

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Digital addressable lighting interface –
Part 102: General requirements – Control gear**

**Interface d'éclairage adressable numérique –
Partie 102: Exigences générales – Appareillages de commande**



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2014 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

IEC Central Office
3, rue de Varembe
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

IEC Catalogue - webstore.iec.ch/catalogue

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in 14 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

More than 55 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.

A propos de l'IEC

La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC

Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Catalogue IEC - webstore.iec.ch/catalogue

Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

Recherche de publications IEC - www.iec.ch/searchpub

La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études,...). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans 14 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary

Plus de 55 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

Service Clients - webstore.iec.ch/csc

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: csc@iec.ch.

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Digital addressable lighting interface –
Part 102: General requirements – Control gear**

**Interface d'éclairage adressable numérique –
Partie 102: Exigences générales – Appareillages de commande**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX

XH

ICS 29.140, 29.140.50

ISBN 978-2-8322-1906-5

**Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**

CONTENTS

FOREWORD.....	12
INTRODUCTION.....	14
1 Scope.....	15
2 Normative references	15
3 Terms and definitions	15
4 General	18
4.1 General.....	18
4.2 Version number	18
5 Electrical specification	18
6 Interface power supply	18
7 Transmission protocol structure.....	19
7.1 General.....	19
7.2 16 bit forward frame encoding.....	19
7.2.1 General	19
7.2.2 Address byte	19
7.2.3 Opcode byte	19
8 Timing	19
9 Method of operation.....	20
9.1 General.....	20
9.2 Control gear.....	20
9.3 Dimming curve.....	20
9.4 Calculating “ <i>targetLevel</i> ”	23
9.5 Fading	23
9.5.1 General	23
9.5.2 Fade time	24
9.5.3 Fade rate.....	25
9.5.4 Extended fade time.....	26
9.5.5 Using the fade time.....	28
9.5.6 Using the fade rate	28
9.5.7 Behaviour during a fade.....	28
9.5.8 Behaviour during startup.....	28
9.5.9 Stopping a fade	28
9.6 Min and max level.....	29
9.7 Commands	29
9.7.1 General	29
9.7.2 Level instructions without fade.....	30
9.7.3 Level instructions initiating a fade.....	30
9.7.4 Configuration instructions	30
9.7.5 Queries	30
9.7.6 Special commands	31
9.7.7 Application extended commands.....	31
9.8 Command iterations.....	31
9.8.1 General	31
9.8.2 Command iteration of “UP” and “DOWN” commands.....	31
9.8.3 DAPC SEQUENCE (deprecated)	31

9.9	Modes of operation	32
9.9.1	General	32
9.9.2	Operating mode 0x00: standard mode	32
9.9.3	Operating mode 0x01 to 0x7F: reserved	32
9.9.4	Operating mode 0x80 to 0xFF: manufacturer specific modes	32
9.10	Memory banks	33
9.10.1	General	33
9.10.2	Memory map.....	33
9.10.3	Selecting a memory bank location	34
9.10.4	Memory bank reading	34
9.10.5	Memory bank writing.....	34
9.10.6	Memory bank 0.....	35
9.10.7	Memory bank 1.....	37
9.10.8	Manufacturer specific memory banks.....	39
9.10.9	Reserved memory banks	39
9.11	Reset.....	39
9.11.1	Reset operation	39
9.11.2	Reset memory bank operation	39
9.12	System failure.....	39
9.13	Power on	40
9.14	Assigning short addresses	41
9.14.1	General	41
9.14.2	Random address allocation.....	41
9.14.3	Identification of a device	42
9.14.4	Direct address allocation	43
9.15	Failure state behaviour	43
9.16	Status information.....	43
9.16.1	General	43
9.16.2	Bit 0: Control gear failure.....	43
9.16.3	Bit 1: Lamp failure	44
9.16.4	Bit 2: Lamp on	44
9.16.5	Bit 3: Limit error.....	44
9.16.6	Bit 4: Fade running	44
9.16.7	Bit 5: Reset state	44
9.16.8	Bit 6: Missing short address.....	44
9.16.9	Bit 7: Power cycle seen	45
9.17	Non-volatile memory	45
9.18	Device types and features.....	45
9.19	Using scenes	46
10	Declaration of variables	47
11	Definition of commands	49
11.1	General.....	49
11.2	Overview Sheets.....	49
11.3	Level instructions.....	55
11.3.1	DAPC (<i>level</i>)	55
11.3.2	OFF.....	55
11.3.3	UP.....	55
11.3.4	DOWN.....	55
11.3.5	STEP UP	55

11.3.6	STEP DOWN	56
11.3.7	RECALL MAX LEVEL	56
11.3.8	RECALL MIN LEVEL	56
11.3.9	STEP DOWN AND OFF	57
11.3.10	ON AND STEP UP	57
11.3.11	ENABLE DAPC SEQUENCE	57
11.3.12	GO TO LAST ACTIVE LEVEL	58
11.3.13	GO TO SCENE (<i>sceneNumber</i>)	58
11.4	Configuration instructions	58
11.4.1	General	58
11.4.2	RESET	58
11.4.3	STORE ACTUAL LEVEL IN DTR0	58
11.4.4	SAVE PERSISTENT VARIABLES	58
11.4.5	SET OPERATING MODE (<i>DTR0</i>)	59
11.4.6	RESET MEMORY BANK (<i>DTR0</i>)	59
11.4.7	IDENTIFY DEVICE	59
11.4.8	SET MAX LEVEL (<i>DTR0</i>)	60
11.4.9	SET MIN LEVEL (<i>DTR0</i>)	60
11.4.10	SET SYSTEM FAILURE LEVEL (<i>DTR0</i>)	60
11.4.11	SET POWER ON LEVEL (<i>DTR0</i>)	61
11.4.12	SET FADE TIME (<i>DTR0</i>)	61
11.4.13	SET FADE RATE (<i>DTR0</i>)	61
11.4.14	SET EXTENDED FADE TIME (<i>DTR0</i>)	61
11.4.15	SET SCENE (<i>DTR0</i> , <i>sceneX</i>)	62
11.4.16	REMOVE FROM SCENE (<i>sceneX</i>)	62
11.4.17	ADD TO GROUP (<i>group</i>)	62
11.4.18	REMOVE FROM GROUP (<i>group</i>)	62
11.4.19	SET SHORT ADDRESS (<i>DTR0</i>)	62
11.4.20	ENABLE WRITE MEMORY	63
11.5	Queries	63
11.5.1	General	63
11.5.2	QUERY STATUS	63
11.5.3	QUERY CONTROL GEAR PRESENT	63
11.5.4	QUERY CONTROL GEAR FAILURE	63
11.5.5	QUERY LAMP FAILURE	63
11.5.6	QUERY LAMP POWER ON	63
11.5.7	QUERY LIMIT ERROR	63
11.5.8	QUERY RESET STATE	63
11.5.9	QUERY MISSING SHORT ADDRESS	63
11.5.10	QUERY VERSION NUMBER	64
11.5.11	QUERY CONTENT DTR0	64
11.5.12	QUERY DEVICE TYPE	64
11.5.13	QUERY NEXT DEVICE TYPE	64
11.5.14	QUERY PHYSICAL MINIMUM	64
11.5.15	QUERY POWER FAILURE	64
11.5.16	QUERY CONTENT DTR1	64
11.5.17	QUERY CONTENT DTR2	64
11.5.18	QUERY OPERATING MODE	65
11.5.19	QUERY LIGHT SOURCE TYPE	

11.5.20	QUERY ACTUAL LEVEL	65
11.5.21	QUERY MAX LEVEL.....	65
11.5.22	QUERY MIN LEVEL.....	66
11.5.23	QUERY POWER ON LEVEL	66
11.5.24	QUERY SYSTEM FAILURE LEVEL	66
11.5.25	QUERY FADE TIME/FADE RATE	66
11.5.26	QUERY EXTENDED FADE TIME.....	66
11.5.27	QUERY MANUFACTURER SPECIFIC MODE	66
11.5.28	QUERY SCENE LEVEL (<i>sceneX</i>).....	66
11.5.29	QUERY GROUPS 0-7	66
11.5.30	QUERY GROUPS 8-15	66
11.5.31	QUERY RANDOM ADDRESS (H)	67
11.5.32	QUERY RANDOM ADDRESS (M).....	67
11.5.33	QUERY RANDOM ADDRESS (L).....	67
11.5.34	READ MEMORY LOCATION (<i>DTR1</i> , <i>DTR0</i>).....	67
11.6	Application extended commands.....	67
11.6.1	General	67
11.6.2	QUERY EXTENDED VERSION NUMBER	67
11.7	Special commands.....	68
11.7.1	General	68
11.7.2	TERMINATE.....	68
11.7.3	DTR0 (<i>data</i>)	68
11.7.4	INITIALISE (<i>device</i>).....	68
11.7.5	RANDOMISE	68
11.7.6	COMPARE	69
11.7.7	WITHDRAW.....	69
11.7.8	SEARCHADDRH (<i>data</i>).....	69
11.7.9	SEARCHADDRM (<i>data</i>)	69
11.7.10	SEARCHADDRL (<i>data</i>)	69
11.7.11	PROGRAM SHORT ADDRESS (<i>data</i>)	70
11.7.12	VERIFY SHORT ADDRESS (<i>data</i>)	70
11.7.13	QUERY SHORT ADDRESS	70
11.7.14	ENABLE DEVICE TYPE (<i>data</i>).....	70
11.7.15	DTR1 (<i>data</i>)	71
11.7.16	DTR2 (<i>data</i>)	71
11.7.17	WRITE MEMORY LOCATION (<i>DTR1</i> , <i>DTR0</i> , <i>data</i>)	71
11.7.18	WRITE MEMORY LOCATION – NO REPLY (<i>DTR1</i> , <i>DTR0</i> , <i>data</i>)	71
11.7.19	PING	72
12	Test procedures	72
12.1	General notes on test.....	72
12.1.1	Abbreviations.....	72
12.1.2	Test execution.....	72
12.1.3	Data transmission.....	72
12.1.4	Test setup	73
12.1.5	Test output	73
12.1.6	Fade time measurements based on light output	73
12.1.7	Description of test scheme for fast fade times on PWM dimmer	74
12.1.8	Test notation	75
12.1.9	Test execution limitation	75

12.1.10	Test results	75
12.1.11	Exception handling	75
12.1.12	Unexpected answer	75
12.2	Preamble	77
12.2.1	Test preamble	77
12.3	Physical operational parameters	94
12.3.1	Polarity test	94
12.3.2	Maximum and minimum system voltage	94
12.3.3	Overvoltage protection test	95
12.3.4	Current rating test	96
12.3.5	Transmitter voltages	97
12.3.6	Transmitter rising and falling edges	99
12.3.7	Transmitter bit timing	101
12.3.8	Transmitter frame timing	103
12.3.9	Receiver start-up behavior	104
12.3.10	Receiver threshold	105
12.3.11	Receiver bit timing	106
12.3.12	Extended receiver bit timing	110
12.3.13	Receiver forward frame violation	112
12.3.14	Receiver settling timing	112
12.3.15	Receiver frame timing FF-FF send twice	113
12.4	Configuration instructions	115
12.4.1	RESET	115
12.4.2	RESET: timeout / command in-between	119
12.4.3	Send-twice timeout	121
12.4.4	Commands in-between	129
12.4.5	STORE ACTUAL LEVEL IN DTR0	135
12.4.6	SAVE PERSISTENT VARIABLES	136
12.4.7	SET OPERATING MODE	136
12.4.8	SET MAX LEVEL	137
12.4.9	SET MIN LEVEL	137
12.4.10	SET SYSTEM FAILURE LEVEL	138
12.4.11	SET POWER ON LEVEL	139
12.4.12	SET FADE TIME	143
12.4.13	SET FADE RATE	143
12.4.14	SET SCENE / REMOVE FROM SCENE	144
12.4.15	ADD TO GROUP / REMOVE FROM GROUP	145
12.4.16	SET SHORT ADDRESS	146
12.4.17	SET EXTENDED FADE TIME	147
12.4.18	Reset/Power-on values	148
12.4.19	DTR0 / DTR1 / DTR2	154
12.5	Memory banks	154
12.5.1	READ MEMORY LOCATION on Memory Bank 0	154
12.5.2	READ MEMORY LOCATION on Memory Bank 1	159
12.5.3	READ MEMORY LOCATION on other Memory Banks	161
12.5.4	Memory bank writing	163
12.5.5	ENABLE WRITE MEMORY: writeEnableState	168
12.5.6	ENABLE WRITE MEMORY: timeout / command in-between	171
12.5.7	RESET MEMORY BANK: timeout / command in-between	171

