYTE := 30, 30, 30, 30; (* 0.3 *)
tLSCycleTime        : TIME := t#1m;
```

bInitialize: Durch eine positiven Flanke an diesem Eingang wird das DALI-Steuergerät initialisiert. Die Parameter werden in das DALI-Steuergerät geschrieben welcher über die Kurzadresse *nAddr* erreichbar sein muss. Während der Initialisierung ist der Ausgang *bInitializing* TRUE.

nAddr: Kurzadresse des DALI-Steuergerätes.

bPIREnable: Parameter: Ist dieser Eingang TRUE, so wird der Bewegungsmelder (Instanz 0) freigegeben. Durch ein FALSE wird der Präsenzmelder deaktiviert.

nPIRHold: Parameter: Setzt den Wert für den *Hold Timer*. Wird keine Bewegung erkannt, so wird erst nach Ablauf des *Hold Timer* der Status des Bewegungsmelders geändert. Die Einheit ist 10 s. Somit sind Zeiten bis 42 min 20 s (Wert 254) möglich. Der Wert 0 entspricht 1 s während der Wert 255 ignoriert wird.

nPIRReport: Parameter: Setzt den Wert für den *Report Timer*. Der Status des Bewegungsmelders wird nach Ablauf des *Report Timer* erneut gesendet, auch dann, wenn der Status sich nicht geändert hat. Die Einheit ist 1 s. Durch den Wert 0 wird der *Report Timer* deaktiviert. Somit sind Zeiten bis 4 min 15 s (Wert 255) möglich.

nPIRSensitivity: Parameter: Die Erfassungsempfindlichkeit kann in fünf Stufen angepasst werden (1: sehr unempfindlich bis 5: sehr empfindlich).

nPIRDetectionRange: Parameter: Der Sensor unterstützt zwei verschiedene Größen für den Erfassungsbereich (0: standard und 1: reduziert).

arrLSEnable: Parameter: Ist dieser Eingang TRUE, so wird der Helligkeitssensor (Instanz 1) freigegeben. Durch ein FALSE wird der Helligkeitssensor deaktiviert.

arrLSRoomCorrectionValue: Parameter: Mit dem Raum-Korrekturfaktor kann der von dem Melder gemessene Wert mit einem Referenzgerät (Luxmeter) gemessenen Wert abgeglichen werden.

tLSCycleTime: Zykluszeit, in der die aktuellen Istwerte der Helligkeitssensoren ausgelesen werden.

Weitere Einzelheiten zu den Parametern sind aus der Norm IEC 62386 und der Dokumentation des Herstellers zu entnehmen.

VAR_OUTPUT

```
bInitializing        : BOOL;
bError               : BOOL;
nErrorId             : UDINT;
nInputDeviceError    : BYTE;
(* Occupancy Sensor *)
bPIROccupied         : BOOL;
(* Light Sensor *)
nActualLightLevel01  : UINT; (* Light sensor 1 (integral) *)
```

```
nActualLightLevel02 : UINT; (* Light sensor 2 (inner) *)
nActualLightLevel03 : UINT; (* Light sensor 3 (middle) *)
nActualLightLevel04 : UINT; (* Light sensor 4 (window) *)
```

bInitializing: Dieser Ausgang ist während der Initialisierung TRUE.

bError: Dieser Ausgang wird auf TRUE geschaltet, wenn bei der Ausführung eines Befehls ein Fehler aufgetreten ist. Der befehlspezifische Fehlercode ist in *nErrorId* enthalten.

nErrorId: Enthält den befehlspezifischen Fehlercode des zuletzt ausgeführten Befehls. Siehe Fehlercodes [▶ 391].

nInputDeviceError: Vor der Initialisierung wird der Status des DALI-Steuergerätes (INPUT DEVICE ERROR) abgefragt. 0 bedeutet kein Fehler. Die einzelnen Fehlernummern sind herstellerspezifisch.

bPIROccupied: Dieser Ausgang zeigt den Status des Bewegungsmelder an.

nActualLightLevel01 ... nActualLightLevel04: Diese Ausgänge zeigen den Status der Helligkeitssensoren an.

VAR_IN_OUT

