

JEDEC PUBLICATION

PartModel Electrical Guidelines for Electronic-Device Packages – XML Requirements

JEP30-E100I

(Revision of JEP30-E100H, June 2025)

September 2025

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION



NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

All risk and liability relating to the use of JEDEC standards is assumed by the user, who agrees to indemnify and hold JEDEC harmless.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

Copyright ©JEDEC Solid State Technology Association 2025. All rights reserved.

JEDEC retains the copyright on this material. By downloading this file the individual agrees not to charge for or resell the resulting material.

PRICE: Contact JEDEC
3103 10th Street North, Suite 240S, Arlington, VA 22201

**DO NOT VIOLATE
THE
LAW!**

This document is copyrighted by JEDEC and may not be reproduced without permission.

For information, contact:

JEDEC Solid State Technology Association
3103 10th Street North
Suite 240S
Arlington, VA 22201
<https://www.jedec.org/contact>

This page intentionally left blank.

PartModel Electrical Guidelines for Electronic-Device Packages – XML Requirements

Contents

	Page	
1	Scope.....	1
1.1.	PURPOSE	1
2	Applicable Documents	2
2.1.	JEDEC (WWW.JEDEC.ORG).....	2
2.2.	CHIPS ALLIANCE (HTTPS://CHIPSALLIANCE.ORG/)	3
2.3.	IEEE (WWW.IEEE.ORG)	3
2.4.	IEEE/ANSI/CSA (WWW.ANSI.ORG)	4
2.5.	IEC (WWW.IEC.ORG)	4
2.6.	IPC (WWW.IPC.ORG)	4
2.7.	INCITIS (WWW.INCITIS.ORG)	4
2.8.	MIPI (WWW.MIPI.ORG).....	4
2.9.	OIF (WWW.OIFORUM.COM)	4
2.10.	OPEN COMPUTE PROJECT (HTTPS://WWW.OPENCOMPUTE.ORG/)	5
2.11.	HDMI LICENSING, LLC (WWW.HDMI.COM)	5
2.12.	PCI-SIG (WWW.PCISIG.COM).....	5
2.13.	HP (WWW.HP.COM).....	5
2.14.	SDA (WWW.SDCARD.ORG)	5
2.15.	SMI (WWW.POWERSIG.ORG)/.....	5
2.16.	USB-IF (WWW.USB.ORG)	5
2.17.	UCIE (HTTPS://WWW.UCIEXPRESS.ORG/)	5
2.18.	ACCELLERA (HTTPS://WWW.ACCELLERA.ORG/).	5
2.19.	AMERICAN MATHEMATICAL SOCIETY	5
2.20.	MATHML - HTTPS://WWW.W3.ORG/TR/MATHML4/#INTRO_OVERVIEW.....	5
3	Requirements.....	6
3.1.	TERMS AND DEFINITIONS	6
3.2.	XML SCHEMA KEY TERMS AND DEFINITIONS.....	6
4	PartModel Schema Definition	7
4.1.	PARTMODEL -> ELECTRICAL SECTION.....	7
4.2.	MANUFACTURER PART NUMBER-ARRAY.....	9
4.3.	STANDARDS IDENTIFIER - ARRAY	10
4.4.	PROCESS TECHNOLOGY IDENTIFIER - ARRAY	11
4.5.	LINKING THE MANUFACTURING PART NUMBER TO A SPECIFIC ELECTRICAL DATA SET	12
4.5.1.	Linking the Manufacturing Part Number to Electrical Parameters.....	13
4.5.2.	Linking the Manufacturing Part Number to Part Classification	14
4.5.3.	Linking the Manufacturing Part Number to Terminal Details.....	15
4.5.4.	Linking the Manufacturing Part Number to Super Interface	16
4.5.5.	Linking the Manufacturing Part Number to Functions	17
4.5.6.	Linking the Manufacturing Part Number to Electrical Specification.....	18
4.5.7.	Linking the Manufacturing Part Number to Truth Table	19
4.5.8.	Linking the Manufacturing Part Number to ESD.....	20
4.5.9.	Linking the Manufacturing Part Number to Schematic Data	21
4.5.10.	Linking the Manufacturing Part Number to Symbol	22
4.5.11.	Linking the Manufacturing Part Number to Required Circuitry	23
4.5.12.	Linking the Manufacturing Part Number to Mapping data	24
4.5.13.	Linking the Manufacturing Part Number to Electrical Map data	25
4.5.14.	Linking the Manufacturing Part Number to Terminal Map data.....	26
4.5.15.	Linking the Manufacturing Part Number to Package Terminal Map data.....	27
4.5.16.	Linking the Manufacturing Part Number to Die Terminal Map data	28
4.5.17.	Linking the Manufacturing Part Number to Simulation Map data	29
4.5.18.	Linking the Manufacturing Part Number to Simulation Models	30

Contents (cont'd)

	Page	
4.5.19.	Linking the Manufacturing Part Number to Reference Design	31
4.5.20.	Linking the Manufacturing Part Number to Software Interface Description	32
4.6.	ELECTRICAL SECTION.....	33
4.7.	ELECTRICAL PARAMETERS	34
4.7.1.	Part Classification - Array	35
4.7.1.1.	Cable and Wiring Classification.....	38
4.7.1.2.	Connector Classification	39
4.7.1.2.1.	CardEdge Classification	40
4.7.1.2.1.1.	CardEdge Interface Connector Classification.....	41
4.7.1.2.1.2.	CardEdge Overflow Categories Group	42
4.7.1.2.2.	Optical Classification	42
4.7.1.2.3.	Socket Classification.....	43
4.7.1.2.3.1.	PCB-Mountable Socket Classification	43
4.7.1.2.3.2.	Non-PCB-Mountable Socket Classification	44
4.7.1.2.3.3.	Socket Overflow Categories Group	44
4.7.1.2.4.	Board Classification	45
4.7.1.2.4.1.	Electrical Rating	46
4.7.1.2.4.2.	Backplane Classification	46
4.7.1.2.4.2.1.	Board Backplane Connector Classification Property – Array Type	47
4.7.1.2.4.3.	Interface Classification.....	48
4.7.1.2.4.4.	Power And Signal Classification	48
4.7.1.2.4.5.	Power And Signal Classification	49
4.7.1.2.4.6.	Signal Classification.....	49
4.7.1.2.4.7.	Board Overflow Categories.....	50
4.7.1.2.5.	Cable Classification	51
4.7.1.2.5.1.	Power And Signal Classification	52
4.7.1.2.5.2.	Power Classification	52
4.7.1.2.5.3.	Signal Classification.....	53
4.7.1.2.5.4.	Cable Overflow Categories Type.....	53
4.7.1.3.	Electrical Classification	54
4.7.1.3.1.	Amplifier Classification.....	55
4.7.1.3.2.	Audio Classification	56
4.7.1.3.3.	Battery Classification	57
4.7.1.3.3.1.	Rechargeable Battery Classification.....	57
4.7.1.3.3.1.1.	Rechargeable Battery Material Property	58
4.7.1.3.3.2.	Non-rechargeable Battery Classification	58
4.7.1.3.3.2.1.	Non-rechargeable Battery Material Property	59
4.7.1.3.3.3.	Material Overflow Categories Group	59
4.7.1.3.4.	Capacitor Classification	60
4.7.1.3.4.1.	Ceramic Capacitor Classification and Property-Array	61
4.7.1.3.4.1.1.	Ceramic Capacitor Class Property	61
4.7.1.3.4.1.2.	Ceramic Capacitor Dielectric Property	62
4.7.1.3.4.2.	Electrolytic Capacitor Classification and Property-Array	63
4.7.1.3.4.2.1.	Electrolytic Capacitor Material Property	63
4.7.1.3.4.2.2.	Electrolytic Capacitor Electrolyte Property	64
4.7.1.3.4.3.	Film Capacitor Classification and Property-Array	65
4.7.1.3.4.3.1.	Film Capacitor Dielectric Property	65
4.7.1.3.4.4.	Silicon Capacitor Classification.....	66
4.7.1.3.4.5.	Super-cap Capacitor Classification.....	66
4.7.1.3.4.5.1.	Super-cap Capacitor Class Property	66
4.7.1.3.4.6.	Variable Capacitor Classification and Property-Array	67
4.7.1.3.4.7.	Capacitor Overflow Categories Group.....	67

Contents (cont'd)

	Page	
4.7.1.3.5.	Circuit Protection Classification	68
4.7.1.3.5.1.	Over Current Circuit Protection Classification	68
4.7.1.3.5.1.1.	Resettable Over Current Circuit Protection Classification.....	69
4.7.1.3.5.1.2.	Resettable Over Current Circuit Protection Technology Property.....	69
4.7.1.3.5.1.3.	Non-Resettable Over Current Circuit Protection Classification	69
4.7.1.3.5.1.4.	Non-Resettable Over Current Circuit Protection Technology Property	70
4.7.1.3.5.2.	Over Voltage Circuit Protection Classification	70
4.7.1.3.5.2.1.	Over Voltage Circuit Protection Varistor Classification	71
4.7.1.3.5.2.2.	Over Voltage Circuit Protection Varistor Type Property	71
4.7.1.3.5.3.	Over Temperature Circuit Protection Classification.....	72
4.7.1.3.6.	Data Converter Classification	73
4.7.1.3.7.	Diode Classification	74
4.7.1.3.7.1.	ESD Diode Classification.....	75
4.7.1.3.7.2.	LED Classification.....	75
4.7.1.3.7.3.	Rectifier Diode Classification	76
4.7.1.3.7.4.	Schottky Diode Classification	76
4.7.1.3.7.5.	Signal Diode Classification	76
4.7.1.3.7.6.	Silicon Carbide Diode Classification.....	77
4.7.1.3.7.7.	Array Overflow Categories Group.....	77
4.7.1.3.8.	Filter Classification.....	78
4.7.1.3.8.1.	EMI-RFI Filter Classification	79
4.7.1.3.8.2.	Filter Classification Property - Array	80
4.7.1.3.9.	Frequency Source Classification	81
4.7.1.3.10.	IC Classification	82
4.7.1.3.11.	Inductor Classification.....	83
4.7.1.3.11.1.	Ferrite Inductor Classification	84
4.7.1.3.11.1.1.	Variable Ferrite Inductor Property-Array.....	84
4.7.1.3.11.1.2.	Ferrite Inductor Property-Array	85
4.7.1.3.11.2.	MetalAlloys Inductor Classification	86
4.7.1.3.11.2.1.	Variable.....	86
4.7.1.3.11.2.2.	Property - Array.....	87
4.7.1.3.11.3.	Inductor Property - Array	87
4.7.1.3.11.3.1.	Inductor Shielding Property	88
4.7.1.3.11.3.2.	Inductor Technology Property.....	88
4.7.1.3.12.	Memory Classification.....	89
4.7.1.3.12.1.	Volatile Memory Classification.....	90
4.7.1.3.12.2.	NonVolatile Memory Classification	91
4.7.1.3.13.	Optoelectronics Classification.....	92
4.7.1.3.13.1.	Display Classification	93
4.7.1.3.13.1.1.	TFT Display Classification Type	93
4.7.1.3.13.1.2.	LED Display Classification Type	94
4.7.1.3.13.1.3.	Organic Display Classification	94
4.7.1.3.13.2.	Photoemitter Classification	95
4.7.1.3.13.2.1.	Photoemitter LED Classification	95
4.7.1.3.13.3.	Photosensitive Classification	96
4.7.1.3.13.3.1.	Photodiode Photosensitive Classification	97
4.7.1.3.13.3.2.	Phototransistor Photosensitive Classification	97
4.7.1.3.13.4.	Optocoupler Classification	98
4.7.1.3.13.4.1.	Photodiode Optocoupler Classification.....	99
4.7.1.3.13.4.2.	Phototransistor Optocoupler Classification.....	99
4.7.1.3.14.	Regulator Classification	100
4.7.1.3.14.1.	Linear Current Regulator Classification	101
4.7.1.3.14.2.	Linear Current Regulator Property-Array.....	101

Contents (cont'd)

	Page
4.7.1.3.14.3.	102
4.7.1.3.14.4.	102
4.7.1.3.14.5.	103
4.7.1.3.14.5.1.	104
4.7.1.3.14.5.2.	105
4.7.1.3.15.	106
4.7.1.3.15.1.	107
4.7.1.3.15.2.	107
4.7.1.3.15.3.	108
4.7.1.3.16.	108
4.7.1.3.16.1.	109
4.7.1.3.16.2.	110
4.7.1.3.16.3.	111
4.7.1.3.16.4.	112
4.7.1.3.17.	113
4.7.1.3.17.1.	114
4.7.1.3.17.2.	115
4.7.1.3.18.	116
4.7.1.3.19.	117
4.7.1.3.19.1.	118
4.7.1.3.20.	119
4.7.1.3.20.1.	120
4.7.1.3.20.2.	120
4.7.1.3.20.3.	121
4.7.1.3.21.	121
4.7.1.3.22.	122
4.7.1.3.22.1.	123
4.7.1.3.22.2.	124
4.7.1.3.22.3.	125
4.7.1.3.22.4.	126
4.7.1.3.22.5.	127
4.7.1.3.22.6.	128
4.7.1.3.22.6.1.	129
4.7.1.3.23.	130
4.7.1.4.	131
4.7.1.4.1.	132
4.7.1.5.	133
4.7.2.	134
4.7.2.1.	135
4.7.2.1.1.	139
4.7.2.1.2.	140
4.7.2.1.3.	141
4.7.2.1.4.	142
4.7.2.1.5.	143
4.7.2.1.6.	144
4.7.2.1.6.1.	144
4.7.2.1.6.1.1.	145
4.7.2.1.6.1.2.	146
4.7.2.1.6.1.3.	147
4.7.2.1.6.1.4.	148
4.7.2.1.6.2.	151
4.7.2.1.6.3.	153

Contents (cont'd)

	Page	
4.7.2.1.7.	Reference	153
4.7.2.2.	Terminal Function – Array	157
4.7.2.2.1.	Digital Function – Signal.....	158
4.7.2.2.2.	Digital Function – Terminal State.....	159
4.7.2.3.	Terminal Grouping – Array	160
4.7.2.3.1.	Same Potential Group - Array.....	161
4.7.2.3.2.	Terminal Swap - Array	162
4.7.2.3.3.	Function Swap - Array	163
4.7.2.3.4.	Internal Electrical Connection - Array	165
4.7.2.3.5.	Differential Pair - Array	166
4.7.2.3.6.	Logical Group - Array.....	169
4.7.2.3.7.	Terminal-to-Terminal Signal Path – Array	172
4.7.2.3.7.1.	Uni-directional Terminal-to-Terminal Signal Path.....	173
4.7.2.3.7.2.	Bi-directional Terminal-to-Terminal Signal Path.....	176
4.7.2.4.	External Connection - Array.....	179
4.7.2.4.1.	Decoupling	180
4.7.2.4.2.	Pullup to Power.....	181
4.7.2.4.3.	Pulldown to Ground	182
4.7.3.	Function Group - Array	183
4.7.3.1.	Super Interface - Array	187
4.7.3.2.	Amplifier	188
4.7.3.2.1.	Differential Input Differential Output	188
4.7.3.2.2.	Differential Input Single Ended Output	189
4.7.3.2.3.	Single Ended Input Differential Output	190
4.7.3.2.4.	Single Ended Input Single Ended Output	191
4.7.3.3.	Audio	191
4.7.3.3.1.	Analog Microphone	192
4.7.3.3.2.	Digital Microphone	192
4.7.3.3.3.	Speaker.....	193
4.7.3.4.	Capacitor.....	193
4.7.3.4.1.	Fixed Non Polarized	194
4.7.3.4.2.	Fixed Polarized	194
4.7.3.4.3.	Differential Non Polarized	195
4.7.3.4.4.	Feed Through	195
4.7.3.4.5.	Variable Non Polarized	196
4.7.3.5.	Diode.....	197
4.7.3.5.1.	Diode Standard Terminal Name Assignment Type	198
4.7.3.6.	Filter	199
4.7.3.7.	Frequency Source	199
4.7.3.7.1.	Generator.....	200
4.7.3.7.2.	Timer.....	201
4.7.3.7.3.	Crystal.....	202
4.7.3.7.4.	Oscillator	203
4.7.3.7.5.	Resonator	204
4.7.3.7.6.	Voltage Controlled Oscillator	205
4.7.3.8.	Fuse	206
4.7.3.9.	Inductor	206
4.7.3.9.1.	Basic Inductor Standard Terminal Name Assignment.....	207
4.7.3.9.2.	Coupled Inductor Standard Terminal Name Assignment	207
4.7.3.10.	Interface	208
4.7.3.10.1.	AIB	209
4.7.3.10.1.1.	AIB-Base.....	210
4.7.3.10.1.2.	AIB-Plus	210

Contents (cont'd)

	Page	
4.7.3.10.2.	Battery Interface Function.....	211
4.7.3.10.3.	Camera Interface Function	212
4.7.3.10.3.1.	CSI	213
4.7.3.10.3.2.	CSI-2 C-PHY-x1.....	213
4.7.3.10.3.3.	CSI-2 C-PHY-x2.....	214
4.7.3.10.3.4.	CSI-2 C-PHY-x3.....	214
4.7.3.10.3.5.	CSI-2 C-PHY-x4.....	215
4.7.3.10.3.6.	CSI-2 C-PHY-x5.....	215
4.7.3.10.3.7.	CSI-2 C-PHY-x6.....	216
4.7.3.10.3.8.	CSI-2 D-PHY-x1.....	216
4.7.3.10.3.9.	CSI-2 D-PHY-x2.....	217
4.7.3.10.3.10.	CSI-2 D-PHY-x3.....	217
4.7.3.10.3.11.	CSI-2 D-PHY-x4.....	218
4.7.3.10.3.12.	CSI-2 D-PHY-x5.....	218
4.7.3.10.3.13.	CSI-2 D-PHY-x6.....	219
4.7.3.10.3.14.	CSI-2 D-PHY-x7.....	219
4.7.3.10.3.15.	CSI-2 D-PHY-x8.....	220
4.7.3.10.3.16.	CSI-3.....	220
4.7.3.10.3.1.	CPI	221
4.7.3.10.4.	Compute Express Link Function.....	221
4.7.3.10.4.1.	CXL-x4	222
4.7.3.10.4.2.	CXL-x8	223
4.7.3.10.4.3.	CXL-x16	224
4.7.3.10.5.	Display Bus Function	225
4.7.3.10.5.1.	DBI-TypeA	226
4.7.3.10.5.2.	DBI-TypeB	227
4.7.3.10.5.3.	DBI-TypeC	228
4.7.3.10.5.4.	DPI-Type1	228
4.7.3.10.5.5.	DPI-Type2-3.....	229
4.7.3.10.5.6.	DPI-Type4.....	230
4.7.3.10.5.7.	DSI-2-OptionC	231
4.7.3.10.5.8.	DSI-2-OptionD	232
4.7.3.10.6.	DDR3 Interface Function	233
4.7.3.10.6.1.	DDR3-x4	234
4.7.3.10.6.2.	DDR3-x4 – Dual Die	235
4.7.3.10.6.3.	DDR3-x4 – Quad Die	236
4.7.3.10.6.4.	DDR3-x8	237
4.7.3.10.6.5.	DDR3-x8 – Dual Die	238
4.7.3.10.6.6.	DDR3-x8 – Quad Die	239
4.7.3.10.6.7.	DDR3-x16	240
4.7.3.10.6.8.	DDR3-x16 – Dual Die	241
4.7.3.10.6.9.	DDR3-x16 – Quad Die	242
4.7.3.10.6.10.	DDR3 Controller.....	243
4.7.3.10.7.	DDR4 Interface Function	245
4.7.3.10.7.1.	DDR4-x4	246
4.7.3.10.7.2.	DDR4-x4 – Dual Die	247
4.7.3.10.7.3.	DDR4-x8	248
4.7.3.10.7.4.	DDR4-x8 Dual Die	249
4.7.3.10.7.5.	DDR4-x16	250
4.7.3.10.7.6.	DDR4-x16 – Dual Die	251
4.7.3.10.7.7.	DDR4-x32	252
4.7.3.10.7.8.	DDR4-x72	253

Contents (cont'd)

	Page
4.7.3.10.7.9.	254
4.7.3.10.7.10.	255
4.7.3.10.7.11.	256
4.7.3.10.7.12.	257
4.7.3.10.7.13.	259
4.7.3.10.7.14.	260
4.7.3.10.8.	261
4.7.3.10.8.1.	262
4.7.3.10.8.2.	263
4.7.3.10.8.3.	264
4.7.3.10.8.4.	265
4.7.3.10.8.5.	266
4.7.3.10.8.6.	267
4.7.3.10.8.7.	268
4.7.3.10.9.	269
4.7.3.10.9.1.	269
4.7.3.10.10.	270
4.7.3.10.11.	270
4.7.3.10.12.	271
4.7.3.10.13.	272
4.7.3.10.14.	273
4.7.3.10.14.1.	273
4.7.3.10.14.2.	274
4.7.3.10.14.3.	274
4.7.3.10.15.	275
4.7.3.10.16.	275
4.7.3.10.17.	276
4.7.3.10.17.1.	277
4.7.3.10.17.2.	288
4.7.3.10.17.3.	299
4.7.3.10.17.4.	311
4.7.3.10.17.4.1.	312
4.7.3.10.18.	318
4.7.3.10.18.1.	319
4.7.3.10.18.2.	319
4.7.3.10.18.3.	320
4.7.3.10.18.4.	320
4.7.3.10.18.5.	321
4.7.3.10.19.	321
4.7.3.10.20.	322
4.7.3.10.21.	322
4.7.3.10.22.	323
4.7.3.10.23.	323
4.7.3.10.24.	324
4.7.3.10.25.	324
4.7.3.10.26.	325
4.7.3.10.26.1.	325
4.7.3.10.26.2.	326
4.7.3.10.26.3.	326
4.7.3.10.27.	327
4.7.3.10.27.1.	328
4.7.3.10.27.2.	330
4.7.3.10.27.3.	331

Contents (cont'd)

	Page	
4.7.3.10.27.4.	RGMII.....	332
4.7.3.10.27.5.	RMII	333
4.7.3.10.27.6.	SMII.....	333
4.7.3.10.27.7.	XGMII.....	334
4.7.3.10.27.8.	XLGMII.....	335
4.7.3.10.28.	OIF-CEI-04.0 Interface Function.....	336
4.7.3.10.29.	PCIe Interface Function.....	337
4.7.3.10.29.1.	PCIe-x1	338
4.7.3.10.29.2.	PCIe-x2	339
4.7.3.10.29.3.	PCIe-x4	340
4.7.3.10.29.4.	PCIe-x8	341
4.7.3.10.29.5.	PCIe-x16	342
4.7.3.10.29.6.	PCIe-x32	343
4.7.3.10.29.7.	SFF-8639 Connector	345
4.7.3.10.29.8.	ATX - Power Connector 150W	346
4.7.3.10.29.9.	AuxiliaryPowerConnector2x4.....	346
4.7.3.10.30.	Cabling PCIe Interface Function.....	347
4.7.3.10.30.1.	CablingPCle-x1	347
4.7.3.10.30.2.	CablingPCle-x4	348
4.7.3.10.30.3.	CablingPCle-x8	348
4.7.3.10.30.4.	CablingPCle-x16.....	349
4.7.3.10.31.	A-PHY	350
4.7.3.10.32.	BoW-PHY.....	350
4.7.3.10.33.	C-PHY Interface Function.....	351
4.7.3.10.33.1.	C-PHY-x1	351
4.7.3.10.33.2.	C-PHY-x2	352
4.7.3.10.33.3.	C-PHY-x3	352
4.7.3.10.33.4.	C-PHY-x4	353
4.7.3.10.33.5.	C-PHY-x5	353
4.7.3.10.33.6.	C-PHY-x6	354
4.7.3.10.34.	D-PHY Interface Function.....	355
4.7.3.10.34.1.	D-PHY-x1	356
4.7.3.10.34.2.	D-PHY-x2	356
4.7.3.10.34.3.	D-PHY-x3	357
4.7.3.10.34.4.	D-PHY-x4	357
4.7.3.10.34.5.	D-PHY-x5	358
4.7.3.10.34.6.	D-PHY-x6	358
4.7.3.10.34.7.	D-PHY-x7	359
4.7.3.10.34.8.	D-PHY-x8	359
4.7.3.10.35.	M-PHY	360
4.7.3.10.36.	OpenHBI	360
4.7.3.10.37.	PTI Interface Function	361
4.7.3.10.38.	Radio Front End Interface Function.....	361
4.7.3.10.38.1.	RBDP	362
4.7.3.10.38.2.	RF-BB	362
4.7.3.10.39.	RFFE.....	363
4.7.3.10.40.	SD	363
4.7.3.10.41.	SD-UHS-II.....	364
4.7.3.10.42.	Serial Interface Function.....	364
4.7.3.10.42.1.	SI.....	365
4.7.3.10.42.2.	SPI	365
4.7.3.10.42.3.	eSPI	366

Contents (cont'd)

	Page	
4.7.3.10.42.4.	xSPI	366
4.7.3.10.42.5.	SPD5118 Hub.....	367
4.7.3.10.43.	SLIMbus.....	367
4.7.3.10.44.	SMB	368
4.7.3.10.45.	SoundWire Interface Function	368
4.7.3.10.46.	SPMI	369
4.7.3.10.47.	UART	369
4.7.3.10.48.	UCle.....	370
4.7.3.10.48.1.	UCle - Advanced	371
4.7.3.10.48.2.	UCleStandard_x16	372
4.7.3.10.48.3.	UCleStandard_x32	373
4.7.3.10.49.	UniPro	374
4.7.3.10.50.	Universal Flash Storage Interface Function	374
4.7.3.10.50.1.	UFS – Single Channel	375
4.7.3.10.50.2.	UFS – Dual Channel.....	376
4.7.3.10.50.3.	UFSHCl.....	377
4.7.3.10.51.	USB Interface Function.....	377
4.7.3.10.51.1.	USB2.0.....	378
4.7.3.10.51.2.	USB3.0.....	379
4.7.3.10.51.3.	USB_Type-C_Receptacle.....	380
4.7.3.10.51.4.	USB_Type-C_Plug	380
4.7.3.10.51.5.	XFM	381
4.7.3.10.52.	Other Interface Standard	381
4.7.3.11.	Non Linear – Frequency Mixer	382
4.7.3.11.1.	Balanced	383
4.7.3.11.2.	Other Non Linear Function Standard.....	383
4.7.3.12.	Optoelectronics	384
4.7.3.12.1.	Photoemitter.....	384
4.7.3.12.1.1.	Infrared Emitting Diode	385
4.7.3.12.1.2.	LED	385
4.7.3.12.1.3.	Laser	386
4.7.3.12.2.	Photosensitive Device	387
4.7.3.12.2.1.	Photodiode	387
4.7.3.12.2.2.	Photothyristor.....	388
4.7.3.12.2.3.	Phototriac.....	388
4.7.3.12.2.4.	Phototransistor	389
4.7.3.12.2.5.	Photodarlington.....	389
4.7.3.12.2.6.	Photovoltaic Diode	390
4.7.3.12.3.	Optocoupler	390
4.7.3.12.3.1.	Photodiode	391
4.7.3.12.3.2.	Photothyristor	391
4.7.3.12.3.3.	Phototriac.....	392
4.7.3.12.3.4.	Phototransistor	393
4.7.3.12.3.5.	Photodarlington.....	394
4.7.3.13.	Relay	395
4.7.3.13.1.	Solid State Relay – Single Throw	396
4.7.3.13.2.	Solid State Relay – Double Throw.....	397
4.7.3.13.3.	Electromagnetic Relay – Single Throw	398
4.7.3.13.3.1.	Electromagnetic Relay – Single Throw – Contact Array	398
4.7.3.13.4.	Electromagnetic Relay – Double Throw	399
4.7.3.13.5.	Electromagnetic Relay – Double Throw – Contact Array	399
4.7.3.14.	Resistor	400
4.7.3.14.1.	Two Terminal Resistor Standard Terminal Name Assignment Type	401

Contents (cont'd)

	Page
4.7.3.14.2.	402
4.7.3.15.	404
4.7.3.15.1.	405
4.7.3.15.2.	405
4.7.3.15.3.	406
4.7.3.15.4.	406
4.7.3.15.5.	407
4.7.3.15.5.1.	407
4.7.3.15.5.2.	408
4.7.3.15.6.	408
4.7.3.15.7.	409
4.7.3.15.8.	410
4.7.3.15.9.	410
4.7.3.15.10.	411
4.7.3.15.11.	411
4.7.3.15.12.	412
4.7.3.15.12.1.	412
4.7.3.15.12.1.1.	413
4.7.3.15.12.1.2.	414
4.7.3.15.12.2.	414
4.7.3.15.13.	415
4.7.3.15.14.	415
4.7.3.15.15.	415
4.7.3.15.16.	416
4.7.3.16.	416
4.7.3.16.1.	417
4.7.3.16.2.	417
4.7.3.16.3.	418
4.7.3.17.	418
4.7.3.17.1.	419
4.7.3.17.2.	419
4.7.3.17.3.	420
4.7.3.18.	420
4.7.3.18.1.	421
4.7.3.18.2.	421
4.7.3.18.3.	422
4.7.3.18.3.1.	422
4.7.3.18.3.2.	423
4.7.3.18.3.3.	423
4.7.3.18.3.4.	424
4.7.3.19.	424
4.7.3.19.1.	425
4.7.3.20.	425
4.7.3.20.1.	426
4.7.3.20.1.1.	426
4.7.3.20.1.2.	427
4.7.3.20.2.	428
4.7.3.20.2.1.	428
4.7.3.20.2.2.	429
4.7.3.20.3.	430
4.7.3.20.3.1.	431
4.7.3.20.4.	432

Contents (cont'd)

	Page
4.7.3.20.4.1.	IGBT – Standard Terminal Name Assignment Type 433
4.7.3.20.5.	Programmable Unijunction Transistor 433
4.7.3.20.5.1.	Terminal Mapping 434
4.7.3.21.	Other Standard Electrical Functions 434
4.7.4.	Electrical Specification 435
4.7.4.1.	Test Condition 436
4.7.4.1.1.	Ascii Units 438
4.7.4.1.1.1.	Complex UOM 443
4.7.4.1.2.	Units 444
4.7.4.1.2.1.	Complex UOM 458
4.7.4.2.	Parameter Set 460
4.7.4.2.1.	Parameter 461
4.7.4.2.1.1.	Values 462
4.7.4.2.1.2.	Rule 463
4.7.4.2.1.2.1.	Rule Context 463
4.7.4.3.	Parameter Graph 464
4.7.4.3.1.	Formatting 466
4.7.4.3.1.1.	Linking the Data-Array to the Appropriate Parameter Definition 469
4.7.4.3.2.	Data-Array 472
4.7.4.3.3.	Graph Formula 481
4.7.5.	Truth Table 482
4.7.6.	ESD 484
4.7.6.1.	HBM – Human Body Model 485
4.7.6.2.	CDM – Charged Device Model 485
4.8.	SCHEMATIC DATA - ARRAY 486
4.8.1.	Symbol - Array 486
4.8.1.1.	Symbol Graphics – Array 487
4.8.1.1.1.	Part Symbol Version 488
4.8.1.1.2.	Part Reference Designation 489
4.8.1.1.3.	Fracture – Array 490
4.8.1.1.3.1.	Symbol Name 490
4.8.1.1.3.1.1.	Function Symbol Version 491
4.8.1.1.3.2.	Function Reference Designation 491
4.8.1.1.3.3.	Graphical Representation 492
4.8.1.1.3.3.1.	Body 493
4.8.1.1.3.3.1.1.	Shape 494
4.8.1.1.3.3.1.1.1.	Vertex – Array 495
4.8.1.1.3.3.1.1.2.	Vertex 495
4.8.1.1.3.3.1.1.3.	Edge 496
4.8.1.1.3.3.1.1.4.	Line 497
4.8.1.1.3.3.1.1.5.	Arc 498
4.8.1.1.3.3.1.1.6.	Elliptical Arc 499
4.8.1.1.3.3.1.1.7.	Primative-Shape 500
4.8.1.1.3.3.1.1.8.	Shape Origin 501
4.8.1.1.3.3.1.1.9.	Transformation 502
4.8.1.1.3.3.1.2.	SVG-Shape 503
4.8.1.1.3.3.1.3.	Image 504
4.8.1.1.3.3.1.4.	Shape Text 505
4.8.1.1.3.3.2.	Graphical Format – Array 506
4.8.1.1.3.3.2.1.	Graphical Format 507
4.8.1.1.3.3.2.2.	Default Graphical Format 508
4.8.1.1.3.3.3.	Attribute – Array 509
4.8.1.1.3.3.3.1.	Text Variant for One of the Views – Array 510

Contents (cont'd)

	Page
4.8.1.1.3.3.4.4. Text Format – Array	511
4.8.1.1.3.3.4.1. Text Format.....	511
4.8.1.1.3.3.4.2. Default Text Format	512
4.8.1.1.3.3.5. Terminal – Array	513
4.8.1.1.3.3.5.1. Location	515
4.8.1.1.3.3.5.2. Label	516
4.8.1.1.3.3.5.3. Function Graphics.....	517
4.8.2. Required Circuitry - Array	517
4.8.2.1. Net - Array.....	518
4.9. MAPPING - ARRAY.....	520
4.9.1. Electrical Map – Array.....	521
4.9.1.1. Operational Mode	524
4.9.1.1.1. Functional Operational Mode.....	525
4.9.1.1.2. Power Operational Mode	526
4.9.1.1.3. Load Operational Mode	526
4.9.1.1.4. Test Operational Mode	527
4.9.2. Terminal Map	527
4.9.2.1. Terminal Name	528
4.9.2.1.1.1. Terminal Selection	529
4.9.2.1.2. Naming Field.....	530
4.9.2.1.2.1. Duplicate Suffix - Array	531
4.9.2.1.2.1.1. Sub Pattern Group Reference	531
4.9.2.1.2.1.2. Start Position.....	531
4.9.2.1.2.1.3. Sequence.....	533
4.9.2.1.2.2. Terminal Selection - Array	538
4.9.2.2. Standard Terminal Name.....	538
4.9.2.3. Companion Standard Terminal Map.....	539
4.9.3. Package Terminal Map.....	539
4.9.4. Die Terminal Map	542
4.9.5. Simulation Map	542
4.10. SIMULATION MODEL - ARRAY.....	543
4.11. REFERENCE DESIGN - ARRAY.....	544
4.12. SOFTWARE INTERFACE DESCRIPTION - ARRAY	545
4.12.1. IP-XACT	545
4.12.2. System RDL.....	546
5 Quick TeX reference.....	547
5.1. SYNTAX	547
5.2. SYMBOLS.....	547
6 Rule Syntax.....	548
6.1. RULE APPLIED TO A LOGICAL-GROUP	549
6.2. RULE APPLIED TO A TERMINAL MAP.....	550
6.3. RULE FUNCTIONS.....	551
6.3.1. Arithmetic Operations	551
6.3.2. Modifying Precedence	551
6.3.3. Function to Reference Other Terminals	551
6.3.3.1. Functions	552
6.3.4. Constants.....	552
Annex A (informative) Graphic Symbol Definition.....	553
A.1. General Recommended Graphic Sizing.....	553
A.1.1. Input and Output Terminals Graphics.....	554
A.1.2. Bidirectional Signal – Only One Direction at a time.....	555
A.1.3. Bidirectional Signal – Both directions Simultaneously	555

Contents (cont'd)

	Page	
A.1.4.	Text Height.....	555
A.1.5.	Line to Line with Text	555
A.1.6.	Line to Line without Text.....	555
A.1.7.	Graphic Sizing for Bus without Text.....	555
A.1.8.	Graphic Sizing for Bus with Text	556
A.2.	Recommended Graphic Symbol Representation of Signals, Properties, Functions ..	556
A.2.1.	Negation.....	556
A.2.2.	Active Low & Active High.....	556
A.2.3.	Clock.....	557
A.2.4.	Differential Clock.....	557
A.2.5.	Enable	557
A.2.6.	Amplifier	558
A.2.7.	Tri-State Output.....	558
A.2.8.	Passive Pulldown.....	559
A.2.9.	Passive Pullup.....	559
A.2.10.	Open Circuit Output.....	559
A.2.11.	Open Circuit Output (H-Type)	559
A.2.12.	Open Circuit Output (L-Type).....	560
A.2.13.	Schmitt Trigger	560
A.2.14.	Analog Terminal	561
A.2.15.	Digital Terminal	561
A.2.18.	Subsidiary Connection	561
A.2.19.	Postponed Output.....	562
A.3.	Recommended Rules.....	562
A.3.1.	Symbol Ordering Rule	563
A.3.2.	Terminal Grouping	563
A.3.2.1	Bit Grouping	564
A.3.2.2	Function Grouping	564
A.3.2.3	Common Name Grouping.....	565
A.3.3.	Group Direction Symbols.....	565
Annex B (informative) Differences between JEP30-E100 and its predecessors.....		567

Tables

Table 2 — Switch Function Table	177
Table 2 — UOM Enumerated Lists	441
Table 3 — UOM Enumerated Lists	451
Table 4 — NOR logic States	483
Table 5 – Sub Pattern Start Position Relative-to-Parent Pattern Group.....	532
Table 6 – Naming Nomenclature Ordering Code & Description	534
Table 7 - Naming Field Patterns	535
Table 8 — LaTeX Syntax Sample.....	547
Table 9 — LaTeX Symbols and Descriptions Sample	547
Table 10 — Arithmetic Operations	551
Table 11 — Rule Functions.....	552
Table 12 — Mathematical Functions	552

Figures

Figure 1 - Ceramic Capacitor Array	61
Figure 2— Fused Electrolytic Capacitor	63
Figure 3 — Coupled Inductor	87
Figure 4— NPN Darlington	123
Figure 5 — BJT	123
Figure 6 — p-type UJT	124
Figure 7 — Gas Discharge Tube	130
Figure 8 — Vacuum Fluorescent Display	130
Figure 9 — Sample Device with no internal die connection	136
Figure 10 — Clamp Circuit.....	136
Figure 11 — Hysteresis Loop Curve	137
Figure 12 — Schmitt Trigger	137
Figure 13 — Signal Inversion.....	138
Figure 14 — Tri-State.....	138
Figure 15 — Analog Connection Types	140
Figure 16 — Internal Pullup / Pulldown Circuits	142
Figure 17 — Series Component Types.....	143
Figure 18 — Passive Pull-up Output Circuit	145
Figure 19 — Passive Pull-down Output Circuit.....	146
Figure 20 — Totem Pole	147
Figure 21 — Rail-to-Rail Push-Pull	148
Figure 22 — High Side Driver (Source Driver).....	149
Figure 23 — Low Side Driver (Sink Driver).....	150
Figure 24 — Open-circuit Low Side Driver Type	152
Figure 25 — Open-circuit High Side Driver Type.....	152
Figure 26 — Signal Reference.....	154
Figure 27 — Same potential	161
Figure 28 — Sample Mixed Gate Device	162
Figure 29 — Sample NAND Gate Device.....	164
Figure 30 — Internal Electrical Connection	165
Figure 31 — System with Differential Receiver	166
Figure 32 — Differential Pair Device.....	167
Figure 33 — Function Block Diagram of a 12-bit Bus Switching Device	170
Figure 34 — Clock Buffer	173
Figure 35 — Differential Multiplexer.....	175
Figure 36 — Circuit with DC-Blocking.....	179
Figure 37 — Decoupling Circuit	180
Figure 38 — Pull-up Resistor to Power.....	181
Figure 39 — Pull-down Current Source to Ground	182
Figure 40 — Sample Transistor Circuit	184
Figure 41 — Pull-up Pull-down Resistor Array	402
Figure 42 — Aging Rate	470
Figure 43 — Capacitance Value versus Temperature.....	475
Figure 44 — Voltage Coefficient of Capacitance	477
Figure 45 — Alternative Symbols	487
Figure 46 — Text Views for Alternative Symbol Rotation	509
Figure 47 – Terminal Naming Field Pattern Samples.....	537
Figure 47 — Sample NAND Gate Device	540
Figure 48 — Sample Transistor Circuit.....	541

PartModel Electrical Guidelines for Electronic-Device Packages – XML Requirements

(From JEDEC Board Ballots JCB-17-48, JCB-23-10, JCB-23-27, JCB-23-33, JCB-24-08, JCB-24-29, JCB-24-50, JCB-24-53, and JCB-25-59 formulated under the cognizance of the JC-11 Committee on PartModel XML Schema Definition.)

1 Scope

The JEP30 document establishes the requirements for exchanging part data between part manufacturers and their customers for electrical and electronic products. The JEP30 documents are part of a series to describe XML data exchange structure and hierarchy. The JEP30 document series will detail data exchange between companies for design at the next level, analysis, and interconnection. The parent JEP30 document specifically focuses on the parental structure, under which several sub-sections are listed, such as electrical, physical, thermal, supply chain, assembly process classification, design kit, generated ECAD models, product substrate and assemblies, and environment including material declaration. This document specifically focuses on the Electrical sub-section of the PartModel.

All releases of the *ElectricalSection* sub-schema must be under the umbrella of the PartModel Schema to ensure that the PartModel schema is referencing the correct version of the Electrical sub-schema. In addition, this will enable the *ElectricalSection* sub-schema. In addition, this will enable the *ElectricalSection* sub-schema to connect to the Manufacturer Part Number and the Manufacturer of the Part.

1.1. Purpose

This standard is intended to benefit part manufacturers and their customers by providing consistency and efficiency to the transfer of part design rule data from part manufacturer to customers. It establishes standard electronic data exchange formats that will facilitate and improve data transfer along the entire global supply chain, at every stage in the product life cycle. A key aspect therefore is the structure of the content that is contained in this format, which the committee believes should be based on the following two principles:

- 1) Data that is required to be consumed by software tools, and
- 2) Data that is not required to be consumed by software tools but is provided for informational purposes.

This standard specifically covers data applicable to the electrical definition of the device.

2 Applicable Documents

The following documents form a part of this standard to the extent specified herein. The revision of the document in effect at the time of solicitation shall take precedence.

2.1. JEDEC (www.jedec.org)

JEDEC Publication, JEP30, PartModel Guideline for Electronic Device packages – XML Requirements.

JEDEC Publication, JEP30-10, PartModel Schema.

JEDEC Publication, JEP30-E101, PartModel Electrical Schema.

JEDEC Publication, JEP30-D10, PartModel Schema Types Dictionary (Required to support the PartModel Schema and each of its sectional sub-schemas.).

JEDEC Publication, JEP95, JEDEC Registered and Standard Outlines for Solid State Products.

JEDEC Publication, JEP104, Reference Guide to Letter Symbols for Semiconductor Devices.

JEDEC/ESD Publication, JEP157 - Recommended ESD-CDM Target Levels

JEDEC Standard, JESD30J, Descriptive Designation System for Electronic-device Packages.

JEDEC Standard, JESD77, Terms, Definitions, and Letter Symbols for Discrete Semiconductor and Optoelectronic Devices.

JEDEC Standard, JESD79-3F, (DDR3 SDRAM Standard)

JEDEC Standard, JESD79-4B, (DDR4 SDRAM)

JEDEC Standard, JESD79-5A, (DDR5 SDRAM)

JEDEC Standard, JESD82-32A, DDR4 Data Buffer Definition (DDR4DB02)

JEDEC Standard, JESD84-B42, MultiMediaCard (MMC) Electrical Standard, High Capacity (MMCA, 4.2)

JEDEC Standard, JESD84-B51A, Embedded Multi-Media Card (eMMC) Electrical Standard (5.1A)

JEDEC Standard, JESD84-A441, Embedded MultiMediaCard(e•MMC) e•MMC/Card Product Standard, High Capacity, including Reliable Write, Boot, Sleep Modes, Dual Data Rate, Multiple Partitions Supports, Security Enhancement, Background Operation and High Priority Interrupt (MMCA, 4.41)

JEDEC Standard, JESD88E, JEDEC Dictionary of Terms for Solid-State Technology.

2.1 JEDEC (www.jedec.org) (cont'd)

JEDEC Standard, JESD99, *Terms, Definitions, and Letter Symbols for Microelectronic Devices.*

JEDEC Standard, JESD100, *Terms, Definitions, and Letter Symbols for Microcomputers, Microprocessors, and Memory Integrated Circuits.*

JEDEC Standard, JESD209-4D, *Low Power Double Data Rate 4 (LPDDR4)*

JEDEC Standard, JESD209-5B, *Low Power Double Data Rate 5 (LPDDR5)*

JEDEC Standard, JESD212C, *Graphics Double Data Rate (GDDR5) SGRAM Standard*

JEDEC Standard, JESD220E, *Universal Flash Storage (UFS) Version 3.1*

JEDEC Standard, JESD223C, *Universal Flash Storage Host Controller Interface (UFSHCI) Version 2.1*

JEDEC Standard, JESD232A, *Graphics Double Data Rate (GDDR5X) SGRAM Standard*

JEDEC Standard, JESD235D, *High Bandwidth Memory DRAM (HBM1, HBM2)*

JEDEC Standard, JESD238, *High Bandwidth Memory DRAM (HBM3)*

JEDEC Standard, JESD250C, *Graphics Double Data Rate (GDDR6) SGRAM Standard*

JEDEC Standard, JESD270-4, *High Bandwidth Memory (HBM4) DRAM Standard*

JEDEC/ESDA Standard, JS-001-2014, *For Electrostatic Discharge Sensitivity Testing – Human Body Model (HBM) – Component Level.*

JEDEC/ESDA Standard, JS-002-2014, *For Electrostatic Discharge Sensitivity Testing – Charged Device Model (CDM) – Device Level.*

2.2 CHIPS ALLIANCE (<https://chipsalliance.org/>)

AIB Specification: <https://github.com/chipsalliance/AIB-specification>

2.3 IEEE (www.ieee.org)

IEEE std 802.3, *IEEE Standard for Ethernet.*

IEEE 802.3ba-2010, *IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 3: CSMA/CD Access Method and Physical Layer Specifications Amendment 4: Media Access Control Parameters, Physical Layers, and Management Parameters for 40 Gb/s and 100 Gb/s Operation.*

IEEE 802.3ae-2002, *IEEE Standard for Information technology - Local and metropolitan area networks - Part 3: CSMA/CD Access Method and Physical Layer Specifications - Media Access*

Control (MAC) Parameters, Physical Layer, and Management Parameters for 10 Gb/s Operation.

2.3 IEEE (www.ieee.org) (cont'd)

IEEE 802.3u-1995, IEEE Standards for Local and Metropolitan Area Networks: Supplement - Media Access Control (MAC) Parameters, Physical Layer, Medium Attachment Units, and Repeater for 100Mb/s Operation, Type 100BASE-T (Clauses 21-30).

IEEE std 802.3z-1998, Gigabit Task Force (<https://www.ieee802.org/3/z/>)

IEEE Standard, 2977-2021, IEEE Standard for Adoption of MIPI Alliance Specification for A-PHY Interface (A-PHY) Version 1.0.

2.4. IEEE/ANSI/CSA (www.ansi.org)

ANSI Y32.2-1975 (Reaffirmed 1989), Graphic Symbols for Electrical and Electronics Diagrams.

2.5. IEC (www.iec.org)

IEC 60617, Graphical symbols for diagrams.

2.6. IPC (www.ipc.org)

IPC-T-50, Terms and Definitions for Interconnecting and Packaging Electronic Circuits.

2.7. INCITIS (www.incitis.org)

2221-D, Fibre Channel Physical Interface-6.

2.8. MIPI (www.mipi.org)

Specification for C-PHY

Specification for D-PHY Version 1.00.00

Specification for M-PHY Version 4.1.

Specification for RF Front-End Control Interface Version 1.10

Specification for Unified Protocol (UniPro) Version 1.8

System Power Management Interface V2.0

2.9. OIF (www.oiforum.com)

IA # OIF-CEI-04.0, Common Electrical I/O (CEI) - Electrical and Jitter Interoperability agreements for 6G+ bps, 11G+ bps, 25G+ bps I/O and 56G+ bps.

2.10. OPEN Compute Project (<https://www.opencompute.org/>)

BoW-PHY: https://opencomputeproject.github.io/ODSA-BoW/bow_specification.html

OpenHBI: <https://www.opencompute.org/documents/odsa-openhbi-v1-0-spec-rc-final-1-pdf>

2.11. HDMI Licensing, LLC (www.hDMI.com)

High-Definition Multimedia Interface Specification Version 1.3a

2.12. PCI-SIG (www.pcisig.com)

PCI Express Card Electromechanical Specification Rev 2.0

PCI Express® External Cabling Specification Revision 2.0

2.13. HP (www.hp.com)

RGMII - Reduced Gigabit Media Independent Interface (RGMII) Version 2.0 -
http://www.hp.com/rnd/pdfs/RGMIIv2_0_final_hp.pdf

2.14. SDA (www.sdcards.org)

2.15. SMI (www.powersig.org/)

System Management Bus (SMBus) Specification Version 3.0

2.16. USB-IF (www.usb.org)

Universal Serial Bus Specification Revision 2.0

Universal Serial Bus 3.2 Specification Revision 1.0

2.17. UCIe (<https://www.uciexpress.org/>)

2.18. Accellera (<https://www.accellera.org/>)

2.19. American Mathematical Society

"Short Math Guide for L^AT_EX", Version 1.09 (2002-03-22), currently available at
<http://www.ams.org/tex/short-math-guide.html>.

2.20. MathML - https://www.w3.org/TR/mathml4/#intro_overview

3 Requirements

The following terms and definitions are applicable to this XML Schema.

3.1. Terms and Definitions

All definitions and terms associated with the Electrical Data are defined in the JESD51 series of documents, as listed in the applicable documents section. The Electrical details of the part are defined in the *ElectricalSection* of the PartModel XML Schema.

All common Terms and Definitions that are used by more than one sectional sub-schema, such as any of the Electrical, Package, Environmental, Assembly Process Classification, are defined in the "PartModel Common Types Library"

All other definitions and terms necessary to define the schema, are defined in this document.

PartModel: A PartModel is a data representation described in an XML file that conforms to the rules and structure of the PartModel XML Schema.

NOTE 1 Companies who use the PartModel XML Files and claim compliance to JEDEC, must ensure that their PartModel XML file conforms to the specific released version of the PartModel XML Schema released by JEDEC.

NOTE 2 Section 4 will define the outline of the structure of the Electrical XML Schema. Specific components of the XML Schema and their hierarchy are specifically controlled by the JC-16 and JC-42 Standards Committee who retain the expertise for these structures.

NOTE 3 The *ElectricalSection* of the schema forms part of the PartModel XML Schema and is not intended to act as a standalone schema. In addition, there is a "PartModel Schema Types Library" XML Schema, which is a common set of xml structures shared across the PartModel XML Schema and all its sub-section schemas.

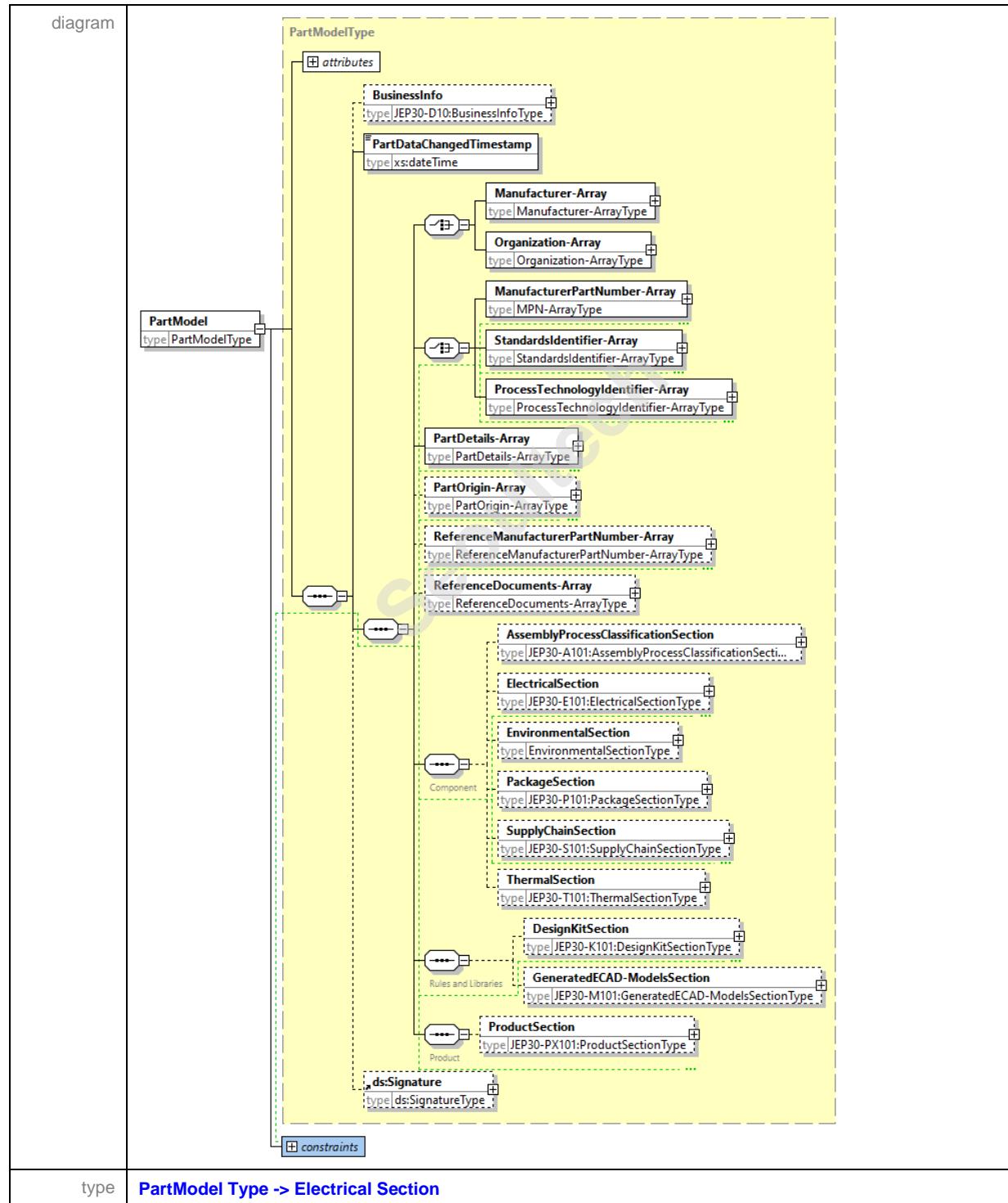
3.2. XML Schema Key Terms and Definitions

Reference the JEP30 publication for details of the "XML Schema Key Terms and Definitions".

4 PartModel Schema Definition

The following section describes the XML Schema structure.

4.1. PartModel -> Electrical Section



4.1 PartModel -> Electrical Section (cont'd)

The *PartModelType* belongs to the “PartModel XML Schema”. The *ElectricalSection* belongs to the “PartModel Electrical XML Schema”. The primary purpose of the PartModel Schema is to provide the structure for identifying unique parts (Manufacturer and MPN) or unique standards (Organization and Standards Identifier) and the structure to include the sub schemas which define the part details, as outline in the JEP30 - PartModel Guidelines for Electronic-Device Packages – XML Requirements.

This document covers the *ElectricalSection*, which is referenced from its parent's structure, the *PartModel*. The contents under the *ElectricalSection* are

1. Tied to the Manufacturer's name and Manufacturer's part number if the electrical section is applicable to a part or a chiplet, or
2. Tied to a Standards Body and a Standards Model Identifier, if the electrical section represents a standard set of Interfaces or Functions or Electrical standards as defined by a specific Standards Body.

The *ComplianceToPartModelSchemaVersion* indicates the version of the Schema to which the XML file is to be validated against. All new releases to this document or XML Schema are governed by the rules outlined in the JEP30 and must be release in sync with the PartModel.

“Each time that a Sub-schema gets updated, then the PartModel version also gets updated in order to release that Sub-schema under the umbrella of the PartModel. This is because the PartModel must now reference the new version of Sub-schema, since all subschemas have their own version number. The parent schema includes them by referring to a precise version, so a version bump in the subschema requires a version bump in the parent only at the time of release of the Parent.”

The *PartModelContentRevision* indicates the revision of the data for the Part that is submitted in the XML file. This enables the Component Manufacturer to provide a new XML file for a Part each time they wish to upgrade a new set of data for a part, in any of the sub-sections such as this *ElectricalSection*.

4.2. Manufacturer Part Number-Array

path	PartModel/ManufacturerPartNumber-Array.
diagram	<pre> classDiagram class MPNArrayType { <<MPN-ArrayType>> } class ManufacturerPartNumberArray { <<MPN-ArrayType>> } class ManufacturerPartNumbers { <<ManufacturerPartNumbersType>> } class ManufacturerPartNumbersType { <<attributes>> ID type xs:string } class PartNumberSeries { <<JEP30-D10:PartNumberSeriesType>> } class OrderablePartNumber { <<JEP30-D10:OrderablePartNumberType>> } class FuturePart { <<FuturePartType>> } class ManufacturerID { <<xs:string>> } class ManufacturerSignatureDigest { <<JEP30-D10:SignatureDigestLinkType>> } class dsSignature { <<ds:SignatureType>> } ManufacturerPartNumberArray "1..>"--> ManufacturerPartNumbers : ManufacturerPartNumbers "1..>"--> ManufacturerPartNumbersType : ManufacturerPartNumbers "1..>"--> PartNumberSeries : ManufacturerPartNumbers "1..>"--> OrderablePartNumber : ManufacturerPartNumbers "1..>"--> FuturePart : ManufacturerPartNumbers "1..>"--> ManufacturerID : ManufacturerPartNumbers "1..>"--> ManufacturerSignatureDigest : ManufacturerPartNumbers "1..>"--> dsSignature : </pre>
type	MPN-ArrayType, ManufacturerPartNumbersType, PartNumberType, OrderablePartNumber-ArrayType, FuturePartType, JEP30-D10:SignatureDigestLinkType, ds:SignatureType.

The *ManufacturerPartNumber-Array/ManufacturerPartNumber* provides the definition of the part number or a specific Standard, so that it can be connected to the technical specification details in the *ElectricalSection* via the *PartDetails-Array* section.

4.3. Standards Identifier - Array

path	PartModel/StandardsIdentifier-Array.
diagram	<pre> classDiagram class StandardsIdentifier-Array { <<StandardsIdentifier-ArrayType>> } class StandardsIdentifier { <<StandardsIdentifierType>> ID : xs:string Name : xs:string StandardsNumber : xs:string Version : xs:string BaselineIdentifier : xs:string ModelVariationIdentifier : xs:string Description : xs:string StandardsOrganizationIdentityID : xs:string StandardsOrganizationIdentitySignatureDigest : JEP30-D10:SignatureDigestLinkType ds:Signature : ds:SignatureType } StandardsIdentifier-Array "0..∞" -- "0..∞" StandardsIdentifier </pre>
type	StandardsIdentifier-ArrayType, StandardsIdentifierType, JEP30-D10:SignatureDigestLinkType, ds:SignatureType.

The [StandardsIdentifier-Array/StandardsIdentifier](#) provides the definition of a specific Standard, so that it can be connected to the technical specification details in the [ElectricalSection](#) via the [PartDetails-Array](#) section.

4.4. Process Technology Identifier - Array

path	PartModel/ProcessTechnologyIdentifier-Array.
diagram 1 of 2	<pre> classDiagram class ProcessTechnologyIdentifierArrayType { <<ProcessTechnologyIdentifier-ArrayType>> } class ProcessTechnologyIdentifier { <<ProcessTechnologyIdentifierType>> } ProcessTechnologyIdentifier-ArrayType "3..1" --> "1..1" ProcessTechnologyIdentifier : type </pre>
diagram 2 of 2	<pre> classDiagram class ProcessTechnologyIdentifierType { <<ProcessTechnologyIdentifierType>> ID xs:string Name xs:string Version xs:string DevCodeName xs:string 0..∞ BaselineIdentifier xs:string 0..∞ Variation xs:string 0..∞ Framework xs:string Stage-in-Development xs:string JEDEC-Stage-in-Development JEDEC-Stage-in-DevelopmentType AvailabilityDate xs:date EstimatedAvailabilityDate xs:string Description xs:string ManufacturerID xs:string ManufacturerSignatureDigest JEP30-D10:SignatureDigestLinkType ProcessTechnologyOrganizationID xs:string ProcessTechnologyOrganizationSignatureDigest JEP30-D10:SignatureDigestLinkType ProcessTechnologyIdentifierSignature ds:SignatureType } class ProcessTechnologyIdentifier { <<ProcessTechnologyIdentifierType>> } ProcessTechnologyIdentifier-ArrayType "3..1" --> "1..1" ProcessTechnologyIdentifier : type </pre>
type	ProcessTechnologyIdentifier-ArrayType , ProcessTechnologyIdentifier , JEDEC-Stage-in-DevelopmentType , JEP30-D10:SignatureDigestLinkType , ds:SignatureType .

The [ProcessTechnologyIdentifier-Array/ProcessTechnologyIdentifier](#) provides the definition of the process technology identifier, so that it can be connected to the technical specification details in the [ElectricalSection](#) via the [PartDetails-Array](#) section.

4.5. Linking the Manufacturing Part Number to a specific Electrical Data set

The linking of the Parts to its technical data is done via the *PartDetails-Array* section as outline in the JEP30 - PartModel Guidelines for Electronic-Device Packages – XML Requirements. This consists of two sections called *PartsSelection-Array* and *Association-Array* which defines the relationship between identifying the specific set of parts and how they are associated with the electrical content. Reference the JEP30 parent document for more details on this association.

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array.
diagram at the Association level	
type	ElectricalAssociation-ArrayType, ElectricalParametersAssociationType, SchematicDataAssociationType, MappingAssociationType, ReferenceDesignAssociationType, SoftwareInterfaceDescriptionAssociationType
diagram at the Electrical Section level	
type	ElectricalSectionType, ElectricalParameters-ArrayType, SchematicData-ArrayType, Mapping-ArrayType, SimulationModel-ArrayType, ReferenceDesign-ArrayType, SoftwareInterfaceDescription-ArrayType.

The electrical content is now sub-grouped into six major sections as shown in the diagram. This enables each section to be digitally signed independently of each other. The linkage between the two sections is shown below.

4.5.1. Linking the Manufacturing Part Number to Electrical Parameters

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<pre> classDiagram class ElectricalParametersAssociationType { ElectricalParameterID : xs:string PartClassificationID : xs:string TerminalDetailsID : xs:string SuperInterfaceID : xs:string < --> 0..infinity FunctionID : xs:string < --> 0..infinity ElectricalSpecificationID : xs:string < --> 0..infinity TruthTableID : xs:string < --> 0..infinity ESD-ID : xs:string < --> 0..infinity ElectricalParametersSignature : JEP30-D10:SignatureDigestLinkType } class ElectricalParameters { < -- ElectricalParametersAssociationType } ElectricalParameters "1..>"--> ElectricalParametersAssociationType </pre>
type	ElectricalParametersAssociationType , JEP30-D10:SignatureDigestLinkType .
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters
diagram at the Electrical Parameters-Array level	<pre> classDiagram class ElectricalParametersArrayType { ID : xs:string class ElectricalParametersType { PartClassification-Array : PartClassification-ArrayType TerminalDetails-Array : TerminalDetails-ArrayType FunctionGroup-Array : FunctionGroup-ArrayType ElectricalSpecification-Array : ElectricalSpecification-ArrayType ESD-Array : ESD-ArrayType ElectricalParametersSignature : SignatureType } } class ElectricalParametersArray { < -- ElectricalParametersType } ElectricalParametersArray "1..>"--> ElectricalParametersType </pre>
type	ElectricalParametersType , ElectricalParametersType , Electrical-ArrayType , ElectricalSpecification-ArrayType .

The [ElectricalParameterID](#) references the [ElectricalParameters/ID](#) under the [ElectricalSection/ElectricalParameters-Array](#). This is enforced by the key named as [ElectricalParametersKey](#) that is assigned to the [ElectricalParameters/ID](#) element, which is referenced by the [ElectricalParameterID](#) which has a KeyRef that refers to the [JEP30-E101:ElectricalParametersKey](#).

4.5.2. Linking the Manufacturing Part Number to Part Classification

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<p>The diagram shows the ElectricalParametersAssociationType class with the following attributes:</p> <ul style="list-style-type: none"> ElectricalParametersID: type xs:string PartClassificationID: type xs:string TerminalDetailsID: type xs:string SuperInterfaceID: type xs:string, multiplicity 0..∞ FunctionID: type xs:string, multiplicity 0..∞ ElectricalSpecificationID: type xs:string, multiplicity 0..∞ TruthTableID: type xs:string, multiplicity 0..∞ ESD-ID: type xs:string, multiplicity 0..∞ ElectricalParametersSignature: type JEP30-D10:SignatureDigestLinkType <p>A red arrow labeled A1 points from the PartClassificationID attribute to the PartClassification section of the second diagram.</p>
type	ElectricalParametersAssociationType , JEP30-D10:SignatureDigestLinkType .
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification
diagram at the Part Classification-Array level	<p>The diagram shows the PartClassification-ArrayType class with the following association:</p> <ul style="list-style-type: none"> PartClassification-Array: type PartClassification-ArrayType, multiplicity 1..∞ PartClassification: type PartClassificationType, multiplicity 1..∞ <p>The PartClassification class has a reference to the PartClassificationType class:</p> <ul style="list-style-type: none"> PartClassification → PartClassificationType PartClassificationType contains attributes: ID (type xs:string), CableAndWiring (type CableAndWiringClassificationType), Connector (type ConnectorClassificationType), Electrical (type ElectricalClassificationType), Hardware (type HardwareClassificationType), Optics (type OpticsClassificationType), and CompanionPart (type CompanionPartType, multiplicity 0..∞). <p>A red arrow labeled A1 points from the PartClassificationID attribute in the first diagram to the ID attribute in the PartClassificationType class.</p>
type	PartClassification-ArrayType , PartClassificationType , ...

The **PartClassificationID** references the **PartClassification/ID** under the [ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array](#). This is enforced by the key named as **PartClassificationKey** that is assigned to the **PartClassification/ID** element, which is referenced by the **PartClassificationID** which has a KeyRef that refers to the [JEP30-E101:PartClassificationKey](#).

4.5.3. Linking the Manufacturing Part Number to Terminal Details

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<pre> classDiagram class ElectricalParametersAssociationType { ElectricalParametersID : xs:string PartClassificationID : xs:string TerminalDetailsID : xs:string SuperInterfaceID : xs:string FunctionID : xs:string ElectricalSpecificationID : xs:string TruthTableID : xs:string ESD-ID : xs:string ElectricalParametersSignature : JEP30-D10:SignatureDigestLinkType } class ElectricalParameters { <<ElectricalParametersAssociationType>> } ElectricalParameters "1" --> "1" ElectricalParametersAssociationType </pre>
type	ElectricalParametersAssociationType , JEP30-D10:SignatureDigestLinkType .
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array
diagram at the Electrical-Array level	<pre> classDiagram class TerminalDetailsArrayType { TerminalDetails-Array : TerminalDetailsArrayType TerminalDetails : TerminalDetailsType <<constraints>> } class TerminalDetails { <<TerminalDetailsType>> } TerminalDetails-Array "1..∞" --> "1" TerminalDetails </pre>
type	TerminalDetails-ArrayType , TerminalDetailsType , ...

The unbounded element `TerminalDetailsID` references the `TerminalDetails/ID` under the `ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array`. This is enforced by the key named as `TerminalDetailsKey` that is assigned to the `TerminalDetails/ID` element, which is referenced by the `TerminalDetailsID` which has a KeyRef that refers to the `JEP30-E101:TerminalDetailsKey`.