12.5.8	RESET MEMORY BANK	175
12.6	Level instructions	176
12.6.1	Level instructions: Basic behaviour	176
12.6.2	FADE TIME: possible values	181
12.6.3	FADE TIME: transitions	182
12.6.4	FADE TIME: fading to 0	184
12.6.5	FADE TIME: small steps fading	186
12.6.6	FADE TIME: extended fade time	189
12.6.7	FADE RATE: possible values	191
12.6.8	FADE RATE: transitions	194
12.6.9	FADE RATE: extended fade time	195
12.6.10	FADE TIME/FADE RATE: stop fading by setting MIN/MAX levels	196
12.6.11	FADE TIME/FADE RATE: stop fading	201
12.6.12	FADE TIME/FADE RATE: stop fading when a command is sent, check timing	204
12.6.13	FADE TIME/FADE RATE: stop fading during startup	210
12.6.14	Level instructions: combined instructions	211
12.6.15	Power On Level - System Failure Level combined	213
12.6.16	ENABLE DAPC SEQUENCE	218
12.6.17	GO TO LAST ACTIVE LEVEL	219
12.6.18	GO TO SCENE	221
12.6.19	Power on: level control commands	222
12.6.20	Logarithmic dimming curve	224
12.6.21	Dimming curve: DAPC	225
12.6.22	Dimming curve: UP / DOWN	226
12.6.23	Dimming curve: STEP UP / STEP DOWN	227
12.6.24	FADE TIME/EXTENDED FADE TIME: light output behaviour	228
12.6.25	EXTENDED FADE TIME: light output behaviour	229
12.6.26	Behaviour during a fade	230
12.7	Special commands	232
12.7.1	INITIALISE – timer	232
12.7.2	TERMINATE	234
12.7.3	INITIALISE - device addressing	234
12.7.4	RANDOMISE	235
12.7.5	COMPARE	236
12.7.6	WITHDRAW	237
12.7.7	SEARCHADDRH / SEARCHADDRM / SEARCHADDRL	238
12.7.8	PROGRAM SHORT ADDRESS	239
12.7.9	VERIFY SHORT ADDRESS	241
12.7.10	QUERY SHORT ADDRESS	242
12.7.11	IDENTIFY DEVICE	244
12.7.12	IDENTIFY DEVICE THROUGH RECALL MIN/MAX LEVEL	246
12.8	Queries and reserved commands	250
12.8.1	QUERY STATUS - lampFailure/lampOn	250
12.8.2	QUERY STATUS - lampOn	253
12.8.3	QUERY STATUS - limitError/lampOn	254
12.8.4	QUERY STATUS - powerCycleSeen	258
12.8.5	QUERY CONTROL GEAR PRESENT	259
12.8.6	QUERY VERSION NUMBER	259

12.8.7	PING	260
12.8.8	Broadcast unaddressed	260
12.8.9	Reserved commands: standard commands	262
12.8.10	Reserved commands: special commands	262
12.8.11	Application extended commands	263
12.8.12	Not supported device types	264
12.8.13	Removed functionality	264
12.9	Cross contamination	266
12.9.1	DTR0	266
12.9.2	NVM variables	266
12.9.3	Random address generation	268
12.9.4	Addressing 1	269
12.9.5	Addressing 2	270
12.9.6	Addressing 3	272
12.10	General subsequences	273
12.10.1	GetVersionNumber	273
12.10.2	GetExtendedVersionNumber	273
12.10.3	GetSupportedDeviceTypes	274
12.10.4	GetSupportedLightSources	274
12.10.5	WaitForPowerOnPhaseToFinish	276
12.10.6	WaitForLampOn	276
12.10.7	WaitForLampOnAddressed	277
12.10.8	WaitForLampLevel	277
12.10.9	WaitForFadeToFinish	277
12.10.10	SetShortAddress	278
12.10.11	GetRandomAddress	278
12.10.12	GetLimitedRandomAddress	279
12.10.13	SetSearchAddress	279
12.10.14	ReadMemBankMultibyteLocation	279
12.10.15	FindImplementedMemoryBank	280
12.10.16	FindAllImplementedMemoryBanks	280
12.10.17	GetNumberOfLogicalUnits	280
12.10.18	GetIndexOfLogicalUnit	281
12.10.19	ConnectLamps	281
12.10.20	DisconnectLamps	281
12.10.21	PowerCycle	282
12.10.22	PowerCycleAndWaitForBusPower	282
12.10.23	PowerCycleAndWaitForDecoder	283
Annex A (informative)	Examples of algorithms	284
A.1	Random address allocation	284
A.2	One single control gear connected to the control device	284
A.3	Using application extended commands	285
Annex B (normative)	High resolution dimmer	286
Bibliography		288
Figure 1 – IEC 62386 graphical overview		14
Figure 2 – Control gear directly operating a light source		20
Figure 3 – Dimming curve		21

Figure 4 – Level over time, fading up and down	24
Figure 5 – Timing and response when receiving a command iteration	31
Figure 6 – Fading from MIN LEVEL to MAX LEVEL	74
Figure 7 – Fading from MAX LEVEL to off	74
Figure 8 – Normal fading for a PWM dimmer	74
Figure 9 – Fading from MAX LEVEL to off for a PWM dimmer	75
Figure 10 – Current rating test	97
Figure B.1 – Level behaviour in cases of off-grid starting points	287
Table 1 – 16-bit command frame encoding	19
Table 2 – Dimming curve tolerance (% , rounded to two decimals)	21
Table 3 – Dimming curve	22
Table 4 – Fade times	25
Table 5 – Fade rates	26
Table 6 – Extended fade time - base value	27
Table 7 – Extended fade time - multiplier	27
Table 8 – Basic memory map of memory banks	33
Table 9 – Memory map of memory bank 0	36
Table 10 – Memory map of memory bank 1	38
Table 11 – Power on timing	41
Table 12 – Control gear status	43
Table 13 – Scenes	46
Table 14 – Declaration of variables	47
Table 15 – Standard commands	49
Table 16 – Special commands	53
Table 17 – Light source type encoding	65
Table 18 – Device addressing with “INITIALISE”	68
Table 19 – Unexpected outcome	76
Table 20 – Parameters for test sequence CheckFactoryDefault102	82
Table 21 – Parameters for test sequence CheckFactoryDefault201	86
Table 22 – Parameters for test sequence CheckFactoryDefault202	87
Table 23 – Parameters for test sequence CheckFactoryDefault203	88
Table 24 – Parameters for test sequence CheckFactoryDefault204	88
Table 25 – Parameters for test sequence CheckFactoryDefault205	89
Table 26 – Parameters for test sequence CheckFactoryDefault206	89
Table 27 – Parameters for test sequence CheckFactoryDefault207	90
Table 28 – Parameters for test sequence CheckFactoryDefault208	90
Table 29 – Parameters for test sequence CheckFactoryDefault209	91
Table 30 – Parameters for test sequence Maximum and minimum system voltage	95
Table 31 – Parameters for test sequence Transmitter voltages	99
Table 32 – Parameters for test sequence Transmitter rising and falling edges	100
Table 33 – Parameters for test sequence Transmitter rising and falling edges	101
Table 34 – Parameters for test sequence Transmitter bit timing	102

Table 35 – Parameters for test sequence Receiver frame timing	104
Table 36 – Parameters for test sequence Receiver start-up behavior	105
Table 37 – Parameters for test sequence Receiver bit timing	107
Table 38 – Parameters for test sequence Extended receiver bit timing	111
Table 39 – Parameters for test sequence Receiver frame violation and recovering after frame size violation	112
Table 40 – Parameters for test sequence Receiver frame timing	113
Table 41 – Parameters for test sequence RESET	117
Table 42 – Parameters for test sequence Send twice timeout	122
Table 43 – Parameters for test sequence Commands in-between	131
Table 44 – Parameters for test sequence SET MAX LEVEL	137
Table 45 – Parameters for test sequence SET MIN LEVEL	138
Table 46 – Parameters for test sequence SET SYSTEM FAILURE LEVEL	139
Table 47 – Parameters for test sequence SET POWER ON LEVEL	142
Table 48 – Parameters for test sequence SET FADE TIME	143
Table 49 – Parameters for test sequence SET FADE RATE	144
Table 50 – Parameters for test sequence SET SCENE / REMOVE FROM SCENE	145
Table 51 – Parameters for test sequence ADD TO GROUP / REMOVE FROM GROUP	146
Table 52 – Parameters for test sequence SET SHORT ADDRESS	147
Table 53 – Parameters for test sequence SET EXTENDED FADE TIME	148
Table 54 – Parameters for test sequence Reset/Power-on values	151
Table 55 – Parameters for test sequence DTR0 / DTR1 / DTR2	154
Table 56 – Parameters for test sequence READ MEMORY LOCATION on Memory Bank 0	158
Table 57 – Parameters for test sequence READ MEMORY LOCATION on Memory Bank 1	161
Table 58 – Parameters for test sequence Memory bank writing	166
Table 59 – Parameters for test sequence ENABLE WRITE MEMORY: writeEnableState	169
Table 60 – Parameters for test sequence ENABLE WRITE MEMORY: timeout / command in-between	172
Table 61 – Parameters for test sequence RESET MEMORY BANK: timeout / command in-between	174
Table 62 – Parameters for test sequence RESET MEMORY BANK	176
Table 63 – Parameters for test sequence Level instructions: Basic behaviour	178
Table 64 – Parameters for test sequence FADE TIME: possible values	181
Table 65 – Parameters for test sequence FADE TIME: transitions	184
Table 66 – Parameters for test sequence FADE TIME: fading to 0	186
Table 67 – Parameters for test sequence FADE TIME: small steps fading	189
Table 68 – Parameters for test sequence FADE TIME: extended fade time	190
Table 69 – Parameters for test sequence FADE RATE: possible values	193
Table 70 – Parameters for test sequence FADE RATE: possible values	194
Table 71 – Parameters for test sequence FADE RATE: transitions	195
Table 72 – Parameters for test sequence FADE RATE: extended fade time	196
Table 73 – Parameters for test sequence FADE TIME/FADE RATE: stop fading by setting MIN/MAX levels	

Table 74 – Parameters for test sequence FADE TIME/FADE RATE: stop fading	203
Table 75 – Parameters for test sequence FADE TIME/FADE RATE: stop fading when a command is sent, check timing	205
Table 76 – Parameters for test sequence FADE TIME/FADE RATE: stop fading during startup	210
Table 77 – Parameters for test sequence Level instructions: combined instructions	212
Table 78 – Parameters for test sequence PowerOnLevel and SystemFailureLevel	217
Table 79 – Parameters for test sequence ENABLE DAPC SEQUENCE	219
Table 80 – Parameters for test sequence GO TO LAST ACTIVE LEVEL	220
Table 81 – Parameters for test sequence GO TO SCENE	222
Table 82 – Parameters for test sequence Power on: level control commands	224
Table 83 – Parameters for test sequence Logarithmic dimming curve	225
Table 84 – Parameters for test sequence Dimming curve: DAPC	226
Table 85 – Parameters for test sequence FADE TIME/EXTENDED FADE TIME: light output behaviour	229
Table 86 – Parameters for test sequence Behaviour during a fade	232
Table 87 – Parameters for test sequence INITIALISE - device addressing	235
Table 88 – Parameters for test sequence COMPARE	237
Table 89 – Parameters for test sequence WITHDRAW	238
Table 90 – Parameters for test sequence PROGRAM SHORT ADDRESS	241
Table 91 – Parameters for test sequence VERIFY SHORT ADDRESS	242
Table 92 – Parameters for test sequence QUERY SHORT ADDRESS	243
Table 93 – Parameters for test sequence IDENTIFY DEVICE	246
Table 94 – Parameters for test sequence IDENTIFY DEVICE THROUGH RECALL MIN/MAX LEVEL	250
Table 95 – Parameters for test sequence QUERY STATUS - lampFailure/lampOn	253
Table 96 – Parameters for test sequence QUERY STATUS - lampOn	254
Table 97 – Parameters for test sequence QUERY STATUS - limitError/lampOn	256
Table 98 – Parameters for test sequence QUERY STATUS - powerCycleSeen	258
Table 99 – Parameters for test sequence QUERY CONTROL GEAR PRESENT	259
Table 100 – Parameters for test sequence Broadcast unaddressed	261
Table 101 – Parameters for test sequence Reserved commands: standard commands	262
Table 102 – Parameters for test sequence Reserved commands: special commands	263
Table 103 – Parameters for test sequence Addressing 2	272

INTERNATIONAL ELECTROTECHNICAL COMMISSION

DIGITAL ADDRESSABLE LIGHTING INTERFACE –**Part 102: General requirements –
Control gear****FOREWORD**

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as “IEC Publication(s)”). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62386-102 has been prepared by subcommittee 34C: Auxiliaries for lamps, of IEC technical committee 34: Lamps and related equipment.

This second edition cancels and replaces the first edition published in 2009. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

- a) elimination of all non-control gear relevant definitions,
- b) improvement of the requirements for control gear by clarifying the description,
- c) improvement of the test command iterations to increase the compatibility,
- d) addition of new commands, and
- e) the deletion of the requirements for:

- 1) timing;
- 2) control devices.

The requirements for timing are now in Part 101 and the requirements for control devices are in Part 103.

The text of this standard is based on the following documents:

FDIS	Report on voting
34C/1099/FDIS	34C/1112/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

This Part 102 is intended to be used in conjunction with Part 101, which contains general requirements for the relevant product type (system), and with the appropriate Part 2xx (particular requirements for control gear) containing clauses to supplement or modify the corresponding clauses in Parts 101 and 102 in order to provide the relevant requirements for each type of product.

A list of all parts of the IEC 62386 series, under the general title: *Digital addressable lighting interface*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the maintenance result date indicated on the IEC website under <http://webstore.iec.ch> in the data related to the specific publication. At this date, the publication will be

- reconfirmed;
- withdrawn;
- replaced by a revised edition, or
- amended.