```
stCommandBuffer : ST_DALIV2CommandBuffer;
```

stCommandBuffer: Verweis auf die interne Struktur zur Kommunikation mit dem Baustein FB_KL6821Communication() [▶ 104] (KL6821).

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2260	PC/CX, BX oder BC	TcDALIV2-Bibliothek ab V2.12.0

5.2 Variablen

5.2.1 Vorschaltgeräte - Variablen

Jedes DALI-Vorschaltgerät besitzt eine bestimmte Anzahl von Variablen (Parameter), aus denen verschiedene Informationen ausgelesen oder auch einzelne Parameter verändert werden können.

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
<u>ACTUAL DIM LEVEL</u> [▶ 370]	?	254	0, MIN LEVEL ... MAX LEVEL	1 Byte	
<u>POWER ON LEVEL</u> [▶ 370]	254	254	1 ... 254	1 Byte	
<u>SYSTEM FAILURE LEVEL</u> [▶ 370]	254	254	0 ... 255	1 Byte	
<u>MIN LEVEL</u> [▶ 371]	PHYSICAL MIN LEVEL	PHYSICAL MIN LEVEL	PHYSICAL MIN LEVEL ... MAX LEVEL	1 Byte	
<u>MAX LEVEL</u> [▶ 371]	254	254	MIN LEVEL ... 254	1 Byte	
<u>FADE RATE</u> [▶ 371]	7	7	1 ... 15	1 Byte	

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
FADE TIME [▶ 371]	0	0	0 ... 15	1 Byte	
SHORT ADDRESS [▶ 372]	255	keine Änderung	0 ... 63, 255	1 Byte	
SEARCH ADDRESS [▶ 372]	FF FF FF	FF FF FF	00 00 00 ... FF FF FF	3 Byte	
RANDOM ADDRESS [▶ 372]	FF FF FF	FF FF FF	00 00 00 ... FF FF FF	3 Byte	
GROUP 0-7 [▶ 372]	0	0	0 ... 255	1 Byte	
GROUP 8-15 [▶ 372]	0	0	0 ... 255	1 Byte	
SCENE 0 [▶ 372]	255	255	0 ... 255	1 Byte	
...	
SCENE 15 [▶ 372]	255	255	0 ... 255	1 Byte	
STATUS INFORMATION [▶ 372]	???? ?????	0?10 0????	0 ... 255	1 Byte	nur Lesen
VERSION NUMBER [▶ 373]	herstellerabhängig	herstellerabhängig	0 ... 255	1 Byte	nur Lesen
DEVICE TYPE [▶ 373]	herstellerabhängig	herstellerabhängig	0 ... 255	1 Byte	nur Lesen
PHYSICAL MIN LEVEL [▶ 373]	herstellerabhängig	herstellerabhängig	1 ... 254	1 Byte	nur Lesen

?: unbestimmt

ACTUAL DIM LEVEL

Diese Variable beinhaltet den aktuellen Lampenleistungswert der Lampe. Mit dem Baustein [FB_DALIV2QueryActualLevel\(\)](#) [▶ 132] kann der Wert ausgelesen werden.

POWER ON LEVEL

Wird das Vorschaltgerät mit Spannung versorgt, wird die Lampe mit dem Leistungswert angesteuert, der in der Variablen POWER ON LEVEL abgelegt ist. Voraussetzung ist, dass der DALI-Bus schon mit Spannung versorgt wird und in Ruhepegel ist. Der Bereich von POWER ON LEVEL wird durch die beiden Variablen MIN LEVEL und MAX LEVEL begrenzt.

Die Variable kann mit dem Baustein [FB_DALIV2QueryPowerOnLevel\(\)](#) [▶ 148] gelesen und mit dem Baustein [FB_DALIV2StoreDTRAsPowerOnLevel\(\)](#) [▶ 119] geschrieben werden.

SYSTEM FAILURE LEVEL

Tritt ein Fehler am DALI-Bus auf (Ruhespannung länger als 500ms unterhalb des festgelegten Pegelbereichs), so wird die Lampe mit dem Leistungswert aus der Variablen SYSTEM FAILURE LEVEL angesteuert. Steht 255 (Maske) in der Variablen, so ändert sich die Lampenleistung nicht. Begrenzt wird der Wertebereich durch MIN LEVEL und MAX LEVEL.

Die Variable kann mit dem Baustein [FB_DALIV2QuerySystemFailureLevel\(\)](#) [▶ 155] gelesen und mit dem Baustein [FB_DALIV2StoreDTRAsSystemFailureLevel\(\)](#) [▶ 121] geschrieben werden.

MIN LEVEL / MAX LEVEL

Die Ausgabe des Lampenleistungswertes wird durch die Variablen MIN LEVEL und MAX LEVEL innerhalb des Vorschaltgeräts begrenzt. Ausnahme sind die Leistungswerte 0 (Aus) und 255 (Maske).

FADE RATE

Die FADE RATE (Stufengeschwindigkeit) legt die Veränderungsgeschwindigkeit (in Stufen pro Sekunde) für die Änderung des Lampenleistungswertes fest. Auswirkung hat diese Variable auf die Befehle [FB_DALIV2Up\(\)](#) [▶ 131] und [FB_DALIV2Down\(\)](#) [▶ 123]. Die absolute Stufengeschwindigkeit wird nicht direkt angegeben, sondern nach folgender Formel berechnet:

$$T = \frac{506}{\sqrt{2^n}}$$

T = absolute Stufengeschwindigkeit
n = Wert, der in der Variablen FADE RATE abgelegt wurde

Es ergeben sich folgende Werte:

n	absolute Stufengeschwindigkeit
0	nicht erlaubt
1	357,796 Stufen/s
2	253,000 Stufen/s
3	178,898 Stufen/s
4	126,500 Stufen/s
5	89,449 Stufen/s
6	63,250 Stufen/s
7	44,725 Stufen/s
8	31,625 Stufen/s
9	22,362 Stufen/s
10	15,813 Stufen/s
11	11,181 Stufen/s
12	7,906 Stufen/s
13	5,591 Stufen/s
14	3,953 Stufen/s
15	2,795 Stufen/s

FADE TIME

Die FADE TIME (Stufenzeit) legt die Zeit für die Veränderung des aktuellen Lampenleistungswertes zum geforderten Wert fest. Im Falle einer ausgeschalteten Lampe ist die Vorheiz- und Zündzeit nicht in der Stufenzeit enthalten. Betroffen sind dabei die Bausteine [FB_DALIV2DirectArcPowerControl\(\)](#) [▶ 122] und [FB_DALIV2GoToScene\(\)](#) [▶ 125]. Die absolute Stufenzeit wird nicht direkt angegeben, sondern nach folgender Formel berechnet:

$$T = \frac{1}{2} \sqrt{2^n}$$

T = absolute Stufenzeit
n = Wert, der in der Variablen FADE TIME abgelegt wurde

Es ergeben sich folgende Werte:

n	absolute Stufenzeit
0	< 0,707 s
1	0,707 s
2	1,000 s

n	absolute Stufenzeit
3	1,414 s
4	2,000 s
5	2,828 s
6	4,000 s
7	5,657 s
8	8,000 s
9	11,314 s
10	16,000 s
11	22,627 s
12	32,000 s
13	45,255 s
14	64,000 s
15	90,510 s

SHORT ADDRESS

In dieser Variable ist die Kurzadresse abgelegt. Eine gültige Kurzadresse liegt im Wertebereich von 0 bis 63. Wird 255 in die Variable geschrieben, so gilt die Kurzadresse als gelöscht. Gesetzt werden kann die Kurzadresse mit den Bausteinen [FB_DALIV2SetShortAddress\(\) \[▶ 114\]](#) und [FB_DALIV2StoreDTRAsShortAddress\(\) \[▶ 120\]](#). Durch den Aufruf des Bausteins [FB_DALIV2QueryMissingShortAddress\(\) \[▶ 145\]](#) wird abgefragt, ob ein Vorschaltgerät noch keine Kurzadresse besitzt.

SEARCH ADDRESS

Die Suchadresse wird nur bei der Vergabe von den Kurzadressen benötigt.

RANDOM ADDRESS

Die Zufallsadresse, auch als Langadresse bezeichnet, wird vom Hersteller bei Auslieferung der Vorschaltgeräte vorgegeben. Mit Hilfe des Bausteins [FB_DALIV2QueryRandomAddress\(\) \[▶ 149\]](#) kann die 3 Byte lange Zufallsadresse ausgelesen werden.

GROUP 0-7 / GROUP 8-15

Innerhalb eines DALI-Netzes existieren 16 Gruppen. Jedes Vorschaltgerät kann einer, mehreren oder keiner Gruppe gehören. Befehle, die an eine Gruppe geschickt werden, wirken sich auf alle Vorschaltgeräte aus, die zu der entsprechenden Gruppe gehören. Der Baustein [FB_DALIV2QueryGroups\(\) \[▶ 139\]](#) liest die beiden 8-Bit Variablen aus und fasst diese zu einem 16-Bit-Wert zusammen. Jedes Bit repräsentiert die Zugehörigkeit zu einer Gruppe.

SCENE 0-15

Jedes DALI-Vorschaltgerät kann Lampenleistungswerte für 16 verschiedene Szenen abspeichern. Pro Szene gilt ein Lampenleistungswert. Wird der Befehl zum Aufruf einer Szene ([FB_DALIV2GoToScene\(\) \[▶ 125\]](#)) an einem Teilnehmer, eine Gruppe oder an alle (Sammelruf) aufgerufen, so werden jeweils die entsprechenden Lampen auf den abgespeicherten Wert gesetzt. Begrenzt wird die Ausgabe durch MAX LEVEL, MIN LEVEL und PHYSICAL MIN LEVEL.

STATUS INFORMATION

Die Statusinformation enthält die wichtigsten Zustandsmeldungen eines Vorschaltgerätes. Mit dem Baustein [FB_DALIV2QueryStatus\(\) \[▶ 154\]](#) kann der 8-Bit-Wert ausgelesen werden. Die Bedeutung der einzelnen Bit ist wie folgt definiert:

Bit	Beschreibung
0	Status des Vorschaltgerätes. 0: OK.
1	Lampenausfall. 0: OK.
2	Lampenleistung ein. 0: AUS.
3	Grenzwertfehler. 0: letzter angeforderter Lampenleistungswert liegt zwischen MIN LEVEL und MAX LEVEL oder AUS.
4	Stufung fertig: 0: Stufung beendet. 1: Stufung aktiv.
5	Rücksetzzustand. 0: Nein.
6	Kurzadresse fehlt. 0: Nein.
7	Störung Stromversorgung. 0: Nein, Rücksetzen oder Lampenleistungssteuerbefehl wurde nach dem letzten Einschalten empfangen.

VERSION NUMBER

Die Versionsnummer entspricht der Versionsnummer der IEC-Norm, nach der das Vorschaltgerät entwickelt und hergestellt wurde. Die Versionsnummer kann nur gelesen werden und wird vom Hersteller der Vorschaltgeräte vorgegeben. Die Hauptversion (*nMajorVersion*) und die Unterversion (*nMinorVersion*) können einen Wertebereich von je 0 bis 15 (4-Bit) annehmen.

DEVICE TYPE

Der Wert kann mit dem Baustein `FB_DALIV2QueryDeviceType()` [▶ 137] ausgelesen werden. Folgende Gerätetypen sind laut Norm IEC 62386 definiert:

Wert	Beschreibung
0	Standardgerät.
1	Gerät für <u>Notbeleuchtung</u> [▶ 32].
2	Gerät für <u>Entladungslampen</u> [▶ 36].
3	Gerät für Niedervolt-Halogenlampen.
4	Gerät zum Dimmen von Glühlampen.
5	Gerät zur Umwandlung von digitalen Signalen in Gleichspannungssignale.
6	Gerät für <u>Licht emittierende Dioden (LED)</u> [▶ 36].
7	Gerät für Schaltfunktionen.
8	Gerät zur <u>Farb-/Farbtemperatursteuerung</u> [▶ 34].
9	Sequenz.

PHYSICAL MIN LEVEL

Der physikalisch kleinste mögliche Lampenleistungswert wird vom Hersteller in der Variablen PHYSICAL MIN LEVEL abgelegt. Der Wert kann mit dem Baustein `FB_DALIV2QueryPhysicalMinimumLevel()` [▶ 146] nur ausgelesen werden.

5.2.2 Notbeleuchtung - Variablen