4.5.4. Linking the Manufacturing Part Number to Super Interface

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<pre> classDiagram class ElectricalParametersAssociationType { ElectricalParametersID PartClassificationID TerminalDetailsID SuperInterfaceID "0..oo" FunctionID "0..oo" ElectricalSpecificationID "0..oo" TruthTableID "0..oo" ESD-ID "0..oo" ElectricalParametersSignature } class ElectricalParameters { type ElectricalParametersAssociationType } ElectricalParameters --> ElectricalParametersAssociationType </pre>
type	ElectricalParametersAssociationType, JEP30-D10:SignatureDigestLinkType.
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array
diagram at the Electrical-Array level	<pre> classDiagram class FunctionGroupArrayType { FunctionGroup-Array } class SuperInterfaceArrayType { SuperInterface-Array } class SuperInterfaceType { ID Name Sequence } class SuperInterface { type SuperInterfaceType } SuperInterface-Array --> SuperInterface SuperInterface --> SuperInterfaceType FunctionGroup-Array --> constraints SuperInterfaceType --> constraints </pre>
type	Electrical-ArrayType, ElectricalType, ...

The unbounded element *SuperInterfaceID* references the *SuperInterface/ID* under the *ElectricalSection/ElectricalParameters-Array/ElectricalParameters/Function-Array*. This is enforced by the key named as *SuperInterfaceKey* that is assigned to the *SuperInterface/ID* element, which is referenced by the *SuperInterfaceID* which has a KeyRef that refers to the *JEP30-E101:SuperInterface*.

4.5.5. Linking the Manufacturing Part Number to Functions

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<pre> classDiagram class ElectricalParametersAssociationType { ElectricalParametersID : xs:string PartClassificationID : xs:string TerminalDetailsID : xs:string SuperInterfaceID : xs:string FunctionID : xs:string ElectricalSpecificationID : xs:string TruthTableID : xs:string ESD-ID : xs:string ElectricalParametersSignature : JEP30-D10:SignatureDigestLinkType } class ElectricalParameters { *ElectricalParametersAssociationType } ElectricalParameters "1" --> "0..oo" ElectricalParametersAssociationType ElectricalParametersAssociationType "0..oo" --> "1" FunctionID </pre>
type	ElectricalParametersAssociationType, JEP30-D10:SignatureDigestLinkType.
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array
diagram at the Electrical-Array level	<pre> classDiagram class FunctionGroupArrayType { SuperInterface-Array : SuperInterface-ArrayType Function : FunctionType } class FunctionType { ID : xs:string Amplifier : AmplifierFunctionType Audio : AudioFunctionType Capacitor : CapacitorFunctionType Diode : DiodeFunctionType Filter : FilterFunctionType FrequencySource : FrequencySourceType } class FunctionGroup-Array { *FunctionGroup-ArrayType } FunctionGroup-Array "1..oo" --> "1..oo" Function Function "1..oo" --> "1..oo" FunctionType </pre>
type	Electrical-ArrayType, ElectricalType, ...

The unbounded element *FunctionID* references the *Function/ID* under the *ElectricalSection/ElectricalParameters-Array/ElectricalParameters/Function-Array*. This is enforced by the key named as *FunctionKey* that is assigned to the *Function/ID* element, which is referenced by the *FunctionID* which has a KeyRef that refers to the *JEP30-E101:FunctionKey*.

4.5.6. Linking the Manufacturing Part Number to Electrical Specification

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<p>The diagram shows the ElectricalParametersAssociationType class with the following attributes:</p> <ul style="list-style-type: none"> ElectricalParametersID (xs:string) PartClassificationID (xs:string) TerminalDetailsID (xs:string) SuperInterfaceID (xs:string) with multiplicity 0..∞ FunctionID (xs:string) with multiplicity 0..∞ ElectricalSpecificationID (xs:string) with multiplicity 0..∞ (highlighted in red circle A5) TruthTableID (xs:string) with multiplicity 0..∞ ESD-ID (xs:string) with multiplicity 0..∞ ElectricalParametersSignature (JEP30-D10:SignatureDigestLinkType)
type	ElectricalParametersAssociationType , JEP30-D10:SignatureDigestLinkType .
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification
diagram at the Electrical Specification-Array level	<p>The diagram shows the ElectricalSpecification-ArrayType class with the following attributes:</p> <ul style="list-style-type: none"> ElectricalSpecification (ElectricalSpecificationType) with multiplicity 0..∞ TruthTable (TruthTableType) with multiplicity 0..∞ Footnote-Array (JEP30-D10:Footnote-ArrayType) <p>The ElectricalSpecification attribute points to the ElectricalSpecificationType class, which contains:</p> <ul style="list-style-type: none"> ID (xs:string) with multiplicity 0..∞ (highlighted in red circle A5) TestCondition (ElectricalSpecificationTestConditionType) with multiplicity 0..∞ ParameterSet (ElectricalSpecificationParameterSetType) with multiplicity 1..∞ ParameterGraph (ElectricalSpecificationParameterGraphType) with multiplicity 1..∞
type	ElectricalSpecification-ArrayType , ElectricalSpecificationType , ...

The **ElectricalSpecificationID** references the **ElectricalSpecification/ID** under the [ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array](#). This is enforced by the key named as **ElectricalSpecificationKey** that is assigned to the **ElectricalSpecification/ID** element, which is referenced by the **ElectricalSpecificationID** which has a KeyRef that refers to the [JEP30-E101:ElectricalSpecificationKey](#).

4.5.7. Linking the Manufacturing Part Number to Truth Table

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<pre> classDiagram class ElectricalParametersAssociationType { ElectricalParametersID PartClassificationID TerminalDetailsID SuperInterfaceID FunctionID ElectricalSpecificationID TruthTableID ESD-ID ElectricalParametersSignature } class ElectricalParameters { <<ElectricalParametersAssociationType>> } ElectricalParameters "1" --> "1" ElectricalParametersAssociationType </pre>
type	ElectricalParametersAssociationType, JEP30-D10:SignatureDigestLinkType.
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/TruthTable
diagram at the Electrical Specification-Array level	<pre> classDiagram class ElectricalSpecificationArrayType { ElectricalSpecification } class TruthTableType { ID HeaderRow TerminalName DataRowLogicStates } class ElectricalSpecification-Array { <<ElectricalSpecificationArrayType>> } class TruthTable { <<TruthTableType>> } ElectricalSpecification-Array "1" --> "1" TruthTable </pre>
type	TruthTable-ArrayType, TruthTableType, ...

The *TruthTableID* references the *TruthTable/ID* under the *ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TruthTableID-Array*. This is enforced by the key named as *TruthTableKey* that is assigned to the *TruthTable/ID* element, which is referenced by the *TruthTableID* which has a KeyRef that refers to the *JEP30-E101:TruthTableKey*.

4.5.8. Linking the Manufacturing Part Number to ESD

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters
diagram at the Electrical Parameters Association level	<pre> classDiagram class ElectricalParametersAssociationType { ElectricalParametersID PartClassificationID TerminalDetailsID SuperInterfaceID "0..∞" FunctionID "0..∞" ElectricalSpecificationID "0..∞" TruthTableID "0..∞" ESD-ID "0..∞" ElectricalParametersSignature } class ElectricalParameters { type ElectricalParametersAssociationType } ElectricalParameters "0..∞" --> ElectricalParametersAssociationType </pre>
type	ElectricalParametersAssociationType, JEP30-D10:SignatureDigestLinkType.
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array
diagram at the Electrical Specification-Array level	<pre> classDiagram class ESDArrayType { ESDType ID "1..1" HBM-HumanBodyModel "1..1" CDM-ChargedDeviceModel "1..1" } class ESDArray { type ESDArrayType } ESDArray "0..∞" --> ESDType </pre>
type	ESD-ArrayType, ESDType, ...

The [ESD-ID](#) references the [ESD/ID](#) under the [ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array](#). This is enforced by the key named as [ESD-Key](#) that is assigned to the [ESD/ID](#) element, which is referenced by the [ESD-ID](#) which has a KeyRef that refers to the [JEP30-E101:ESD-Key](#).

4.5.9. Linking the Manufacturing Part Number to Schematic Data

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/SchematicData
diagram at the Schematic Data Association level	<pre> classDiagram class SchematicDataAssociationType { SchematicDataID SymbolID RequiredCircuitryID SchematicDataSignature } class SchematicData { <<SchematicDataAssociationType>> } SchematicData "1" --> "1" SchematicDataAssociationType SchematicDataAssociationType "1" --> "1" B </pre>
type	SchematicDataAssociationType , JEP30-D10:SignatureDigestLinkType .
path	PartModel/ElectricalSection/SchematicData-Array
diagram at the Schematic Data-Array level	<pre> classDiagram class SchematicData-ArrayType { SchematicData-Array SchematicData SchematicDataSignature } class SchematicData { <<SchematicData-ArrayType>> } SchematicData-Array "1..>" --> "1" SchematicData SchematicData "1" --> "1" B </pre>
type	SchematicData-ArrayType , SchematicDataType , ...

The [SchematicDataID](#) references the [SchematicData/ID](#) under the [ElectricalSection/SchematicData-Array](#). This is enforced by the key named as [SchematicDataKey](#) that is assigned to the [SchematicData/ID](#) element, which is referenced by the [SchematicDataID](#) which has a KeyRef that refers to the [JEP30-E101:SchematicDataKey](#).

4.5.10. Linking the Manufacturing Part Number to Symbol

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/SchematicData
diagram at the Schematic Data Association level	<pre> classDiagram class SchematicDataAssociationType { SchematicData SchematicDataID SymbolID RequiredCircuitryID SchematicDataSignature } SchematicData < -- SchematicDataAssociationType SchematicDataID < -- SchematicDataAssociationType SymbolID < -- SchematicDataAssociationType RequiredCircuitryID < -- SchematicDataAssociationType SchematicDataSignature < -- SchematicDataAssociationType SchematicData --> SchematicDataAssociationType SchematicDataAssociationType --> SchematicDataID SchematicDataAssociationType --> SymbolID SchematicDataAssociationType --> RequiredCircuitryID SchematicDataAssociationType --> SchematicDataSignature </pre>
type	SchematicDataAssociationType, JEP30-D10:SignatureDigestLinkType.
path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array
diagram at the Symbol-Array level	<pre> classDiagram class JEP30-D10:Symbol-ArrayType { Symbol-Array Symbol ID SymbolGraphics-Array SymbolSignature } Symbol-Array < -- JEP30-D10:Symbol-ArrayType Symbol < -- JEP30-D10:Symbol-ArrayType ID < -- JEP30-D10:Symbol-ArrayType SymbolGraphics-Array < -- JEP30-D10:Symbol-ArrayType SymbolSignature < -- JEP30-D10:Symbol-ArrayType constraints < -- JEP30-D10:Symbol-ArrayType </pre>
type	Symbol-ArrayType, SymbolType, ...

The *SymbolID* references the *Symbol/ID* under the *ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array*. This is enforced by the key named as *SymbolKey* that is assigned to the *Symbol /ID* element, which is referenced by the *SymbolID* which has a KeyRef that refers to the *JEP30-E101:SymbolKey*.

4.5.11. Linking the Manufacturing Part Number to Required Circuitry

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/SchematicData
diagram at the Schematic Data Association level	<pre> classDiagram class SchematicDataAssociationType { SchematicDataID SymbolID RequiredCircuitryID SchematicDataSignature } class SchematicData { <<SchematicDataAssociationType>> } SchematicData "1" -- "1" SchematicDataAssociationType SchematicDataAssociationType "1" -- "1" RequiredCircuitryID RequiredCircuitryID --> B2 </pre>
type	SchematicDataAssociationType, JEP30-D10:SignatureDigestLinkType.
path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array
diagram at the Required Circuitry-Array level	<pre> classDiagram class RequiredCircuitry { ID Net-Array ElectricalSpecificationID RequiredCircuitrySignature } class RequiredCircuitry-Array { <<RequiredCircuitry-ArrayType>> } RequiredCircuitry-Array "1" -- "1" RequiredCircuitry RequiredCircuitry "1..<--> 0..<--> ElectricalSpecificationID ElectricalSpecificationID --> B2 </pre>
type	RequiredCircuitry-ArrayType, RequiredCircuitryType, ...

The *RequiredCircuitryID* references the *RequiredCircuitry/ID* under the *ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array*. This is enforced by the key named as *RequiredCircuitryKey* that is assigned to the *RequiredCircuitry/ID* element, which is referenced by the *RequiredCircuitryID* which has a KeyRef that refers to the *JEP30-E101:RequiredCircuitryKey*.

4.5.12. Linking the Manufacturing Part Number to Mapping data

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping
diagram at the Mapping Association level	<p>The diagram shows the MappingAssociationType class with the following structure:</p> <ul style="list-style-type: none"> Attributes: <ul style="list-style-type: none"> MappingID: type xs:string ElectricalMapID: type xs:string TerminalMapID: type xs:string PackageTerminalMapID: type xs:string DieTerminalMapID: type xs:string SimulationMapID: type xs:string SimulationModel: type SimulationModelAssociationType MappingSignature: type JEP30-D10:SignatureDigestLinkType Associations: <ul style="list-style-type: none"> A dashed line connects Mapping (type MappingAssociationType) to MappingAssociationType.
type	MappingAssociationType, JEP30-D10:SignatureDigestLinkType, ...
path	PartModel/ElectricalSection/Mapping-Array
diagram at the Mapping-Array level	<p>The diagram shows the MappingType class with the following structure:</p> <ul style="list-style-type: none"> Attributes: <ul style="list-style-type: none"> ID: type xs:string ElectricalMap: type ElectricalMapType (multiplicity 0..∞) TerminalMap: type TerminalMapType (multiplicity 0..∞) PackageTerminalMap: type PackageTerminalMapType (multiplicity 0..∞) DieTerminalMap: type DieTerminalMapType (multiplicity 0..∞) SimulationMap: type SimulationMapType (multiplicity 0..∞) MappingSignature: type IdsSignatureType Associations: <ul style="list-style-type: none"> A dashed line connects Mapping (type MappingType) to MappingType. <p>Annotations for the TerminalMap, PackageTerminalMap, and DieTerminalMap attributes provide information about their obsolescence:</p> <ul style="list-style-type: none"> TerminalMap: Use this Terminal Map structure which combines the package and die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. The Package and Die Terminal Map structures below will go obsolete in first release in 2028. PackageTerminalMap: Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Package Terminal Map will go obsolete in first release in 2028. DieTerminalMap: Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Die Terminal Map will go obsolete in first release in 2028.
type	Mapping-ArrayType, MappingType, ...

The **MappingID** references the **Mapping/ID** under the **ElectricalSection/Mapping-Array**. This is enforced by the key named as **MappingKey** that is assigned to the **Mapping/ID** element, which is referenced by the **MappingID** which has a KeyRef that refers to the **JEP30-E101:MappingKey**.

4.5.13. Linking the Manufacturing Part Number to Electrical Map data

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping
diagram at the Mapping Association level	<p>The diagram shows the <code>MappingAssociationType</code> class with the following fields:</p> <ul style="list-style-type: none"> <code>MappingID</code>: type xs:string <code>ElectricalMapID</code>: type xs:string <code>TerminalMapID</code>: type xs:string <code>PackageTerminalMapID</code>: type xs:string <code>DieTerminalMapID</code>: type xs:string <code>SimulationMapID</code>: type xs:string <code>SimulationModel</code>: type <code>SimulationModelAssociationType</code> <code>MappingSignature</code>: type <code>JEP30-D10:SignatureDigestLinkType</code>
type	MappingAssociationType , JEP30-D10:SignatureDigestLinkType , ...
path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap
diagram at Terminal Map level	<p>The diagram shows the <code>ElectricalMapType</code> class with the following fields:</p> <ul style="list-style-type: none"> <code>ID</code>: type xs:string <code>OperationalMode</code>: type <code>OperationalModeType</code> <code>Terminal</code>: type <code>TerminalElectricalMapType</code> (multiplicity 1..∞) <code>DifferentialPairID</code>: type xs:string (multiplicity 1..∞) <code>Logical-GroupID</code>: type xs:string (multiplicity 1..∞) <code>SuperInterfaceID</code>: type xs:string (multiplicity 1..∞) <code>FunctionID</code>: type xs:string (multiplicity 1..∞) <code>RequiredCircuitryID</code>: type xs:string (multiplicity 1..∞) <code>ReferenceDesignID</code>: type xs:string (multiplicity 1..∞)
type	ElectricalMapType , ...

The `ElectricalMapID` references the `Mapping/ID` under the `ElectricalSection/Mapping-Array/Mapping/ElectricalMap`. This is enforced by the key named as `ElectricalMapKey` that is assigned to the `ElectricalMap/ID` element, which is referenced by the `ElectricalMapID` which has a KeyRef that refers to the `JEP30-E101:ElectricalMapKey`.

4.5.14. Linking the Manufacturing Part Number to Terminal Map data

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping
diagram at the Mapping Association level	<p>The diagram shows the MappingAssociationType class with the following fields:</p> <ul style="list-style-type: none"> MappingID: type xs:string ElectricalMapID: type xs:string TerminalMapID: type xs:string PackageTerminalMapID: type xs:string DieTerminalMapID: type xs:string SimulationMapID: type xs:string SimulationModel: type SimulationModelAssociationType MappingSignature: type JEP30-D10:SignatureDigestLinkType
type	MappingAssociationType , JEP30-D10:SignatureDigestLinkType , ...
path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap
diagram at Terminal Map level	<p>The diagram shows the TerminalMapType class with the following fields:</p> <ul style="list-style-type: none"> ID: type xs:string PackageID: type xs:string DieID: type xs:string Map: type MapType <p>A note below states: "Use this Terminal Map structure which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. The Package and Die Terminal Map structures below will go obsolete in first release in 2028"</p>
type	PackageTerminalMapType , ...

The **TerminalMapID** references the **TerminalMap/ID** under the [ElectricalSection/Mapping-Array/Mapping/TerminalMap](#). This is enforced by the key named as **TerminalMapKey** that is assigned to the **TerminalMap /ID** element, which is referenced by the **TerminalMapID** which has a KeyRef that refers to the [JEP30-E101: TerminalMapKey](#).

NOTE: Use this Terminal Map structure which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. The Package and Die Terminal Map structures below will go obsolete in first release in 2028

4.5.15. Linking the Manufacturing Part Number to Package Terminal Map data

	path PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping
diagram at the Mapping Association level	<p>The diagram shows the MappingAssociationType class with associations to Mapping, MappingID, ElectricalMapID, TerminalMapID, PackageTerminalMapID, DieTerminalMapID, SimulationMapID, SimulationModel, and MappingSignature. A red arrow points from the PackageTerminalMapID association to a circled 'C3'.</p>
type	MappingAssociationType , JEP30-D10:SignatureDigestLinkType , ...
path	PartModel/ElectricalSection/Mapping-Array/Mapping/PackageTerminalMap
diagram at Terminal Map level	<p>The diagram shows the PackageTerminalMapType class with associations to PackageTerminalMap, ID, PackageID, and TerminalMap. A red arrow points from the ID association to a circled 'C3'. A note states: "Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Package Terminal Map will go obsolete in first release in 2028".</p>
type	PackageTerminalMapType , ...

The **PackageTerminalMapID** references the **PackageTerminalMap/ID** under the **ElectricalSection/Mapping-Array/Mapping/PackageTerminalMap**. This is enforced by the key named as **PackageTerminalMapKey** that is assigned to the **PackageTerminalMap /ID** element, which is referenced by the **PackageTerminalMapID** which has a KeyRef that refers to the **JEP30-E101:PackageTerminalMapKey**.

NOTE: Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Package Terminal Map will go obsolete in first release in 2028

4.5.16. Linking the Manufacturing Part Number to Die Terminal Map data

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping
diagram at the Mapping Association level	<p>The diagram shows the MappingAssociationType class with the following attributes:</p> <ul style="list-style-type: none"> MappingID: type xs:string ElectricalMapID: type xs:string TerminalMapID: type xs:string PackageTerminalMapID: type xs:string DieTerminalMapID: type xs:string (highlighted with a red arrow labeled C4) SimulationMapID: type xs:string SimulationModel: type SimulationModelAssociationType MappingSignature: type JEP30-D10:SignatureDigestLinkType
type	MappingAssociationType , JEP30-D10:SignatureDigestLinkType , ...
path	PartModel/ElectricalSection/Mapping-Array/Mapping/DieTerminalMap
diagram at Terminal Map level	<p>The diagram shows the DieTerminalMapType class with the following attributes:</p> <ul style="list-style-type: none"> ID: type xs:string DieID: type xs:string TerminalMap: type PackageDieTerminalMapType (highlighted with a red arrow labeled C4) <p>A note below the diagram states: "Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Die Terminal Map will go obsolete in first release in 2028." A red arrow labeled C4 points from the DieTerminalMapID in the previous diagram to the ID attribute here.</p>
type	TerminalMapType , ...

The **DieTerminalMapID** references the **DieTerminalMap/ID** under the [ElectricalSection/Mapping-Array//Mapping/DieTerminalMap](#). This is enforced by the key named as [DieTerminalMapKey](#) that is assigned to the **DieTerminalMap/ID** element, which is referenced by the **DieTerminalMapID** which has a KeyRef that refers to the [JEP30-E101: DieTerminalMapKey](#).

NOTE: Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Die Terminal Map will go obsolete in first release in 2028.

4.5.17. Linking the Manufacturing Part Number to Simulation Map data

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping
diagram at the Mapping Association level	<pre> classDiagram class MappingAssociationType { MappingID ElectricalMapID TerminalMapID PackageTerminalMapID DieTerminalMapID SimulationMapID SimulationModel MappingSignature } class Mapping { <<MappingAssociationType>> } Mapping "1" -- "1" MappingAssociationType : <<MappingAssociationType>> MappingAssociationType "1" -- "1" C5 : SimulationMapID </pre>
type	MappingAssociationType , JEP30-D10:SignatureDigestLinkType , ...
path	PartModel/ElectricalSection/Mapping-Array/Mapping/SimulationMap
diagram at the Simulation Map level	<pre> classDiagram class SimulationMapType { ID TerminalMapID DifferentialPairID Logical-GroupID SuperInterfaceID FunctionID RequiredCircuitryID ReferenceDesignID SimulationModelID } class SimulationMap { <<SimulationMapType>> } SimulationMap "0..>" -- "1..>" SimulationMapType : <<SimulationMapType>> SimulationMapType "1..>" C5 : ID </pre>
type	SimulationMapType .

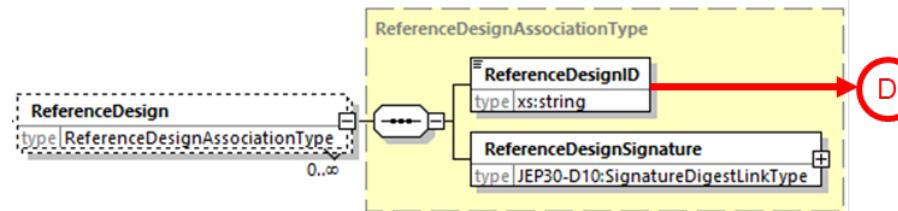
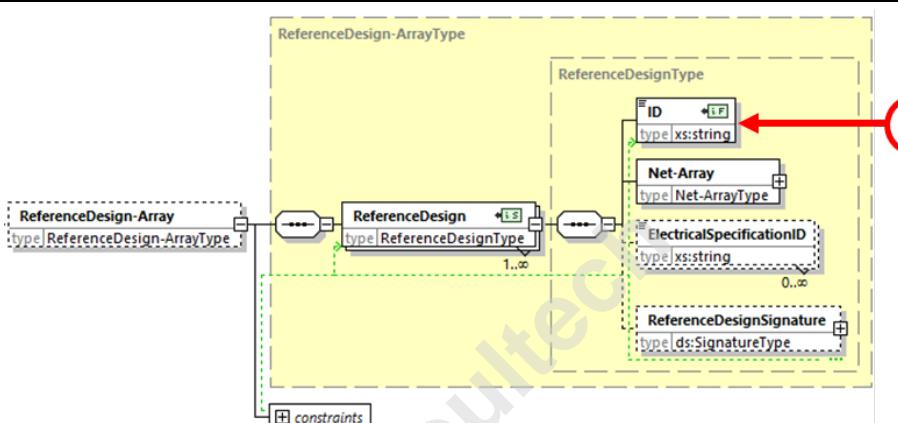
The [SimulationMapID](#) references the [SimulationMap/ID](#) under the [ElectricalSection/Mapping-Array/Mapping/SimulationMap](#). This is enforced by the key named as [SimulationMapKey](#) that is assigned to the [SimulationMap/ID](#) element, which is referenced by the [SimulationMapID](#) which has a KeyRef that refers to the [JEP30-E101:SimulationMapKey](#).

4.5.18. Linking the Manufacturing Part Number to Simulation Models

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping/SimulationModel
diagram at the Simulation Model Association level	<pre> classDiagram class JEP30-E101:SimulationModelAssociationType { SimulationModelID : xs:string SimulationModelSignature : JEP30-D10:SignatureDigestLinkType } class SimulationModel { <> } JEP30-E101 < -- SimulationModel JEP30-E101 "1..1" --> SimulationModel </pre>
type	JEP30-E101:SimulationModelAssociationType , JEP30-D10:SignatureDigestLinkType .
path	PartModel/ElectricalSection/SimulationModel-Array
diagram at the Simulation Model-Array level	<pre> classDiagram class SimulationModel-ArrayType { <> } class SimulationModel { ID : xs:string SPICE-Functional : SPICE-FunctionalType SPICE-SignalIntegrity : SPICE-SignalIntegrityType IBIS : JEP30-D10:EmptyType OtherModelType : xs:string Model : xs:string ModelDescription : xs:string SimulationModelSignature : ds:SignatureType } class constraints SimulationModel-ArrayType "1..>" --> SimulationModel </pre>
type	SimulationModel-ArrayType , SimulationModelType , ...

The [SimulationModelID](#) references the [SimulationModel/ID](#) under the [ElectricalSection/SimulationModel-Array/SimulationModel](#). This is enforced by the key named [SimulationModelKey](#) is assigned to the [SimulationModel/ID](#) element, which is referenced by the [SimulationModelID](#) which has a KeyRef that refers to the [JEP30-E101: SimulationModelKey](#).

4.5.19. Linking the Manufacturing Part Number to Reference Design

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ReferenceDesign
diagram at the Reference Design Association level	 <pre> classDiagram class ReferenceDesignAssociationType { ReferenceDesignID : xs:string ReferenceDesignSignature : JEP30-D10:SignatureDigestLinkType } class ReferenceDesign { * ReferenceDesign * type ReferenceDesignAssociationType } ReferenceDesign "0..>" ReferenceDesignAssociationType </pre>
type	ReferenceDesignAssociationType , JEP30-D10:SignatureDigestLinkType , ...
path	PartModel/ElectricalSection/ReferenceDesign-Array
diagram at the Reference Design-Array level	 <pre> classDiagram class ReferenceDesign-ArrayType { ReferenceDesign * type ReferenceDesignArrayType } class ReferenceDesignType { ID : xs:string Net-Array : Net-ArrayType ElectricalSpecificationID : xs:string 0..> ReferenceDesignSignature : ds:SignatureType } ReferenceDesign "1..>" ReferenceDesignType </pre>
type	ReferenceDesign-ArrayType , ReferenceDesignType , ...

The [ReferenceDesignID](#) references the [ReferenceDesign/ID](#) under the [ElectricalSection/SchematicData-Array/ReferenceDesign-Array](#). This is enforced by the key named as [ReferenceDesignKey](#) is assigned to the [ReferenceDesign/ID](#) element, which is referenced by the [ReferenceDesignID](#) which has a KeyRef that refers to the [JEP30-E101:ReferenceDesignKey](#).

4.5.20. Linking the Manufacturing Part Number to Software Interface Description

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/SoftwareInterfaceDescription
diagram at the Software Interface Description Association level	<p>The diagram shows a class named SoftwareInterfaceDescriptionAssociationType. It contains two associations: one to SoftwareInterfaceDescription and another to SoftwareInterfaceDescriptionDesignSignature. The association to SoftwareInterfaceDescription has a key named SoftwareInterfaceDescriptionKey pointing to the SoftwareInterfaceDescriptionID element. A red arrow labeled 'E' points to the SoftwareInterfaceDescriptionID element.</p>
type	SoftwareInterfaceDescriptionAssociationType , JEP30-D10:SignatureDigestLinkType , ...
path	PartModel/ElectricalSection/SoftwareInterfaceDescription-Array
diagram at the Software Interface Description -Array level	<p>The diagram shows a class named SoftwareInterfaceDescription-ArrayType. It contains an association to SoftwareInterfaceDescription. The SoftwareInterfaceDescription class has a multiplicity of 1..*. The SoftwareInterfaceDescription class also has an association to SoftwareInterfaceDescriptionType. The SoftwareInterfaceDescriptionType class contains elements: ID (with a KeyRef), IP-XACT, SystemRDL, and SoftwareInterfaceSignature. A red arrow labeled 'E' points to the ID element.</p>
type	SoftwareInterfaceDescription-ArrayType , SoftwareInterfaceDescription Type , ...

The **SoftwareInterfaceDescriptionID** references the **SoftwareInterfaceDescription/ID** under the [ElectricalSection/SchematicData-Array/SoftwareInterfaceDescription-Array](#). This is enforced by the key named as **SoftwareInterfaceDescriptionKey** is assigned to the **SoftwareInterfaceDescription/ID** element, which is referenced by the **SoftwareInterfaceDescriptionID** which has a KeyRef that refers to the [JEP30-E101: SoftwareInterfaceDescriptionKey](#).

4.6. Electrical Section

path	PartModel/ElectricalSection
diagram	<pre> classDiagram class ElectricalSection { type ElectricalSectionType } class ElectricalSectionType { "ElectricalSection" "ElectricalParameters-Array" "SchematicData-Array" "Mapping-Array" "SimulationModel-Array" "ReferenceDesign-Array" "constraints" } ElectricalSection "o"--> ElectricalSectionType </pre>
type	ElectricalParameters-ArrayType, SchematicData-ArrayType, Mapping-ArrayType, SimulationModel-ArrayType, ReferenceDesign-ArrayType.

The Electrical section is grouped into 5 individual sections to facilitate the digital signing of the content, so that the customer can obtain a high degree of confidence in the integrity of the content. The content is organized such that all the electrical properties are grouped under the **ElectricalParameters-Array** branch. Content that is required to support circuitry design, namely Symbols and Required Circuitry are grouped under **SchematicData-Array**.

When the electrical representation of the data is mapped to a package or to simulation models, then this data is represented under the **Mapping-Array** branch. Some parts have various kinds of functional and simulation models that describe the operation and / or performance of the part. These models are grouped under the **SimulationModel-Array** branch. Component manufacturers, distributors, data aggregators or any service bureaus sometimes take hero parts and develop partial or full reference designs. These reference designs are grouped under the **ReferenceDesign-Array** branch.

These are described in more detail in the following sections.

4.7. Electrical Parameters

path	PartModel/ElectricalSection/ElectricalParameters-Array
diagram	<p>The diagram illustrates the UML class structure for the ElectricalParameters-ArrayType. It features a main class ElectricalParameters with the following associations:</p> <ul style="list-style-type: none"> A dashed line connects ElectricalParameters to ElectricalParameters-Array, labeled type ElectricalParameters-ArrayType. A dashed line connects ElectricalParameters to ElectricalParametersType, labeled type ElectricalParametersType. A dashed line connects ElectricalParameters to a multiplicity of 1..∞ (represented by a green dashed box), which further connects to PartClassification-Array, TerminalDetails-Array, FunctionGroup-Array, ElectricalSpecification-Array, and ESD-Array. A dashed line connects ElectricalParameters to ds:Signature. <p>Additional components shown include attributes (with ID and type xs:string) and constraints.</p>
type	ElectricalParameters-ArrayType , ElectricalParametersType , PartClassification-ArrayType , Electrical-ArrayType , ElectricalSpecification-ArrayType , ds:SignatureType

The **ElectricalParameters-Array** section captures electrical content about the Part in basically the following groups of data, namely:-

1. Part Classification
2. Terminal Details,
3. FunctionGroup-Array details which can be assigned to a discrete device or can represent just one component of the device, and
4. Electrical Specifications
5. Electrostatic Discharge (**ESD**) limitations.

4.7.1. Part Classification - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array
diagram	<p>The diagram illustrates the UML Class Diagram for PartClassification-ArrayType. It shows a class PartClassification with multiplicity 1..* associated with PartClassificationType (multiplicity 1..*) and CompanionPart (multiplicity 0..*). PartClassificationType contains attributes ID (xs:string) and CableAndWiring (CableAndWiringClassificationType). CompanionPart contains attributes ID (xs:string), OrderablePartNumberID (xs:string), PartDetailsID (xs:string), and ReferencePartDetailsID (xs:string). A constraint labeled constraints is present at the bottom.</p>
type	PartClassification-ArrayType , PartClassificationType , CableAndWiringClassificationType , ConnectorClassificationType , ElectricalClassificationType , HardwareClassificationType , OpticsClassificationType , CompanionPartType .

A Part can be classified into one of 5 major types, namely as:-

1. *CableAndWiring*,
2. *Connector*,
3. *Electrical*,
4. *Hardware*, or
5. *Optic*.

Even if a Part is classified as Connector, Hardware or Optics, it may also have electrical data that needs to be captured under some of the other sections.

A *CompanionPart* is a part that accompanies this Part that is being reference in the PartModel. This *CompanionPart* could be for example

- A heatsink that is required for this electrical device

4.7.1 Part Classification - Array (cont'd)

- Attachment Hardware for use with a connector during assembly, such as screws, nuts, clips, etc
- A mating connector
- An electrical component that is fitted into a Socket in order to be attached to the PB Assembly, such as Memory Modules requiring a DIMM socket.
- A metal cage or housing for various connector or for an electrical device that requires shielding.

The *CompanionPart* is another entry within the PartModel file and can have its own technical content structure. Alternatively, the *CompanionPart* could be stored in a separate PartModel file. The identity of the *CompanionPart* could be via an Orderable Part Number that is already within this PartModel file or referenced to a Manufacturer's Part Number from a specific Manufacturer. In some cases, a hard identity is not required such as attachment hardware, in such case a reference to a PartDetails ID or a ReferencePartDetails ID adequate where definition of a simple set of specifications for the part is adequate.

Each of the 5 classifications above have a hierarchical structure that sub-divide the higher-level classification into smaller and smaller grouping of parts. At all levels, a set of properties can be defined under a Property-Array that provide additional details for the parts. The purpose of classifying parts into a hierarchical structure is to assist the user to search for the parts that they are looking for. Because of this a deep hierarchical structure is not desired and an optimal 2 or 3 levels is preferred. Below level 3, any sub-classification is suggested to be represented as opposed to a further deepening of the classification structure.

To clearly distinguish the separation of classification levels from properties, an *Attribute* called *ClassificationProperties* is added to the element “*Property-Array*” under which all properties for that part (or group of parts) are added. If the PartModel represents a single part, then typically all the relevant properties are assigned to the lowest level classification. If the Part is assigned to a parent classification and not to the child classifications, then the part properties are assigned to the *Property-Array* at the level in the hierarchy where the classification map is made. This enables the scalability of defining *PartModel* data for parts whose sub-classifications have not yet been defined. All schema structures containing the element *Property-Array* will contain the structure as shown below by default. If additional properties are added under the *Property-Array*, then that *Property-Array* structure will be expanded.

Each level of the *PartClassification* tree in which there are sub-categories, an *OverflowCategories* group is added. This group contains the original *Other* sub-category but is joined by the addition of *Parts* and *Accessories*. The *Other* sub-category contains the objects that cannot be classified into other specified sub-categories in the existing structure, but that are classified to their parent-classification category. The *Parts* sub-category contains the objects with spare part characteristics that maintain or re-store the original condition of the objects classified under the parent-classification category. The *Accessories* sub-category contains the objects with complementary characteristics, without which the basic function of the objects classified under the parent-classification category is still guaranteed. Some *OverflowCategories* groups have additional properties identified under some of the sub-category branches under the *Property-Array* because it is deemed such properties are ideal for the parent-classification category.

4.7.1 Part Classification - Array (cont'd)

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/....(all classification branches to lowest level classification branch. The Resistor branch is just one of these branches)
diagram part 1 of 2	<pre> classDiagram class ResistorClassificationType class FixedResistorClassificationType class Fixed class Adjustable class NonLinearResistor class Other class Parts class Accessories class Property class ParameterSetID class ParameterID ResistorClassificationType < -- FixedResistorClassificationType ResistorClassificationType < -- Other ResistorClassificationType < -- Parts ResistorClassificationType < -- Accessories FixedResistorClassificationType < -- Fixed FixedResistorClassificationType < -- Adjustable FixedResistorClassificationType < -- NonLinearResistor FixedResistorClassificationType < -- Other FixedResistorClassificationType < -- Parts FixedResistorClassificationType < -- Accessories FixedResistorClassificationType < -- Property FixedResistorClassificationType < -- ParameterSetID FixedResistorClassificationType < -- ParameterID </pre>
type (sample specifically for Resistor category)	ResistorClassificationType, FixedResistorClassificationType, FixedResistorClassificationProperty-ArrayType, JEP30-D10:EmptyType, FixedResistorMaterialPropertyType, FixedResistorClassificationPropertyType, AdjustableResistorClassificationType, NonLinearResistorClassificationType, OverflowCategoriesType, ResistorClassificationProperty-ArrayType, PropertyKeyValuePairType.
diagram part 2 of 2	<pre> classDiagram class OverflowCategoriesType class OverflowCategoriesProperty-ArrayType class Property class ParameterSetID class ParameterID OverflowCategoriesType < -- Other OverflowCategoriesType < -- Parts OverflowCategoriesType < -- Accessories OverflowCategoriesType < -- SubCategoryName OverflowCategoriesProperty-ArrayType < -- attributes OverflowCategoriesProperty-ArrayType < -- SearchableProperties OverflowCategoriesProperty-ArrayType < -- Property OverflowCategoriesProperty-ArrayType < -- ParameterSetID OverflowCategoriesProperty-ArrayType < -- ParameterID </pre>
type	OverflowCategoriesType, OverflowCategoriesProperty-ArrayType, PropertyKeyValuePairType.
group	OverflowCategories, SearchableProperties.

4.7.1 Part Classification Array (cont'd)

Classifications under the category *Other*, *Parts* and *Accessories* are candidates for future standardization via this publication. The *Sub-CategoryName* enables the component manufacturer to propose specific sub-classifications via this structure.

The ParameterID and ParameterSetID allows the connection of electrical parameters that are specified under the ElectricalSpecification-Array to be referenced under the PartClassification as suitable parameters for the purpose of enabling the user to search for a part in a component DB.

4.7.1.1 Cable and Wiring Classification

path	ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/CableAndWiring
diagram	<pre> classDiagram class CableAndWiringClassificationType { <<CableAndWiringClassificationType>> <<CableAndWiring>> <<OverflowCategoriesType>> <<Property-Array>> <<CableAndWiringClassificationProperty-ArrayType>> <<attributes>> <<Gauge>> <<SearchableProperties>> } class RoundCable { <<RoundCableClassificationType>> } class TwistedPair { <<TwistedPairClassificationType>> } class FlatCable { <<FlatCableClassificationType>> } class RibbonCable { <<RibbonCableClassificationType>> } class Other { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } class OverflowCategories { <<OverflowCategoriesType>> } CableAndWiringClassificationType < -- CableAndWiring CableAndWiringClassificationType < -- OverflowCategories CableAndWiringClassificationType < -- Property-Array CableAndWiringClassificationType < -- attributes CableAndWiringClassificationType < -- Gauge CableAndWiringClassificationType < -- SearchableProperties CableAndWiring --> > RoundCable CableAndWiring --> > TwistedPair CableAndWiring --> > FlatCable CableAndWiring --> > RibbonCable CableAndWiring --> > Other CableAndWiring --> > Parts CableAndWiring --> > Accessories CableAndWiring --> > OverflowCategories CableAndWiring --> > Property-Array CableAndWiring --> > attributes CableAndWiring --> > Gauge CableAndWiring --> > SearchableProperties </pre>
type	CableAndWiringClassificationType , RoundCableClassificationType , TwistedPairClassificationType , FlatCableClassificationType , RibbonCableClassificationType , OverflowCategoriesType , CableAndWiringClassificationProperty-ArrayType .
group	OverflowCategories , SearchableProperties .

Each of the above sub-classifications of *RoundCable*, *TwistedPair*, *FlatCable*, *RibbonCable* and *Other* have their own dedicated *Property-Array*, which will enable future dedicated properties to be added to any one sub-classification and be unique to that sub-classification.

The *Property-Array* that is located directly under the parent *CableAndWiring*, with the *CableAndWiringClassificationProperty-ArrayType*, has a unique property *Gauge* defined. This therefore applies to all sub-classifications of *RoundCable*, *TwistedPair*, *FlatCable*, *RibbonCable* and each of the sub-categories under the *OverflowCategories* group, namely *Other*, *Parts*, and *Accessories*. Each *Property* has its own key value pairs, namely *Name* and *Value*.

4.7.1.2. Connector Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector
diagram	<pre> classDiagram class ConnectorClassificationType { <<ConnectorClassificationType>> } class CardEdge { <<CardEdgeConnectorClassificationType>> } class Optical { <<OpticalConnectorClassificationType>> } class Socket { <<SocketConnectorClassificationType>> } class Board { <<BoardConnectorClassificationType>> } class Cable { <<CableConnectorClassificationType>> } class Other { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } class OverflowCategories { <<OverflowCategories>> } class Property_Array { <<ConnectorClassificationProperty-ArrayType>> } ConnectorClassificationType < -- CardEdge ConnectorClassificationType < -- Optical ConnectorClassificationType < -- Socket ConnectorClassificationType < -- Board ConnectorClassificationType < -- Cable ConnectorClassificationType < -- Other ConnectorClassificationType < -- Parts ConnectorClassificationType < -- Accessories ConnectorClassificationType --> OverflowCategories OverflowCategories --> Property_Array </pre> <p>The diagram illustrates the UML class structure for ConnectorClassificationType. It features a main class, ConnectorClassificationType, which is a generalization of several specific connector types: CardEdge, Optical, Socket, Board, and Cable. Additionally, it includes categories for Other, Parts, and Accessories, all of which are instances of OverflowCategoriesType. There is also a relationship between ConnectorClassificationType and OverflowCategories, and between OverflowCategories and Property_Array.</p>
type	ConnectorClassificationType , CardEdgeConnectorClassificationType , OpticalConnectorClassificationType , SocketConnectorClassificationType , BoardConnectorClassificationType , CableConnectorClassificationType , OverflowCategoriesType , ConnectorClassificationProperty-ArrayType .
group	OverflowCategories .

4.7.1.2.1. CardEdge Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/CardEdge
diagram	<pre> classDiagram class CardEdge { type CardEdgeConnectorClassificationType } class CardEdgeConnectorClassificationType { Backplane type CardEdgeBackplaneConnectorClassificationType Interface type CardEdgeInterfaceConnectorClassificationType Memory type CardEdgeMemoryConnectorClassificationType CardEdgeOverflowCategories type CardEdgeOverflowCategoriesType Property-Array type CardEdgeConnectorClassificationProperty-ArrayType } class Other { type CardEdgeOverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } CardEdge < -- CardEdgeConnectorClassificationType CardEdgeConnectorClassificationType < -- Backplane CardEdgeConnectorClassificationType < -- Interface CardEdgeConnectorClassificationType < -- Memory CardEdgeConnectorClassificationType < -- CardEdgeOverflowCategories CardEdgeOverflowCategories --> Property-Array CardEdgeOverflowCategories --> Other CardEdgeOverflowCategories --> Parts CardEdgeOverflowCategories --> Accessories </pre>
type	CardEdgeConnectorClassificationType , CardEdgeBackplaneConnectorClassificationType , CardEdgeInterfaceConnectorClassificationType , CardEdgeMemoryConnectorClassificationType , CardEdgeOverflowCategoriesType , OverflowCategoriesType , CardEdgeConnectorClassificationProperty-ArrayType
group	CardEdgeOverflowCategories

4.7.1.2.1.1. CardEdge Interface Connector Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/CardEdge/Interface
diagram 1 of 2	<pre> classDiagram class CardEdgeInterfaceConnectorClassificationType { <<CardEdgeInterfaceConnectorClassificationType>> } class Property { <<Property>> } class CardEdgeInterfaceConnectorClassificationPropertyArrayType { <<CardEdgeInterfaceConnectorClassificationPropertyArrayType>> } class CableStyle { <<CardEdgeInterfaceConnectorClassificationCableStylePropertyType>> } class SearchableProperties { <<SearchableProperties>> } class ParameterSetID { <<ParameterSetID>> } class ParameterID { <<ParameterID>> } CardEdgeInterfaceConnectorClassificationType "1" --> "1" Property-Array : CardEdgeInterfaceConnectorClassificationType "1" --> "1" CardEdgeInterfaceConnectorClassificationPropertyArrayType : CardEdgeInterfaceConnectorClassificationType "1" --> "1" CableStyle : CardEdgeInterfaceConnectorClassificationPropertyArrayType "*" --> "1" attributes : CardEdgeInterfaceConnectorClassificationPropertyArrayType "*" --> "1" SearchableProperties : CardEdgeInterfaceConnectorClassificationPropertyArrayType "*" --> "1" ParameterSetID : CardEdgeInterfaceConnectorClassificationPropertyArrayType "*" --> "1" ParameterID : </pre>
diagram 2 of 2	<pre> classDiagram class CardEdgeInterfaceConnectorClassificationCableStylePropertyType { <<CardEdgeInterfaceConnectorClassificationCableStylePropertyType>> } class FlatCable { <<FlatCable>> } class RibbonCable { <<RibbonCable>> } class RoundCable { <<RoundCable>> } class TwistedPair { <<TwistedPair>> } class Other { <<Other>> } CardEdgeInterfaceConnectorClassificationCableStylePropertyType "*" --> "1" FlatCable : CardEdgeInterfaceConnectorClassificationCableStylePropertyType "*" --> "1" RibbonCable : CardEdgeInterfaceConnectorClassificationCableStylePropertyType "*" --> "1" RoundCable : CardEdgeInterfaceConnectorClassificationCableStylePropertyType "*" --> "1" TwistedPair : CardEdgeInterfaceConnectorClassificationCableStylePropertyType "*" --> "1" Other : </pre>
type	CardEdgeInterfaceConnectorClassificationType , CardEdgeInterfaceConnectorClassificationProperty-ArrayType , CardEdgeInterfaceConnectorClassificationCableStylePropertyType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.2.1.2. CardEdge Overflow Categories Group

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/CardEdge
diagram 1 of 2	<pre> classDiagram class CardEdgeOverflowCategoriesType { Sub-CategoryName : xs:string Property-Array : CardEdgeOverflowCategoriesProperty-ArrayType } class CableStyle { type : CardEdgeOverflowCategoriesCableStylePropertyType } class Other { type : CardEdgeOverflowCategoriesType } class Parts { type : OverflowCategoriesType } class Accessories { type : OverflowCategoriesType } CardEdgeOverflowCategoriesType "1" -- "*" CableStyle CardEdgeOverflowCategoriesType "1" -- "*" Other CardEdgeOverflowCategoriesType "1" -- "*" Parts CardEdgeOverflowCategoriesType "1" -- "*" Accessories </pre>
diagram 2 of 2	<pre> classDiagram class CardEdgeOverflowCategoriesCableStylePropertyType { FlatCable RibbonCable RoundCable TwistedPair Other } class FlatCable { type : JEP30-D10:EmptyType } class RibbonCable { type : JEP30-D10:EmptyType } class RoundCable { type : JEP30-D10:EmptyType } class TwistedPair { type : JEP30-D10:EmptyType } class Other { type : xs:string } </pre>
type	CardEdgeOverflowCategoriesType , CardEdgeOverflowCategoriesProperty-ArrayType , CardEdgeOverflowCategoriesCableStylePropertyType , OverflowCategoriesType .
group	CardEdgeOverflowCategories , SearchableProperties .

4.7.1.2.2. Optical Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Optical
diagram	<pre> classDiagram class OpticalConnectorClassificationType { Sub-CategoryName : xs:string Property-Array : OpticalConnectorClassificationProperty-ArrayType } class Optical { type : OpticalConnectorClassificationType } class attributes class MetalCage { type : JEP30-D10:EmptyType } class MetalCageWithHeatsink { type : JEP30-D10:EmptyType } OpticalConnectorClassificationType "1" -- "*" attributes OpticalConnectorClassificationType "1" -- "*" MetalCage OpticalConnectorClassificationType "1" -- "*" MetalCageWithHeatsink Optical "1" -- "*" OpticalConnectorClassificationType </pre>
type	OpticalConnectorClassificationType , OpticalConnectorClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.2.3. Socket Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket
diagram	<pre> classDiagram class Socket { type: SocketConnectorClassificationType } class SocketConnectorClassificationType { <> PCB-Mountable <> Non-PCB-Mountable <> Other <> SocketOverflowCategories <> Property-Array } class PCB-Mountable { type: PCB-MountableSocketClassificationType } class Non-PCB-Mountable { type: Non-PCB-MountableSocketClassificationType } class Other { type: SocketOverflowCategoriesType } class SocketOverflowCategories { type: OverflowCategoriesType } class Parts { type: OverflowCategoriesType } class Accessories { type: OverflowCategoriesType } class Property-Array { type: SocketConnectorClassificationProperty-ArrayType } class attributes class SearchableProperties <> Socket <> SocketConnectorClassificationType <> PCB-Mountable <> Non-PCB-Mountable <> Other <> SocketOverflowCategories <> Property-Array <> attributes <> SearchableProperties <> Parts <> Accessories </pre>
type	SocketConnectorClassificationType , PCB-MountableSocketClassificationType , Non-PCB-MountableSocketClassificationType , SocketOverflowCategoriesType , OverflowCategoriesType , SocketConnectorClassificationProperty-ArrayType .
group	SocketOverflowCategories , SearchableProperties .

4.7.1.2.3.1. PCB-Mountable Socket Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket/PCB-Mountable
diagram	<pre> classDiagram class PCB-Mountable { type: PCB-MountableSocketClassificationType } class PCB-MountableSocketClassificationType { <> Property-Array <> attributes <> MountingTechnology <> SearchableProperties } class Property-Array { type: PCB-MountableSocketProperty-ArrayType } class attributes class MountingTechnology { type: SMD-TH-MountingTechnologyType } class SearchableProperties <> PCB-Mountable <> PCB-MountableSocketClassificationType <> Property-Array <> attributes <> MountingTechnology <> SearchableProperties </pre>
type	PCB-MountableSocketClassificationType , PCB-MountableSocketProperty-ArrayType , SMD-TH-MountingTechnologyType .
group	SearchableProperties .

The [MountingTechnology](#) element has enumerated values of [SMD](#) and [Thru-Hole](#).

4.7.1.2.3.2. Non-PCB-Mountable Socket Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket/Non-PCB-Mountable
diagram	<pre> classDiagram class Non-PCB-MountableSocketClassificationType { <<Non-PCB-Mountable>> <<Non-PCB-MountableSocketClassificationType>> } class Property-Array { <<Non-PCB-MountableSocketProperty-ArrayType>> } class SearchableProperties { <<SearchableProperties>> } Non-PCB-MountableSocketClassificationType "2..1" --> Property-Array : Non-PCB-MountableSocketClassificationType "2..1" --> SearchableProperties : </pre>
type	Non-PCB-MountableSocketClassificationType , Non-PCB-MountableSocketProperty-ArrayType .
group	SearchableProperties .

4.7.1.2.3.3. Socket Overflow Categories Group

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket
diagram	<pre> classDiagram class SocketOverflowCategoriesType { <<SocketOverflowCategoriesType>> <<Sub-CategoryName>> <<xsstring>> <<Property-Array>> <<SocketOverflowCategoriesProperty-ArrayType>> <<MountingTechnology>> <<SMD-TH-MountingTechnologyType>> } class Other { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } SocketOverflowCategoriesType "2..1" --> Other : SocketOverflowCategoriesType "2..1" --> Sub-CategoryName : SocketOverflowCategoriesType "2..1" --> Property-Array : SocketOverflowCategoriesType "2..1" --> MountingTechnology : </pre>
type	SocketOverflowCategoriesType , OverflowCategoriesType , SocketOverflowCategoriesProperty-ArrayType , SMD-TH-MountingTechnologyType .
group	SocketOverflowCategories , SearchableProperties .

MountingTechnology has an enumerated value of [SMD](#), or [Thru-Hole](#). While this information can be defined in the [PartModel/PackageSection](#) of the PartModel, many users find value in being able to search for the Backplane connector via this attribute.

4.7.1.2.4. Board Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board
diagram	<p>The diagram illustrates the classification of Board connectors. It starts with a general class Board (type: BoardConnectorClassificationType) which can be categorized into several specific types. These categories include Backplane, Interface, PowerAndSignal, Power, Signal, RF, and TerminalAndTerminalBlock. A connector can also fall into the Other category, which is further divided into Parts and Accessories. Additionally, a connector can belong to the BoardOverflowCategories category. The Board class is associated with a Property-Array (type: BoardConnectorClassificationProperty-ArrayType). This array contains properties such as attributes, ElectricalRating (type: ElectricalRatingType), and SearchableProperties. The ElectricalRating property is further subdivided into Shielded and Unshielded types.</p>
type	BoardConnectorClassificationType , BoardBackplaneConnectorClassificationType , BoardInterfaceConnectorClassificationType , PowerAndSignalBoardConnectorClassificationType , PowerBoardConnectorClassificationType , BoardSignalConnectorClassificationType , RF-BoardConnectorClassificationType , TerminalAndTerminalBlockBoardConnectorClassificationType , BoardOverflowCategoriesType , BoardConnectorClassificationProperty-ArrayType , ElectricalRatingType , JEP30-D10:EmptyType
group	BoardOverflowCategories , SearchableProperties .

A **Board** connector can be sub-classified into one of the above categories or can have a new category specified under the sub-category **Other**. The **Backplane** category has additional classification as shown below, whereas **Interface**, **PowerAndSignal**, **Power**, **Signal**, **RF** and **TerminalAndTerminalBlock** do not have additional sub-classifications in this release.

4.7.1.2.4.1. Electrical Rating

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/Property-Array/ElectricalRating</code>
diagram	<pre> classDiagram class ElectricalRatingType { Value } class ElectricalRating { <<ElectricalRatingType>> } class Current { <<JEP30-D10:CurrentUOMType>> } class Voltage { <<JEP30-D10:VoltageUOMType>> } class Power { <<JEP30-D10:PowerUOMType>> } ElectricalRating "3" -- "1" Current : ElectricalRating "3" -- "1" Voltage : ElectricalRating "3" -- "1" Power : </pre>
type	<code>ElectricalRatingType, JEP30-D10:CurrentUOMType, JEP30-D10:VoltageUOMType, JEP30-D10:PowerUOMType.</code>

The enumerated values of *Current*, *Voltage*, and *Power* can be seen in Table 3 — UOM Enumerated Lists below.

4.7.1.2.4.2. Backplane Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/Backplane</code>
diagram	<pre> classDiagram class Backplane { <<BoardBackplaneConnectorClassificationType>> } class BoardBackplaneConnectorClassificationType { Vertical RightAngle Orthogonal Cable Other Parts Accessories Property-Array } Backplane "3" -- "1" BoardBackplaneConnectorClassificationType : </pre>
type	<code>BoardBackplaneConnectorClassificationType, VerticalBoardBackplaneConnectorClassificationType, RightAngleBoardBackplaneConnectorClassificationType, OrthogonalBoardBackplaneConnectorClassificationType, CableBoardBackplaneConnectorClassificationType, OverflowCategoriesType, BoardBackplaneConnectorClassificationProperty-ArrayType</code>
group	<code>OverflowCategories.</code>

Board/Backplane connectors can be sub-classified into *Vertical*, *RightAngle*, *Orthogonal*, or connect to a *Cable*.

4.7.1.2.4.2.1. Board Backplane Connector Classification Property – Array Type

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/Backplane/Property-Array
diagram	<pre> classDiagram class Property-Array { <<BoardBackplaneConnectorClassificationProperty-ArrayType>> } class BoardBackplaneConnectorClassificationProperty-ArrayType { <<JEP30-D10:EmptyType>> <<xs:integer>> <<SMD-TH-PF-MountingTechnologyType>> <<JEP30-D10:EmptyType>> <<SearchableProperties>> } Property-Array "1" --> "1" BoardBackplaneConnectorClassificationProperty-ArrayType BoardBackplaneConnectorClassificationProperty-ArrayType "*" --> "1" DifferentialPair BoardBackplaneConnectorClassificationProperty-ArrayType "*" --> "1" DifferentialPairsQuantity BoardBackplaneConnectorClassificationProperty-ArrayType "*" --> "1" MountingTechnology BoardBackplaneConnectorClassificationProperty-ArrayType "*" --> "1" ConnectorRetention BoardBackplaneConnectorClassificationProperty-ArrayType "*" --> "1" SearchableProperties </pre>
type	BoardBackplaneConnectorClassificationProperty-ArrayType , JEP30-D10:EmptyType , SMD-TH-PF-MountingTechnologyType .
group	SearchableProperties .

If the *Backplane* connector contains differential pairs, then *DifferentialPair* is inserted into the PartModel file along with the *DifferentialPairsQuantity* that is supported by that connector column.

MountingTechnology has an enumerated value of *SMD*, *Thru-Hole* or *Pressfit*. While this information can be defined in the *PartModel/PackageSection* of the PartModel, many users find value in being able to search for the Backplane connector via this attribute.

If the *Backplane* connector has retention capability to its mating connector, then *ConnectorRetention* element is inserted into the PartModel file.

4.7.1.2.4.3. Interface Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/Interface
diagram	<pre> classDiagram class Interface { type BoardInterfaceConnectorClassificationType } class Property_Array { type BoardInterfaceConnectorClassificationPropertyArrayType } class CableStyle { type CableStylePropertyType } class Coaxial { type JEP30-D10:EmptyType } class FlatCable { type JEP30-D10:EmptyType } class RibbonCable { type JEP30-D10:EmptyType } class RoundCable { type JEP30-D10:EmptyType } class TwistedPair { type JEP30-D10:EmptyType } class Other { type xs:string } class Shielded { type JEP30-D10:EmptyType } class Unshielded { type JEP30-D10:EmptyType } class SearchableProperties Interface "1..1" --> "1..1" Property_Array Property_Array "1..1" --> "1..1" CableStyle CableStyle "1..1" --> "1..1" Coaxial CableStyle "1..1" --> "1..1" FlatCable CableStyle "1..1" --> "1..1" RibbonCable CableStyle "1..1" --> "1..1" RoundCable CableStyle "1..1" --> "1..1" TwistedPair CableStyle "1..1" --> "1..1" Other CableStyle "1..1" --> "1..1" Shielded CableStyle "1..1" --> "1..1" Unshielded CableStyle "1..1" --> "1..1" SearchableProperties </pre>
type	BoardInterfaceConnectorClassificationType, BoardInterfaceConnectorClassificationProperty-ArrayType, CableStylePropertyType, JEP30-D10:EmptyType.
group	SearchableProperties.

4.7.1.2.4.4. Power And Signal Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/PowerAndSignal
diagram	<pre> classDiagram class PowerAndSignal { type PowerAndSignalBoardConnectorClassificationType } class Property_Array { type PowerAndSignalBoardConnectorClassificationPropertyArrayType } class ElectricalRating { type ElectricalRatingType } class Shielded { type JEP30-D10:EmptyType } class Unshielded { type JEP30-D10:EmptyType } class SearchableProperties PowerAndSignal "1..1" --> "1..1" Property_Array Property_Array "1..1" --> "1..1" ElectricalRating ElectricalRating "1..1" --> "1..1" Shielded ElectricalRating "1..1" --> "1..1" Unshielded ElectricalRating "1..1" --> "1..1" SearchableProperties </pre>
type	PowerAndSignalBoardConnectorClassificationType, PowerAndSignalBoardConnectorClassificationProperty-ArrayType, ElectricalRatingType, JEP30-D10:EmptyType.
group	SearchableProperties.

4.7.1.2.4.5. Power And Signal Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/Power</code>
diagram	<pre> classDiagram class Power { type PowerBoardConnectorClassificationType } class Property_Array { type PowerBoardConnectorClassificationPropertyArrayType } class ElectricalRating { type ElectricalRatingType } class Shielded { type JEP30-D10:EmptyType } class Unshielded { type JEP30-D10:EmptyType } class SearchableProperties Power "3..>" Property_Array Property_Array "3..>" ElectricalRating ElectricalRating "3..>" Shielded ElectricalRating "3..>" Unshielded Shielded "3..>" SearchableProperties Unshielded "3..>" SearchableProperties </pre>
type	<code>PowerBoardConnectorClassificationType, PowerBoardConnectorClassificationProperty-ArrayType, ElectricalRatingType, JEP30-D10:EmptyType.</code>
group	<code>SearchableProperties.</code>

4.7.1.2.4.6. Signal Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/Signal</code>
diagram	<pre> classDiagram class Signal { type BoardSignalConnectorClassificationType } class Property_Array { type BoardSignalConnectorClassificationPropertyArrayType } class ElectricalRating { type ElectricalRatingType } class Shielded { type JEP30-D10:EmptyType } class Unshielded { type JEP30-D10:EmptyType } class SearchableProperties Signal "3..>" Property_Array Property_Array "3..>" ElectricalRating ElectricalRating "3..>" Shielded ElectricalRating "3..>" Unshielded Shielded "3..>" SearchableProperties Unshielded "3..>" SearchableProperties </pre>
type	<code>SignalBoardConnectorClassificationType, SignalBoardConnectorClassificationProperty-ArrayType, ElectricalRatingType, JEP30-D10:EmptyType.</code>
group	<code>SearchableProperties.</code>

4.7.1.2.4.7. Board Overflow Categories

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board/Signal
diagram	<pre> classDiagram class BoardOverflowCategoriesType { Sub-CategoryName : xs:string Property-Array : BoardOverflowCategoriesProperty-ArrayType } class Other { type : BoardOverflowCategoriesType } class Parts { type : OverflowCategoriesType } class Accessories { type : OverflowCategoriesType } class SearchableProperties { type : CableStylePropertyType } BoardOverflowCategoriesType "1" -- "1" Other BoardOverflowCategoriesType "1" -- "1" Parts BoardOverflowCategoriesType "1" -- "1" Accessories BoardOverflowCategoriesType "1" -- "1" SearchableProperties </pre>
type	BoardOverflowCategoriesType , BoardOverflowCategoriesProperty-ArrayType , ElectricalRatingType , JEP30-D10:EmptyType , CableStylePropertyType .
group	BoardOverflowCategories , SearchableProperties .

4.7.1.2.5. Cable Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable
diagram	<pre> classDiagram class Cable { type: CableConnectorClassificationType } class CableConnectorClassificationType { type: CableConnectorClassificationType "PowerAndSignal" "Power" "Signal" "Interface" "RF" "CableOverflowCategories" "Parts" "Accessories" "Property-Array" } class PowerAndSignal { type: PowerAndSignalCableConnectorClassificationType } class Power { type: PowerCableConnectorClassificationType } class Signal { type: SignalCableConnectorClassificationType } class Interface { type: CableInterfaceConnectorClassificationType } class RF { type: RF-CableConnectorClassificationType } class CableOverflowCategories { type: CableOverflowCategoriesType } class Parts { type: OverflowCategoriesType } class Accessories { type: OverflowCategoriesType } class Property-Array { type: CableConnectorClassificationProperty-ArrayType } class attributes class SearchableProperties Cable --> CableConnectorClassificationType CableConnectorClassificationType --> PowerAndSignal CableConnectorClassificationType --> Power CableConnectorClassificationType --> Signal CableConnectorClassificationType --> Interface CableConnectorClassificationType --> RF CableConnectorClassificationType --> CableOverflowCategories CableConnectorClassificationType --> Parts CableConnectorClassificationType --> Accessories CableConnectorClassificationType --> Property-Array PowerAndSignal --> PowerAndSignalCableConnectorClassificationType Power --> PowerCableConnectorClassificationType Signal --> SignalCableConnectorClassificationType Interface --> CableInterfaceConnectorClassificationType RF --> RF-CableConnectorClassificationType CableOverflowCategories --> CableOverflowCategoriesType Parts --> OverflowCategoriesType Accessories --> OverflowCategoriesType Property-Array --> attributes Property-Array --> SearchableProperties </pre>
type	CableConnectorClassificationType , PowerAndSignalCableConnectorClassificationType , PowerCableConnectorClassificationType , SignalCableConnectorClassificationType , CableInterfaceConnectorClassificationType , RF-CableConnectorClassificationType , CableOverflowCategoriesType , OverflowCategoriesType , CableConnectorClassificationProperty-ArrayType .
group	CableOverflowCategories , SearchableProperties .

If the [Backplane](#) connector contains differential pairs then [DifferentialPair](#) is inserted into the PartModel file along with the [DifferentialPairsQuantity](#) that is supported by that connector column.

[MountingTechnology](#) has an enumerated value of [SMD](#), [Thru-Hole](#) or [Pressfit](#). While this information can be defined in the [PartModel/PackageSection](#) of the PartModel, many users find value in being able to search for the Backplane connector via this attribute.

If the [Backplane](#) connector has retention capability to its mating connector, then [ConnectorRetention](#) element is inserted into the PartModel file.

4.7.1.2.5.1. Power And Signal Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable/PowerAndSignal
diagram	<pre> classDiagram class PowerAndSignal { <<PowerAndSignalCableConnectorClassificationType>> } class Property_Array { <<PowerAndSignalCableConnectorClassificationProperty-ArrayType>> } class ElectricalRating { <<ElectricalRatingType>> } class SearchableProperties class Shielded { <<JEP30-D10:EmptyType>> } class Unshielded { <<JEP30-D10:EmptyType>> } PowerAndSignal "1" -- "*" Property_Array : Property_Array "1" -- "*" attributes : ElectricalRating Property_Array "1" -- "*" SearchableProperties ElectricalRating "*" -- "*" Shielded ElectricalRating "*" -- "*" Unshielded </pre>
type	PowerAndSignalCableConnectorClassificationType , PowerAndSignalCableConnectorClassificationProperty-ArrayType , ElectricalRatingType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.2.5.2. Power Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable/Power
diagram	<pre> classDiagram class Power { <<PowerCableConnectorClassificationType>> } class Property_Array { <<PowerCableConnectorClassificationProperty-ArrayType>> } class ElectricalRating { <<ElectricalRatingType>> } class SearchableProperties class Shielded { <<JEP30-D10:EmptyType>> } class Unshielded { <<JEP30-D10:EmptyType>> } Power "1" -- "*" Property_Array : Property_Array "1" -- "*" attributes : ElectricalRating Property_Array "1" -- "*" SearchableProperties ElectricalRating "*" -- "*" Shielded ElectricalRating "*" -- "*" Unshielded </pre>
type	PowerCableConnectorClassificationType , PowerCableConnectorClassificationProperty-ArrayType , ElectricalRatingType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.2.5.3. Signal Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable/Signal
diagram	<pre> classDiagram class Signal { <<SignalCableConnectorClassificationType>> } class SignalCableConnectorClassificationType { <<SignalCableConnectorClassificationType>> } class SignalCableConnectorClassificationPropertyArrayType { <<SignalCableConnectorClassificationProperty-ArrayType>> } class attributes { <<SignalCableConnectorClassificationProperty-ArrayType>> } class Shielded { <<JEP30-D10:EmptyType>> } class Unshielded { <<JEP30-D10:EmptyType>> } class SearchableProperties { <<SearchableProperties>> } Signal "1" --> "1" SignalCableConnectorClassificationType : SignalCableConnectorClassificationType "1" --> "1" SignalCableConnectorClassificationPropertyArrayType : SignalCableConnectorClassificationPropertyArrayType "1" --> "1" attributes : attributes "1" --> "1" Shielded : attributes "1" --> "1" Unshielded : Shielded "1" --> "1" SearchableProperties : Unshielded "1" --> "1" SearchableProperties </pre>
type	SignalCableConnectorClassificationType, SignalCableConnectorClassificationProperty-ArrayType, JEP30-D10:EmptyType.
group	SearchableProperties.