A bilingual version of this publication may be issued at a later date.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

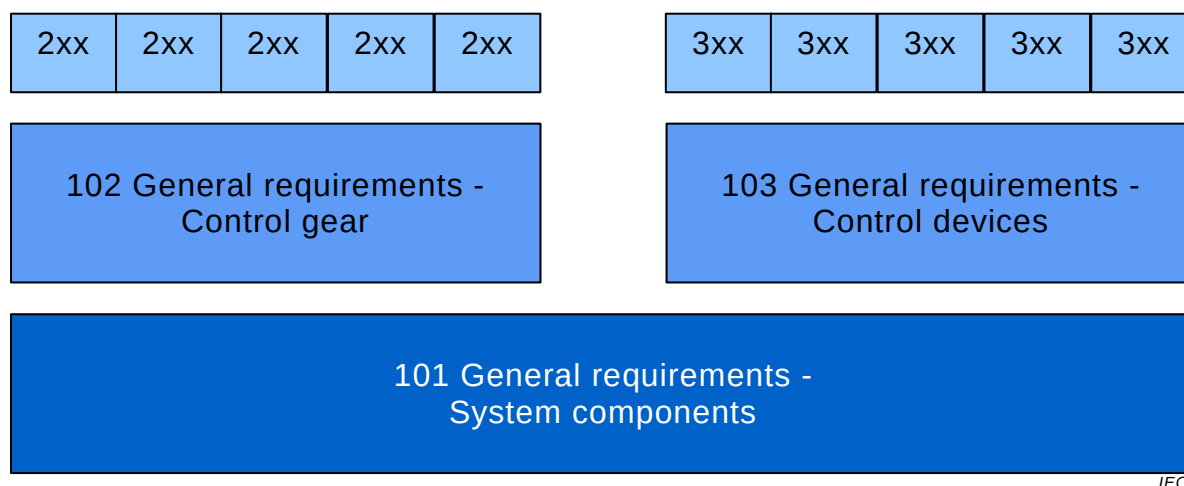
IEC 62386 contains several parts, referred to as series. The 1xx series includes the basic specifications. Part 101 contains general requirements for system components, Part 102 extends this information with general requirements for control gear and Part 103 extends it further with general requirements for control devices.

The 2xx parts extend the general requirements for control gear with lamp specific extensions (mainly for backward compatibility with Edition 1 of IEC 62386) and with control gear specific features.

The 3xx parts extend the general requirements for control devices with input device specific extensions describing the instance types as well as some common features that can be combined with multiple instance types.

This second edition of IEC 62386-102 is published in conjunction with IEC 62386-101:2014 and with the various parts that make up the IEC 62386-2xx series for control gear, together with IEC 62386-103:2014 and the various parts that make up the IEC 62386-3xx series of particular requirements for control devices. The division into separately published parts provides for ease of future amendments and revisions. Additional requirements will be added as and when a need for them is recognised.

The setup of the standard is graphically represented in Figure 1 below.



IEC

Figure 1 – IEC 62386 graphical overview

When this part of IEC 62386 refers to any of the clauses of the other two parts of the IEC 62386-1xx series, the extent to which such a clause is applicable and the order in which the tests are to be performed are specified. The other parts also include additional requirements, as necessary.

All numbers used in this International Standard are decimal numbers unless otherwise noted. Hexadecimal numbers are given in the format 0xVV, where VV is the value. Binary numbers are given in the format XXXXXXXXb or in the format XXXX XXXX, where X is 0 or 1 and "x" in binary numbers means "don't care".

The following typographic expressions are used:

Variables: *variableName* or *variableName*[3:0], giving only bits 3 to 0 of *variableName*

Range of values: [lowest, highest]

Command: "COMMAND NAME"

DIGITAL ADDRESSABLE LIGHTING INTERFACE –

Part 102: General requirements – Control gear

1 Scope

This Part of IEC 62386 is applicable to control gear in a bus system for control by digital signals of electronic lighting equipment. This electronic lighting equipment should be in line with the requirements of IEC 61347, with the addition of d.c. supplies.

NOTE Tests in this standard are type tests. Requirements for testing individual control gear during production are not included.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61347 (all parts), *Lamp controlgear*

IEC 62386-101:2014, *Digital addressable lighting interface – Part 101: General requirements – System components*

IEC 62386-103:2014, *Digital addressable lighting interface – Part 103: General requirements – Control devices*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 62386-101 and the following apply.

3.1

actual level

value representing the current light output

3.2

arc power

power supplied to the light sources (lamps)

3.3

broadcast

type of address used to address all control gear in the system at once

3.4

broadcast unaddressed

type of address used to address all control devices in the system that have no short address at once

3.5

DAPC

direct arc power control

a method to directly control the light output

Note 1 to entry: The note to entry in French concerns the French text only.

3.6

DTR

data transfer register

multipurpose register used to exchange data

Note 1 to entry: The note to entry in French concerns the French text only.

3.7

group address

type of address used to address a group of control gear in the system all at once

3.8

GTIN

global trade item number

number used for the unique identification of trade items worldwide

Note 1 to entry: For further information see <http://en.wikipedia.org/wiki/GTIN>

Note 2 to entry: The number is comprised of a GS1 or U.P.C. company prefix followed by an item reference number and a check digit. It is described in the "GS1 General Specifications".

Note 3 to entry: The note 3 to entry in French concerns the French text only.

3.9

identification

temporary state used during commissioning that allows the installer to identify particular control gear

3.10

level

8 bit value

3.11

MASK

the value 0xFF

3.12

monotonic

a function f defined on a subset of the real numbers with real values is called monotonically non-decreasing, if for all x and y such that $x \leq y$ one has $f(x) \leq f(y)$, so f preserves the order. Likewise, a function is called monotonically non-increasing if, whenever $x \leq y$, then $f(x) \geq f(y)$, so it reverses the order. For this standard monotonic is defined as either monotonically non-decreasing or monotonically non-increasing

3.13

NO

if a query is asked where the answer is NO, there will be no response, such that the sender of the query will conclude "no backward frame" following subclause 8.2.5 of IEC 62386-101:2014

Note 1 to entry: The answer NO could also be triggered by a missed query.

3.14**NVM**

non-volatile read/write memory, the content of which can be changed and will not be lost due to a power cycle

3.15**opcode****operation code**

that part of a command frame that identifies the command to be executed

3.16**operating mode**

set of states identified by a number in the range [0,255], characterised by a collection of variables and memory settings, and used to select a set of functionality to be exhibited by a control gear, including its required reaction to commands

Note 1 to entry: Control gear may support more than one operating mode

3.17**PHM**

physical minimum level corresponding to the minimum light output the control gear can operate at

3.18**RAM**

volatile read/write memory, the content of which can be changed and will be lost due to a power cycle

3.19**random address**

random 24 bit number generated by the control gear on request during system initialisation

Note 1 to entry: Annex A.1 provides an example of how the search and random addresses are used.

3.20**reset state**

state in which all NVM variables of the control gear have their reset value, except those that are marked “no change” or are otherwise explicitly excluded

3.21**ROM**

non-volatile read only memory, the content of which is fixed

Note 1 to entry: In this standard read only is meant from a system perspective. A ROM variable may actually be implemented in NVM, but this standard does not provide any mechanism to change its value.

3.22**scene**

configurable preset level

3.23**search address**

24 bit number used to identify an individual control gear in the system during initialisation

Note 1 to entry: Annex A.1 provides an example of how the search and random addresses are used.

3.24**short address**

type of address used to address an individual control gear in the system

3.25

startup

time needed to change from lamp off to normal operation of the lamp or failure state

Note 1 to entry: This time includes preheat and ignition.

3.26

strictly monotonic

a function f defined on a subset of the real numbers with real values is called monotonically increasing, if for all x and y such that $x < y$ one has $f(x) < f(y)$, so f preserves the order. Likewise, a function is called monotonically decreasing if, whenever $x < y$, then $f(x) > f(y)$, so it reverses the order. For this standard strictly monotonic is defined as either monotonically increasing or monotonically decreasing

3.27

target level

the target light output expected after completion of the current level command

3.28

YES

if a query is asked where the answer is YES, the response will be a backward frame containing the value of *MASK*

4 General

4.1 General

The requirements of IEC 62386-101:2014, Clause 4 apply, with the restrictions, changes and additions identified below.

4.2 Version number

This subclause replaces IEC 62386-101:2014, Subclause 4.2.

The version shall be in the format "x.y", where the major version number x is in the range of 0 to 62 and the minor version number y is in the range of 0 to 2. When the version number is encoded into a byte, the major version number x shall be placed in bits 7 to 2 and the minor version number y shall be placed in bits 1 to 0.

At each amendment to an edition of IEC 62386-102 the minor version number shall be incremented by one.

At a new edition of IEC 62386-102 the major version number shall be incremented by one and the minor version number shall be set to 0.

The current version number is "2.0".

NOTE Normally 2 amendments on IEC documents are made before a new edition is created.

5 Electrical specification

The requirements of IEC 62386-101:2014, Clause 5 apply.

6 Interface power supply

If a bus power supply is integrated into a control gear, the requirements of IEC 62386-101:2014, Clause 6 apply.

7 Transmission protocol structure

7.1 General

The requirements of Clause 7 of IEC 62386-101:2014 apply, with the following additions.

7.2 16 bit forward frame encoding

7.2.1 General

For commands, the 16 bit forward frame shall be encoded as is depicted in Table 1.

Table 1 – 16-bit command frame encoding

Bytes/Bits								Device addressing method	
Address byte							Opcode byte		
15	14	13	12	11	10	9	8 ^a		7...0
0	64 short addresses						x		Short addressing
1	0	0	16 group addresses				x		Group addressing
1	1	1	1	1	1	0	x		Broadcast unaddressed
1	1	1	1	1	1	1	x		Broadcast
1010 0000 to 1100 1011								Special command	
1100 1100 to 1111 1011								Reserved	
^a Selector bit, see 7.2.2; 0 indicates DAPC, 1 indicates other command									

7.2.2 Address byte

The address byte provides

- the method of device addressing used by the application controller;
- the type of command transmitted in the opcode byte;
Bit 8 = '1': standard command;
Bit 8 = '0': direct arc power control (DAPC) command;
- address spaces for special commands;
- reserved device addresses.

Reserved addresses should not be used by the application controller.

7.2.3 Opcode byte

The opcode byte provides

- for DAPC commands, the requested light output;
- for standard commands, the opcode;
- command specific information for special commands;
- reserved information for reserved commands.

8 Timing

The requirements of IEC 62386-101, Clause 8 apply.

9 Method of operation

9.1 General

The requirements of IEC 62386-101, Clause 9 apply with the following additions.

9.2 Control gear

Control gear may receive commands from an application controller. The application controller is specified by IEC 62386-103:2014.

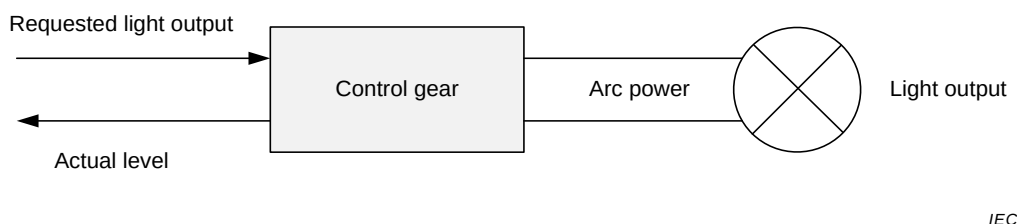


Figure 2 – Control gear directly operating a light source

Figure 2 shows how the various levels lead to light output. The maximum (light) output level of a control gear is referred to as 100 %. All levels are specified in a relative way. Physically there is a minimum that the control gear can supply whilst there is still light output. This is known as the physical minimum level (PHM).

NOTE PHM is gear specific, and is greater than 0.

Depending on the light source various phases of operation can be identified within a control gear. In general these are as follows.

- Standby: during this phase, the lamp is off.
- Startup: this is a transitional phase changing from standby to normal operation. This phase is sometimes noticeable as a delay. Examples are:
 - preheat: the lamp is heated to prepare for ignition. This is typically seen for fluorescent light sources;
 - ignition: the lamp is ignited. This is typically seen for HID light sources and fluorescent light sources after preheat;
 - power stage preparation.
- Normal operation: the lamp is emitting light and can be operated as expected.
- Failure: the lamp cannot be operated as expected.

9.3 Dimming curve

The dimming curve determines how the level shall be translated into light output.

An “*actualLevel*” greater than or equal to 1 and less than or equal to 254 shall be translated into light output according to

$$\text{Light output (actualLevel)} = 10^{\frac{\text{actualLevel}-1}{253/3}} - 1 \quad \%$$

Light output is expressed relative to the maximum possible light output of a given control-gear-lamp combination. The dimming curve starts at 0,1 % for “*actualLevel*” equal to 0x01 and

ends at 100 % for “*actualLevel*” equal to 0xFE. The dimming curve is strictly monotonic, and the relative accuracy shall be $\pm\frac{1}{2}$ step. This shall be tested using a fade, excluding PHM.

NOTE 1 The dimming curve is intended to compensate the light sensitivity curve of the human eye.