Jedes DALI-Notlichtversorgungsgerät besitzt eine bestimmte Anzahl von Variablen (Parameter), aus denen verschiedene Informationen ausgelesen, oder auch einzelne Parameter verändert werden können.

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
<u>EMERGENCY LEVEL</u> [▶ 375]	EMERGENCY MAX LEVEL	keine Änderung	EMERGENCY MIN LEVEL ... EMERGENCY MAX LEVEL oder MASK	1 Byte	

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
<u>EMERGENCY MIN LEVEL</u> [▶ 375]	herstellerabhängig	keine Änderung	1 ... EMERGENCY MAX LEVEL oder MASK	1 Byte	nur Lesen
<u>EMERGENCY MAX LEVEL</u> [▶ 375]	herstellerabhängig	keine Änderung	EMERGENCY MIN LEVEL ... 254 oder MASK	1 Byte	nur Lesen
<u>PROLONG TIME</u> [▶ 375]	4	keine Änderung	0 ... 255	1 Byte	
<u>TEST DELAY TIME</u> [▶ 375]	0	0	00 00 ... FF FF	2 Byte	
<u>FUNCTION TEST DELAY TIME</u> [▶ 376]	0	keine Änderung	00 00 ... FF FF	2 Byte	
<u>DURATION TEST DELAY TIME</u> [▶ 376]	0	keine Änderung	00 00 ... FF FF	2 Byte	
<u>FUNCTION TEST INTERVAL</u> [▶ 376]	7	keine Änderung	0, 1 ... 255	1 Byte	
<u>DURATION TEST INTERVAL</u> [▶ 376]	13	keine Änderung	0, 1 ... 97	1 Byte	
<u>TEST EXECUTION TIMEOUT</u> [▶ 376]	7	keine Änderung	0 ... 255	1 Byte	
<u>BATTERY CHARGE</u> [▶ 376]	???? ????	keine Änderung	0 ... 255	1 Byte	
<u>DURATION TEST RESULT</u> [▶ 376]	0	keine Änderung	0 ... 255	1 Byte	
<u>LAMP EMERGENCY TIME</u> [▶ 377]	0	keine Änderung	0 ... 255	1 Byte	
<u>LAMP TOTAL OPERATION TIME</u> [▶ 377]	0	keine Änderung	0 ... 255	1 Byte	
<u>RATED DURATION</u> [▶ 377]	herstellerabhängig	keine Änderung	0 ... 255	1 Byte	nur Lesen
<u>EMERGENCY MODE</u> [▶ 377]	???? ????	keine Änderung	0 ... 255	1 Byte	
<u>FEATURES</u> [▶ 377]	herstellerabhängig	keine Änderung	0 ... 255	1 Byte	nur Lesen
<u>FAILURE STATUS</u> [▶ 377]	???? ????	keine Änderung	0 ... 255	1 Byte	
<u>EMERGENCY STATUS</u> [▶ 378]	???? ????	keine Änderung	0 ... 255	1 Byte	

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
<u>EXTENDED VERSION NUMBER</u> [▶ 378]	1	keine Änderung	0 ... 255	1 Byte	nur Lesen

?: unbestimmt

EMERGENCY LEVEL

Diese Variable beinhaltet die Notbeleuchtungsstärke der Lampe. Begrenzt wird dieser Wert durch die Variablen EMERGENCY MAX LEVEL und EMERGENCY MIN LEVEL. Mit dem Baustein [FB_DALIV2QueryEmergencyLevel\(\)](#) [▶ 227] kann der Wert ausgelesen werden.

EMERGENCY MIN LEVEL / EMERGENCY MAX LEVEL

Die Ausgabe der Notbeleuchtungsstärke wird durch die Variablen EMERGENCY MIN LEVEL und EMERGENCY MAX LEVEL innerhalb des Vorschaltgerätes begrenzt. Ausnahmen sind die Leistungswerte 0 (Aus) und 255 (Maske). Durch den Hersteller des Vorschaltgerätes werden der EMERGENCY MIN LEVEL und der EMERGENCY MAX LEVEL fest vorgegeben. Mit den Bausteinen [FB_DALIV2QueryEmergencyMinLevel\(\)](#) [▶ 229] und [FB_DALIV2QueryEmergencyMaxLevel\(\)](#) [▶ 228] können die beiden Variablen aus dem Vorschaltgerät ausgelesen werden.

PROLONG TIME

Die Verlängerungszeit gibt an, um welche Zeitspanne der Notbetrieb verlängert werden soll nach der Rückkehr der Netzspannung. Die Einheit beträgt 30 Sekunden pro Schritt. Über den Baustein [FB_DALIV2QueryTestTiming\(\)](#) [▶ 238] kann der Wert aus dem Vorschaltgerät ausgelesen werden. Beschrieben wird diese Variable über den Baustein [FB_DALIV2StoreDTRAsProlongTime\(\)](#) [▶ 251].

TEST DELAY TIME

Über die Variable TEST DELAY TIME werden die Variablen FUNCTION TEST DELAY TIME, DURATION TEST DELAY TIME FUNCTION TEST INTERVAL und DURATION TEST INTERVAL gesetzt. Hierdurch wird das Zeitverhalten für den Funktionstest und den Betriebsdauertest wie folgt festgelegt:

Funktionstest:

Mit [FB_DALIV2SetDTR\(\)](#) [▶ 166] das höherwertige Byte für die Variable FUNCTION TEST DELAY TIME (Verzögerungszeit für den Funktionstest) in das DTR (Data Transfer Register) schreiben.

Mit [FB_DALIV2StoreDTRAsTestDelayTimeHighByte\(\)](#) [▶ 252] den Inhalt des DTR in das höherwertige Byte der 16-Bit-Variable TEST DELAY TIME schreiben.

Mit [FB_DALIV2SetDTR\(\)](#) [▶ 166] das niederwertige Byte für die Variable FUNCTION TEST DELAY TIME (Verzögerungszeit für den Funktionstest) in das DTR (Data Transfer Register) schreiben.

Mit [FB_DALIV2StoreDTRAsTestDelayTimeLowByte\(\)](#) [▶ 253] den Inhalt des DTR in das niederwertige Byte der 16-Bit-Variable TEST DELAY TIME schreiben.

Mit [FB_DALIV2SetDTR\(\)](#) [▶ 166] den Wert für die Variable FUNCTION TEST INTERVAL (Intervallzeit für den Funktionstest) in das DTR (Data Transfer Register) schreiben.

Mit [FB_DALIV2StoreDTRAsFunctionTestInterval\(\)](#) [▶ 250] wird der Inhalt des DTR in die Variable FUNCTION TEST INTERVAL geschrieben. Außerdem wird der Inhalt der Variablen TEST DELAY TIME in die Variable FUNCTION TEST DELAY TIME kopiert.

Betriebsdauertest:

Mit [FB_DALIV2SetDTR\(\)](#) [▶ 166] das höherwertige Byte für die Variable DURATION TEST DELAY TIME (Verzögerungszeit für den Betriebsdauertest) in das DTR (Data Transfer Register) schreiben.

Mit [FB_DALIV2StoreDTRAsTestDelayTimeHighByte\(\)](#) [▶ 252] den Inhalt des DTR in das höherwertige Byte der 16-Bit-Variable TEST DELAY TIME schreiben.

Mit [FB_DALIV2SetDTR\(\)](#) [▶ 166] das niederwertige Byte für die Variable DURATION TEST DELAY TIME (Verzögerungszeit für den Betriebsdauertest) in das DTR (Data Transfer Register) schreiben.

Mit [FB_DALIV2StoreDTRAsTestDelayTimeLowByte\(\)](#) [► 253] den Inhalt des DTR in das niederwertige Byte der 16-Bit-Variable TEST DELAY TIME schreiben.

Mit [FB_DALIV2SetDTR\(\)](#) [► 166] den Wert für die Variable FUNCTION TEST INTERVAL (Intervallzeit für den Funktionstest) in das DTR (Data Transfer Register) schreiben.

Mit [FB_DALIV2StoreDTRAsDurationTestInterval\(\)](#) [► 248] wird der Inhalt des DTR in die Variable DURATION TEST INTERVAL geschrieben. Außerdem wird der Inhalt der Variablen TEST DELAY TIME in die Variable DURATION TEST DELAY TIME kopiert.

FUNCTION TEST DELAY TIME

In dieser Variablen ist die Verzögerungszeit für den Funktionstest festgelegt. Nach dieser Zeit wird zum ersten Mal der Funktionstest ausgeführt. Die Einheit dieser Variablen beträgt 15 Minuten pro Schritt.

Abgefragt werden kann diese Variable über den Baustein [FB_DALIV2QueryTestTiming\(\)](#) [► 238]. Beschrieben wird diese Variable über TEST DELAY TIME (siehe oben).

DURATION TEST DELAY TIME

In dieser Variablen ist die Verzögerungszeit für den Betriebsdauertest festgelegt. Nach dieser Zeit wird zum ersten Mal der Betriebsdauertest ausgeführt. Die Einheit dieser Variablen beträgt 15 Minuten pro Schritt.

Abgefragt werden kann diese Variable über den Baustein [FB_DALIV2QueryTestTiming\(\)](#) [► 238]. Beschrieben wird diese Variable über TEST DELAY TIME (siehe oben).

FUNCTION TEST INTERVAL

In dieser Variablen ist die Intervallzeit für den Funktionstest festgelegt. In diesen Zeitabständen wird der Funktionstest periodisch ausgeführt. Die Einheit dieser Variablen beträgt 1 Tag pro Schritt. Abgefragt

werden kann diese Variable über den Baustein [FB_DALIV2QueryTestTiming\(\)](#) [► 238]. Beschrieben wird diese Variable über TEST DELAY TIME (siehe oben).

DURATION TEST INTERVAL

In dieser Variablen ist die Intervallzeit für den Betriebsdauertest festgelegt. In diesen Zeitabständen wird der Betriebsdauertest periodisch ausgeführt. Die Einheit dieser Variablen beträgt 1 Woche pro Schritt. Abgefragt

werden kann diese Variable über den Baustein [FB_DALIV2QueryTestTiming\(\)](#) [► 238]. Beschrieben wird diese Variable über TEST DELAY TIME (siehe oben).

TEST EXECUTION TIMEOUT

Der Funktionstest oder Betriebsdauertest kann durch verschiedene Ereignisse unterbrochen werden. Mit der Variablen TEST EXECUTION TIMEOUT kann die maximale Ausführungszeit festgelegt werden, in der der jeweilige Test abgeschlossen sein muss. Die Einheit dieser Variablen beträgt 1 Tag pro Schritt. Abgefragt werden kann diese Variable über den Baustein [FB_DALIV2QueryTestTiming\(\)](#) [► 238]. Beschrieben wird diese Variable über den Baustein [FB_DALIV2StoreDTRAsTestExecutionTimeout\(\)](#) [► 254].

BATTERY CHARGE

Der aktuelle Ladezustand vom Akku ist in dieser Variablen abrufbar. 0 bedeutet minimale Ladung und 254 bedeutet maximale Ladung. Kann der Ladezustand vom Vorschaltgerät nicht ermittelt werden, so enthält diese Variable den Wert 255. Mit dem Baustein [FB_DALIV2QueryBatteryCharge\(\)](#) [► 225] kann diese Variable ausgelesen werden.

DURATION TEST RESULT

Das Ergebnis eines Betriebsdauertestes ist in dieser Variablen abgelegt. Die Einheit beträgt 2 Minuten pro Schritt. Der Wert ist nur dann gültig, wenn in der Variablen EMERGENCY STATUS das Bit 2 gesetzt ist (siehe unten). Abgefragt werden kann die Variable DURATION TEST RESULT mit der Funktion

[FB_DALIV2QueryDurationTestResult\(\)](#) [► 226].

LAMP EMERGENCY TIME