4.7.1.2.5.4. Cable Overflow Categories Type

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable
diagram	<pre> classDiagram class CableOverflowCategoriesType { <<xs:string>> Sub-CategoryName <<CableOverflowCategoriesType>> } class CableOverflowCategoriesPropertyArrayType { <<CableOverflowCategoriesProperty-ArrayType>> } class CableStylePropertyType { <<CableStylePropertyType>> } class Coaxial { <<JEP30-D10:EmptyType>> } class FlatCable { <<JEP30-D10:EmptyType>> } class RibbonCable { <<JEP30-D10:EmptyType>> } class RoundCable { <<JEP30-D10:EmptyType>> } class TwistedPair { <<JEP30-D10:EmptyType>> } class Other { <<xs:string>> } class SearchableProperties { <<SearchableProperties>> } CableOverflowCategoriesType "1" --> "1" CableOverflowCategoriesPropertyArrayType : CableOverflowCategoriesPropertyArrayType "1" --> "1" CableOverflowCategoriesType : CableOverflowCategoriesType "1" --> "1" CableStylePropertyType : CableStylePropertyType "1" --> "1" CableOverflowCategoriesPropertyArrayType : CableOverflowCategoriesPropertyArrayType "1" --> "1" SearchableProperties : CableStylePropertyType "1" --> "1" Coaxial : CableStylePropertyType "1" --> "1" FlatCable : CableStylePropertyType "1" --> "1" RibbonCable : CableStylePropertyType "1" --> "1" RoundCable : CableStylePropertyType "1" --> "1" TwistedPair : CableStylePropertyType "1" --> "1" Other </pre>
type	CableOverflowCategoriesType, CableOverflowCategoriesProperty-ArrayType, CableStylePropertyType' JEP30-D10:EmptyType.
group	SearchableProperties.

4.7.1.3. Electrical Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical
diagram	<pre> classDiagram class Electrical { type: ElectricalClassificationType } class ElectricalClassificationType { Amplifier Audio Battery Capacitor CircuitProtection DataConverter Diode Filter FrequencySource ElectromechanicalDriver IC Inductor Memory Optoelectronic } class PropertyArray { type: ElectricalClassificationPropertyArrayType } class Other { type: OverflowCategoriesType } class Parts { type: OverflowCategoriesType } class Accessories { type: OverflowCategoriesType } Electrical < -- ElectricalClassificationType Electrical < -- PropertyArray Electrical < -- Other Electrical < -- Parts Electrical < -- Accessories ElectricalClassificationType < -- Amplifier ElectricalClassificationType < -- Audio ElectricalClassificationType < -- Battery ElectricalClassificationType < -- Capacitor ElectricalClassificationType < -- CircuitProtection ElectricalClassificationType < -- DataConverter ElectricalClassificationType < -- Diode ElectricalClassificationType < -- Filter ElectricalClassificationType < -- FrequencySource ElectricalClassificationType < -- ElectromechanicalDriver ElectricalClassificationType < -- IC ElectricalClassificationType < -- Inductor ElectricalClassificationType < -- Memory ElectricalClassificationType < -- Optoelectronic </pre>
type	ElectricalClassificationType , AmplifierClassificationType , AudioClassificationType , BatteryClassificationType , CapacitorClassificationType , CircuitProtectionClassificationType , DataConversionClassificationType , DiodeClassificationType , FilterClassificationType , FrequencySourceClassificationType , ElectromechanicalDriverClassificationType , IC-ClassificationType , InductorClassificationType , MemoryClassificationType , OptoelectronicsClassificationType , PowerRegulatorClassificationType , RelayClassificationType , ResistorClassificationType , RF-ClassificationType , SensorClassificationType , SwitchClassificationType , ThyristorClassificationType , TransformerClassificationType , TransistorClassificationType , TubeClassificationType , OverflowCategoriesType , ElectricalClassificationPropertyArrayType .
group	OverflowCategories .

There are several high-level *Electrical* classifications for a Part, with the capability of further sub-level classifications as shown in the following sections. A component manufacturer can also create their own classification by using the category *Other* and entering in their classification definition. However, caution should be taken in using this category, since it will reduce the effectiveness of the software tools to leverage off this data for more efficient processing of the data, since such manually defined classification may not be recognized by the software tools.

Classifications under the category *Other* are candidates for future standardization via this publication.

4.7.1.3.1. Amplifier Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Amplifier
diagram	<pre> classDiagram class AmplifierClassificationType { Audio type AudioAmplifierClassificationType Comparator type ComparatorAmplifierClassificationType Instrumentation type InstrumentationAmplifierClassificationType Isolation type IsolationAmplifierClassificationType Logarithmic type LogarithmicAmplifierClassificationType OperationalAmplifier type OperationalAmplifierClassificationType PowerAmplifier type PowerAmplifierClassificationType RF type RF-AmplifierClassificationType Specialty type SpecialtyAmplifierClassificationType VariableGain type VariableGainAmplifierClassificationType Video type VideoAmplifierClassificationType Other type OverflowCategoriesType Parts type OverflowCategoriesType Accessories type OverflowCategoriesType Property-Array type AmplifierClassificationProperty-ArrayType } class Amplifier { type AmplifierClassificationType } Amplifier < --> AmplifierClassificationType Other < --> AmplifierClassificationType Parts < --> AmplifierClassificationType Accessories < --> AmplifierClassificationType </pre> <p>The diagram illustrates the classification of amplifiers. It starts with a general class 'Amplifier' (type: AmplifierClassificationType) which is associated with 'AmplifierClassificationType'. This association is shown with a multiplicity of '***' on the 'Amplifier' side. Inside 'AmplifierClassificationType', there are ten specific categories: 'Audio', 'Comparator', 'Instrumentation', 'Isolation', 'Logarithmic', 'OperationalAmplifier', 'PowerAmplifier', 'RF', 'Specialty', and 'VariableGain', each with its own classification type. Below these, there are three overflow categories: 'Other', 'Parts', and 'Accessories', all of which also have their own classification types. A final class 'Property-Array' (type: AmplifierClassificationProperty-ArrayType) is shown at the bottom.</p>
type	AmplifierClassificationType , AudioAmplifierClassificationType , ComparatorAmplifierClassificationType , InstrumentationAmplifierClassificationType , IsolationAmplifierClassificationType , LogarithmicAmplifierClassificationType , OperationalAmplifierClassificationType , PowerAmplifierClassificationType , RF-AmplifierClassificationType , SpecialtyAmplifierClassificationType , VariableGainAmplifierClassificationType , VideoAmplifierClassificationType , OverflowCategoriesType , AmplifierClassificationProperty-ArrayType
group	OverflowCategories .

An [Amplifier](#) can be sub-classified into one of the above categories or can have a new category specified under the category [Other](#).

Classifications under the category [Other](#) are candidates for future standardization via this publication. The [Sub-CategoryName](#) enables the component manufacturer to propose specific sub-classifications via this structure.

4.7.1.3.2. Audio Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Audio
diagram	<pre> classDiagram class AudioClassificationType class MicrophoneClassificationType class Analog class Digital class Other class Parts class Accessories class OverflowCategories class SpeakerClassificationType class MicrophoneClassificationType class PropertyArray class MicrophoneClassificationPropertyArrayType class SpeakerClassificationPropertyArrayType class AudioClassificationPropertyArrayType AudioClassificationType < -- MicrophoneClassificationType MicrophoneClassificationType < -- Analog MicrophoneClassificationType < -- Digital MicrophoneClassificationType < -- Other MicrophoneClassificationType < -- Parts MicrophoneClassificationType < -- Accessories MicrophoneClassificationType < -- OverflowCategories MicrophoneClassificationType < -- SpeakerClassificationType SpeakerClassificationType < -- SpeakerClassificationPropertyArrayType SpeakerClassificationType < -- OverflowCategories SpeakerClassificationType < -- MicrophoneClassificationPropertyArrayType SpeakerClassificationType < -- PropertyArray SpeakerClassificationType < -- AudioClassificationPropertyArrayType SpeakerClassificationType < -- AudioClassificationType </pre> <p>The diagram illustrates the classification hierarchy for audio components. It starts with a general AudioClassificationType, which branches into MicrophoneClassificationType and SpeakerClassificationType. The MicrophoneClassificationType further divides into Analog, Digital, Other, Parts, and Accessories, each associated with an OverflowCategories type. The SpeakerClassificationType also has an OverflowCategories type. Additionally, there are Property-Array types associated with both MicrophoneClassificationType and SpeakerClassificationType.</p>
type	AudioClassificationType , MicrophoneClassificationType , AnalogMicrophoneClassificationType , DigitalMicrophoneClassificationType , MicrophoneClassificationPropertyArrayType , SpeakerClassificationType , OverflowCategoriesType , AudioClassificationPropertyArrayType .
group	OverflowCategories .

An **Audio** part can be sub-classified into one of the above categories or can have a new category specified under the category **Other**.

4.7.1.3.3. Battery Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery
diagram	<pre> classDiagram class Battery { type: BatteryClassificationType } class BatteryClassificationType { <<BatteryClassification>> <<Rechargeable>> <<Non-rechargeable>> <<Other>> <<Parts>> <<Accessories>> <<Property-Array>> } class Rechargeable { type: RechargeableBatteryClassificationType } class Non-rechargeable { type: Non-rechargeableBatteryClassificationType } class Other { type: MaterialOverflowCategoriesType } class Parts { type: OverflowCategoriesType } class Accessories { type: OverflowCategoriesType } class PropertyArray { type: BatteryClassificationProperty-ArrayType } </pre>
type	BatteryClassificationType , RechargeableBatteryClassificationType , Non-rechargeableBatteryClassificationType , MaterialOverflowCategoriesType , OverflowCategoriesType , BatteryClassificationProperty-ArrayType .
group	MaterialOverflowCategories .

A *Battery* is either *Rechargeable* or *Non-rechargeable*. A *Non-rechargeable* battery, otherwise known as Primary Cell (single-use or "disposable") battery is used once and discarded; the electrode materials are irreversibly changed during discharge. Common examples are the alkaline battery used for flashlights and a multitude of portable electronic devices. A *Rechargeable* battery, otherwise known as Secondary batteries, can be discharged and recharged multiple times using an applied electric current; the original composition of the electrodes can be restored by reverse current.

4.7.1.3.3.1. Rechargeable Battery Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery/Rechargeable
diagram	<pre> classDiagram class Rechargeable { type: RechargeableBatteryClassificationType } class RechargeableBatteryClassificationType { <<RechargeableBatteryClassification>> <<Property-Array>> } class RechargeableBatteryProperty-ArrayType { <<attributes>> <<Material>> <<SearchableProperties>> } class Material { type: RechargeableBatteryMaterialType } </pre>
type	RechargeableBatteryClassificationType , RechargeableBatteryProperty-ArrayType , RechargeableBatteryMaterialType .
group	SearchableProperties .

4.7.1.3.3.1.1. Rechargeable Battery Material Property

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery/Rechargeable/Property-Array/Material
diagram	<pre> classDiagram class RechargeableBatteryMaterialType { Nickel-Cadmium Lead-Acid Nickel-MetalHydride Nickel-Zinc Silver-Zinc Lithium-Iron-Phosphate LithiumIonLithiumCobaltOxide LithiumIonManganeseOxideBattery LithiumIonPolymerBattery Other } class Material { type RechargeableBatteryMaterialType } Material < -- RechargeableBatteryMaterialType </pre>
type	RechargeableBatteryMaterialType, JEP30-D10:EmptyType.

The rechargeable battery **Material** property can be set to one of the above materials or in the event of new materials, can be specified under **Other**.

4.7.1.3.3.2. Non-rechargeable Battery Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery/Non-rechargeable
diagram	<pre> classDiagram class Non-rechargeableBatteryClassificationType { <><> Property-Array <><> Material } class Non-rechargeableBatteryProperty-ArrayType { attributes <><> SearchableProperties } Non-rechargeableBatteryClassificationType < -- Non-rechargeableBatteryProperty-ArrayType Non-rechargeableBatteryProperty-ArrayType < -- Material </pre>
type	Non-rechargeableBatteryClassificationType, Non-rechargeableBatteryProperty-ArrayType, Non-rechargeableBatteryMaterialType.
group	SearchableProperties.

4.7.1.3.3.2.1. Non-rechargeable Battery Material Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery/Non-rechargeable/Property-Array/Material</code>
diagram	<pre> classDiagram class Non-rechargeableBatteryMaterialType { Zinc-Carbon Zinc-Chloride Alkaline Lithium-IronDisulfide Lithium-ManganeseDioxide Lithium-CarbonFluoride Lithium-ChromiumOxide Zinc-Air ZamboniPile SilverZinc Magnesium Other } class Material { type Non-rechargeableBatteryMaterialType } Material < -- Non-rechargeableBatteryMaterialType </pre>
type	<code>Non-rechargeableBatteryMaterialType, JEP30-D10:EmptyType.</code>

A Non-rechargeable battery *Material* property can be set to one of the above materials or in the event of new materials, can be specified under *Other*.

4.7.1.3.3.3. Material Overflow Categories Group

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery</code>
diagram	<pre> classDiagram class MaterialOverflowCategoriesType { <<Sub-CategoryName>> <<Property-Array>> <<Parts>> <<Accessories>> <<Other>> } class MaterialOverflowCategoriesProperty-ArrayType { <<attributes>> <<Material>> <<SearchableProperties>> } MaterialOverflowCategoriesType < --> Other MaterialOverflowCategoriesType < --> Sub-CategoryName MaterialOverflowCategoriesType < --> Property-Array MaterialOverflowCategoriesType < --> Parts MaterialOverflowCategoriesType < --> Accessories </pre>
type	<code>MaterialOverflowCategoriesType, MaterialOverflowCategoriesProperty-ArrayType.</code>
group	<code>MaterialOverflowCategories, SearchableProperties.</code>

4.7.1.3.4. Capacitor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor
diagram	<pre> classDiagram class CapacitorClassificationType { <<Capacitor>> } class Ceramic { <<CeramicCapacitorClassificationType>> } class Electrolytic { <<ElectrolyticCapacitorClassificationType>> } class Film { <<FilmCapacitorClassificationType>> } class Silicon { <<SiliconCapacitorClassificationType>> } class Super_cap { <<Super-capCapacitorClassificationType>> } class Variable { <<VariableCapacitorClassificationType>> } class Other { <<CapacitorOverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } CapacitorClassificationType < -- Ceramic CapacitorClassificationType < -- Electrolytic CapacitorClassificationType < -- Film CapacitorClassificationType < -- Silicon CapacitorClassificationType < -- Super_cap CapacitorClassificationType < -- Variable CapacitorClassificationType < -- Other CapacitorClassificationType < -- Parts CapacitorClassificationType < -- Accessories Ceramic < -- Capacitor Electrolytic < -- Capacitor Film < -- Capacitor Silicon < -- Capacitor Super_cap < -- Capacitor Variable < -- Capacitor Other < -- CapacitorOverflowCategories Parts < -- CapacitorOverflowCategories Accessories < -- CapacitorOverflowCategories Capacitor --> ---> CapacitorOverflowCategories CapacitorOverflowCategories --> ---> Property_Array </pre> <p>The diagram illustrates the UML class structure for capacitor classification. It starts with a general class CapacitorClassificationType which branches into specific types: Ceramic, Electrolytic, Film, Silicon, Super-cap, and Variable. Below these, there is a category Other which further divides into Parts and Accessories. A Capacitor object (type CapacitorClassificationType) is shown interacting with a CapacitorOverflowCategories object, which in turn connects to a Property-Array.</p>
type	CapacitorClassificationType , CeramicCapacitorClassificationType , ElectrolyticCapacitorClassificationType , FilmCapacitorClassificationType , SiliconCapacitorClassificationType , Super-capCapacitorClassificationType , VariableCapacitorClassificationType , CapacitorOverflowCategoriesType , OverflowCategoriesType , CapacitorClassificationProperty-ArrayType .
group	CapacitorOverflowCategories .

A **Capacitor** is either **Fixed** or **Variable**. If **Fixed**, then the capacitor can be sub-classified into either

1. **Ceramic**,
2. **Electrolytic**,
3. **Film**, or
4. **Silicon**, or
5. **Super-cap**.

Alternatively, special fixed capacitors can be specified under the category **Other**.

4.7.1.3.4.1. Ceramic Capacitor Classification and Property-Array

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Ceramic</code>
diagram	
type	<code>CeramicCapacitorClassificationType, CeramicCapacitorClassificationProperty-ArrayType,</code> <code>JEP30-D10:EmptyType, CeramicCapacitorClassType, CeramicCapacitorDielectricType,</code> <code>CeramicCapacitorMultiLayerClassificationType.</code>
group	<code>SearchableProperties.</code>

Ceramic capacitors are often consolidated in array form, and this can be additionally classified via the optional element *Array*. The enumerated list of values for *MultiLayer* are *Yes*, *No* and *Unspecified*. If this element is not specified in the PartModel file, it is assumed to be *Unspecified*.



Figure 1 - Ceramic Capacitor Array

4.7.1.3.4.1.1. Ceramic Capacitor Class Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Ceramic/Property-Array/Class</code>
diagram	
type	<code>CeramicCapacitorClassType, JEP30-D10:EmptyType.</code>

Most *Ceramic* capacitors can be sub-classified as either *Class1* or *Class2*, however some can be specified under the category *Other*.

4.7.1.3.4.1.2. Ceramic Capacitor Dielectric Property

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Ceramic/Property-Array/Dielectric
diagram	<pre> classDiagram class CeramicCapacitorDielectricType { P100 NPO N33 N75 N150 N220 N330 N470 N750 N1000 } class Dielectric { <<CeramicCapacitorDielectricType>> } CeramicCapacitorDielectricType "1..*" -- "1" Dielectric N1500 X5R X6R X7R X7S X8R Y5V Z5U Other </pre>
type	CeramicCapacitorDielectricType , JEP30-D10:EmptyType .

4.7.1.3.4.2. Electrolytic Capacitor Classification and Property-Array

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Electrolytic</code>
diagram	<pre> classDiagram class ElectrolyticCapacitorClassificationType class ElectrolyticCapacitorClassificationPropertyArrayType { <<ElectrolyticCapacitorClassificationType>> <<Property-Array>> <<attributes>> <<Array>> <<Fused>> <<Material>> <<Electrolyte>> <<SearchableProperties>> } class Electrolytic { <<ElectrolyticCapacitorClassificationType>> } class Property-Array { <<ElectrolyticCapacitorClassificationPropertyArrayType>> } class attributes { <<ElectrolyticCapacitorClassificationPropertyArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class Fused { <<JEP30-D10:EmptyType>> } class Material { <<ElectrolyticCapacitorMaterialType>> } class Electrolyte { <<ElectrolyticCapacitorElectrolyteType>> } class SearchableProperties { <<ElectrolyticCapacitorClassificationPropertyArrayType>> } Electrolytic --> Property-Array Property-Array --> attributes attributes --> Array attributes --> Fused attributes --> Material attributes --> Electrolyte attributes --> SearchableProperties </pre>
type	<code>ElectrolyticCapacitorClassificationType, ElectrolyticCapacitorClassificationProperty-ArrayType, JEP30-D10:EmptyType, ElectrolyticCapacitorMaterialType, ElectrolyticCapacitorElectrolyteType, PropertyKeyValuePairType.</code>

Some *Electrolytic* capacitors are “Fused” which can be defined by adding the *Fused* property to the property-array.



Figure 2— Fused Electrolytic Capacitor

4.7.1.3.4.2.1. Electrolytic Capacitor Material Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Electrolytic/Property-Array/Material</code>
diagram	<pre> classDiagram class ElectrolyticCapacitorMaterialType { <<Material>> <<Aluminum>> <<Niobium>> <<Tantalum>> <<Other>> } class Material { <<ElectrolyticCapacitorMaterialType>> } class Aluminum { <<JEP30-D10:EmptyType>> } class Niobium { <<JEP30-D10:EmptyType>> } class Tantalum { <<JEP30-D10:EmptyType>> } class Other { <<xsd:string>> } Material --> Aluminum Material --> Niobium Material --> Tantalum Material --> Other </pre>
type	<code>ElectrolyticCapacitorMaterialType, JEP30-D10:EmptyType.</code>

4.7.1.3.4.2.1 Electrolytic Capacitor Material Property (cont'd)

Standard *Electrolytic* capacitors are polarized components due to their asymmetrical construction. It is the generic term for typically three different capacitor family members, namely:-

1. *Aluminum* electrolytic capacitors,
2. *Niobium* electrolytic capacitors,
3. *Tantalum* electrolytic capacitors,

However, other types of *Electrolytic* capacitors can be specified under the category *Other*.

4.7.1.3.4.2.2. Electrolytic Capacitor Electrolyte Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Electrolytic/Property-Array/Electrolyte</code>
diagram	<pre> classDiagram class ElectrolyteCapacitorElectrolyteType { <<ElectrolyticCapacitorElectrolyteType>> <<JEP30-D10:EmptyType>> } class EthyleneGlycolBoraxElectrolyte { <<EthyleneGlycolBoraxElectrolyte>> <<JEP30-D10:EmptyType>> } class MnO2Electrolyte { <<MnO2Electrolyte>> <<JEP30-D10:EmptyType>> } class MultianodeMnO2Electrolyte { <<MultianodeMnO2Electrolyte>> <<JEP30-D10:EmptyType>> } class OrganicPolymer { <<OrganicPolymer>> <<JEP30-D10:EmptyType>> } class PolymerElectrolyte { <<PolymerElectrolyte>> <<JEP30-D10:EmptyType>> } class PolymerNonSolidElectrolyte { <<PolymerNonSolidElectrolyte>> <<JEP30-D10:EmptyType>> } class WaterBasedElectrolyte { <<WaterBasedElectrolyte>> <<JEP30-D10:EmptyType>> } class MultianodePolymerElectrolyte { <<MultianodePolymerElectrolyte>> <<JEP30-D10:EmptyType>> } ElectrolyteCapacitorElectrolyteType "1" -- "*" EthyleneGlycolBoraxElectrolyte ElectrolyteCapacitorElectrolyteType "1" -- "*" MnO2Electrolyte ElectrolyteCapacitorElectrolyteType "1" -- "*" MultianodeMnO2Electrolyte ElectrolyteCapacitorElectrolyteType "1" -- "*" OrganicPolymer ElectrolyteCapacitorElectrolyteType "1" -- "*" PolymerElectrolyte ElectrolyteCapacitorElectrolyteType "1" -- "*" PolymerNonSolidElectrolyte ElectrolyteCapacitorElectrolyteType "1" -- "*" WaterBasedElectrolyte ElectrolyteCapacitorElectrolyteType "1" -- "*" MultianodePolymerElectrolyte </pre>
type	<code>ElectrolyticCapacitorElectrolyteType, JEP30-D10:EmptyType.</code>

4.7.1.3.4.3. Film Capacitor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Film
diagram	<pre> classDiagram class Film { <<FilmCapacitorClassificationType>> } class Property_Array { <<FilmCapacitorProperty-ArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class Dielectric { <<FilmCapacitorDielectricType>> } Film "3..>" Property_Array Property_Array "3..>" Array Array "3..>" Dielectric </pre>
type	FilmCapacitorClassificationType , FilmCapacitorProperty-ArrayType , FilmCapacitorDielectricType .
group	SearchableProperties .

4.7.1.3.4.3.1. Film Capacitor Dielectric Property

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Film/Property-Array/Dielectric
diagram	<pre> classDiagram class Dielectric { <<FilmCapacitorDielectricType>> } class FilmCapacitorDielectricType { <<JEP30-D10:EmptyType>> Mica Paper PET PEN PP PPS PTFE Other } Dielectric "3..>" FilmCapacitorDielectricType </pre>
type	FilmCapacitorDielectricType , JEP30-D10:EmptyType .

Film capacitors are electrical capacitors with an insulating plastic film as the dielectric. They come in several types as listed above. Alternative Dielectric can be listed under the *Other* property.

4.7.1.3.4.4. Silicon Capacitor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Silicon
diagram	<pre> classDiagram class Silicon { type SiliconCapacitorClassificationType } class Property_Array { type SiliconCapacitorProperty_ArrayType } class SiliconCapacitorProperty_ArrayType { <<attributes>> class Array { type JEP30-D10:EmptyType } class Dielectric { type xs:string } <<SearchableProperties>> } Silicon "*" --> "2..3" Property_Array Property_Array "*" --> SiliconCapacitorProperty_ArrayType </pre>
type	SiliconCapacitorClassificationType , SiliconCapacitorProperty-ArrayType .
group	SearchableProperties .

4.7.1.3.4.5. Super-cap Capacitor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Super-cap
diagram	<pre> classDiagram class Super-cap { type Super-capCapacitorClassificationType } class Property_Array { type Super-capCapacitorProperty_ArrayType } class Super-capCapacitorProperty_ArrayType { <<Class>> class Super-capCapacitorClassType { type Super-capCapacitorClassType } <<SearchableProperties>> } Super-cap "*" --> "2..3" Property_Array Property_Array "*" --> Super-capCapacitorProperty_ArrayType </pre>
type	Super-capCapacitorClassificationType , Super-capCapacitorProperty-ArrayType , Super-capCapacitorClassType .
group	SearchableProperties .

4.7.1.3.4.5.1. Super-cap Capacitor Class Property

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Super-cap/Property-Array/Class
diagram	<pre> classDiagram class Class { type Super-capCapacitorClassType } class Super-capCapacitorClassType { <<Class1>> class Class1 { type JEP30-D10:EmptyType } <<Class2>> class Class2 { type JEP30-D10:EmptyType } <<Class3>> class Class3 { type JEP30-D10:EmptyType } <<Class4>> class Class4 { type JEP30-D10:EmptyType } } Class "*" --> "2..3" Super-capCapacitorClassType </pre>
type	Super-capCapacitorClassificationType .

A [Super-cap](#) is a high-capacity capacitor with capacitance values much higher than other capacitors (but lower voltage limits) that bridge the gap between electrolytic capacitors and rechargeable batteries. They typically store 10 to 100 times more energy per unit volume or mass than electrolytic capacitors, can accept and deliver charge much faster than batteries, and tolerate many more charge and discharge cycles than rechargeable batteries. There are 4 classes of [Super-cap](#) as shown above.

4.7.1.3.4.6. Variable Capacitor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Variable
diagram	<pre> classDiagram class Variable { type VariableCapacitorClassificationType } class Property_Array { type VariableCapacitorProperty-ArrayType } class VariableCapacitorProperty-ArrayType { <<attributes>> <<Material>> <<Dielectric>> <<Electrolyte>> <<SearchableProperties>> } Variable "3" --> Property_Array Property_Array "3" --> VariableCapacitorProperty-ArrayType </pre>
type	VariableCapacitorClassificationType , VariableCapacitorProperty-ArrayType , PropertyValuePairType .

4.7.1.3.4.7. Capacitor Overflow Categories Group

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor
diagram	<pre> classDiagram class CapacitorOverflowCategoriesType class Other { type CapacitorOverflowCategoriesType } class CapacitorOverflowCategoriesProperty-ArrayType { <<attributes>> <<Array>> <<Class>> <<Material>> <<Dielectric>> <<Electrolyte>> <<SearchableProperties>> } Other "3" --> CapacitorOverflowCategoriesProperty-ArrayType CapacitorOverflowCategoriesProperty-ArrayType "3" --> CapacitorOverflowCategoriesProperty-ArrayType </pre> <p style="text-align: center;">(dashed boxes)</p> <p>Parts Accessories</p>
type	CapacitorOverflowCategoriesType , CapacitorOverflowCategoriesProperty-ArrayType , JEP30-D10:EmptyType , OverflowCategoriesType , PropertyValuePairType .
group	CapacitorOverflowCategories .

4.7.1.3.5. Circuit Protection Classification

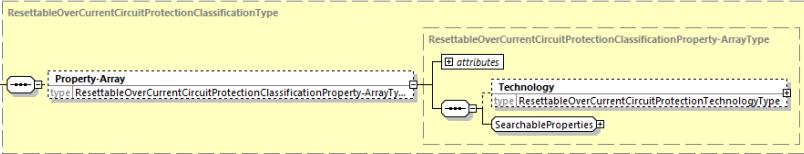
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Circuit-Protection
diagram	<pre> classDiagram class CircuitProtection { type CircuitProtectionClassificationType } class CircuitProtectionClassificationType { <<CircuitProtection>> <<OverCurrentProtection>> <<OverVoltageProtection>> <<OverTemperatureProtection>> <<Property-Array>> <<OverflowCategories>> <<Other>> <<Parts>> <<Accessories>> } <<OverflowCategories>> --> Other <<OverflowCategories>> --> Parts <<OverflowCategories>> --> Accessories </pre>
type	CircuitProtectionClassificationType , OverCurrentCircuitProtectionClassificationType , OverVoltageCircuitProtectionClassificationType , OverTemperatureCircuitProtectionClassificationType , OverflowCategoriesType , CircuitProtectionClassificationProperty-ArrayType
group	OverflowCategories .

CircuitProtection devices are used to protect the circuit's wires and components from circuit overload. An overloaded circuit occurs when there's too much current flowing through the circuit, i.e. *OverCurrentProtection*. It can damage components and wiring that are sensitive to high current. Other *Circuit-Protection* devices can also be classified under the category *OverVoltageProtection*, *OverTemperatureProtection*, or *Other*.

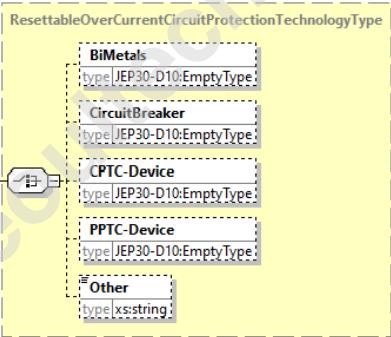
4.7.1.3.5.1. Over Current Circuit Protection Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection
diagram	<pre> classDiagram class OverCurrentProtection { type OverCurrentCircuitProtectionClassificationType } class OverCurrentCircuitProtectionClassificationType { <<Resettable>> <<Non-Resettable>> <<Property-Array>> <<OverflowCategories>> <<Other>> <<Parts>> <<Accessories>> } <<OverflowCategories>> --> Other <<OverflowCategories>> --> Parts <<OverflowCategories>> --> Accessories </pre>
type	OverCurrentCircuitProtectionClassificationType , ResettableOverCurrentCircuitProtectionClassificationType , Non-ResettableOverCurrentCircuitProtectionClassificationType , OverflowCategoriesType , OverCurrentCircuitProtectionClassificationProperty-ArrayType
group	OverflowCategories .

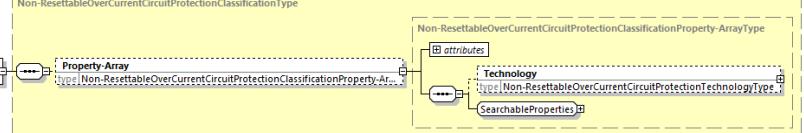
4.7.1.3.5.1.1. Resettable Over Current Circuit Protection Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection/Resettable</code>
diagram	
type	<code>ResettableOverCurrentCircuitProtectionClassificationType,</code> <code>ResettableOverCurrentCircuitProtectionClassificationProperty-ArrayType,</code> <code>ResettableOverCurrentCircuitProtectionTechnologyType.</code>
group	<code>SearchableProperties.</code>

4.7.1.3.5.1.2. Resettable Over Current Circuit Protection Technology Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection/Resettable/Property-Array/Technology</code>
diagram	
type	<code>ResettableOverCurrentCircuitProtectionTechnologyType, JEP30-D10:EmptyType</code>

4.7.1.3.5.1.3. Non-Resettable Over Current Circuit Protection Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection/Non-Resettable</code>
diagram	
type	<code>Non-ResettableOverCurrentCircuitProtectionClassificationType, Non-</code> <code>ResettableOverCurrentCircuitProtectionClassificationProperty-ArrayType,</code> <code>Non-ResettableOverCurrentCircuitProtectionTechnologyType.</code>
group	<code>SearchableProperties.</code>

4.7.1.3.5.1.4. Non-Resettable Over Current Circuit Protection Technology Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection/Non-Resettable/Property-Array/Technology</code>
diagram	<pre> classDiagram class Technology { <<Non-ResettableOverCurrentCircuitProtectionTechnologyType>> } class Fuse { <<JEP30-D10:EmptyType>> } class SolidStateFuse { <<JEP30-D10:EmptyType>> } class Other { <<xs:string>> } Technology "3" --> > Fuse Technology "3" --> > SolidStateFuse Technology "3" --> > Other </pre>
type	<code>Non-ResettableOverCurrentCircuitProtectionTechnologyType, JEP30-D10:EmptyType.</code>

4.7.1.3.5.2. Over Voltage Circuit Protection Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverVoltageProtection</code>
diagram	<pre> classDiagram class OverVoltageProtection { <<OverVoltageCircuitProtectionClassificationType>> } class OverflowCategories { <<OverflowCategoriesType>> } class Varistor { <<VaristorClassificationType>> } class AvalancheBreakdownDiode { <<AvalancheBreakdownDiodeClassificationType>> } class TransientVoltageSuppressor { <<TransientVoltageSuppressorClassificationType>> } class ProtectiveGasDischargeTube { <<ProtectiveGasDischargeTubeClassificationType>> } class SparkGap { <<SparkGapCircuitProtectionClassificationType>> } OverVoltageProtection "3" --> > OverflowCategories OverflowCategories "3" --> > Other OverflowCategories "3" --> > Parts OverflowCategories "3" --> > Accessories OverflowCategories "3" --> > PropertyArray </pre>
type	<code>OverVoltageCircuitProtectionClassificationType, VaristorClassificationType, AvalancheBreakdownDiodeClassificationType, TransientVoltageSuppressorClassificationType, ProtectiveGasDischargeTubeClassificationType, SparkGapCircuitProtectionClassificationType, OverflowCategoriesType, OverVoltageCircuitProtectionClassificationProperty-ArrayType</code>
group	<code>OverflowCategories.</code>

4.7.1.3.5.2.1. Over Voltage Circuit Protection Varistor Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverVoltageProtection/Varistor</code>
diagram	<pre> classDiagram class Varistor { <<VaristorClassificationType>> } class VaristorClassificationProperty { <<VaristorClassificationProperty-ArrayType>> } class VaristorType { <<OverVoltageCircuitProtectionVaristorType>> } class SearchableProperties Varistor "1" -- "1" VaristorClassificationProperty : VaristorClassificationProperty "*" -- "*" VaristorType : attributes VaristorType "*" -- "*" SearchableProperties : </pre>
type	<code>VaristorClassificationType, VaristorClassificationProperty-ArrayType,</code> <code>OverVoltageCircuitProtectionVaristorType.</code>
group	<code>SearchableProperties.</code>

4.7.1.3.5.2.2. Over Voltage Circuit Protection Varistor Type Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverVoltageProtection/Varistor/Property-Array/VaristorType</code>
diagram	<pre> classDiagram class VaristorType { <<OverVoltageCircuitProtectionVaristorType>> } class OverVoltageCircuitProtectionVaristorType { <<MetalOxide>> <<Multi-Layer>> <<GMOV>> <<IsoMOV>> } VaristorType "*" -- "*" OverVoltageCircuitProtectionVaristorType </pre>
type	<code>OverVoltageCircuitProtectionVaristorType, JEP30-D10:EmptyType.</code>

4.7.1.3.5.3. Over Temperature Circuit Protection Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/ OverTemperatureProtection
diagram	<pre> classDiagram class OverTemperatureProtection { type OverTemperatureCircuitProtectionClassificationType } class ThermalCutoffs { type ThermalCutoffsCircuitProtectionClassificationType } class OverflowCategories { type OverflowCategoriesType } class PropertyArray { type OverTemperatureProtectionClassificationPropertyArrayType } OverTemperatureProtection "1" -- "1" ThermalCutoffs : OverTemperatureProtection "1" -- "1" OverflowCategories : OverTemperatureProtection "1" -- "1" PropertyArray : </pre> <p>The diagram illustrates the UML class structure for Over Temperature Circuit Protection Classification. It features a central class, <code>OverTemperatureCircuitProtectionClassificationType</code>, which is associated with three other classes: <code>ThermalCutoffs</code>, <code>OverflowCategories</code>, and <code>Property-Array</code>. The <code>ThermalCutoffs</code> association is marked with a multiplicity of "1" on both ends. The <code>OverflowCategories</code> and <code>Property-Array</code> associations are also marked with "1" on both ends.</p>
type	OverTemperatureCircuitProtectionClassificationType , ThermalCutoffsCircuitProtectionClassificationType , OverflowCategoriesType , OverTemperatureProtectionClassificationPropertyArrayType .
group	OverflowCategories .

4.7.1.3.6. Data Converter Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/DataConverter
diagram	<pre> classDiagram class DataConversionClassificationType { AnalogToDigital DigitalToAnalog Current-Voltage DigitalPotentiometer Frequency-Voltage Mechanical-Signal DataConverter OverflowCategories Property-Array } class AnalogToDigital { type AnalogToDigitalConverterClassificationType } class DigitalToAnalog { type DigitalToAnalogConverterClassificationType } class CurrentVoltage { type CurrentVoltageConverterClassificationType } class DigitalPotentiometer { type DigitalPotentiometerConverterClassificationType } class FrequencyVoltage { type FrequencyVoltageConverterClassificationType } class MechanicalSignal { type MechanicalSignalConverterClassificationType } class DataConverter { type DataConversionClassificationType } class OverflowCategories { type OverflowCategoriesType } class PropertyArray { type DataConversionClassificationPropertyArrayType } DataConversionClassificationType < -- AnalogToDigital DataConversionClassificationType < -- DigitalToAnalog DataConversionClassificationType < -- CurrentVoltage DataConversionClassificationType < -- DigitalPotentiometer DataConversionClassificationType < -- FrequencyVoltage DataConversionClassificationType < -- MechanicalSignal DataConversionClassificationType < -- DataConverter DataConversionClassificationType < -- OverflowCategories DataConversionClassificationType < -- PropertyArray DataConversionClassificationType "1" -- "1" DataConverter DataConversionClassificationType "1" -- "1" OverflowCategories DataConversionClassificationType "1" -- "1" PropertyArray </pre> <p>The diagram illustrates the classification of Data Converters. It shows a main class, DataConversionClassificationType, which contains several subclasses: AnalogToDigital, DigitalToAnalog, Current-Voltage, DigitalPotentiometer, Frequency-Voltage, and Mechanical-Signal. There is also a reference to a class named DataConverter and two properties: OverflowCategories and Property-Array. The DataConversionClassificationType class has three associations: one to DataConverter, one to OverflowCategories, and one to Property-Array.</p>
type	Data-ConversionClassificationType , AnalogToDigitalConverterClassificationType , DigitalToAnalogConverterClassificationType , Current-VoltageConverterClassificationType , DigitalPotentiometerConverterClassificationType , Frequency-VoltageConverterClassificationType , Mechanical-SignalConverterClassificationType , OverflowCategories Type , DataConversionClassificationProperty-ArrayType .
group	OverflowCategories .

A *Data-Converter* is a device that converts one signal type into another signal type, as identified above, but the less common types can be classified under the category *Other*.

4.7.1.3.7. Diode Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode</code>
diagram	<pre> classDiagram class DiodeClassificationType { <<DiodeClassificationType>> } class BridgeRectifier { <<BridgeRectifierClassificationType>> } class CurrentRegulator { <<CurrentRegulatorClassificationType>> } class ESD { <<ESD-DiodeClassificationType>> } class LED { <<LED-DiodeClassificationType>> } class Microwave { <<MicrowaveDiodeClassificationType>> } class PIN { <<PIN-DiodeClassificationType>> } class Rectifier { <<RectifierDiodeClassificationType>> } class Schottky { <<SchottkyDiodeClassificationType>> } class Signal { <<SignalDiodeClassificationType>> } class SiliconCarbide { <<SiliconCarbideDiodeClassificationType>> } class Tunnel { <<TunnelDiodeClassificationType>> } class Uni_tunnel { <<Uni-tunnelDiodeClassificationType>> } class Varactor { <<VaractorDiodeClassificationType>> } class VoltageRegulator { <<VoltageRegulatorClassificationType>> } class ZenerDiode { <<ZenerDiodeClassificationType>> } class Other { <<ArrayOverflowCategoriesType>> } class ArrayOverflowCategories { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } DiodeClassificationType < -- BridgeRectifier DiodeClassificationType < -- CurrentRegulator DiodeClassificationType < -- ESD DiodeClassificationType < -- LED DiodeClassificationType < -- Microwave DiodeClassificationType < -- PIN DiodeClassificationType < -- Rectifier DiodeClassificationType < -- Schottky DiodeClassificationType < -- Signal DiodeClassificationType < -- SiliconCarbide DiodeClassificationType < -- Tunnel DiodeClassificationType < -- Uni_tunnel DiodeClassificationType < -- Varactor DiodeClassificationType < -- VoltageRegulator DiodeClassificationType < -- ZenerDiode DiodeClassificationType --> ArrayOverflowCategories : <<OverflowCategoriesType>> ArrayOverflowCategories --> Parts : <<OverflowCategoriesType>> ArrayOverflowCategories --> Accessories : <<OverflowCategoriesType>> Property-Array < -- DiodeClassificationType </pre>
type	<code>DiodeClassificationType, BridgeRectifierClassificationType, CurrentRegulatorClassificationType, ESD-DiodeClassificationType, LED-DiodeClassificationType, MicrowaveDiodeClassificationType, PIN-DiodeClassificationType, RectifierDiodeClassificationType, SchottkyDiodeClassificationType, SignalDiodeClassificationType, SiliconCarbideDiodeClassificationType, TunnelDiodeClassificationType, Uni-tunnelDiodeClassificationType, VaractorDiodeClassificationType, VoltageRegulatorClassificationType, ZenerDiodeClassificationType, ArrayOverflowCategoriesType, OverflowCategoriesType, DiodeClassificationProperty-ArrayType.</code>
group	<code>OverflowCategories.</code>

4.7.1.3.7.1. ESD Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/ESD
diagram	<pre> classDiagram class ESD-DiodeClassificationType class Property-Array { <<ESD-DiodeClassificationProperty-ArrayType>> } class attributes { <<JEP30-D10:EmptyType>> } class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties ESD-DiodeClassificationType "3" --> Property-Array Property-Array "3" --> attributes Property-Array "3" --> Array Array --> SearchableProperties </pre>
type	ESD-DiodeClassificationType, ESD-DiodeClassificationProperty-ArrayType, JEP30-D10:EmptyType.
group	SearchableProperties.

4.7.1.3.7.2. LED Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/LED
diagram	<pre> classDiagram class LED-DiodeClassificationType class Property-Array { <<LED-DiodeClassificationProperty-ArrayType>> } class attributes { <<JEP30-D10:EmptyType>> } class Emitter { <<LED-DiodeClassificationPropertyEmitterType>> } class Material { <<LED-DiodeClassificationPropertyMaterialType>> } class Color-Phosphor-Converted class Infrared class UV class Other class AlGaAs class GaAs class GaAsP class GaN-SiC class GaP class Other LED-DiodeClassificationType "3" --> Property-Array Property-Array "3" --> attributes Property-Array "3" --> Emitter Property-Array "3" --> Material Emitter --> Color-Phosphor-Converted Emitter --> Infrared Emitter --> UV Emitter --> Other Material --> AlGaAs Material --> GaAs Material --> GaAsP Material --> GaN-SiC Material --> GaP Material --> Other </pre>
type	LED-DiodeClassificationType, LED-DiodeClassificationProperty-ArrayType, JEP30-D10:EmptyType, LED-DiodeClassificationPropertyEmitterType, LED-DiodeClassificationPropertyMaterialType.
group	SearchableProperties.

4.7.1.3.7.3. Rectifier Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/Rectifier
diagram	<pre> classDiagram class Rectifier { <<RectifierDiodeClassificationType>> } class RectifierDiodeClassificationType { <<RectifierDiodeClassificationProperty-ArrayType>> } class RectifierDiodeClassificationProperty-ArrayType { <<attributes>> class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties } Rectifier "3" -- "3" RectifierDiodeClassificationType RectifierDiodeClassificationType "3" -- "3" RectifierDiodeClassificationProperty-ArrayType RectifierDiodeClassificationProperty-ArrayType "*" -- "*" Array RectifierDiodeClassificationProperty-ArrayType "*" -- "*" SearchableProperties </pre>
type	RectifierDiodeClassificationType , RectifierDiodeClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.7.4. Schottky Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/Schottky
diagram	<pre> classDiagram class Schottky { <<SchottkyDiodeClassificationType>> } class SchottkyDiodeClassificationType { <<SchottkyDiodeClassificationProperty-ArrayType>> } class SchottkyDiodeClassificationProperty-ArrayType { <<attributes>> class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties } Schottky "3" -- "3" SchottkyDiodeClassificationType SchottkyDiodeClassificationType "3" -- "3" SchottkyDiodeClassificationProperty-ArrayType SchottkyDiodeClassificationProperty-ArrayType "*" -- "*" Array SchottkyDiodeClassificationProperty-ArrayType "*" -- "*" SearchableProperties </pre>
type	SchottkyDiodeClassificationType , SchottkyDiodeClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.7.5. Signal Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/Signal
diagram	<pre> classDiagram class Signal { <<SignalDiodeClassificationType>> } class SignalDiodeClassificationType { <<SignalDiodeClassificationProperty-ArrayType>> } class SignalDiodeClassificationProperty-ArrayType { <<attributes>> class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties } Signal "3" -- "3" SignalDiodeClassificationType SignalDiodeClassificationType "3" -- "3" SignalDiodeClassificationProperty-ArrayType SignalDiodeClassificationProperty-ArrayType "*" -- "*" Array SignalDiodeClassificationProperty-ArrayType "*" -- "*" SearchableProperties </pre>
type	SignalDiodeClassificationType , SignalDiodeClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.7.6. Silicon Carbide Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/SiliconCarbide
diagram	<pre> classDiagram class SiliconCarbideDiodeClassificationType class SiliconCarbide { <<SiliconCarbideDiodeClassificationType>> } class Property-Array { <<SiliconCarbideDiodeClassificationProperty-ArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties SiliconCarbide "1" -- "*" SiliconCarbideDiodeClassificationType SiliconCarbideDiodeClassificationType "1" -- "*" Property-Array Property-Array "*" -- "*" attributes attributes "*" -- "*" Array attributes "*" -- "*" SearchableProperties </pre>
type	ArrayOverflowCategoriesType , ArrayOverflowCategoriesProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.7.7. Array Overflow Categories Group

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode
diagram	<pre> classDiagram class ArrayOverflowCategoriesType class Other { <<ArrayOverflowCategoriesType>> } class Property-Array { <<ArrayOverflowCategoriesProperty-ArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties Other "1" -- "*" ArrayOverflowCategoriesType ArrayOverflowCategoriesType "1" -- "*" Property-Array Property-Array "*" -- "*" attributes attributes "*" -- "*" Array attributes "*" -- "*" SearchableProperties class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } class Other { <<OverflowCategoriesType>> } </pre>
type	ArrayOverflowCategoriesType , ArrayOverflowCategoriesProperty-ArrayType , JEP30-D10:EmptyType , OverflowCategoriesType .
group	SearchableProperties .

4.7.1.3.8. Filter Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Filter</code>
diagram	<pre> classDiagram class FilterClassificationType class Ceramic { <<CeramicFilterClassificationType>> } class EMI_RFI { <<EMI-RFI-FilterClassificationType>> } class PhaseShift { <<PhaseShiftFilterClassificationType>> } class SAW { <<SAW-FilterClassificationType>> } class SwitchedCapacitor { <<SwitchedCapacitorFilterClassificationType>> } class Other { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } class OverflowCategories class Property_Array { <<FilterClassificationProperty-ArrayType>> } FilterClassificationType < -- Ceramic FilterClassificationType < -- EMI_RFI FilterClassificationType < -- PhaseShift FilterClassificationType < -- SAW FilterClassificationType < -- SwitchedCapacitor FilterClassificationType < -- Other FilterClassificationType < -- Parts FilterClassificationType < -- Accessories FilterClassificationType < -- OverflowCategories FilterClassificationType < -- Property_Array </pre>
type	<code>FilterClassificationType, CeramicFilterClassificationType, EMI-RFI-FilterClassificationType, PhaseShiftFilterClassificationType, SAW-FilterClassificationType, SwitchedCapacitorFilterClassificationType, OverflowCategoriesType, FilterClassificationProperty-ArrayType.</code>
group	<code>OverflowCategories.</code>

In signal processing, a *Filter* is a device or process that removes some unwanted components or features from a signal. Filtering is a class of signal processing. The defining feature of filters being the complete or partial suppression of some aspect of the signal. Most often, this means removing some frequencies or frequency bands. However, filters do not exclusively act in the frequency domain; especially in the field of image processing where other filtering targets exist.

There are many different bases of classifying filters, as shown above and additional types can be classified under the category *Other*.

4.7.1.3.8.1. EMI-RFI Filter Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Filter/EMI-RFI
diagram	<pre> classDiagram class EMI_RFI { <<EMI-RFI>> type EMI-RFI-FilterClassificationType } class FerriteBead { <<FerriteBead>> type FerriteBeadEMI-RFI-FilterClassificationType } class CommonModeChoke { <<CommonModeChoke>> type CommonModeChokeEMI-RFI-FilterClassificationType } class FeedThruCapacitor { <<FeedThruCapacitor>> type FeedThruCapacitorEMI-RFI-FilterClassificationType } class LC_type { <<LC-type>> type LC-typeEMI-RFI-FilterClassificationType } class CL_type { <<CL-type>> type CL-typeEMI-RFI-FilterClassificationType } class Pi_Type { <<Pi-Type>> type Pi-TypeEMI-RFI-FilterClassificationType } class T_Type { <<T-Type>> type T-TypeEMI-RFI-FilterClassificationType } class Other { <<Other>> type OverflowCategoriesType } class Parts { <<Parts>> type OverflowCategoriesType } class Accessories { <<Accessories>> type OverflowCategoriesType } class OverflowCategories { <<OverflowCategories>> type OverflowCategoriesType } class Property_Array { <<Property-Array>> type EMI-RFI-FilterClassificationProperty-ArrayType } EMI_RFI < -- FerriteBead EMI_RFI < -- CommonModeChoke EMI_RFI < -- FeedThruCapacitor EMI_RFI < -- LC_type EMI_RFI < -- CL_type EMI_RFI < -- Pi_Type EMI_RFI < -- T_Type EMI_RFI < -- Other EMI_RFI < -- Parts EMI_RFI < -- Accessories EMI_RFI < -- OverflowCategories EMI_RFI < -- Property_Array </pre> <p>The diagram illustrates the classification of EMI-RFI filters. It starts with a general class 'EMI-RFI' which branches into various specific filter types: FerriteBead, CommonModeChoke, FeedThruCapacitor, LC-type, CL-type, Pi-Type, and T-Type. These specific types are grouped under a general category 'OverflowCategories'. Additionally, there are three specific categories: 'Other', 'Parts', and 'Accessories', which are also grouped under 'OverflowCategories'. Finally, there is a specific property array 'Property-Array' associated with the 'EMI-RFI' class.</p>
type	EMI-RFI-FilterClassificationType , FerriteBeadEMI-RFI-FilterClassificationType , CommonModeChokeEMI-RFI-FilterClassificationType , FeedThruCapacitorEMI-RFI-FilterClassificationType , LC-typeEMI-RFI-FilterClassificationType , CL-typeEMI-RFI-FilterClassificationType , Pi-TypeEMI-RFI-FilterClassificationType , T-TypeEMI-RFI-FilterClassificationType , OverflowCategoriesType , EMI-RFI-FilterClassificationProperty-ArrayType .
group	OverflowCategories .

4.7.1.3.8.2. Filter Classification Property - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Filter/Other
diagram	<pre> classDiagram class FilterClassificationProperty-ArrayType { <<attributes>> } class Property-Array { <<type FilterClassificationProperty-ArrayType>> } class TransferFunction { <<type TransferFunctionType>> } class TransferFunctionType { LowPass BandPass BandStop HighPass } class SearchableProperties </pre>
type	FilterClassificationProperty-ArrayType , TransferFunctionType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.9. Frequency Source Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/FrequencySource
diagram	<pre> classDiagram class FrequencySourceClassificationType { Generator Timer Crystal Oscillator Resonator VoltageControlledOscillator FrequencySource OverflowCategories Property_Array } class Generator { type GeneratorClassificationType } class Timer { type TimerClassificationType } class Crystal { type CrystalClassificationType } class Oscillator { type OscillatorClassificationType } class Resonator { type ResonatorClassificationType } class VoltageControlledOscillator { type VoltageControlledOscillatorClassificationType } class FrequencySource { type FrequencySourceClassificationType } class OverflowCategories { type OverflowCategoriesType } class Property_Array { type FrequencySourceClassificationPropertyArrayType } FrequencySourceClassificationType < -- Generator FrequencySourceClassificationType < -- Timer FrequencySourceClassificationType < -- Crystal FrequencySourceClassificationType < -- Oscillator FrequencySourceClassificationType < -- Resonator FrequencySourceClassificationType < -- VoltageControlledOscillator FrequencySourceClassificationType < -- FrequencySource FrequencySourceClassificationType < -- OverflowCategories FrequencySourceClassificationType < -- Property_Array Generator < -- GeneratorClassificationType Timer < -- TimerClassificationType Crystal < -- CrystalClassificationType Oscillator < -- OscillatorClassificationType Resonator < -- ResonatorClassificationType VoltageControlledOscillator < -- VoltageControlledOscillatorClassificationType FrequencySource < -- FrequencySourceClassificationType OverflowCategories < -- OverflowCategoriesType Property_Array < -- FrequencySourceClassificationPropertyArrayType </pre> <p>The diagram illustrates the classification of Frequency Sources. It starts with a general class, FrequencySourceClassificationType, which branches into specific types: Generator, Timer, Crystal, Oscillator, Resonator, and VoltageControlledOscillator. Each of these specific types has its own classification type (GeneratorClassificationType, TimerClassificationType, etc.). Below this main classification, there is a separate class, FrequencySource, which also has its own classification type (FrequencySourceClassificationType). Additionally, there is a class, OverflowCategories, which contains three sub-categories: Other, Parts, and Accessories, each with its own classification type (OverflowCategoriesType).</p>
type	FrequencySourceClassificationType , GeneratorClassificationType , TimerClassificationType , CrystalClassificationType , OscillatorClassificationType , ResonatorClassificationType , VoltageControlledOscillatorClassificationType , OverflowCategoriesType , FilterClassificationPropertyArrayType
group	OverflowCategories .

A *FrequencySource* is an electronic device that generates repeating or non-repeating electronic signals in either the analog or the digital domain. There are many kinds of *FrequencySource* types with different purposes and applications, as identified above, and additional types can be classified under the category *Other*.

4.7.1.3.10. IC Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/IC
diagram	<p>The diagram illustrates the classification of Integrated Circuits (ICs) within a Part Model. It features a main class, IC, which is associated with a collection of overflow categories. This association is mediated by a class named OverflowCategories, which contains three sub-classes: Other, Parts, and Accessories. The IC class also has a relationship to a PropertyArray.</p> <pre> classDiagram class IC { <<IC>> <<IC-ClassificationType>> } class OverflowCategories { <<OverflowCategoriesType>> } class Other { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } class PropertyArray { <<IC-ClassificationProperty-ArrayType>> } IC "1..1" -- "*" OverflowCategories OverflowCategories "*" -- "1..1" Other OverflowCategories "*" -- "1..1" Parts OverflowCategories "*" -- "1..1" Accessories IC --> PropertyArray </pre>
type	IC-ClassificationType, CPLD-IC-ClassificationType, DataAcquisitionIC-ClassificationType, DigitalSignalProcessingIC-ClassificationType, FPGA-IC-ClassificationType, DriversAndInterfaceIC-ClassificationType, LogicIC-ClassificationType, MCM-IC-ClassificationType, MicrocontrollerAndProcessorsIC-ClassificationType, PhotoIC-ClassificationType, PowerManagementIC-ClassificationType, TelecommunicationIC-ClassificationType, TimingIC-ClassificationType, OverflowCategoriesType, IC-ClassificationProperty-ArrayType.
group	OverflowCategories.

4.7.1.3.10 IC Classification (cont'd)

An integrated circuit or monolithic integrated circuit (also referred to as an *IC*, a chip, or a microchip) is a set of electronic circuits on one small plate of semiconductor material, normally silicon. This can be made much smaller than a discrete circuit made from independent electronic components. The above categories are high level categories, and other categories can be captured under the category *Other*.

4.7.1.3.11 Inductor Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor</code>
diagram	<pre> classDiagram class Inductor { type InductorClassificationType } class InductorClassificationType { <<Ferrite>> <<MetalAlloys>> <<Air>> <<OverflowCategories>> <<Property-Array>> } class OverflowCategories { <<Other>> <<Parts>> <<Accessories>> } <<Inductor>> --> <<InductorClassificationType>> <<InductorClassificationType>> --> <<Ferrite>> <<InductorClassificationType>> --> <<MetalAlloys>> <<InductorClassificationType>> --> <<Air>> <<InductorClassificationType>> --> <<OverflowCategories>> <<OverflowCategories>> --> <<Other>> <<OverflowCategories>> --> <<Parts>> <<OverflowCategories>> --> <<Accessories>> </pre>
type	<code>InductorClassificationType, FerriteInductorClassificationType, MetalAlloysInductorClassificationType, AirInductorClassificationType, OverflowCategoriesType, InductorClassificationProperty-ArrayType</code>
group	<code>OverflowCategories</code> .

An *Inductor*, also called a coil or reactor, is a passive electrical component that stores electrical energy in a magnetic field when electric current is flowing through it. An inductor typically consists of an electric conductor, such as a wire, that is wound into a coil around a core.

When the current flowing through an inductor changes, the time-varying magnetic field induces a voltage in the conductor, described by Faraday's law of induction. According to Lenz's law, the direction of induced electromotive force (e.m.f.) opposes the change in current that created it. As a result, inductors oppose any changes in current through them.

4.7.1.3.11.1. Ferrite Inductor Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Ferrite</code>
diagram	<pre> classDiagram class FerriteInductorClassificationType { Fixed Variable <> OverflowCategories } class Fixed { <> type FixedFerriteInductorClassificationType } class Variable { <> type VariableFerriteInductorClassificationType } class OverflowCategories { Other Parts Accessories <> type OverflowCategoriesType } class Property_Array { <> type FerriteInductorClassificationProperty-ArrayType } </pre>
type	<code>FerriteInductorClassificationType</code> , <code>FixedFerriteInductorClassificationType</code> , <code>VariableFerriteInductorClassificationType</code> , <code>OverflowCategoriesType</code> , <code>FerriteInductorClassificationProperty-ArrayType</code>
group	<code>OverflowCategories</code> .

A **Variable** Inductor is a passive inductor wherein the inductor device includes a movable element which may be adjusted to different positions or adjusted to vary its physical dimensions to change the effective inductance from one value to another.

4.7.1.3.11.1.1. Variable Ferrite Inductor Property-Array

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Ferrite/Variable/Property-Array</code>
diagram	<pre> classDiagram class VariableFerriteInductorClassificationType { <> Property-Array <> type VariableFerriteInductorClassificationProperty-ArrayType } class VariableFerriteInductorClassificationProperty-ArrayType { attributes <> SearchableProperties } class Slug-Tuned { <> type JEP30-D10:EmptyType } class Tapped { <> type JEP30-D10:EmptyType } class Other { <> type xs:string } </pre>
type	<code>VariableFerriteInductorClassificationType</code> , <code>VariableFerriteInductorClassificationProperty-ArrayType</code> , <code>VariableInductorAdjustmentType</code> , <code>JEP30-D10:EmptyType</code> .
group	<code>SearchableProperties</code> .

Examples are **Adjustment** types on a **Variable** inductor are **Slug-tuned** or **Tapped**, but other categories can be captured under the category **Other**.

4.7.1.3.11.1.2. Ferrite Inductor Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Ferrite/Property-Array
diagram	<pre> classDiagram class FerriteInductorClassificationPropertyArrayType { <<attributes>> } class PropertyArrayType { <<type>> FerriteInductorClassificationPropertyArrayType } class Material { <<type>> FerriteInductorMaterialPropertyType } class FerriteInductorMaterialPropertyType { Air type JEP30-D10:EmptyType IronPowder type JEP30-D10:EmptyType NiFeMoPowder type JEP30-D10:EmptyType NiFePowder type JEP30-D10:EmptyType FeSiAlPowder type JEP30-D10:EmptyType FeSiPowder type JEP30-D10:EmptyType ManganeseZincFerrite type JEP30-D10:EmptyType NickelZincFerrite type JEP30-D10:EmptyType Other type xs:string } PropertyArrayType "1" --> "1" Material Material "1" --> "1" FerriteInductorMaterialPropertyType </pre>
type	FerriteInductorClassificationProperty-ArrayType , FerriteInductorMaterialPropertyType , JEP30-D10:EmptyType .
group	SearchableProperties .

Ferrite Inductors have typically one of the core materials as outlined in the *Material* property above, however other material can be captured under the property *Other* as a string.

4.7.1.3.11.2. MetalAlloys Inductor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/MetalAlloys
diagram	<pre> classDiagram class MetalAlloysInductorClassificationType { Fixed Variable OverflowCategories Property-Array } class MetalAlloys { <<MetalAlloysInductorClassificationType>> } MetalAlloys "3..1" --> MetalAlloysInductorClassificationType class OverflowCategories { type OverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } class PropertyArray { type MetalAlloysInductorClassificationPropertyArrayType } MetalAlloysInductorClassificationType "1..1" --> OverflowCategories MetalAlloysInductorClassificationType "1..1" --> Parts MetalAlloysInductorClassificationType "1..1" --> Accessories MetalAlloysInductorClassificationType "1..1" --> PropertyArray </pre>
type	MetalAlloysInductorClassificationType, FixedMetalAlloysInductorClassificationType, VariableMetalAlloysInductorClassificationType, OverflowCategoriesType, MetalAlloysInductorClassificationPropertyArrayType.
group	OverflowCategories.

4.7.1.3.11.2.1. Variable

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/MetalAlloys/Variable
diagram	<pre> classDiagram class VariableMetalAlloysInductorClassificationType { Variable Property-Array attributes Adjustment SearchableProperties } class Variable { type VariableMetalAlloysInductorClassificationType } Variable "3..1" --> VariableMetalAlloysInductorClassificationType class PropertyArray { type VariableMetalAlloysInductorClassificationPropertyArrayType } PropertyArray "1..1" --> VariableMetalAlloysInductorClassificationType class attributes { <<VariableInductorAdjustmentType>> } attributes "1..1" --> Adjustment class Adjustment { type VariableInductorAdjustmentType } class SearchableProperties </pre>
type	VariableMetalAlloysInductorClassificationType, VariableMetalAlloysInductorClassificationPropertyArrayType, VariableInductorAdjustmentType.
group	SearchableProperties.

4.7.1.3.11.2.2. Property - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/MetalAlloys/Property-Array
diagram	<pre> classDiagram class Property-Array { type MetalAlloysInductorClassificationProperty-ArrayType } class MetalAlloysInductorClassificationProperty-ArrayType { <<MetalAlloysInductorClassificationProperty-ArrayType>> attributes <<MetalAlloysInductorClassificationProperty-ArrayType>> Material <<MetalAlloysInductorMaterialPropertyType>> <<MetalAlloysInductorMaterialPropertyType>> SearchableProperties } class Material { type MetalAlloysInductorMaterialPropertyType } class MetalAlloysInductorMaterialPropertyType { <<MetalAlloysInductorMaterialPropertyType>> Amorphous <<JEP30-D10:EmptyType>> SiliconSteel <<JEP30-D10:EmptyType>> Other <<xs:string>> } class Amorphous class SiliconSteel class Other class SearchableProperties </pre>
type	MetalAlloysInductorClassificationProperty-ArrayType , MetalAlloysInductorMaterialPropertyType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.11.3. Inductor Property - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Property-Array
diagram	<pre> classDiagram class Property-Array { type InductorClassificationProperty-ArrayType } class InductorClassificationProperty-ArrayType { <<InductorClassificationProperty-ArrayType>> attributes <<InductorClassificationProperty-ArrayType>> Shielding <<InductorShieldingClassificationType>> Technology <<InductorTechnologyClassificationType>> Coupled <<JEP30-D10:EmptyType>> SearchableProperties } class Shielding class Technology class Coupled class SearchableProperties </pre>
type	InductorClassificationProperty-ArrayType , InductorShieldingClassificationType , InductorTechnologyClassificationType , JEP30-D10:EmptyType .
group	SearchableProperties .

An *Inductor* can come with various levels of *Shielding* and can be either *Fixed* or *Variable* in value.

Mutual inductance occurs when the change in current in one inductor induces a voltage in another nearby inductor. It is the mechanism by which transformers work.

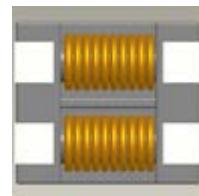


Figure 3 — Coupled Inductor

4.7.1.3.11.3.1. Inductor Shielding Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Property-Array/Shielding</code>
diagram	<pre> classDiagram class InductorShieldingClassificationType { <<InductorShieldingClassificationType>> <<Shielded>> <<SemiShielded>> <<NonShielded>> } class Shielding { <<InductorShieldingClassificationType>> } Shielding --> InductorShieldingClassificationType </pre>
type	<code>InductorShieldingClassificationType, JEP30-D10:EmptyType.</code>

4.7.1.3.11.3.2. Inductor Technology Property

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Property-Array/Technology</code>
diagram	<pre> classDiagram class InductorTechnologyClassificationType { <<InductorTechnologyClassificationType>> <<Bobbin>> <<Toroidal>> <<Multilayer>> <<Film>> <<PotCore>> } class Technology { <<InductorTechnologyClassificationType>> } Technology --> InductorTechnologyClassificationType </pre> <pre> classDiagram class InductorFerriteCoreShapePropertyType { <<InductorFerriteCoreShapePropertyType>> <<E-I>> <<EFD>> <<ETD>> <<EER>> <<EC>> <<U-I>> <<UR>> <<PlanarE-I>> <<ER>> <<Other>> } </pre>
type	<code>InductorTechnologyClassificationType, JEP30-D10:EmptyType, InductorFerriteCoreShapePropertyType.</code>

All Inductors whether fixed or variable can have different `Technology` construction as outlined above, but other technologies can be captured under the `Other` property elements.

4.7.1.3.12. Memory Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Memory
diagram	<pre> classDiagram class MemoryClassificationType class Volatile { <<type MemoryClassificationType>> } class NonVolatile { <<type MemoryClassificationType>> } class Other { <<type OverflowCategoriesType>> } class Parts { <<type OverflowCategoriesType>> } class Accessories { <<type OverflowCategoriesType>> } class Property_Array { <<type MemoryClassificationProperty-ArrayType>> } MemoryClassificationType "1" -- "1" Memory : MemoryClassificationType MemoryClassificationType "1" -- "1" Property_Array : MemoryClassificationProperty-ArrayType MemoryClassificationType "1" -- "1" Other : OverflowCategoriesType Other "1" -- "1" Parts : OverflowCategoriesType Other "1" -- "1" Accessories : OverflowCategoriesType Volatile "1" -- "1" Memory : MemoryClassificationType NonVolatile "1" -- "1" Memory : MemoryClassificationType Other "1" -- "1" Memory : MemoryClassificationType </pre>
type	MemoryClassificationType , VolatileMemoryClassificationType , NonVolatileMemoryClassificationType , OverflowCategoriesType .
group	OverflowCategories .