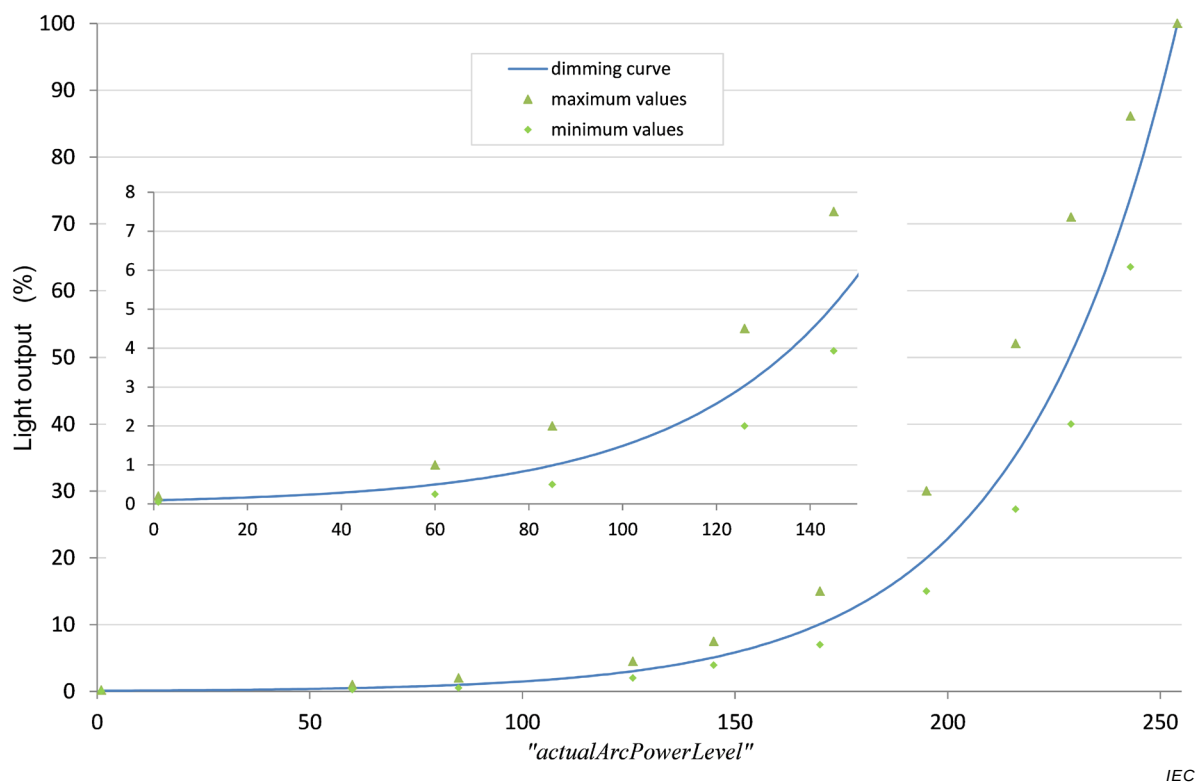


Figure 3 – Dimming curve

The accuracy of light output is specified by the test points given in Table 2. The test points of Table 2 and the dimming curve are depicted in Figure 3, and Table 3 shows the light output versus the level. The lamp type or load used during testing shall be stated for reproducibility.

NOTE 2 The minimum and maximum values are based on the test values that can be found in IEC 62386-102:2009.

Table 2 – Dimming curve tolerance (% , rounded to two decimals)

Arc power level	1	60	85	126	145	170	195	216	229	243	254
Minimum value	0,05	0,25	0,50	2,00	3,93	7,00	15,00	27,28	40,00	63,53	
Nominal value	0,10	0,50	0,99	3,04	5,10	10,09	19,97	35,43	50,53	74,05	100,00
Maximum value	0,20	1,00	2,00	4,50	7,50	15,0	30,00	52,09	71,00	86,14	

Table 3 – Dimming curve

Level	Light output	Level	Light output	Level	Light output	Level	Light output	Level	Light output
1	0.100	52	0.402	103	1.620	154	6.520	205	26.241
2	0.103	53	0.414	104	1.665	155	6.700	206	26.967
3	0.106	54	0.425	105	1.711	156	6.886	207	27.713
4	0.109	55	0.437	106	1.758	157	7.076	208	28.480
5	0.112	56	0.449	107	1.807	158	7.272	209	29.269
6	0.115	57	0.461	108	1.857	159	7.473	210	30.079
7	0.118	58	0.474	109	1.908	160	7.680	211	30.911
8	0.121	59	0.487	110	1.961	161	7.893	212	31.767
9	0.124	60	0.501	111	2.015	162	8.111	213	32.646
10	0.128	61	0.515	112	2.071	163	8.336	214	33.550
11	0.131	62	0.529	113	2.128	164	8.567	215	34.479
12	0.135	63	0.543	114	2.187	165	8.804	216	35.433
13	0.139	64	0.559	115	2.248	166	9.047	217	36.414
14	0.143	65	0.574	116	2.310	167	9.298	218	37.422
15	0.147	66	0.590	117	2.374	168	9.555	219	38.457
16	0.151	67	0.606	118	2.440	169	9.820	220	39.522
17	0.155	68	0.623	119	2.507	170	10.091	221	40.616
18	0.159	69	0.640	120	2.577	171	10.371	222	41.740
19	0.163	70	0.658	121	2.648	172	10.658	223	42.895
20	0.168	71	0.676	122	2.721	173	10.953	224	44.083
21	0.173	72	0.695	123	2.797	174	11.256	225	45.303
22	0.177	73	0.714	124	2.874	175	11.568	226	46.557
23	0.182	74	0.734	125	2.954	176	11.888	227	47.846
24	0.187	75	0.754	126	3.035	177	12.217	228	49.170
25	0.193	76	0.775	127	3.119	178	12.555	229	50.531
26	0.198	77	0.796	128	3.206	179	12.902	230	51.930
27	0.203	78	0.819	129	3.294	180	13.260	231	53.367
28	0.209	79	0.841	130	3.386	181	13.627	232	54.844
29	0.215	80	0.864	131	3.479	182	14.004	233	56.362
30	0.221	81	0.888	132	3.576	183	14.391	234	57.922
31	0.227	82	0.913	133	3.675	184	14.790	235	59.526
32	0.233	83	0.938	134	3.776	185	15.199	236	61.173
33	0.240	84	0.964	135	3.881	186	15.620	237	62.866
34	0.246	85	0.991	136	3.988	187	16.052	238	64.607
35	0.253	86	1.018	137	4.099	188	16.496	239	66.395
36	0.260	87	1.047	138	4.212	189	16.953	240	68.233
37	0.267	88	1.076	139	4.329	190	17.422	241	70.121
38	0.275	89	1.105	140	4.449	191	17.905	242	72.062
39	0.282	90	1.136	141	4.572	192	18.400	243	74.057
40	0.290	91	1.167	142	4.698	193	18.909	244	76.107
41	0.298	92	1.200	143	4.828	194	19.433	245	78.213
42	0.306	93	1.233	144	4.962	195	19.971	246	80.378
43	0.315	94	1.267	145	5.099	196	20.524	247	82.603
44	0.324	95	1.302	146	5.240	197	21.092	248	84.889
45	0.332	96	1.338	147	5.385	198	21.675	249	87.239
46	0.342	97	1.375	148	5.535	199	22.275	250	89.654
47	0.351	98	1.413	149	5.688	200	22.892	251	92.135
48	0.361	99	1.452	150	5.845	201	23.526	252	94.686
49	0.371	100	1.492	151	6.007	202	24.177	253	97.307
50	0.381	101	1.534	152	6.173	203	24.846	254	100.000
51	0.392	102	1.576	153	6.344	204	25.534		

9.4 Calculating “targetLevel”

An application controller instructs the control gear on the requested light output and on the behaviour during the transition from the “actualLevel” to the “targetLevel” by means of appropriate opcodes.

The “targetLevel” shall be calculated on the basis of the requested light output as follows:

- 0x00 shall be accepted as “targetLevel” and turn off the light.
- Any value between 0x01 and “minLevel” shall result in “targetLevel” = “minLevel”.
- Any value between “maxLevel” and 0xFE shall result in “targetLevel” = “maxLevel”.
- “MASK” shall have no effect on “targetLevel” except when a fade is running, see subclause 9.5.9.
- All other values shall be accepted as “targetLevel”.

The requested target level calculation of “targetLevel” shall also be applied if the request is based on an internally stored value, such as a scene, “powerOnLevel”, or “systemFailureLevel”.

On every change of “targetLevel”, with the exception of the initialisation caused by a power cycle, the control gear shall update “limitError” (see subclause 9.16.5) and shall set “lastLightLevel” to the new “targetLevel”. If “targetLevel” is not 0x00, “lastActiveLevel” shall be set to “targetLevel”.

9.5 Fading

9.5.1 General

Fading is a linear transition in time from “actualLevel” to “targetLevel”. The “actualLevel”, and thus the light output, shall be strictly monotonic according to the applicable dimming curve.

A fade can be started in two ways:

- using a fade time: this sets a time to use for the fade process;
- using a fade rate: this sets a speed to use for the fade process.

A fade shall not be started if the calculated “targetLevel” is equal to “actualLevel”.

When a fade starts, the fade timer shall be started and “fadeRunning” shall be set to TRUE (see 9.16.6.).

During the fade, the light output shall be maintained as close to the ideal fading curve as possible.

During a process of fading up, “actualLevel” shall be incremented at a time corresponding to the intersection of an ideal fading curve with the mid-point between “actualLevel” and “actualLevel” + 1. Likewise, when fading down, “actualLevel” shall be decremented at a time corresponding to the intersection of an ideal fading curve with the mid-point between “actualLevel” and “actualLevel” - 1. Figure 4 illustrates this.

Measurements of fade time / fade rate shall start after the stop condition of the command that triggers it. If the fade takes place immediately after startup, measurement shall be done from the moment “lampOn” is TRUE or, in case of total lamp failure, from the moment “lampFailure” is confirmed TRUE. A fade shall automatically end when the fade timer has been active for the applicable fade time. At this point the fade timer shall be stopped and “fadeRunning” shall be set to FALSE (see subclause 9.16.6.).

This means that the control gear fades to the target level even in case of a total lamp error. If a lamp is to be switched off at the end of the fade, the step from "*minLevel*" to 0x00 shall not contribute to the fade time. The step from "*minLevel*" to 0x00 shall be taken immediately after the fade time has elapsed.

If a lamp is to be lit at the beginning of the fade and dimmed to a certain value, the step from 0x00 to "*minLevel*" shall not contribute to the fade time. This means that the fade time starts when the lamp is on.

NOTE The transition from 0x00 to "*minLevel*" incorporates startup.

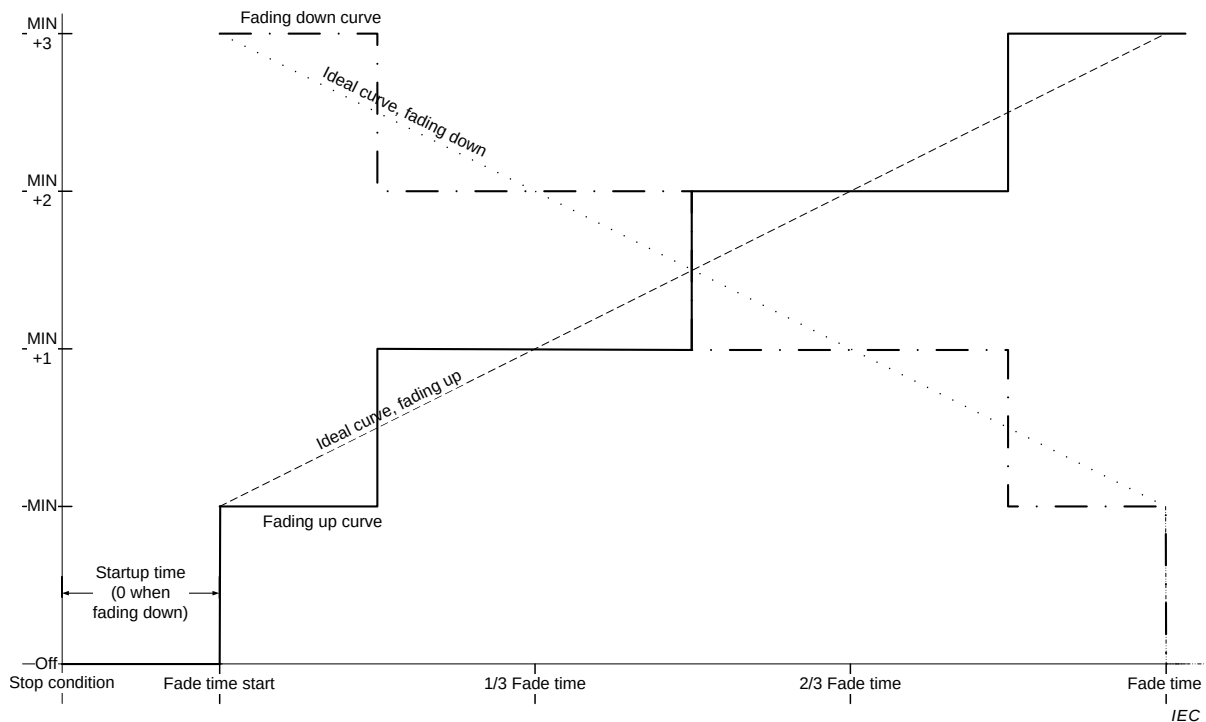


Figure 4 – Level over time, fading up and down

Testing shall be done with "*minLevel*" $\geq$ PHM+1. For further information, see Annex B.

9.5.2 Fade time

The fade time shall be according to Table 4:

"*fadeTime*" shall be set on receipt of the command "SET FADE TIME (DTR0)". "*fadeTime*" can be queried using QUERY FADE TIME/FADE RATE.

The "*fadeTime*" shall be set to a value according to the following steps:

- if "*DTR0*" > 15: 15
- in all other cases: "*DTR0*"

The fade time shall be calculated on the basis of "*fadeTime*" as follows:

- if "*fadeTime*" = 0: use Extended fade time
- if "*fadeTime*" is in the range [1,15]: $\frac{1}{2} \cdot \sqrt{2^{\text{"fadeTime"}}} \cdot 1 \text{ s}$

Table 4 lists the possible fade time values.

Table 4 – Fade times

<i>“fadeTime”</i>	Minimum fade time s	Nominal fade time s	Maximum fade time s
0	Extended fade		
1	0,6	0,7	0,8
2	0,9	1,0	1,1
3	1,3	1,4	1,6
4	1,8	2,0	2,2
5	2,5	2,8	3,1
6	3,6	4,0	4,4
7	5,1	5,7	6,2
8	7,2	8,0	8,8
9	10,2	11,3	12,4
10	14,4	16,0	17,6
11	20,4	22,6	24,9
12	28,8	32,0	35,2
13	40,7	45,3	49,8
14	57,6	64,0	70,4
15	81,5	90,5	99,6

9.5.3 Fade rate

The fade rate shall be according to Table 5, where for testing purposes the time is considered to be precisely 200 ms for the last command using the fade rate.