Die Laufzeit der Lampe im Notbetrieb (Versorgung mit Hilfe des Akkus) wird in dieser Variablen abgespeichert. Die Einheit beträgt 1 Stunde pro Schritt. Abgefragt wird diese Variable mit dem Baustein [FB_DALIV2QueryLampEmergencyTime\(\)](#) [▶ 235] und zurückgesetzt mit dem Baustein [FB_DALIV2ResetLampTime\(\)](#) [▶ 243].

LAMP TOTAL OPERATION TIME

Die gesamte Laufzeit der Lampe wird in dieser Variablen abgespeichert. Die Einheit beträgt 4 Stunden pro Schritt. Abgefragt wird diese Variable mit dem Baustein [FB_DALIV2QueryLampTotalOperationTime\(\)](#) [▶ 236] und zurückgesetzt mit dem Baustein [FB_DALIV2ResetLampTime\(\)](#) [▶ 243].

RATED DURATION

Die Nennbetriebsdauer des Akkus ist in dieser Variablen abgelegt. Die Einheit beträgt 2 Minuten pro Schritt. Der Wert kann mit dem Baustein [FB_DALIV2QueryRatedDuration\(\)](#) [▶ 237] gelesen werden und ist vom Hersteller des Vorschaltgerätes fest vorgegeben.

EMERGENCY MODE

In EMERGENCY MODE wird der aktuelle Modus vom Vorschaltgerät abgelegt. Mit dem Baustein [FB_DALIV2QueryEmergencyMode\(\)](#) [▶ 230] kann die Variable ausgelesen werden.

Bit	Beschreibung
0	Rücksetz Modus. 0: Nein.
1	Bereitschaft Notbetrieb (Normalbetrieb). 0: Nein.
2	Notbetrieb. 0: Nein.
3	verlängerter Notbetrieb nach Rückkehr der Netzspannung. 0: Nein.
4	Funktionstest aktiv. 0: Nein.
5	Betriebsdauertest aktiv. 0: Nein.
6	angeschlossener Unterdrückungstaster aktiv. 0: nicht aktiv bzw. nicht vorhanden.
7	angeschlossene Netzspannung aktiv. 0: Aus.

FEATURES

Die unterstützten Leistungsmerkmale des Vorschaltgerätes können mit dem Baustein [FB_DALIV2QueryFeatures\(\)](#) [▶ 234] aus dieser Variablen ausgelesen werden. Der Inhalt dieser Variablen wird vom Hersteller des Vorschaltgerätes fest vorgegeben und kann nicht geändert werden.

Bit	Beschreibung
0	integriertes Notlichtversorgungsgerät. 0: Nein.
1	Notlichtversorgungsgerät in Dauerschaltung. 0: Nein.
2	schaltbares Notlichtversorgungsgerät in Dauerschaltung. 0: Nein.
3	Autotestfähigkeit. 0: Nein.
4	Einstellbare Stärke der Notbeleuchtung. 0: Nein.
5	angeschlossener Unterdrückungstaster wird unterstützt. 0: Nein.
6	Physikalische Auswahl wird unterstützt. 0: Nein.
7	reserve

FAILURE STATUS

Mögliche Fehlerzustände werden in dieser Variablen angezeigt und können mit dem Baustein [FB_DALIV2QueryFailureStatus\(\)](#) [▶ 233] ausgelesen werden.

Bit	Beschreibung
0	Störung im Schaltkreis des Vorschaltgerätes. 0: Nein.
1	Störung Akkubetriebsdauer. 0: Nein.
2	Störung Akku. 0: Nein.
3	Störung am Leuchtmittel der Notbeleuchtung. 0: Nein.
4	Zeitüberschreitung beim Funktionstest aufgetreten. 0: Nein.
5	Zeitüberschreitung beim Betriebsdauertest aufgetreten. 0: Nein.
6	Funktionstest fehlgeschlagen. 0: Nein.
7	Betriebsdauertest fehlgeschlagen. 0: Nein

EMERGENCY STATUS

Der aktuelle Zustand vom Vorschaltgerät wird angezeigt. Mit dem Baustein `FB DALIV2QueryEmergencyStatus()` [► 232] kann der Inhalt der Variablen ausgelesen werden.

Bit	Beschreibung
0	Unterdrückungsmodus. 0: Nein.
1	Funktionstest abgeschlossen und Ergebnis ist gültig. 0: Nein.
2	Betriebsdauertest abgeschlossen und Ergebnis ist gültig. 0: Nein.
3	Akkuladung betriebsbereit. 0: in Gang.
4	verzögerter Start vom Funktionstest. 0: Nein.
5	verzögerter Start vom Betriebsdauertest. 0: Nein.
6	Identifikation aktiv. 0: Nein.
7	ausgewählt. 0: Nein

EXTENDED VERSION NUMBER

Die erweiterte Versionsnummer kann mit dem Baustein `FB DALIV2QueryExtendedVersionNumber()` [► 357] ausgelesen werden. Die Versionsnummer kann nur gelesen werden und wird vom Hersteller der Vorschaltgeräte vorgegeben.

5.2.3 Entladungslampen - Variablen

Jedes DALI-Vorschaltgerät für Entladungslampen besitzt eine bestimmte Anzahl von Variablen (Parameter), aus denen verschiedene Informationen ausgelesen, oder auch einzelne Parameter verändert werden können.

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
<u>HID STATUS</u> [► 379]	0	0	0 ... 255	1 Byte	
<u>ACTUAL HID FAILURE</u> [► 379]	???? ????	keine Änderung	0 ... 255	1 Byte	
<u>STORED HID FAILURE</u> [► 379]	???? ????	keine Änderung	0 ... 255	1 Byte	
<u>HID FEATURES</u> [► 380]	herstellerabhängig	keine Änderung	0 ... 255	1 Byte	nur Lesen
<u>THERMAL OVERLOAD TIME</u> [► 380]	0	keine Änderung	00 00 ... FF FF	2 Byte	
<u>THERMAL LOAD</u> [► 380]	???? ????	keine Änderung	0 ... 255	1 Byte	

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
<u>EXTENDED VERSION NUMBER</u> [▶ 380]	1	keine Änderung	0 ... 255	1 Byte	nur Lesen

?: unbestimmt

HID STATUS

In HID STATUS wird der aktuelle Zustand vom Vorschaltgerät abgelegt. Mit dem Baustein `FB_DALIV2QueryHIDStatus()` [▶ 257] kann der Wert ausgelesen werden.

Bit	Beschreibung
0	Anlaufzeit betriebsbereit. 0: Nein.
1	Der Lampenleistungswert hat seinen angeforderten Sollwert erreicht. 0: Nein.
2	Vorschaltgerät wartet auf das Zünden der Lampe. 0: Nein.
3	reserve
4	reserve
5	reserve
6	Identifikation aktiv. 0: Nein.
7	reserve

ACTUAL HID FAILURE

In dieser Variablen stehen alle Informationen über den Fehlerzustand vom Vorschaltgerät. Ausgelesen wird die Variable mit dem Baustein `FB_DALIV2QueryActualHIDFailure()` [▶ 255]. Die jeweiligen Bits werden gesetzt, sobald ein Fehler auftritt und automatisch wieder zurückgesetzt sobald der Fehler wieder zurückgegangen ist.

Solange Bit 4 oder Bit 7 gesetzt sind, ist auch das Bit 1 in der Variablen `STATUS INFORMATION` [▶ 372] gesetzt. Der Baustein `FB_DALIV2QueryLampeFailure()` [▶ 141] liefert in diesem Fall am Ausgang `bLampFailure` ein TRUE.

Bit	Beschreibung
0	Versorgungsspannung zu niedrig. 0: Nein.
1	Versorgungsspannung zu hoch. 0: Nein.
2	Übertemperatur am Umformer. 0: Nein.
3	reserve
4	Zeitüberschreitung beim Zünden der Lampe. 0: Nein.
5	reserve
6	Lampenspannung außerhalb der Spezifikation. 0: Nein.
7	Lampenzyklusfehler. 0: Nein.

STORED HID FAILURE

In dieser Variablen stehen alle Informationen über den Fehlerzustand vom Vorschaltgerät. Ausgelesen wird die Variable mit dem Baustein `FB_DALIV2QueryStoredHIDFailure()` [▶ 258]. Zurückgesetzt werden die Fehlermeldungen durch Ausschalten vom Vorschaltgerät oder durch den Baustein `FB_DALIV2ResetStoredHIDFailure()` [▶ 263].

Bit	Beschreibung
0	Versorgungsspannung zu niedrig. 0: Nein.
1	Versorgungsspannung zu hoch. 0: Nein.
2	Übertemperatur am Umformer. 0: Nein.

Bit	Beschreibung
3	reserve
4	Zeitüberschreitung beim Zünden der Lampe. 0: Nein.
5	reserve
6	Lampenspannung außerhalb der Spezifikation. 0: Nein.
7	Lampenzyklusfehler. 0: Nein.

HID FEATURES

Die unterstützten Leistungsmerkmale des Vorschaltgerätes können mit dem Baustein [FB_DALIV2QueryHIDFeatures\(\)](#) [▶ 256] aus dieser Variablen ausgelesen werden. Der Inhalt dieser Variablen wird vom Hersteller des Vorschaltgerätes fest vorgegeben und kann nicht geändert werden.

Bit	Beschreibung
0	'Versorgungsspannung zu niedrig' kann abgefragt werden. 0: Nein.
1	'Versorgungsspannung zu hoch' kann abgefragt werden. 0: Nein.
2	'Übertemperatur am Umformer' kann abgefragt werden. 0: Nein.
3	reserve
4	reserve
5	reserve
6	'Lampenspannung außerhalb der Spezifikation' kann abgefragt werden. 0: Nein.
7	Physikalische Auswahl wird unterstützt. 0: Nein.

THERMAL OVERLOAD TIME

Sobald Übertemperatur erkannt wird, wird die Variable THERMAL OVERLOAD TIME in 15-Minuten-Schritten erhöht. Diese Variable kann nicht zurückgesetzt werden. Ein Wert von 65535 (0xFFFF) entspricht hierbei einer Zeit von 16.383 Stunden und 45 Minuten oder höher. Mit den Bausteinen [FB_DALIV2QueryThermalOverloadTimeHB\(\)](#) [▶ 261] und [FB_DALIV2QueryThermalOverloadTimeLB\(\)](#) [▶ 262] kann die Variable THERMAL OVERLOAD TIME ausgelesen werden. Der Schwellwert für die Erkennung der Übertemperatur wird durch die Variable THERMAL LOAD festgelegt (siehe unten).

THERMAL LOAD

In der Variablen THERMAL LOAD wird ein Prozentwert im Bereich von 0% bis 127,5% in 0,5%-Schritten definiert. Somit entspricht 255 einen Wert von 127,5%. Ausgelesen werden kann die Variable mit dem Baustein [FB_DALIV2QueryThermalLoad\(\)](#) [▶ 259].

EXTENDED VERSION NUMBER

Die erweiterte Versionsnummer kann mit dem Baustein [FB_QueryV2ExtendedVersionNumber\(\)](#) [▶ 357] ausgelesen werden. Die Versionsnummer kann nur gelesen werden und wird vom Hersteller der Vorschaltgeräte vorgegeben.

5.2.4 Farb-/Farbtemperatursteuerung - Variablen

Jedes DALI-Vorschaltgerät für Farb-/Farbtemperatursteuerung besitzt eine bestimmte Anzahl von Variablen (Parameter), aus denen verschiedene Informationen ausgelesen, oder auch einzelne Parameter verändert werden können.

Einige Variablen können direkt über DALI-Befehle ausgelesen werden (z.B. [FB_QueryColourStatus\(\)](#) [▶ 296] oder [FB_QueryRGBWAFControl\(\)](#) [▶ 303]). Der Baustein [FB_QueryColourValue\(\)](#) [▶ 298] kann dazu genutzt werden, weitere Variablen auszulesen.