A *Memory* device is any device that is used to store data or information. *Volatile* memory is computer storage that only maintains its data while the device is powered. *Non-volatile* memory (NVM) is a type of computer memory that has the capability to hold saved data even if the power is turned off. Unlike *Volatile* memory, *Non-volatile* memory does not require its memory data to be periodically refreshed. It is commonly used for secondary storage or long-term consistent storage.

4.7.1.3.12.1. Volatile Memory Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Memory/Volatile
diagram	<pre> classDiagram class Volatile { type VolatileMemoryClassificationType } class CAM { type CAM-MemoryClassificationType } class DRAM { type DRAM-MemoryClassificationType } class PSRAM { type PSRAM-MemoryClassificationType } class SDRAM { type SDRAM-MemoryClassificationType } class SGRAM { type SGRAM-MemoryClassificationType } class SRAM { type SRAM-MemoryClassificationType } class Other { type OverflowCategoriesType } class OverflowCategories { type OverflowCategoriesType } class Property_Array { type VolatileMemoryClassificationProperty-ArrayType } Volatile < -- CAM Volatile < -- DRAM Volatile < -- PSRAM Volatile < -- SDRAM Volatile < -- SGRAM Volatile < -- SRAM Volatile < -- Other Other < -- OverflowCategories OverflowCategories < -- Property_Array </pre>
type	VolatileMemoryClassificationType , CAM-MemoryClassificationType , DRAM-MemoryClassificationType , PSRAM-MemoryClassificationType , SDRAM-MemoryClassificationType , SGRAM-MemoryClassificationType , SRAM-MemoryClassificationType , OverflowCategoriesType , VolatileMemoryClassificationProperty-ArrayType .
group	OverflowCategories .

There are several sub-classifications for both [Volatile](#) memory devices, as shown above, however new sub-classifications can be captured under the category [Other](#).

4.7.1.3.12.2. NonVolatile Memory Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Memory/NonVolatile
diagram	<p>The diagram illustrates the classification of Non-Volatile memory devices. It features a main class, NonVolatileMemoryClassificationType, which contains several sub-classes: CBRAM, EPROM, EEPROM, FLASH, FRAM, MASKROM, MRAM, NVSRAM, PCM, PROM, and ReRam. Below this main class is a Property-Array named NonVolatileMemoryClassificationProperty-ArrayType. A connector links the NonVolatile class to the NonVolatileMemoryClassificationType class. Additionally, there is a OverflowCategories section containing three categories: Other, Parts, and Accessories, each represented by a dashed-line box.</p> <pre> classDiagram class NonVolatileMemoryClassificationType { CBRAM EPROM EEPROM FLASH FRAM MASKROM MRAM NVSRAM PCM PROM ReRam } class NonVolatile { type NonVolatileMemoryClassificationType } class NonVolatileMemoryClassificationPropertyArrayType { type NonVolatileMemoryClassificationPropertyArrayType } class OverflowCategories { class Other { type OverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } } NonVolatile --> NonVolatileMemoryClassificationType </pre>
type	NonVolatileMemoryClassificationType , CBRAM-MemoryClassificationType , EPROM-MemoryClassificationType , EEPROM-MemoryClassificationType , FLASH-MemoryClassificationType , FRAM-MemoryClassificationType , MASKROM-MemoryClassificationType , MRAM-MemoryClassificationType , NVSRAM-MemoryClassificationType , PCM-MemoryClassificationType , PROM-MemoryClassificationType , ReRam-MemoryClassificationType , OverflowCategoriesType , NonVolatileMemoryClassificationProperty-ArrayType
group	OverflowCategories .

There are several sub-classifications for both *Non-volatile* memory devices, as shown above, however new sub-classifications can be captured under the category *Other*.

4.7.1.3.13. Optoelectronics Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic
diagram	<pre> classDiagram class OptoelectronicClassificationType { Display Photoemitter Photosensitive Optocoupler OpticalPositionEncoderClassification SlottedSwitchClassification Photodarlington Other ArrayOverflowCategories Parts Accessories } class Optoelectronic { type OptoelectronicClassificationType } Optoelectronic < --> OptoelectronicClassificationType OptoelectronicClassificationType < --> ArrayOverflowCategories OptoelectronicClassificationType < --> Parts OptoelectronicClassificationType < --> Accessories </pre> <p>The diagram illustrates the classification of optoelectronic devices. It starts with a general class 'Optoelectronic' which has a relationship with 'OptoelectronicClassificationType'. This classification type is further divided into specific categories: 'Display', 'Photoemitter', 'Photosensitive', 'Optocoupler', 'OpticalPositionEncoderClassification', 'SlottedSwitchClassification', and 'Photodarlington'. Additionally, there is a category for 'Other' and two overflow categories: 'ArrayOverflowCategories' and 'Parts' (which also includes 'Accessories').</p>
type	OptoelectronicsClassificationType , DisplayClassification , PhotoemitterClassificationType , PhotosensitiveClassificationType , OptocouplerClassificationType , OpticalPositionEncoderClassificationType , SlottedSwitchClassificationType , PhotodarlingtonOptocouplerClassificationType , ArrayOverflowCategoriesType , OverflowCategoriesType .
group	ArrayOverflowCategories .

Optoelectronic devices are devices that responds to, emits, or modifies electromagnetic radiation in the visible, infrared, and/or ultraviolet spectral regions. These devices utilize electromagnetic radiation in the visible, infrared, and/or ultraviolet spectral regions for its internal operation.

4.7.1.3.13.1. Display Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display</code>
diagram	<pre> classDiagram class Display { type: DisplayClassificationType } class DisplayClassificationType { TFT type: TFT-DisplayClassificationType LED type: LED-DisplayClassificationType Organic type: OrganicDisplayClassificationType ArrayOverflowCategories type: ArrayOverflowCategoriesType Parts type: OverflowCategoriesType Accessories type: OverflowCategoriesType Property-Array type: DisplayClassificationProperty-ArrayType } </pre>
type	<code>DisplayClassificationType, TFT-DisplayClassificationType, LED-DisplayClassificationType, OrganicDisplayClassificationType, ArrayOverflowCategoriesType, OverflowCategoriesType, DisplayClassificationProperty-ArrayType.</code>
group	<code>ArrayOverflowCategories.</code>

A *Display* device is an output device for presentation of information in visual or tactile form (the latter used for example in tactile electronic displays for blind people). Additional sub-classification can be defined under the category *Other*.

The *TFT*, *LED* and the *Organic* displays can be provided in *Array* form as shown below.

4.7.1.3.13.1.1. TFT Display Classification Type

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display/TFT</code>
diagram	<pre> classDiagram class TFT { type: TFT-DisplayClassificationType } class TFT-DisplayClassificationType { Property-Array type: TFT-DisplayClassificationProperty-ArrayType } class Array type: JEP30-D10:EmptyType class SearchableProperties </pre>
type	<code>TFT-DisplayClassificationType, TFT-DisplayClassificationProperty-ArrayType, JEP30-D10:EmptyType.</code>
group	<code>SearchableProperties.</code>

4.7.1.3.13.1.2. LED Display Classification Type

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display/LED
diagram	<pre> classDiagram class LED { <<LED-DisplayClassificationType>> } class LED-DisplayClassificationType { <<LED-DisplayClassificationType>> } class Property-Array { <<LED-DisplayClassificationProperty-ArrayType>> } class LED-DisplayClassificationProperty-ArrayType { <<LED-DisplayClassificationProperty-ArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class JEP30-D10:EmptyType { <<JEP30-D10:EmptyType>> } class SearchableProperties { <<SearchableProperties>> } LED "3" --> Property-Array Property-Array "3" --> Array Array "3" --> SearchableProperties Property-Array "3" --> attributes </pre>
type	LED-DisplayClassificationType , LED-DisplayClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.13.1.3. Organic Display Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display/Organic
diagram	<pre> classDiagram class Organic { <<OrganicDisplayClassificationType>> } class OrganicDisplayClassificationType { <<OrganicDisplayClassificationType>> } class Property-Array { <<OrganicDisplayClassificationProperty-ArrayType>> } class OrganicDisplayClassificationProperty-ArrayType { <<OrganicDisplayClassificationProperty-ArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class JEP30-D10:EmptyType { <<JEP30-D10:EmptyType>> } class SearchableProperties { <<SearchableProperties>> } Organic "3" --> Property-Array Property-Array "3" --> Array Array "3" --> SearchableProperties Property-Array "3" --> attributes </pre>
type	OrganicDisplayClassificationType , OrganicDisplayClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.13.2. Photoemitter Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photoemitter
diagram	<pre> classDiagram class Photoemitter { type PhotoemitterClassificationType } class InfraredEmittingDiode { type InfraredEmittingDiodePhotoemitterClassificationType } class LED { type LED-PhotoemitterClassificationType } class Laser { type LaserPhotoemitterClassificationType } class Other { type ArrayOverflowCategoriesType } class ArrayOverflowCategories { type ArrayOverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } class PropertyArray { type PhotoemitterClassificationProperty-ArrayType } Photoemitter < -- InfraredEmittingDiode Photoemitter < -- LED Photoemitter < -- Laser Photoemitter --> Other Other < -- ArrayOverflowCategories ArrayOverflowCategories < -- Parts ArrayOverflowCategories < -- Accessories Photoemitter --> PropertyArray </pre>
type	PhotoemitterClassificationType , InfraredEmittingDiodePhotoemitterClassificationType , LED-PhotoemitterClassificationType , LaserPhotoemitterClassificationType , ArrayOverflowCategoriesType , OverflowCategoriesType , PhotoemitterClassificationProperty-ArrayType .
group	ArrayOverflowCategories .

Photoemitter devices are device that emits electromagnetic radiation in the visible, infrared, and/or ultraviolet spectral regions. Some of the more common classifications are defined above, but additional classification can be defined under the category *Other*.

4.7.1.3.13.2.1. Photoemitter LED Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photoemitter/LED
diagram	<pre> classDiagram class LED { type LED-PhotoemitterClassificationType } class LEDPhotoemitterClassificationPropertyArrayType { type LED-PhotoemitterClassificationProperty-ArrayType } class attributes class Array { type JEP30-D10:EmptyType } class SearchableProperties LED --> LEDPhotoemitterClassificationPropertyArrayType LEDPhotoemitterClassificationPropertyArrayType < -- attributes LEDPhotoemitterClassificationPropertyArrayType < -- Array Array < -- JEP30-D10:EmptyType Array < -- SearchableProperties </pre>
type	LED-PhotoemitterClassificationType , LED-PhotoemitterClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

NOTE The *LED* category can be supplied in an *Array* form.

4.7.1.3.13.3. Photosensitive Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photosensitive
diagram	<pre> classDiagram class PhotosensitiveClassificationType { Photodiode Phototransistor Phototriac Photodarlington PhotovoltaicDiode Other Parts Accessories ArrayOverflowCategories Property-Array } class Photosensitive { type PhotosensitiveClassificationType } Photosensitive < -- PhotosensitiveClassificationType PhotosensitiveClassificationType < -- Photodiode PhotosensitiveClassificationType < -- Phototransistor PhotosensitiveClassificationType < -- Phototriac PhotosensitiveClassificationType < -- Photodarlington PhotosensitiveClassificationType < -- PhotovoltaicDiode PhotosensitiveClassificationType < -- Other PhotosensitiveClassificationType < -- Parts PhotosensitiveClassificationType < -- Accessories PhotosensitiveClassificationType < -- ArrayOverflowCategories PhotosensitiveClassificationType < -- Property-Array </pre> <p>The diagram shows a class hierarchy. At the top is a dashed-line box labeled 'PhotosensitiveClassificationType'. Inside this box are seven classes: Photodiode, Phototransistor, Phototriac, Photodarlington, PhotovoltaicDiode, Other, and Parts. Below this box is another dashed-line box labeled 'Photosensitive'. A directed association line connects 'Photosensitive' to 'PhotosensitiveClassificationType'. To the right of 'PhotosensitiveClassificationType' is a class 'ArrayOverflowCategories'. Below 'PhotosensitiveClassificationType' is a class 'Property-Array'.</p>
type	PhotosensitiveClassificationType, PhotodiodePhotosensitiveClassificationType, PhototransistorPhotosensitiveClassificationType, PhototriacPhotosensitiveClassificationType, PhotodarlingtonPhotosensitiveClassificationType, PhotovoltaicDiodePhotosensitiveClassificationType, ArrayOverflowCategoriesType, OverflowCategoriesType, PhotosensitiveClassificationProperty-ArrayType.
group	ArrayOverflowCategories .

A **Photosensitive** device is a device that is responsive to electromagnetic radiation in the visible, infrared, and/or ultraviolet spectral regions. Some of the more common classifications are defined above, but additional classification can be defined under the category **Other**. Note that some classifications such as the **Photodiode** and the **Phototransistor** can come in **Array** form.

4.7.1.3.13.3.1. Photodiode Photosensitive Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photosensitive/Photodiode
diagram	<pre> classDiagram class Photodiode { <<PhotodiodePhotosensitiveClassificationType>> } class PhotodiodePhotosensitiveClassificationType { <<PhotodiodePhotosensitiveClassificationProperty-ArrayType>> } class Property_Array { <<PhotodiodePhotosensitiveClassificationProperty-ArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties Photodiode "1" -- "*" PhotodiodePhotosensitiveClassificationType : PhotodiodePhotosensitiveClassificationType "*" -- "*" Property_Array : Property_Array "*" -- "*" Array : classPhotodiodePhotosensitiveClassificationPropertyArrayType { attributes SearchableProperties } </pre>
type	PhotodiodePhotosensitiveClassificationType , PhotodiodePhotosensitiveClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.13.3.2. Phototransistor Photosensitive Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photosensitive/Phototransistor
diagram	<pre> classDiagram class Phototransistor { <<PhototransistorPhotosensitiveClassificationType>> } class PhototransistorPhotosensitiveClassificationType { <<PhototransistorPhotosensitiveClassificationProperty-ArrayType>> } class Property_Array { <<PhototransistorPhotosensitiveClassificationProperty-ArrayType>> } class Array { <<JEP30-D10:EmptyType>> } class SearchableProperties Phototransistor "1" -- "*" PhototransistorPhotosensitiveClassificationType : PhototransistorPhotosensitiveClassificationType "*" -- "*" Property_Array : Property_Array "*" -- "*" Array : classPhototransistorPhotosensitiveClassificationPropertyArrayType { attributes SearchableProperties } </pre>
type	PhototransistorPhotosensitiveClassificationType , PhototransistorPhotosensitiveClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.13.4. Optocoupler Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Opto electronic/Optocoupler
diagram	<pre> classDiagram class Optocoupler { type OptocouplerClassificationType } class OptocouplerClassificationType { Photodiode { type PhotodiodeOptocouplerClassificationType } Photothyristor { type PhotothyristorOptocouplerClassificationType } Phototransistor { type PhototransistorOptocouplerClassificationType } Photodarlington { type PhotodarlingtonOptocouplerClassificationType } Other { type ArrayOverflowCategoriesType } Parts { type OverflowCategoriesType } Accessories { type OverflowCategoriesType } Property-Array { type OptocouplerClassificationProperty-ArrayType } } Optocoupler "3" --> OptocouplerClassificationType </pre>
type	OptocouplerClassificationType , PhotodiodeOptocouplerClassificationType , PhotothyristorOptocouplerClassificationType , PhototransistorOptocouplerClassificationType , PhotodarlingtonOptocouplerClassificationType , ArrayOverflowCategoriesType , OverflowCategoriesType , OptocouplerClassificationProperty-ArrayType .
group	ArrayOverflowCategories .

An *Optocoupler* device designed for the transformation of electrical signals by utilizing optical radiant energy to provide coupling with electrical isolation between the input and the output. (Ref. IEC 747-5.). Some of the more common classifications are defined above, but additional classification can be defined under the category *Other*.

Some classifications such as the *Photodiode* and the *Phototransistor* can come in *Array* form as shown bellow.

4.7.1.3.13.4.1. Photodiode Optocoupler Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Opto electronic/Optocoupler/Photodiode
diagram	<pre> classDiagram class PhotodiodeOptocouplerClassificationType class Property-Array class PhotodiodeOptocouplerClassificationProperty-ArrayType { <<attributes>> <<Array<<JEP30-D10:EmptyType>>> } PhotodiodeOptocouplerClassificationType "1" --> "1" Property-Array Property-Array "1" --> "1" PhotodiodeOptocouplerClassificationProperty-ArrayType </pre>
type	PhotodiodeOptocouplerClassificationType , PhotodiodeOptocouplerClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.13.4.2. Phototransistor Optocoupler Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Opto electronic/Optocoupler/ Phototransistor, JEP30-D10:EmptyType, PropertyKeyValuePairType
diagram	<pre> classDiagram class PhototransistorOptocouplerClassificationType class Property-Array class PhototransistorOptocouplerClassificationProperty-ArrayType { <<attributes>> <<Array<<JEP30-D10:EmptyType>>> } PhototransistorOptocouplerClassificationType "1" --> "1" Property-Array Property-Array "1" --> "1" PhototransistorOptocouplerClassificationProperty-ArrayType </pre>
type	PhototransistorOptocouplerClassificationType , PhototransistorOptocouplerClassificationProperty-ArrayType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.14. Regulator Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator
diagram	<pre> classDiagram class PowerRegulatorClassificationType { <<PowerRegulatorClassificationType>> <<LinearCurrentRegulatorClassificationType>> <<LinearVoltageRegulatorClassificationType>> <<SwitchingPowerRegulatorClassificationType>> <<SCR_PowerRegulatorClassificationType>> <<OverflowCategoriesType>> <<PowerRegulatorClassificationPropertyArrayType>> } class Regulator { <<PowerRegulatorClassificationType>> } class LinearCurrent { <<LinearCurrentRegulatorClassificationType>> } class LinearVoltage { <<LinearVoltageRegulatorClassificationType>> } class Switching { <<SwitchingPowerRegulatorClassificationType>> } class SCR { <<SCR_PowerRegulatorClassificationType>> } class Other { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } PowerRegulatorClassificationType < -- Regulator PowerRegulatorClassificationType < -- LinearCurrent PowerRegulatorClassificationType < -- LinearVoltage PowerRegulatorClassificationType < -- Switching PowerRegulatorClassificationType < -- SCR PowerRegulatorClassificationType < -- Other PowerRegulatorClassificationType < -- Parts PowerRegulatorClassificationType < -- Accessories PowerRegulatorClassificationType < -- Property-Array </pre> <p>The diagram illustrates the classification of power regulators. It starts with a general class PowerRegulatorClassificationType, which is further divided into specific categories: LinearCurrent, LinearVoltage, Switching, SCR, Other, Parts, and Accessories. Additionally, there is a Property-Array associated with the classification type. A Regulator object is shown to inherit from PowerRegulatorClassificationType.</p>
type	PowerRegulatorClassificationType , LinearCurrentRegulatorClassificationType , LinearVoltageRegulatorClassificationType , SwitchingPowerRegulatorClassificationType , SCR-PowerRegulatorClassificationType , OverflowCategoriesType , PowerRegulatorClassificationPropertyArrayType
group	OverflowCategories .

A power **Regulator** is a device that is designed to have minimal changes in the regulated power over a broad input range. There are various kinds of regulators available as listed above, and other variations can be defined under the category **Other**. Typically, these **Regulator** have one or more various **Protections**.

4.7.1.3.14.1. Linear Current Regulator Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearCurrent
diagram	<pre> classDiagram class LinearCurrentRegulatorClassificationType { <<Fixed>> <<Variable>> <<Other>> class OverflowCategories class Parts class Accessories <<Property-Array>> type LinearCurrentRegulatorClassificationProperty-ArrayType } class LinearCurrent { type LinearCurrentRegulatorClassificationType } LinearCurrent "1" --> "1" LinearCurrentRegulatorClassificationType </pre>
type	LinearCurrentRegulatorClassificationType , FixedLinearCurrentRegulatorClassificationType , VariableLinearCurrentRegulatorClassificationType , OverflowCategoriesType , LinearCurrentRegulatorClassificationProperty-ArrayType
group	OverflowCategories .

A [LinearCurrent](#) regulator device is a system used to maintain a steady current. The resistance of the regulator varies in accordance with the load resulting in a constant output current.

4.7.1.3.14.2. Linear Current Regulator Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearCurrent/Property-Array
diagram	<pre> classDiagram class LinearCurrentRegulatorClassificationProperty-ArrayType { <<attributes>> <<LowDropout>> <<Protection>> class OverTemperature class OverVoltage class UnderVoltage <<SearchableProperties>> } class Property-Array { type LinearCurrentRegulatorClassificationProperty-ArrayType } Property-Array "1..3" --> "1" Protection </pre>
type	LinearCurrentRegulatorClassificationProperty-ArrayType , JEP30-D10:EmptyType , LinearCurrentRegulatorProtectionType .
group	SearchableProperties .

Linear Current regulators come with a [LowDropout](#) option, and have [Protection](#) against [OverTemperature](#), [OverVoltage](#), and or [UnderVoltage](#).

4.7.1.3.14.3. Linear Voltage Regulator Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearVoltage
diagram	<pre> classDiagram class LinearVoltage { type LinearVoltageRegulatorClassificationType } class LinearVoltageRegulatorClassificationType { Fixed type FixedLinearVoltageRegulatorClassificationType Variable type VariableLinearVoltageRegulatorClassificationType Other type OverflowCategoriesType <> OverflowCategories <> Parts <> Accessories <> OverflowCategories } class Property_Array { type LinearVoltageRegulatorClassificationPropertyArrayType } </pre>
type	LinearVoltageRegulatorClassificationType , FixedLinearVoltageRegulatorClassificationType , VariableLinearVoltageRegulatorClassificationType , OverflowCategoriesType , LinearVoltageRegulatorClassificationPropertyArrayType
group	OverflowCategories .

A *Linear* regulator device is a system used to maintain a steady voltage. The resistance of the regulator varies in accordance with the load resulting in a constant output voltage. The regulating device is made to act like a variable resistor, continuously adjusting a voltage divider network to maintain a constant output voltage and continually dissipating the difference between the input and regulated voltages as waste heat. Linear regulators come with a *LowDropout* option.

4.7.1.3.14.4. Linear Voltage Regulator Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearVoltage/Property-Array
diagram	<pre> classDiagram class Property_Array { type LinearVoltageRegulatorClassificationPropertyArrayType } class LinearVoltageRegulatorClassificationPropertyArrayType { <> attributes LowDropout type JEP30-D10:EmptyType Protection type LinearVoltageRegulatorProtectionType <> SearchableProperties } class LinearVoltageRegulatorProtectionType { ShortCircuit type JEP30-D10:EmptyType OverCurrent type JEP30-D10:EmptyType OverTemperature type JEP30-D10:EmptyType } </pre>
type	LinearVoltageRegulatorClassificationPropertyArrayType , JEP30-D10:EmptyType , LinearVoltageRegulatorProtectionType .
group	SearchableProperties .

Linear Voltage regulators come with a *LowDropout* option, and have *Protection* against *ShortCircuit*, *OverCurrent*, and/or *OverTemperature*.

4.7.1.3.14.5. Switching Regulator Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching
diagram	<pre> classDiagram class SwitchingPowerRegulatorClassificationType { <<Switching>> <<SwitchingPowerRegulatorClassificationType>> } class Isolated { <<IsolatedSwitchingPowerRegulatorClassificationType>> } class NonIsolated { <<Non-isolatedSwitchingPowerRegulatorClassificationType>> } class OverflowCategories class Parts class Accessories class PropertyArray { <<SwitchingPowerRegulatorClassificationPropertyArrayType>> } SwitchingPowerRegulatorClassificationType "1" -- "*" Isolated : SwitchingPowerRegulatorClassificationType "1" -- "*" NonIsolated : SwitchingPowerRegulatorClassificationType "1" -- "*" OverflowCategories : SwitchingPowerRegulatorClassificationType "1" -- "*" Parts : SwitchingPowerRegulatorClassificationType "1" -- "*" Accessories : SwitchingPowerRegulatorClassificationType "1" -- "*" PropertyArray : </pre>
type	SwitchingPowerRegulatorClassificationType , IsolatedSwitchingPowerRegulatorClassificationType , Non-isolatedSwitchingPowerRegulatorClassificationType , OverflowCategoriesType , SwitchingPowerRegulatorClassificationPropertyArrayType
group	OverflowCategories.

In contrast to a *Linear* regulator, a *Switching* regulator uses an active device that switches on and off to maintain an average value of output. The duty cycle of the switch sets how much charge is transferred to the load.

4.7.1.3.14.5.1. Isolated Switching Regulator Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching/Isolated
diagram	<pre> classDiagram class IsolatedSwitchingPowerRegulatorClassificationType { Flyback Forward Push-Pull Isolated OverflowCategories Property-Array } class Flyback { type FlybackIsolatedSwitchingPowerRegulatorClassificationType } class Forward { type ForwardIsolatedSwitchingPowerRegulatorClassificationType } class PushPull { type PushPullIsolatedSwitchingPowerRegulatorClassificationType } class Isolated { type IsolatedSwitchingPowerRegulatorClassificationType } class OverflowCategories { type OverflowCategoriesType } class PropertyArray { type IsolatedSwitchingPowerRegulatorClassificationPropertyArrayType } IsolatedSwitchingPowerRegulatorClassificationType < -- Flyback IsolatedSwitchingPowerRegulatorClassificationType < -- Forward IsolatedSwitchingPowerRegulatorClassificationType < -- PushPull IsolatedSwitchingPowerRegulatorClassificationType < -- Isolated IsolatedSwitchingPowerRegulatorClassificationType < -- OverflowCategories IsolatedSwitchingPowerRegulatorClassificationType < -- PropertyArray </pre> <p>The diagram illustrates the classification of isolated switching power regulators. It starts with a general class, <code>IsolatedSwitchingPowerRegulatorClassificationType</code>, which is a supertype for four specific types: <code>Flyback</code>, <code>Forward</code>, <code>Push-Pull</code>, and <code>Isolated</code>. Additionally, it includes two overflow categories, <code>OverflowCategories</code> and <code>Property-Array</code>. The <code>Isolated</code> type is shown with a dashed line, indicating it is a separate entity within the classification hierarchy.</p>
type	IsolatedSwitchingPowerRegulatorClassificationType , FlybackIsolatedSwitchingPowerRegulatorClassificationType , ForwardIsolatedSwitchingPowerRegulatorClassificationType , Push-PullIsolatedSwitchingPowerRegulatorClassificationType , OverflowCategoriesType , IsolatedSwitchingPowerRegulatorClassificationPropertyArrayType
group	OverflowCategories .

4.7.1.3.14.5.2. Non-isolated Switching Regulator Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching/Non-isolated
diagram	<pre> classDiagram class Non-isolatedSwitchingPowerRegulatorClassificationType { <<Non-isolated>> <<Non-isolatedSwitchingPowerRegulatorClassificationType>> } class Boost { <<BoostNon-isolatedSwitchingPowerRegulatorClassificationType>> } class Buck { <<BuckNon-isolatedSwitchingPowerRegulatorClassificationType>> } class BuckBoost { <<Buck-BoostNon-isolatedSwitchingPowerRegulatorClassificationType>> } class ChargePump { <<ChargePumpNon-isolatedSwitchingPowerRegulatorClassificationType>> } class Other { <<OverflowCategoriesType>> } class Parts { <<OverflowCategoriesType>> } class Accessories { <<OverflowCategoriesType>> } class OverflowCategories class Property_Array { <<Non-isolatedSwitchingPowerRegulatorClassificationProperty-ArrayType>> } Non-isolatedSwitchingPowerRegulatorClassificationType < -- Boost Non-isolatedSwitchingPowerRegulatorClassificationType < -- Buck Non-isolatedSwitchingPowerRegulatorClassificationType < -- BuckBoost Non-isolatedSwitchingPowerRegulatorClassificationType < -- ChargePump Non-isolatedSwitchingPowerRegulatorClassificationType < -- Other Other < -- Parts Other < -- Accessories Non-isolatedSwitchingPowerRegulatorClassificationType --> OverflowCategories Non-isolatedSwitchingPowerRegulatorClassificationType --> Property_Array </pre>
type	Non-isolatedSwitchingPowerRegulatorClassificationType , BoostNon-isolatedSwitchingPowerRegulatorClassificationType , BuckNon-isolatedSwitchingPowerRegulatorClassificationType , Buck-BoostNon-isolatedSwitchingPowerRegulatorClassificationType , ChargePumpNon-isolatedSwitchingPowerRegulatorClassificationType , OverflowCategoriesType , Non-isolatedSwitchingPowerRegulatorClassificationProperty-ArrayType
group	OverflowCategories .

4.7.1.3.15. Relay Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Relay
diagram	<pre> classDiagram class RelayClassificationType { <<RelayClassificationType>> <<ElectroMechanical>> <<SolidState>> <<RelayOverflowCategories>> <<Property-Array>> } class ElectroMechanical { <<ElectroMechanicalRelayClassificationType>> } class SolidState { <<SolidStateRelayClassificationType>> } class RelayOverflowCategories { <<OverflowCategoriesType>> } class RelayClassificationPropertyArrayType { <<RelayClassificationProperty-ArrayType>> } RelayClassificationType "3..1" -- "1..1" ElectroMechanical RelayClassificationType "3..1" -- "1..1" SolidState RelayClassificationType "3..1" -- "1..1" RelayOverflowCategories RelayOverflowCategories "*" -- "1..1" Property-Array </pre>
type	RelayClassificationType , ElectroMechanicalRelayClassificationType , SolidStateRelayClassificationType , RelayOverflowCategoriesType , OverflowCategoriesType , RelayClassificationProperty-ArrayType .
group	RelayOverflowCategories .

A [Relay](#) is an electrically operated switch. Many relays use an electromagnet ([Coil](#)) to mechanically operate a switch, but other operating principles are also used, such as [SolidState](#). Relays are used where it is necessary to control a circuit by a separate low-power signal, or where several circuits must be controlled by one signal.

4.7.1.3.15.1. ElectroMechanical Relay Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Relay/ElectroMechanical
diagram	<pre> classDiagram class ElectroMechanicalRelayClassificationType { <<ElectroMechanical>> <<type ElectroMechanicalRelayClassificationType>> } class Property_Array { <<type ElectroMechanicalRelayClassificationPropertyArrayType>> } class ContactForm { <<type RelayContactFormType>> } class FunctionalClassification { <<type RelayFunctionalClassificationType>> } class RelayContactFormType { <<SinglePole-SingleThrow>> <<SinglePole-DoubleThrow>> <<DoublePole-SingleThrow>> <<DoublePole-DoubleThrow>> <<MultiPole-SingleThrow>> <<MultiPole-DoubleThrow>> } class RelayFunctionalClassificationType { <<Power>> <<RF>> <<Signal>> } class SearchableProperties ElectroMechanicalRelayClassificationType "0..1" -- "1" Property_Array Property_Array "0..1" -- "1" ContactForm Property_Array "0..1" -- "1" FunctionalClassification ContactForm "0..1" -- "1" RelayContactFormType FunctionalClassification "0..1" -- "1" RelayFunctionalClassificationType ContactForm "*" -- "*" SearchableProperties FunctionalClassification "*" -- "*" SearchableProperties </pre>
type	ElectroMechanicalRelayClassificationType , ElectroMechanicalRelayClassificationPropertyArrayType , RelayContactFormType , RelayFunctionalClassificationType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.15.2. SolidState Relay Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Relay/SolidState
diagram	<pre> classDiagram class SolidStateRelayClassificationType { <<SolidState>> <<type SolidStateRelayClassificationType>> } class Property_Array { <<type SolidStateRelayClassificationPropertyArrayType>> } class ContactForm { <<type RelayContactFormType>> } class FunctionalClassification { <<type RelayFunctionalClassificationType>> } class RelayContactFormType class SearchableProperties SolidStateRelayClassificationType "0..1" -- "1" Property_Array Property_Array "0..1" -- "1" ContactForm Property_Array "0..1" -- "1" FunctionalClassification ContactForm "0..1" -- "1" RelayContactFormType ContactForm "*" -- "*" SearchableProperties FunctionalClassification "*" -- "*" SearchableProperties </pre>
type	SolidStateRelayClassificationType , SolidStateRelayClassificationPropertyArrayType , RelayContactFormType , RelayFunctionalClassificationType .
group	SearchableProperties .

4.7.1.3.15.3. Relay Overflow Categories Group

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Relay/Other
diagram	<pre> classDiagram class RelayOverflowCategoriesType { <<RelayOverflowCategories>> Other Parts Accessories Property-Array } class RelayOverflowCategoriesPropertyArrayType { ContactForm FunctionalClassification SearchableProperties } Other "Sub-CategoryName" Property-Array "type: RelayOverflowCategoriesProperty-ArrayType" RelayOverflowCategoriesPropertyArrayType "type: RelayOverflowCategoriesProperty-ArrayType" </pre>
type	RelayOverflowCategoriesType , RelayOverflowCategoriesProperty-ArrayType , RelayContactFormType , RelayFunctionalClassificationType , PropertyValuePairType , OverflowCategoriesType .
group	RelayOverflowCategories , SearchableProperties .

4.7.1.3.16. Resistor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor
diagram	<pre> classDiagram class Resistor { <<Resistor>> ResistorClassificationType } class ResistorClassificationType { Fixed Adjustable Non-Linear Other } class OverflowCategories class Property-Array Other "Parts" Other "Accessories" </pre>
type	ResistorClassificationType , FixedResistorClassificationType , AdjustableResistorClassificationType , NonLinearResistorClassificationType , OverflowCategoriesType , ResistorClassificationProperty-ArrayType .
group	OverflowCategories .

A **Resistor** is a passive electrical component that implements electrical resistance as a circuit element. They are used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses.

4.7.1.3.16.1. Fixed Resistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Fixed
diagram	<pre> classDiagram class FixedResistorClassificationType class FixedResistorClassificationPropertyArrayType { <<Attributes>> class Array { type JEP30-D10:EmptyType } class Material class SearchableProperties } class FixedResistorMaterialPropertyType { <<CarbonComposition, Shunt, MetalFilm, ThinFilm, ThickFilm, WireWound, Other>> } class Fixed class Property-Array Fixed < -- FixedResistorClassificationType FixedResistorClassificationType < -- FixedResistorClassificationPropertyArrayType FixedResistorClassificationPropertyArrayType < -- Array FixedResistorClassificationPropertyArrayType < -- Material FixedResistorClassificationPropertyArrayType < -- SearchableProperties FixedResistorMaterialPropertyType < -- CarbonComposition FixedResistorMaterialPropertyType < -- Shunt FixedResistorMaterialPropertyType < -- MetalFilm FixedResistorMaterialPropertyType < -- ThinFilm FixedResistorMaterialPropertyType < -- ThickFilm FixedResistorMaterialPropertyType < -- WireWound FixedResistorMaterialPropertyType < -- Other </pre>
type	FixedResistorClassificationType, FixedResistorClassificationProperty-ArrayType, JEP30-D10:EmptyType, FixedResistorMaterialPropertyType.
group	SearchableProperties.

Fixed resistors have resistances that only change slightly with temperature, time or operating voltage.

4.7.1.3.16.2. Adjustable Resistor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Adjustable
diagram	<pre> classDiagram class Adjustable { type: AdjustableResistorClassificationType } class OverflowCategories { type: OverflowCategoriesType } class Potentiometer { type: PotentiometerClassificationType } class Rheostat { type: RheostatClassificationType } class Trimmer { type: TrimmerClassificationType } class Other { type: OverflowCategoriesType } class Parts { type: OverflowCategoriesType } class Accessories { type: OverflowCategoriesType } class Property_Array { type: AdjustableResistorClassificationPropertyArrayType } Adjustable "1" --> "1" OverflowCategories OverflowCategories "1" --> "1" Potentiometer OverflowCategories "1" --> "1" Rheostat OverflowCategories "1" --> "1" Trimmer OverflowCategories "1" --> "1" Other OverflowCategories "1" --> "1" Parts OverflowCategories "1" --> "1" Accessories </pre>
type	AdjustableResistorClassificationType , PotentiometerClassificationType , RheostatClassificationType , TrimmerClassificationType , OverflowCategoriesType , AdjustableResistorClassificationProperty-ArrayType .
group	OverflowCategories .

Adjustable resistors can be used to adjust circuit elements, or as sensing devices for heat, light, humidity, force, or chemical activity.

4.7.1.3.16.3. Non-Linear Resistor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Non-Linear
diagram	<pre> classDiagram class NonLinearResistorClassificationType { <<Non-Linear>> <<Non-LinearResistorClassificationType>> } class Thermistor { <<Thermistor>> <<ThermistorClassificationType>> } class Varistor { <<Varistor>> <<VaristorResistorClassificationType>> } class PhotoResistor { <<PhotoResistor>> <<PhotoResistorClassificationType>> } class Magnetic { <<Magnetic>> <<MagneticResistorClassificationType>> } class Other { <<Other>> <<OverflowCategoriesType>> } class Parts { <<Parts>> <<OverflowCategoriesType>> } class Accessories { <<Accessories>> <<OverflowCategoriesType>> } class PropertyArray { <<Property-Array>> <<Non-LinearResistorProperty-ArrayType>> } NonLinearResistorClassificationType "3" -- "1" Thermistor NonLinearResistorClassificationType "3" -- "1" Varistor NonLinearResistorClassificationType "3" -- "1" PhotoResistor NonLinearResistorClassificationType "3" -- "1" Magnetic NonLinearResistorClassificationType "3" -- "1" Other NonLinearResistorClassificationType "3" -- "1" Parts NonLinearResistorClassificationType "3" -- "1" Accessories NonLinearResistorClassificationType "3" -- "1" PropertyArray </pre> <p>The diagram illustrates the classification of Non-Linear Resistors. It starts with a general class 'Non-Linear' which branches into specific types: Thermistor, Varistor, PhotoResistor, Magnetic, and Other. The 'Other' category further branches into Parts and Accessories. Additionally, there is a 'Property-Array' class.</p>
type	LinearResistorClassificationType , ThermistorClassificationType , VaristorResistorClassificationType , PhotoResistorClassificationType , MagneticResistorClassificationType , OverflowCategoriesType , Non-LinearResistorProperty-ArrayType .
group	OverflowCategories .

4.7.1.3.16.4. Thermistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Non-Linear/Termistor
diagram	<pre> classDiagram class ThermistorClassificationType class ThermistorClassificationPropertyArrayType { <<attributes>> } class TemperatureCoefficient { <<ThermistorTemperatureCoefficientType>> } class Material { <<ThermistorMaterialType>> } class SearchableProperties ThermistorClassificationType "1" -- "1" Thermistor : type Thermistor --> "1..1" Property-Array : type Property-Array --> "1..1" TemperatureCoefficient : type Property-Array --> "1..1" Material : type TemperatureCoefficient --> "1..1" NTC : type TemperatureCoefficient --> "1..1" PTC : type TemperatureCoefficient --> "1..1" Other : type Material --> "1..1" Ceramic : type Material --> "1..1" Polymer : type Material --> "1..1" OtherMaterial : type </pre> <p>The diagram illustrates the UML class structure for Thermistor classification and properties. It starts with a Thermistor class (type ThermistorClassificationType) which has a relationship to a Property-Array (type ThermistorClassificationPropertyArrayType). The Property-Array has relationships to TemperatureCoefficient and Material. The TemperatureCoefficient class (type ThermistorTemperatureCoefficientType) is associated with three subtypes: NTC, PTC, and Other. The Material class (type ThermistorMaterialType) is associated with three subtypes: Ceramic, Polymer, and OtherMaterial. Additionally, there is a generalization relationship from SearchableProperties to both TemperatureCoefficient and Material.</p>
type	ThermistorClassificationType , ThermistorClassificationPropertyArrayType , ThermistorTemperatureCoefficientType , ThermistorMaterialType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.17. RF Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/RF
diagram	<p>The diagram illustrates the classification of RF components. It features a main class, RF-ClassificationType, which contains 18 subclasses: Antenna, Attenuator, Balun, Circulator, Combiner, Coupler, Detector, Divider, Isolator, Limiter, Mixer, Multiplier, PhaseShifter, Receiver, Transceiver, and Transmitter. Below these, there is a generalization relationship where RF (also labeled as RF-ClassificationType) is the superclass of RF-ClassificationType. Additionally, there is a Property-Array associated with RF-ClassificationType. At the bottom right, there is a group named OverflowCategories containing three items: Parts, Accessories, and Other.</p> <pre> classDiagram class RF { <<RF-ClassificationType>> } class RFClassificationType { <<RF-ClassificationType>> Antenna Attenuator Balun Circulator Combiner Coupler Detector Divider Isolator Limiter Mixer Multiplier PhaseShifter Receiver Transceiver Transmitter } RF < -- RFClassificationType RFClassificationType < -- Property-Array class OverflowCategories { <<OverflowCategoriesType>> Parts Accessories Other } </pre>
type	RF-ClassificationType , RF-AntennaClassificationType , RF-AttenuatorClassificationType , RF-BalunClassificationType , RF-CirculatorClassificationType , RF-CombinerClassificationType , RF-CouplerClassificationType , RF-DetectorClassificationType , RF-DividerClassificationType , RF-IsolatorClassificationType , RF-LimiterClassificationType , RF-MixerClassificationType , RF-MultiplierClassificationType , RF-PhaseShifterClassificationType , RF-ReceiverClassificationType , RF-TranceiverClassificationType , RF-TransmitterClassificationType , OverflowCategoriesType , RF-ClassificationProperty-ArrayType
group	OverflowCategories .

4.7.1.3.17 RF Classification (cont'd)

RF is any frequency within the electromagnetic spectrum associated with radio wave propagation. When an RF current is supplied to an antenna, an electromagnetic field is created that then can propagate through space. This section covers devices that operate with RF type signals. The section has a broad set of sub-classifications, with the provision to define other categories, if needed under the category *Other*.

4.7.1.3.17.1. RF Detector Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/RF/Detector
diagram	<pre> classDiagram class RF-DetectorClassificationType { class BalancedInput { type BalancedInputRF-DetectorClassificationType } class UnBalancedInput { type UnBalancedInputRF-DetectorClassificationType } class OverflowCategories { type OverflowCategoriesType } class Other { type OverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } class Property-Array { type RF-DetectorClassificationProperty-ArrayType } } class Detector { type RF-DetectorClassificationType } Detector < -- RF-DetectorClassificationType Detector --> BalancedInput Detector --> UnBalancedInput Detector --> OverflowCategories Detector --> Other Detector --> Parts Detector --> Accessories Detector --> Property-Array </pre>
type	RF-DetectorClassificationType, BalancedInputRF-DetectorClassificationType, UnBalancedInputRF-DetectorClassificationType, OverflowCategoriesType, RF-DetectorClassificationProperty-ArrayType
group	OverflowCategories.

4.7.1.3.17.2. RF Mixer Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/RF/Mixer</code>
diagram	<pre> classDiagram class RF-MixerClassificationType { Balanced DoubleBalanced TripleBalanced Other Parts Accessories OverflowCategories Property-Array } class Mixer { type RF-MixerClassificationType } Mixer --> RF-MixerClassificationType RF-MixerClassificationType < -- Balanced RF-MixerClassificationType < -- DoubleBalanced RF-MixerClassificationType < -- TripleBalanced RF-MixerClassificationType < -- Other RF-MixerClassificationType < -- Parts RF-MixerClassificationType < -- Accessories RF-MixerClassificationType < -- OverflowCategories RF-MixerClassificationType < -- Property-Array </pre> <p>The diagram illustrates the classification of RF Mixers. It features a main class, <code>RF-MixerClassificationType</code>, which is a generalization of several specific types: <code>Balanced</code>, <code>DoubleBalanced</code>, <code>TripleBalanced</code>, <code>Other</code>, <code>Parts</code>, and <code>Accessories</code>. Additionally, it includes two arrays: <code>OverflowCategories</code> and <code>Property-Array</code>. A <code>Mixer</code> object is shown to inherit from <code>RF-MixerClassificationType</code>.</p>
type	<code>RF-MixerClassificationType, BalancedRF-MixerClassificationType,</code> <code>DoubleBalancedRF-MixerClassificationType, TripleBalancedRF-MixerClassificationType,</code> <code>OverflowCategoriesType, RF-MixerClassificationProperty-ArrayType</code>
group	<code>OverflowCategories.</code>

4.7.1.3.18. Sensor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Sensor
diagram	<pre> classDiagram class SensorClassificationType { Accelerometer Capacitive-Touch Current Encoder Flow Gyro Humidity Magnetic Photo Position Pressure Proximity Resistance Temperature Tilt Vibration Other Parts Accessories OverflowCategories Property-Array } SensorClassificationType < -- Sensor SensorClassificationType < -- OverflowCategories SensorClassificationType < -- Property-Array </pre> <p>The diagram shows a class hierarchy for sensor classification. At the top is a dashed box labeled "SensorClassificationType". Inside this box are 15 specific sensor types: Accelerometer, Capacitive-Touch, Current, Encoder, Flow, Gyro, Humidity, Magnetic, Photo, Position, Pressure, Proximity, Resistance, Temperature, Tilt, and Vibration. Below this box is another dashed box labeled "OverflowCategories", which contains three items: Other, Parts, and Accessories. To the right of the main box is a dashed box labeled "Property-Array". A line connects the "SensorClassificationType" box to a "Sensor" class below it. Another line connects the "OverflowCategories" box to a "Property-Array" box.</p>
type	SensorClassificationType , AccelerometerSensorClassificationType , Capacitive-TouchSensorClassificationType , CurrentSensorClassificationType , EncoderSensorClassificationType , FlowSensorClassificationType , GyroSensorClassificationType , HumiditySensorClassificationType , MagneticSensorClassificationType , PhotoSensorClassificationType , PositionSensorClassificationType , PressureSensorClassificationType , ProximitySensorClassificationType , ResistanceSensorClassificationType , TemperatureSensorClassificationType , TiltSensorClassificationType , VibrationSensorClassificationType , OverflowCategoriesType , SensorClassificationProperty-ArrayType
group	OverflowCategories .

4.7.1.3.18 Sensor Classification (cont'd)

A *Sensor* is an electronic component, module, or subsystem whose purpose is to detect events or changes in its environment and send the information to other electronic devices. There is a broad classification of sensors as shown above, however new sub-classifications can be captured under the category *Other*.

4.7.1.3.19. Switch Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Switch</code>
diagram	<pre> classDiagram class SwitchClassificationType { DIP KeyLock Rotary Rocker PushButton Reed Slide Tactile Toggle Other Parts Accessories OverflowCategories Property-Array } class Switch { type SwitchClassificationType } class DIP { type DIP-SwitchClassificationType } class KeyLock { type KeyLockSwitchClassificationType } class Rotary { type RotarySwitchClassificationType } class Rocker { type RockerSwitchClassificationType } class PushButton { type PushButtonSwitchClassificationType } class Reed { type ReedSwitchClassificationType } class Slide { type SlideSwitchClassificationType } class Tactile { type TactileSwitchClassificationType } class Toggle { type ToggleSwitchClassificationType } class Other { type OverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } class OverflowCategories class Property-Array { type SwitchClassificationProperty-ArrayType } Switch < -- DIP Switch < -- KeyLock Switch < -- Rotary Switch < -- Rocker Switch < -- PushButton Switch < -- Reed Switch < -- Slide Switch < -- Tactile Switch < -- Toggle Switch < -- Other Switch < -- Parts Switch < -- Accessories Switch --> OverflowCategories OverflowCategories --> Property-Array </pre>
type	<code>SwitchClassificationType, DIP-SwitchClassificationType, KeyLockSwitchClassificationType, RotarySwitchClassificationType, RockerSwitchClassificationType, PushButtonSwitchClassificationType, ReedSwitchClassificationType, SlideSwitchClassificationType, TactileSwitchClassificationType, ToggleSwitchClassificationType, OverflowCategoriesType, SwitchClassificationProperty-ArrayType.</code>
group	<code>OverflowCategories.</code>

A *Switch* is an electrical component that can "make" or "break" an electrical circuit, interrupting the current or diverting it from one conductor to another. The mechanism of a switch removes or restores the conducting path in a circuit when it is operated.

4.7.1.3.19.1. Switch Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Switch/Property-Array
diagram	<pre> classDiagram class SwitchClassificationPropertyArrayType { <<attributes>> } class PropertyArray { <<type>> SwitchClassificationPropertyArrayType } class ContactForm { <<type>> SwitchContactFormType } class SearchableProperties { <<+>> } SwitchClassificationPropertyArrayType "1" -- "1" PropertyArray PropertyArray "*" -- "1" ContactForm ContactForm "*" -- "1" SearchableProperties </pre> <p>The diagram illustrates the UML class structure for the <code>Switch Property-Array</code>. It features a main class <code>SwitchClassificationPropertyArrayType</code> which has an association with <code>Property-Array</code> (type <code>SwitchClassificationPropertyArrayType</code>). This association is multiplicity 1 on both sides. The <code>Property-Array</code> class has a multiplicity * on its side and an association with <code>ContactForm</code> (type <code>SwitchContactFormType</code>). This association is multiplicity 1 on the <code>ContactForm</code> side. The <code>ContactForm</code> class also has a multiplicity * on its side and an association with <code>SearchableProperties</code>. This association is multiplicity 1 on the <code>SearchableProperties</code> side.</p>
type	SwitchClassificationPropertyArrayType , SwitchContactFormType , JEP30-D10:EmptyType , PropertyValuePairType

4.7.1.3.20. Thyristor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Thyristor
diagram	<pre> classDiagram class ThyristorClassificationType { DIAC MCT SCR SIDAC TRIAC Triode } class Thyristor { type ThyristorClassificationType } class ThyristorOverflowCategories { type ThyristorClassificationProperty-ArrayType } class Property-Array { type ThyristorClassificationProperty-ArrayType } ThyristorClassificationType < -- Thyristor ThyristorClassificationType < -- ThyristorOverflowCategories ThyristorClassificationType < -- Property-Array </pre> <p>The diagram illustrates the classification of thyristors. It features a main class, ThyristorClassificationType, which is a generalization of several specific thyristor types: DIAC, MCT, SCR, SIDAC, TRIAC, and Triode. Below this, there is a Thyristor class that inherits from ThyristorClassificationType. Additionally, ThyristorClassificationType is associated with ThyristorOverflowCategories and Property-Array via aggregation relationships.</p>
type	ThyristorClassificationType , DIAC-ThyristorClassificationType , MCT-ThyristorClassificationType , SCR-ThyristorClassificationType , SIDAC-ThyristorClassificationType , TRIAC-ThyristorClassificationType , TriodeThyristorClassificationType , ThyristorOverflowCategoriesType , OverflowCategoriesType , ThyristorClassificationProperty-ArrayType
group	ThyristorOverflowCategories .

A **Thyristor**, also known as a semiconductor-controlled rectifier (SCR) or silicon-controlled rectifier (SCR), is a solid-state semiconductor device with four layers of alternating N and P-type materials. It acts exclusively as a bistable switch, conducting when the gate receives a current trigger, and continuing to conduct while the voltage across the device is not reversed (forward biased). A three-terminal thyristor is designed to control the larger current of its two terminals by combining that current with the smaller current of its other terminals, known as its control terminal. In contrast, a two-terminal thyristor is designed to switch on if the potential difference between its terminals is sufficiently large (breakdown voltage).

4.7.1.3.20.1. Triode Thyristor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Thyristor/Triode
diagram	<pre> classDiagram class TriodeThyristorClassificationType class Property_Array { <<type TriodeThyristorClassificationProperty-ArrayType>> } class TriodeThyristorClassificationPropertyArrayType { <<type TriodeThyristorClassificationProperty-ArrayType>> <<attributes>> } class GateType { <<type TriodeThyristorGateType>> } class N_gate { <<type JEP30-D10:EmptyType>> } class P_gate { <<type JEP30-D10:EmptyType>> } class SearchableProperties TriodeThyristorClassificationType "1" -- "1" Property_Array Property_Array "1" -- "1" TriodeThyristorClassificationPropertyArrayType TriodeThyristorClassificationPropertyArrayType "*" -- "1" attributes Property_Array "1" -- "1" GateType GateType "*" -- "1" N_gate GateType "*" -- "1" P_gate Property_Array "*" -- "1" SearchableProperties </pre>
type	TriodeThyristorClassificationType , TriodeThyristorClassificationProperty-ArrayType , TriodeThyristorGateType , JEP30-D10:EmptyType .
group	SearchableProperties .

4.7.1.3.20.2. Thyristor Overflow Categories Group

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Thyristor
diagram	<pre> classDiagram class ThyristorOverflowCategoriesType class Other { <<type ThyristorOverflowCategoriesType>> } class Sub_UserName { <<type xs:string>> } class Property_Array { <<type ThyristorOverflowCategoriesProperty-ArrayType>> } class ThyristorOverflowCategoriesPropertyArrayType { <<type ThyristorOverflowCategoriesProperty-ArrayType>> <<attributes>> } class SearchableProperties ThyristorOverflowCategoriesType "1" -- "1" Other Other "1" -- "1" Sub_UserName Other "1" -- "1" Property_Array Property_Array "1" -- "1" ThyristorOverflowCategoriesPropertyArrayType ThyristorOverflowCategoriesPropertyArrayType "*" -- "1" attributes ThyristorOverflowCategoriesPropertyArrayType "*" -- "1" SearchableProperties class Parts { <<type OverflowCategoriesType>> } class Accessories { <<type OverflowCategoriesType>> } class Other { <<type OverflowCategoriesType>> } </pre>
type	ThyristorOverflowCategoriesType , ThyristorOverflowCategoriesProperty-ArrayType , JEP30-D10:EmptyType , TriodeThyristorGateType , OverflowCategoriesType ,
group	ThyristorOverflowCategories , SearchableProperties .

4.7.1.3.20.3. Thyristor Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Thyristor/Property-Array
diagram	<pre> classDiagram class ThyristorClassificationPropertyArrayType { <<attributes>> <<Direction : ThyristorDirectionType>> <<Conduction : ThyristorConductionType>> <<SearchableProperties>> } class ThyristorDirectionType { <<Bidirectional : JEP30-D10:EmptyType>> <<UnidirectionalThyristor : JEP30-D10:EmptyType>> } class ThyristorConductionType { <<Reverse-Blocking : JEP30-D10:EmptyType>> <<Reverse-Conducting : JEP30-D10:EmptyType>> } class Bidirectional { type JEP30-D10:EmptyType } class UnidirectionalThyristor { type JEP30-D10:EmptyType } class ReverseBlocking { type JEP30-D10:EmptyType } class ReverseConducting { type JEP30-D10:EmptyType } class SearchableProperties </pre>
type	ThyristorClassificationProperty-ArrayType , ThyristorDirectionType , ThyristorConductionType , JEP30-D10:EmptyType .
group	SearchableProperties .

The *Bidirectional* thyristor is where two separate thyristors are integrated into the same device.

4.7.1.3.21. Transformer Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transformer
diagram	<pre> classDiagram class TransformerClassificationType { <<Transformer : TransformerClassificationType>> <<Signal : SignalTransformerClassificationType>> <<PowerTransformerClassificationType>> <<OverflowCategories>> <<Property-Array : TransformerClassificationProperty-ArrayType>> } class Transformer class Signal { type SignalTransformerClassificationType } class PowerTransformerClassificationType class OverflowCategories { <<Other : OverflowCategoriesType>> <<Parts : OverflowCategoriesType>> <<Accessories : OverflowCategoriesType>> } class Other { type OverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } class TransformerClassificationPropertyArrayType </pre>
type	TransformerClassificationType , SignalTransformerClassificationType , PowerTransformerClassificationType , OverflowCategoriesType , TransformerClassificationProperty-ArrayType .
group	OverflowCategories .

4.7.1.3.21. Transformer Classification (cont'd)

A transformer is an electrical device that transfers electrical energy between two or more circuits through electromagnetic induction. A varying current in one coil of the transformer produces a varying magnetic field, which induces a voltage in a second coil. Power is transferred between the two coils through the magnetic field, without a connection between the two circuits. Typically, there are two types of transformers – Signal and Power, but other types can be defined.

4.7.1.3.22. Transistor Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor</code>
diagram	<pre> classDiagram class Transistor { type TransistorClassificationType } class TransistorClassificationType { type TransistorClassificationProperty-ArrayType } class BipolarJunctionTransistor { type BJT-ClassificationType } class UnijunctionTransistor { type UJT-ClassificationType } class ProgrammableUnijunctionTransistors { type UJT-ClassificationType } class FieldEffectTransistor { type FET-ClassificationType } class Insulated-GateBipolarTransistor { type IGBT-ClassificationType } class TransistorOverflowCategories { type TransistorOverflowCategoriesType } class Other { type TransistorOverflowCategoriesType } class Parts { type OverflowCategoriesType } class Accessories { type OverflowCategoriesType } Transistor --> TransistorClassificationType TransistorClassificationType --> BipolarJunctionTransistor TransistorClassificationType --> UnijunctionTransistor TransistorClassificationType --> ProgrammableUnijunctionTransistors TransistorClassificationType --> FieldEffectTransistor TransistorClassificationType --> Insulated-GateBipolarTransistor TransistorClassificationType --> TransistorOverflowCategories TransistorOverflowCategories --> Other TransistorOverflowCategories --> Parts TransistorOverflowCategories --> Accessories </pre>
type	<code>TransistorClassificationType, BJT-ClassificationType, UJT-ClassificationType, FET-ClassificationType, IGBT-ClassificationType, TransistorOverflowCategoriesType, OverflowCategoriesType, TransistorClassificationProperty-ArrayType.</code>
group	<code>TransistorOverflowCategories.</code>

A *Transistor* is an electronic device that controls the flow of an electric current, most often used as an amplifier or switch. Transistors usually consist of three layers of semiconductor material, in which the flow of electric current across the outer layer is regulated by the voltage or current applied at the middle layer.

There are several classifications of transistors as shown above, however new classifications can be captured under the category *Other*.

4.7.1.3.22.1. Bipolar Junction Transistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/BipolarJunctionTransistor
diagram	<pre> classDiagram class BipolarJunctionTransistor { <<BJT-ClassificationType>> } class BJT-ClassificationType class BJT-ClassificationPropertyArrayType { <<attributes>> class Array { <<JEP30-D10:EmptyType>> } } class BJT-Type class NPN-PNP class NPN class PNP class DarlingtonNPN class DarlingtonPNP class Other class BJT-Material.PropertyType class GaAs class GaN class Other class SearchableProperties BipolarJunctionTransistor < -- BJT-ClassificationType BipolarJunctionTransistor < -- BJT-ClassificationPropertyArrayType BJT-ClassificationPropertyArrayType < -- Array BJT-ClassificationPropertyArrayType < -- BJT-Type BJT-Type < -- NPN-PNP BJT-Type < -- NPN BJT-Type < -- PNP BJT-Type < -- DarlingtonNPN BJT-Type < -- DarlingtonPNP BJT-Type < -- Other BJT-Type < -- BJT-Material.PropertyType BJT-Material.PropertyType < -- GaAs BJT-Material.PropertyType < -- GaN BJT-Material.PropertyType < -- Other BJT-ClassificationPropertyArrayType < -- SearchableProperties </pre> <p>The diagram illustrates the classification and property-array of a Bipolar Junction Transistor (BJT). It starts with a BipolarJunctionTransistor class that inherits from BJT-ClassificationType and BJT-ClassificationPropertyArrayType. The BJT-ClassificationPropertyArrayType contains an Array of type JEP30-D10:EmptyType, which points to a BJT-Type class. The BJT-Type class includes categories for NPN-PNP, NPN, PNP, DarlingtonNPN, DarlingtonPNP, and Other. It also points to a BJT-Material.PropertyType class, which includes GaAs, GaN, and another Other category. A SearchableProperties class is also associated with the BJT-ClassificationPropertyArrayType.</p>
type	BJT-ClassificationType , BJT-ClassificationPropertyArrayType , JEP30-D10:EmptyType , BJT-Type , BJT-Material.PropertyType .
group	SearchableProperties .

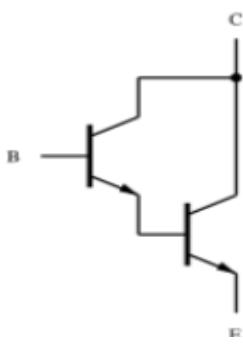


Figure 4—NPN Darlington

A *BipolarJunctionTransistor* (bipolar transistor or BJT) is a type of transistor that uses both electron and hole charge carriers. BJTs are manufactured in two types, NPN and PNP, and are available as individual components, combined or fabricated in integrated circuits, often in large numbers. The basic function of a BJT is to amplify current. This allows BJTs to be used as amplifiers or switches, giving them wide applicability in electronic.

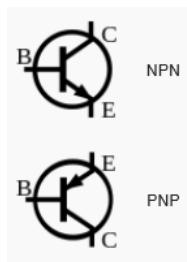


Figure 5 — BJT

4.7.1.3.22.1 Bipolar Junction Transistor Classification and Property–Array (cont'd)

The Darlington transistor is a compound structure of a particular design made by two bipolar transistors connected in such a way that the current amplified by the first transistor is amplified further by the second one. This configuration gives a much higher current gain than each transistor taken separately.

Other classifications can be captured under the category *Other*.

4.7.1.3.22.2. Unijunction Transistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/UnijunctionTransistor
diagram	<pre> classDiagram class UJT-ClassificationType class UnijunctionTransistor { <<UJT-ClassificationType>> } class Property-Array { <<UJT-ClassificationProperty-ArrayType>> } class UJT-ClassificationProperty-ArrayType { <<attributes>> } class UJT-Type { <<UJT-Type>> } class SearchableProperties { <<+>> } class N-emitter { <<JEP30-D10:EmptyType>> } class P-emitter { <<JEP30-D10:EmptyType>> } UnijunctionTransistor --> Property-Array Property-Array --> UJT-ClassificationProperty-ArrayType UJT-ClassificationProperty-ArrayType --> attributes UJT-ClassificationProperty-ArrayType --> UJT-Type UJT-Type --> N-emitter UJT-Type --> P-emitter UJT-Type --> SearchableProperties </pre>
type	UJT-ClassificationType , UJT-ClassificationProperty-ArrayType , UJT-Type , JEP30-D10:EmptyType .
group	SearchableProperties .

A *Unijunction Transistor* (UJT) is a three-terminal electronic semiconductor device with only one junction that acts exclusively as an electrically controlled switch.

A P-type UJT is where the base is formed by a lightly doped n-type bar of silicon. Two ohmic contacts B1 and B2 are attached at its ends. The emitter is of p-type and is heavily doped; this single PN junction gives the device its name. A complementary N-Type UJT uses a p-type base and an n-type emitter and operates the same as the n-type base device but with all voltage polarities reversed.

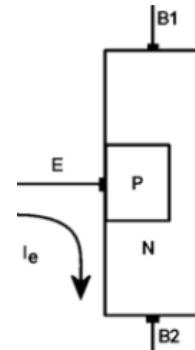


Figure 6 — p-type UJT

4.7.1.3.22.3. Programmable Unijunction Transistor Classification and Property-Array

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/UnijunctionTransistor</code>
diagram	<pre> classDiagram class ProgrammableUnijunctionTransistors { <<UJT-ClassificationType>> } class Property_Array { <<UJT-ClassificationProperty-ArrayType>> } class UJT_Classification_Property_Array_Type { <<UJT-ClassificationProperty-ArrayType>> } class attributes { <<UJT-ClassificationProperty-ArrayType>> } class UJT_Type { <<UJT-Type>> } class SearchableProperties { <<SearchableProperties>> } class N_emitter { <<JEP30-D10:EmptyType>> } class P_emitter { <<JEP30-D10:EmptyType>> } ProgrammableUnijunctionTransistors "1" -- "*" UJT_Classification_Property_Array_Type : <<UJT-ClassificationType>> ProgrammableUnijunctionTransistors "*" -- "*" Property_Array : <<UJT-ClassificationProperty-ArrayType>> UJT_Classification_Property_Array_Type "*" -- "*" attributes : <<UJT-ClassificationProperty-ArrayType>> UJT_Classification_Property_Array_Type "*" -- "*" UJT_Type : <<UJT-Type>> UJT_Type "*" -- "*" SearchableProperties : <<SearchableProperties>> UJT_Type "*" -- "*" N_emitter : <<JEP30-D10:EmptyType>> UJT_Type "*" -- "*" P_emitter : <<JEP30-D10:EmptyType>> </pre> <p>The diagram illustrates the classification and property-array for a Programmable Unijunction Transistor (PUT). It shows the inheritance relationship from <code>ProgrammableUnijunctionTransistors</code> to <code>UJT-ClassificationProperty-ArrayType</code>, and the aggregation relationship from <code>UJT-ClassificationProperty-ArrayType</code> to <code>attributes</code>, <code>UJT-Type</code>, and <code>SearchableProperties</code>. Additionally, it shows the aggregation relationship from <code>UJT-Type</code> to <code>N-emitter</code> and <code>P-emitter</code>.</p>
type	<code>UJT-ClassificationType</code> , <code>UJT-ClassificationProperty-ArrayType</code> , <code>UJT-Type</code> , <code>JEP30-D10:EmptyType</code> .
group	<code>SearchableProperties</code> .

The *ProgrammableUnijunctionTransistor*, or PUT, is a multi-junction device that, with two external resistors, displays similar characteristics to the UJT.

4.7.1.3.22.4. Field Effect Transistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/FieldEffectTransistor
diagram	<pre> classDiagram class FET-ClassificationType class FieldEffectTransistor { type FET-ClassificationType } class Property-Array { type FET-ClassificationProperty-ArrayType } class FET-ClassificationProperty-ArrayType { <<attributes>> } class Array { type JEP30-D10:EmptyType } class FET-Type { type FET-Type } class FET-Mode { type FET-ModeType } class Depletion-mode { type JEP30-D10:EmptyType } class Enhancement-mode { type JEP30-D10:EmptyType } class JunctionGate { type JEP30-D10:EmptyType } class Other { type xs:string } FET-ClassificationType "2..1" -- "1..1" FieldEffectTransistor : type FET-ClassificationType FET-ClassificationType "2..1" -- "1..1" Property-Array : type FET-ClassificationProperty-ArrayType Property-Array "1..1" -- "1..1" <<attributes>> : attributes FET-ClassificationProperty-ArrayType "1..1" -- "1..1" Array : type JEP30-D10:EmptyType FET-ClassificationProperty-ArrayType "1..1" -- "1..1" FET-Type : type FET-Type FET-ClassificationProperty-ArrayType "1..1" -- "1..1" FET-Mode : type FET-ModeType FET-Type "1..1" -- "1..1" N-channelAndP-channel : type JEP30-D10:EmptyType FET-Type "1..1" -- "1..1" N-channel : type JEP30-D10:EmptyType FET-Type "1..1" -- "1..1" P-channel : type JEP30-D10:EmptyType FET-Mode "1..1" -- "1..1" Depletion-mode : type JEP30-D10:EmptyType FET-Mode "1..1" -- "1..1" Enhancement-mode : type JEP30-D10:EmptyType FET-Mode "1..1" -- "1..1" JunctionGate : type JEP30-D10:EmptyType FET-Mode "1..1" -- "1..1" Other : type xs:string </pre>
type	FET-ClassificationType, FET-ClassificationProperty-ArrayType, JEP30-D10:EmptyType, FET-Type, FET-ModeType.
group	SearchableProperties.

The *FieldEffectTransistor* (FET) is a transistor that uses an electric field to control the electrical behaviour of the device. FET's are also known as unipolar transistors since they involve single-carrier-type operation. Many different implementations of field effect transistors exist, as shown above, and other classifications can be captured under the category *Other*.

FET's generally display very high input impedance at low frequencies. The conductivity between the drain and source terminals is controlled by an electric field in the device, which is generated by the voltage difference between the body and the gate of the device.

Enhancement-mode devices are OFF at zero gate–source voltage, and can be turned on by pulling the gate voltage either higher than the source voltage, for NMOS, or lower than the source voltage, for PMOS. In most circuits, this means pulling an enhancement-mode MOSFET's gate voltage towards its drain voltage turns it ON.