“fadeRate” shall be set on receipt of the command “SET FADE RATE (*DTR0*)”. *“fadeRate”* can be queried using QUERY FADE TIME/FADE RATE.

The *“fadeRate”* shall be set to a value according to the following steps:

- if *“DTR0”* > 15: 15
- if *“DTR0”* = 0: 1
- in all other cases: *“DTR0”*

The fade rate shall be calculated on the basis of *“fadeRate”* as follows:

$$\text{Fade rate} = \frac{506}{\sqrt{2^{\text{“fadeRate”}}}} \text{ steps/s.}$$

Table 5 lists the possible fade rate values.

Table 5 – Fade rates

<i>“fadeRate”</i>	Minimum fade rate steps/s	Nominal fade rate steps/s	Maximum fade rate steps/s
1	322	358	394
2	228	253	278
3	161	179	197
4	114	127	139
5	80,5	89,4	98,4
6	56,9	63,3	69,6
7	40,3	44,7	49,2
8	28,5	31,6	34,8
9	20,1	22,4	24,6
10	14,2	15,8	17,4
11	10,1	11,2	12,3
12	7,1	7,9	8,7
13	5,0	5,6	6,1
14	3,6	4,0	4,3
15	2,5	2,8	3,1

9.5.4 Extended fade time

If *“fadeTime”* equals 0, and the fast fade time as defined in IEC 62386 Part 207 is implemented and equals 0, the extended fade time shall be used.

The extended fade time shall be according to Table 7.

The extended fade time can be set using a base value and a multiplier according to Table 6 and Table 7. The extended fade time can be calculated based on the base value and the multiplication factor.

Fade time = *extendedFadeTimeBase* * *extendedFadeTimeMultiplier*

This yields a range of 100 ms to 16 min, and a special value indicating no fade (as quickly as possible).

Table 6 – Extended fade time - base value

Base bits	Base value
0000b	1
0001b	2
0010b	3
0011b	4
0100b	5
0101b	6
0110b	7
0111b	8
1000b	9
1001b	10
1011b	11
1011b	12
1100b	13
1101b	14
1110b	15
1111b	16

Table 7 – Extended fade time - multiplier

Multiplier bits	Multiplication factor		
	Minimum	Nominal	Maximum
000b	0 ms ^a	0 ms ^{a)}	0 ms ^a
001b	95 ms	100 ms	105 ms
010b	0,95 s	1 s	1,05 s
011b	9,5 s	10 s	10,5 s
100b	0,95 min	1 min	1,05 min
101b		Reserved	
110b		Reserved	
111b		Reserved	
^a No fade (as quickly as possible)			

On reception of “SET EXTENDED FADE TIME (*DTR0*)” the control gear shall set the following values based on “*DTR0*”. The format used shall be 0YYYAAAAb, where YYYb equals the fade time multiplier, and AAAAb the fade time base: The resulting fade time shall be monotonically increasing when the base time increases.

- If “*DTR0*” > 0100 1111b:
 - “*extendedFadeTimeBase*” shall be set to 0
 - “*extendedFadeTimeMultiplier*” shall be set to 0 ms, effectively setting the fade time to 0 s meaning no fade (as quickly as possible). The transition from “*actualLevel*” to “*targetLevel*” shall take place immediately and the light output shall be adjusted as quickly as possible.
- In all other cases:
 - “*extendedFadeTimeBase*” shall be set to AAAAb
 - “*extendedFadeTimeMultiplier*” shall be set to YYYb

The extended fade time can be queried using “QUERY EXTENDED FADE TIME”. The answer shall be 0 YYY AAAAb, where YYYb equals “*extendedFadeTimeMultiplier*” and AAAAb equals “*extendedFadeTimeBase*”.

9.5.5 Using the fade time

Commands that use a fade time shall start a fade using the applicable fade time. This time can be determined based on the following rules:

- If “*fadeTime*” > 0: see Table 4 – Fade times
- If “*fadeTime*” = 0: The extended fade time shall be used, see Table 6 – Extended fade time - base value and Table 7 – Extended fade time - multiplier. The extended fade time can be calculated by multiplying the base value and the multiplier.
- If “*extendedFadeTimeMultiplier*” = 0 ms, the fade time equals 0 s, meaning no fade (as quickly as possible). The transition from “*actualLevel*” to “*targetLevel*” shall take place immediately and the light output shall be adjusted as quickly as possible.

The target level shall be calculated on the basis of the command parameter. After the fade time has expired, the calculated target level shall be reached.

Since the extended fade time also supports fade times below 0,7 s that might not be realised by all control gear and light source combinations, such gear may simply adjust the light output as quickly as possible when an extended fade time is requested that it physically cannot support. However, it should respond as if the fade has finished within the requested time.

9.5.6 Using the fade rate

Commands that use the fade rate shall start a 200 ms ± 20 ms fade.

“*targetLevel*” shall be calculated on the basis of the “*actualLevel*” using the applicable fade rate. After the 200 ms fade has expired, the calculated target level shall be reached.

NOTE 1 Since the fade rate is used, it is possible to reach “*minLevel*” or “*maxLevel*” before the end of the fade. This does not result in the “*fadeRunning*” bit being cleared.

NOTE 2 Because there are fade rate tolerances, different gear may react to commands that use the fade rate at slightly different effective rates. Consequently, after the processing of these relative dimming commands, different gear might have different values for “*targetLevel*” (and therefore also for “*actualLevel*” and “*lastLightLevel*”).

9.5.7 Behaviour during a fade

If “*fadeTime*”, “*extendedFadeTimeBase*”, “*extendedFadeTimeMultiplier*” and/or “*fadeRate*” is changed during a running fade, then the running fade shall finish without the fade time and/or fade rate being recalculated. The next fade shall use the recalculated values.

9.5.8 Behaviour during startup

During startup, the fade process shall be pended with “*actualLevel*” equal to “*minLevel*”. The reaction to level commands shall be the same as if the lamp(s) were operating at “*minLevel*”. The fade shall start:

- As soon as “*lampOn*” is TRUE
- or, in case of total lamp failure, as soon as “*lampFailure*” is confirmed TRUE

9.5.9 Stopping a fade

Any command setting one or more of the following variables

- “*targetLevel*”, “*minLevel*”, “*maxLevel*”

as well as the reception of one of the following commands

- “DAPC(MASK)”, “SAVE PERSISTENT VARIABLES”, “IDENTIFY DEVICE”

shall stop a running fade.

NOTE 1 The fade stops even if the value of the affected variable does not change.

When a running fade is stopped by an application controller, the fade timer shall be stopped immediately. After the fade timer has been stopped, “*targetLevel*” shall be set to “*actualLevel*” and the command shall be executed (if applicable).

If a running fade is stopped whilst it was pending at “*minLevel*” during startup, the control gear shall finish the startup process.

NOTE 2 This implies that in such a case both “*targetLevel*” and “*actualLevel*” are equal to “*minLevel*”.

9.6 Min and max level

Changing the min or max level shall stop any running fade, before the storage of the new min or max level.

“SET MIN LEVEL (*DTR0*)” shall set “*minLevel*” depending on the “*DTR0*” value:

- if $0 \leq \text{“DTR0”} \leq \text{PHM}$: PHM
- if “*DTR0*” $\geq$ “*maxLevel*” or MASK: “*maxLevel*”
- in all other cases: “*DTR0*”

If “*actualLevel*” > 0 and “*actualLevel*” $<$ “*minLevel*” as a result of setting a new min level, “*targetLevel*” shall be re-calculated on the basis of the new “*minLevel*”. “*actualLevel*” shall be changed to “*targetLevel*” immediately and the light output shall be adjusted as quickly as possible.

“SET MAX LEVEL (*DTR0*)” shall set “*maxLevel*” depending on the “*DTR0*” value, as follows:

- if “*minLevel*” $\geq$ “*DTR0*”: “*minLevel*”
- if “*DTR0*” = MASK: 0xFE
- in all other cases: “*DTR0*”

If “*actualLevel*” $>$ “*maxLevel*” as a result of setting a new max level, “*targetLevel*” shall be re-calculated on the basis of the new “*maxLevel*”. “*actualLevel*” shall be changed to “*targetLevel*” immediately and the light output shall be adjusted as quickly as possible.

NOTE “*minLevel*” and “*maxLevel*” can be used to compensate for differences in control gear properties. E.g. if control gear have different values for PHM, they can be made to behave in a similar way by adjusting “*minLevel*”.

9.7 Commands

9.7.1 General

A control gear shall check the device addressing scheme to see if it is addressed by a command. The control gear shall accept the command, unless any of the following conditions hold:

- The command is sent using Short addressing and given short address is not equal to “*shortAddress*”.
- The command is sent using Group addressing and given group does not match any of the groups identified by “*gearGroups*”.
- The command is sent using Reserved addressing.

- The command is sent using Broadcast Unaddressed addressing and “*shortAddress*” is not MASK.
- The command is not defined.

The following command groups can be identified:

- Level instructions
 - Level instructions without fade
 - Level instructions initiating a fade
- Configuration instructions
- Queries
- Special commands
 - Instructions
 - Queries
- Application extended commands

9.7.2 Level instructions without fade

Level instructions without fade are instructions where the “*targetLevel*” shall be calculated; the transition from “*actualLevel*” to “*targetLevel*” shall take place immediately and the light output shall be adjusted as quickly as possible.

These commands can be divided into three categories:

- Absolute level commands
 - “OFF”, “RECALL MIN LEVEL”, “RECALL MAX LEVEL”,
- Relative level commands
 - “STEP UP”, “STEP DOWN”, “ON AND STEP UP”, “STEP DOWN AND OFF”
- Configuration commands
 - “RESET”, “SET MIN LEVEL (*DTR0*)”, “SET MAX LEVEL (*DTR0*)”

9.7.3 Level instructions initiating a fade

Level instructions initiating a fade are instructions where the “*targetLevel*” shall be calculated; “*actualLevel*” shall fade to the “*targetLevel*” using the applicable fade time/rate. If the fade time equals 0 s, the transition from “*actualLevel*” to “*targetLevel*” shall take place immediately and the light output shall be adjusted as quickly as possible.

These commands can be divided into two categories:

- Absolute level instructions using the fade time
 - “DAPC (*level*)”, “GO TO SCENE (*sceneNumber*)”, “GO TO LAST ACTIVE LEVEL”
- Relative level instructions using the fade rate
 - “UP”, “DOWN”

9.7.4 Configuration instructions

Configuration instructions can be used to modify several control gear properties.

9.7.5 Queries

Queries can be used to request the value of several control gear properties.

9.7.6 Special commands

The special commands are a group of commands that are not addressable. All control gear shall interpret the special commands.

9.7.7 Application extended commands

Commands with their opcode in the range 0xE0 to 0xFF are reserved for special device types or features. Each device type or feature re-defines these commands, except for the command with opcode 0xFF (“QUERY EXTENDED VERSION NUMBER”). See 9.18 for further information.

9.8 Command iterations

9.8.1 General

The requirements of subclause 9.4 of IEC 62386-101:2014 apply with the following additions.

9.8.2 Command iteration of “UP” and “DOWN” commands

“UP” and “DOWN” instructions can be sent as a command iteration. Upon reception of the first instruction of such an iteration, unless this is precluded by the values of “minLevel” or “maxLevel”, one step (“targetLevel” = “targetLevel” ± 1) shall be made.

NOTE 1 This ensures that there is an effect at the start of an iteration.

After that first step, the 200 ms fade shall start using the applicable fade rate. Subsequent steps shall be executed at intervals determined by the applicable fade rate, as long as the iteration continues. Every “UP” or “DOWN” instruction received as a part of the iteration shall cause the 200 ms fade time to be restarted and “targetLevel” to be recalculated on the basis of “actualLevel” and the set fade rate.

NOTE 2 If the fade rate changes during a command iteration, the new fade rate is not used during the execution of this command iteration.

Figure 5 summarizes the required behaviour. The iterations start at Cmd 1, and end at Time out.

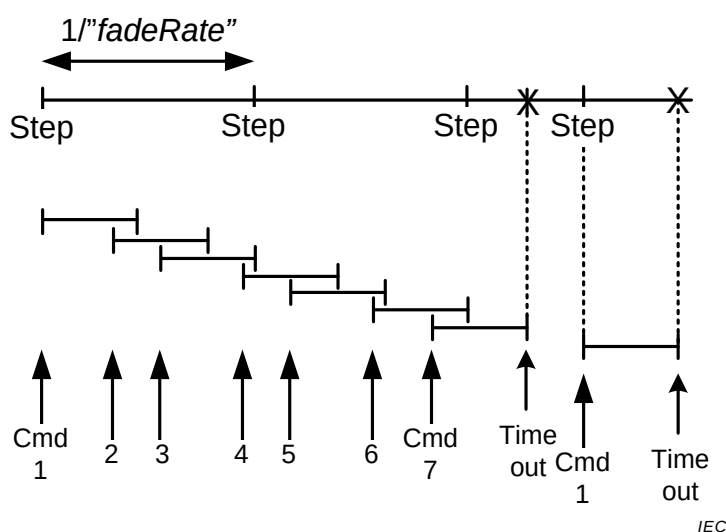


Figure 5 – Timing and response when receiving a command iteration

9.8.3 DAPC SEQUENCE (deprecated)

“ENABLE DAPC SEQUENCE” starts a direct arc power control (DAPC) command iteration that allows dynamic control of the light output.

Upon reception of “ENABLE DAPC SEQUENCE” the control gear shall temporarily use a fade time of $200\text{ ms} \pm 20\text{ ms}$ while the command iteration is active independent of the actual fade/extended fade time. After the last fade of the sequence has finished, the original values shall be used.