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	notwendige Farbdarstellung (1) [▶ 380]	Bemerkung
TEMPORARY x-COORDINATE	65535	65535	0 ... 65535	2 Bytes	0, 2	
REPORT x-COORDINATE	65535	65535	0 ... 65535	2 Bytes	0	
x-COORDINATE	?	keine Änderung	0 ... 65535	2 Bytes	0	
TEMPORARY x-COORDINATE	65535	65535	0 ... 65535	2 Bytes	0, 2	
REPORT x-COORDINATE	65535	65535	0 ... 65535	2 Bytes	0	
x-COORDINATE	?	keine Änderung	0 ... 65535	2 Bytes	0	
TEMPORARY COLOUR TEMPERATURE Tc	65535	65535	1 ... 65535	2 Bytes	1	
REPORT COLOUR TEMPERATURE Tc	65535	65535	1 ... 65535	2 Bytes	1	
COLOUR TEMPERATURE Tc	?	keine Änderung	1 ... 65535	2 Bytes	1	
COLOUR TEMPERATURE Tc COOLEST	?	COLOUR TEMPERATURE Tc PHYSICAL COOLEST	COLOUR TEMPERATURE Tc PHYSICAL COOLEST ... COLOUR TEMPERATURE Tc WARMEST, 65535	2 Bytes	1	nur Lesen
COLOUR TEMPERATURE Tc WARMEST	?	COLOUR TEMPERATURE Tc PHYSICAL WARMEST	COLOUR TEMPERATURE Tc COOLEST ... COLOUR TEMPERATURE Tc PHYSICAL WARMEST, 65535	2 Bytes	1	nur Lesen

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	notwendige Farbdarstellung (1) [▶ 380]	Bemerkung
COLOUR TEMPERATURE Tc PHYSICAL COOLEST	?	keine Änderung	1 - COLOUR TEMPERATURE Tc PHYSICAL WARMEST, 65535	2 Bytes	1	nur Lesen
COLOUR TEMPERATURE Tc PHYSICAL WARMEST	?	keine Änderung	COLOUR TEMPERATURE Tc PHYSICAL COOLEST - 65534, 65535	2 Bytes	1	nur Lesen
TEMPORARY PRIMARY N DIMLEVEL	65535	65535	0 ... 65535	bis zu 12 Bytes	2	
REPORT PRIMARY N DIMLEVEL	65535	65535	0 ... 65535	bis zu 12 Bytes	2	
PRIMARY N DIMLEVEL	?	keine Änderung	0 ... 65535	bis zu 12 Bytes	2	
x-COORDINATE PRIMARY N	?	keine Änderung	0 ... 65535	bis zu 12 Bytes	0, 2	nur Lesen
y-COORDINATE PRIMARY N	?	keine Änderung	0 ... 65535	bis zu 12 Bytes	0, 2	nur Lesen
TY PRIMARY N	?	keine Änderung	0 ... 65535	bis zu 12 Bytes	0, 2	nur Lesen
TEMPORARY RED DIMLEVEL	255	255	0 ... 255	1 Byte	3	
REPORT RED DIMLEVEL	255	255	0 ... 255	1 Byte	3	
RED DIMLEVEL	?	keine Änderung	0 ... 255	1 Byte	3	
TEMPORARY GREEN DIMLEVEL	255	255	0 ... 255	1 Byte	3	
REPORT GREEN DIMLEVEL	255	255	0 ... 255	1 Byte	3	
GREEN DIMLEVEL	?	keine Änderung	0 ... 255	1 Byte	3	
TEMPORARY BLUE DIMLEVEL	255	255	0 ... 255	1 Byte	3	
REPORT BLUE DIMLEVEL	255	255	0 ... 255	1 Byte	3	

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	notwendige Farbdarstellung (1) [▶ 380]	Bemerkung
BLUE DIMLEVEL	?	keine Änderung	0 ... 255	1 Byte	3	
TEMPORARY WHITE DIMLEVEL	255	255	0 ... 255	1 Byte	3	
REPORT WHITE DIMLEVEL	255	255	0 ... 255	1 Byte	3	
WHITE DIMLEVEL	?	keine Änderung	0 ... 255	1 Byte	3	
TEMPORARY AMBER DIMLEVEL	255	255	0 ... 255	1 Byte	3	
REPORT AMBER DIMLEVEL	255	255	0 ... 255	1 Byte	3	
AMBER DIMLEVEL	?	keine Änderung	0 ... 255	1 Byte	3	
TEMPORARY FREECOLOUR DIMLEVEL	255	255	0 ... 255	1 Byte	3	
REPORT FREECOLOUR DIMLEVEL	255	255	0 ... 255	1 Byte	3	
FREECOLOUR DIMLEVEL	?	keine Änderung	0 ... 255	1 Byte	3	
TEMPORARY RGBWAF CONTROL	255	255	0 ... 255	1 Byte	3	
REPORT RGBWAF CONTROL	255	255	0 ... 255	1 Byte	3	
<u>RGBWAF CONTROL</u> [▶ 386]	63	keine Änderung	0 ... 255	1 Byte	3	
<u>ASSIGNED COLOUR</u> [▶ 384]	0x0102 0304 0506	0x0102 0304 0506	0x0000 0000 0000 ... 0x0606 0606 0606	6 Bytes	3	nur Lesen MSB: Kanal 0 LSB: Kanal 5
<u>TEMPORARY COLOUR TYPE</u> [▶ 385]	255	255	0x10, 0x20, 0x40, 0x80, 0xFF	1 Byte	0, 1, 2, 3	
<u>REPORT COLOUR TYPE</u> [▶ 385]	255	255	0x10, 0x20, 0x40, 0x80, 0xFF	1 Byte	0, 1, 2, 3	

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	notwendige Farbdarstellung (1) [▶ 380]	Bemerkung
<u>SCENE 0–15 COLOUR TYPE</u> [▶ 385]	65535	65535	0x10, 0x20, 0x40, 0x80, 0xFF	16 Bytes	0, 1, 2, 3	nur Lesen
<u>SCENE 0–15 COLOUR VALUE</u>	65535	65535	0 ... 65535	32 Bytes ... 192 Bytes	0, 1, 2, 3	nur Lesen
<u>POWER ON COLOUR TYPE</u> [▶ 385]	herstellerabhängig	herstellerabhängig	0x10, 0x20, 0x40, 0x80, 0xFF	1 Byte	0, 1, 2, 3	nur Lesen
<u>POWER ON COLOUR VALUE</u>	herstellerabhängig	herstellerabhängig	0 ... 65535	2 Bytes ... 12 Bytes	0, 1, 2, 3	nur Lesen
<u>SYSTEM FAILURE COLOUR TYPE</u> [▶ 385]	herstellerabhängig	herstellerabhängig	0x10, 0x20, 0x40, 0x80, 0xFF	1 Byte	0, 1, 2, 3	nur Lesen
<u>SYSTEM FAILURE COLOUR VALUE</u>	herstellerabhängig	herstellerabhängig	0 ... 65535	2 Bytes ... 12 Bytes	0, 1, 2, 3	nur Lesen
<u>GEAR FEATURES/ STATUS</u> [▶ 385]	??00 0001b	??00 0001b	??00 0000b, ??00 0001b	1 Byte	0, 1, 2, 3	
<u>COLOUR STATUS</u> [▶ 385]	?	keine Änderung	0 ... 255	1 Byte	0, 1, 2, 3	
<u>COLOUR TYPE FEATURES</u> [▶ 386]	?	keine Änderung	0 ... 255	1 Byte	0, 1, 2, 3	nur Lesen

?: unbestimmt

Bei 1-Byte-Werten wird der Wert 255 auch als *MASK* bezeichnet.

Bei 2-Byte-Werten wird der Wert 65535 wird auch als *MASK* bezeichnet.

(1): Gibt an, welche Farbdarstellung das DALI-Vorschaltgerät unterstützen muss, damit es die entsprechende Variable enthält:

Wert	Beschreibung
0	xy-Koordinaten
1	Farbtemperatur Tc
2	Primär(farbe) N
3	RGBWAF

ASSIGNED COLOUR

In ASSIGNED COLOUR wird die Zuordnung zwischen Ausgangskanal und Farbe definiert. Jedes Byte enthält die Farbe des entsprechenden Kanals.

Mit dem Baustein `FB_DALIV2QueryAssignedColour()` [► 294] kann der Wert ausgelesen werden.

Wert	Beschreibung
0	keine Farbe zugeordnet
1	Rot
2	Grün
3	Blau
4	Weiß
5	Bernsteingelb
6	frei wählbare Farbe

COLOUR TYPE

COLOUR TYPE definiert die vom DALI-Vorschaltgerät unterstützten Farbdarstellungen.

Mit dem Baustein `FB_DALIV2QueryColourValue()` [► 298] können die Werte ausgelesen werden.

Wert	Beschreibung
0x10	xy-Koordinaten
0x20	Farbtemperatur Tc
0x40	Primär(farbe) N
0x80	RGBWAF
0xFF	keine Farbänderung

COLOUR STATUS

COLOUR TYPE enthält Informationen über den aktuellen Status des DALI-Vorschaltgerätes.

Mit dem Baustein `FB_DALIV2QueryColourStatus()` [► 296] können die Werte ausgelesen werden.

Bit	Beschreibung
0	xy-coordinate-Farbpunkt liegt außerhalb des gültigen Bereichs.
1	Farbtemperatur Tc liegt außerhalb des gültigen Bereichs.
2	automatische Kalibrierung ist aktiv.
3	automatische Kalibrierung war erfolgreich.
4	Farbdarstellung <i>xy-coordinate</i> aktiv.
5	Farbdarstellung <i>colour temperature Tc</i> aktiv.
6	Farbdarstellung <i>primary N</i> aktiv.
7	Farbdarstellung <i>RGBWAF</i> aktiv.

GEAR FEATURES/STATUS

GEAR FEATURES/STATUS enthält Informationen über den aktuellen Status des DALI-Vorschaltgerätes.

Mit dem Baustein `FB_DALIV2QueryGearFeaturesStatus()` [► 301] können die Werte ausgelesen und mit dem Baustein `FB_DALIV2StoreGearFeaturesStatus()` [► 317] beschrieben werden.

Bit	Beschreibung
0	automatische Aktivierung.
1 - 5	reserviert.
6	automatische Kalibrierung wird unterstützt.
7	Wiederherstellung der automatischen Kalibrierung wird unterstützt.

RGBWAF CONTROL

RGBWAF CONTROL enthält weitere Informationen über die Zuordnung zwischen Ausgangskanal und Farbe.

Mit dem Baustein `FB_DALIV2QueryColourValue()` [▶ 298] können die Werte ausgelesen werden.

Bit	Beschreibung
0	Ausgangskanal 0 / Rot
1	Ausgangskanal 1 / Grün
2	Ausgangskanal 2 / Blau
3	Ausgangskanal 3 / Weiß
4	Ausgangskanal 4 / Bernstein gelb
5	Ausgangskanal 5 / Frei wählbare Farbe
6 - 7	00 = Kanalsteuerung 01 = Farbsteuerung 10 = Standardisierte Farbsteuerung 11 = Reserviert

COLOUR TYPE FEATURES

COLOUR TYPE FEATURES enthält die vom DALI-Vorschaltgerät unterstützten Farbdarstellungen.

Mit dem Baustein `FB_DALIV2QueryColourTypeFeatures()` [▶ 297] können die Werte ausgelesen werden.

Bit	Beschreibung
0	Das Vorschaltgerät unterstützt die Farbdarstellung per xy-Koordinaten.
1	Das Vorschaltgerät unterstützt die Farbdarstellung per Farbtemperatur Tc.
2 - 4	Anzahl der vom Vorschaltgerät unterstützten Primärfarben. Ein Wert von 0 bedeutet, dass diese Farbdarstellung per Primärfarben nicht unterstützt wird.
5 - 7	Anzahl der vom Vorschaltgerät unterstützten RGBWAF-Kanäle. Ein Wert von 0 bedeutet, dass diese Farbdarstellung per RGBWAF nicht unterstützt wird.

5.2.5 Philips - Variablen

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
PAEC_ENABLE D	0	keine Änderung	0 ... 1	1 Byte	
CONTROL GEAR SEGMENT ADDRESS	0	keine Änderung	0 ... 255	1 Byte	
SEGMENT ADDRESS	0	0	0 ... 255	1 Byte	
UIC	herstellerabhängig	keine Änderung	00 00 00 00 00 00 00 00 ... FF FF FF FF FF FF FF FF	8 Byte	nur Lesen
LAMPTYPE	0	keine Änderung	0 ... 255	1 Byte	
HID LAMP LEVEL	???? ????	0	0 ... 255	1 Byte	
FADE UP	0	keine Änderung	0 ... 255	1 Byte	
FADE DOWN	0	keine Änderung	0 ... 255	1 Byte	
FAILURE STATUS	0	0	0 ... 255	1 Byte	

Name	Vorgabewert	Rücksetzwert	Gültigkeitsbereich	Größe	Bemerkung
OPERATION TIME	0	keine Änderung	0 ... 255	1 Byte	
OVERTEMPERATURE TIME	0	keine Änderung	0 ... 255	1 Byte	
CONTROL GEAR TEMPERATURE	0	0	0 ... 255	1 Byte	
OVERTEMPERATURE LEVEL	255	keine Änderung	0 ... 255	1 Byte	
TIMEOUT	0	keine Änderung	0 ... 255	1 Byte	
MAINS VOLTAGE	0	0	0 ... 255	1 Byte	
LAMP VOLTAGE	0	0	0 ... 255	1 Byte	

?: unbestimmt

5.3 Datentypen

5.3.1 E_DALIV2AddrType