A *Depletion-mode* device is normally ON at zero gate–source voltage.

4.7.1.3.22.5. Insulated Gate Bipolar Transistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/Insulated-GateBipolarTransistor
diagram	<pre> classDiagram class Insulated-GateBipolarTransistor { <<IGBT-ClassificationType>> } class Property-Array { <<IGBT-ClassificationProperty-ArrayType>> attribute Array : JEP30-D10:EmptyType } class IGBT-Type { <<IGBT-Type>> } class N-channelAndP-channel { <<JEP30-D10:EmptyType>> } class N-channel { <<JEP30-D10:EmptyType>> } class P-channel { <<JEP30-D10:EmptyType>> } class IGBT-Mode { <<IGBT-ModeType>> } class Depletion-mode { <<JEP30-D10:EmptyType>> } class Enhancement-mode { <<JEP30-D10:EmptyType>> } class Other { <<xs:string>> } Insulated-GateBipolarTransistor "1" --> "1" Property-Array Property-Array "1" --> "1" Array : JEP30-D10:EmptyType Array --> "1" IGBT-Type IGBT-Type --> "1" N-channelAndP-channel IGBT-Type --> "1" N-channel IGBT-Type --> "1" P-channel Property-Array --> "1" IGBT-Mode IGBT-Mode --> "1" Depletion-mode IGBT-Mode --> "1" Enhancement-mode IGBT-Mode --> "1" Other </pre>
type	IGBT-ClassificationType , IGBT-ClassificationProperty-ArrayType , JEP30-D10:EmptyType , IGBT-Type , IGBT-ModeType .
group	SearchableProperties .

An [Insulated-GateBipolarTransistor](#) (IGBT) is a three-terminal power semiconductor device primarily used as an electronic switch which combine high efficiency and fast switching. The IGBT is a semiconductor device with four alternating layers (P-N-P-N) that are controlled by a metal-oxide-semiconductor (MOS) gate structure without regenerative action.

These devices can also be provided in array form. Other IGBT classifications can be captured under the category [Other](#).

4.7.1.3.22.6. Transistor Overflow Categories Type

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/Other
diagram	<pre> classDiagram class TransistorOverflowCategoriesType { "Sub-CatagoryName" : xs:string "Other" : TransistorOverflowCategoriesType } class TransistorOverflowCategoriesProperty-ArrayType { "Property-Array" : TransistorOverflowCategoriesProperty-ArrayType } class TransistorOverflowCategoriesFunctionalTypeType { "Mode" : ModeType "FunctionalType" : TransistorOverflowCategoriesFunctionalTypeType } class ModeType { "Depletion-mode" : JEP30-D10:EmptyType "Enhancement-mode" : JEP30-D10:EmptyType "JunctionGate" : JEP30-D10:EmptyType "Other" : xs:string } class BJT-MaterialPropertyType { "Material" : BJT-MaterialPropertyType } class SearchableProperties TransistorOverflowCategoriesType < -- TransistorOverflowCategoriesProperty-ArrayType TransistorOverflowCategoriesType < -- TransistorOverflowCategoriesFunctionalTypeType TransistorOverflowCategoriesType < -- BJT-MaterialPropertyType TransistorOverflowCategoriesFunctionalTypeType < -- ModeType TransistorOverflowCategoriesFunctionalTypeType < -- TransistorOverflowCategoriesFunctionalTypeType BJT-MaterialPropertyType < -- SearchableProperties </pre>
type	TransistorOverflowCategoriesType , TransistorOverflowCategoriesProperty-ArrayType , JEP30-D10:EmptyType , TransistorOverflowCategoriesFunctionalTypeType , ModeType , BJT-MaterialPropertyType .
group	TransistorOverflowCategories , SearchableProperties .

4.7.1.3.22.6.1. Functional Type

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/Other/TransistorOverflowCategoriesProperty-Array/FunctionalType
diagram	<pre> classDiagram class TransistorOverflowCategoriesFunctionalTypeType { N-emitter P-emitter N-channelAndP-channel N-channel P-channel NPN-PNP NPN PNP DarlingtonNPN DarlingtonPNP Other } class FunctionalType { <<TransistorOverflowCategoriesFunctionalTypeType>> } TransistorOverflowCategoriesFunctionalTypeType < -- FunctionalType </pre> <p>The diagram shows a UML class hierarchy. At the top is a yellow-shaded box containing several classes: N-emitter, P-emitter, N-channelAndP-channel, N-channel, P-channel, NPN-PNP, NPN, PNP, DarlingtonNPN, DarlingtonPNP, and Other. Below this yellow box is a dashed-line box labeled 'FunctionalType'. A directed association line connects the 'FunctionalType' box to the 'TransistorOverflowCategoriesFunctionalTypeType' box above it.</p>
type	TransistorOverflowCategoriesFunctionalTypeType, JEP30-D10:EmptyType.

4.7.1.3.23. Tube Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Tube
diagram	<pre> classDiagram class Tube { type: TubeClassificationType } class TubeClassificationType { } class GasDischarge { type: GasDischargeTubeClassificationType } class VacuumFluorescentDisplay { type: VacuumFluorescentDisplayTubeClassificationType } class Other { type: OverflowCategoriesType } class Parts { type: OverflowCategoriesType } class Accessories { type: OverflowCategoriesType } class PropertyArray { type: TubeClassificationPropertyArrayType } Tube "3" --> "3" TubeClassificationType TubeClassificationType "3" --> GasDischarge TubeClassificationType "3" --> VacuumFluorescentDisplay TubeClassificationType "3" --> Other Other "3" --> Parts Other "3" --> Accessories </pre>
type	TubeClassificationType , GasDischargeTubeClassificationType , VacuumFluorescentDisplayTubeClassificationType , OverflowCategoriesType , TubeClassificationProperty-ArrayType
group	OverflowCategories .

A gas or vapor filled tube that is used to conduct electricity when voltage is applied is called a [GasDischarge](#) Tube. Gas discharge tubes dissipate voltage transients through a contained plasma gas. They have high insulation resistance plus low capacitance and leakage to ensure minimal effect on normal operation of equipment.



Figure 7 — Gas Discharge Tube

A [VacuumFluorescentDisplay](#) (VFD) is a display device, sometimes named as ice tube indicator, operates on the principle of cathodoluminescence, roughly similar to a cathode ray tube, but operating at much lower voltages. Each tube in a VFD has a phosphor coated anode that is bombarded by electrons emitted from the cathode filament. In fact, each tube in VFD is a triode vacuum tube because it also has a mesh control grid.



Unlike liquid crystal displays, a VFD emits a very bright light with high contrast and can support display elements of various colors. Other Tube classifications can be captured under the category [Other](#).

Figure 8 — Vacuum Fluorescent Display

4.7.1.4. Hardware Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Hardware
diagram	<pre> classDiagram class HardwareClassificationType { <<HardwareAccessories>> <<CableHardware>> <<Clamp>> <<Clips>> <<EMI-Shield>> <<Faceplate>> <<Fasteners>> <<Gasket>> <<Heatsink>> <<InsulatorsMountsSpacers>> <<JackScrews>> <<MetalCage>> <<Nut>> <<PickAndPlaceHardware>> <<Screw>> <<SpringClassification>> <<Studs>> <<Washer>> <<Other>> <<Parts>> <<Accessories>> <<Property-Array>> } class Hardware { <<HardwareClassificationType>> } Hardware --> HardwareClassificationType HardwareClassificationType < -- HardwareAccessories HardwareClassificationType < -- CableHardware HardwareClassificationType < -- Clamp HardwareClassificationType < -- Clips HardwareClassificationType < -- EMI-Shield HardwareClassificationType < -- Faceplate HardwareClassificationType < -- Fasteners HardwareClassificationType < -- Gasket HardwareClassificationType < -- Heatsink HardwareClassificationType < -- InsulatorsMountsSpacers HardwareClassificationType < -- JackScrews HardwareClassificationType < -- MetalCage HardwareClassificationType < -- Nut HardwareClassificationType < -- PickAndPlaceHardware HardwareClassificationType < -- Screw HardwareClassificationType < -- SpringClassification HardwareClassificationType < -- Studs HardwareClassificationType < -- Washer HardwareClassificationType < -- Other HardwareClassificationType < -- Parts HardwareClassificationType < -- Accessories HardwareClassificationType < -- Property-Array </pre> <p>The diagram illustrates the UML class structure for hardware classification. It features a main class, <code>HardwareClassificationType</code>, which contains several subclasses representing different hardware components: <code>BatteryAccessories</code>, <code>CableHardware</code>, <code>Clamp</code>, <code>Clips</code>, <code>EMI-Shield</code>, <code>Faceplate</code>, <code>Fasteners</code>, <code>Gasket</code>, <code>Heatsink</code>, <code>InsulatorsMountsSpacers</code>, <code>JackScrews</code>, <code>MetalCage</code>, <code>Nut</code>, <code>PickAndPlaceHardware</code>, <code>Screw</code>, <code>SpringClassification</code>, <code>Studs</code>, <code>Washer</code>, <code>Other</code>, <code>Parts</code>, <code>Accessories</code>, and <code>Property-Array</code>. Additionally, there is a class named <code>Hardware</code> that has a directed association with <code>HardwareClassificationType</code>.</p>

4.7.1.4. Hardware Classification (cont'd)

type	<code>HardwareClassificationType, BatteryAccessoriesClassificationType, CableHardwareClassificationType, HardwareClampClassificationType, HardwareClipsClassificationType, EMI-ShieldClassificationType, FaceplateClassificationType, HardwareFastenersClassificationType, GasketClassificationType, HeatsinkClassificationType, HardwareInsulatorsMountsSpacersClassificationType, JackScrewsClassificationType, MetalCageClassificationType, HardwareNutClassificationType, PickAndPlaceHardwareClassificationType, HardwareScrewClassificationType, HardwareSpringClassificationType, StudsClassificationType, HardwareWasherClassificationType, OverflowCategoriesType, HardwareClassificationProperty-ArrayType.</code>
group	<code>OverflowCategories.</code>

4.7.1.4.1. MetalCage Classification

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Hardware/MetalCage</code>
diagram	<pre> classDiagram class MetalCage { <<MetalCageClassificationType>> } class Property_Array { <<MetalCageClassificationProperty-ArrayType>> } class MetalCageClassificationType { <<MetalCageClassificationType>> } class MetalCageClassificationPropertyArrayType { <<MetalCageClassificationProperty-ArrayType>> } class EMI { <<JEP30-D10:EmptyType>> } class SearchableProperties MetalCage --> Property_Array : Property_Array --> MetalCageClassificationType : MetalCageClassificationType --> MetalCageClassificationPropertyArrayType : MetalCageClassificationPropertyArrayType --> attributes : attributes --> EMI : EMI --> SearchableProperties : </pre> <p>The diagram shows a UML class hierarchy for MetalCage classification. It starts with a 'MetalCage' class (type: MetalCageClassificationType) which has a directed association to a 'Property-Array' class (type: MetalCageClassificationProperty-ArrayType). This association is multiplicity 1..1 on both sides. The 'Property-Array' class then has a directed association to a 'MetalCageClassificationType' class (type: MetalCageClassificationType), also with multiplicity 1..1. From there, it goes to a 'MetalCageClassificationProperty-ArrayType' class (type: MetalCageClassificationProperty-ArrayType), which has an association to an 'attributes' class. Finally, an 'EMI' class (type: JEP30-D10:EmptyType) is associated with the 'attributes' class, and it has an association to a 'SearchableProperties' class.</p>
type	<code>MetalCageClassificationType, MetalCageClassificationProperty-ArrayType, JEP30-D10:EmptyType.</code>
group	<code>SearchableProperties</code>

4.7.1.5. Optics Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Optics
diagram	<pre> classDiagram class OpticsClassificationType { <<OpticsClassificationType>> <<OpticalAmplifierClassificationType>> <<OpticalAttenuatorClassificationType>> <<OpticalCirculatorClassificationType>> <<OpticalCouplerClassificationType>> <<OpticalDemultiplexersClassificationType>> <<OpticalReceiverClassificationType>> <<OpticalSwitchClassificationType>> <<OpticalTransceiversClassificationType>> <<OpticalTransmittersClassificationType>> <<OverflowCategoriesType>> <<OpticClassificationProperty-ArrayType>> } class Options { <<OpticsClassificationType>> } Options < -- OpticsClassificationType Options < -- OpticalAmplifierClassificationType Options < -- OpticalAttenuatorClassificationType Options < -- OpticalCirculatorClassificationType Options < -- OpticalCouplerClassificationType Options < -- OpticalDemultiplexersClassificationType Options < -- OpticalReceiverClassificationType Options < -- OpticalSwitchClassificationType Options < -- OpticalTransceiversClassificationType Options < -- OpticalTransmittersClassificationType Options < -- OverflowCategoriesType Options < -- OpticClassificationProperty-ArrayType Options < -- Other Options < -- Parts Options < -- Accessories </pre> <p>The diagram illustrates the classification of optics components. It starts with a general class 'Optics' which is a specialization of 'OpticsClassificationType'. This generalization relationship is shown with a dashed line connecting 'Optics' to 'OpticsClassificationType'. The 'OpticsClassificationType' class contains several specific types: 'Amplifier', 'Attenuator', 'Circulator', 'Coupler', 'Demultiplexers', 'Receiver', 'Switch', 'Transceivers', and 'Transmitters'. Below these, there is a group of three categories: 'Other', 'Parts', and 'Accessories', all of which are also specializations of 'OpticsClassificationType'. Additionally, there is a 'OverflowCategories' class and a 'Property-Array' class, both of which are also specializations of 'OpticsClassificationType'. The 'Optics' class also has associations with 'OverflowCategories' and 'Property-Array'.</p>
type	OpticsClassificationType , OpticalAmplifierClassificationType , OpticalAttenuatorClassificationType , OpticalCirculatorClassificationType , OpticalCouplerClassificationType , OpticalDemultiplexersClassificationType , OpticalReceiverClassificationType , OpticalSwitchClassificationType , OpticalTransceiversClassificationType , OpticalTransmittersClassificationType , OverflowCategoriesType , OpticClassificationProperty-ArrayType .
group	OverflowCategories .

4.7.2. Terminal Details - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array
diagram	<pre> classDiagram class TerminalDetails-ArrayType { <<TerminalDetails>> <<constraints>> } class TerminalDetailsType { <<ID : xs:string>> <<Properties-Array : Properties-ArrayType>> <<TerminalFunction-Array : TerminalFunction-ArrayType>> <<TerminalGrouping : TerminalGroupingType>> <<ExternalConnection-Array : ExternalConnection-ArrayType>> } TerminalDetails-ArrayType "1..S" --> TerminalDetailsType constraints "constraints" </pre>
type	TerminalDetails-ArrayType, TerminalDetailsType, Properties-ArrayType, TerminalFunction-ArrayType, TerminalGroupingType, ExternalConnection-ArrayType.

The [TerminalDetails-Array](#) section basically captures all the electrical detail associate with the Terminal, or group of terminals.

Some of these details may reference other arrays that are shared with both the Part and the Terminal since their structure is the same. These will be linked via ID's and will be described in more detail following sections, whenever applicable.

4.7.2.1. Properties - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array
diagram	<p>The diagram shows the UML class Properties-ArrayType (highlighted in yellow) which contains a list of Properties (highlighted in green). The Properties class has a multiplicity of 1..* and is associated with the Properties-ArrayType via a directed association. The Properties class also has a directed association to a PropertiesType class (highlighted in blue). The PropertiesType class contains several properties, each with a type of JEP30-D10:EmptyType. These properties include: ID, Digital, Analog, Power, Reserved, NoDisConnection, Ground, ActiveLow, ActiveHigh, EdgeTriggered, Amplifier, Clamp, Hysteresis, Inversion, SchmittTriggered, TriState, Direction, InternalPullupPulldown, SeriesComponent, OutputCircuit, and Reference.</p>
type	Properties-ArrayType , PropertiesType , JEP30-D10:EmptyType , AnalogConnectionType , PowerConnectionType , SignalDirectionType , SignallInternalPullUp-DownType , SignalSeriesComponentType , OutputCircuitPropertyType , SignalReferenceType

4.7.2.1 Properties – Array (cont'd)

The *Properties-Array* section captures a set of electrical *Properties* which are then assigned to their respective *TerminalName* or *InternalNode* via an *ID* as shown in the following section. The *Properties-Array* contains the following properties:-

Signal Type is defined as one of the following types.

1. *Digital*,
 - a. Reference JESD99 for “Signal, digital” definition.
2. *Analog*,
 - a. Reference JESD99 for “Signal, analog” definition.
3. *Power*,
4. *Reserved* – This terminal should not be used. It is usually required by the part manufacturer for their internal processing requirements, such as part verification or part configuration.
5. *NoDieConnection* is where a terminal has no internal connection, as shown in Figure 9.

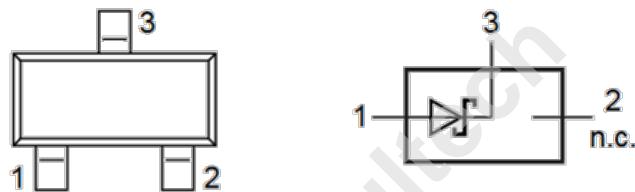


Figure 9 — Sample Device with no internal die connection

6. *Ground*.

When a signal is referred to be *ActiveLow* in a digital circuit, it signifies that the signal will execute its function when the logic level of the signal is between 0 to 0.7 V. It is necessary to “pull” the terminal LOW if it is an active-low terminal by connecting it to ground.

ActiveHigh refers to voltage levels between 3.3V to 5V. Usually in digital circuits, the active high terminal is pulled to VCC.

EdgeTriggered is a type of triggering that allows a circuit to become active at the positive edge or the negative edge of the clock signal.

Amplifier boosts the input or output signal depending on the direction of the signal.

Clamp is a circuit that prevents the terminal signal from exceeding the clamp voltage (usually the supply voltage of the Part), by typically 0.4 to 0.7 V (which is the forward voltage drop of the diode), or from dropping below the lower clamp voltage (usually the negative supply or Ground), by again typically 0.4 to 0.7 V.

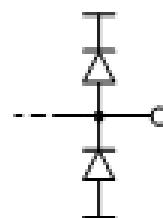


Figure 10 — Clamp Circuit

4.7.2.1 Properties – Array (cont'd)

Hysteresis is the dependence of the state of a system on its history. Plots of a single component of the moment often form a loop or hysteresis curve, where there are different values of one variable depending on the direction of change of another variable. It prevents unwanted frequent switching in Schmitt triggered devices. Hysteresis can be a dynamic lag between an input and an output that disappears if the input is varied more slowly; this is known as rate-dependent hysteresis. In control systems, hysteresis can be used to filter signals so that the output reacts less rapidly than it otherwise would, by taking recent history into account.

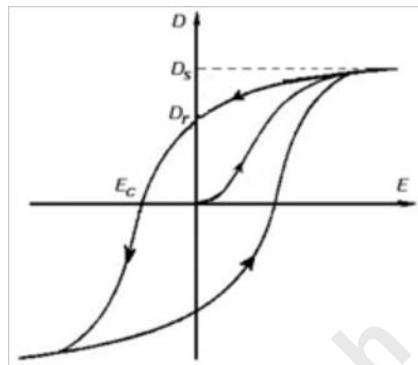


Figure 11 — Hysteresis Loop Curve

Often, some amount of hysteresis is intentionally added to an electronic circuit to prevent unwanted rapid switching. This and similar techniques are used to compensate for contact bounce in switches, or noise in an electrical signal. **Schmitt Triggered** is a simple electronic circuit that exhibits this property. Schmitt trigger is a comparator circuit with hysteresis implemented by applying positive feedback to the non-inverting input of a comparator or differential amplifier. It is an active circuit which converts an analog input signal to a digital output signal. The circuit is named a "trigger" because the output retains its value until the input changes sufficiently to trigger a change. Figure 12 shows a typical response of a Schmitt Trigger with a much sharper transition than in a normal hysteresis circuit. The horizontal and vertical axes are input voltage and output voltage, respectively. T and $-T$ are the switching thresholds, and M and $-M$ are the output voltage levels.

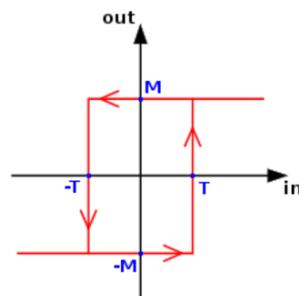


Figure 12 — Schmitt Trigger

4.7.2.1 Properties – Array (cont'd)

Inversion is where the signal is electrically inverted from positive peak to negative peak but does not translate the signal to any other form, as shown in the attached wave form. A phase inversion is neither a time shift nor a phase shift, but simply a swap of plus and minus. In digital logic, an inverter implements logical negation, similar to an Inverter or a NOT gate. So a Logic “0” input is considered true and a logic “1” input is considered false.

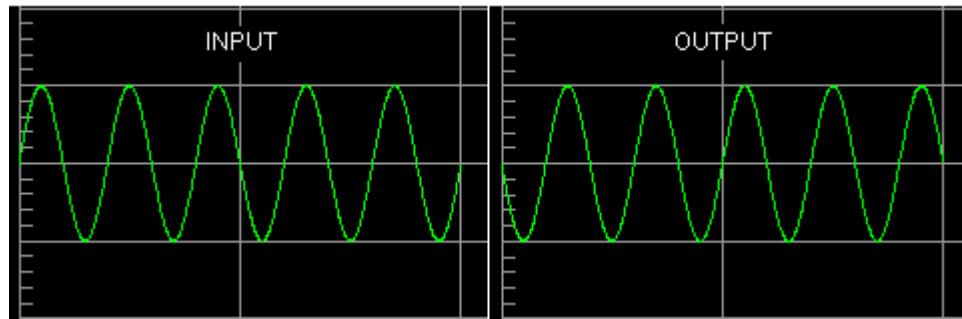


Figure 13 — Signal Inversion

An electronic output stage consisting of a logic gate, commonly an inverter or buffer, that exhibits three possible logic states, namely logic 1, logic 0, and an inactive (high-impedance or open-circuit) state, effectively removing the output from the circuit. This allows multiple circuits to share the same output line or lines (such as a bus which cannot listen to more than one device at a time). A *TriState* buffer can be thought of as a switch. If B is on, the switch is closed. If B is off, the switch is open.



Figure 14 — Tri-State

4.7.2.1.1. Linking the Terminal Map and the Internal Node to the Property ID

path	<pre>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails- Array/TerminalDetails/ElectricalMap-Array/ElectricalMap/PropertyID -> PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails- Array/TerminalDetails/Properties-Array/Properties/ID</pre>
diagram	<p>The diagram illustrates the UML Class Diagram for the specified XSD schema. It shows the following classes and their associations:</p> <ul style="list-style-type: none"> TerminalDetailsType: Contains ElectricalMap-ArrayType. ElectricalMap-ArrayType: Contains ElectricalMap. ElectricalMap: Has a multiplicity of 1..* and connects to ElectricalMapType. ElectricalMapType: Contains Terminal, PropertyID, TerminalGroupDescription, TerminalFunctionID, and ElectricalSpecificationID. Properties-ArrayType: Contains Properties. Properties: Has a multiplicity of 1..* and connects to PropertiesType. PropertiesType: Contains ID, ActiveLow, ActiveHigh, EdgeTriggered, Amplifier, Clamp, Hysteresis, and Inversion. TerminalDetails and Properties-Array both connect to Properties. <p>A red arrow highlights the association between PropertyID in ElectricalMapType and ID in PropertiesType, indicating the specific connection being analyzed.</p>
type	TerminalDetailsType , ElectricalMap-ArrayType , ElectricalMapType , Properties-ArrayType , PropertiesType .

The **PropertyID** under **ElectricalMap** connect to the **Properties/ID** under the **Properties-Type**, thereby connecting the set of electrical **Properties** identified under a specific ID back to the **Terminal**.

4.7.2.1.2. Analog and Power Connection Type

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/Analog. 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/PowerConnectionType
diagram	<pre> classDiagram class PropertiesType class Analog { <<AnalogConnectionType>> } class Power { <<PowerConnectionType>> } class AC class DC class ACDC class Load class Source PropertiesType < -- Analog PropertiesType < -- Power Analog < -- AnalogConnectionType Power < -- PowerConnectionType AnalogConnectionType < -- AC AnalogConnectionType < -- DC AnalogConnectionType < -- ACDC PowerConnectionType < -- AC PowerConnectionType < -- DC PowerConnectionType < -- Load PowerConnectionType < -- Source </pre>
type	AnalogConnectionType, PowerConnectionType.

The Analog signal can be one of

1. **AC**,
2. **DC**, or
3. **ACDC** represents an AC signal superimposed on a DC signal as shown in Figure 17.

The power connection can be either AC or DC and in addition be either a

1. **Load**, or
 - a. The total power consumed by the device.
2. **Source**.

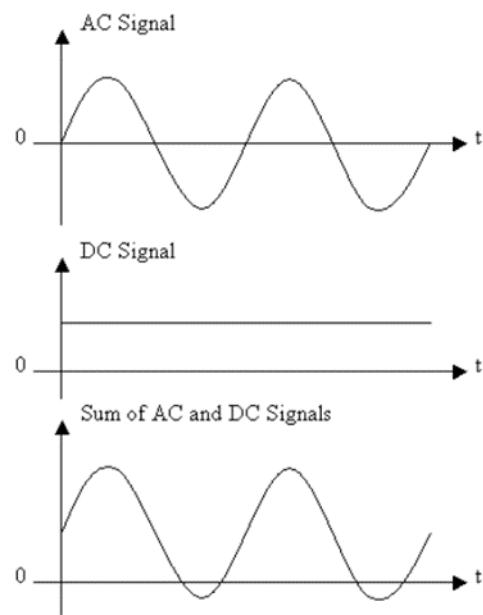


Figure 15 — Analog Connection Types

4.7.2.1.3. Signal Direction

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/Direction</code>
diagram	<pre> classDiagram class SignalDirectionType { <<SignalDirectionType>> <<SignalPropertyBidirectionalType>> <<BothDirectionsSimultaneously>> <<OnlyOneDirection-at-a-time>> <<Input>> <<Output>> <<Programmable>> <<Direction>> } class Bidirectional { <<SignalPropertyBidirectionalType>> } class BothDirectionsSimultaneously { <<JEP30-D10:EmptyType>> } class OnlyOneDirection-at-a-time { <<JEP30-D10:EmptyType>> } class Input { <<JEP30-D10:EmptyType>> } class Output { <<JEP30-D10:EmptyType>> } class Programmable { <<JEP30-D10:EmptyType>> } class Direction { <<SignalDirectionType>> } SignalDirectionType < -- Bidirectional SignalDirectionType < -- BothDirectionsSimultaneously SignalDirectionType < -- OnlyOneDirection-at-a-time SignalDirectionType < -- Input SignalDirectionType < -- Output SignalDirectionType < -- Programmable SignalDirectionType < -- Direction </pre> <p>The diagram illustrates the UML class structure for signal direction. It starts with a general class <code>SignalDirectionType</code> which branches into several specific types: <code>Bidirectional</code>, <code>BothDirectionsSimultaneously</code>, <code>OnlyOneDirection-at-a-time</code>, <code>Input</code>, <code>Output</code>, <code>Programmable</code>, and <code>Direction</code>. The <code>Bidirectional</code> class is associated with the <code>SignalPropertyBidirectionalType</code>. The <code>BothDirectionsSimultaneously</code> and <code>OnlyOneDirection-at-a-time</code> classes are both associated with the type <code>JEP30-D10:EmptyType</code>.</p>
type	<code>SignalDirectionType, SignalPropertyBidirectionalType, JEP30-D10:EmptyType.</code>

A **Bidirectional** communication system is a point-to-point system composed of two connected devices that can communicate with one another in both directions. There are two types of bidirectional communication:-

1. *BothDirectionsSimultaneously*, and
2. *OnlyOneDirection-at-a-time*.

If the signal being transmitted is a RF signal, then both devices can communicate with each other simultaneously.

If the signal being transmitted is a digital signal, even though each party can communicate with the other, they can't communicate simultaneously. The communication is one direction at a time.

Programmable direction is where the terminal can be configured to be either an output source of the signal or an input recipient of the signal. It is also possible for the terminal to be programmed to be a bidirectional signal.

4.7.2.1.4. Internal Pullup / Pulldown

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/InternalPullupPulldown</code>
diagram	<pre> classDiagram class SignalInternalPullupPulldownType { InternalPullupPulldownPropertyType ResistanceType CurrentType TerminalName } class InternalPullupPulldownPropertyType { <<Pullup>> <<Pulldown>> <<Programmable>> } class ResistanceType { Resistance ResistanceValueUOM } class CurrentType { Current CurrentUOM } class Resistance { type JEP30-D10:ValueSetType } class ResistanceValueUOM { type ResistanceUOMType } class Current { type JEP30-D10:ValueSetType } class CurrentUOM { type JEP30-D10:CurrentUOMType } </pre>
type	<code>SignalInternalPullUp–DownType, InternalPullUp–PullDownPropertyType, ResistanceType, ResistanceUOMType, CurrentType, JEP30-D10:CurrentUOMType, ProgrammableInternalPullUp–DownType.</code>

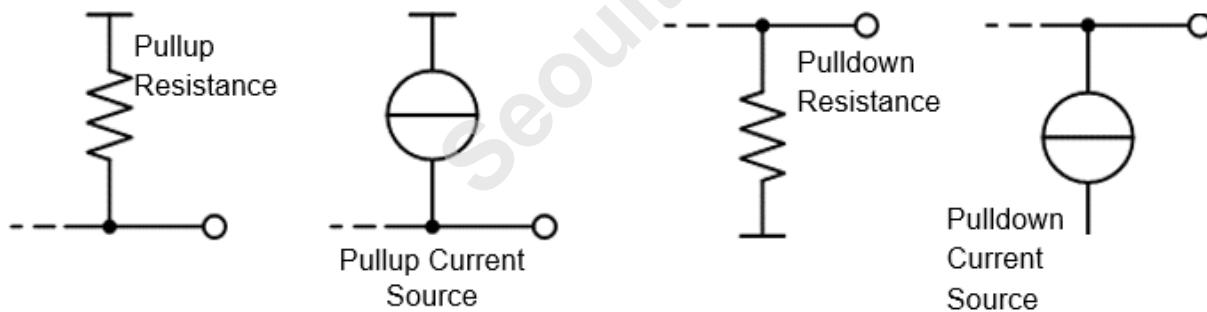


Figure 16 — Internal Pullup / Pulldown Circuits

In electronic logic circuits, a *Pullup* resistor is a resistor connected between a signal conductor and a positive power supply voltage to ensure that the signal will be a valid logic level if external devices are disconnected or high-impedance is introduced. They may also be used at the interface between two different types of logic devices, possibly operating at different logic levels and power supply voltages.

A *Pulldown* resistor works in the same way but is connected to ground. It holds the logic signal at a low logic level when no other active device is connected. An active *CurrentSource* instead of a *Resistance* element, may also be used as wide voltage range *Pullup* links within power supplies and other wide voltage range circuits. If ordinary resistors were used then the current would vary considerably over the voltage range.

4.7.2.1.5. Series Component

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/SeriesComponent
diagram	<pre> classDiagram class SignalSeriesComponentType { class Resistance { type ResistanceValueType } class Capacitance { type CapacitanceValueType } } class ResistanceValueType { class Resistance { type JEP30-D10:ValueSetType } class ResistanceValueUOM { type ResistanceAsciiUOMType } } class CapacitanceValueType { class Capacitance { type JEP30-D10:ValueSetType } class CapacitanceValueUOM { type CapacitanceUOMType } } class SeriesComponent { type SignalSeriesComponentType } </pre>
type	SignalSeriesComponentType , ResistanceValueType , JEP30-D10:ValueSetType , ResistanceUOMType , CapacitanceValueType , CapacitanceUOMType .

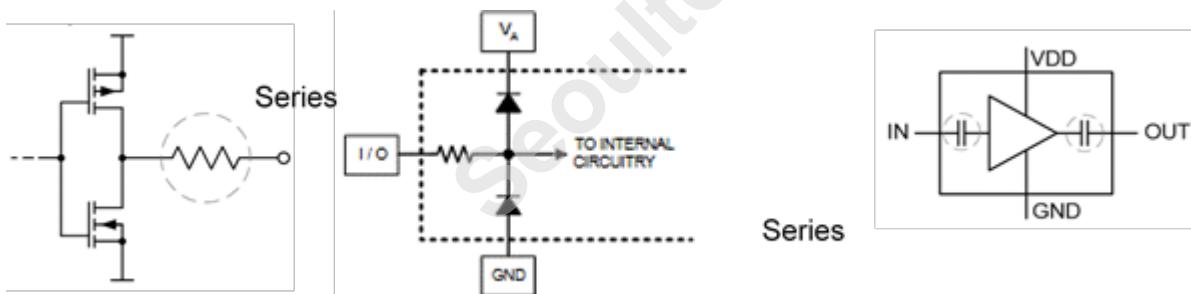


Figure 17 — Series Component Types

A series *Resistance* as a *SeriesComponent* on the output provides opposition to current flow to protect the load network being connected to it. Series termination is effective in reducing the driver's edge rate, and it consumes low power. Series termination provides good signal quality by damping overshoot and undershoot, and effectively reducing line noise and EMI. Its drawbacks are that it slows the signal's rise and fall time, and that it should not be used with distributed loads. When connected to an input, the series resistance protects the device itself for internal damage in the event of a too high input current or voltage.

The *Capacitance* as a *SeriesComponent* is to remove the DC component of the signal, thereby only allowing the AC component to pass through it.

4.7.2.1.6. Output Circuit

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit
diagram	<pre> classDiagram class OutputCircuitPropertyType { <<OutputCircuit>> } class Bipolar { <<BipolarOutputCircuitType>> } class Unipolar { <<UnipolarOutputCircuitType>> } class Programmable { <<ProgrammableOutputCircuitType>> } OutputCircuitPropertyType < -- Bipolar OutputCircuitPropertyType < -- Unipolar OutputCircuitPropertyType < -- Programmable </pre>
type	OutputCircuitPropertyType , BipolarOutputCircuitType , UnipolarOutputCircuitType , ProgrammableOutputCircuitType .

OutputCircuit described in this section can be categorized as either *Bipolar* or *Unipolar*. The *Programmable* output circuit is a combination of some of the outputs available in the *Bipolar* and *Unipolar* branches. Ref. JESD99 “Types of outputs”.

A *Bipolar* output is an output having internal connections through two active devices to two supply voltages so that, according to the relative states of the active devices, the output can source or sink current through the load. Ref. JESD99 “bipolar output”.

A *Unipolar* output is an output that, depending on its design, can either source or sink current, but not both. Ref. JESD99 “unipolar output”.

4.7.2.1.6.1. Bipolar

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar
diagram	<pre> classDiagram class BipolarOutputCircuitType { <<Passive-Pullup>> <<Passive-Pulldown>> <<Totem-Pole>> <<Push-Pull>> } class Bipolar { <<BipolarOutputCircuitType>> } Bipolar < -- BipolarOutputCircuitType </pre>
type	BipolarOutputCircuitType , BipolarOutputPassive-PullupType , BipolarOutputPassive-PulldownType , BipolarOutputTotem-PoleType , BipolarOutputPush-PullType ,

There are four basic types of *Bipolar* outputs which are described in the following sections.

4.7.2.1.6.1.1. Passive – Pullup

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Passive-Pullup
diagram	<pre> classDiagram class BipolarOutputPassive-PullupType { PNP-EmitterFollower P-ChannelSourceFollower Resistance CurrentSource Resistance Current } class ResistanceType { Resistance CurrentSource } class CurrentType { Current } class JEP30-D10:EmptyType class JEP30-D10:ValueSetType class JEP30-D10:ResistanceUOMType class JEP30-D10:CurrentUOMType </pre>
type	BipolarOutputPassive-PullupType , JEP30-D10:EmptyType , ResistanceType , CurrentType , JEP30-D10:CurrentUOMType

A *Passive-Pullup* output, as shown in Figure 18 below is an output similar to an open-circuit output except that, in addition to having an internal connection through an active device to a supply voltage, it also has an internal connection through a passive device, usually a resistor, to a second supply voltage that is more positive (less negative) than the first supply voltage. Ref JESD99 “passive-pullup output”.

An emitter follower is an output circuit whose output load is connected in the emitter circuit of a transistor and whose input is applied between the base and the remote end of the emitter load, which may be at ground potential. Ref JESD99 “emitter follower output”. The *PNP-EmitterFollower* is a *Passive-Pullup* output.

A source follower is an output circuit whose output load is connected in the source circuit of a field-effect transistor and whose input is applied between the gate and the remote end of the source load, which may be at ground potential. Ref JESD99 “source follower output”. The *P-ChannelSourceFollower* is a *Passive-Pullup* output.

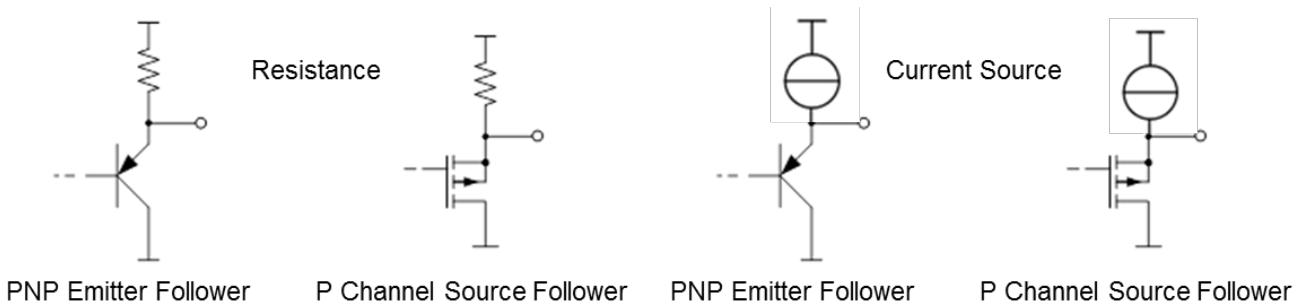


Figure 18 — Passive Pull-up Output Circuit

4.7.2.1.6.1.2. Passive - Pulldown

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Passive-Pulldown
diagram	<pre> classDiagram class BipolarOutputPassive-PulldownType { <<BipolarOutputPassive-PulldownType>> Passive-Pulldown < -- BipolarOutputPassive-PulldownType NPN-EmitterFollower < -- BipolarOutputPassive-PulldownType N-ChanellSourceFollower < -- BipolarOutputPassive-PulldownType Resistance < -- BipolarOutputPassive-PulldownType CurrentSource < -- BipolarOutputPassive-PulldownType ResistanceValueUOM < -- BipolarOutputPassive-PulldownType CurrentUOM < -- BipolarOutputPassive-PulldownType } class JEP30-D10:EmptyType { <<JEP30-D10:EmptyType>> NPN-EmitterFollower < -- JEP30-D10:EmptyType N-ChanellSourceFollower < -- JEP30-D10:EmptyType } class JEP30-D10:ValueSetType { <<JEP30-D10:ValueSetType>> Resistance < -- JEP30-D10:ValueSetType Current < -- JEP30-D10:ValueSetType } class JEP30-D10:ResistanceUOMType { <<JEP30-D10:ResistanceUOMType>> ResistanceValueUOM < -- JEP30-D10:ResistanceUOMType } class JEP30-D10:CurrentUOMType { <<JEP30-D10:CurrentUOMType>> CurrentUOM < -- JEP30-D10:CurrentUOMType } class ResistanceType { <<ResistanceType>> Resistance < -- ResistanceType } class CurrentType { <<CurrentType>> Current < -- CurrentType } </pre>
type	BipolarOutputPassive-PulldownType, JEP30-D10:EmptyType, ResistanceType, ResistanceUOMType, CurrentType, JEP30-D10:CurrentUOMType.

A *Passive-Pulldown* output, as shown in Figure 19 below is an output similar to an open-circuit except that, in addition to having an internal connection through an active device to a supply voltage, it also has an internal connection through a passive device, usually a resistor, to a second supply voltage that is more negative (less positive) than the first supply voltage. Ref. JESD99 “passive-pulldown output”. The *NPN-EmitterFollower* and the *N-ChanellSourceFollower* are *Passive-Pulldown* outputs.

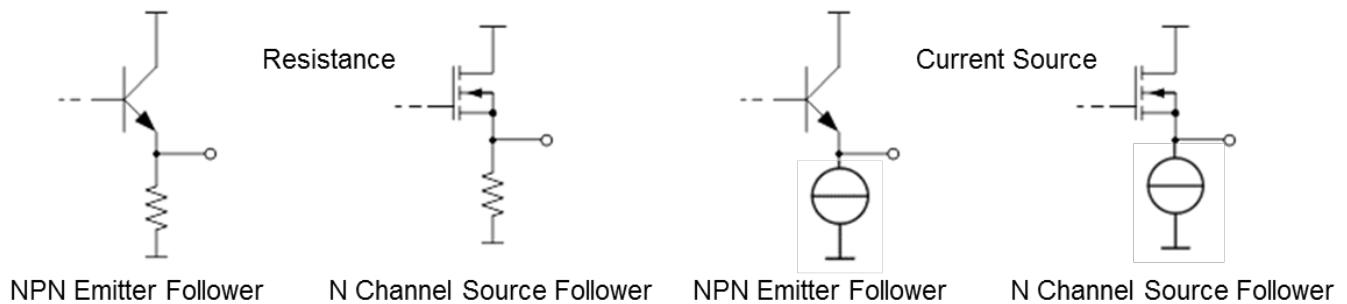


Figure 19 — Passive Pull-down Output Circuit

4.7.2.1.6.1.3. Totem - Pole

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Totem-Pole
diagram	<pre> graph TD BipolarOutputTotemPoleType[BipolarOutputTotem-PoleType] --- ThreeState[Three-state] BipolarOutputTotemPoleType --- RailToRail[Rail-to-Rail] BipolarOutputTotemPoleType --- ActivePullup[Active-Pullup] BipolarOutputTotemPoleType --- ActivePulldown[Active-Pulldown] BipolarOutputTotemPoleType --- HalfBridge[Half-Bridge] </pre>
type	BipolarOutputTotem-PoleType , JEP30-D10:EmptyType.

A *Totem-Pole* output is a bipolar output whose active devices are so controlled that as the resistance of one increases, the resistance of the other decreases so that, according to the relative states of the two active devices, the output voltage can swing between levels approaching the two supply voltages. Ref. JESD99 “totem-pole output”.

A *Three-state* output is a bipolar output both of whose active devices can be caused to be in the off state at the same time, thus presenting a high-impedance state at the output similar to the off state of an open circuit output. Ref. JESD99 “three-state output”.

A *Rail-to-Rail* driver is a bipolar (three-state or totem-pole) output that can swing between voltage levels that are essentially equal to the supply voltages. Ref. JESD99 “rail-to-rail driver”.

An *Active-Pullup* output is a bipolar (three-state or totem-pole) output whose sink-current capability significantly exceeds its source-current capability. An *Active-Pulldown* output is a bipolar (three-state or totem-pole) output whose source-current capability significantly exceeds its sink-current capability. *Active-Pullup* and *Active-Pulldown* are features that allow to limit power consumption by the output stage. Example: If the output is driven low, then active pullup increase its resistance to limit the current. Ref. JESD99 “active-pullup output” and “active-pulldown output” respectively.

A *Half-Bridge* (output) is a bipolar (three-state or totem-pole) power-driver output. Ref. JESD99 “half-bridge (output)”.

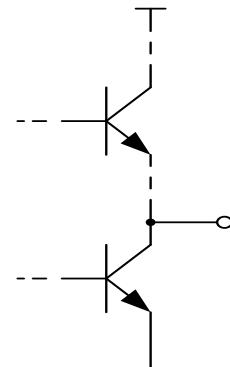


Figure 20 — Totem Pole

4.7.2.1.6.1.4. Push - Pull

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Push-Pull
diagram	<pre> classDiagram class Push_Pull { type BipolarOutputPush-PullType } class Rail_to_Rail { type EmptyType } class HighSideDriver { type BipolarOutputPush-PullHighSideDriverType } class LowSideDriver { type BipolarOutputPush-PullLowSideDriverType } class NPN_EmitterFollower { type EmptyType } class PNP_CommonEmitter { type EmptyType } class PNP_CommonEmitterDarlington { type EmptyType } class NPN_EmitterFollowerSziklai { type EmptyType } class P_ChannelSourceFollower { type EmptyType } class SourceCurrentLimit { type OutputCircuitSourcePropertyType } class SinkCurrentLimit { type OutputCircuitSourcePropertyType } class N_ChannelCommonSource { type EmptyType } class P_ChannelCommonSource { type EmptyType } Push_Pull < -- Rail_to_Rail Push_Pull < -- HighSideDriver Push_Pull < -- LowSideDriver Rail_to_Rail < -- NPN_EmitterFollower Rail_to_Rail < -- PNP_CommonEmitter Rail_to_Rail < -- PNP_CommonEmitterDarlington Rail_to_Rail < -- NPN_EmitterFollowerSziklai Rail_to_Rail < -- P_ChannelSourceFollower Rail_to_Rail --> SourceCurrentLimit HighSideDriver < -- NPN_EmitterFollower HighSideDriver < -- PNP_CommonEmitter HighSideDriver < -- PNP_CommonEmitterDarlington HighSideDriver < -- NPN_EmitterFollowerSziklai HighSideDriver < -- P_ChannelSourceFollower HighSideDriver --> SinkCurrentLimit LowSideDriver < -- NPN_EmitterFollower LowSideDriver < -- PNP_CommonEmitter LowSideDriver < -- PNP_CommonEmitterDarlington LowSideDriver < -- NPN_EmitterFollowerSziklai LowSideDriver < -- P_ChannelSourceFollower LowSideDriver --> N_ChannelCommonSource </pre>
type	BipolarOutputPush-PullType , JEP30-D10:EmptyType , BipolarOutputPush-PullHighSideDriverType , BipolarOutputPush-PullLowSideDriverType , OutputCircuitSourcePropertyType .

A *Push-Pull* output is two open-circuit outputs operating in complementary fashion so that as the resistance of one increases, the resistance of the other decreases. Ref JESD99 “push-pull output”.

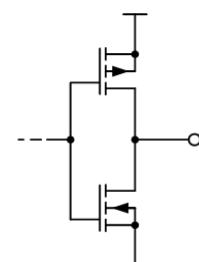


Figure 21 — Rail-to-Rail Push-Pull

4.7.2.1.6.1.4 Push – Pull (cont'd)

The following diagrams represent the variations of the High-side Driver.

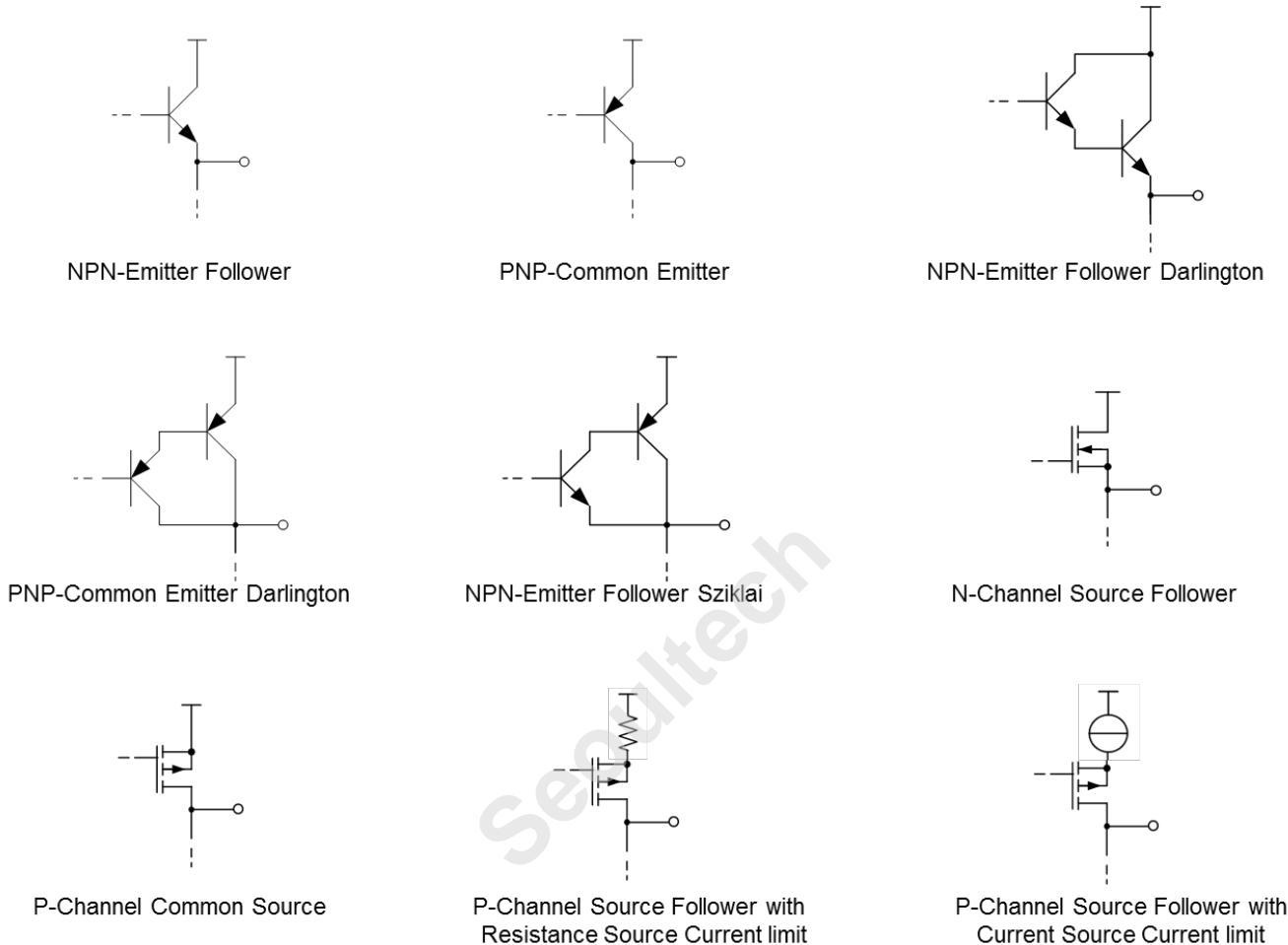


Figure 22 — High Side Driver (Source Driver)

A *HighSideDriver* is a source driver whose primary connection within the integrated circuit is through an active device to a positive supply voltage

4.7.2.1.6.1.4 Push – Pull (cont'd)

The following diagrams represent the variations of the Low-side Driver.

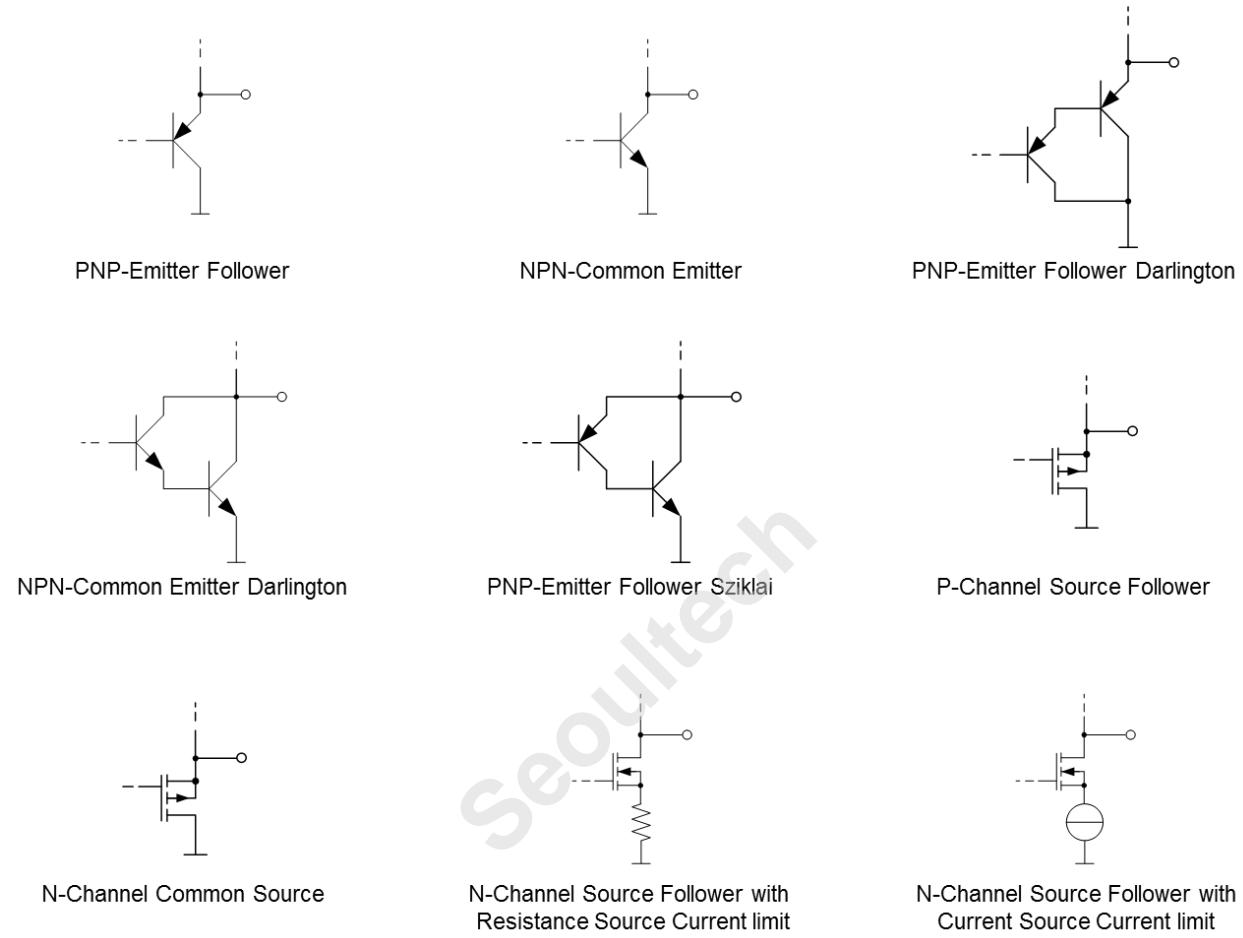


Figure 23 — Low Side Driver (Sink Driver)

A *LowSideDriver* is a sink driver whose primary connection within the integrated circuit is through an active device to the circuit common

4.7.2.1.6.2. Unipolar

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Unipolar
diagram	<pre> classDiagram class Unipolar { type UnipolarOutputCircuitType } class HighSideDriver { type UnipolarOutputSourceDriverType } class LowSideDriver { type UnipolarOutputSinkDriverType } class NPNOpenEmitter { type EmptyType } class PNPOpenCollector { type EmptyType } class NChannelOpenSource { type EmptyType } class PChannelOpenDrain { type EmptyType } class PNPOpenEmitter { type EmptyType } class NPNOopenCollector { type EmptyType } class PChannelOpenSource { type EmptyType } class NChannelOpenDrain { type EmptyType } Unipolar < -- HighSideDriver Unipolar < -- LowSideDriver HighSideDriver < -- NPNOpenEmitter HighSideDriver < -- PNPOpenCollector HighSideDriver < -- NChannelOpenSource HighSideDriver < -- PChannelOpenDrain LowSideDriver < -- PNPOpenEmitter LowSideDriver < -- NPNOopenCollector LowSideDriver < -- PChannelOpenSource LowSideDriver < -- NChannelOpenDrain </pre>
type	UnipolarOutputCircuitType , UnipolarOutputSinkDriverType , UnipolarOutputSourceDriverType , JEP30-D10:EmptyType.

A *LowSideDriver* is a sink driver whose primary connection within the integrated circuit is through an active device to the circuit common. Ref JESD99 “low-side driver”.

A *HighSideDriver* is a source driver whose primary connection within the integrated circuit is through an active device to a positive supply voltage. Ref JESD99 “high-side driver t”.

An open-circuit output (of an integrated circuit) is a unipolar output whose only connection within the integrated circuit is through an active device, usually a transistor, to one of the supply voltages. When the active device is in its ON state, the output voltage approaches the voltage of the supply to which it is connected (through the active device). When the device is in its OFF state, the output impedance to any other internal node of the integrated circuit is high and the output voltage is determined by the external circuit to which the output is connected. Outputs of this generic class are usually classified according to the name of the element of the active device to which they are connected within the integrated circuit, e.g.,

3. Open-emitter output,
 - a. An open-circuit output whose internal connection is to the emitter of a bipolar transistor. Ref. JESD99 “open-emitter output”.
NOTE For graphic symbols, see “source driver” (for npn) or “sink driver” (for pnp).
4. Open-collector output,
 - a. An open-circuit output whose internal connection is to the collector of a bipolar transistor. Ref. JESD99 “open-collector output”.

4.7.2.1.6.2 Unipolar (cont'd)

5. Open-source output,

- a. An open-circuit output whose internal connection is to the source of a field-effect transistor. Ref. JESD99 "open-source output".

NOTE For graphic symbols, see "source driver" (for p-channel outputs) or "sink driver" (for n-channel outputs).

6. Open-drain output,

- a. An open-circuit output whose internal connection is to the drain of a field-effect transistor. Ref JESD99 "open-drain output".

NOTE For graphic symbols, see "sink driver" (for n-channel outputs) or "source driver" (for p-channel outputs).

**Figure 24 — Open-circuit Low Side Driver Type****Figure 25 — Open-circuit High Side Driver Type**

4.7.2.1.6.3. Programmable

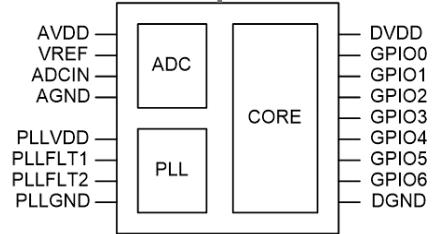
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Programmable
diagram	
type	ProgrammableOutputCircuitType, OutputCircuitSourcePropertyType, ResistanceType, JEP30-D10:ValueType, ElectricResistanceUOMType, CurrentType, JEP30-D10:ElectricCurrentUOMType.

4.7.2.1.7. Reference

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/Reference
diagram	
type	SignalReferenceType.

4.7.2.1.7 Reference (cont'd)

In electrical engineering, ground or earth is the reference point in an electrical circuit from which voltages are measured, a common return path for electric current, or a direct physical connection to the Earth. In a multilayer board, the ground plane and the power plane can be used as a reference for the signal line.



```

<Mapping-Array>
  <Mapping>
    <ID>Mapping ID 1</ID>
    <PackageTerminalMap>
      <ID>Package Terminal Map ID 1</ID>
      <PackageID>Package ID 1</PackageID> <!—PBGA-B17...-->
      <TerminalMap>
        <ID>Terminal Map ID 1</ID>
        <TerminalName>ADCIN</TerminalName>
        <TerminalNumber>3</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 2</ID>
        <TerminalName>VREF</TerminalName>
        <TerminalNumber>2</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 3</ID>
        <TerminalName>PLLFLT1</TerminalName>
        <TerminalNumber>6</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 4</ID>
        <TerminalName>PLLFLT2</TerminalName>
        <TerminalNumber>7</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 5</ID>
        <TerminalName>GPIO0</TerminalName>
        <TerminalNumber>16</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 6</ID>
        <TerminalName>GPIO1</TerminalName>
        <TerminalNumber>15</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 7</ID>
        <TerminalName>GPIO2</TerminalName>
        <TerminalNumber>14</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 8</ID>
        <TerminalName>GPIO3</TerminalName>
        <TerminalNumber>13</TerminalNumber>
      </TerminalMap>
    </PackageTerminalMap>
  </Mapping>
</Mapping-Array>
  
```

Figure 26 — Signal Reference

4.7.2.1.7 Reference (cont'd)

```
<TerminalMap>
    <ID>Terminal Map ID 9</ID>
    <TerminalName>GPIO4</TerminalName>
    <TerminalNumber>12</TerminalNumber>
</TerminalMap>
<TerminalMap>
    <ID>Terminal Map ID 10</ID>
    <TerminalName>GPIO5</TerminalName>
    <TerminalNumber>11</TerminalNumber>
</TerminalMap>
<TerminalMap>
    <ID>Terminal Map ID 11</ID>
    <TerminalName>GPIO6</TerminalName>
    <TerminalNumber>10</TerminalNumber>
</TerminalMap>
    :
    :
    :
</PackageTerminalMap>
</Mapping>
</ Mapping-Array>
    :
    :
    :
<ElectricalMap-Array>
    <ElectricalMap>
        <Terminal>
            <TerminalMapID>Terminal Map ID 1</TerminalMapID>
        </ Terminal>
        <PropertyID>Property ID 1</PropertyID>
    </ElectricalMap>
    <ElectricalMap>
        <Terminal>
            <TerminalMapID>Terminal Map ID 2</TerminalMapID>
        </ Terminal>
        <PropertyID>Property ID 2</PropertyID>
    </ElectricalMap>
    <ElectricalMap>
        <Terminal>
            <TerminalMapID>Terminal Map ID 3</TerminalMapID>
        </ Terminal>
        <Terminal>
            <TerminalMapID>Terminal Map ID 4</TerminalMapID>
        </ Terminal>
        <PropertyID>Property ID 3</PropertyID>
    </ElectricalMap>
    <ElectricalMap>
        <Terminal>
            <TerminalMapID>Terminal Map ID 5</TerminalMapID>
        </ Terminal>
        <Terminal>
            <TerminalMapID>Terminal Map ID 6</TerminalMapID>
        </ Terminal>
        <Terminal>
            <TerminalMapID>Terminal Map ID 7</TerminalMapID>
        </ Terminal>
```

4.7.2.1.7 Reference (cont'd)

```

<Terminal>
    <TerminalMapID>Terminal Map ID 8</TerminalMapID>
</Terminal>
<Terminal>
    <TerminalMapID>Terminal Map ID 9</TerminalMapID>
</Terminal>
<Terminal>
    <TerminalMapID>Terminal Map ID 10</TerminalMapID>
</Terminal>
<Terminal>
    <TerminalMapID>Terminal Map ID 11</TerminalMapID>
</Terminal>
<PropertyID>Property ID 4</PropertyID>
</ElectricalMap>
</ElectricalMap-Array>
:
:
:
<Properties-Array>
    <Properties>
        <ID>Property ID 1</ID> <!-- ADCIN references AGND and AVDD -->
        <Reference>
            <GroundReference-TerminalName>AGND</GroundReference-TerminalName>
            <PositiveReference-TerminalName>AVDD</PositiveReference-TerminalName>
        </Reference>
    </Properties>
    <Properties>
        <ID>Property ID 2</ID> <!-- VREF references AGND -->
        <Reference>
            <GroundReference-TerminalName>AGND</GroundReference-TerminalName>
        </Reference>
    </Properties>
    <Properties>
        <ID>Property ID 3</ID> <!-- PLLFLT1 and PLLFLT2 references PLLGND and
PLLVDD -->
        <Reference>
            <GroundReference-TerminalName>PLLGND</GroundReference-TerminalName>
            <PositiveReference-TerminalName>PLLVDD</PositiveReference-TerminalName>
        </Reference>
    </Properties>
    <Properties>
        <ID>Property ID 4</ID>
        <Reference>
            <GroundReference-TerminalName>DGND</GroundReference-TerminalName>
            <PositiveReference-TerminalName>DVDD</PositiveReference-TerminalName>
        </Reference>
    </Properties>
</Properties-Array>

```

4.7.2.2. Terminal Function – Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalFunction-Array
diagram	<pre> classDiagram class TerminalFunctionArrayType { <<TerminalFunction-Array>> <<TerminalFunctionType>> <<DigitalFunctionType>> <<SignalClassificationType>> <<DigitalTerminalStateType>> } class TerminalFunctionType { <<ID>> <<xs:string>> } class DigitalFunctionType { <<Signal>> } class SignalClassificationType { <<Terminal State>> <<DigitalTerminalStateType>> } class DigitalTerminalStateType TerminalFunctionArrayType "1..>" TerminalFunctionType TerminalFunctionType "1..>" DigitalFunctionType DigitalFunctionType "1..>" SignalClassificationType SignalClassificationType "1..>" DigitalTerminalStateType </pre> <p>The diagram illustrates the UML Class Diagram for the <code>TerminalFunction-Array</code>. It defines several classes and their associations:</p> <ul style="list-style-type: none"> <code>TerminalFunction-Array</code> (type <code>TerminalFunctionArrayType</code>) is associated with <code>TerminalFunctionType</code> via a multiplicity of <code>1..></code>. <code>TerminalFunctionType</code> (type <code>TerminalFunctionType</code>) is associated with <code>DigitalFunctionType</code> via a multiplicity of <code>1..></code>. <code>DigitalFunctionType</code> (type <code>DigitalFunctionType</code>) is associated with <code>SignalClassificationType</code> via a multiplicity of <code>1..></code>. <code>SignalClassificationType</code> (type <code>SignalClassificationType</code>) is associated with <code>DigitalTerminalStateType</code> via a multiplicity of <code>1..></code>. <p>A constraint block labeled <code>constraints</code> is also present.</p>
type	TerminalFunction-ArrayType , TerminalFunctionType , DigitalFunctionType , SignalClassificationType , DigitalTerminalStateType .

This section captures additional information about digital terminals, basically the classification of the digital signal and the logic state of the terminal necessary to perform various functions for certain types of devices.

4.7.2.2.1. Digital Function – Signal

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalFunction-Array/TerminalFunction/DigitalFunction/Signal
diagram	<p>The diagram illustrates the SignalClassificationType class and its associations. A Signal object (type: SignalClassificationType) is associated with the SignalClassificationType class via a multiplicity of 1..* at the Signal end. The SignalClassificationType class contains the following attributes:</p> <ul style="list-style-type: none"> ID: type: xs:string Description: type: xs:string Address: type: JEP30-D10:EmptyType Clock: type: JEP30-D10:EmptyType Command: type: JEP30-D10:EmptyType Control: type: JEP30-D10:EmptyType Data: type: JEP30-D10:EmptyType Enable: type: JEP30-D10:EmptyType Reset: type: JEP30-D10:EmptyType Select: type: JEP30-D10:EmptyType Strobe: type: JEP30-D10:EmptyType Synch: type: JEP30-D10:EmptyType Trigger: type: JEP30-D10:EmptyType Other: type: xs:string
type	SignalClassificationType, DigitalTerminalStateType, JEP30-D10:EmptyType.

The various classifications of a digital **Signal** are as outlined above; however other classifications can be specified in the category **Other**. The above **Signal** classifications can assist software tools to be more efficient in terms of schematic symbol generation, net connectivity, and schematic DRC (Design Rule Checking).

4.7.2.2.2. Digital Function – Terminal State

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalFunction-Array/TerminalFunction/DigitalFunction/TerminalState</code>
diagram	<pre> classDiagram class DigitalTerminalStateType { class Reset { type TerminalSignalStateType } class Programming { type TerminalSignalStateType } class Sleeping { type TerminalSignalStateType } class Unprogrammed { type TerminalSignalStateType } class TerminalState { type DigitalTerminalStateType } class OtherState { type TerminalOtherSignalStateArrayType } } class TerminalSignalStateType { class IO_SignalState { type SignalStateType } } class TerminalOtherSignalStateType { class Name { type xs:string } class SignalState { type SignalStateType } } DigitalTerminalStateType < -- Reset DigitalTerminalStateType < -- Programming DigitalTerminalStateType < -- Sleeping DigitalTerminalStateType < -- Unprogrammed DigitalTerminalStateType < -- TerminalState DigitalTerminalStateType < -- OtherState TerminalSignalStateType < -- IO_SignalState TerminalOtherSignalStateType < -- Name TerminalOtherSignalStateType < -- SignalState </pre>
type	<code>DigitalTerminalStateType, TerminalSignalStateType, SignalStateType,</code> <code>TerminalOtherSignalState-ArrayType, TerminalOtherSignalStateType.</code>

The four most common types of terminal signal states for complex and/or programmable devices are

1. *Reset*,
2. *Programming*,
3. *Sleeping*, and
4. *Unprogrammed*.