NOTE As the fade time/rate variables do not change, the fade time/rate can be set and/or queried as normal.

The DAPC sequence shall end if 200 ms elapse without the control gear receiving a “DAPC (*level*)” command. The DAPC sequence shall be aborted on reception of an indirect arc power control command. “ENABLE DAPC SEQUENCE” received during an enabled DAPC command iteration, shall have no effect.

Upon reception of the first “DAPC (*level*)” after reception of the “ENABLE DAPC SEQUENCE” command the 200 ms fade shall start.

Since the DAPC sequence uses a fade time of 200 ms that might not be realised by all control gear and light source combinations, such gear may simply adjust the light output as quickly as possible. However, it should respond as if the fade has finished within the requested time.

9.9 Modes of operation

9.9.1 General

Different operating modes can be selected by means of command “SET OPERATING MODE (*DTR0*)”. The currently selected “*operatingMode*” can be queried by means of “QUERY OPERATING MODE”.

Operating modes 0x00 to 0x7F are defined in this standard. At least operating mode 0x00 shall be available. Operating modes 0x80 to 0xFF are manufacturer specific. The query “QUERY MANUFACTURER SPECIFIC MODE” can be used to determine whether the control gear is in an IEC 62386 standard operating mode or in a manufacturer specific mode.

9.9.2 Operating mode 0x00: standard mode

If a device is in standard mode (“*operatingMode*” = 0x00), its behaviour shall be as is required per this specification, until it is set in an operating mode different from 0x00.

9.9.3 Operating mode 0x01 to 0x7F: reserved

Operating modes 0x01 to 0x7F are reserved and shall not be used.

9.9.4 Operating mode 0x80 to 0xFF: manufacturer specific modes

Manufacturer specific modes should only be used if the features required by the application are not covered by the standard. If a control gear is in a manufacturer specific operating mode, the behaviour of the control gear may be manufacturer specific as well, with the following exceptions:

- as far as the control gear accesses the bus, it shall adhere to IEC 62386-101:2014;
- the control gear shall adhere to this specification at least as far as the following commands are concerned:
 - “SET OPERATING MODE (*DTR0*)”, “QUERY OPERATING MODE”, and “QUERY MANUFACTURER SPECIFIC MODE”.
 - all special commands (see 11.7) except WRITE MEMORY LOCATION (*DTR1*, *DTR0*, *data*), WRITE MEMORY LOCATION – NO REPLY (*DTR1*, *DTR0*, *data*) and PING.

For the above commands the various addressing methods shall apply, see 7.2.2.

It is recommended that even in manufacturer specific modes, the commands as specified in this standard still be obeyed.

9.10 Memory banks

9.10.1 General

Memory banks are freely accessible memory spaces defined for e.g. identification of the control gear in a system. Not all consecutive memory banks need to be implemented. Also within a memory bank not all consecutive locations need to be implemented. All implemented memory bank locations of all implemented memory banks are readable using memory access commands. Part of the memory is read-only and programmed by the manufacturer of the control gear. For all other parts, write access using memory access commands can be enabled by the manufacturer. Write access to a memory bank location can be locked. Memory banks can be implemented using RAM, ROM or NVM.

The addressable memory space is limited to a maximum of almost 64 kBytes, organized in maximum 256 memory banks of maximum 255 bytes each. As this standard prescribes how to implement memory bank 0 and 1 (if present), and reserves memory banks 200 to 255, this leaves room for 198 memory banks for manufacturer specific purposes in the range of [2,199].

9.10.2 Memory map

If a manufacturer specific memory bank in the range of [2,199] is implemented, allocation of its content shall comply with the memory map provided in Table 8.

Table 8 – Basic memory map of memory banks

Address	Description	Default value (factory)	RESET value ^b	Memory type
0x00	Address of last accessible memory location	Factory burn-in, range [0x03,0xFE]	No change	ROM
0x01	Indicator byte ^a	a	a	Any ^a
0x02	Memory bank lock byte. Lockable bytes in the memory bank shall be read-only while the lock byte has a value different from 0x55.	0xFF	0xFF ^c	RAM
[0x03,0xFE]	Memory bank content ^a	a	a	Any ^a
0xFF	Reserved – not implemented	Answer NO	No change	n.a.

^a Purpose, default/power on/reset value and memory access of these bytes shall be defined by the manufacturer.

^b Reset value after "RESET MEMORY BANK".

^c Also used as power on value unless explicitly stated otherwise.

The byte in location 0x00 of each bank contains the address of the last accessible memory location of the bank. The value shall be in the range [0x03,0xFE].

The byte in location 0x01 is manufacturer specific. If implemented, the usage of this byte should be described by the manufacturer (as well as the entire content of the memory bank).

NOTE 1 It could be used for example to store a checksum in case of a memory bank with static content. Using a checksum on a memory bank where the content is changed by the control gear is not useful.

The byte in location 0x02 shall be used to lock write access. Memory location 0x02 itself shall never be locked for writing. While this memory location contains any value different from 0x55, all memory locations marked "(lockable)" of the corresponding memory bank shall be read only. The control gear shall not change the value of the lock byte other than as a

consequence of power cycle or a “RESET MEMORY BANK (*DTR0*)” or other command affecting the lock byte.

Location 0xFF is a reserved location in every memory bank, and is not accessible. This location shall not be implemented as a normal memory bank location. When addressed, the control gear shall respond as if this location is not implemented, and it shall not increment “*DTR0*”.

NOTE 2 This location is reserved in order to stop the auto increment of *DTR0*.

9.10.3 Selecting a memory bank location

In order to select a memory bank location, a combination of memory bank number and location inside the memory bank is required.

The memory bank shall be selected by setting the memory bank number in “*DTR1*”. The location in the memory bank shall be selected by the value in “*DTR0*”.

9.10.4 Memory bank reading

A selected memory bank location can be read by means of command “READ MEMORY LOCATION (*DTR1*, *DTR0*)”. The answer shall be the value of the byte at the addressed memory bank location.

If the selected memory bank is not implemented, the command shall be ignored. If the memory bank exists, and selected memory bank location is

- not implemented, or
- above the last accessible memory location,

the answer shall be NO.

If the selected memory bank location is below location 0xFF, “*DTR0*” shall be incremented by one, even if the memory location is not implemented. Otherwise, “*DTR0*” shall not change. This mechanism allows for easy consecutive reading of memory bank locations.

To ensure consistent data when reading a multi-byte value from a memory bank, it is recommended that a mechanism be implemented that latches all bytes of the multi-byte value when the first byte of the multi-byte value is read and that unlatches the bytes at any other command than “READ MEMORY LOCATION (*DTR1*, *DTR0*)”.

After reading a number of bytes from a memory bank, the application controller should check the value of “*DTR0*” to verify it is at the expected/desired location. Any mismatch indicates an error while reading.

9.10.5 Memory bank writing

Write commands are special commands and therefore not addressable. In order to select the correct control gear(s) the addressable command “ENABLE WRITE MEMORY” shall be used. Upon reception of “ENABLE WRITE MEMORY”, the addressed control gear(s) shall set “*writeEnableState*” to ENABLED.

Only while “*writeEnableState*” is ENABLED, and the addressed memory bank is implemented, the control gear shall accept the following commands to write to a selected memory bank location:

- “WRITE MEMORY LOCATION (*DTR1*, *DTR0*, *data*)”: The control gear shall confirm writing a memory location with an answer equal to the value *data*.

NOTE 1 The value that can be read from the memory bank location is not necessarily *data*.

- “WRITE MEMORY LOCATION – NO REPLY (*DTR1*, *DTR0*, *data*)”: Writing a memory location shall not cause the control gear to reply.

A control gear shall set “*writeEnableState*” to DISABLED if any command other than one of the following commands is received:

- “WRITE MEMORY LOCATION (*DTR1*, *DTR0*, *data*)”, “WRITE MEMORY LOCATION – NO REPLY (*DTR1*, *DTR0*, *data*)”
- “*DTR0* (*data*)”, “*DTR1* (*data*)”, “*DTR2* (*data*)”
- “QUERY CONTENT *DTR0*”, “QUERY CONTENT *DTR1*”, “QUERY CONTENT *DTR2*”

If the selected memory bank location is

- not implemented, or
- above the last accessible memory location, or
- locked (see subclause 9.10.2), or
- not writeable,

the answer to “WRITE MEMORY LOCATION (*DTR1*, *DTR0*, *data*)” shall be NO and no memory location shall be written to.

If the selected memory bank location is below location 0xFF, “*DTR0*” shall be incremented by one. Otherwise, “*DTR0*” shall not change. This mechanism allows for easy consecutive writing to memory bank locations.

To ensure consistent data when writing a multi-byte value into a memory bank, it is recommended that a mechanism be implemented that only accepts the new multi-byte value for writing after all bytes of the multi-byte value have been received.

After writing a number of bytes to a memory bank, the application controller should check the value of “*DTR0*” to verify it is at the expected/desired location. Any mismatch indicates an error while writing.

NOTE 2 “*DTR0*” is also incremented if a non-implemented memory bank location is addressed before 0xFF is reached.

9.10.6 Memory bank 0

Memory bank 0 contains information about the control gear. Memory bank 0 shall be implemented in all control gear.

Memory bank 0 shall be implemented using the memory map shown in Table 9, with at least the memory locations up to address 0x7F implemented, excluding reserved locations.

Table 9 – Memory map of memory bank 0

Address	Description	Default value (factory)	Memory type
0x00	Address of last accessible memory location	factory burn-in	ROM
0x01	Reserved – not implemented	answer NO	n.a.
0x02	Number of last accessible memory bank	factory burn-in, range [0,0xFF]	ROM
0x03	GTIN byte 0 (MSB) ^a	factory burn-in	ROM
0x04	GTIN byte 1	factory burn-in	ROM
0x05	GTIN byte 2	factory burn-in	ROM
0x06	GTIN byte 3	factory burn-in	ROM
0x07	GTIN byte 4	factory burn-in	ROM
0x08	GTIN byte 5 (LSB)	factory burn-in	ROM
0x09	Firmware version (major)	factory burn-in	ROM
0x0A	Firmware version (minor)	factory burn-in	ROM
0x0B	Identification number byte 0 (MSB)	factory burn-in	ROM
0x0C	Identification number byte 1	factory burn-in	ROM
0x0D	Identification number byte 2	factory burn-in	ROM
0x0E	Identification number byte 3	factory burn-in	ROM
0x0F	Identification number byte 4	factory burn-in	ROM
0x10	Identification number byte 5	factory burn-in	ROM
0x11	Identification number byte 6	factory burn-in	ROM
0x12	Identification number byte 7 (LSB)	factory burn-in	ROM
0x13	Hardware version (major)	factory burn-in	ROM
0x14	Hardware version (minor)	factory burn-in	ROM
0x15	101 version number	factory burn-in, according to implemented version number	ROM
0x16	102 version number of all integrated control gear ^b	factory burn-in, according to implemented version number	ROM
0x17	103 version number of all integrated control devices ^b	factory burn-in, according to implemented version number	ROM
0x18	Number of logical control device units in the bus unit	factory burn-in, range [0,64]	ROM
0x19	Number of logical control gear units in the bus unit	factory burn-in, range [1,64]	ROM
0x1A	Index number of this logical control gear unit	factory burn-in, range [0,(location 0x19)-1]	ROM
[0x1B,0x7F]	Reserved – not implemented	answer NO	n.a.
[0x80,0xFE]	Additional control gear information ^c	^c	ROM
0xFF	Reserved – not implemented	answer NO	n.a.

^a It is recommended that the product GTIN is not re-used within the expected lifetime of the product after installation.

^b Format of the version number is defined in Subclause 4.2. If not implemented, this is indicated by 0xFF.

^c Purpose and (default) value of these bytes shall be defined by the manufacturer.

If there is more than one logical unit built into one bus unit, all logical units shall have the same values in memory bank locations 0x03 up to and including 0x19.

A bus unit might contain both control gear and control devices. They share various numbers (e.g. GTIN, unique identification number...). To avoid problems when reading, and getting different answers depending on the addressing scheme used, the memory bank layout are the same for control gear and for control devices up to and including location 0x19. The data shall be the same as well. The application controller can use either the 102 or the 103 commands to identify the basic data, provided both are implemented.

The bytes in locations 0x03 to 0x08 (“GTIN 0” to “GTIN 5”) shall contain the Global Trade Item Number (GTIN), e.g. the EAN, in binary. The bytes shall be stored most significant first and filled with leading zeroes.

The bytes in locations 0x09 and 0x0A (“firmware version”) shall contain the firmware version of the bus unit.

The bytes in locations 0x0B to 0x12 (“identification number byte 0” to “identification number byte 7”) shall contain 64 bits of an identification number of the bus unit, preferably the serial number. The identification number shall be stored with least significant byte in “identification number byte 7” and unused bits shall be filled with 0.

The combination of the identification number and the GTIN number shall be unique.

The byte in location 0x13 and 0x14 (“hardware version”) shall contain the hardware version of the bus unit.

The byte in location 0x15 shall contain the implemented IEC 62386-101 version number of the bus unit.

The byte in location 0x16 shall contain the implemented IEC 62386-102 version number of the bus unit. If no control gear is implemented, the version number shall be 0xFF.

The byte in location 0x17 shall contain the implemented IEC 62386-103 version number of the bus unit. If no control device is implemented, the version number shall be 0xFF.

The byte in location 0x18 shall contain the number of logical control device units integrated into the bus unit. The number of logical units shall be in the range of 0 to 64.

The byte in location 0x19 shall contain the number of logical control gear units integrated into the bus unit. The number of logical units shall be in the range of 1 to 64.