```

TYPE E_DALIV2AddrType :
(
  eDALIV2AddrTypeShort      := 0,
  eDALIV2AddrTypeGroup      := 1,
  eDALIV2AddrTypeBroadcast  := 2,
  eDALIV2AddrTypeBroadcastUnaddr := 3      (* only DALI Rev. 2 *)
);
END_TYPE

```

5.3.2 E_DALIV2CommandPriority

```

TYPE E_DALIV2CommandPriority :
(
  eDALIV2CommandPriorityHigh := 0,
  eDALIV2CommandPriorityMiddle := 1,
  eDALIV2CommandPriorityLow := 2
);
END_TYPE

```

5.3.3 E_DALIV2ConfigurationCommands

```

TYPE E_DALIV2ConfigurationCommands :
(
  eDALIV2CommandDoNothing      := 0,
  eDALIV2CommandOff            := 1,
  eDALIV2CommandRecallMaxLevel := 2,
  eDALIV2CommandRecallMinLevel := 3,
  eDALIV2CommandGoToScene0     := 4,
  eDALIV2CommandGoToScene1     := 5,
  eDALIV2CommandGoToScene2     := 6,
  eDALIV2CommandGoToScene3     := 7,
  eDALIV2CommandGoToScene4     := 8,
  eDALIV2CommandGoToScene5     := 9,
  eDALIV2CommandGoToScene6     := 10,
  eDALIV2CommandGoToScene7     := 11,
  eDALIV2CommandGoToScene8     := 12,
  eDALIV2CommandGoToScene9     := 13,
  eDALIV2CommandGoToScene10    := 14,
  eDALIV2CommandGoToScene11    := 15,

```

```
eDALIV2CommandGoToScene12 := 16,
eDALIV2CommandGoToScene13 := 17,
eDALIV2CommandGoToScene14 := 18,
eDALIV2CommandGoToScene15 := 19
);
END_TYPE
```

5.3.4 E_DALIV2CurrentAddressingState

```
TYPE E_DALIV2CurrentAddressingState :
(
eDALIV2AddrStateIdle := 0,
eDALIV2AddrStateRemoveLamp := 1,
eDALIV2AddrStateReinsertLamp := 2,
eDALIV2AddrStateAddressingLamp := 3
);
END_TYPE
```

5.3.5 E_DALIV2DataFrameType

```
TYPE E_DALIV2DataFrameType :
(
eDALIV2DataFrameType16Bit := 1,
eDALIV2DataFrameType24Bit := 3,
eDALIV2DataFrameTypeOsram := 6
);
END_TYPE
```

5.3.6 E_DALIV2DimmingCurve

```
TYPE E_DALIV2DimmingCurve :
(
eDALIV2DimmingCurveLogarithmic := 0,
eDALIV2DimmingCurveLinear := 1
);
END_TYPE
```

5.3.7 E_DALIV2EventScheme

```
TYPE E_DALIV2EventScheme :
(
eDALIV2EventSchemeUnkown := -1,
eDALIV2EventSchemeInstance := 0,
(* (default) Instance addressing, using instance type and number. *)
eDALIV2EventSchemeDevice := 1,
(* Device addressing, using short address and instance type. *)
eDALIV2EventSchemeDeviceInstance := 2, (* Device/
instance addressing, using short address and instance number. *)
eDALIV2EventSchemeDeviceGroup := 3,
(* Device group addressing, using device group and instance type. *)
eDALIV2EventSchemeInstanceGroup := 4
(* Instance group addressing, using instance group and type. *)
);
END_TYPE
```

5.3.8 E_DALIV2InstAddrType

```
TYPE E_DALIV2InstAddrType :
(
eDALIV2InstAddrTypeNumber := 0, (* Instance number (0-31) *)
eDALIV2InstAddrTypeGroup := 1, (* Instance group (0-31) *)
eDALIV2InstAddrTypeType := 2, (* Instance type (0-31) *)
eDALIV2InstAddrTypeFeatureNumber := 3, (* Feature on instance number level (0-31) *)
eDALIV2InstAddrTypeFeatureGroup := 4, (* Feature on instance group level (0-31) *)
eDALIV2InstAddrTypeFeatureType := 5, (* Feature on instance type level (0-31) *)
eDALIV2InstAddrTypeFeatureBroadcast := 6, (* Feature on instance broadcast level *)
eDALIV2InstAddrTypeBroadcast := 7, (* Instance broadcast *)
eDALIV2InstAddrTypeFeatureDevice := 8, (* Feature on device level *)
);
```

```
eDALIV2InstAddrTypeDevice      := 9      (* Device *)
);
END_TYPE
```

5.3.9 E_DALIV2OperationMode