Other Terminal states can be captured under the category *OtherState-Array*. The enumerated values of the state are as follows:-

1. *0*,
2. *OPulldown*,
3. *1*,
4. *1Pullup*,
5. *HighImpedanceState*,
6. *LowState*,
7. *HighState*,
8. *Input*,
9. *Output*,

4.7.2.3. Terminal Grouping – Array

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping</code>
diagram	<pre> classDiagram class TerminalGroupingType { <<TerminalGrouping>> } class SamePotentialGroupArrayType { <<SameSupplyPotentialGroup-ArrayType>> } class SamePotentialGroupType { <<SameSupplyPotentialGroupType>> } class TerminalSwapArrayType { <<TerminalSwap-ArrayType>> } class TerminalSwapType { <<TerminalSwapType>> } class FunctionSwapArrayType { <<FunctionSwap-ArrayType>> } class FunctionSwapType { <<FunctionSwapType>> } class InternalElectricalConnectionArrayType { <<InternalElectricalConnection-ArrayType>> } class InternalElectricalConnectionType { <<InternalElectricalConnectionType>> } class DifferentialPairArrayType { <<DifferentialPair-ArrayType>> } class DifferentialPairType { <<DifferentialPairType>> } class LogicalGroupArrayType { <<Logical-Group-ArrayType>> } class LogicalGroupType { <<LogicalGroupType>> } class TerminalToTerminalSignalPathArrayType { <<Terminal-to-TerminalSignalPath-ArrayType>> } class SignalPathConditionType { <<SignalPathCondition-ArrayType>> } class TerminalToTerminalSignalPathType { <<Terminal-to-TerminalSignalPathType>> } TerminalGroupingType < -- SamePotentialGroupArrayType TerminalGroupingType < -- TerminalSwapArrayType TerminalGroupingType < -- FunctionSwapArrayType TerminalGroupingType < -- InternalElectricalConnectionArrayType TerminalGroupingType < -- DifferentialPairArrayType TerminalGroupingType < -- LogicalGroupArrayType TerminalGroupingType < -- TerminalToTerminalSignalPathArrayType SamePotentialGroupArrayType < -- SamePotentialGroupType TerminalSwapArrayType < -- TerminalSwapType FunctionSwapArrayType < -- FunctionSwapType InternalElectricalConnectionArrayType < -- InternalElectricalConnectionType DifferentialPairArrayType < -- DifferentialPairType LogicalGroupArrayType < -- LogicalGroupType TerminalToTerminalSignalPathArrayType < -- SignalPathConditionType TerminalToTerminalSignalPathType < -- TerminalToTerminalSignalPathType </pre> <p>The diagram illustrates the UML class structure for <code>TerminalGroupingType</code>. It shows inheritance relationships where <code>TerminalGroupingType</code> is the base class for several array types: <code>SamePotentialGroup-Array</code>, <code>TerminalSwap-Array</code>, <code>FunctionSwap-Array</code>, <code>InternalElectricalConnection-Array</code>, <code>DifferentialPair-Array</code>, and <code>Logical-Group-Array</code>. Each array type has its corresponding specific type (e.g., <code>SameSupplyPotentialGroupType</code>, <code>TerminalSwapType</code>, etc.). Additionally, <code>TerminalGroupingType</code> is the base class for <code>Terminal-to-TerminalSignalPath-Array</code>, which further includes <code>SignalPathCondition</code> and <code>Terminal-to-TerminalSignalPathType</code>. Each class is associated with a multiplicity of 1..∞, except for <code>SignalPathCondition</code> which is 0..∞.</p>
type	<code>TerminalGroupingType</code> , <code>SameSupplyPotentialGroup-ArrayType</code> , <code>SameSupplyPotentialGroupType</code> , <code>TerminalSwap-ArrayType</code> , <code>TerminalSwapType</code> , <code>FunctionSwap-ArrayType</code> , <code>FunctionSwapType</code> , <code>InternalElectricalConnection-ArrayType</code> , <code>InternalElectricalConnectionType</code> , <code>DifferentialPair-ArrayType</code> , <code>DifferentialPairType</code> , <code>LogicalGroup-ArrayType</code> , <code>LogicalGroupType</code> , <code>Terminal-to-TerminalSignalPath-ArrayType</code> , <code>SignalPathCondition-ArrayType</code> , <code>Terminal-to-TerminalSignalPathType</code> ,

This section captures data about various terminal groups based on their similarities.

4.7.2.3.1. Same Potential Group - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/SamePotentialGroup-Array
diagram	<pre> classDiagram class SameSupplyPotentialGroup-ArrayType class SameSupplyPotentialGroupType { <<SamePotentialGroup>> <<TerminalName>> <<TerminalNumber>> <<FunctionID>> <<StandardTerminalName>> } class SamePotentialGroup { <<SameSupplyPotentialGroupType>> } SameSupplyPotentialGroup-ArrayType "1..∞" --> SamePotentialGroup SameSupplyPotentialGroupType "1..∞" --> Name SameSupplyPotentialGroupType "1..∞" --> TerminalName SameSupplyPotentialGroupType "1..∞" --> TerminalNumber SameSupplyPotentialGroupType "1..∞" --> FunctionID SameSupplyPotentialGroupType "1..∞" --> StandardTerminalName </pre>
type	SameSupplyPotentialGroup-ArrayType , SameSupplyPotentialGroupType

This device shown in Figure 27 has 3 terminal names that have the same potential.

```

<SamePotentialGroup->
  <TerminalName>DGND</TerminalName>
  <TerminalName>AGND</TerminalName>
  <TerminalName>PLLGND</TerminalName>
</SamePotentialGroup>
  
```

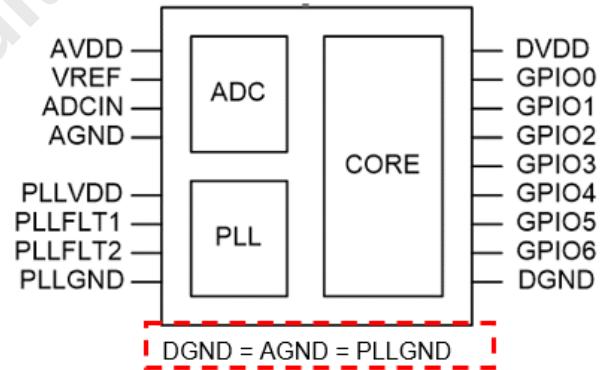


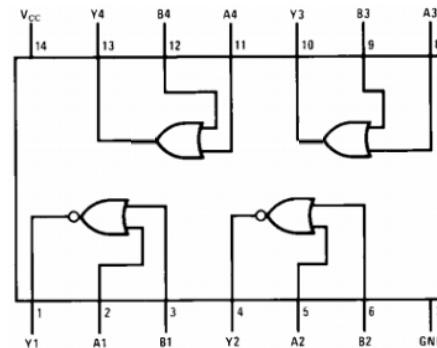
Figure 27 – Same potential

4.7.2.3.2. Terminal Swap - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/TerminalSwap-Array
diagram	<p>The diagram illustrates the UML class structure for the <code>TerminalSwap-ArrayType</code>. It features a main class <code>TerminalSwap-ArrayType</code> with a dashed association line pointing to a detailed view enclosed in a yellow box. This detailed view shows the inheritance path from <code>TerminalSwap-ArrayType</code> to <code>TerminalSwapType</code>, which then branches into two specific types: <code>TerminalName</code> and <code>TerminalNumber</code>. Both of these types have multiplicity <code>1..∞</code> indicated by arrows. Additionally, there is a dashed association line from <code>TerminalSwapType</code> to another class <code>FunctionID</code>, also with multiplicity <code>1..∞</code>.</p> <pre> classDiagram class TerminalSwap-ArrayType { <<TerminalSwap-Array type TerminalSwap-ArrayType>> } class TerminalSwapType { <<TerminalSwap type TerminalSwapType>> } class TerminalName { <<TerminalName type xs:string>> } class TerminalNumber { <<TerminalNumber type xs:string>> } class FunctionID { <<FunctionID type xs:string>> } class StandardTerminalName { <<StandardTerminalName type xs:string>> } TerminalSwap-ArrayType "1..∞" --> TerminalSwapType TerminalSwapType "1..∞" --> TerminalName TerminalSwapType "1..∞" --> TerminalNumber TerminalSwapType "1..∞" --> FunctionID </pre>
type	<code>TerminalSwap-ArrayType</code> , <code>TerminalSwapType</code> .

If terminals names can be swapped, as in Figure 28, where,

1. Terminals 2 and 3 can be swapped,
 2. Terminals 5 and 6 can be swapped,
 3. Terminal 8 and 9 can be swapped, and
 4. Terminal 11 and 12 can be swapped.



This data can be captured under the *TerminalSwap–Array* section, as follows:

Figure 28 — Sample Mixed Gate Device

```
<TerminalSwap-Array>
  <TerminalSwap>
    <TerminalNumber>2</TerminalNumber>
    <TerminalNumber>3</TerminalNumber>
  </TerminalSwap>
  <TerminalSwap>
    <TerminalNumber>5</TerminalNumber>
    <TerminalNumber>6</TerminalNumber>
  </TerminalSwap>
```

4.7.2.3.2 Terminal Swap – Array (cont'd)

```

<TerminalSwap>
  <TerminalNumber>8</TerminalNumber>
  <TerminalNumber>9</TerminalNumber>
</TerminalSwap>
<TerminalSwap>
  <TerminalNumber>11</TerminalNumber>
  <TerminalNumber>12</TerminalNumber>
</TerminalSwap>
</TerminalSwap-Array>

```

See next section to capture the details of how to swap one gate function with the next gate function.

4.7.2.3.3 Function Swap - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/FunctionSwap-Array
diagram	
type	FunctionSwap-ArrayType, FunctionSwapType.

Figure 28 — Sample Mixed Gate Device gave an example of four gates that could be interchangeable, as follows:

Gate 1 can be swapped with Gate 2, but not with either Gate 3 or Gate 4, because Gates 1 and 2 are OR gates while Gates 3 and 4 are NOR gates.

1. Gate 1 has Terminals 1, 2, 3,
2. Gate 2 has Terminals 4, 5, 6.
3. Gate 3 has Terminals 8, 9, 10,
4. Gate 4 has Terminals 11, 12, 13.

This is an ordered terminal list, meaning that the sequence of the numbers in the list are in the same order for each gate that is swappable with each other.

4.7.2.3.3 Function Swap – Array (cont'd)

```
<FunctionSwap-Array>
  <FunctionSwap>
    <TerminalNumberOrderedList>1, 2, 3</TerminalNumberOrderedList>
    <TerminalNumberOrderedList>4, 5, 6</TerminalNumberOrderedList>
  </FunctionSwap>
  <FunctionSwap>
    <TerminalNumberOrderedList>8, 9, 10</TerminalNumberOrderedList>
    <TerminalNumberOrderedList>11, 12, 13</TerminalNumberOrderedList>
  </FunctionSwap>
</FunctionSwap-Array>
```

Alternatively, terminal name could have been used as follows

```
<FunctionSwap-Array>
  <FunctionSwap>
    <TerminalNameOrderedList>Y1, A1, B1</TerminalNameOrderedList>
    <TerminalNameOrderedList> Y2, A2, B2</TerminalNameOrderedList>
  </FunctionSwap>
  <FunctionSwap>
    <TerminalNameOrderedList> Y3, A3, B3</TerminalNameOrderedList>
    <TerminalNameOrderedList> Y4, A4, B4</TerminalNameOrderedList>
  </FunctionSwap>
</FunctionSwap-Array>
```

If all the four gates were swappable as in Figure 29, then the XML structure would look like this. Note how the terminal number sequence in the ordered list, represents the same terminal name (function) of each gate. This means that if I swap gate 1 with gate 3, that terminal 1 would swap with terminal 10, terminal 2 would swap with terminal 9, and terminal 3 would swap with terminal 8, all in one operation.

```
<FunctionSwap-Array>
  <FunctionSwap>
    <TerminalNumberOrderedList>1, 2,
  3</TerminalNumberOrderedList>
    <TerminalNumberOrderedList>4, 5, 6</TerminalNumberOrderedList>
    <TerminalNumberOrderedList>10, 9, 8</TerminalNumberOrderedList>
    <TerminalNumberOrderedList>13, 12, 11</TerminalNumberOrderedList>
  </FunctionSwap>
</FunctionSwap-Array>
```

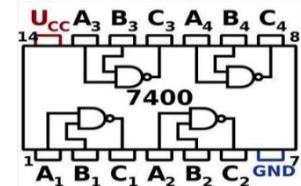


Figure 29 – Sample NAND Gate Device

4.7.2.3.4. Internal Electrical Connection - Array

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/InternalElectricalConnection-Array</code>
diagram	<pre> classDiagram class InternalElectricalConnection-ArrayType { <<InternalElectricalConnection>> } class InternalElectricalConnectionType { <<InternalElectricalConnection>> <<Name : xs:string>> <<TerminalNumber : xs:string>> } InternalElectricalConnection-ArrayType "1..∞" -- "1..∞" InternalElectricalConnectionType InternalElectricalConnection "1..∞" -- "1..∞" InternalElectricalConnectionType </pre>
type	<code>InternalElectricalConnection-ArrayType, InternalElectricalConnectionType.</code>

This section captures all the Terminals that are electrically connected together inside the device, under the `InternalElectricalConnection` branch.

```

<InternalElectricalConnection-Array>
  <InternalElectricalConnection>
    <TerminalNumber>1</TerminalNumber>
    <TerminalNumber>2</TerminalNumber>
    <TerminalNumber>3</TerminalNumber>
    <TerminalNumber>4</TerminalNumber>
  </InternalElectricalConnection>
  <InternalElectricalConnection>
    <TerminalNumber>5</TerminalNumber>
    <TerminalNumber>6</TerminalNumber>
    <TerminalNumber>7</TerminalNumber>
  </InternalElectricalConnection>
</InternalElectricalConnection-Array>
  
```

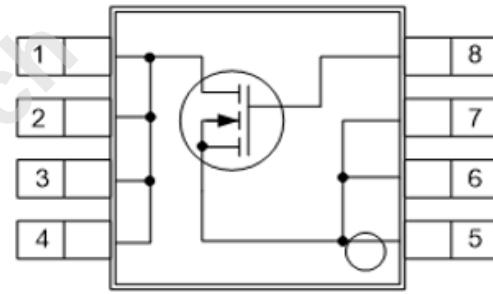


Figure 30 — Internal Electrical Connection

4.7.2.3.5. Differential Pair - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/DifferentialPair-Array
diagram	<pre> classDiagram class DifferentialPair { <<DifferentialPair>> <<DifferentialPairType>> <<1..>> } class DifferentialPairType { <<1..>> ID : xs:string Name : xs:string PositiveTerminalName : xs:string NegativeTerminalName : xs:string FunctionID : xs:string PositiveStandardTerminalName : xs:string NegativeStandardTerminalName : xs:string DifferentialDescription : xs:string } class DifferentialPair-ArrayType { <<1..>> <<DifferentialPair-ArrayType>> } DifferentialPair-ArrayType "1..>" DifferentialPair DifferentialPair "1..>" DifferentialPairType </pre> <p>The diagram illustrates the UML Class Diagram for the <code>DifferentialPair-ArrayType</code> and its associated <code>DifferentialPairType</code>. The <code>DifferentialPair-ArrayType</code> is represented by a dashed box containing a generalization relationship to the <code>DifferentialPair</code> class. The <code>DifferentialPair</code> class is also associated with the <code>DifferentialPairType</code> class via another generalization relationship. The <code>DifferentialPairType</code> class contains several attributes: <code>ID</code> (type <code>xs:string</code>), <code>Name</code> (type <code>xs:string</code>), <code>PositiveTerminalName</code> (type <code>xs:string</code>), <code>NegativeTerminalName</code> (type <code>xs:string</code>), <code>FunctionID</code> (type <code>xs:string</code>), <code>PositiveStandardTerminalName</code> (type <code>xs:string</code>), <code>NegativeStandardTerminalName</code> (type <code>xs:string</code>), and <code>DifferentialDescription</code> (type <code>xs:string</code>).</p>

This section captures the terminal data that makes up the *DifferentialPair*. Differential signaling is a method for electrically transmitting information using two complementary signals. The technique sends the same electrical signal as a differential pair of signals, each in its own conductor. The receiving circuit responds to the electrical difference between the two signals, rather than the difference between a single wire and ground. Desired Signals are added, and noise is subtracted away.

Desired Signals are added, and noise is subtracted away.

The opposite technique is called single-ended signaling. Provided that the source and receiver impedances in the differential signaling circuit are equal, external electromagnetic interference tends to affect both conductors identically. Since the receiving circuit only detects the difference between the wires, the technique resists electromagnetic noise compared to one conductor with an un-paired reference (ground).

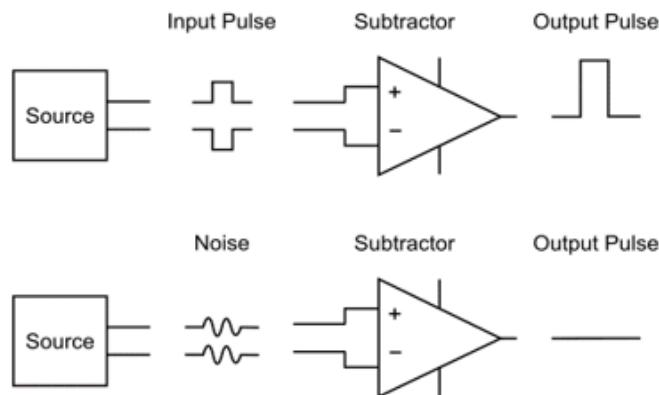


Figure 31 — System with Differential Receiver

4.7.2.3.5 Differential Pair - Array (cont'd)

The technique minimizes electronic crosstalk and electromagnetic interference, both noise emission and noise acceptance, and can achieve a constant or known characteristic impedance, allowing impedance matching techniques important in a high-speed signal transmission line or high-quality balanced line and balanced circuit audio signal path.

The technique works for both analog and digital signaling.

A *DifferentialPair* may also have a unique set of electrical specifications that can be referenced via the *ElectricalSpecificationID*.

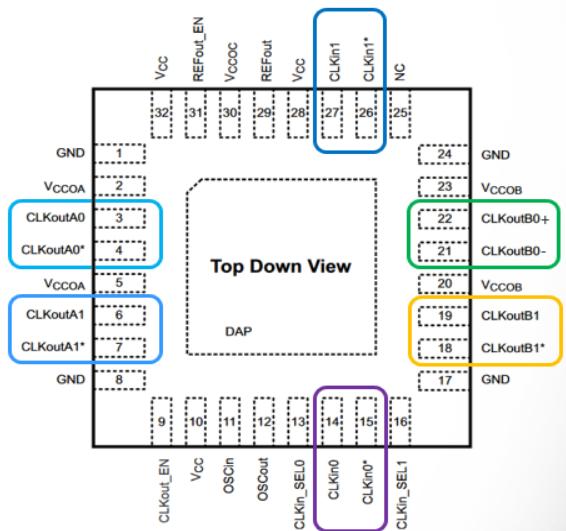


Figure 32 — Differential Pair Device

```

<DifferentialPair-Array>
  <DifferentialPair>
    <Name>CLK_outA0</Name>
    <PositiveTerminalName>CLK_outA0</PositiveTerminalName>
    <NegativeTerminalName>CLK_outA0*</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>CLK_outA1</Name>
    <PositiveTerminalName>CLK_outA1</PositiveTerminalName>
    <NegativeTerminalName>CLK_outA1*</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>CLK_in0</Name>
    <PositiveTerminalName>CLK_in0</PositiveTerminalName>
    <NegativeTerminalName>CLK_in0*</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>CLK_outB1</Name>
    <PositiveTerminalName>CLK_outB1</PositiveTerminalName>
    <NegativeTerminalName>CLK_outB1*</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>CLK_outB0</Name>
    <PositiveTerminalName>CLK_outB0+</PositiveTerminalName>
    <NegativeTerminalName>CLK_outB0-</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>CLK_in1</Name>
    <PositiveTerminalName>CLK_in1</PositiveTerminalName>
    <NegativeTerminalName>CLK_in1*</NegativeTerminalName>
  </DifferentialPair>
</DifferentialPair-Array>

```

4.7.2.3.5 Differential Pair - Array (cont'd)

If a Differential Pair will be also captured in a *Terminal-to-TerminalSignalPath*, as seen in section 4.5.2.3.7 below, then the *Name* element should be captured, so that it can be referenced in that section. Typically, the name assigned to a Differential Pair, is the same as the common characters of the positive and negative terminal names. The XML example below represents the differential pairs shown in Figure 35 — Differential Multiplexer, and can be used to demonstrate how the differential pair names can be leveraged in the section.

```
<DifferentialPair-Array>
  <DifferentialPair>
    <Name>IN0</Name>
    <PositiveTerminalName>INP0</PositiveTerminalName>
    <NegativeTerminalName>INN0</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>IN1</Name>
    <PositiveTerminalName>INP1</PositiveTerminalName>
    <NegativeTerminalName>INN1</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>OUT0</Name>
    <PositiveTerminalName>OUTP0</PositiveTerminalName>
    <NegativeTerminalName>OUTN0</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>OUT1</Name>
    <PositiveTerminalName>OUTP1</PositiveTerminalName>
    <NegativeTerminalName>OUTN1</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>OUT2</Name>
    <PositiveTerminalName>OUTP2</PositiveTerminalName>
    <NegativeTerminalName>OUTN2</NegativeTerminalName>
  </DifferentialPair>
  <DifferentialPair>
    <Name>OUT3</Name>
    <PositiveTerminalName>OUTP3</PositiveTerminalName>
    <NegativeTerminalName>OUTN3</NegativeTerminalName>
  </DifferentialPair>
</DifferentialPair-Array>
```

4.7.2.3.6. Logical Group - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/LogicalGroup-Array
diagram	<p>The diagram illustrates the UML Class Diagram for the LogicalGroup-ArrayType and LogicalGroupType. The LogicalGroup-ArrayType is a composite structure containing a LogicalGroupType element. The LogicalGroupType has attributes: ID (xs:string), Name (xs:string), Bus (JEP30-D10:EmptyType), Function (JEP30-D10:EmptyType), CommonName (JEP30-D10:EmptyType), Other (xs:string), TerminalName (xs:string), FunctionID (xs:string), and StandardTerminalName (xs:string). The LogicalGroup-ArrayType has a multiplicity of 1..∞ for its contained LogicalGroupType elements.</p>
type	LogicalGroup-ArrayType , LogicalGroupType , JEP30-D10:EmptyType .

A [LogicalGroup](#) of Terminals can provide significant efficiencies within software tools, such as schematic capture tools and PB Layout tools. The grouping of all the terminals involved in a [Bus](#) can significantly reduce the visualization complexity of bus connections in a schematic. Instead of showing 64 separate routes (where 1 route represents 1 bit of a 64-bit bus), the entire set of 64 routes can be routed throughout the schematic as a “Bus identifiable trace” which is then tapped for the appropriate bit routes at the various route designations.

Rules can be applied to logical groupings, thereby applying to all terminals identified by that [LogicalGroup](#).

Figure 33 a function block diagram of a 12-bit wide bus switch. The A port can be routed to the B or C port for all bits simultaneously. The switches can be bi-directional depending on the setting of SEL 1 and SEL 2. The EN terminal can be toggled high to put all channels into high-Z mode.

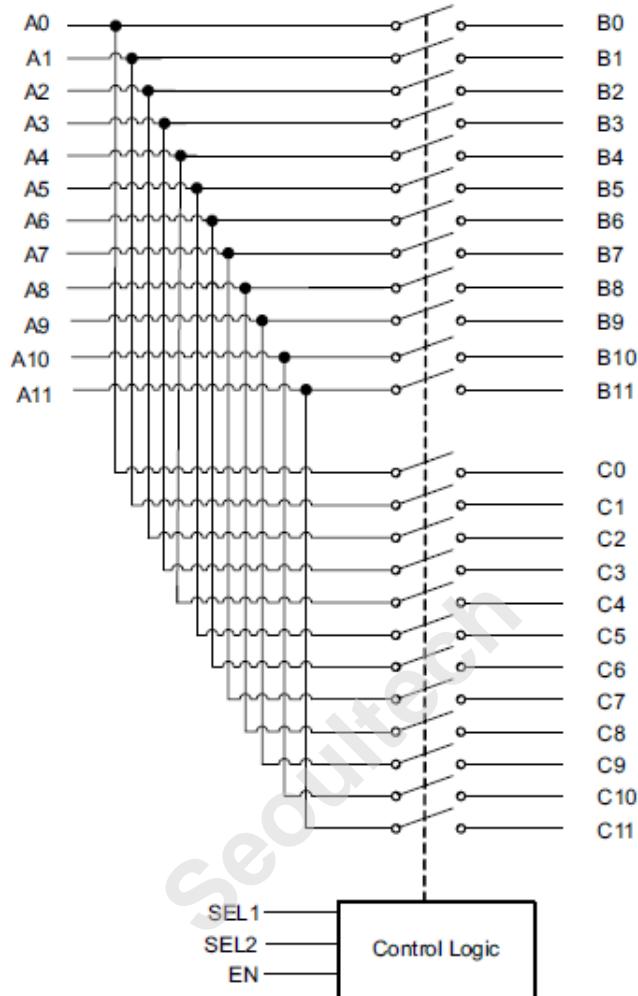
4.7.2.3.6 Logical Group - Array (cont'd)

Figure 33 — Function Block Diagram of a 12-bit Bus Switching Device

Table 2 in 4.5.2.3.7 Terminal-to-Terminal Signal Path – Array shows the logic for the operation of this function block shown in Figure 33.

4.7.2.3.6 Logical Group - Array (cont'd)

The following XML example shows the XML representation of the Logical Groups of the above function block diagram.

```
<LogicalGroup-Array>
  <LogicalGroup>
    <Name>A</Name>
    <Bus/>
    <TerminalName>A0</TerminalName>
    <TerminalName>A1</TerminalName>
    <TerminalName>A2</TerminalName>
    <TerminalName>A3</TerminalName>
    <TerminalName>A4</TerminalName>
    <TerminalName>A5</TerminalName>
    <TerminalName>A6</TerminalName>
    <TerminalName>A7</TerminalName>
    <TerminalName>A8</TerminalName>
    <TerminalName>A9</TerminalName>
    <TerminalName>A10</TerminalName>
    <TerminalName>A11</TerminalName>
  </LogicalGroup>
  <LogicalGroup>
    <Name>B</Name>
    <Bus/>
    <TerminalName>B0</TerminalName>
    <TerminalName>B1</TerminalName>
    <TerminalName>B2</TerminalName>
    <TerminalName>B3</TerminalName>
    <TerminalName>B4</TerminalName>
    <TerminalName>B5</TerminalName>
    <TerminalName>B6</TerminalName>
    <TerminalName>B7</TerminalName>
    <TerminalName>B8</TerminalName>
    <TerminalName>B9</TerminalName>
    <TerminalName>B10</TerminalName>
    <TerminalName>B11</TerminalName>
  </LogicalGroup>
  <LogicalGroup>
    <Name>C</Name>
    <Bus/>
    <TerminalName>C0</TerminalName>
    <TerminalName>C1</TerminalName>
    <TerminalName>C2</TerminalName>
    <TerminalName>C3</TerminalName>
    <TerminalName>C4</TerminalName>
    <TerminalName>C5</TerminalName>
    <TerminalName>C6</TerminalName>
    <TerminalName>C7</TerminalName>
    <TerminalName>C8</TerminalName>
    <TerminalName>C9</TerminalName>
    <TerminalName>C10</TerminalName>
    <TerminalName>C11</TerminalName>
  </LogicalGroup>
</LogicalGroup-Array>
```

4.7.2.3.7. Terminal-to-Terminal Signal Path – Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/Terminal-to-TerminalSignalPath-Array
diagram	<pre> classDiagram class Terminal-to-TerminalSignalPath-ArrayType { <<Terminal-to-TerminalSignalPath-ArrayType>> <<SignalPathCondition-ArrayType>> <<SignalStateType>> } class Terminal-to-TerminalSignalPath-Array { <<Terminal-to-TerminalSignalPath-Array>> <<Terminal-to-TerminalSignalPath-ArrayType>> } class Terminal-to-TerminalSignalPathType { <<Terminal-to-TerminalSignalPathType>> <<SignalPathCondition-ArrayType>> <<Name>> <<Description>> <<Uni-directional>> <<Bi-directional>> } Terminal-to-TerminalSignalPath-ArrayType "0..∞" --> "1..∞" SignalPathCondition-ArrayType Terminal-to-TerminalSignalPath-ArrayType "1..∞" --> SignalStateType Terminal-to-TerminalSignalPath-Array --> Terminal-to-TerminalSignalPath-ArrayType Terminal-to-TerminalSignalPathType "0..∞" --> SignalPathCondition-ArrayType Terminal-to-TerminalSignalPathType "1..∞" --> Name Terminal-to-TerminalSignalPathType "1..∞" --> Description Terminal-to-TerminalSignalPathType "1..∞" --> Uni-directional Terminal-to-TerminalSignalPathType "1..∞" --> Bi-directional </pre>
type	Terminal-to-TerminalSignalPath-ArrayType , SignalPathCondition-ArrayType , SignalStateType , Terminal-to-TerminalSignalPathType , Uni-directionalType , Bi-directionalType .

In a two terminal device, the signal path is easily defined, as the signal enters one terminal and exits the other. However, in more complex circuits, this is not so easily defined.

Terminal-to-Terminal Signal Paths can be *Uni-directional* or *Bi-directional*. The following shows several examples of [Terminal-to-TerminalSignalPath-Array](#) with their XML representation.

4.7.2.3.7.1. Uni-directional Terminal-to-Terminal Signal Path

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/Terminal-to-TerminalSignalPath-Array/Terminal-to-TerminalSignalPath/Uni-directional
diagram	<p>The diagram shows a class named Uni-directional which has a multiplicity of 1..∞ and is associated with a class Uni-directionalType. The Uni-directionalType class contains several attributes:</p> <ul style="list-style-type: none"> StartTerminalName: type xs:string EndTerminalName: type xs:string StartDifferentialPairName: type xs:string EndDifferentialPairName: type xs:string StartLogicalGroupName: type xs:string EndLogicalGroupName: type xs:string FunctionID: type xs:string StartStandardTerminalName: type xs:string EndStandardTerminalName: type xs:string
type	Terminal-to-TerminalSignalPath-ArrayType, SignalPathCondition-ArrayType, SignalStateType, Terminal-to-TerminalSignalPathType, Uni-directionalType, Bi-directionalType.

The example shown here is a uni-directional terminal to terminal signal path for a clock buffer.

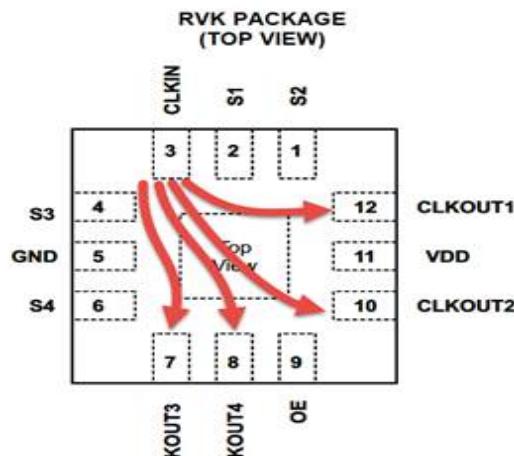


Figure 34 — Clock Buffer

4.7.2.3.7.1 Uni-directional Terminal-to-Terminal Signal Path (cont'd)

```
<Terminal-to-TerminalSignalPath-Array>
  <SignalPathCondition>
    <TerminalName>OE</TerminalName>
    <SignalSate>LowState</SignalSate>
  </SignalPathCondition>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>S1</TerminalName>
      <SignalSate>HighState</SignalSate>
    </SignalPathCondition>
    <Uni-directional>
      <StartTerminalName>CLKIN</StartTerminalName>
      <EndTerminalName>CLKOUT1</EndTerminalName>
    </Uni-directional>
  </Terminal-to-TerminalSignalPath>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>S2</TerminalName>
      <SignalSate>HighState</SignalSate>
    </SignalPathCondition>
    <Uni-directional>
      <StartTerminalName>CLKIN</StartTerminalName>
      <EndTerminalName>CLKOUT2</EndTerminalName>
    </Uni-directional>
  </Terminal-to-TerminalSignalPath>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>S3</TerminalName>
      <SignalSate>HighState</SignalSate>
    </SignalPathCondition>
    <Uni-directional>
      <StartTerminalName>CLKIN</StartTerminalName>
      <EndTerminalName>CLKOUT3</EndTerminalName>
    </Uni-directional>
  </Terminal-to-TerminalSignalPath>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>S4</TerminalName>
      <SignalSate>HighState</SignalSate>
    </SignalPathCondition>
    <Uni-directional>
      <StartTerminalName>CLKIN</StartTerminalName>
      <EndTerminalName>CLKOUT4</EndTerminalName>
    </Uni-directional>
  </Terminal-to-TerminalSignalPath>
</Terminal-to-TerminalSignalPath-Array>
```

4.7.2.3.7.1 Uni-directional Terminal-to-Terminal Signal Path (cont'd)

In this differential multiplexer example, the Differential Pair *Terminal-to-TerminalSignalPath* map could be captured by using the *DifferentialPair/Name*. This would make the data representation more condense.

```

<Terminal-to-TerminalSignalPath-Array>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>IN_SEL</TerminalName>
      <SignalSate>LowState</SignalSate>
    </SignalPathCondition>
    <Uni-directional>
      <StartDifferentialPairName>IN0</StartDifferentialPairName>
        <EndDifferentialPairName>OUT0</EndDifferentialPairName>
      </Uni-directional>
      <Uni-directional>
        <StartDifferentialPairName>IN0</StartDifferentialPairName>
        <EndDifferentialPairName>OUT1</EndDifferentialPairName>
      </Uni-directional>
      <Uni-directional>
        <StartDifferentialPairName>IN0</StartDifferentialPairName>
        <EndDifferentialPairName>OUT2</EndDifferentialPairName>
      </Uni-directional>
      <Uni-directional>
        <StartDifferentialPairName>IN0</StartDifferentialPairName>
        <EndDifferentialPairName>OUT3</EndDifferentialPairName>
      </Uni-directional>
    </Terminal-to-TerminalSignalPath>
    <Terminal-to-TerminalSignalPath>
      <SignalPathCondition>
        <TerminalName>IN_SEL</TerminalName>
        <SignalSate>HighState</SignalSate>
      </SignalPathCondition>
      <Uni-directional>
        <StartDifferentialPairName>IN1</StartDifferentialPairName>
        <EndDifferentialPairName>OUT0</EndDifferentialPairName>
      </Uni-directional>
      <Uni-directional>
        <StartDifferentialPairName>IN1</StartDifferentialPairName>
        <EndDifferentialPairName>OUT1</EndDifferentialPairName>
      </Uni-directional>
      <Uni-directional>
        <StartDifferentialPairName>IN1</StartDifferentialPairName>
        <EndDifferentialPairName>OUT2</EndDifferentialPairName>
      </Uni-directional>
      <Uni-directional>
        <StartDifferentialPairName>IN1</StartDifferentialPairName>
        <EndDifferentialPairName>OUT3</EndDifferentialPairName>
      </Uni-directional>
    </Terminal-to-TerminalSignalPath>
  </Terminal-to-TerminalSignalPath-Array>

```

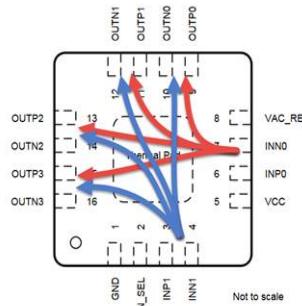


Figure 35 —
Differential Multiplexer

4.7.2.3.7.2. Bi-directional Terminal-to-Terminal Signal Path

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/Terminal-to-TerminalSignalPath-Array/Terminal-to-TerminalSignalPath/Bi-directional
diagram	<p>The diagram illustrates a function block for a bi-directional terminal-to-terminal signal path. It features a central 'Bi-directional' block (type Bi-directionalType) connected to four parallel signal paths. Each path is composed of a sequence of blocks: TerminalName1, DifferentialPairName1, Logical-GroupName1, FunctionID, and StandardTerminalName1. The connections are represented by small circles at the ends of lines.</p>
type	Terminal-to-TerminalSignalPath-ArrayType, SignalPathCondition-ArrayType, SignalStateType, Terminal-to-TerminalSignalPathType, Uni-directionalType, Bi-directionalType.

Table 2 shows the logic for the operation of the function block shown in Figure 33 — Function Block Diagram of a 12-bit Bus Switching Device. This is used to provide an example of the Signal Path that changes upon a set of conditions for a bidirectional connection.

4.7.2.3.7.2 Bi-directional Terminal-to-Terminal Signal Path (cont'd)

Table 1 — Switch Function Table

SEL 1	SEL 2	EN	Bx	Ax	Cx
X	X	H	Z	Z	Z
L	L	L		↔	Z
L	H	L		→	Z
H	L	L	Z	↔	
H	H	L	Z	↔	

```

<Terminal-to-TerminalSignalPath-Array>
  <SignalPathCondition>
    <TerminalName>EN</TerminalName>
    <SignalSate>LowState</SignalSate>
  </SignalPathCondition>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>SEL 1</TerminalName>
      <SignalSate>LowState</SignalSate>
    </SignalPathCondition>
    <SignalPathCondition>
      <TerminalName>SEL2</TerminalName>
      <SignalSate>LowState</SignalSate>
    </SignalPathCondition>
    <Bi-directional>
      <Logical-GroupName1>A</Logical-GroupName1>
      <Logical-GroupName2>B</Logical-GroupName2>
    </Bi-directional>
  </Terminal-to-TerminalSignalPath>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>SEL 1</TerminalName>
      <SignalSate>LowState</SignalSate>
    </SignalPathCondition>
    <SignalPathCondition>
      <TerminalName>SEL2</TerminalName>
      <SignalSate>HighState</SignalSate>
    </SignalPathCondition>
    <Uni-directional>
      <StartLogical-GroupName>B</StartLogical-GroupName>
      <EndLogical-GroupName>A</EndLogical-GroupName>
    </Uni-directional>
  </Terminal-to-TerminalSignalPath>
  <Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
      <TerminalName>SEL 1</TerminalName>
      <SignalSate>HighState</SignalSate>
    </SignalPathCondition>

```

4.7.2.3.7.2 Bi-directional Terminal-to-Terminal Signal Path (cont'd)

```
<SignalPathCondition>
    <TerminalName>SEL2</TerminalName>
    <SignalSate>LowState</SignalSate>
</SignalPathCondition>
<Bi-directional>
    <Logical-GroupName1>A</Logical-GroupName1>
    <Logical-GroupName2>C</Logical-GroupName2>
</Bi-directional>
</Terminal-to-TerminalSignalPath>
<Terminal-to-TerminalSignalPath>
    <SignalPathCondition>
        <TerminalName>SEL 1</TerminalName>
        <SignalSate>HighState</SignalSate>
    </SignalPathCondition>
    <SignalPathCondition>
        <TerminalName>SEL2</TerminalName>
        <SignalSate>HighState</SignalSate>
    </SignalPathCondition>
    <Uni-directional>
        <StartLogical-GroupName>C</StartLogical-GroupName>
        <EndLogical-GroupName>A</EndLogical-GroupName>
    </Uni-directional>
</Terminal-to-TerminalSignalPath>
</Terminal-to-TerminalSignalPath-Array>
```

4.7.2.4. External Connection - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/ExternalConnection-Array
diagram	<pre> classDiagram class ExternalConnection-ArrayType { ExternalConnection-Array } class ExternalConnectionType { TerminalName TerminalNumber } class Condition { type ExternalConnectionConditionType } class ExternalConnection { type ExternalConnectionType } class LeaveFloating class DCBlock class TieToPower class TieToGround class Decoupling class Pullup class Pulldown ExternalConnection-Array "1..*" -- "1..*" ExternalConnection ExternalConnection "*" -- "*" Condition </pre>
type	ExternalConnection–ArrayType, ExternalConnectionType, ExternalConnectionConditionType, Tie-to-PowerExternalConnectionType, Tie-to-GroundExternalConnectionType, DecouplingType, Pullup-to-PowerType, Pulldown-to-GroundType, JEP30-D10:EmptyType.

There are several different types of *ExternalConnection* that may be mandatory for the operation of the device. Some terminals are *Leave-Floating*, because for example they are *Reserved* or they have *NoDieConnection*. For details on these terminal properties see section 4.5.2.1 Properties - Array above. Other reasons may also exist for terminals to be left floating.

DC-Block prevents the flow of direct current (DC) through radio frequency (RF) circuits. They serve as high-pass filters that prevent DC voltages, which have a frequency of zero Hertz (Hz), from interfering with sensitive RF components such as receivers. In other words, DC blocks are capacitors in series with a transmission line. They prevent the flow of DC energy while allowing RF signals to pass with little (if any) attenuation.

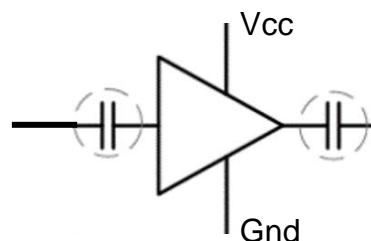


Figure 36 — Circuit with DC-Blocking

Tie-to-Power or *Tie-to-Ground* may be external requirements for terminals that will be unused in a circuit. In this case the condition of *Unused* is set. Other types of External Connection such as *Decoupling*, *Pullup* and *Pulldown* are described in the next sections.

4.7.2.4.1. Decoupling

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/ExternalConnection-Array/ExternalConnection/Decoupling
diagram	<pre> classDiagram class Decoupling { <<DecouplingType>> } class DecouplingRecommendationType { <<DecouplingRecommendationType>> <<GroundTerminalName>> <<Description>> <<Capacitance>> } class CapacitanceType { <<Capacitance>> <<CapacitanceUOMType>> } Decoupling "1..>" DecouplingRecommendationType DecouplingRecommendationType "1..>" GroundTerminalName DecouplingRecommendationType "1..>" Description DecouplingRecommendationType "1..>" Capacitance Capacitance "1..>" CapacitanceUOMType </pre>
type	DecouplingType , DecouplingRecommendationType , CapacitanceType , CapacitanceUOMType .

Active devices of an electronic system (transistors, ICs, vacuum tubes, for example) are connected to their power supplies through conductors with finite resistance and inductance. If the current drawn by an active device changes, voltage drops from power supply to device will also change due to these impedances. If several active devices share a common path to the power supply, changes in the current drawn by one element may produce voltage changes large enough to affect the operation of others - voltage spikes or ground bounce, for example - so the change of state of one device is coupled to others through the common impedance to the power supply. A [Decoupling](#) capacitor provides a bypass path for transient currents, instead of flowing through the common impedance.

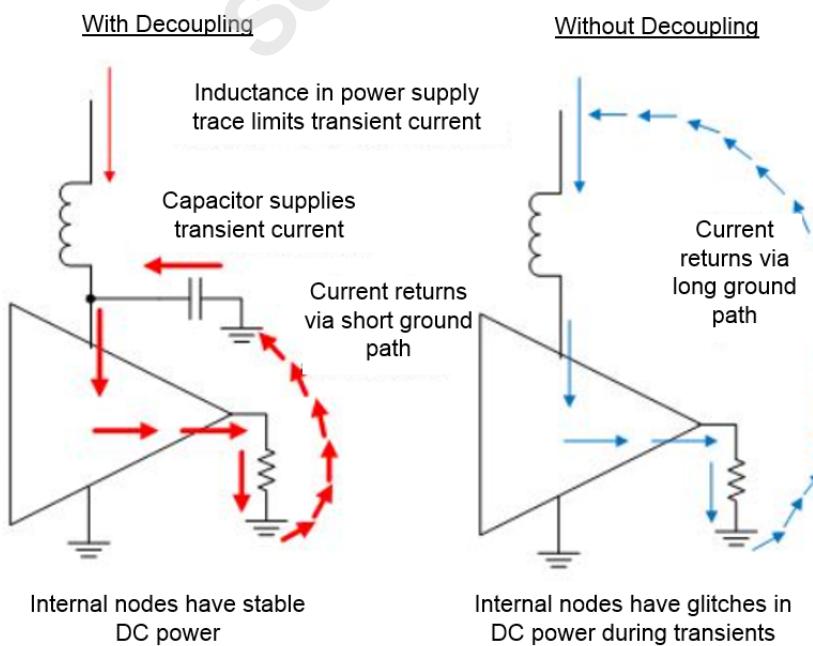


Figure 37 — Decoupling Circuit

4.7.2.4.2. Pullup to Power

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/ExternalConnection-Array/ExternalConnection/Pullup
diagram	
type	Pull-up-to-PowerType , Pull-up-to-PowerRecommendationType , ResistanceType , ResistanceUOMType , PowerType , JEP30-D10:PowerUOMType , CurrentType , JEP30-D10:CurrentUOMType .

In electronic logic circuits, a *Pullup* resistor is a resistor connected between a signal conductor and a positive power supply voltage (*PowerTerminalName*) to ensure that the signal will be a valid logic level if external devices are disconnected or high-impedance is introduced. They may also be used at the interface between two different types of logic devices, possibly operating at different logic levels and power supply voltages.

A *Pullup* resistor pulls the voltage of the signal it is connected to towards its voltage source level. When the other components associated with the signal are inactive, the voltage supplied by the *Pullup* prevails and brings the signal up to a logical high level. When another component on the line goes active, it overrides the *Pullup* resistor. The *Pullup* resistor ensures that the net is at a defined logic level even if no active devices are connected to it. A *Pullup* can also be achieved through a *CurrentSource*.

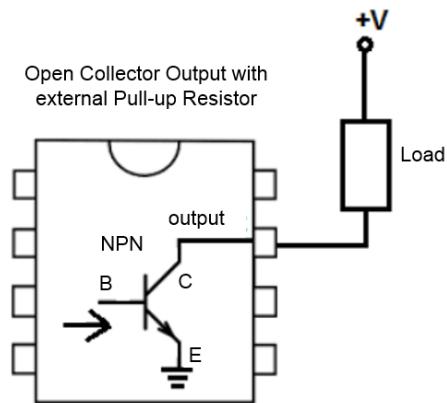


Figure 38 — Pull-up Resistor to Power

4.7.2.4.3. Pulldown to Ground

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/ExternalConnection-Array/ExternalConnection/Pulldown</code>
diagram	
type	<code>Pull-down-to-GroundType, Pull-up-to-GroundRecommendationType, ResistanceType, ResistanceUOMType, PowerType, JEP30-D10:PowerUOMType, CurrentType.</code>

Similar to a *Pullup* resistor, a *Pulldown* resistor or *CurrentSource* is connected between a signal conductor and ground to ensure that the signal will be a valid logic level if external devices are disconnected or high-impedance is introduced.

Whereas Figure 38 shows a resistor as the pull-up component, Figure 39 shows a current source in place of a resistor.

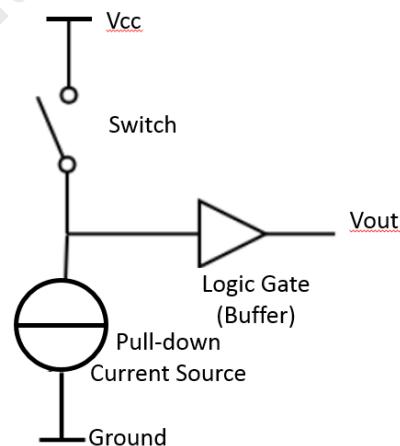


Figure 39 — Pull-down Current Source to Ground

4.7.3. Function Group - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array
diagram	<p>The diagram illustrates the UML Class Diagram for the FunctionGroup-ArrayType. It features a central class FunctionType with a primary attribute ID of type xs:string. A multiplicity of 1..* connects FunctionType to a class Function, which in turn connects to a class SuperInterface-Array via a multiplicity of 1..*. The SuperInterface-Array class contains a list of function types, each represented by a small square icon. These include Amplifier, Audio, Capacitor, Diode, Filter, FrequencySource, Fuse, Inductor, Interface, NonLinear, Optoelectronic, Relay, Resistor, RF, Source, Switch, Thyristor, Transformer, Transistor, and OtherStandard. A constraint labeled constraints is associated with the FunctionType class.</p>
type	FunctionGroup-ArrayType , SuperInterface-ArrayType , FunctionType , AudioFunctionType , AmplifierFunctionType , CapacitorFunctionType , DiodeFunctionType , FilterFunctionType , FrequencySourceFunctionType , FuseFunctionType , InductorFunctionType , InterfaceFunctionType , NonLinearFunctionType , OptoelectronicFunctionType , RelayFunctionType , ResistorFunctionType , SourceFunctionType , SwitchFunctionType , ThyristorFunctionType , TransformerFunctionType , TransistorFunctionType , FunctionMap-to-StandardNameType .

4.7.3 Function Group – Array (cont'd)

The *FunctionGroup-Array* section captures some of the basic internal functions of the device. Simple discrete parts might consist of a single function, while other may have an array of such functions. Other more complex parts can multiple different functions, while at the furthest extreme, some parts can have many millions of internal functions. This section is not intended to capture all the functions that a device may contain, however it is intended to capture sufficient information that would improve the efficiency of the software tools that would consume this data.

The *Function* is an unbounded element, providing the capability to define a single or multiple functions for a single device.

The principal objective for every function type listed under the Function Group is to capture the *StandardTerminalNameAssignment* mapping to the terminals on the device. Section 4.7.2 Package Terminal Map below captures the *TerminalName* as defined by the device manufacturer and maps that to the *TerminalNumber*. This section captures that mapping defined by the device manufacturer, over to the *StandardTerminalNameAssignment* as defined by JEDEC and other standard bodies.

Figure 40 shows a device that contains 3 different functions connected via an internal node A. The functions are:

1. Function R1
 - a. Resistor, Fixed, $R = 10k$,
 - b. Terminal1 = Internal Node A,
 - c. Terminal2 = External Terminal 1
2. Function R2
 - a. Resistor, Fixed, $R = 10k$,
 - b. Terminal1 = Internal Node A,
 - c. Terminal2 = External Terminal 2
3. Function Q1
 - a. Transistor, Bipolar Junction, NPN,
 - b. Base = Internal Node A,
 - c. Collector = External Terminal 3,
 - d. Emitter = External Terminal 2,
4. Device Specification
 - a. $I_{CMax} = 100mA$, $f_T = 250MHz$, etc...

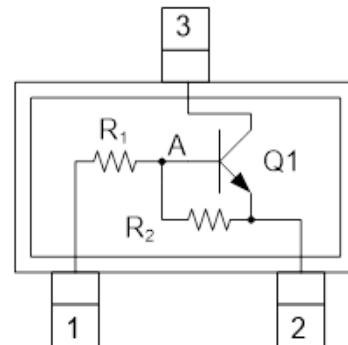


Figure 40 — Sample Transistor Circuit

The above data is shown in the XML structure below for the device connectivity to its various functions. Note how the electrical specification for resistors R1 and R2 are referenced with the function, whereas the electrical specification for the device is outside of the *FunctionGroup-Array* since this specific specification is for the entire device and not a specific function within the device.

While 4.5.1.3 Electrical Classification captures the data associated with the primary classification, a part may have many different functions, to its device *PartClassification/Electrical* classification above. These *Function* classifications are required to capture the *TerminalName* mapping to the standard terminal names for each function within the device. For example, Figure 40 shows functions for a device that may be classified under the *PartClassification/Electrical* classification as an NPN Bipolar Junction Transistor.

4.7.3 Function Group – Array (cont'd)

```
<Electrical-Array>
  <Electrical>
    <ID>NPN Device ID 1</ID>
    <TerminalDetails>
      <TerminalMap-Array>
        <TerminalMap>
          <Map>
            <TerminalName>B</TerminalName>
            <TerminalNumber>1</TerminalNumber>
          </Map>
          <Map>
            <TerminalName>E</TerminalName>
            <TerminalNumber>2</TerminalNumber>
          </Map>
          <Map>
            <TerminalName>C</TerminalName>
            <TerminalNumber>3</TerminalNumber>
          </Map>
        </TerminalMap>
      </TerminalMap-Array>
      <InternalNode-Array>
        <InternalNode>
          <Name>A</Name>
        </InternalNode>
      </InternalNode-Array>
    </TerminalDetails>
  <FunctionalGroup-Array>
    <Function>
      <Resistor>
        <Fixed>
          <StandardTerminalNameAssignment>
            <Terminal1>
              <TerminalNumber>1</TerminalNumber>
            </Terminal1>
            <Terminal2>
              <InternalNodeName>A</InternalNodeName>
            </Terminal2>
          </StandardTerminalNameAssignment>
        </Fixed>
      </Resistor>
      <ElectricalSpecificationID>Res ID 1</ElectricalSpecificationID>
    </Function>
    <Function>
      <Resistor>
        <Fixed>
          <StandardTerminalNameAssignment>
            <Terminal1>
              <TerminalNumber>2</TerminalNumber>
            </Terminal1>
            <Terminal2>
              <InternalNodeName>A</InternalNodeName>
            </Terminal2>
          </StandardTerminalNameAssignment>
        </Fixed>
```

4.7.3 Function Group – Array (cont'd)

```
</Fixed>
</Resistor>
<ElectricalSpecificationID>Res ID 1</ElectricalSpecificationID>
</Function>
<Function>
  <Transistor>
    <BipolarJunction>
      <NPN>
        <StandardTerminalNameAssignment>
          <Base>
            <InternalNodeName>A</InternalNodeName>
          </Base>
          <Collector>
            <TerminalNumber>3</TerminalNumber>
          </Collector>
          <Emitter>
            <TerminalNumber>3</TerminalNumber>
          </Emitter>
        </StandardTerminalNameAssignment>
      </NPN>
    </BipolarJunction>
  </Transistor>
</Function>
</FunctionalGroup-Array>
</Electrical>
</Electrical-Array>
<ElectricalSpecification-Array>
  <ElectricalSpecification>
    <ID>Res ID 1</ID>
    <ParameterSet>
      <Parameter>
        <LaTeX-Symbol>R</LaTeX-Symbol>
        <SymbolDescription>Resistance Value</SymbolDescription>
        <Values>
          <Nominal>10</Nominal>
        </Values>
        <Units>
          <ElectricResistance>kΩ</ElectricResistance>
        </Unitss>
      </Parameter>
    </ParameterSet>
  </ElectricalSpecification>
</ElectricalSpecification-Array>
```

4.7.3.1. Super Interface - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/SuperInterface-Array
diagram	<p>The diagram illustrates the UML Class Diagram for the SuperInterface-ArrayType. It shows the following classes and their associations:</p> <ul style="list-style-type: none"> FunctionGroup-ArrayType: A class with a multiplicity of 1..* at its end. SuperInterface-ArrayType: A class with a multiplicity of 1..* at its end. SuperInterfaceType: A class with two associations: <ul style="list-style-type: none"> One to Function with multiplicity 1..*. One to Sequence with multiplicity 1..*. Sequence: A class with a multiplicity of 1..* at its end. SuperInterfaceSequenceType: A class with one association to Sequence with multiplicity 1..*. Function: A class with a multiplicity of 1..* at its end. constraints: A note indicating constraints for the associations.
type	SuperInterface-ArrayType , SuperInterfaceType , SuperInterfaceSequenceType

A **SuperInterface-Array** is a collection of **Functions** (typically Interfaces) and merge them into one larger group of **Functions/Interfaces**, so that software tools can reference their collection as one group.

A typical example of how this may be used would be in a multi-channel device, in which each channel was made up a set of Interfaces and functions. Let's assume the following example. A device has 2 or more Banks, where each Bank contains 3 channels A to C. Each channel consists of a DDR6 interface, and a PCIe Interface. Channels A through C are swappable, whereas the Banks are not swapable, but do share the same functions and specifications. In this scenario, it makes sense to create a SuperInterface instance for the Channel. The Bank SuperInterface is then made up of 3 Channel Super Interfaces.

```

<SuperInterface-Array>
  <SuperInterface>
    <ID>Super Interface ID 1</ID>
    <Name>Channel</Name>
    <Sequence>
      <FunctionID>DDR6 ID</FunctionID>
      <FunctionID>PCIe ID</FunctionID>
    </Sequence>
  </SuperInterface>
  <SuperInterface>
    <ID>Super Interface ID 2</ID>
    <Name>Bank</Name>
    <Sequence>
      <SuperInterfaceID>Super Interface ID 1</SuperInterfaceID>
      <SuperInterfaceID>Super Interface ID 1</SuperInterfaceID>
      <SuperInterfaceID>Super Interface ID 1</SuperInterfaceID>
    </Sequence>
  </SuperInterface>
</SuperInterface-Array>

```

4.7.3.2. Amplifier

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier
diagram	<pre> classDiagram class Amplifier { type AmplifierFunctionType } class DifferentialInputAmplifierType { type DifferentialInput } class SingleEndedInputAmplifierType { type SingleEndedInput } class DifferentialOutput { type DifferentialInputDifferentialOutput } class SingleEndedOutput { type DifferentialInputSingleEndedOutput } Amplifier "1..n" --> "1..n" DifferentialInputAmplifierType Amplifier "1..n" --> "1..n" SingleEndedInputAmplifierType DifferentialInputAmplifierType "1..n" --> "1..n" DifferentialOutput DifferentialInputAmplifierType "1..n" --> "1..n" SingleEndedOutput SingleEndedInputAmplifierType "1..n" --> "1..n" DifferentialOutput SingleEndedInputAmplifierType "1..n" --> "1..n" SingleEndedOutput </pre>
type	AmplifierFunctionType , DifferentialInputAmplifierType , DifferentialInputDifferentialOutputAmplifierType , DifferentialInputSingleEndedOutputAmplifierType , SingleEndedInputAmplifierType , SingleEndedInputDifferentialOutputAmplifierType , SingleEndedInputSingleEndedOutputAmplifierType .

An *Amplifier* can have two different types of inputs (*DifferentialInput* and *SingleEndedInput*) and two different types of outputs (*DifferentialOutput* and *SingleEndedOutput*) giving rise to 4 different combinations of inputs and outputs *StandardTerminalNameAssignment*, as shown below.

4.7.3.2.1. Differential Input Differential Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/DifferentialInput/DifferentialOutput																														
diagram	<pre> classDiagram class DifferentialInputDifferentialOutputAmplifierType { type DifferentialInputDifferentialOutput } class StandardTerminalNameAssignment { type DifferentialInputDifferentialOutput } class MandatoryMapping { type DifferentialInputDifferentialOutput } class OptionalMapping { type DifferentialInputDifferentialOutput } class DifferentialInputDifferentialOutputAmplifierMandatoryStandardTerminalMappingType { type DifferentialInputDifferentialOutput } class DifferentialInputDifferentialOutputAmplifierOptionalStandardTerminalMappingType { type DifferentialInputDifferentialOutput } class StandardTerminalName { type DifferentialInputDifferentialOutput } DifferentialInputDifferentialOutputAmplifierType "1..n" --> "1..n" StandardTerminalNameAssignment DifferentialInputDifferentialOutputAmplifierType "1..n" --> "1..n" MandatoryMapping DifferentialInputDifferentialOutputAmplifierType "0..1" --> "1..n" OptionalMapping StandardTerminalNameAssignment "1..n" --> "1..n" DifferentialInputDifferentialOutputAmplifierMandatoryStandardTerminalMappingType StandardTerminalNameAssignment "1..n" --> "1..n" DifferentialInputDifferentialOutputAmplifierOptionalStandardTerminalMappingType DifferentialInputDifferentialOutputAmplifierMandatoryStandardTerminalMappingType "1..n" --> "1..n" StandardTerminalName DifferentialInputDifferentialOutputAmplifierOptionalStandardTerminalMappingType "1..n" --> "1..n" StandardTerminalName </pre>																														
type	DifferentialInputDifferentialOutputAmplifierType , DifferentialInputDifferentialOutputAmplifierStandardTerminalNameAssignmentType , DifferentialInputDifferentialOutputAmplifierMandatoryStandardTerminalMappingType , DifferentialInputDifferentialOutputAmplifierMandatoryStandardTerminalNameType , DifferentialInputDifferentialOutputAmplifierOptionalStandardTerminalMappingType , DifferentialInputDifferentialOutputAmplifierOptionalStandardTerminalNameType .																														
list of enumerate values	<table border="1"> <tr> <td colspan="6">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Positive Input</td> <td>2. Negative Input</td> <td>3. Positive Output</td> <td>4. Negative Output</td> <td>5. Negative Rail</td> <td>6. Positive Rail</td> </tr> <tr> <td colspan="6">7. Common</td> </tr> <tr> <td colspan="6">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Gain Resistor Terminal 1</td> <td>2. Gain Resistor Terminal 2</td> <td>3. Offset Node 1</td> <td>4. Offset Node 2</td> <td colspan="2"></td> </tr> </table>	MandatoryMapping/StandardTerminalName						1. Positive Input	2. Negative Input	3. Positive Output	4. Negative Output	5. Negative Rail	6. Positive Rail	7. Common						OptionalMapping/StandardTerminalName						1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2		
MandatoryMapping/StandardTerminalName																															
1. Positive Input	2. Negative Input	3. Positive Output	4. Negative Output	5. Negative Rail	6. Positive Rail																										
7. Common																															
OptionalMapping/StandardTerminalName																															
1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2																												

4.7.3.2.2. Differential Input Single Ended Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/DifferentialInput/SingleEndedOutput																				
diagram	<pre> classDiagram class DifferentialInputSingleEndedOutputAmplifierType { <<DifferentialInputSingleEndedOutputAmplifierStandardTerminalNameAssignment>> <<DifferentialInputSingleEndedOutputAmplifierMandatoryStandardTerminalMapping>> <<DifferentialInputSingleEndedOutputAmplifierOptionalStandardTerminalMapping>> <<constraints>> } class SingleEndedOutput { <<DifferentialInputSingleEndedOutput>> } class StandardTerminalNameAssignment { <<DifferentialInputSingleEndedOutput>> } class MandatoryMapping { <<DifferentialInputSingleEndedOutput>> } class OptionalMapping { <<DifferentialInputSingleEndedOutput>> } class StandardTerminalName { <<DifferentialInputSingleEndedOutput>> } DifferentialInputSingleEndedOutputAmplifierType "1..n" -- "1..n" StandardTerminalNameAssignment : <<DifferentialInputSingleEndedOutput>> DifferentialInputSingleEndedOutputAmplifierType "6" -- "6" MandatoryMapping : <<DifferentialInputSingleEndedOutput>> DifferentialInputSingleEndedOutputAmplifierType "0..4" -- "0..4" OptionalMapping : <<DifferentialInputSingleEndedOutput>> DifferentialInputSingleEndedOutputAmplifierType "1..n" -- "1..n" constraints : <<constraints>> SingleEndedOutput --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> MandatoryMapping StandardTerminalNameAssignment --> OptionalMapping MandatoryMapping --> StandardTerminalName </pre>																				
type	DifferentialInputSingleEndedOutputAmplifierType , DifferentialInputSingleEndedOutputAmplifierStandardTerminalNameAssignmentType , DifferentialInputSingleEndedOutputAmplifierMandatoryStandardTerminalMappingType , DifferentialInputSingleEndedOutputAmplifierOptionalStandardTerminalMappingType , DifferentialInputSingleEndedOutputAmplifierMandatoryStandardTerminalNameType , DifferentialInputSingleEndedOutputAmplifierOptionalStandardTerminalNameType .																				
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Positive Input</td> <td>2. Negative Input</td> <td>3. Output</td> <td>4. Negative Rail</td> </tr> <tr> <td>5. Positive Rail</td> <td>6. Common</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Gain Resistor Terminal 1</td> <td>2. Gain Resistor Terminal 2</td> <td>3. Offset Node 1</td> <td>4. Offset Node 2</td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Positive Input	2. Negative Input	3. Output	4. Negative Rail	5. Positive Rail	6. Common			OptionalMapping/StandardTerminalName				1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2
MandatoryMapping/StandardTerminalName																					
1. Positive Input	2. Negative Input	3. Output	4. Negative Rail																		
5. Positive Rail	6. Common																				
OptionalMapping/StandardTerminalName																					
1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2																		

4.7.3.2.3. Single Ended Input Differential Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/SingleEndedInput/DifferentialOutput												
diagram	<pre> classDiagram class SingleEndedInputDifferentialOutputAmplifierType { <<Single Ended Input Differential Output Amplifier Type>> } class DifferentialOutput { <<Single Ended Input Differential Output>> } class StandardTerminalNameAssignment { <<Single Ended Input Differential Output Standard Terminal Name Assignment Type>> } class MandatoryMapping { <<Single Ended Input Differential Output Amplifier Mandatory Standard Terminal Mapping Type>> } class OptionalMapping { <<Single Ended Input Differential Output Amplifier Optional Standard Terminal Mapping Type>> } class StandardTerminalName { <<Single Ended Input Differential Output Standard Terminal Name>> } SingleEndedInputDifferentialOutputAmplifierType "1..x" --> DifferentialOutput : SingleEndedInputDifferentialOutputAmplifierType "1..x" --> StandardTerminalNameAssignment : SingleEndedInputDifferentialOutputAmplifierType "6" --> MandatoryMapping : SingleEndedInputDifferentialOutputAmplifierType "0..4" --> OptionalMapping : constraints </pre>												
type	SingleEndedInputDifferentialOutputAmplifierType , SingleEndedInputDifferentialOutputAmplifierStandardTerminalNameAssignmentType , SingleEndedInputDifferentialOutputAmplifierMandatoryStandardTerminalMappingType , SingleEndedInputDifferentialOutputAmplifierMandatoryStandardTerminalNameType , SingleEndedInputDifferentialOutputAmplifierOptionalStandardTerminalMappingType , SingleEndedInputDifferentialOutputAmplifierOptionalStandardTerminalNameType .												
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. Input</td> <td>2. Positive Output</td> <td>3. Negative Output</td> <td>4. Negative Rail</td> </tr> <tr> <td>5. Positive Rail</td> <td>6. Common</td> <td></td> <td></td> </tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. Gain Resistor Terminal 1</td> <td>2. Gain Resistor Terminal 2</td> <td>3. Offset Node 1</td> <td>4. Offset Node 2</td> </tr> </table>	1. Input	2. Positive Output	3. Negative Output	4. Negative Rail	5. Positive Rail	6. Common			1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2
1. Input	2. Positive Output	3. Negative Output	4. Negative Rail										
5. Positive Rail	6. Common												
1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2										

4.7.3.2.4. Single Ended Input Single Ended Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/SingleEndedInput/SingleEndedOutput																				
diagram	<pre> classDiagram class SingleEndedInputSingleEndedOutputAmplifierType { <<Single Ended Input Single Ended Output Amplifier Type>> } class StandardTerminalNameAssignment { <<Single Ended Input Single Ended Out...>> } class MandatoryMapping { <<Single Ended Input Single Ended Out...>> } class OptionalMapping { <<Single Ended Input Single Ended Out...>> } class StandardTerminalName { <<Single Ended Input Single Ended Out...>> } SingleEndedInputSingleEndedOutputAmplifierType < -- SingleEndedInputSingleEndedOutputAmplifierStandardTerminalNameAssignmentType SingleEndedInputSingleEndedOutputAmplifierType < -- SingleEndedInputSingleEndedOutputAmplifierMandatoryStandardTerminalMappingType SingleEndedInputSingleEndedOutputAmplifierType < -- SingleEndedInputSingleEndedOutputAmplifierOptionalStandardTerminalMappingType SingleEndedInputSingleEndedOutputAmplifierType "1..x" --> StandardTerminalNameAssignment : StandardTerminalNameAssignment SingleEndedInputSingleEndedOutputAmplifierType "1..1" --> MandatoryMapping : MandatoryMapping SingleEndedInputSingleEndedOutputAmplifierType "0..4" --> OptionalMapping : OptionalMapping StandardTerminalNameAssignment --> StandardTerminalName : StandardTerminalName </pre>																				
type	SingleEndedInputDifferentialOutputAmplifierType , SingleEndedInputSingleEndedOutputAmplifierStandardTerminalNameAssignmentType , SingleEndedInputSingleEndedOutputAmplifierMandatoryStandardTerminalMappingType , SingleEndedInputSingleEndedOutputAmplifierMandatoryStandardTerminalNameType , SingleEndedInputSingleEndedOutputAmplifierOptionalStandardTerminalMappingType , SingleEndedInputSingleEndedOutputAmplifierOptionalStandardTerminalNameType .																				
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Input</td> <td>2. Output</td> <td>3. Negative Rail</td> <td>4. Positive Rail</td> </tr> <tr> <td>5. Common</td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Gain Resistor Terminal 1</td> <td>2. Gain Resistor Terminal 2</td> <td>3. Offset Node 1</td> <td>4. Offset Node 2</td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Input	2. Output	3. Negative Rail	4. Positive Rail	5. Common				OptionalMapping/StandardTerminalName				1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2
MandatoryMapping/StandardTerminalName																					
1. Input	2. Output	3. Negative Rail	4. Positive Rail																		
5. Common																					
OptionalMapping/StandardTerminalName																					
1. Gain Resistor Terminal 1	2. Gain Resistor Terminal 2	3. Offset Node 1	4. Offset Node 2																		

4.7.3.3. Audio

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio
diagram	<pre> classDiagram class AudioFunctionType { <<Audio Function Type>> } class Microphone { <<Microphone Function Type>> } class Speaker { <<Speaker Function Type>> } class MicrophoneFunctionType { <<Microphone Function Type>> } class Analog { <<Analog Microphone Function Type>> } class Digital { <<Digital Microphone Function Type>> } AudioFunctionType < -- MicrophoneFunctionType MicrophoneFunctionType < -- Analog MicrophoneFunctionType < -- Digital AudioFunctionType < -- Speaker </pre>
type	AudioFunctionType , MicrophoneFunctionType , AnalogMicrophoneFunctionType , DigitalMicrophoneFunctionType , SpeakerFunctionType .