The byte in location 0x1A shall represent the unique index number of the logical control gear unit that implements that memory bank. The valid range of this index number is 0 to the total number of logical control gear units in the bus unit minus one.

NOTE As example there might be a product containing three logical devices with three different short addresses. Each of these control gear has the same GTIN and identification number, each reports as number of devices the value 3 and the index of the three control gear is reported as 0, 1 or 2 respectively. Reading location 0x1A using broadcast yields a backward frame according to IEC 62386-101:2014, Subclause 9.5.2 (overlapping backward frame).

9.10.7 Memory bank 1

Memory bank 1 is reserved for use by an OEM (original equipment manufacturer, e.g. a luminaire manufacturer) to store additional information, which has no impact on the functionality of the control gear. The control gear manufacturer may implement memory bank 1.

If implemented, memory bank 1 shall at least implement the memory locations up to and including address 0x10. The fixed usage for location 0x00 to 0x02 and the recommended memory map usage for location 0x03 to 0x10 is shown in Table 10.

Table 10 – Memory map of memory bank 1

Address	Description	Default value (factory)	RESET value ^b	Memory type
0x00	Address of last accessible memory location	factory burn-in, range [0x10,0xFE]	no change	ROM
0x01	Indicator byte ^a	a	a	any ^a
0x02	Memory bank 1 lock byte. Lockable bytes in the memory bank shall be read-only while the lock byte has a value different from 0x55.	0xFF	0xFF ^c	RAM
0x03	OEM GTIN byte 0 (MSB)	0xFF	no change	NVM (lockable)
0x04	OEM GTIN byte 1	0xFF	no change	NVM (lockable)
0x05	OEM GTIN byte 2	0xFF	no change	NVM (lockable)
0x06	OEM GTIN byte 3	0xFF	no change	NVM (lockable)
0x07	OEM GTIN byte 4	0xFF	no change	NVM (lockable)
0x08	OEM GTIN byte 5 (LSB)	0xFF	no change	NVM (lockable)
0x09	OEM identification number byte 0 (MSB)	0xFF	no change	NVM (lockable)
0x0A	OEM identification number byte 1	0xFF	no change	NVM (lockable)
0x0B	OEM identification number byte 2	0xFF	no change	NVM (lockable)
0x0C	OEM identification number byte 3	0xFF	no change	NVM (lockable)
0x0D	OEM identification number byte 4	0xFF	no change	NVM (lockable)
0x0E	OEM identification number byte 5	0xFF	no change	NVM (lockable)
0x0F	OEM identification number byte 6	0xFF	no change	NVM (lockable)
0x10	OEM identification number byte 7 (LSB)	0xFF	no change	NVM (lockable)
≥ 0x11	Additional control gear information ^a	a	a	a
0xFF	Reserved – not implemented	answer NO	no change	n.a.

^a Purpose, default/power on/reset value and memory access of these bytes shall be defined by the manufacturer.

^b Reset value after “RESET MEMORY BANK”.

^c Also used as power on value.

The bytes in locations 0x03 to 0x08 (“OEM GTIN 0” to “OEM GTIN 5”) should be used to identify the product containing the control gear. If the bytes are used for GTIN the bytes shall be stored most significant bit first and filled with leading zeroes. These bytes should be programmed by the OEM.

The bytes in locations 0x09 to 0x10 (“OEM identification number byte 0” to “OEM identification number byte 7”) should contain 64 bits of an identification number of the OEM product. If the bytes are used for the identification number, it shall be stored with the least significant byte in “Identification number byte 7” and unused bits shall be filled with 0. These bytes should be programmed by the OEM.

The combination of OEM GTIN and OEM identification number should be unique.

9.10.8 Manufacturer specific memory banks

The manufacturer may use additional memory banks in the range of 2 to 199 to store additional information. The memory map of additional banks shall comply with Table 8.

9.10.9 Reserved memory banks

Memory banks 200 to 255 are reserved for future use and shall not be implemented.

9.11 Reset

9.11.1 Reset operation

A control gear shall implement a reset operation to set all variables to their reset values (see Table 14).

NOTE For some variables this operation could have no effect at all.

The reset operation shall take at most 300 ms to complete. While the reset operation is in progress, the control gear may or may not respond to any command. However, until the reset operation is complete, none of the affected variables needs to have a defined value.

An application controller can trigger the reset operation using the “RESET” instruction and should wait at least 350 ms to ensure all gear have finished the reset operation.

9.11.2 Reset memory bank operation

A control gear shall implement a reset operation to set the content of all unlocked memory banks to their reset values (see 9.10), followed by locking the memory banks.

NOTE For some memory bank locations this operation could have no effect at all.

The reset operation shall take at most 10 s to complete. While the reset operation is in progress, the control gear may or may not respond to any command. However, until the reset operation is complete, none of the affected memory bank locations have a defined value.

An application controller can trigger the reset operation for a specific memory bank, or for all implemented memory banks, using the “RESET MEMORY BANK (*DTR0*)” instruction and it should wait for at least 10,1 s so as to allow all gear enough time to finish the reset memory bank operation.

9.12 System failure

If the control gear detects system failure (see IEC 62386-101:2014, Subclause 4.11) and “*systemFailureLevel*” is not MASK, “*targetLevel*” shall be calculated on the basis of “*systemFailureLevel*”. The transition from “*actualLevel*” to “*targetLevel*” shall take place immediately and the light output shall be adjusted as quickly as possible.

If “*systemFailureLevel*” is MASK, the control gear shall not react to a system failure.

On restoration of the bus idle voltage the control gear shall not react.

“*systemFailureLevel*” can be set and queried with “SET SYSTEM FAILURE LEVEL (*DTR0*)” and “QUERY SYSTEM FAILURE LEVEL” respectively.

When bus power is restored after a system failure, bus-powered control gear shall follow the power-on procedure defined in subclause 9.13 below. Consequently, the variable “*systemFailureLevel*” is not used. Nevertheless, all control gear, including bus-powered control gear, shall maintain “*systemFailureLevel*” and conform to the requirements of the specifications of all the commands relating to it.

NOTE Implementing “*systemFailureLevel*” although this variable is normally not applicable for bus powered devices, is done to avoid separate test conditions of control gear.

9.13 Power on

After an external power cycle (see IEC 62386-101 subclause 4.11.1), the device shall retain its most recent configuration, with the following exceptions:

- the memory bank write enable state shall be disabled for all memory banks and the lock byte shall be set to 0xFF;
- all running timers shall be stopped and cancelled/reset;
- “*powerCycleSeen*” shall be set to TRUE;
- “*actualLevel*” shall be set to 0x00 keeping the lamp off;
- “*lampOn*” shall be set to FALSE;
- “*limitError*” shall be set to FALSE;
- “*targetLevel*” shall be set to 0x00;
- the control gear may start preheating the lamp but the lamp shall not ignite. While preheating, “*actualLevel*” shall be kept at 0x00 contrary to normal startup activity.

Bus powered devices shall activate the power on level immediately. For externally powered devices the following holds:

If a level control command other than “GO TO SCENE (*sceneNumber*)” where the value of the scene equals MASK and other than DAPC(MASK) is received it shall be executed.

If “GO TO SCENE (*sceneNumber*)” where the value of the scene equals MASK is received, the control gear shall ignore the command and continue as if no level control command has been received.

If DAPC(MASK) is received, the control gear shall stop any startup activity.

NOTE 1 Since “*actualLevel*” = 0, this effectively keeps the lamp off.

The control gear shall activate the power on level according to Table 11 by calculating the “*targetLevel*” on the basis of “*powerOnLevel*”. If “*powerOnLevel*” equals MASK, “*targetLevel*” shall be set to “*lastLightLevel*”. “*actualLevel*” shall be set to “*targetLevel*” immediately and the light output shall be adjusted as quickly as possible.

If a level control command is received before the power on level is activated, this command shall be executed immediately and the control gear shall not activate the power on level.

Table 11 – Power on timing

Power on behavior	Minimum time	Maximum time
Lamp off		540 ms
Grey area	> 540 ms	< 660 ms
Power on level	660 ms	

NOTE 2 Thus, there is an interval during which a control device can send a level control command which will be obeyed immediately, so DAPC(0x00) or DAPC(MASK) can be used to prevent from going automatically to “powerOnLevel”.

NOTE 3 It is possible that system failure is detected before the power on level has been reached. If “systemFailureLevel” is not MASK, the “targetLevel” is recalculated on the basis of “systemFailureLevel”.

“powerOnLevel” can be set and queried with “SET POWER ON LEVEL (DTR0)” and “QUERY POWER ON LEVEL” respectively.

After receiving the first 16 bit forward frame on the interface after power-on, the control gear shall only respond to frames described in IEC 62386-101.

9.14 Assigning short addresses

9.14.1 General

“shortAddress” shall be derived from *data* or “DTR0” depending on the command used. It shall be set on receipt of “PROGRAM SHORT ADDRESS (*data*)” or “SET SHORT ADDRESS (DTR0)” as follows:

- if *data* or “DTR0” = MASK: MASK (effectively deleting the short address)
- if *data* or “DTR0” = 1xxxxxxb or xxxxxx0b: no change
- in all other cases (0AAAAAA1b): 00AAAAAAb.

9.14.2 Random address allocation

A control gear shall implement an initialisation state, only in which, in addition to the other operations identified in this standard, a set of commands are enabled that allow an application controller to detect and uniquely identify control gear available on the bus and assign short addresses to these devices.

The initialisation state is a temporary state which is entered with the command “INITIALISE (*device*)”. It shall end automatically 15 min ± 1,5 min after the last “INITIALISE (*device*)” command was received. Additionally, a power cycle or the command “TERMINATE” shall cause the control gear to leave the initialisation state immediately.

The control gear shall have three possible values for “initialisationState”:

- DISABLED, not in initialisation state;
- ENABLED, in initialisation state;
- WITHDRAWN, in initialisation state, yet identified and withdrawn.

The following (special) commands are initialisation commands:

- “RANDOMISE”, “COMPARE” and “WITHDRAW”
- “SEARCHADDRH (*data*)”, “SEARCHADDRM (*data*)” and “SEARCHADDRL (*data*)”
- “PROGRAM SHORT ADDRESS (*data*)”, “VERIFY SHORT ADDRESS (*data*)” and “QUERY SHORT ADDRESS”
- “IDENTIFY DEVICE”

NOTE “IDENTIFY DEVICE” is by itself not an initialisation command, but typically used during initialisation

9.14.3 Identification of a device

9.14.3.1 General

During identification no variables shall be affected unless explicitly stated otherwise. Where appropriate, variables can be temporarily ignored, so that after the identification has ended, there are no side effects.

When identification is active, the light output may be at any level between off and 100 %, “*minLevel*” and “*maxLevel*” as well as “*actualLevel*” being in effect temporarily ignored.

Identification shall be stopped upon reception of any instruction other than INITIALISE (*device*), RECALL MIN LEVEL, RECALL MAX LEVEL or IDENTIFY DEVICE.

After identification has stopped, the light output shall be adjusted as quickly as possible to reflect “*actualLevel*” and the command shall be executed (if applicable).

9.14.3.2 Method one: single instruction

Identification can be started by sending the instruction “IDENTIFY DEVICE”. This shall start or restart a $10\text{ s} \pm 1\text{ s}$ timer. While the timer is running, a procedure enabling an observer to identify the selected control gear shall run. If the timer expires, identification shall stop.

NOTE The actual procedure is manufacturer specific.

While identification is active, the control gear shall, without interrupting the identification procedure:

- on RECALL MIN LEVEL: set “*actualLevel*” and “*targetLevel*” to “*minLevel*”;
- on RECALL MAX LEVEL: set “*actualLevel*” and “*targetLevel*” to “*maxLevel*”.

When identification is stopped by an application controller, the corresponding timer shall be cancelled immediately.

For examples of how to use the commands, see Annex A.

9.14.3.3 Method two: using “RECALL MAX LEVEL” and/or “RECALL MIN LEVEL” (deprecated)

While “*initialisationState*” is not DISABLED, the control gear shall:

- on RECALL MIN LEVEL: set “*actualLevel*” and “*targetLevel*” to “*minLevel*”, and then adjust the light output as quickly as possible to its PHM level. If, however, PHM is not visibly significantly different from 100 %, then the lamp shall be temporarily switched off instead;
- on RECALL MAX LEVEL: set “*actualLevel*” and “*targetLevel*” to “*maxLevel*”, and then adjust the light output as quickly as possible to 100 %.

If the device is unable to visually identify itself in this way, the control gear shall respond as if it received “IDENTIFY DEVICE” as well, starting or re-triggering the identification procedure.

NOTE It is acceptable for the process of identifying individual control gear to depend upon both commands being received in an alternating sequence.

Identification shall be stopped immediately when one of the following conditions hold:

- the “*initialisationState*” changes to DISABLED;
- upon reception of any instruction other than INITIALISE (*device*), RECALL MIN LEVEL, RECALL MAX LEVEL or IDENTIFY DEVICE.

For examples of how to use the commands, see Annex A.

9.14.4 Direct address allocation

“SET SHORT ADDRESS (*DTR0*)” can be used to directly program a short address to the addressed gear.

9.15 Failure state behaviour

If the control gear is in a failure state, in which operation of the lamp(s) is not possible as intended (lamp failure and/or control gear failure) it shall react to level commands in the following way:

The control gear shall calculate “*targetLevel*” in accordance with the commands received, and control the lamp insofar as that is practicable. As a consequence of the fault, the normal relationship between “*actualLevel*” and light output could temporarily change.

NOTE For example, a control gear might, on detecting an excessively high temperature, protect itself from the risk of thermal damage by limiting the light output.

If the failure state is resolved, the control gear shall re-establish the normal relationship between “*actualLevel*” and light output.