```
TYPE E_DALIV2OperationMode :
(
  eDALIV2OperationModeDALI      := 0,
  eDALIV2OperationModeDSI       := 1
);
END_TYPE
```

5.3.10 E_DALIV2PowerSupplyMode

```
TYPE E_DALIV2PowerSupplyMode :
(
  eDALIV2PowerSupplyModeOn      := 0,
  eDALIV2PowerSupplyModeOff     := 1,
  eDALIV2PowerSupplyModeAuto    := 2
);
END_TYPE
```

5.3.11 ST_DALIV2ChangeAddressList

```
TYPE ST_DALIV2ChangeAddressList :
STRUCT
  nOldAddress      : BYTE;
  nNewAddress      : BYTE;
  nRandomAddressHigh : BYTE;
  nRandomAddressMiddle : BYTE;
  nRandomAddressLow  : BYTE;
  nErrors          : DWORD;
END_STRUCT
END_TYPE
```

5.3.12 ST_DALIV2ControlTable

```
TYPE ST_DALIV2ControlTable :
STRUCT
  nActualValue      : UINT;
  nControlValue     : BYTE;
  nSwitchRange      : UINT;
END_STRUCT
END_TYPE
```

nActualValue: Gemessener Helligkeitswert.

nControlValue: Zugehöriger Umschaltpunkt/Stützpunkt für die Stellgröße. Gültiger Wertebereich: 0 oder *nMinLevelMasterDev ... nMaxLevelMasterDev*.

nSwitchRange: Schwellwert um die Stützstelle, bei der umgeschaltet wird. Der Eintrag "0" kennzeichnet den Anfang des nicht genutzten Bereiches der Tabelle.

5.3.13 ST_DALIV2DeviceSettings

```
TYPE ST_DALIV2DeviceSettings :
STRUCT
  nErrors          : DWORD;
  bPresent         : BOOL;
  nActualLevel     : BYTE;
  nPowerOnLevel    : BYTE;
  nSystemFailureLevel : BYTE;
  nMinLevel        : BYTE;
  nMaxLevel        : BYTE;
  nFadeRate        : BYTE;
  nFadeTime        : BYTE;
  nRandomAddress   : DWORD;
END_STRUCT
```

```
nGroups          : WORD;
nSceneLevels     : ARRAY [0..15] OF BYTE;
nStatus          : BYTE;
nMajorVersion   : BYTE;
nMinorVersion   : BYTE;
nDeviceType     : BYTE;
nPhysicalMinLevel : BYTE;
END_STRUCT
END_TYPE
```

5.3.14 ST_DALIV2DeviceSettingsType01

```
TYPE ST_DALIV2DeviceSettingsType01 :
STRUCT
  nErrors          : DWORD;
  bPresent         : BOOL;
  nBatteryCharge   : UINT;      (* 0..254, 255->Error *)
  tDurationTestResult : TIME;  (* 0..510 min *)
  tLampEmergencyTime : TIME;  (* 0..255 h *)
  tLampTotalOperationTime : TIME; (* 0..1024 h *)
  nEmergencyLevel  : BYTE;     (* 0..254 *)
  nEmergencyMinLevel : BYTE;  (* 0..254 *)
  nEmergencyMaxLevel : BYTE;  (* 0..254 *)
  tRatedDuration   : TIME;     (* 0..510 min *)
  nNextFunctionTest : UINT;    (* 0..255 *)
  nNextDurationTest : UINT;    (* 0..255 *)
  nFunctionTestInterval : UINT; (* 0..255 *)
  nDurationTestInterval : UINT; (* 0..255 *)
  nTestExecutionTimeout : UINT; (* 0..255 *)
  nProlongTime     : UINT;    (* 0..255 *)
  nEmergencyMode   : BYTE;
  nFeatures        : BYTE;
  nFailureStatus   : BYTE;
  nEmergencyStatus : BYTE;
END_STRUCT
END_TYPE
```

Folgende Variablen erfahren beim Auslesen eine Anpassung an ihren Ziel-Darstellungsbereich, also abweichend von der Auslese-Darstellung des DALI-Gerätes:

- tDurationTestResult
- tLampEmergencyTime
- tLampTotalOperationTime
- tRatedDuration

Bei den übrigen Variablen macht die Umrechnung entweder keinen Sinn (z.B. *nEmergencyLevel*) oder eine Darstellung ist nicht möglich, wie beispielsweise *nNextDurationTest* da der Bereich des Variablentyps TIME nicht ausreicht.

5.3.15 ST_DALIV2FileLogging

```
TYPE ST_DALIV2FileLogging :
STRUCT
  sTimestamp      : STRING(30);
  sController     : STRING(20);
  sLineName       : STRING(10);
  sAddress        : STRING(2);
  sDescription    : STRING(20);
  sLocation       : STRING(20);
  sTestDuration   : STRING(8);
  sResult         : STRING(240);
END_STRUCT
END_TYPE
```

5.3.16 ST_DALIV2InData

```
TYPE ST_DALIV2InData :
STRUCT
  nStatus        : BYTE;
```



```
nDummy : BYTE;
nData : WORD;
END_STRUCT
```

5.3.17 ST_DALIV2OutData

```
TYPE ST_DALIV2OutData :
STRUCT
nCtrl : BYTE;
nDummy : BYTE;
nData : WORD;
END_STRUCT
END_TYPE
```

5.3.18 ST_DALIV2SequenceTable

```
TYPE ST_DALIV2SequenceTable :
STRUCT
nTargetValue : BYTE;
tRampTime : TIME;
tProlongTime : TIME;
END_STRUCT
END_TYPE
```

nTargetValue: Zielwert.

tRampTime: Zeit zum Erreichen des Zielwertes.

tProlongTime: Verweilzeit auf dem Zielwert.

5.3.19 ST_DALIV2SwapShortAddressList

```
TYPE ST_DALIV2SwapShortAddressList :
STRUCT
bShortAddressValid : BOOL;
nNewShortAddress : BYTE;
END_STRUCT
END_TYPE
```

5.3.20 ST_KL6821InData

```
TYPE ST_KL6821InData :
STRUCT
nStatus : WORD;
arrData : ARRAY [0..3] OF BYTE;
END_STRUCT
```

5.3.21 ST_KL6821OutData