An [Audio](#) can be one of two types: [Speaker](#) or [Microphone](#), which itself can be of type [Analog](#) or [Digital](#). Each of these types is described in further detail below.

4.7.3.3.1. Analog Microphone

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio/Microphone								
diagram	<pre> classDiagram class Analog { type AnalogMicrophoneFunctionType } class StandardTerminalNameAssignment { type AnalogMicrophoneStandardTerminalNameAssignmentType } class MandatoryMapping { type AnalogMicrophoneMandatoryStandardTerminalMappingType } class StandardTerminalName { type AnalogMicrophoneMandatoryStandardTerminalNameType } class OptionalMapping { type AnalogMicrophoneOptionalStandardTerminalMappingType } class StandardTerminalName { type AnalogMicrophoneOptionalStandardTerminalNameType } Analog "1..∞" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" --> MandatoryMapping MandatoryMapping "2" --> StandardTerminalName StandardTerminalNameAssignment "1..∞" --> OptionalMapping OptionalMapping "0..2" --> StandardTerminalName </pre> <p style="text-align: center;">constraints</p>								
type	AnalogMicrophoneFunctionType, AnalogMicrophoneStandardTerminalNameAssignmentType, AnalogMicrophoneMandatoryStandardTerminalMappingType, AnalogMicrophoneMandatoryStandardTerminalNameType, AnalogMicrophoneOptionalStandardTerminalMappingType, AnalogMicrophoneOptionalStandardTerminalNameType.								
list of enumerate values	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: left;">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td style="width: 50%;">1. Output</td> <td style="width: 50%;">2. Ground</td> </tr> <tr> <td colspan="2">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>3. Case</td> <td>4. Power</td> </tr> </table>	MandatoryMapping/StandardTerminalName		1. Output	2. Ground	OptionalMapping/StandardTerminalName		3. Case	4. Power
MandatoryMapping/StandardTerminalName									
1. Output	2. Ground								
OptionalMapping/StandardTerminalName									
3. Case	4. Power								

4.7.3.3.2. Digital Microphone

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio/Microphone						
diagram	<pre> classDiagram class Digital { type DigitalMicrophoneFunctionType } class StandardTerminalNameAssignment { type DigitalMicrophoneStandardTerminalNameAssignmentType } class Mapping { type DigitalMicrophoneStandardTerminalMappingType } class StandardTerminalName { type DigitalMicrophoneStandardTerminalNameType } Digital "1..∞" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" --> Mapping Mapping "4" --> StandardTerminalName </pre> <p style="text-align: center;">constraints</p>						
type	DigitalMicrophoneFunctionType, DigitalMicrophoneStandardTerminalNameAssignmentType, DigitalMicrophoneStandardTerminalMappingType, DigitalMicrophoneStandardTerminalNameType.						
list of enumerate values	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: left;">Mapping/StandardTerminalName</td> </tr> <tr> <td style="width: 50%;">1. Clock</td> <td style="width: 50%;">2. Data</td> </tr> <tr> <td>3. Ground</td> <td>4. Power</td> </tr> </table>	Mapping/StandardTerminalName		1. Clock	2. Data	3. Ground	4. Power
Mapping/StandardTerminalName							
1. Clock	2. Data						
3. Ground	4. Power						

4.7.3.3.3. Speaker

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio/Speaker												
diagram	<pre> classDiagram class Speaker { <<SpeakerFunctionType>> } class StandardTerminalNameAssignment { <<SpeakerStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<SpeakerMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<SpeakerOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<SpeakerStandardTerminalNameType>> } Speaker < -- SpeakerFunctionType StandardTerminalNameAssignment < -- SpeakerStandardTerminalNameAssignmentType MandatoryMapping < -- SpeakerMandatoryStandardTerminalMappingType OptionalMapping < -- SpeakerOptionalStandardTerminalMappingType StandardTerminalName < -- SpeakerStandardTerminalNameType Speaker --> StandardTerminalNameAssignment : 1..∞ StandardTerminalNameAssignment --> MandatoryMapping : 1..∞ StandardTerminalNameAssignment --> OptionalMapping : 1..∞ MandatoryMapping --> StandardTerminalName : 2..∞ OptionalMapping --> StandardTerminalName : 2..∞ </pre> <p>The diagram shows the UML Class Diagram for the Speaker Function Type. It includes classes for Speaker, StandardTerminalNameAssignment, MandatoryMapping, OptionalMapping, and StandardTerminalName, each with their respective type annotations. Associations show Speaker mapping to StandardTerminalNameAssignment (multiplicity 1..∞), StandardTerminalNameAssignment mapping to MandatoryMapping (multiplicity 1..∞), StandardTerminalNameAssignment mapping to OptionalMapping (multiplicity 1..∞), MandatoryMapping mapping to StandardTerminalName (multiplicity 2..∞), and OptionalMapping mapping to StandardTerminalName (multiplicity 2..∞). A constraint block is also present.</p>												
type	SpeakerFunctionType, SpeakerStandardTerminalNameAssignmentType, SpeakerMandatoryStandardTerminalMappingType, SpeakerOptionalStandardTerminalMappingType, SpeakerOptionalStandardTerminalNameType, SpeakerMandatoryStandardTerminalNameType												
list of enumerate values	<table border="1"> <tr> <td>MandatoryMapping/StandardTerminalName</td> <td></td> <td></td> </tr> <tr> <td>1. Positive</td> <td>2. Negative</td> <td></td> </tr> <tr> <td>OptionalMapping/StandardTerminalName</td> <td></td> <td></td> </tr> <tr> <td>1. Case</td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName			1. Positive	2. Negative		OptionalMapping/StandardTerminalName			1. Case		
MandatoryMapping/StandardTerminalName													
1. Positive	2. Negative												
OptionalMapping/StandardTerminalName													
1. Case													

4.7.3.4. Capacitor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor
diagram	<pre> classDiagram class Capacitor { <<CapacitorFunctionType>> } class FixedNonPolarized { <<FixedNonPolarizedCapacitorType>> } class FixedPolarized { <<FixedPolarizedCapacitorType>> } class DifferentialNonPolarized { <<DifferentialNonPolarizedCapacitorType>> } class FeedThrough { <<FeedThroughCapacitorType>> } class VariableNonPolarized { <<VariableNonPolarizedCapacitorType>> } Capacitor < -- CapactorFunctionType FixedNonPolarized < -- FixedNonPolarizedCapacitorType FixedPolarized < -- FixedPolarizedCapacitorType DifferentialNonPolarized < -- DifferentialNonPolarizedCapacitorType FeedThrough < -- FeedThroughCapacitorType VariableNonPolarized < -- VariableNonPolarizedCapacitorType </pre> <p>The diagram shows the UML Class Diagram for the Capacitor Function Type. It includes classes for Capacitor, FixedNonPolarized, FixedPolarized, DifferentialNonPolarized, FeedThrough, and VariableNonPolarized, each with their respective type annotations. Associations show Capacitor mapping to each of the five capacitor types.</p>
type	CapacitorFunctionType, FixedNonPolarizedCapacitorType, FixedPolarizedCapacitorType, DifferentialNonPolarizedCapacitorType, FeedThroughCapacitorType, VariableNonPolarizedCapacitorType.

A capacitor can be one of the following five types: *FixedNonPolarized*, *FixedPolarized*, *DifferentialNonPolarized*, *FeedThrough*, and *VariableNonPolarized*, each specified in more detail below.

4.7.3.4.1. Fixed Non Polarized

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/FixedNonPolarized							
diagram								
type	FixedNonPolarizedCapacitorType , FixedNonPolarizedCapacitorStandardTerminalNameAssignmentType , FixedNonPolarizedCapacitorStandardTerminalMappingType , FixedNonPolarizedCapacitorStandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Terminal 1</td> <td style="padding: 2px;">2. Terminal 2</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>				1. Terminal 1	2. Terminal 2		
1. Terminal 1	2. Terminal 2							

4.7.3.4.2. Fixed Polarized

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/FixedPolarized							
diagram								
type	FixedPolarizedCapacitorType , FixedPolarizedCapacitorStandardTerminalNameAssignmentType , FixedPolarizedCapacitorStandardTerminalMappingType , FixedPolarizedCapacitorStandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Positive</td> <td style="padding: 2px;">2. Negative</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>				1. Positive	2. Negative		
1. Positive	2. Negative							

4.7.3.4.3. Differential Non Polarized

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/DifferentialNonPolarized							
diagram	<pre> classDiagram class DifferentialNonPolarizedCapacitorType class StandardTerminalNameAssignment class Mapping class StandardTerminalName DifferentialNonPolarizedCapacitorType "1..0" -- "1..0" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..0" -- "1..0" Mapping Mapping "1..0" -- "1..0" StandardTerminalName StandardTerminalNameAssignment < -- constraints Mapping < -- constraints </pre>							
type	DifferentialNonPolarizedCapacitorType , DifferentialNonPolarizedCapacitorStandardTerminalNameAssignmentType , DifferentialNonPolarizedCapacitorStandardTerminalMappingType , DifferentialNonPolarizedCapacitorStandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Terminal 1</td> <td style="padding: 2px;">2. Terminal 2</td> <td style="padding: 2px;">3. Common</td> <td style="padding: 2px;"></td> </tr> </table>				1. Terminal 1	2. Terminal 2	3. Common	
1. Terminal 1	2. Terminal 2	3. Common						

4.7.3.4.4. Feed Through

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/FeedThrough							
diagram	<pre> classDiagram class FeedThroughCapacitorType class StandardTerminalNameAssignment class Mapping class StandardTerminalName FeedThroughCapacitorType "1..0" -- "1..0" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..0" -- "1..0" Mapping Mapping "1..0" -- "1..0" StandardTerminalName StandardTerminalNameAssignment < -- constraints Mapping < -- constraints </pre>							
type	FeedThroughCapacitorType , FeedThroughCapacitorStandardTerminalNameAssignmentType , FeedThroughCapacitorStandardTerminalMappingType , FeedThroughCapacitorStandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Terminal 1</td> <td style="padding: 2px;">2. Terminal 2</td> <td style="padding: 2px;">3. Common</td> <td style="padding: 2px;"></td> </tr> </table>				1. Terminal 1	2. Terminal 2	3. Common	
1. Terminal 1	2. Terminal 2	3. Common						

4.7.3.4.5. Variable Non Polarized

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/VariableNonPolarized				
diagram	<pre> classDiagram class VariableNonPolarizedCapacitorType { VariableNonPolarized } class StandardTerminalNameAssignment { StandardTerminalNameAssignment } class Mapping { Mapping } class StandardTerminalName { StandardTerminalName } VariableNonPolarizedCapacitorType "1..∞" -- "1..∞" StandardTerminalNameAssignment : VariableNonPolarizedCapacitorType "1..∞" -- "2" Mapping : StandardTerminalNameAssignment "2" -- "2" StandardTerminalName : constraints </pre>				
type	VariableNonPolarizedCapacitorType , VariableNonPolarizedCapacitorStandardTerminalNameAssignmentType , VariableNonPolarizedCapacitorStandardTerminalMappingType , VariableNonPolarizedCapacitorStandardTerminalNameType .				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Terminal 1</td> <td>2. Terminal 2</td> <td></td> <td></td> </tr> </table>	1. Terminal 1	2. Terminal 2		
1. Terminal 1	2. Terminal 2				

4.7.3.5. Diode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode
diagram	<pre> classDiagram class Diode { <<Diode>> } class DiodeFunctionType { <<DiodeFunctionType>> } class CurrentRegulator { <<CurrentRegulator>> } class ESD { <<ESD>> } class LED { <<LED>> } class Microwave { <<Microwave>> } class PIN { <<PIN>> } class Rectifier { <<Rectifier>> } class Schottky { <<Schottky>> } class Signal { <<Signal>> } class SiliconCarbide { <<SiliconCarbide>> } class Tunnel { <<Tunnel>> } class Uni_tunnel { <<Uni_tunnel>> } class Varactor { <<Varactor>> } class VoltageRegulator { <<VoltageRegulator>> } class Zener { <<Zener>> } Diode < -- DiodeFunctionType DiodeFunctionType < -- CurrentRegulator DiodeFunctionType < -- ESD DiodeFunctionType < -- LED DiodeFunctionType < -- Microwave DiodeFunctionType < -- PIN DiodeFunctionType < -- Rectifier DiodeFunctionType < -- Schottky DiodeFunctionType < -- Signal DiodeFunctionType < -- SiliconCarbide DiodeFunctionType < -- Tunnel DiodeFunctionType < -- Uni_tunnel DiodeFunctionType < -- Varactor DiodeFunctionType < -- VoltageRegulator DiodeFunctionType < -- Zener CurrentRegulator --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> ESD --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> LED --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Microwave --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> PIN --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Rectifier --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Schottky --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Signal --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> SiliconCarbide --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Tunnel --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Uni_tunnel --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Varactor --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> VoltageRegulator --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> Zener --> StandardTerminalNameAssignment : <<DiodeStandardTerminalNameAssignment>> </pre>

4.7.3.10 Diode (cont'd)

type	DiodeFunctionType, CurrentRegulatorDiodeFunctionType, ESD-DiodeFunctionType, LED-DiodeFunctionType, MicrowaveDiodeFunctionType, PIN-DiodeFunctionType, RectifierDiodeFunctionType, SchottkyDiodeFunctionType, SignalDiodeFunctionType, SiliconCarbideDiodeFunctionType, TunnelDiodeFunctionType, Uni-tunnelDiodeFunctionType, VaractorDiodeFunctionType, VoltageRegulatorDiodeFunctionType, ZenerDiodeFunctionType, DiodeStandardTerminalNameAssignmentType.
------	--

A diode can be one of the following types: *CurrentRegulator*, *ESD*, *LED*, *Microwave*, *PIN*, *Rectifier*, *Schottky*, *Signal*, *SiliconCarbide*, *Tunnel*, *Uni-tunnel*, *Varactor*, *VoltageRegulator*, and *Zener*, each ending with the same type.

4.7.3.5.1. Diode Standard Terminal Name Assignment Type

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/CurrentRegulator/StandardTerminalNameAssignment 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/ESD/StandardTerminalNameAssignment/Mapping/StandardTerminalName 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/LED/StandardTerminalNameAssignment/Mapping/StandardTerminalName 4. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Microwave/StandardTerminalNameAssignment 5. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/PIN /StandardTerminalNameAssignment 6. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Rectifier/StandardTerminalNameAssignment 7. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Schottky/StandardTerminalNameAssignment 8. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Signal/StandardTerminalNameAssignment 9. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/SiliconCarbide/StandardTerminalNameAssignment 10. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Tunnel/StandardTerminalNameAssignment 11. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Uni-tunnel/StandardTerminalNameAssignment 12. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Varactor/StandardTerminalNameAssignment 13. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/VoltageRegulator/StandardTerminalNameAssignment 14. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Zener/StandardTerminalNameAssignment 			
diagram	<pre> classDiagram class CurrentRegulator { <<CurrentRegulatorDiodeFunctionType>> } class StandardTerminalNameAssignment { <<DiodeStandardTerminalNameAssignmentType>> } class Mapping { <<DiodeStandardTerminalMappingType>> } class StandardTerminalName { <<DiodeStandardTerminalNameType>> } CurrentRegulator "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "2" StandardTerminalName </pre>			
type	DiodeStandardTerminalNameAssignmentType, DiodeStandardTerminalMappingType, DiodeStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">1. Anode</td> <td style="width: 33%;">2. Cathode</td> <td style="width: 33%;"></td> </tr> </table>	1. Anode	2. Cathode	
1. Anode	2. Cathode			

4.7.3.6. Filter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Filter								
diagram	<pre> classDiagram class Filter { type FilterFunctionType } class StandardTerminalNameAssignment { type FilterStandardTerminalNameAssignmentType } class MandatoryMapping { type FilterMandatoryStandardTerminalNameMappingType } class OptionalMapping { type FilterOptionalStandardTerminalNameMappingType } class StandardTerminalName { type FilterMandatoryStandardTerminalNameType type FilterOptionalStandardTerminalNameType } Filter "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping StandardTerminalNameAssignment "1..>" OptionalMapping MandatoryMapping "2..>" StandardTerminalName OptionalMapping "k..>" StandardTerminalName </pre>								
type	FilterFunctionType , FilterStandardTerminalNameAssignmentType , FilterStandardTerminalMappingType , FilterStandardTerminalNameType .								
list of enumerate values	<table border="1"> <tr> <td colspan="2">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Terminal 1</td> <td>2. Terminal 2</td> </tr> <tr> <td colspan="2">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Ground</td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName		1. Terminal 1	2. Terminal 2	OptionalMapping/StandardTerminalName		1. Ground	
MandatoryMapping/StandardTerminalName									
1. Terminal 1	2. Terminal 2								
OptionalMapping/StandardTerminalName									
1. Ground									

4.7.3.7. Frequency Source

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource
diagram	<pre> classDiagram class FrequencySource { type FrequencySourceFunctionType } class Generator { type GeneratorFrequencySourceType } class Timer { type TimerFrequencySourceType } class Crystal { type CrystalFrequencySourceType } class Oscillator { type OscillatorFrequencySourceType } class Resonator { type ResonatorFrequencySourceType } class VoltageControlledOscillator { type VoltageControlledOscillatorFrequencySourceType } FrequencySource < -- Generator FrequencySource < -- Timer FrequencySource < -- Crystal FrequencySource < -- Oscillator FrequencySource < -- Resonator FrequencySource < -- VoltageControlledOscillator </pre>
type	FrequencySourceFunctionType , GeneratorFrequencySourceFunctionType , TimerFrequencySourceFunctionType , CrystalFrequencySourceFunctionType , OscillatorFrequencySourceFunctionType , ResonatorFrequencySourceFunctionType , VoltageControlledOscillatorFrequencySourceFunctionType .

A frequency source can be one of the following six types: [Generator](#), [Timer](#), [Crystal](#), [Oscillator](#), [Resonator](#), and [VoltageControlledOscillator](#), each specified in more detail below.

4.7.3.7.1. Generator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Generator								
diagram	<pre> classDiagram class GeneratorFrequencySourceFunctionType { Generator StandardTerminalNameAssignment MandatoryMapping OptionalMapping } class StandardTerminalNameAssignmentType { StandardTerminalNameAssignment } class MandatoryMappingType { MandatoryMapping } class OptionalMappingType { OptionalMapping } Generator < -- GeneratorFrequencySourceFunctionType StandardTerminalNameAssignment < -- StandardTerminalNameAssignmentType MandatoryMapping < -- MandatoryMappingType OptionalMapping < -- OptionalMappingType Generator --> StandardTerminalNameAssignment : 1..0 StandardTerminalNameAssignment --> MandatoryMapping : 1..0 StandardTerminalNameAssignment --> OptionalMapping : 1..0 MandatoryMapping --> StandardTerminalName : 1..0 </pre>								
type	GeneratorFrequencySourceFunctionType , GeneratorFrequencySourceStandardTerminalNameAssignmentType , GeneratorFrequencySourceMandatoryStandardTerminalMappingType , GeneratorFrequencySourceMandatoryStandardTerminalNameType , GeneratorFrequencySourceOptionalStandardTerminalMappingType , GeneratorFrequencySourceOptionalStandardTerminalNameType .								
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. Output</td> <td>2. Ground</td> <td>3. Power</td> <td></td> </tr> </table> OptionalMapping/StandardTerminalName <table border="1"> <tr> <td>1. Enable</td> <td></td> <td></td> <td></td> </tr> </table>	1. Output	2. Ground	3. Power		1. Enable			
1. Output	2. Ground	3. Power							
1. Enable									

4.7.3.7.2. Timer

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Timer																
diagram	<pre> classDiagram class Timer { <<TimerFrequencySourceFunctionType>> } class StandardTerminalNameAssignment { <<TimerFrequencySourceStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<TimerFrequencySourceMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<TimerFrequencySourceOptionalStandardTerminalMappingType>> } class OtherMapping { <<TimerFrequencySourceOtherStandardTerminalMappingType>> } class StandardTerminalName { <<TimerFrequencySourceStandardTerminalNameType>> } Timer "1..oo" -- "0..2" StandardTerminalNameAssignment StandardTerminalNameAssignment "4" -- "1" MandatoryMapping StandardTerminalNameAssignment "0..2" -- "1" OptionalMapping StandardTerminalNameAssignment "0..2" -- "1" OtherMapping MandatoryMapping "1" -- "1" StandardTerminalName OptionalMapping "1" -- "1" StandardTerminalName OtherMapping "1" -- "1" StandardTerminalName </pre>																
type	TimerFrequencySourceFunctionType , TimerFrequencySourceStandardTerminalNameAssignmentType , TimerFrequencySourceMandatoryStandardTerminalMappingType , TimerFrequencySourceMandatoryStandardTerminalNameType , TimerFrequencySourceOptionalStandardTerminalMappingType , TimerFrequencySourceOptionalStandardTerminalNameType , TimerFrequencySourceOtherStandardTerminalMappingType , TimerFrequencySourceOtherStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Output</td> <td>2. Trigger</td> <td>3. Power</td> <td>4. Ground</td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Enable</td> <td>2. Reset</td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Output	2. Trigger	3. Power	4. Ground	OptionalMapping/StandardTerminalName				1. Enable	2. Reset		
MandatoryMapping/StandardTerminalName																	
1. Output	2. Trigger	3. Power	4. Ground														
OptionalMapping/StandardTerminalName																	
1. Enable	2. Reset																

4.7.3.7.3. Crystal

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Crystal																
diagram	<pre> classDiagram class Crystal { <<CrystalFrequencySourceFunctionType>> } class StandardTerminalNameAssignment { <<CrystalFrequencySourceStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<CrystalFrequencySourceMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<CrystalFrequencySourceOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<CrystalFrequencySourceMandatoryStandardTerminalNameType>> } Crystal "1..*" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..*" --> MandatoryMapping StandardTerminalNameAssignment "1..*" --> OptionalMapping MandatoryMapping "2" --> StandardTerminalName OptionalMapping --> StandardTerminalName </pre>																
type	CrystalFrequencySourceFunctionType , CrystalFrequencySourceStandardTerminalNameAssignmentType , CrystalFrequencySourceMandatoryStandardTerminalMappingType , CrystalFrequencySourceMandatoryStandardTerminalNameType , CrystalFrequencySourceOptionalStandardTerminalMappingType , CrystalFrequencySourceOptionalStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Terminal 1</td> <td>2. Terminal 2</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Case</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Terminal 1	2. Terminal 2			OptionalMapping/StandardTerminalName				1. Case			
MandatoryMapping/StandardTerminalName																	
1. Terminal 1	2. Terminal 2																
OptionalMapping/StandardTerminalName																	
1. Case																	

4.7.3.7.4. Oscillator

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Oscillator</code>																
diagram	<pre> classDiagram class Oscillator { <<OscillatorFrequencySourceFunctionType>> } class StandardTerminalNameAssignment { <<OscillatorFrequencySourceStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<OscillatorFrequencySourceMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<OscillatorFrequencySourceOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<OscillatorFrequencySourceStandardTerminalNameType>> } Oscillator "1..n" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..1" --> MandatoryMapping StandardTerminalNameAssignment "1..1" --> OptionalMapping MandatoryMapping "3" --> StandardTerminalName OptionalMapping "3" --> StandardTerminalName </pre>																
type	<code>OscillatorFrequencySourceFunctionType,</code> <code>OscillatorFrequencySourceStandardTerminalNameAssignmentType,</code> <code>OscillatorFrequencySourceMandatoryStandardTerminalMappingType,</code> <code>OscillatorFrequencySourceMandatoryStandardTerminalNameType,</code> <code>OscillatorFrequencySourceOptionalStandardTerminalMappingType,</code> <code>OscillatorFrequencySourceOptionalStandardTerminalNameType.</code>																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Output</td> <td>2. Ground</td> <td>3. Power</td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Enable</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Output	2. Ground	3. Power		OptionalMapping/StandardTerminalName				1. Enable			
MandatoryMapping/StandardTerminalName																	
1. Output	2. Ground	3. Power															
OptionalMapping/StandardTerminalName																	
1. Enable																	

4.7.3.7.5. Resonator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Resonator																
diagram	<pre> classDiagram class ResonatorFrequencySourceFunctionType { Resonator StandardTerminalNameAssignment MandatoryMapping OptionalMapping } class ResonatorFrequencySourceStandardTerminalNameAssignmentType { StandardTerminalName } class ResonatorFrequencySourceMandatoryStandardTerminalMappingType { StandardTerminalName } class ResonatorFrequencySourceOptionalStandardTerminalMappingType { StandardTerminalName } ResonatorFrequencySourceFunctionType "1..n" --> StandardTerminalNameAssignment : StandardTerminalNameAssignment "1..n" --> MandatoryMapping : StandardTerminalNameAssignment "1..n" --> OptionalMapping : MandatoryMapping "2" --> StandardTerminalName : OptionalMapping "2" --> StandardTerminalName : </pre> <p>The diagram illustrates the structure of the ResonatorFrequencySourceFunctionType. It consists of several interconnected classes and their associations:</p> <ul style="list-style-type: none"> ResonatorFrequencySourceFunctionType: This is the main container class. StandardTerminalNameAssignment: An association class that connects the main type to two specific mapping types. MandatoryMapping: A class that maps to a single standard terminal name. OptionalMapping: A class that maps to a single standard terminal name. StandardTerminalName: The target class for both mandatory and optional mappings. constraints: A note indicating the presence of constraints. 																
type	ResonatorFrequencySourceFunctionType , ResonatorFrequencySourceStandardTerminalNameAssignmentType , ResonatorFrequencySourceMandatoryStandardTerminalMappingType , ResonatorFrequencySourceMandatoryStandardTerminalNameType , ResonatorFrequencySourceOptionalStandardTerminalMappingType , ResonatorFrequencySourceOptionalStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Terminal 1</td> <td>2. Terminal 2</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Case</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Terminal 1	2. Terminal 2			OptionalMapping/StandardTerminalName				1. Case			
MandatoryMapping/StandardTerminalName																	
1. Terminal 1	2. Terminal 2																
OptionalMapping/StandardTerminalName																	
1. Case																	

4.7.3.7.6. Voltage Controlled Oscillator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/VoltageControlledOscillator																																
diagram																																	
type	VoltageControlledOscillatorFrequencySourceFunctionType , VoltageControlledOscillatorFrequencySourceStandardTerminalNameAssignmentType , VoltageControlledOscillatorFrequencySourceMandatoryStandardTerminalMappingType , VoltageControlledOscillatorFrequencySourceMandatoryStandardTerminalNameType , VoltageControlledOscillatorFrequencySourceSingleEndOutputStandardTerminalMappingType , VoltageControlledOscillatorFrequencySourceSingleEndOutputStandardTerminalNameType , VoltageControlledOscillatorFrequencySourceDifferentialOutputStandardTerminalMappingType , VoltageControlledOscillatorFrequencySourceDifferentialOutputStandardTerminalNameType , VoltageControlledOscillatorFrequencySourceOptionalStandardTerminalMappingType , VoltageControlledOscillatorFrequencySourceOptionalStandardTerminalNameType , VoltageControlledOscillatorFrequencySourceOtherStandardTerminalMappingType , VoltageControlledOscillatorFrequencySourceOtherStandardTerminalNameType .																																
list of enum values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Output</td> <td>2. Ground</td> <td>3. Power</td> <td></td> </tr> <tr> <td colspan="4">SingleEndedOutputStandardTerminalMapping/ StandardTerminalName</td> </tr> <tr> <td>1. Output</td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="4">DifferentialOutputStandardTerminalMapping/ StandardTerminalName</td> </tr> <tr> <td>1. Positive Output</td> <td>2. Negative Output</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Enable</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Output	2. Ground	3. Power		SingleEndedOutputStandardTerminalMapping/ StandardTerminalName				1. Output				DifferentialOutputStandardTerminalMapping/ StandardTerminalName				1. Positive Output	2. Negative Output			OptionalMapping/StandardTerminalName				1. Enable			
MandatoryMapping/StandardTerminalName																																	
1. Output	2. Ground	3. Power																															
SingleEndedOutputStandardTerminalMapping/ StandardTerminalName																																	
1. Output																																	
DifferentialOutputStandardTerminalMapping/ StandardTerminalName																																	
1. Positive Output	2. Negative Output																																
OptionalMapping/StandardTerminalName																																	
1. Enable																																	

4.7.3.8. Fuse

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Fuse
diagram	<pre> classDiagram class Fuse { <<type: FuseFunctionType>> } class StandardTerminalNameAssignment { <<type: FuseStandardTerminalNameAssignmentType>> } class Mapping { <<type: FuseStandardTerminalMappingType>> } class StandardTerminalName { <<type: FuseStandardTerminalNameType>> } Fuse "1..∞" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "2" --> Mapping Mapping --> StandardTerminalName </pre>
type	FuseType , FuseStandardTerminalNameAssignmentType , FuseStandardTerminalMappingType , FuseStandardTerminalNameType .
list of enumerate values	Mapping/StandardTerminalName 1. Terminal 1 2. Terminal 2

4.7.3.9. Inductor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor
diagram	<pre> classDiagram class Inductor { <<type: InductorFunctionType>> } class Air { <<type: AirInductorFunctionType>> } class Coupled { <<type: CoupledInductorFunctionType>> } class Ferrite { <<type: FerriteInductorFunctionType>> } class Variable { <<type: VariableInductorFunctionType>> } class StandardTerminalNameAssignment { <<type: BasicInductorStandardTerminalNameAssignmentType>> } Inductor "1..∞" --> Air Inductor "1..∞" --> Coupled Inductor "1..∞" --> Ferrite Inductor "1..∞" --> Variable Air --> AirInductorFunctionType Coupled --> CoupledInductorFunctionType Ferrite --> FerriteInductorFunctionType Variable --> VariableInductorFunctionType AirInductorFunctionType "1..∞" --> StandardTerminalNameAssignment CoupledInductorFunctionType "1..∞" --> StandardTerminalNameAssignment FerriteInductorFunctionType "1..∞" --> StandardTerminalNameAssignment VariableInductorFunctionType "1..∞" --> StandardTerminalNameAssignment </pre>
type	InductorFunctionType , AirInductorFunctionType , BasicInductorStandardTerminalNameAssignmentType , CoupledInductorFunctionType , CoupledInductorStandardTerminalNameAssignmentType , FerriteInductorFunctionType , VariableInductorFunctionType .

An inductor can be one of the following four types: [Air](#), [Coupled](#), [Ferrite](#), and [Variable](#). Each of these types is explained below in further detail.

4.7.3.9.1. Basic Inductor Standard Terminal Name Assignment

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Air 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Ferrite 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Variable 				
diagram	<pre> classDiagram class StandardTerminalNameAssignment { type BasicInductorStandardTerminalN... } class Mapping { type BasicInductorStandardTerminal... } class StandardTerminalName { type BasicInductorStandardTerminalN... } StandardTerminalNameAssignment "1..∞" -- "2" Mapping Mapping -- "1" StandardTerminalName class BasicInductorStandardTerminalNameAssignmentType { StandardTerminalNameAssignment Mapping StandardTerminalName } +constraints </pre>				
type	BasicInductorStandardTerminalNameAssignmentType, BasicInductorStandardTerminalMappingType, BasicInductorStandardTerminalNameType.				
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Terminal 1</td> <td style="padding: 2px;">2. Terminal 2</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>	1. Terminal 1	2. Terminal 2		
1. Terminal 1	2. Terminal 2				

4.7.3.9.2. Coupled Inductor Standard Terminal Name Assignment

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Coupled				
diagram	<pre> classDiagram class StandardTerminalNameAssignment { type CoupledInductorStandardTermin... } class Mapping { type CoupledInductorStandardTermin... } class StandardTerminalName { type CoupledInductorStandardTermin... } StandardTerminalNameAssignment "1..∞" -- "4" Mapping Mapping -- "1" StandardTerminalName class CoupledInductorStandardTerminalNameAssignmentType { StandardTerminalNameAssignment Mapping StandardTerminalName } +constraints </pre>				
type	InductorFunctionType, AirInductorFunctionType, BasicInductorStandardTerminalNameAssignmentType, CoupledInductorFunctionType, CoupledInductorStandardTerminalNameAssignmentType, FerriteInductorFunctionType, VariableInductorFunctionType.				
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Primary Coil Terminal 1</td> <td style="padding: 2px;">2. Primary Coil Terminal 2</td> <td style="padding: 2px;">3. Secondary Coil Terminal 1</td> <td style="padding: 2px;">4. Secondary Coil Terminal 2</td> </tr> </table>	1. Primary Coil Terminal 1	2. Primary Coil Terminal 2	3. Secondary Coil Terminal 1	4. Secondary Coil Terminal 2
1. Primary Coil Terminal 1	2. Primary Coil Terminal 2	3. Secondary Coil Terminal 1	4. Secondary Coil Terminal 2		

4.7.3.10. Interface

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface	
Sample diagram (See list of defined Interfaces below)	<pre> classDiagram class Interface { <<type InterfaceFunctionType>> } class PCIe { <<type PCIe-InterfaceFunctionType>> } class CablingPCIe { <<type CablingPCIe-InterfaceFunctionType>> } class APHY { <<type A-PHY-InterfaceFunctionType>> } class CPHY { <<type C-PHY-InterfaceFunctionType>> } class DPHY { <<type D-PHY-InterfaceFunctionType>> } Interface < -- PCIe Interface < -- CablingPCIe Interface < -- APHY Interface < -- CPHY Interface < -- DPHY </pre>	
Standard interface versus type	InterfaceFunctionType, 1. AIB AIB-InterfaceFunctionType 2. Battery BatteryInterfaceFunctionType 3. Camera CameraInterfaceFunctionType 4. ComputerExpressLink ComputerExpressLink-InterfaceFunctionType 5. DisplayBus DisplayBusInterfaceFunctionType 6. DDR3 DDR3-InterfaceFunctionType 7. DDR4 DDR4-InterfaceFunctionType 8. DDR5 DDR5-InterfaceFunctionType 9. DDR6 DDR6-InterfaceFunctionType 10. DigRF3G DigRF3G-InterfaceFunctionType 11. DigRFv4 DigRFv4-InterfaceFunctionType 12. EE1002-SPD-EEPROM EE1002-SPD-EEPROM-InterfaceFunctionType 13. EmbeddedDisplayPort EmbeddedDisplayPort-InterfaceFunctionType 14. Ethernet EthernetInterfaceFunctionType 15. eTrak eTrakInterfaceFunctionType 16. FC-PI-6 FC-PI-6-InterfaceFunctionType 17. HBM HBM-InterfaceFunctionType 18. HDMI HDMI-InterfaceFunctionType 19. HSI HSI-InterfaceFunctionType 20. HTI HTI-InterfaceFunctionType 21. HTIv1 HTIv1-InterfaceFunctionType 22. I2C I2C-InterfaceFunctionType 23. I3C I3C-InterfaceFunctionType 24. LLI-Serial LLI-Serial-InterfaceFunctionType 25. LVSTL06 LVSTL06-InterfaceFunctionType 26. MultiMediaCard MultiMediaCard-InterfaceFunctionType 27. MII MII-InterfaceFunctionType 28. OIF-CEI-04.0 OIF-CEI-04.0-InterfaceFunctionType 29. PCIe PCIe-InterfaceFunctionType	
	1. AIB	AIB-InterfaceFunctionType
	2. Battery	BatteryInterfaceFunctionType
	3. Camera	CameraInterfaceFunctionType
	4. ComputerExpressLink	ComputerExpressLink-InterfaceFunctionType
	5. DisplayBus	DisplayBusInterfaceFunctionType
	6. DDR3	DDR3-InterfaceFunctionType
	7. DDR4	DDR4-InterfaceFunctionType
	8. DDR5	DDR5-InterfaceFunctionType
	9. DDR6	DDR6-InterfaceFunctionType
	10. DigRF3G	DigRF3G-InterfaceFunctionType
	11. DigRFv4	DigRFv4-InterfaceFunctionType
	12. EE1002-SPD-EEPROM	EE1002-SPD-EEPROM-InterfaceFunctionType
	13. EmbeddedDisplayPort	EmbeddedDisplayPort-InterfaceFunctionType
	14. Ethernet	EthernetInterfaceFunctionType
	15. eTrak	eTrakInterfaceFunctionType
	16. FC-PI-6	FC-PI-6-InterfaceFunctionType
	17. HBM	HBM-InterfaceFunctionType
	18. HDMI	HDMI-InterfaceFunctionType
	19. HSI	HSI-InterfaceFunctionType
	20. HTI	HTI-InterfaceFunctionType
	21. HTIv1	HTIv1-InterfaceFunctionType
	22. I2C	I2C-InterfaceFunctionType
	23. I3C	I3C-InterfaceFunctionType
	24. LLI-Serial	LLI-Serial-InterfaceFunctionType
	25. LVSTL06	LVSTL06-InterfaceFunctionType
	26. MultiMediaCard	MultiMediaCard-InterfaceFunctionType
	27. MII	MII-InterfaceFunctionType
	28. OIF-CEI-04.0	OIF-CEI-04.0-InterfaceFunctionType
	29. PCIe	PCIe-InterfaceFunctionType

4.7.3.10 Interface Function (cont'd)

Standard interface versus type	30. CablingPCIe	CablingPCIe-InterfaceFunctionType
	31. A-PHY	A-PHY-InterfaceFunctionType
	32. BoW-PHY	BoW-PHY-InterfaceFunctionType
	33. C-PHY	C-PHY-InterfaceFunctionType
	34. D-PHY	D-PHY-InterfaceFunctionType
	35. M-PHY	M-PHY-InterfaceFunctionType
	36. OpenHBI	OpenHBI-InterfaceFunctionType
	37. PTI	PTI-InterfaceFunctionType
	38. RadioFrontEnd	RadioFrontEnd-interfaceFunctionType
	39. RFFE	RFFE-InterfaceFunctionType
	40. SD	SD-InterfaceFunctionType
	41. SD-UHS-II	SD-UHS-II -InterfaceFunctionType
	42. SerialInterface	SerialInterfaceFunctionType
	43. SLIMbus	SLIMbus-InterfaceFunctionType
	44. SMB	SMB-InterfaceFunctionType
	45. SoundWire	SoundWire-InterfaceFunctionType
	46. SPMI	SPMI-InterfaceFunctionType
	47. UART	UART-InterfaceFunctionType
	48. UniPro	UniPro-InterfaceFunctionType
	49. UCle	UCle-InterfaceFunctionType
	50. UniversalFlashStorage	UniversalFlashStorage-InterfaceFunctionType
	51. USB	USB-InterfaceFunctionType
	52. XFM	XFM-InterfaceFunctionType
	53. OtherInterfaceStandard	FunctionMap-to-StandardNameType

4.7.3.10.1. AIB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/AIB
diagram	<pre> classDiagram class AIB { type AIB-InterfaceFunctionType } class AIB-Base { type AIB-BaseType } class AIB-Plus { type AIB-PlusType } AIB < -- AIB-Base AIB < -- AIB-Plus </pre>
type	AIB-InterfaceFunctionType, AIB-BaseType, AIB-PlusType.

For more information about the AIB Interface, refer to the CHIPS ALLIANCE standard Advanced Interface Bus (AIB) Specification.

4.7.3.10.1.1. AIB-Base

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/AIB/AIB-Base								
diagram	<pre> classDiagram class AIB-BaseType { <<AIB-Base<< } class StandardTerminalNameAssignment { <<AIB-BaseStandardTerminalNameAssignmentType<< } class Mapping { <<AIB-BaseStandardTerminalMappingType<< } class StandardTerminalName { <<AIB-BaseStandardTerminalNameType<< } AIB-BaseType "1..∞" -- "8" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" -- "8" Mapping Mapping "8" -- "8" StandardTerminalName </pre>								
type	AIB-BaseType, BaseStandardTerminalNameAssignmentType, AIB-BaseStandardTerminalMappingType, AIB-BaseStandardTerminalNameType.								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table> <tr> <td>1. TX</td> <td>2. RX</td> <td>3. ns_fwd_clk</td> <td>4. ns_fwd_clkb</td> </tr> <tr> <td>5. fs_fwd_clk</td> <td>6. fs_fwd_clkb</td> <td>7. ns_mac_rdy</td> <td>8. fs_mac_rdy</td> </tr> </table>	1. TX	2. RX	3. ns_fwd_clk	4. ns_fwd_clkb	5. fs_fwd_clk	6. fs_fwd_clkb	7. ns_mac_rdy	8. fs_mac_rdy
1. TX	2. RX	3. ns_fwd_clk	4. ns_fwd_clkb						
5. fs_fwd_clk	6. fs_fwd_clkb	7. ns_mac_rdy	8. fs_mac_rdy						

4.7.3.10.1.2. AIB-Plus

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/AIB/AIB-Base																								
diagram	<pre> classDiagram class AIB-PlusType { <<AIB-Plus<< } class StandardTerminalNameAssignment { <<AIB-PlusStandardTerminalNameAssignmentType<< } class IMapping { <<AIB-PlusStandardTerminalMappingType<< } class StandardTerminalName { <<AIB-PlusStandardTerminalNameType<< } AIB-PlusType "1..∞" -- "22" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" -- "22" IMapping IMapping "22" -- "22" StandardTerminalName </pre>																								
type	AIB-BaseType, BaseStandardTerminalNameAssignmentType, AIB-BaseStandardTerminalMappingType, AIB-BaseStandardTerminalNameType.																								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table> <tr> <td>1. TX</td> <td>2. RX</td> <td>3. ns_fwd_clk</td> <td>4. ns_fwd_clkb</td> </tr> <tr> <td>5. fs_fwd_clk</td> <td>6. fs_fwd_clkb</td> <td>7. ns_rcv_clk</td> <td>8. ns_rcv_clkb</td> </tr> <tr> <td>9. fs_rcv_clk</td> <td>10. fs_rcv_clkb</td> <td>11. ns_sr_clk</td> <td>12. ns_sr_clkb</td> </tr> <tr> <td>13. fs_sr_clk</td> <td>14. fs_sr_clkb</td> <td>15. ns_sr_data</td> <td>16. fs_sr_data</td> </tr> <tr> <td>17. ns_sr_load</td> <td>18. fs_sr_load</td> <td>19. ns_mac_rdy</td> <td>20. fs_mac_rdy</td> </tr> <tr> <td>21. ns_adapter_rstn</td> <td>22. fs_adapter_rstn</td> <td></td> <td></td> </tr> </table>	1. TX	2. RX	3. ns_fwd_clk	4. ns_fwd_clkb	5. fs_fwd_clk	6. fs_fwd_clkb	7. ns_rcv_clk	8. ns_rcv_clkb	9. fs_rcv_clk	10. fs_rcv_clkb	11. ns_sr_clk	12. ns_sr_clkb	13. fs_sr_clk	14. fs_sr_clkb	15. ns_sr_data	16. fs_sr_data	17. ns_sr_load	18. fs_sr_load	19. ns_mac_rdy	20. fs_mac_rdy	21. ns_adapter_rstn	22. fs_adapter_rstn		
1. TX	2. RX	3. ns_fwd_clk	4. ns_fwd_clkb																						
5. fs_fwd_clk	6. fs_fwd_clkb	7. ns_rcv_clk	8. ns_rcv_clkb																						
9. fs_rcv_clk	10. fs_rcv_clkb	11. ns_sr_clk	12. ns_sr_clkb																						
13. fs_sr_clk	14. fs_sr_clkb	15. ns_sr_data	16. fs_sr_data																						
17. ns_sr_load	18. fs_sr_load	19. ns_mac_rdy	20. fs_mac_rdy																						
21. ns_adapter_rstn	22. fs_adapter_rstn																								

4.7.3.10.2. Battery Interface Function

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/Battery</code>		
diagram	<pre> classDiagram class Battery { <<BatteryInterfaceFunctionType>> } class StandardTerminalNameAssignment { <<BatteryStandardTerminalNameAssignmentType>> } class Mapping { <<BatteryStandardTerminalMappingType>> } class StandardTerminalName { <<BatteryStandardTerminalNameType>> } Battery "1..∞" -- "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" -- "3" Mapping Mapping "3" -- "3" StandardTerminalName </pre>		
type	BatteryInterfaceFunctionType, BatteryStandardTerminalNameAssignmentType, BatteryStandardTerminalMappingType, BatteryStandardTerminalNameType.		
list of enumerate values	Mapping/StandardTerminalName 1. VBAT 2. BCL 3. GND		

For more information about the Battery Interface, refer to the MIPI Alliance standard Specification for Battery Interface Version 1.1.1.

4.7.3.10.3. Camera Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/Camera
diagram	<pre> graph TD Camera[Camera] --- CSI[CSI] Camera --- CPHYx1[CSI-2_C-PHY-x1] Camera --- CPHYx2[CSI-2_C-PHY-x2] Camera --- CPHYx3[CSI-2_C-PHY-x3] Camera --- CPHYx4[CSI-2_C-PHY-x4] Camera --- CPHYx5[CSI-2_C-PHY-x5] Camera --- CPHYx6[CSI-2_C-PHY-x6] Camera --- CPHYx7[CSI-2_D-PHY-x1] Camera --- CPHYx8[CSI-2_D-PHY-x2] Camera --- CPHYx9[CSI-2_D-PHY-x3] Camera --- CPHYx10[CSI-2_D-PHY-x4] Camera --- CPHYx11[CSI-2_D-PHY-x5] Camera --- CPHYx12[CSI-2_D-PHY-x6] Camera --- CPHYx13[CSI-2_D-PHY-x7] Camera --- CPHYx14[CSI-2_D-PHY-x8] Camera --- CSI3[CSI-3] Camera --- CPI[CPI] </pre>
type	CameraInterfaceFunctionType, CSI-Type, CSI-2_C-PHY-x1Type, CSI-2_C-PHY-x2Type, CSI-2_C-PHY-x3Type, CSI-2_C-PHY-x4Type, CSI-2_C-PHY-x5Type, CSI-2_C-PHY-x6Type, CSI-2_D-PHY-x1Type, CSI-2_D-PHY-x2Type, CSI-2_D-PHY-x3Type, CSI-2_D-PHY-x4Type, CSI-2_D-PHY-x5Type, CSI-2_D-PHY-x6Type, CSI-2_D-PHY-x7Type, CSI-2_D-PHY-x8Type, CSI-3-Type, CPI-Type.

4.7.3.10.3.1. CSI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI								
diagram	<pre> classDiagram class CSI { <<CSI>> } class StandardTerminalNameAssignment { <<CSI-StandardTerminalNameAssignmentType>> } class Mapping { <<CSI-StandardTerminalMappingType>> } class StandardTerminalName { <<CSI-StandardTerminalNameType>> } CSI --> StandardTerminalNameAssignment : StandardTerminalNameAssignment --> Mapping : Mapping --> StandardTerminalName : class CSI_Type { <<CSI-Type>> } class CSI_StandardTerminalNameAssignmentType { <<CSI-StandardTerminalNameAssignmentType>> } class CSI_StandardTerminalMappingType { <<CSI-StandardTerminalMappingType>> } class CSI_StandardTerminalNameType { <<CSI-StandardTerminalNameType>> } CSI_Type < -- CSI CSI_Type < -- StandardTerminalNameAssignment CSI_Type < -- Mapping CSI_Type < -- StandardTerminalName CSI_StandardTerminalNameAssignmentType < -- StandardTerminalNameAssignment CSI_StandardTerminalNameAssignmentType < -- Mapping </pre>								
type	CSI-Type, CSI-StandardTerminalNameAssignmentType, CSI-StandardTerminalMappingType, CSI-StandardTerminalNameType.								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Data+</td> <td>2. Data-</td> <td>3. Clk+</td> <td>4. Clk-</td> </tr> <tr> <td>5. SCL</td> <td>6. SDA</td> <td></td> <td></td> </tr> </table>	1. Data+	2. Data-	3. Clk+	4. Clk-	5. SCL	6. SDA		
1. Data+	2. Data-	3. Clk+	4. Clk-						
5. SCL	6. SDA								

For more information about the CSI Interface, refer to the MIPI Alliance standard Specification for Camera Serial Interface Version 1.0.

4.7.3.10.3.2. CSI-2 C-PHY-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x1								
diagram	<pre> classDiagram class CSI_2_C_PHY_x1 { <<CSI-2_C-PHY-x1Type>> } class StandardTerminalNameAssignment { <<CSI-2_C-PHY-x1-StandardTerminalNameAssignmentType>> } class Mapping { <<CSI-2_C-PHY-x1-StandardTerminalMappingType>> } class StandardTerminalName { <<CSI-2_C-PHY-x1-StandardTerminalNameType>> } CSI_2_C_PHY_x1 --> StandardTerminalNameAssignment : StandardTerminalNameAssignment --> Mapping : Mapping --> StandardTerminalName : class CSI_2_C_PHY_x1_Type { <<CSI-2_C-PHY-x1Type>> } class CSI_2_C_PHY_x1_StandardTerminalNameAssignmentType { <<CSI-2_C-PHY-x1-StandardTerminalNameAssignmentType>> } class CSI_2_C_PHY_x1_StandardTerminalMappingType { <<CSI-2_C-PHY-x1-StandardTerminalMappingType>> } class CSI_2_C_PHY_x1_StandardTerminalNameType { <<CSI-2_C-PHY-x1-StandardTerminalNameType>> } CSI_2_C_PHY_x1_Type < -- CSI_2_C_PHY_x1 CSI_2_C_PHY_x1_Type < -- StandardTerminalNameAssignment CSI_2_C_PHY_x1_Type < -- Mapping CSI_2_C_PHY_x1_Type < -- StandardTerminalName CSI_2_C_PHY_x1_StandardTerminalNameAssignmentType < -- StandardTerminalNameAssignment CSI_2_C_PHY_x1_StandardTerminalNameAssignmentType < -- Mapping </pre>								
type	CSI-2_C-PHY-x1Type, CSI-2_C-PHY-x1-StandardTerminalNameAssignmentType, CSI-2_C-PHY-x1-StandardTerminalMappingType, CSI-2_C-PHY-x1-StandardTerminalNameType.								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Data1_A</td> <td>2. Data1_B</td> <td>3. Data1_C</td> <td>4. SCL</td> </tr> <tr> <td>5. SDA</td> <td></td> <td></td> <td></td> </tr> </table>	1. Data1_A	2. Data1_B	3. Data1_C	4. SCL	5. SDA			
1. Data1_A	2. Data1_B	3. Data1_C	4. SCL						
5. SDA									

For more information about the CSI-2 Interface, refer to the MIPI Alliance standard Specification for Camera Serial Interface 2 (CSI-2) Version 4.0.

4.7.3.10.3.3. CSI-2 C-PHY-x2

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x2								
diagram	<pre> classDiagram class CSI_2_C_PHY_x2Type class StandardTerminalNameAssignment class Mapping class StandardTerminalName CSI_2_C_PHY_x2Type --> StandardTerminalNameAssignment : StandardTerminalNameAssignment --> Mapping : 1..0 Mapping --> StandardTerminalName : 0..1 </pre>								
type	CSI-2_C-PHY-x2Type , CSI-2_C-PHY-x2-StandardTerminalNameAssignmentType , CSI-2_C-PHY-x2-StandardTerminalMappingType , CSI-2_C-PHY-x2-StandardTerminalNameType .								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. Data1_A</td> <td>2. Data1_B</td> <td>3. Data1_C</td> <td>4. SCL</td> </tr> <tr> <td>5. SDA</td> <td>6. Data2_A</td> <td>7. Data2_B</td> <td>8. Data2_C</td> </tr> </table>	1. Data1_A	2. Data1_B	3. Data1_C	4. SCL	5. SDA	6. Data2_A	7. Data2_B	8. Data2_C
1. Data1_A	2. Data1_B	3. Data1_C	4. SCL						
5. SDA	6. Data2_A	7. Data2_B	8. Data2_C						

4.7.3.10.3.4. CSI-2 C-PHY-x3

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x3												
diagram	<pre> classDiagram class CSI_2_C_PHY_x3Type class StandardTerminalNameAssignment class Mapping class StandardTerminalName CSI_2_C_PHY_x3Type --> StandardTerminalNameAssignment : StandardTerminalNameAssignment --> Mapping : 1..0 Mapping --> StandardTerminalName : 0..1 </pre>												
type	CSI-2_C-PHY-x3Type , CSI-2_C-PHY-x3-StandardTerminalNameAssignmentType , CSI-2_C-PHY-x3-StandardTerminalMappingType , CSI-2_C-PHY-x3-StandardTerminalNameType .												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. Data1_A</td> <td>2. Data1_B</td> <td>3. Data1_C</td> <td>4. SCL</td> </tr> <tr> <td>5. SDA</td> <td>6. Data2_A</td> <td>7. Data2_B</td> <td>8. Data2_C</td> </tr> <tr> <td>9. Data3_A</td> <td>10. Data3_B</td> <td>11. Data3_C</td> <td></td> </tr> </table>	1. Data1_A	2. Data1_B	3. Data1_C	4. SCL	5. SDA	6. Data2_A	7. Data2_B	8. Data2_C	9. Data3_A	10. Data3_B	11. Data3_C	
1. Data1_A	2. Data1_B	3. Data1_C	4. SCL										
5. SDA	6. Data2_A	7. Data2_B	8. Data2_C										
9. Data3_A	10. Data3_B	11. Data3_C											

4.7.3.10.3.5. CSI-2 C-PHY-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x4																				
diagram																					
type	CSI-2_C-PHY-x4Type , CSI-2_C-PHY-x4_StandardTerminalNameAssignmentType , CSI-2_C-PHY-x4_StandardTerminalMappingType , CSI-2_C-PHY-x4_StandardTerminalNameType .																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Data1_A</td><td>2. Data1_B</td><td>3. Data1_C</td><td>4. SCL</td></tr> <tr> <td>5. SDA</td><td>6. Data2_A</td><td>7. Data2_B</td><td>8. Data2_C</td></tr> <tr> <td>9. Data3_A</td><td>10. Data3_B</td><td>11. Data3_C</td><td>12. Data4_A</td></tr> <tr> <td>13. Data4_B</td><td>14. Data4_C</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. Data1_A	2. Data1_B	3. Data1_C	4. SCL	5. SDA	6. Data2_A	7. Data2_B	8. Data2_C	9. Data3_A	10. Data3_B	11. Data3_C	12. Data4_A	13. Data4_B	14. Data4_C		
Mapping/StandardTerminalName																					
1. Data1_A	2. Data1_B	3. Data1_C	4. SCL																		
5. SDA	6. Data2_A	7. Data2_B	8. Data2_C																		
9. Data3_A	10. Data3_B	11. Data3_C	12. Data4_A																		
13. Data4_B	14. Data4_C																				

4.7.3.10.3.6. CSI-2 C-PHY-x5

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x5																								
diagram																									
type	CSI-2_C-PHY-x5Type , CSI-2_C-PHY-x5_StandardTerminalNameAssignmentType , CSI-2_C-PHY-x5_StandardTerminalMappingType , CSI-2_C-PHY-x5_StandardTerminalNameType .																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Data1_A</td><td>2. Data1_B</td><td>3. Data1_C</td><td>4. SCL</td></tr> <tr> <td>5. SDA</td><td>6. Data2_A</td><td>7. Data2_B</td><td>8. Data2_C</td></tr> <tr> <td>9. Data3_A</td><td>10. Data3_B</td><td>11. Data3_C</td><td>12. Data4_A</td></tr> <tr> <td>13. Data4_B</td><td>14. Data4_C</td><td>15. Data5_A</td><td>16. Data5_B</td></tr> <tr> <td>17. Data5_C</td><td></td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. Data1_A	2. Data1_B	3. Data1_C	4. SCL	5. SDA	6. Data2_A	7. Data2_B	8. Data2_C	9. Data3_A	10. Data3_B	11. Data3_C	12. Data4_A	13. Data4_B	14. Data4_C	15. Data5_A	16. Data5_B	17. Data5_C			
Mapping/StandardTerminalName																									
1. Data1_A	2. Data1_B	3. Data1_C	4. SCL																						
5. SDA	6. Data2_A	7. Data2_B	8. Data2_C																						
9. Data3_A	10. Data3_B	11. Data3_C	12. Data4_A																						
13. Data4_B	14. Data4_C	15. Data5_A	16. Data5_B																						
17. Data5_C																									

4.7.3.10.3.7. CSI-2 C-PHY-x6

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x6																								
diagram																									
type	CSI-2_C-PHY-x6Type , CSI-2_C-PHY-x6-StandardTerminalNameAssignmentType , CSI-2_C-PHY-x6-StandardTerminalMappingType , CSI-2_C-PHY-x6-StandardTerminalNameType .																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Data1_A</td><td>2. Data1_B</td><td>3. Data1_C</td><td>4. SCL</td></tr> <tr> <td>5. SDA</td><td>6. Data2_A</td><td>7. Data2_B</td><td>8. Data2_C</td></tr> <tr> <td>9. Data3_A</td><td>10. Data3_B</td><td>11. Data3_C</td><td>12. Data4_A</td></tr> <tr> <td>13. Data4_B</td><td>14. Data4_C</td><td>15. Data5_A</td><td>16. Data5_B</td></tr> <tr> <td>17. Data5_C</td><td>18. Data6_A</td><td>19. Data6_B</td><td>20. Data6_C</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. Data1_A	2. Data1_B	3. Data1_C	4. SCL	5. SDA	6. Data2_A	7. Data2_B	8. Data2_C	9. Data3_A	10. Data3_B	11. Data3_C	12. Data4_A	13. Data4_B	14. Data4_C	15. Data5_A	16. Data5_B	17. Data5_C	18. Data6_A	19. Data6_B	20. Data6_C
Mapping/StandardTerminalName																									
1. Data1_A	2. Data1_B	3. Data1_C	4. SCL																						
5. SDA	6. Data2_A	7. Data2_B	8. Data2_C																						
9. Data3_A	10. Data3_B	11. Data3_C	12. Data4_A																						
13. Data4_B	14. Data4_C	15. Data5_A	16. Data5_B																						
17. Data5_C	18. Data6_A	19. Data6_B	20. Data6_C																						

4.7.3.10.3.8. CSI-2 D-PHY-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x1												
diagram													
type	CSI-2_D-PHY-x1Type , CSI-2_D-PHY-x1-StandardTerminalNameAssignmentType , CSI-2_D-PHY-x1-StandardTerminalMappingType , CSI-2_D-PHY-x1-StandardTerminalNameType .												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Data1+</td><td>2. Data1-</td><td>3. Clock+</td><td>4. Clock-</td></tr> <tr> <td>5. SCL</td><td>6. SDA</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA		
Mapping/StandardTerminalName													
1. Data1+	2. Data1-	3. Clock+	4. Clock-										
5. SCL	6. SDA												

4.7.3.10.3.9. CSI-2 D-PHY-x2

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x2												
diagram	<pre> classDiagram class CSI-2_D-PHY-x2Type { <<CSI-2_D-PHY-x2Type>> } class StandardTerminalNameAssignment { <<CSI-2_D-PHY-x2_StandardTerminalNameAssignmentType>> } class Mapping { <<CSI-2_D-PHY-x2_StandardTerminalMappingType>> } class StandardTerminalName { <<CSI-2_D-PHY-x2_StandardTerminalNameType>> } CSI-2_D-PHY-x2Type "1..n" -- "1..n" StandardTerminalNameAssignment CSI-2_D-PHY-x2Type "1..n" -- "1..n" Mapping StandardTerminalNameAssignment "1..n" -- "1..n" StandardTerminalName </pre>												
type	CSI-2_D-PHY-x2Type, CSI-2_D-PHY-x2_StandardTerminalNameAssignmentType, CSI-2_D-PHY-x2_StandardTerminalMappingType, CSI-2_D-PHY-x2_StandardTerminalNameType.												
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Data1+</td> <td>2. Data1-</td> <td>3. Clock+</td> <td>4. Clock-</td> </tr> <tr> <td>5. SCL</td> <td>6. SDA</td> <td>7. Data2+</td> <td>8. Data2-</td> </tr> <tr> <td>9. Data3+</td> <td>10. Data3-</td> <td></td> <td></td> </tr> </table>	1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA	7. Data2+	8. Data2-	9. Data3+	10. Data3-		
1. Data1+	2. Data1-	3. Clock+	4. Clock-										
5. SCL	6. SDA	7. Data2+	8. Data2-										
9. Data3+	10. Data3-												

4.7.3.10.3.10. CSI-2 D-PHY-x3

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x3												
diagram	<pre> classDiagram class CSI-2_D-PHY-x3Type { <<CSI-2_D-PHY-x3Type>> } class StandardTerminalNameAssignment { <<CSI-2_D-PHY-x3_StandardTerminalNameAssignmentType>> } class Mapping { <<CSI-2_D-PHY-x3_StandardTerminalMappingType>> } class StandardTerminalName { <<CSI-2_D-PHY-x3_StandardTerminalNameType>> } CSI-2_D-PHY-x3Type "1..n" -- "1..n" StandardTerminalNameAssignment CSI-2_D-PHY-x3Type "1..n" -- "1..n" Mapping StandardTerminalNameAssignment "1..n" -- "1..n" StandardTerminalName </pre>												
type	CSI-2_D-PHY-x3Type, CSI-2_D-PHY-x3_StandardTerminalNameAssignmentType, CSI-2_D-PHY-x3_StandardTerminalMappingType, CSI-2_D-PHY-x3_StandardTerminalNameType.												
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Data1+</td> <td>2. Data1-</td> <td>3. Clock+</td> <td>4. Clock-</td> </tr> <tr> <td>5. SCL</td> <td>6. SDA</td> <td>7. Data2+</td> <td>8. Data2-</td> </tr> <tr> <td>9. Data3+</td> <td>10. Data3-</td> <td></td> <td></td> </tr> </table>	1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA	7. Data2+	8. Data2-	9. Data3+	10. Data3-		
1. Data1+	2. Data1-	3. Clock+	4. Clock-										
5. SCL	6. SDA	7. Data2+	8. Data2-										
9. Data3+	10. Data3-												

4.7.3.10.3.11. CSI-2 D-PHY-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x4												
diagram	<pre> classDiagram class CSI_2_D_PHY_x4Type { CSI_2_D_PHY_x4 } class StandardTerminalNameAssignment { type CSI_2_D_PHY-x4-StandardTerminalNameAssignmentType } class Mapping { type CSI_2_D_PHY-x4-StandardTerminalMappingType } class StandardTerminalName { type CSI_2_D_PHY-x4-StandardTerminalNameType } CSI_2_D_PHY_x4Type "1..x" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..x" Mapping Mapping "1..x" StandardTerminalName </pre>												
type	CSI-2_D-PHY-x4Type, CSI-2_D-PHY-x4-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x4-StandardTerminalMappingType, CSI-2_D-PHY-x4-StandardTerminalNameType.												
list of enumerative values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Data1+</td> <td>2. Data1-</td> <td>3. Clock+</td> <td>4. Clock-</td> </tr> <tr> <td>5. SCL</td> <td>6. SDA</td> <td>7. Data2+</td> <td>8. Data2-</td> </tr> <tr> <td>9. Data3+</td> <td>10. Data3-</td> <td>11. Data4+</td> <td>12. Data4-</td> </tr> </table>	1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA	7. Data2+	8. Data2-	9. Data3+	10. Data3-	11. Data4+	12. Data4-
1. Data1+	2. Data1-	3. Clock+	4. Clock-										
5. SCL	6. SDA	7. Data2+	8. Data2-										
9. Data3+	10. Data3-	11. Data4+	12. Data4-										

4.7.3.10.3.12. CSI-2 D-PHY-x5

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x5																
diagram	<pre> classDiagram class CSI_2_D_PHY_x5Type { CSI_2_D_PHY_x5 } class StandardTerminalNameAssignment { type CSI_2_D_PHY-x5-StandardTerminalNameAssignmentType } class Mapping { type CSI_2_D_PHY-x5-StandardTerminalMappingType } class StandardTerminalName { type CSI_2_D_PHY-x5-StandardTerminalNameType } CSI_2_D_PHY_x5Type "1..x" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..x" Mapping Mapping "1..x" StandardTerminalName </pre>																
type	CSI-2_D-PHY-x5Type, CSI-2_D-PHY-x5-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x5-StandardTerminalMappingType, CSI-2_D-PHY-x5-StandardTerminalNameType.																
list of enumerative values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Data1+</td> <td>2. Data1-</td> <td>3. Clock+</td> <td>4. Clock-</td> </tr> <tr> <td>5. SCL</td> <td>6. SDA</td> <td>7. Data2+</td> <td>8. Data2-</td> </tr> <tr> <td>9. Data3+</td> <td>10. Data3-</td> <td>11. Data4+</td> <td>12. Data4-</td> </tr> <tr> <td>13. Data5+</td> <td>14. Data5-</td> <td></td> <td></td> </tr> </table>	1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA	7. Data2+	8. Data2-	9. Data3+	10. Data3-	11. Data4+	12. Data4-	13. Data5+	14. Data5-		
1. Data1+	2. Data1-	3. Clock+	4. Clock-														
5. SCL	6. SDA	7. Data2+	8. Data2-														
9. Data3+	10. Data3-	11. Data4+	12. Data4-														
13. Data5+	14. Data5-																

4.7.3.10.3.13. CSI-2 D-PHY-x6

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x6																				
diagram																					
type	CSI-2_D-PHY-x6Type , CSI-2_D-PHY-x6-StandardTerminalNameAssignmentType , CSI-2_D-PHY-x6-StandardTerminalMappingType , CSI-2_D-PHY-x6-StandardTerminalNameType .																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Data1+</td><td>2. Data1-</td><td>3. Clock+</td><td>4. Clock-</td></tr> <tr> <td>5. SCL</td><td>6. SDA</td><td>7. Data2+</td><td>8. Data2-</td></tr> <tr> <td>9. Data3+</td><td>10. Data3-</td><td>11. Data4+</td><td>12. Data4-</td></tr> <tr> <td>13. Data5+</td><td>14. Data5-</td><td>15. Data6+</td><td>16. Data6-</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA	7. Data2+	8. Data2-	9. Data3+	10. Data3-	11. Data4+	12. Data4-	13. Data5+	14. Data5-	15. Data6+	16. Data6-
Mapping/StandardTerminalName																					
1. Data1+	2. Data1-	3. Clock+	4. Clock-																		
5. SCL	6. SDA	7. Data2+	8. Data2-																		
9. Data3+	10. Data3-	11. Data4+	12. Data4-																		
13. Data5+	14. Data5-	15. Data6+	16. Data6-																		

4.7.3.10.3.14. CSI-2 D-PHY-x7

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x7																								
diagram																									
type	CSI-2_D-PHY-x7Type , CSI-2_D-PHY-x7-StandardTerminalNameAssignmentType , CSI-2_D-PHY-x7-StandardTerminalMappingType , CSI-2_D-PHY-x7-StandardTerminalNameType .																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Data1+</td><td>2. Data1-</td><td>3. Clock+</td><td>4. Clock-</td></tr> <tr> <td>5. SCL</td><td>6. SDA</td><td>7. Data2+</td><td>8. Data2-</td></tr> <tr> <td>9. Data3+</td><td>10. Data3-</td><td>11. Data4+</td><td>12. Data4-</td></tr> <tr> <td>13. Data5+</td><td>14. Data5-</td><td>15. Data6+</td><td>16. Data6-</td></tr> <tr> <td>17. Data7+</td><td>18. Data7-</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA	7. Data2+	8. Data2-	9. Data3+	10. Data3-	11. Data4+	12. Data4-	13. Data5+	14. Data5-	15. Data6+	16. Data6-	17. Data7+	18. Data7-		
Mapping/StandardTerminalName																									
1. Data1+	2. Data1-	3. Clock+	4. Clock-																						
5. SCL	6. SDA	7. Data2+	8. Data2-																						
9. Data3+	10. Data3-	11. Data4+	12. Data4-																						
13. Data5+	14. Data5-	15. Data6+	16. Data6-																						
17. Data7+	18. Data7-																								

4.7.3.10.3.15. CSI-2 D-PHY-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x8																								
diagram	<pre> classDiagram class CSI_2_D_PHY_x8Type class StandardTerminalNameAssignment { <<CSI-2_D-PHY-x8-StandardTerminalNameAssignmentType>> } class Mapping { <<CSI-2_D-PHY-x8-StandardTerminalMappingType>> } class StandardTerminalName { <<CSI-2_D-PHY-x8-StandardTerminalNameType>> } CSI_2_D_PHY_x8Type --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>																								
type	CSI-2_D-PHY-x8Type , CSI-2_D-PHY-x8-StandardTerminalNameAssignmentType , CSI-2_D-PHY-x8-StandardTerminalMappingType , CSI-2_D-PHY-x8-StandardTerminalNameType .																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Data1+</td><td>2. Data1-</td><td>3. Clock+</td><td>4. Clock-</td></tr> <tr> <td>5. SCL</td><td>6. SDA</td><td>7. Data2+</td><td>8. Data2-</td></tr> <tr> <td>9. Data3+</td><td>10. Data3-</td><td>11. Data4+</td><td>12. Data4-</td></tr> <tr> <td>13. Data5+</td><td>14. Data5-</td><td>15. Data6+</td><td>16. Data6-</td></tr> <tr> <td>17. Data7+</td><td>18. Data7-</td><td>19. Data8+</td><td>20. Data8-</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. Data1+	2. Data1-	3. Clock+	4. Clock-	5. SCL	6. SDA	7. Data2+	8. Data2-	9. Data3+	10. Data3-	11. Data4+	12. Data4-	13. Data5+	14. Data5-	15. Data6+	16. Data6-	17. Data7+	18. Data7-	19. Data8+	20. Data8-
Mapping/StandardTerminalName																									
1. Data1+	2. Data1-	3. Clock+	4. Clock-																						
5. SCL	6. SDA	7. Data2+	8. Data2-																						
9. Data3+	10. Data3-	11. Data4+	12. Data4-																						
13. Data5+	14. Data5-	15. Data6+	16. Data6-																						
17. Data7+	18. Data7-	19. Data8+	20. Data8-																						

4.7.3.10.3.16. CSI-3

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-3								
diagram	<pre> classDiagram class CSI_3_Type class StandardTerminalNameAssignment { <<CSI-3-StandardTerminalNameAssignmentType>> } class Mapping { <<CSI-3-StandardTerminalMappingType>> } class StandardTerminalName { <<CSI-3-StandardTerminalNameType>> } CSI_3_Type --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>								
type	CSI-3-Type , CSI-3-StandardTerminalNameAssignmentType , CSI-3-StandardTerminalMappingType , CSI-3-StandardTerminalNameType								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. M-TX</td><td>2. M-RX</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. M-TX	2. M-RX		
Mapping/StandardTerminalName									
1. M-TX	2. M-RX								

For more information about the CSI-3 Interface, refer to the MIPI Alliance standard Specification for Camera Serial Interface 3 (CSI-3) Version 1.1.