9.16 Status information

9.16.1 General

Each control gear shall expose its status as a combination of device properties as given in Table 12.

Table 12 – Control gear status

Bit	Description	Value	See
0	“ <i>controlGearFailure</i> ” is TRUE?	“1” = “YES”	9.16.2
1	“ <i>lampFailure</i> ” is TRUE?	“1” = “YES”	9.16.3
2	“ <i>lampOn</i> ” is TRUE?	“1” = “YES”	9.16.4
3	“ <i>limitError</i> ” is TRUE?	“1” = “YES”	9.16.5
4	“ <i>fadeRunning</i> ” is TRUE?	“1” = “YES”	9.16.6
5	“ <i>resetState</i> ” is TRUE?	“1” = “YES”	9.16.7
6	“ <i>shortAddress</i> ” is MASK?	“1” = “YES”	9.16.8
7	“ <i>powerCycleSeen</i> ” is TRUE?	“1” = “YES”	9.16.9

The device status can be queried using “QUERY STATUS”. The bits shall reflect the actual situation without delay unless explicitly stated otherwise.

9.16.2 Bit 0: Control gear failure

A control gear failure according to this standard is a situation in which the control gear cannot operate as intended.

NOTE Examples are mains under voltage, over temperature, unexpected watchdog timers firing etc.

If a control gear failure is detected, “*controlGearFailure*” shall be set to TRUE.

If the failure is no longer detected, and normal operation has been resumed, “*controlGearFailure*” shall be set to FALSE.

Control gear failure shall be detected and indicated latest after 30 s.

9.16.3 Bit 1: lamp failure

A lamp failure according to this standard is a situation in which the lamp cannot be operated as intended due to e.g. incorrect lamp connection or lamp defects.

If a lamp failure is detected, *“lampFailure”* shall be set to TRUE. Lamp failure shall be detected and indicated latest after 30 s when the control gear is not in standby (see 9.2).

Partial lamp failure should also be interpreted as lamp failure.

If *“lampFailure”* is TRUE, the control gear shall periodically check to determine whether the lamp situation has improved. This check shall be executed at least whenever *“targetLevel”* changes from 0x00 to a greater value. After a successful startup, *“lampFailure”* shall be set to FALSE.

For lamp type unknown there may be support for this bit. For lamp type none there may be support (e.g. based on load measurement).

9.16.4 Bit 2: lamp on

“lampOn” shall be set to FALSE when the lamp is off, during startup, and in case of total lamp failure, meaning no light output. In all other cases it shall be set to TRUE.

9.16.5 Bit 3: limit error

If the last requested target level has been modified in accordance with *“minLevel”* or *“maxLevel”* limitations, or *“targetLevel”* has been modified due to a change of *“minLevel”* or *“maxLevel”*, *“limitError”* shall be set to TRUE.

If the last target level requested by *“DAPC (level)”* equals *“MASK”*, *“limitError”* shall not change.

In all other cases *“limitError”* shall be set to FALSE.

9.16.6 Bit 4: fade running

“fadeRunning” shall be set to FALSE except for the time during which the fade timer is running. *“fadeRunning”* shall be set to TRUE from the beginning of the fade (after startup) until the end of the fade time, regardless of whether *“targetLevel”* and *“actualLevel”* reach the same level.

9.16.7 Bit 5: reset state

“resetState” shall be set to TRUE if all the NVM variables mentioned in Table 14 except *“lastLightLevel”* are at their reset value. The NVM variables that are marked with ‘no change’ in the reset value column shall not be considered. NVM variables defined in implemented Parts 2xx shall be included.

In all other cases the bit shall be set to FALSE.

9.16.8 Bit 6: missing short address

This bit indicates whether a short address has been assigned to the gear, by checking *“shortAddress”*. The bit shall be TRUE if *“shortAddress”* = MASK.

In all other cases the bit shall be set to FALSE.

9.16.9 Bit 7: power cycle seen

“*powerCycleSeen*” shall be set to TRUE after an external power cycle (see IEC 62386-101, Clause 4.11) has occurred.

“*powerCycleSeen*” shall be set to FALSE once one of the following commands has been received:

“RESET”, “DAPC (*level*)”, “OFF”, “UP”, “DOWN”, “STEP UP”, “STEP DOWN”, “RECALL MAX LEVEL”, “RECALL MIN LEVEL”, “GO TO LAST ACTIVE LEVEL”, “STEP DOWN AND OFF”, “ON AND STEP UP”, “GO TO SCENE (*sceneNumber*)”.

9.17 Non-volatile memory

Physical non-volatile memory typically supports a limited number of write cycles. Since many variables are NVM type, the physical limitations need some attention.

A control gear should store NVM variables in such a way that their content is never lost and the intended lifetime of the device can be reached. This means that it may not be possible to physically write every change in a variable immediately. There may be situations in which the control gear is not able to physically write the variables to NVM, especially if a particular NVM variable is changed very frequently.

Since the application controller cannot know the control gear's internal mechanism for physically saving persistent variables, the instruction “SAVE PERSISTENT VARIABLES” is defined to force the control gear to physically write all variables of type NVM to memory. This command is an addition to the normal writing of NVM variables. Its intended use is to ensure that important changes made by an application controller cannot be lost, e.g. after assigning all short addresses or setting other important (and stable) configuration data. Clearly it is not intended to be used after every level change. Typically, this command is used only a handful of times for an entire installation.

NOTE 1 Typically the command can be used a few thousand times before causing physical damage to the control gear's NVM.

Physically saving the variables in response to the instruction shall take at most 300 ms to complete. While the saving operation is on-going, the light output may fluctuate and the control gear may or may not respond to any command. However, until the operation is complete, the value of the affected variables may be undefined. Moreover, if the light is off when the instruction is received, the light shall stay off; in this case, no flicker shall be visible.

The light output may not fluctuate during saving operations unless these are triggered by this command.

An application controller can trigger the save operation using the “SAVE PERSISTENT VARIABLES” instruction and should wait at least 350 ms to ensure all gear have finished the operation.

9.18 Device types and features

Commands with their opcode in the range 0xE0 to 0xFF are reserved for special device types or features. Each device type/feature re-defines these commands, except for the command with opcode 0xFF (“QUERY EXTENDED VERSION NUMBER”).

The device type/feature specific command set can be selected by the instruction “ENABLE DEVICE TYPE (*data*)”.

This instruction shall select the device type/feature for which only the next following application extended command (refer to subclauses 11.6) is valid. Receiving this instruction shall cancel any previous selection of a device type.

The enabling of the device type/feature shall be cancelled upon execution of the next following command addressed to the same control gear, and that command shall be executed according to its specification, regardless of whether it is an application extended command or not.

A control gear shall not react to a command which belongs to the application extended commands of a device type/feature not supported by this control gear.

The device types shall be coded as specified in the particular parts 2xx of IEC 62386.

An application controller can check which device types are supported by the control gear. "QUERY DEVICE TYPE" reports the supported device type. If more than one device type/feature is supported, "QUERY DEVICE TYPE" reports MASK. In that case, the application controller can check all supported device types by repeating "QUERY NEXT DEVICE TYPE" until 254 is received as an answer. Issuing "QUERY DEVICE TYPE" automatically ensure that the first supported device type/feature will be reported by "QUERY NEXT DEVICE TYPE".

To check the version number of the supported device types, the application controller can send "ENABLE DEVICE TYPE (*data*)" followed by "QUERY EXTENDED VERSION NUMBER". This will report the version number of that specific device type/feature implementation.

Application controllers should be able to identify individual gear and store the relationship between gear's individual address and its device types.

9.19 Using scenes

A control gear shall support the use of 16 scenes. The following commands shall be supported:

"GO TO SCENE (*sceneNumber*)", "REMOVE FROM SCENE (*sceneX*)",
"QUERY SCENE LEVEL (*sceneX*)", and "SET SCENE (*DTR0*, *sceneX*)".

These commands actually comprise 16 commands each, one for each scene. This is accomplished by selecting a block of 16 consecutive opcodes. The number of the scene to be used can thus easily be calculated.

Upon reception of one of the scene commands, *sceneNumber* shall be derived from the opcode: $sceneNumber = opcode - opcodeBase$. This identifies the scene to be used. The opcodeBase can be found in Table 13.

Table 13 – Scenes

Command	opcodeBase	Opcode range
GO TO SCENE (<i>sceneNumber</i>)	0x10	[0x10,0x1F]
REMOVE FROM SCENE (<i>sceneX</i>)	0x50	[0x50,0x5F]
QUERY SCENE LEVEL (<i>sceneX</i>)	0xB0	[0xB0,0xBF]
SET SCENE (<i>DTR0</i> , <i>sceneX</i>)	0x40	[0x40,0x4F]

The "*sceneX*" variable also stands for 16 individual variables, where *X* equals *sceneNumber* in the range of [0,15].

On receiving command “GO TO SCENE (*sceneNumber*)” the reaction of the control gear shall depend upon the current value of “*sceneX*”, where X is derived from *sceneNumber*. If “*sceneX*” equals MASK, “*targetLevel*” shall not be affected. Otherwise, the control gear shall behave exactly as if “DAPC (*level*)” had been received with level equal to “*sceneX*”.

NOTE Using “DAPC (*level*)” implies the transition is made using the set fade time.

10 Declaration of variables

The default values, the reset values, power on values, the range of validity and the type of memory of the defined variables shall be as given in Table 14.

The variables that are declared in this clause shall not be made available for writing through a memory bank.

Table 14 – Declaration of variables

VARIABLE	DEFAULT VALUE (factory)	RESET VALUE	POWER ON VALUE	RANGE OF VALIDITY	MEMORY TYPE
“ <i>actualLevel</i> ”	^a	0xFE	0x00	0, [“ <i>minLevel</i> ”, “ <i>maxLevel</i> ”]	RAM
“ <i>targetLevel</i> ”	^a	0xFE	See 9.13 Power on	0, [“ <i>minLevel</i> ”, “ <i>maxLevel</i> ”]	RAM
“ <i>lastActiveLevel</i> ”	^a	0xFE	“ <i>maxLevel</i> ”	[“ <i>minLevel</i> ”, “ <i>maxLevel</i> ”]	RAM
“ <i>lastLightLevel</i> ”	0xFE	0xFE ^c	no change	0, [“ <i>minLevel</i> ”, “ <i>maxLevel</i> ”]	NVM
“ <i>powerOnLevel</i> ”	0xFE	0xFE	no change	[0,0xFF]	NVM
“ <i>systemFailureLevel</i> ”	0xFE	0xFE	no change	[0,0xFF]	NVM
“ <i>minLevel</i> ”	PHM	PHM	no change	[PHM, “ <i>maxLevel</i> ”]	NVM
“ <i>maxLevel</i> ”	0xFE	0xFE	no change	[“ <i>minLevel</i> ”, 0xFE]	NVM
“ <i>fadeRate</i> ”	7	7	no change	[1,0xF]	NVM
“ <i>fadeTime</i> ”	0	0	no change	[0,0xF]	NVM
“ <i>extendedFadeTimeBase</i> ”	0	0	no change	[0,1111b]	NVM
“ <i>extendedFadeTimeMultiplier</i> ”	0	0	no change	[0,100b]	NVM
“ <i>shortAddress</i> ”	MASK (no address)	no change	no change	[0,63], MASK	NVM
“ <i>searchAddress</i> ”	^a	0xFF FF FF	0xFF FF FF	[0,0xFF FF FF]	RAM
“ <i>randomAddress</i> ”	0xFF FF FF	0xFF FF FF	no change	[0,0xFF FF FF]	NVM
“ <i>operatingMode</i> ”	factory burn-in	no change	no change	0,[0x80,0xFF]	NVM
“ <i>initialisationState</i> ”	^a	no change	DISABLED	[ENABLED, DISABLED, WITHDRAWN]	RAM
“ <i>writeEnableState</i> ”	^a	DISABLED	DISABLED	[ENABLED, DISABLED]	RAM
“ <i>controlGearFailure</i> ”	^a	^b	FALSE ^d	[TRUE, FALSE]	RAM
“ <i>lampFailure</i> ”	^a	^b	FALSE ^d	[TRUE, FALSE]	RAM
“ <i>lampOn</i> ”	^a	^b	FALSE	[TRUE, FALSE]	RAM
“ <i>limitError</i> ”	^a	FALSE	FALSE ^d	[TRUE, FALSE]	RAM
“ <i>fadeRunning</i> ”	^a	FALSE	FALSE	[TRUE, FALSE]	RAM

VARIABLE	DEFAULT VALUE (factory)	RESET VALUE	POWER ON VALUE	RANGE OF VALIDITY	MEMORY TYPE
<i>“resetState”</i>	TRUE	TRUE	TRUE ^d	[TRUE, FALSE]	RAM
<i>“powerCycleSeen”</i>	^a	FALSE	TRUE	[TRUE, FALSE]	RAM
<i>“gearGroups”</i>	0x00 00 (no group)	0x00 00 (no group)	no change	[0,0xFF FF]	NVM
<i>“sceneX”</i> ^e	MASK	MASK	no change	[0,0xFF]	NVM
<i>“DTR0”</i>	^a	no change	0x00	[0,0xFF]	RAM
<i>“DTR1”</i>	^a	no change	0x00	[0,0xFF]	RAM
<i>“DTR2”</i>	^a	no change	0x00	[0,0xFF]	RAM
PHM	factory burn-in	no change	no change	[1,0xFE]	ROM

^a Not applicable.

^b The value could change as a consequence of the RESET command execution.

^c This NVM variable is excluded for *“resetState”*.

^d The value should reflect the actual situation as soon as possible.

^e X is in the range 0x0 to 0xF, effectively there is one variable for each of the 16 scenes.