```
TYPE ST_KL6821OutData :
STRUCT
nCtrl : WORD;
arrData : ARRAY [0..3] OF BYTE;
END_STRUCT
END_TYPE
```

5.4 Fehlercodes


Wert (hex)	Wert (dez)	Beschreibung
0x0000	0	kein Fehler.
0x0001	1	keine Antwort von der DALI-Klemme.
0x0002	2	keine Antwort von dem DALI-Vorschaltgerät.
0x0003	3	Überlauf Kommunikationspuffer.

Wert (hex)	Wert (dez)	Beschreibung
0x0004	4	keine Antwort von dem Kommunikationsbaustein.
0x0005	5	DALI-Kollision auf dem Rückkanal (Backward Channel) erkannt: während der Übertragung eines DALI-Telegramms wurde eine Kollision mit den Sendedaten eines anderen DALI-Slaves erkannt.
0x0006	6	DALI-Kollision auf dem Hinkanal (Foreward Channel) erkannt: während der Übertragung eines DALI-Telegramms wurde eine Kollision mit den Sendedaten eines anderen DALI-Masters erkannt. Der Fehler tritt ebenfalls auf, sobald die 24 V an den Powerkontakten der KL6811 fehlen.
0x0007	7	Bei Nutzung des internen DALI-Netzteils: Überlastung des internen DALI-Netzteils (Busunterspannung).
0x0008	8	Parameter <i>eCommandPriority</i> liegt außerhalb des gültigen Bereichs.
0x0009	9	Parameter <i>eAddrType</i> liegt außerhalb des gültigen Bereichs.
0x000A	10	Parameter <i>nAddr</i> ist eine Kurzadresse und liegt außerhalb des gültigen Bereichs.
0x000B	11	Parameter <i>nAddr</i> ist eine Gruppenadresse und liegt außerhalb des gültigen Bereichs.
0x000C	12	Parameter <i>nGroup</i> liegt außerhalb des gültigen Bereichs.
0x000D	13	Parameter <i>nScene</i> liegt außerhalb des gültigen Bereichs.
0x000E	14	Parameter <i>nStartWithShortAddress</i> liegt außerhalb des gültigen Bereichs.
0x000F	15	keine weiteren freien Kurzadressen.
0x0010	16	Parameter <i>nNewShortAddress</i> liegt außerhalb des gültigen Bereichs.
0x0011	17	Parameter <i>nShortAddress01</i> liegt außerhalb des gültigen Bereichs.
0x0012	18	Parameter <i>nShortAddress02</i> liegt außerhalb des gültigen Bereichs.
0x0013	19	Parameter <i>nFreeShortAddress</i> liegt außerhalb des gültigen Bereichs.
0x0014	20	Die Kurzadresse im Parameter <i>nFreeShortAddress</i> ist innerhalb der DALI-Linie belegt.
0x0015	21	Parameter <i>arrSwapShortAddressList</i> enthält ungültige Werte.
0x0016	22	Parameter <i>nHysteresis</i> liegt außerhalb des gültigen Bereichs.
0x0017	23	Parameter <i>nProlongValue</i> ist außerhalb des gültigen Bereiches.
0x0018	24	Lampenwert des Master-Device ist zu lang 255 -> möglicher Defekt.
0x0019	25	Parameter <i>nEndLevel</i> ist außerhalb des gültigen Bereiches.
0x001A	26	Zielwert <i>nEndLevel</i> ist nach doppelter Rampenzeit noch nicht erreicht worden.
0x001B	27	FB_DALIV2LightControl() [► 69] : In der Wertetabelle <i>arrControlTable</i> existieren Sollwerte (<i>nSetpoint</i>), die außerhalb des gültigen DALI-Bereiches liegen (0..254).
0x001C	28	FB_DALIV2LightControl() [► 69] : Die Switchrange (<i>nSwitchRange</i>) im ersten oder zweiten Element der Wertetabelle <i>arrControlTable</i> ist 0. Damit wird davon ausgegangen, dass die Tabelle keinen oder nur einen Wertesatz hat.
0x001D	29	FB_DALIV2LightControl() [► 69] : 2 benachbarte Eingangswerte <i>nActualValue</i> in der Wertetabelle <i>arrControlTable</i> liegen zu dicht beieinander d.h. im Schaltbereich des anderen.
0x001E	30	FB_DALIV2LightControl() [► 69] : Ein Sollwert (<i>nSetpoint</i>) der Wertetabelle <i>arrControlTable</i> liegt außerhalb des gültigen Bereiches. Wird nur nach <i>bStart</i> abgefragt.
0x001F	31	Interne Statusabfrage des Master-Gerätes liefert nach der Dimmzeit noch zu lange "Stufung aktiv". Siehe STATUS INFORMATION
0x0020	32	Parameter <i>nPresenceValue</i> ist außerhalb des gültigen Bereiches.

Wert (hex)	Wert (dez)	Beschreibung
0x0021	33	Zeitüberschreitung bei der internen Adressierung (siehe auch <code>FB_DALIV2AddressingIntRandomAddressing()</code> [► 42]). Die Klemme hat, nach dem Starten der internen Adressierung, keine Antwort zurückgeliefert.
0x0022	34	Bei der internen Adressierung (siehe auch <code>FB_DALIV2AddressingIntRandomAddressing()</code> [► 42]) hat die Klemme einen Fehler zurückgeliefert.
0x0023	35	Mindestens ein Test läuft gerade oder steht automatisch bevor.
0x0024	36	Das Gerät befindet sich im Automatik-Test-Modus.
0x0025	37	Die Batterien des Notlicht-Gerätes sind für einen Dauertest nicht vollständig aufgeladen.
0x0026	38	Test unterbrochen - Kein gültiger Emergency-Modus / Emergency-Status.
0x0027	39	Test unterbrochen: Test-modus wurde nicht erreicht.
0x0028	40	Test wegen Timeout-Überschreitung unterbrochen.
0x0029	41	Fehler bei der Abarbeitung eines DALI-Kommandos.
0x002A	42	Fehler beim Schreiben in die Log-Datei.
0x002B	43	Gerät ist nicht im "normal-mode".
0x002C	44	<code>FB_DALIV2Sequencer()</code> [► 74]: Der Startindex <i>nStartIndex</i> ist außerhalb des gültigen Bereiches (1..50).
0x002D	45	<code>FB_DALIV2Sequencer()</code> [► 74]: Der Startindex <i>nStartIndex</i> verweist auf eine Stelle, die ihrerseits ein Sequenzende (Nulleinträge) markiert.
0x002E	46	<code>FB_DALIV2ChangeAddressList()</code> [► 47]: Die Änderungsliste <i>arrChangeAddressList</i> ist leer.
0x002F	47	<code>FB_DALIV2ChangeAddressList()</code> [► 47]: Die Änderungsliste <i>arrChangeAddressList</i> enthält einen ungültigen Kurzadressen-Eintrag (>63).
0x0030	48	<code>FB_DALIV2ChangeAddressList()</code> [► 47]: Die Änderungsliste <i>arrChangeAddressList</i> enthält einen doppelten Listeneintrag bei den Kurzadressen.
0x0031	49	<code>FB_DALIV2ChangeAddressList()</code> [► 47]: Die Änderungsliste <i>arrChangeAddressList</i> enthält einen Eintrag für eine neue Kurzadresse, die jedoch schon für ein Gerät vergeben ist, welches nicht von den Änderungen betroffen ist. Die Adressen wurden zurück geändert.
0x0032	50	<code>FB_KL6811Config()</code> [► 102]: Während der Konfiguration der Klemme ist ein Fehler aufgetreten.
0x0033	51	<code>FB_KL6811Config()</code> [► 102]: Parameter <i>eOperationMode</i> liegt außerhalb des gültigen Bereichs.
0x0034	52	Die Konstante <code>DALI_MESSAGE_QUEUE_ENTRIES</code> liegt außerhalb des gültigen Bereichs (2-250) .
0x0035	53	Die Konstante <code>DALI_RESPONSE_TABLE_ENTRIES</code> liegt außerhalb des gültigen Bereichs (2-250).
0x0036	54	Die Konstante <code>DALI_EVENT_TABLE_ENTRIES</code> liegt außerhalb des gültigen Bereichs (2-250).
0x0037	55	Bei Nutzung des internen DALI-Netzteils: Fehler Netzteil erkannt.
0x0038	56	Prozessabbild wurde durch die Eingänge DI1 oder DI2 der Klemme deaktiviert.
0x0039	57	Parameter <i>eInstAddrType</i> liegt außerhalb des gültigen Bereichs.
0x003A	58	Parameter <i>eDataFrameType</i> liegt außerhalb des gültigen Bereichs.
0x003B	59	DSI wird von der Busklemme nicht unterstützt.
0x003C	60	Parameter <i>nEventPriority</i> liegt außerhalb des gültigen Bereichs.
0x003D	61	Parameter <i>nGroup</i> liegt außerhalb des gültigen Bereichs.

Wert (hex)	Wert (dez)	Beschreibung
0x003E	62	Parameter <i>nInstanceGroup</i> liegt außerhalb des gültigen Bereichs.
0x003F	63	Parameter <i>eEventScheme</i> liegt außerhalb des gültigen Bereichs.
0x0040	64	Parameter <i>eEventFilter</i> liegt außerhalb des gültigen Bereichs.
0x0041	65	Parameter <i>nInstAddr</i> liegt außerhalb des gültigen Bereichs.
0x0042	66	Parameter <i>ePowerSupplyMode</i> liegt außerhalb des gültigen Bereichs.
0x0043	67	Parameter <i>eCommandKBusWatchdog</i> liegt außerhalb des gültigen Bereichs.
0x0044	68	Parameter <i>eCommandDI1RisingEdge</i> liegt außerhalb des gültigen Bereichs.
0x0045	69	Parameter <i>eCommandDI1FallingEdge</i> liegt außerhalb des gültigen Bereichs.
0x0046	70	Parameter <i>eCommandDI2RisingEdge</i> liegt außerhalb des gültigen Bereichs.
0x0047	71	Parameter <i>eCommandDI2FallingEdge</i> liegt außerhalb des gültigen Bereichs.
0x0048	72	Bei der internen Adressierung (siehe auch FB_DALIV2AddressingIntRandomAddressing() [► 42]) hat die Klemme erkannt, dass keine freie Kurzadresse vorhanden ist.
0x0049	73	Bei der internen Adressierung (siehe auch FB_DALIV2AddressingIntRandomAddressing() [► 42]) hat die Klemme erkannt, dass mehrere Geräte die gleiche Langadresse besitzen.
0x004A	74	Die interne Adressierung (siehe auch FB_DALIV2AddressingIntRandomAddressing() [► 42]) ist 3x fehlgeschlagen.
0x004B	75	Der Kommunikationspuffer zum Versenden der DALI-Kommandos wurde länger blockiert als erlaubt.
0x004C	76	Die Konstante <code>DALIV2_TIMEOUT_LOCK_MESSAGE_QUEUE</code> liegt außerhalb des gültigen Bereichs (0-15s).
0x004D	77	Bei der internen Adressierung (siehe auch FB_DALIV2AddressingIntRandomAddressing() [► 42]) hat die Klemme einen Kurzschluss auf den Bus erkannt.
0x004E	78	Kurzschluss auf dem DALI-Bus erkannt.

Sehen Sie dazu auch

 [Vorschaltgeräte - Variablen \[► 372\]](#)

6 Support und Service

Beckhoff und seine weltweiten Partnerfirmen bieten einen umfassenden Support und Service, der eine schnelle und kompetente Unterstützung bei allen Fragen zu Beckhoff Produkten und Systemlösungen zur Verfügung stellt.

Beckhoff Niederlassungen und Vertretungen

Wenden Sie sich bitte an Ihre Beckhoff Niederlassung oder Ihre Vertretung für den lokalen Support und Service zu Beckhoff Produkten!

Die Adressen der weltweiten Beckhoff Niederlassungen und Vertretungen entnehmen Sie bitte unseren Internetseiten: <https://www.beckhoff.de>

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

Beckhoff Support

Der Support bietet Ihnen einen umfangreichen technischen Support, der Sie nicht nur bei dem Einsatz einzelner Beckhoff Produkte, sondern auch bei weiteren umfassenden Dienstleistungen unterstützt:

- Support
- Planung, Programmierung und Inbetriebnahme komplexer Automatisierungssysteme
- umfangreiches Schulungsprogramm für Beckhoff Systemkomponenten

Hotline: +49(0)5246 963 157
Fax: +49(0)5246 963 9157
E-Mail: support@beckhoff.com

Beckhoff Service

Das Beckhoff Service-Center unterstützt Sie rund um den After-Sales-Service:

- Vor-Ort-Service
- Reparaturservice
- Ersatzteilservice
- Hotline-Service

Hotline: +49(0)5246 963 460
Fax: +49(0)5246 963 479
E-Mail: service@beckhoff.com

Beckhoff Firmenzentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

Telefon: +49(0)5246 963 0
Fax: +49(0)5246 963 198
E-Mail: info@beckhoff.com
Internet: <https://www.beckhoff.de>

Mehr Informationen:
www.beckhoff.de/tx1200

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Deutschland
Telefon: +49 5246 9630
info@beckhoff.de
www.beckhoff.de