4.7.3.10.3.1. CPI

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CPI</code>												
diagram	<pre> classDiagram class CPI-Type class StandardTerminalNameAssignment class Mapping class StandardTerminalName CPI-Type "1..∞" -- "1..1" StandardTerminalNameAssignment StandardTerminalNameAssignment -- "1..1" StandardTerminalName </pre>												
type	CPI-Type, CPI-StandardTerminalNameAssignmentType, CPI-StandardTerminalMappingType, CPI-StandardTerminalNameType												
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. I2C_SCL</td> <td>2. I2C_SDA</td> <td>3. CPI_CLK</td> <td>4. CPI_DAT[0]</td> </tr> <tr> <td>5. CPI_DAT[1]</td> <td>6. CPI_DAT[2]</td> <td>7. CPI_DAT[3]</td> <td>8. CPI_DAT[4]</td> </tr> <tr> <td>9. CPI_DAT[5]</td> <td>10. CPI_DAT[6]</td> <td>11. CPI_DAT[7]</td> <td></td> </tr> </table>	1. I2C_SCL	2. I2C_SDA	3. CPI_CLK	4. CPI_DAT[0]	5. CPI_DAT[1]	6. CPI_DAT[2]	7. CPI_DAT[3]	8. CPI_DAT[4]	9. CPI_DAT[5]	10. CPI_DAT[6]	11. CPI_DAT[7]	
1. I2C_SCL	2. I2C_SDA	3. CPI_CLK	4. CPI_DAT[0]										
5. CPI_DAT[1]	6. CPI_DAT[2]	7. CPI_DAT[3]	8. CPI_DAT[4]										
9. CPI_DAT[5]	10. CPI_DAT[6]	11. CPI_DAT[7]											

For more information about the CPI Interface, refer to the MIPI Alliance standard Specification for Camera Parallel Interface (CPI) Version 1.0

4.7.3.10.4. Compute Express Link Function

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ComputeExpressLink</code>
diagram	<pre> classDiagram class ComputeExpressLink class ComputeExpressLink-InterfaceFunctionType class CXL-x4 class CXL-x8 class CXL-x16 ComputeExpressLink --> ComputeExpressLink-InterfaceFunctionType ComputeExpressLink-InterfaceFunctionType "1..1" --> CXL-x4 ComputeExpressLink-InterfaceFunctionType "1..1" --> CXL-x8 ComputeExpressLink-InterfaceFunctionType "1..1" --> CXL-x16 </pre>
type	ComputeExpressLink-InterfaceFunctionType, CXL-x4-InterfaceFunctionType, CXL-x8-InterfaceFunctionType, CXL-x4-InterfaceFunctionType.

For more information about the Compute Express Link Interface, refer to the JEDEC standard JESD317.

4.7.3.10.4.1. CXL-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/ComputeExpressLink/CXL-x4																																				
diagram	<pre> classDiagram class CXL_x4_InterfaceFunctionType class StandardTerminalNameAssignment class MandatoryMapping class OptionalMapping class StandardTerminalName CXL_x4_InterfaceFunctionType "1..*" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..31" --> MandatoryMapping StandardTerminalNameAssignment "0..4" --> OptionalMapping MandatoryMapping --> StandardTerminalName OptionalMapping --> StandardTerminalName </pre>																																				
type	CXL-x4-InterfaceFunctionType , CXL-x4-StandardTerminalNameAssignmentType , CXL-x4-MandatoryMappingType , CXL-x4-MandatoryStandardTerminalNameType , CXL-x4-OptionalMappingType , CXL-x4-OptionalStandardTerminalNameType																																				
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. PETp0</td><td>2. PETn0</td><td>3. PETp1</td><td>4. PETn1</td></tr> <tr><td>5. PETp2</td><td>6. PETn2</td><td>7. PETp3</td><td>8. PETn3</td></tr> <tr><td>9. PERp0</td><td>10. PERn0</td><td>11. PERp1</td><td>12. PERn1</td></tr> <tr><td>13. PERp2</td><td>14. PERn2</td><td>15. PERp3</td><td>16. PERn3</td></tr> <tr><td>17. REFCLKp0</td><td>18. REFCLKn0</td><td>19. PERST0#</td><td>20. PRSNT0#</td></tr> <tr><td>21. SMBCLK</td><td>22. SMBDATA</td><td>23. SMBRST#</td><td>24. DUALPORTEN#</td></tr> <tr><td>25. LED</td><td>26. PWRDIS</td><td>27. MFG</td><td>28. RFU</td></tr> <tr><td>29. 12V</td><td>30. 3.3Vaux</td><td>31. GND</td><td></td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. REFCLKp1</td><td>2. REFCLKn1</td><td>3. PERST1#</td><td>4. CLKREQ#</td></tr> </table>	1. PETp0	2. PETn0	3. PETp1	4. PETn1	5. PETp2	6. PETn2	7. PETp3	8. PETn3	9. PERp0	10. PERn0	11. PERp1	12. PERn1	13. PERp2	14. PERn2	15. PERp3	16. PERn3	17. REFCLKp0	18. REFCLKn0	19. PERST0#	20. PRSNT0#	21. SMBCLK	22. SMBDATA	23. SMBRST#	24. DUALPORTEN#	25. LED	26. PWRDIS	27. MFG	28. RFU	29. 12V	30. 3.3Vaux	31. GND		1. REFCLKp1	2. REFCLKn1	3. PERST1#	4. CLKREQ#
1. PETp0	2. PETn0	3. PETp1	4. PETn1																																		
5. PETp2	6. PETn2	7. PETp3	8. PETn3																																		
9. PERp0	10. PERn0	11. PERp1	12. PERn1																																		
13. PERp2	14. PERn2	15. PERp3	16. PERn3																																		
17. REFCLKp0	18. REFCLKn0	19. PERST0#	20. PRSNT0#																																		
21. SMBCLK	22. SMBDATA	23. SMBRST#	24. DUALPORTEN#																																		
25. LED	26. PWRDIS	27. MFG	28. RFU																																		
29. 12V	30. 3.3Vaux	31. GND																																			
1. REFCLKp1	2. REFCLKn1	3. PERST1#	4. CLKREQ#																																		

4.7.3.10.4.2. CXL-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/ComputeExpressLink/CXL-x8																																																				
diagram	<pre> classDiagram class CXL_x8_InterfaceFunctionType class StandardTerminalNameAssignment class MandatoryMapping class OptionalMapping class StandardTerminalName CXL_x8_InterfaceFunctionType < -- StandardTerminalNameAssignment StandardTerminalNameAssignment < -- MandatoryMapping StandardTerminalNameAssignment < -- OptionalMapping MandatoryMapping < -- StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for the CXL-x8 Interface Function Type. It features a main class, CXL-x8-InterfaceFunctionType, which contains a relationship to StandardTerminalNameAssignment. This assignment is further refined into two types: MandatoryMapping and OptionalMapping. Each of these mapping types has a relationship to StandardTerminalName. The multiplicity for the StandardTerminalNameAssignment relationship is 1..oo, while for the MandatoryMapping and OptionalMapping relationships, it is 48. The StandardTerminalNameAssignment class also has a self-loop relationship with multiplicity 1..oo.</p>																																																				
type	CXL-x8-InterfaceFunctionType , CXL-x8-StandardTerminalNameAssignmentType , CXL-x8-MandatoryMappingType , CXL-x8-MandatoryStandardTerminalNameType , CXL-x8-OptionalMappingType , CXL-x8-OptionalStandardTerminalNameType																																																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr><td>1. PETp0</td><td>2. PETn0</td><td>3. PETp1</td><td>4. PETn1</td></tr> <tr><td>5. PETp2</td><td>6. PETn2</td><td>7. PETp3</td><td>8. PETn3</td></tr> <tr><td>9. PETp4</td><td>10. PETn4</td><td>11. PETp5</td><td>12. PETn5</td></tr> <tr><td>13. PETp6</td><td>14. PETn6</td><td>15. PETp7</td><td>16. PETn7</td></tr> <tr><td>17. PERp0</td><td>18. PERn0</td><td>19. PERp1</td><td>20. PERn1</td></tr> <tr><td>21. PERp2</td><td>22. PERn2</td><td>23. PERp3</td><td>24. PERn3</td></tr> <tr><td>25. PERp4</td><td>26. PERn4</td><td>27. PERp5</td><td>28. PERn5</td></tr> <tr><td>29. PERp6</td><td>30. PERn6</td><td>31. PERp7</td><td>32. PERn7</td></tr> <tr><td>33. REFCLKp0</td><td>34. REFCLKn0</td><td>35. PERST0#</td><td>36. PRSNT0#</td></tr> <tr><td>37. PRSNT1#</td><td>38. SMBCLK</td><td>39. SMBDATA</td><td>40. SMBRST#</td></tr> <tr><td>41. DUALPORTEN#</td><td>42. LED</td><td>43. PWRDIS</td><td>44. MFG</td></tr> <tr><td>45. RFU</td><td>46. 12V</td><td>47. 3.3Vaux</td><td>48. GND</td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. REFCLKp1</td><td>2. REFCLKn1</td><td>3. PERST1#</td><td>4. CLKREQ#</td></tr> </table>	1. PETp0	2. PETn0	3. PETp1	4. PETn1	5. PETp2	6. PETn2	7. PETp3	8. PETn3	9. PETp4	10. PETn4	11. PETp5	12. PETn5	13. PETp6	14. PETn6	15. PETp7	16. PETn7	17. PERp0	18. PERn0	19. PERp1	20. PERn1	21. PERp2	22. PERn2	23. PERp3	24. PERn3	25. PERp4	26. PERn4	27. PERp5	28. PERn5	29. PERp6	30. PERn6	31. PERp7	32. PERn7	33. REFCLKp0	34. REFCLKn0	35. PERST0#	36. PRSNT0#	37. PRSNT1#	38. SMBCLK	39. SMBDATA	40. SMBRST#	41. DUALPORTEN#	42. LED	43. PWRDIS	44. MFG	45. RFU	46. 12V	47. 3.3Vaux	48. GND	1. REFCLKp1	2. REFCLKn1	3. PERST1#	4. CLKREQ#
1. PETp0	2. PETn0	3. PETp1	4. PETn1																																																		
5. PETp2	6. PETn2	7. PETp3	8. PETn3																																																		
9. PETp4	10. PETn4	11. PETp5	12. PETn5																																																		
13. PETp6	14. PETn6	15. PETp7	16. PETn7																																																		
17. PERp0	18. PERn0	19. PERp1	20. PERn1																																																		
21. PERp2	22. PERn2	23. PERp3	24. PERn3																																																		
25. PERp4	26. PERn4	27. PERp5	28. PERn5																																																		
29. PERp6	30. PERn6	31. PERp7	32. PERn7																																																		
33. REFCLKp0	34. REFCLKn0	35. PERST0#	36. PRSNT0#																																																		
37. PRSNT1#	38. SMBCLK	39. SMBDATA	40. SMBRST#																																																		
41. DUALPORTEN#	42. LED	43. PWRDIS	44. MFG																																																		
45. RFU	46. 12V	47. 3.3Vaux	48. GND																																																		
1. REFCLKp1	2. REFCLKn1	3. PERST1#	4. CLKREQ#																																																		

4.7.3.10.4.3. CXL-x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/ComputeExpressLink/CXL-x16			
diagram				
type	CXL-x16-InterfaceFunctionType , CXL-x16-StandardTerminalNameAssignmentType , CXL-x16-MandatoryMappingType , CXL-x16-MandatoryStandardTerminalNameType , CXL-x16-OptionalMappingType , CXL-x16-OptionalStandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName 1. PETp0 2. PETn0 3. PETp1 4. PETn1 5. PETp2 6. PETn2 7. PETp3 8. PETn3 9. PETp4 10. PETn4 11. PETp5 12. PETn5 13. PETp6 14. PETn6 15. PETp7 16. PETn7 17. PETp8 18. PETn8 19. PETp9 20. PETn9 21. PETp10 22. PETn10 23. PETp11 24. PETn11 25. PETp12 26. PETn12 27. PETp13 28. PETn13 29. PETp14 30. PETn14 31. PETp15 32. PETn15 33. PERp0 34. PERn0 35. PERp1 36. PERn1 37. PERp2 38. PERn2 39. PERp3 40. PERn3 41. PERp4 42. PERn4 43. PERp5 44. PERn5 45. PERp6 46. PERn6 47. PERp7 48. PERn7 49. PERp8 50. PERn8 51. PERp9 52. PERn9 53. PERp10 54. PERn10 55. PERp11 56. PERn11 57. PERp12 58. PERn12 59. PERp13 60. PERn13 61. PERp14 62. PERn14 63. PERp15 64. PERn15 65. REFCLKp0 66. REFCLKn0 67. PERST0# 68. PRSNT0# 69. PRSNT1# 70. PRSNT2# 71. SMBCLK 72. SMBDATA 73. SMBRST# 74. DUALPORTEN# 75. LED 76. PWRDIS 77. MFG 78. RFU 79. 12V 80. 3.3Vaux 81. GND			
	OptionalMapping/StandardTerminalName 1. REFCLKp1 2. REFCLKn1 3. PERST1# 4. CLKREQ#			

4.7.3.10.5. Display Bus Function

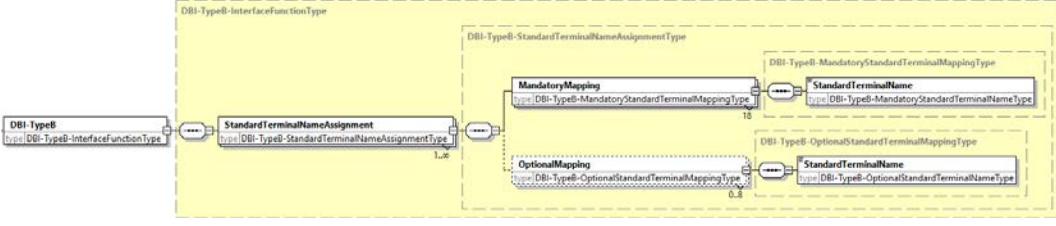
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus
diagram	<pre> classDiagram class DisplayBus { type DisplayBusInterfaceFunctionType } class DBI-TypeA { type DBI-TypeA-InterfaceFunctionType } class DBI-TypeB { type DBI-TypeB-InterfaceFunctionType } class DBI-TypeC { type DBI-TypeC-InterfaceFunctionType } class DPI-Type1 { type DPI-Type1-InterfaceFunctionType } class DPI-Type2-3 { type DPI-Type2-3-InterfaceFunctionType } class DPI-Type4 { type DPI-Type4-InterfaceFunctionType } class DSI-2OptionC { type DSI-2OptionC-InterfaceFunctionType } class DSI-2OptionD { type DSI-2OptionD-InterfaceFunctionType } DisplayBus < -- DBI-TypeA DBI-TypeA < -- DBI-TypeB DBI-TypeA < -- DBI-TypeC DBI-TypeA < -- DPI-Type1 DBI-TypeA < -- DPI-Type2-3 DBI-TypeA < -- DPI-Type4 DBI-TypeA < -- DSI-2OptionC DBI-TypeA < -- DSI-2OptionD </pre>
type	DisplayBusInterfaceFunctionType , DBI-TypeA-InterfaceFunctionType , DBI-TypeB-InterfaceFunctionType , DBI-TypeC-InterfaceFunctionType , DPI-Type1-InterfaceFunctionType , DPI-Type2-3-InterfaceFunctionType , DPI-Type4-InterfaceFunctionType , DSI-2OptionC-InterfaceFunctionType , DSI-2OptionD-InterfaceFunctionType .

For more information about the Display Bus Interface, refer to the MIPI Alliance standard Specification for Display Bus Interface Version 2.0

4.7.3.10.5.1. DBI-TypeA

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DBI-TypeA																																								
diagram	<pre> classDiagram class DBI-TypeA { <<DBI-TypeA>> <<DBI-TypeA-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DBI-TypeA-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<DBI-TypeA-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DBI-TypeA-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<DBI-TypeA-MandatoryStandardTerminalNameType>> } class StandardTerminalName { <<DBI-TypeA-OptionalStandardTerminalNameType>> } DBI-TypeA "1..2" --> StandardTerminalNameAssignment : <<DBI-TypeA-InterfaceFunctionType>> DBI-TypeA "1..2" --> MandatoryMapping : <<DBI-TypeA-StandardTerminalNameAssignmentType>> DBI-TypeA "1..2" --> OptionalMapping : <<DBI-TypeA-StandardTerminalNameAssignmentType>> StandardTerminalNameAssignment "1..2" --> MandatoryMapping : <<DBI-TypeA-MandatoryStandardTerminalMappingType>> StandardTerminalNameAssignment "1..2" --> OptionalMapping : <<DBI-TypeA-OptionalStandardTerminalMappingType>> MandatoryMapping "1..2" --> StandardTerminalName : <<DBI-TypeA-MandatoryStandardTerminalNameType>> OptionalMapping "0..1" --> StandardTerminalName : <<DBI-TypeA-OptionalStandardTerminalNameType>> </pre>																																								
type	DBI-TypeA-InterfaceFunctionType , DBI-TypeA-StandardTerminalNameAssignmentType , DBI-TypeA-MandatoryStandardTerminalMappingType , DBI-TypeA-MandatoryStandardTerminalNameType , DBI-TypeA-OptionalStandardTerminalMappingType , DBI-TypeA-OptionalStandardTerminalNameType .																																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. VDD</td><td>2. AGND</td><td>3. VDDI</td><td>4. DGND</td></tr> <tr> <td>5. CSX</td><td>6. RESX</td><td>7. D/CD</td><td>8. R/WX</td></tr> <tr> <td>9. E</td><td>10. D[0]</td><td>11. D[1]</td><td>12. D[2]</td></tr> <tr> <td>13. D[3]</td><td>14. D[4]</td><td>15. D[5]</td><td>16. D[6]</td></tr> <tr> <td>17. D[7]</td><td></td><td></td><td></td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. D[8]</td><td>2. D[9]</td><td>3. D[10]</td><td>4. D[11]</td></tr> <tr> <td>5. D[12]</td><td>6. D[13]</td><td>7. D[14]</td><td>8. D[15]</td></tr> <tr> <td>9. TE</td><td></td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDD	2. AGND	3. VDDI	4. DGND	5. CSX	6. RESX	7. D/CD	8. R/WX	9. E	10. D[0]	11. D[1]	12. D[2]	13. D[3]	14. D[4]	15. D[5]	16. D[6]	17. D[7]				OptionalMapping/StandardTerminalName				1. D[8]	2. D[9]	3. D[10]	4. D[11]	5. D[12]	6. D[13]	7. D[14]	8. D[15]	9. TE			
MandatoryMapping/StandardTerminalName																																									
1. VDD	2. AGND	3. VDDI	4. DGND																																						
5. CSX	6. RESX	7. D/CD	8. R/WX																																						
9. E	10. D[0]	11. D[1]	12. D[2]																																						
13. D[3]	14. D[4]	15. D[5]	16. D[6]																																						
17. D[7]																																									
OptionalMapping/StandardTerminalName																																									
1. D[8]	2. D[9]	3. D[10]	4. D[11]																																						
5. D[12]	6. D[13]	7. D[14]	8. D[15]																																						
9. TE																																									

4.7.3.10.5.2. DBI-TypeB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DBI-TypeB																																				
diagram																																					
type	<p>DBI-TypeB-InterfaceFunctionType, DBI-TypeB-StandardTerminalNameAssignmentType,</p> <p>DBI-TypeB-MandatoryStandardTerminalMappingType,</p> <p>DBI-TypeB-MandatoryStandardTerminalNameType, DBI-TypeB-OptionalStandardTerminalMappingType,</p> <p>DBI-TypeB-OptionalStandardTerminalNameType.</p>																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. VDD</td><td>2. AGND</td><td>3. VDDI</td><td>4. DGND</td></tr> <tr> <td>5. CSX</td><td>6. RESX</td><td>7. TE</td><td>8. D/CX</td></tr> <tr> <td>9. WRX</td><td>10. RDX</td><td>11. D[0]</td><td>12. D[1]</td></tr> <tr> <td>13. D[2]</td><td>14. D[3]</td><td>15. D[4]</td><td>16. D[5]</td></tr> <tr> <td>17. D[6]</td><td>18. D[7]</td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th></tr> <tr> <td>1. D[8]</td><td>2. D[9]</td><td>3. D[10]</td><td>4. D[11]</td></tr> <tr> <td>5. D[12]</td><td>6. D[13]</td><td>7. D[14]</td><td>8. D[15]</td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDD	2. AGND	3. VDDI	4. DGND	5. CSX	6. RESX	7. TE	8. D/CX	9. WRX	10. RDX	11. D[0]	12. D[1]	13. D[2]	14. D[3]	15. D[4]	16. D[5]	17. D[6]	18. D[7]			OptionalMapping/StandardTerminalName				1. D[8]	2. D[9]	3. D[10]	4. D[11]	5. D[12]	6. D[13]	7. D[14]	8. D[15]
MandatoryMapping/StandardTerminalName																																					
1. VDD	2. AGND	3. VDDI	4. DGND																																		
5. CSX	6. RESX	7. TE	8. D/CX																																		
9. WRX	10. RDX	11. D[0]	12. D[1]																																		
13. D[2]	14. D[3]	15. D[4]	16. D[5]																																		
17. D[6]	18. D[7]																																				
OptionalMapping/StandardTerminalName																																					
1. D[8]	2. D[9]	3. D[10]	4. D[11]																																		
5. D[12]	6. D[13]	7. D[14]	8. D[15]																																		

4.7.3.10.5.3. DBI-TypeC

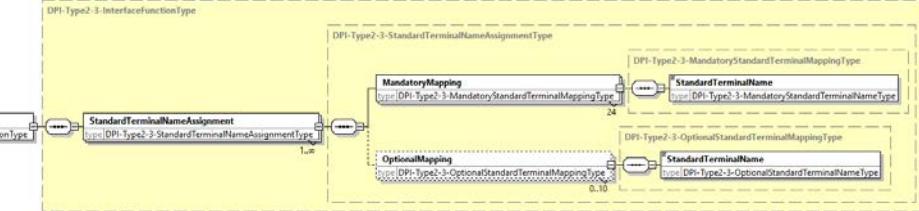
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DBI-TypeC																				
diagram	<pre> classDiagram class DBI-TypeC-InterfaceFunctionType class DBI-TypeC-StandardTerminalNameAssignmentType class DBI-TypeC-MandatoryStandardTerminalMappingType class DBI-TypeC-OptionalStandardTerminalMappingType class StandardTerminalName class StandardTerminalMappingType DBI-TypeC-InterfaceFunctionType < -- DBI-TypeC-StandardTerminalNameAssignmentType DBI-TypeC-StandardTerminalNameAssignmentType < -- DBI-TypeC-MandatoryStandardTerminalMappingType DBI-TypeC-StandardTerminalNameAssignmentType < -- DBI-TypeC-OptionalStandardTerminalMappingType DBI-TypeC-MandatoryStandardTerminalMappingType --> StandardTerminalName DBI-TypeC-OptionalStandardTerminalMappingType --> StandardTerminalName StandardTerminalName --> StandardTerminalMappingType </pre>																				
type	DBI-TypeB-InterfaceFunctionType , DBI-TypeC-StandardTerminalNameAssignmentType , DBI-TypeC-MandatoryStandardTerminalMappingType , DBI-TypeC-MandatoryStandardTerminalNameType , DBI-TypeC-OptionalStandardTerminalMappingType , DBI-TypeC-OptionalStandardTerminalNameType .																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. VDD</td> <td>2. AGND</td> <td>3. VDDI</td> <td>4. DGND</td> </tr> <tr> <td>5. CSX</td> <td>6. RESX</td> <td>7. SCL</td> <td></td> </tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. D/CX</td> <td>2. DOUT</td> <td>3. DIN</td> <td>4. SDA</td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDD	2. AGND	3. VDDI	4. DGND	5. CSX	6. RESX	7. SCL		OptionalMapping/StandardTerminalName				1. D/CX	2. DOUT	3. DIN	4. SDA
MandatoryMapping/StandardTerminalName																					
1. VDD	2. AGND	3. VDDI	4. DGND																		
5. CSX	6. RESX	7. SCL																			
OptionalMapping/StandardTerminalName																					
1. D/CX	2. DOUT	3. DIN	4. SDA																		

4.7.3.10.5.4. DPI-Type1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DPI-Type1																												
diagram	<pre> classDiagram class DPI-Type1-InterfaceFunctionType class DPI-Type1-StandardTerminalNameAssignmentType class StandardTerminalName DPI-Type1-InterfaceFunctionType < -- DPI-Type1-StandardTerminalNameAssignmentType DPI-Type1-StandardTerminalNameAssignmentType --> StandardTerminalName </pre>																												
type	DPI-Type1-InterfaceFunctionType , DPI-Type1-StandardTerminalNameAssignmentType , DPI-Type1-StandardTerminalMappingType , DPI-Type1-StandardTerminalNameType .																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. DPI_PCLK</td> <td>2. DPI_VS</td> <td>3. DPI_HS</td> <td>4. DPI_ACDE</td> </tr> <tr> <td>5. DPI_PixDat[0]</td> <td>6. DPI_PixDat[1]</td> <td>7. DPI_PixDat[2]</td> <td>8. DPI_PixDat[3]</td> </tr> <tr> <td>9. DPI_PixDat[4]</td> <td>10. DPI_PixDat[5]</td> <td>11. DPI_PixDat[6]</td> <td>12. DPI_PixDat[7]</td> </tr> <tr> <td>13. DPI_PixDat[8]</td> <td>14. DPI_PixDat[9]</td> <td>15. DPI_PixDat[10]</td> <td>16. DPI_PixDat[11]</td> </tr> <tr> <td>17. DPI_PixDat[12]</td> <td>18. DPI_PixDat[13]</td> <td>19. DPI_PixDat[14]</td> <td>20. DPI_PixDat[15]</td> </tr> <tr> <td>21. DPI_PixDat[16]</td> <td>22. DPI_PixDat[17]</td> <td></td> <td></td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. DPI_PCLK	2. DPI_VS	3. DPI_HS	4. DPI_ACDE	5. DPI_PixDat[0]	6. DPI_PixDat[1]	7. DPI_PixDat[2]	8. DPI_PixDat[3]	9. DPI_PixDat[4]	10. DPI_PixDat[5]	11. DPI_PixDat[6]	12. DPI_PixDat[7]	13. DPI_PixDat[8]	14. DPI_PixDat[9]	15. DPI_PixDat[10]	16. DPI_PixDat[11]	17. DPI_PixDat[12]	18. DPI_PixDat[13]	19. DPI_PixDat[14]	20. DPI_PixDat[15]	21. DPI_PixDat[16]	22. DPI_PixDat[17]		
MandatoryMapping/StandardTerminalName																													
1. DPI_PCLK	2. DPI_VS	3. DPI_HS	4. DPI_ACDE																										
5. DPI_PixDat[0]	6. DPI_PixDat[1]	7. DPI_PixDat[2]	8. DPI_PixDat[3]																										
9. DPI_PixDat[4]	10. DPI_PixDat[5]	11. DPI_PixDat[6]	12. DPI_PixDat[7]																										
13. DPI_PixDat[8]	14. DPI_PixDat[9]	15. DPI_PixDat[10]	16. DPI_PixDat[11]																										
17. DPI_PixDat[12]	18. DPI_PixDat[13]	19. DPI_PixDat[14]	20. DPI_PixDat[15]																										
21. DPI_PixDat[16]	22. DPI_PixDat[17]																												

For more information about the Display Pixel Interface, refer to the MIPI Alliance standard Specification for Display Pixel Interface Version 1.0

4.7.3.10.5.5. DPI-Type2-3

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DPI-Type2-3																																				
diagram																																					
type	DPI-Type2-3-InterfaceFunctionType , DPI-Type2-3-StandardTerminalNameAssignmentType , DPI-Type2-3-MandatoryStandardTerminalMappingType , DPI-Type2-3-MandatoryStandardTerminalNameType , DPI-Type2-3-OptionalStandardTerminalMappingType , DPI-Type2-3-OptionalStandardTerminalNameType .																																				
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. VDD</td><td>2. AGND</td><td>3. VDDI</td><td>4. DGND</td></tr> <tr><td>5. VSYNC</td><td>6. HSYNC</td><td>7. DE</td><td>8. PCLK</td></tr> <tr><td>9. D[0]</td><td>10. D[1]</td><td>11. D[2]</td><td>12. D[3]</td></tr> <tr><td>13. D[4]</td><td>14. D[5]</td><td>15. D[6]</td><td>16. D[7]</td></tr> <tr><td>17. D[8]</td><td>18. D[9]</td><td>19. D[10]</td><td>20. D[11]</td></tr> <tr><td>21. D[12]</td><td>22. D[13]</td><td>23. D[14]</td><td>24. D[15]</td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. D[16]</td><td>2. D[17]</td><td>3. D[18]</td><td>4. D[19]</td></tr> <tr><td>5. D[20]</td><td>6. D[21]</td><td>7. D[22]</td><td>8. D[23]</td></tr> <tr><td>9. SD</td><td>10. CM</td><td></td><td></td></tr> </table>	1. VDD	2. AGND	3. VDDI	4. DGND	5. VSYNC	6. HSYNC	7. DE	8. PCLK	9. D[0]	10. D[1]	11. D[2]	12. D[3]	13. D[4]	14. D[5]	15. D[6]	16. D[7]	17. D[8]	18. D[9]	19. D[10]	20. D[11]	21. D[12]	22. D[13]	23. D[14]	24. D[15]	1. D[16]	2. D[17]	3. D[18]	4. D[19]	5. D[20]	6. D[21]	7. D[22]	8. D[23]	9. SD	10. CM		
1. VDD	2. AGND	3. VDDI	4. DGND																																		
5. VSYNC	6. HSYNC	7. DE	8. PCLK																																		
9. D[0]	10. D[1]	11. D[2]	12. D[3]																																		
13. D[4]	14. D[5]	15. D[6]	16. D[7]																																		
17. D[8]	18. D[9]	19. D[10]	20. D[11]																																		
21. D[12]	22. D[13]	23. D[14]	24. D[15]																																		
1. D[16]	2. D[17]	3. D[18]	4. D[19]																																		
5. D[20]	6. D[21]	7. D[22]	8. D[23]																																		
9. SD	10. CM																																				

For more information about the Display Pixel Interface Type 2, Type 3, Type 4, refer to the MIPI Alliance standard Specification for Display Pixel Interface (DPI-2) Version 2.0

4.7.3.10.5.6. DPI-Type4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DPI-Type4																																												
diagram	<p>Detailed description of the UML diagram:</p> <ul style="list-style-type: none"> DPI-Type4-InterfaceFunctionType is the base class. DPI-Type4 inherits from it. StandardTerminalNameAssignment has a multiplicity of 1..x. MandatoryMapping and OptionalMapping are associated with StandardTerminalNameAssignment. DPI-Type4-StandardTerminalNameAssignmentType is a type of StandardTerminalNameAssignment. DPI-Type4-MandatoryStandardTerminalMappingType is a type of MandatoryMapping. DPI-Type4-OptionalStandardTerminalMappingType is a type of OptionalMapping. StandardTerminalName is a type of StandardTerminalNameAssignment. DPI-Type4-MandatoryStandardTerminalNameType is a type of StandardTerminalName. 																																												
type	DPI-Type4-InterfaceFunctionType , DPI-Type4-StandardTerminalNameAssignmentType , DPI-Type4-MandatoryStandardTerminalMappingType , DPI-Type4-MandatoryStandardTerminalNameType , DPI-Type4-OptionalStandardTerminalMappingType , DPI-Type4-OptionalStandardTerminalNameType .																																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDD</td><td>2. AGND</td><td>3. VDDI</td><td>4. DGND</td></tr> <tr><td>5. VSYNC</td><td>6. HSYNC</td><td>7. DE</td><td>8. PCLK</td></tr> <tr><td>9. SD</td><td>10. CM</td><td>11. D[0]</td><td>12. D[1]</td></tr> <tr><td>13. D[2]</td><td>14. D[3]</td><td>15. D[4]</td><td>16. D[5]</td></tr> <tr><td>17. D[6]</td><td>18. D[7]</td><td>19. D[8]</td><td>20. D[9]</td></tr> <tr><td>21. D[10]</td><td>22. D[11]</td><td>23. D[12]</td><td>24. D[13]</td></tr> <tr><td>25. D[14]</td><td>26. D[15]</td><td></td><td></td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. D[16]</td><td>2. D[17]</td><td>3. D[18]</td><td>4. D[19]</td></tr> <tr><td>5. D[20]</td><td>6. D[21]</td><td>7. D[22]</td><td>8. D[23]</td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDD	2. AGND	3. VDDI	4. DGND	5. VSYNC	6. HSYNC	7. DE	8. PCLK	9. SD	10. CM	11. D[0]	12. D[1]	13. D[2]	14. D[3]	15. D[4]	16. D[5]	17. D[6]	18. D[7]	19. D[8]	20. D[9]	21. D[10]	22. D[11]	23. D[12]	24. D[13]	25. D[14]	26. D[15]			OptionalMapping/StandardTerminalName				1. D[16]	2. D[17]	3. D[18]	4. D[19]	5. D[20]	6. D[21]	7. D[22]	8. D[23]
MandatoryMapping/StandardTerminalName																																													
1. VDD	2. AGND	3. VDDI	4. DGND																																										
5. VSYNC	6. HSYNC	7. DE	8. PCLK																																										
9. SD	10. CM	11. D[0]	12. D[1]																																										
13. D[2]	14. D[3]	15. D[4]	16. D[5]																																										
17. D[6]	18. D[7]	19. D[8]	20. D[9]																																										
21. D[10]	22. D[11]	23. D[12]	24. D[13]																																										
25. D[14]	26. D[15]																																												
OptionalMapping/StandardTerminalName																																													
1. D[16]	2. D[17]	3. D[18]	4. D[19]																																										
5. D[20]	6. D[21]	7. D[22]	8. D[23]																																										

4.7.3.10.5.7. DSI-2-OptionC

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DSI-2OptionC																								
diagram	<pre> classDiagram class DSI-2OptionC-InterfaceFunctionType class DSI-2OptionC-StandardTerminalNameAssignment class DSI-2OptionC-MandatoryStandardTerminalMapping class DSI-2OptionC-OptionalStandardTerminalMapping class StandardTerminalName DSI-2OptionC-InterfaceFunctionType "1..1" --> "1..1" DSI-2OptionC-StandardTerminalNameAssignment DSI-2OptionC-StandardTerminalNameAssignment "1..1" --> "1..1" DSI-2OptionC-MandatoryStandardTerminalMapping DSI-2OptionC-StandardTerminalNameAssignment "1..1" --> "1..1" DSI-2OptionC-OptionalStandardTerminalMapping DSI-2OptionC-MandatoryStandardTerminalMapping "1..1" --> "1..1" StandardTerminalName DSI-2OptionC-OptionalStandardTerminalMapping "1..1" --> "1..1" StandardTerminalName </pre>																								
type	DSI-2OptionC-InterfaceFunctionType , DSI-2OptionC-StandardTerminalNameAssignmentType , DSI-2OptionC-MandatoryStandardTerminalMappingType , DSI-2OptionC-MandatoryStandardTerminalNameType , DSI-2OptionC-OptionalStandardTerminalMappingType , DSI-2OptionC-OptionalStandardTerminalNameType .																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> <tr> <td>1. Data0_A</td> <td>2. Data0_B</td> <td>3. Data0_C</td> <td></td> </tr> </thead> <tbody> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Data1_A</td> <td>2. Data1_B</td> <td>3. Data1_C</td> <td>4. Data2_A</td> </tr> <tr> <td>5. Data2_B</td> <td>6. Data2_C</td> <td>7. Data3_A</td> <td>8. Data3_B</td> </tr> <tr> <td>9. Data3_C</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. Data0_A	2. Data0_B	3. Data0_C		OptionalMapping/StandardTerminalName				1. Data1_A	2. Data1_B	3. Data1_C	4. Data2_A	5. Data2_B	6. Data2_C	7. Data3_A	8. Data3_B	9. Data3_C			
MandatoryMapping/StandardTerminalName																									
1. Data0_A	2. Data0_B	3. Data0_C																							
OptionalMapping/StandardTerminalName																									
1. Data1_A	2. Data1_B	3. Data1_C	4. Data2_A																						
5. Data2_B	6. Data2_C	7. Data3_A	8. Data3_B																						
9. Data3_C																									

For more information about the Display Serial Interface 2 (DSI-2) refer to the MIPI Alliance standard Specification for Display Serial Interface 2 (Dsi-2) Version 2.0

4.7.3.10.5.8. DSI-2-OptionD

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DSI-2OptionD																				
diagram	<pre> classDiagram class DSI-2OptionD { <<DSI-2OptionD-InterfaceFunctionType>> <<DSI-2OptionD-StandardTerminalNameAssignmentType>> <<DSI-2OptionD-MandatoryStandardTerminalMappingType>> <<DSI-2OptionD-OptionalStandardTerminalMappingType>> } class StandardTerminalNameAssignment { <<DSI-2OptionD-StandardTerminalNameAssignmentType>> <<DSI-2OptionD-MandatoryStandardTerminalMappingType>> <<DSI-2OptionD-OptionalStandardTerminalMappingType>> } class MandatoryMapping { <<DSI-2OptionD-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DSI-2OptionD-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<DSI-2OptionD-MandatoryStandardTerminalNameType>> <<DSI-2OptionD-OptionalStandardTerminalNameType>> } DSI-2OptionD "1..<<DSI-2OptionD-InterfaceFunctionType>>" --> InterfaceFunctionType DSI-2OptionD "1..<<DSI-2OptionD-StandardTerminalNameAssignmentType>>" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..<<DSI-2OptionD-MandatoryStandardTerminalMappingType>>" --> MandatoryMapping StandardTerminalNameAssignment "1..<<DSI-2OptionD-OptionalStandardTerminalMappingType>>" --> OptionalMapping MandatoryMapping "1..<<DSI-2OptionD-MandatoryStandardTerminalNameType>>" --> StandardTerminalName OptionalMapping "0..6<<DSI-2OptionD-OptionalStandardTerminalNameType>>" --> StandardTerminalName </pre>																				
type	DSI-2OptionD-InterfaceFunctionType, DSI-2OptionD-StandardTerminalNameAssignmentType, DSI-2OptionD-MandatoryStandardTerminalMappingType, DSI-2OptionD-MandatoryStandardTerminalNameType, DSI-2OptionD-OptionalStandardTerminalMappingType, DSI-2OptionD-OptionalStandardTerminalNameType.																				
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Data0+</td> <td>2. Data0-</td> <td>3. Clock+</td> <td>4. Clock-</td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Data1+</td> <td>2. Data1-</td> <td>3. Data2+</td> <td>4. Data2-</td> </tr> <tr> <td>5. Data3+</td> <td>6. Data3-</td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Data0+	2. Data0-	3. Clock+	4. Clock-	OptionalMapping/StandardTerminalName				1. Data1+	2. Data1-	3. Data2+	4. Data2-	5. Data3+	6. Data3-		
MandatoryMapping/StandardTerminalName																					
1. Data0+	2. Data0-	3. Clock+	4. Clock-																		
OptionalMapping/StandardTerminalName																					
1. Data1+	2. Data1-	3. Data2+	4. Data2-																		
5. Data3+	6. Data3-																				

4.7.3.10.6. DDR3 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3
diagram	<pre> graph LR DDR3[DDR3 type: DDR3-InterfaceFunctionType] --- DDR3IF[DDR3-InterfaceFunctionType] DDR3IF --- DDR3x4[DDR3-x4 type: DDR3-x4-InterfaceFunctionType] DDR3IF --- DDR3x4DD[DDR3-x4-DualDie type: DDR3-x4-DualDie-InterfaceFunctionType] DDR3IF --- DDR3x4QD[DDR3-x4-QuadDie type: DDR3-x4-QuadDie-InterfaceFunctionType] DDR3IF --- DDR3x8[DDR3-x8 type: DDR3-x8-InterfaceFunctionType] DDR3IF --- DDR3x8DD[DDR3-x8-DualDie type: DDR3-x8-DualDie-InterfaceFunctionType] DDR3IF --- DDR3x8QD[DDR3-x8-QuadDie type: DDR3-x8-QuadDie-InterfaceFunctionType] DDR3IF --- DDR3x16[DDR3-x16 type: DDR3-x16-InterfaceFunctionType] DDR3IF --- DDR3x16DD[DDR3-x16-DualDie type: DDR3-x16-DualDie-InterfaceFunctionType] DDR3IF --- DDR3x16QD[DDR3-x16-QuadDie type: DDR3-x16-QuadDie-InterfaceFunctionType] DDR3IF --- DDR3Controller[DDR3-Controller type: DDR3-Controller-InterfaceFunctionType] </pre>
type	DDR3-InterfaceFunctionType , DDR3-x4-InterfaceFunctionType , DDR3-x4-DualDie-InterfaceFunctionType , DDR3-x4-QuadDie-InterfaceFunctionType , DDR3-x8-InterfaceFunctionType , DDR3-x8-DualDie-InterfaceFunctionType , DDR3-x8-QuadDie-InterfaceFunctionType , DDR3-x16-InterfaceFunctionType , DDR3-x16-DualDie-InterfaceFunctionType , DDR3-x16-QuadDie-InterfaceFunctionType , DDR3-Controller-InterfaceFunctionType .

For more information about the DDR3 Interface, refer to the JEDEC standard JESD79-3F.

4.7.3.10.6.1. DDR3-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x4																																																				
diagram	<pre> classDiagram DDR3-x4 "type DDR3-x4-InterfaceFunctionType" --> StandardTerminalNameAssignment "type DDR3-x4StandardTerminalNameAssignmentType" StandardTerminalNameAssignment --> MandatoryMapping "type DDR3-x4MandatoryStandardTerminalMappingType" MandatoryMapping --> StandardTerminalName "type DDR3-x4MandatoryStandardTerminalNameType" constraints { DDR3-x4 StandardTerminalNameAssignment MandatoryMapping StandardTerminalName } </pre>																																																				
type	<p>DDR3-x4-InterfaceFunctionType, DDR3-x4StandardTerminalNameAssignmentType, DDR3-x4MandatoryStandardTerminalMappingType, DDR3-x4MandatoryStandardTerminalNameType, DDR3-x4OptionalStandardTerminalMappingType, DDR3-x4OptionalStandardTerminalNameType.</p>																																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. A0</td><td>2. A1</td><td>3. A2</td><td>4. A3</td></tr> <tr><td>5. A4</td><td>6. A5</td><td>7. A6</td><td>8. A7</td></tr> <tr><td>9. A8</td><td>10. A9</td><td>11. A10/AP</td><td>12. A11</td></tr> <tr><td>13. A12/BC#</td><td>14. BA0</td><td>15. BA1</td><td>16. BA2</td></tr> <tr><td>17. CAS#</td><td>18. CK</td><td>19. CK#</td><td>20. CKE</td></tr> <tr><td>21. CS#</td><td>22. DM</td><td>23. DQ[0]</td><td>24. DQ[1]</td></tr> <tr><td>25. DQ[2]</td><td>26. DQ[3]</td><td>27. DQS</td><td>28. DQS#</td></tr> <tr><td>29. ODT</td><td>30. RAS#</td><td>31. RESET#</td><td>32. VDD</td></tr> <tr><td>33. VDDQ</td><td>34. VPP</td><td>35. VREFCA</td><td>36. VREFDQ</td></tr> <tr><td>37. VSS</td><td>38. VSSQ</td><td>39. WE#</td><td>40. ZQ</td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. A13</td><td>2. A14</td><td>3. A15</td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. A0	2. A1	3. A2	4. A3	5. A4	6. A5	7. A6	8. A7	9. A8	10. A9	11. A10/AP	12. A11	13. A12/BC#	14. BA0	15. BA1	16. BA2	17. CAS#	18. CK	19. CK#	20. CKE	21. CS#	22. DM	23. DQ[0]	24. DQ[1]	25. DQ[2]	26. DQ[3]	27. DQS	28. DQS#	29. ODT	30. RAS#	31. RESET#	32. VDD	33. VDDQ	34. VPP	35. VREFCA	36. VREFDQ	37. VSS	38. VSSQ	39. WE#	40. ZQ	OptionalMapping/StandardTerminalName				1. A13	2. A14	3. A15	
MandatoryMapping/StandardTerminalName																																																					
1. A0	2. A1	3. A2	4. A3																																																		
5. A4	6. A5	7. A6	8. A7																																																		
9. A8	10. A9	11. A10/AP	12. A11																																																		
13. A12/BC#	14. BA0	15. BA1	16. BA2																																																		
17. CAS#	18. CK	19. CK#	20. CKE																																																		
21. CS#	22. DM	23. DQ[0]	24. DQ[1]																																																		
25. DQ[2]	26. DQ[3]	27. DQS	28. DQS#																																																		
29. ODT	30. RAS#	31. RESET#	32. VDD																																																		
33. VDDQ	34. VPP	35. VREFCA	36. VREFDQ																																																		
37. VSS	38. VSSQ	39. WE#	40. ZQ																																																		
OptionalMapping/StandardTerminalName																																																					
1. A13	2. A14	3. A15																																																			

4.7.3.10.6.2. DDR3-x4 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x4-DualDie																																																								
diagram	<pre> classDiagram class DDR3-x4-DualDie-InterfaceFunctionType { StandardTerminalNameAssignment MandatoryMapping OptionalMapping StandardTerminalName } class DDR3-x4-DualDieStandardTerminalNameAssignmentType { StandardTerminalNameAssignment MandatoryMapping OptionalMapping StandardTerminalName } class DDR3-x4-DualDieMandatoryStandardTerminalMappingType { MandatoryMapping StandardTerminalName } class DDR3-x4-DualDieOptionalStandardTerminalMappingType { OptionalMapping StandardTerminalName } DDR3-x4-DualDie-InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" DDR3-x4-DualDieStandardTerminalNameAssignmentType DDR3-x4-DualDieStandardTerminalNameAssignmentType "1..>" MandatoryMapping MandatoryMapping "1..>" DDR3-x4-DualDieMandatoryStandardTerminalMappingType MandatoryMapping "1..>" DDR3-x4-DualDieOptionalStandardTerminalMappingType DDR3-x4-DualDieMandatoryStandardTerminalMappingType "1..>" StandardTerminalName DDR3-x4-DualDieOptionalStandardTerminalMappingType "0..3" StandardTerminalName StandardTerminalName "1..>" constraints </pre>																																																								
type	DDR3-x4-DualDie-InterfaceFunctionType , DDR3-x4-DualDieStandardTerminalNameAssignmentType , DDR3-x4-DualDieMandatoryStandardTerminalMappingType , DDR3-x4-DualDieMandatoryStandardTerminalNameType , DDR3-x4-DualDieOptionalStandardTerminalMappingType , DDR3-x4-DualDieOptionalStandardTerminalNameType .																																																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS</td><td>7. DQS#</td><td>8. DM</td></tr> <tr><td>9. CK</td><td>10. CK#</td><td>11. CKE0</td><td>12. CKE1</td></tr> <tr><td>13. ZQ0</td><td>14. ZQ1</td><td>15. RESET#</td><td>16. ODT0</td></tr> <tr><td>17. ODT1</td><td>18. CS0#</td><td>19. CS1#</td><td>20. WE#</td></tr> <tr><td>21. RAS#</td><td>22. CAS#</td><td>23. BA0</td><td>24. BA1</td></tr> <tr><td>25. BA2</td><td>26. VREFCA</td><td>27. VREFDQ</td><td>28. A0</td></tr> <tr><td>29. A1</td><td>30. A2</td><td>31. A3</td><td>32. A4</td></tr> <tr><td>33. A5</td><td>34. A6</td><td>35. A7</td><td>36. A8</td></tr> <tr><td>37. A9</td><td>38. A10/AP</td><td>39. A11</td><td>40. A12/BC#</td></tr> <tr><td>41. DQ[0]</td><td>42. DQ[1]</td><td>43. DQ[2]</td><td>44. DQ[3]</td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. A13</td><td>2. A13</td><td>3. A13</td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS	7. DQS#	8. DM	9. CK	10. CK#	11. CKE0	12. CKE1	13. ZQ0	14. ZQ1	15. RESET#	16. ODT0	17. ODT1	18. CS0#	19. CS1#	20. WE#	21. RAS#	22. CAS#	23. BA0	24. BA1	25. BA2	26. VREFCA	27. VREFDQ	28. A0	29. A1	30. A2	31. A3	32. A4	33. A5	34. A6	35. A7	36. A8	37. A9	38. A10/AP	39. A11	40. A12/BC#	41. DQ[0]	42. DQ[1]	43. DQ[2]	44. DQ[3]	OptionalMapping/StandardTerminalName				1. A13	2. A13	3. A13	
MandatoryMapping/StandardTerminalName																																																									
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																						
5. VSS	6. DQS	7. DQS#	8. DM																																																						
9. CK	10. CK#	11. CKE0	12. CKE1																																																						
13. ZQ0	14. ZQ1	15. RESET#	16. ODT0																																																						
17. ODT1	18. CS0#	19. CS1#	20. WE#																																																						
21. RAS#	22. CAS#	23. BA0	24. BA1																																																						
25. BA2	26. VREFCA	27. VREFDQ	28. A0																																																						
29. A1	30. A2	31. A3	32. A4																																																						
33. A5	34. A6	35. A7	36. A8																																																						
37. A9	38. A10/AP	39. A11	40. A12/BC#																																																						
41. DQ[0]	42. DQ[1]	43. DQ[2]	44. DQ[3]																																																						
OptionalMapping/StandardTerminalName																																																									
1. A13	2. A13	3. A13																																																							

4.7.3.10.6.3. DDR3-x4 – Quad Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x4-QuadDie																																																				
diagram	<pre> classDiagram class DDR3-x4-QuadDie { <<DDR3-x4-QuadDie>> } class StandardTerminalNameAssignment { <<DDR3-x4-QuadDieStandardTerminalNameAssignment>> } class DDR3-x4-QuadDie-InterfaceFunctionType { <<DDR3-x4-QuadDie-InterfaceFunctionType>> } class DDR3-x4-QuadDieStandardTerminalNameAssignmentType { <<DDR3-x4-QuadDieStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<DDR3-x4-QuadDieMandatoryStandardTerminalMapping>> } class StandardTerminalName { <<DDR3-x4-QuadDieMandatoryStandardTerminalName>> } class OptionalMapping { <<DDR3-x4-QuadDieOptionalStandardTerminalMapping>> } class DDR3-x4-QuadDieOptionalStandardTerminalMappingType { <<DDR3-x4-QuadDieOptionalStandardTerminalMappingType>> } class DDR3-x4-QuadDieOptionalStandardTerminalNameType { <<DDR3-x4-QuadDieOptionalStandardTerminalNameType>> } DDR3-x4-QuadDie "1..x" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..x" DDR3-x4-QuadDie-InterfaceFunctionType DDR3-x4-QuadDie-InterfaceFunctionType "1..x" DDR3-x4-QuadDieStandardTerminalNameAssignmentType DDR3-x4-QuadDieStandardTerminalNameAssignmentType "1..x" MandatoryMapping MandatoryMapping "1..x" StandardTerminalName DDR3-x4-QuadDieStandardTerminalNameAssignmentType "1..x" OptionalMapping OptionalMapping "1..x" StandardTerminalName StandardTerminalNameAssignment "1..x" constraints </pre>																																																				
type	DDR3-x4-QuadDie-InterfaceFunctionType , DDR3-x4-QuadDieStandardTerminalNameAssignmentType , DDR3-x4-QuadDieMandatoryStandardTerminalMappingType , DDR3-x4-QuadDieMandatoryStandardTerminalNameType , DDR3-x4-QuadDieOptionalStandardTerminalMappingType , DDR3-x4-QuadDieOptionalStandardTerminalNameType .																																																				
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS</td><td>7. DQS#</td><td>8. DM</td></tr> <tr><td>9. CK</td><td>10. CK#</td><td>11. CKE0</td><td>12. CKE1</td></tr> <tr><td>13. ZQ0</td><td>14. ZQ1</td><td>15. ZQ2</td><td>16. ZQ3</td></tr> <tr><td>17. RESET#</td><td>18. ODT0</td><td>19. ODT1</td><td>20. CS0#</td></tr> <tr><td>21. CS1#</td><td>22. CS2#</td><td>23. CS3#</td><td>24. WE#</td></tr> <tr><td>25. RAS#</td><td>26. CAS#</td><td>27. BA0</td><td>28. BA1</td></tr> <tr><td>29. BA2</td><td>30. VREFCA</td><td>31. VREFDQ</td><td>32. A0</td></tr> <tr><td>33. A1</td><td>34. A2</td><td>35. A3</td><td>36. A4</td></tr> <tr><td>37. A5</td><td>38. A6</td><td>39. A7</td><td>40. A8</td></tr> <tr><td>41. A9</td><td>42. A10/AP</td><td>43. A11</td><td>44. A12/BC#</td></tr> <tr><td>45. DQ[0]</td><td>46. DQ[1]</td><td>47. DQ[2]</td><td>48. DQ[3]</td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. A13</td><td>2. A13</td><td>3. A13</td><td></td></tr> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS	7. DQS#	8. DM	9. CK	10. CK#	11. CKE0	12. CKE1	13. ZQ0	14. ZQ1	15. ZQ2	16. ZQ3	17. RESET#	18. ODT0	19. ODT1	20. CS0#	21. CS1#	22. CS2#	23. CS3#	24. WE#	25. RAS#	26. CAS#	27. BA0	28. BA1	29. BA2	30. VREFCA	31. VREFDQ	32. A0	33. A1	34. A2	35. A3	36. A4	37. A5	38. A6	39. A7	40. A8	41. A9	42. A10/AP	43. A11	44. A12/BC#	45. DQ[0]	46. DQ[1]	47. DQ[2]	48. DQ[3]	1. A13	2. A13	3. A13	
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																		
5. VSS	6. DQS	7. DQS#	8. DM																																																		
9. CK	10. CK#	11. CKE0	12. CKE1																																																		
13. ZQ0	14. ZQ1	15. ZQ2	16. ZQ3																																																		
17. RESET#	18. ODT0	19. ODT1	20. CS0#																																																		
21. CS1#	22. CS2#	23. CS3#	24. WE#																																																		
25. RAS#	26. CAS#	27. BA0	28. BA1																																																		
29. BA2	30. VREFCA	31. VREFDQ	32. A0																																																		
33. A1	34. A2	35. A3	36. A4																																																		
37. A5	38. A6	39. A7	40. A8																																																		
41. A9	42. A10/AP	43. A11	44. A12/BC#																																																		
45. DQ[0]	46. DQ[1]	47. DQ[2]	48. DQ[3]																																																		
1. A13	2. A13	3. A13																																																			

4.7.3.10.6.4. DDR3-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x8																																																												
diagram	<pre> classDiagram DDR3-x8 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping MandatoryMapping "1..>" StandardTerminalName StandardTerminalNameAssignment "1..>" OptionalMapping OptionalMapping "1..>" StandardTerminalName constraints </pre>																																																												
type	DDR3-x8-InterfaceFunctionType , DDR3-x8StandardTerminalNameAssignmentType , DDR3-x8MandatoryStandardTerminalMappingType , DDR3-x8MandatoryStandardTerminalNameType , DDR3-x8OptionalStandardTerminalMappingType , DDR3-x8OptionalStandardTerminalNameType .																																																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS</td><td>7. DQS#</td><td>8. VREFCA</td></tr> <tr><td>9. VREFDQ</td><td>10. DM/TDQS</td><td>11. NU/TDQS#</td><td>12. CK</td></tr> <tr><td>13. CK#</td><td>14. CKE</td><td>15. ZQ</td><td>16. RESET#</td></tr> <tr><td>17. ODT</td><td>18. CS#</td><td>19. WE#</td><td>20. RAS#</td></tr> <tr><td>21. CAS#</td><td>22. BA0</td><td>23. BA1</td><td>24. BA2</td></tr> <tr><td>25. A0</td><td>26. A1</td><td>27. A2</td><td>28. A3</td></tr> <tr><td>29. A4</td><td>30. A5</td><td>31. A6</td><td>32. A7</td></tr> <tr><td>33. A8</td><td>34. A9</td><td>35. A10/AP</td><td>36. A11</td></tr> <tr><td>37. A12/BC#</td><td>38. DQ[0]</td><td>39. DQ[1]</td><td>40. DQ[2]</td></tr> <tr><td>41. DQ[3]</td><td>42. DQ[4]</td><td>43. DQ[5]</td><td>44. DQ[6]</td></tr> <tr><td>45. DQ[7]</td><td></td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. A13</td><td>2. A13</td><td>3. A13</td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS	7. DQS#	8. VREFCA	9. VREFDQ	10. DM/TDQS	11. NU/TDQS#	12. CK	13. CK#	14. CKE	15. ZQ	16. RESET#	17. ODT	18. CS#	19. WE#	20. RAS#	21. CAS#	22. BA0	23. BA1	24. BA2	25. A0	26. A1	27. A2	28. A3	29. A4	30. A5	31. A6	32. A7	33. A8	34. A9	35. A10/AP	36. A11	37. A12/BC#	38. DQ[0]	39. DQ[1]	40. DQ[2]	41. DQ[3]	42. DQ[4]	43. DQ[5]	44. DQ[6]	45. DQ[7]				OptionalMapping/StandardTerminalName				1. A13	2. A13	3. A13	
MandatoryMapping/StandardTerminalName																																																													
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																										
5. VSS	6. DQS	7. DQS#	8. VREFCA																																																										
9. VREFDQ	10. DM/TDQS	11. NU/TDQS#	12. CK																																																										
13. CK#	14. CKE	15. ZQ	16. RESET#																																																										
17. ODT	18. CS#	19. WE#	20. RAS#																																																										
21. CAS#	22. BA0	23. BA1	24. BA2																																																										
25. A0	26. A1	27. A2	28. A3																																																										
29. A4	30. A5	31. A6	32. A7																																																										
33. A8	34. A9	35. A10/AP	36. A11																																																										
37. A12/BC#	38. DQ[0]	39. DQ[1]	40. DQ[2]																																																										
41. DQ[3]	42. DQ[4]	43. DQ[5]	44. DQ[6]																																																										
45. DQ[7]																																																													
OptionalMapping/StandardTerminalName																																																													
1. A13	2. A13	3. A13																																																											

4.7.3.10.6.5. DDR3-x8 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x8-DualDie																																																								
diagram	<pre> classDiagram class DDR3-x8-DualDie { <<DDR3-x8-DualDie-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR3-x8-DualDieStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<DDR3-x8-DualDieMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DDR3-x8-DualDieOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<DDR3-x8-DualDieMandatoryStandardTerminalNameType>> } DDR3-x8-DualDie "1..oo" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..3" --> MandatoryMapping StandardTerminalNameAssignment "0..3" --> OptionalMapping MandatoryMapping "1..3" --> StandardTerminalName OptionalMapping "0..3" --> StandardTerminalName </pre>																																																								
type	DDR3-x8-DualDie-InterfaceFunctionType , DDR3-x8-DualDieStandardTerminalNameAssignmentType , DDR3-x8-DualDieMandatoryStandardTerminalMappingType , DDR3-x8-DualDieMandatoryStandardTerminalNameType , DDR3-x8-DualDieOptionalStandardTerminalMappingType , DDR3-x8-DualDieOptionalStandardTerminalNameType .																																																								
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS</td><td>7. DQS#</td><td>8. VREFCA</td></tr> <tr><td>9. VREFDQ</td><td>10. DM/TDQS</td><td>11. NU/TDQS#</td><td>12. CK</td></tr> <tr><td>13. CK#</td><td>14. CKE0</td><td>15. CKE1</td><td>16. ZQ0</td></tr> <tr><td>17. ZQ1</td><td>18. RESET#</td><td>19. ODT0</td><td>20. ODT1</td></tr> <tr><td>21. CS0#</td><td>22. CS1#</td><td>23. WE#</td><td>24. RAS#</td></tr> <tr><td>25. CAS#</td><td>26. BA0</td><td>27. BA1</td><td>28. BA2</td></tr> <tr><td>29. A0</td><td>30. A1</td><td>31. A2</td><td>32. A3</td></tr> <tr><td>33. A4</td><td>34. A5</td><td>35. A6</td><td>36. A7</td></tr> <tr><td>37. A8</td><td>38. A9</td><td>39. A10/AP</td><td>40. A11</td></tr> <tr><td>41. A12/BC#</td><td>42. DQ[0]</td><td>43. DQ[1]</td><td>44. DQ[2]</td></tr> <tr><td>45. DQ[3]</td><td>46. DQ[4]</td><td>47. DQ[5]</td><td>48. DQ[6]</td></tr> <tr><td>49. DQ[7]</td><td></td><td></td><td></td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. A13</td><td>2. A13</td><td>3. A13</td><td></td></tr> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS	7. DQS#	8. VREFCA	9. VREFDQ	10. DM/TDQS	11. NU/TDQS#	12. CK	13. CK#	14. CKE0	15. CKE1	16. ZQ0	17. ZQ1	18. RESET#	19. ODT0	20. ODT1	21. CS0#	22. CS1#	23. WE#	24. RAS#	25. CAS#	26. BA0	27. BA1	28. BA2	29. A0	30. A1	31. A2	32. A3	33. A4	34. A5	35. A6	36. A7	37. A8	38. A9	39. A10/AP	40. A11	41. A12/BC#	42. DQ[0]	43. DQ[1]	44. DQ[2]	45. DQ[3]	46. DQ[4]	47. DQ[5]	48. DQ[6]	49. DQ[7]				1. A13	2. A13	3. A13	
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																						
5. VSS	6. DQS	7. DQS#	8. VREFCA																																																						
9. VREFDQ	10. DM/TDQS	11. NU/TDQS#	12. CK																																																						
13. CK#	14. CKE0	15. CKE1	16. ZQ0																																																						
17. ZQ1	18. RESET#	19. ODT0	20. ODT1																																																						
21. CS0#	22. CS1#	23. WE#	24. RAS#																																																						
25. CAS#	26. BA0	27. BA1	28. BA2																																																						
29. A0	30. A1	31. A2	32. A3																																																						
33. A4	34. A5	35. A6	36. A7																																																						
37. A8	38. A9	39. A10/AP	40. A11																																																						
41. A12/BC#	42. DQ[0]	43. DQ[1]	44. DQ[2]																																																						
45. DQ[3]	46. DQ[4]	47. DQ[5]	48. DQ[6]																																																						
49. DQ[7]																																																									
1. A13	2. A13	3. A13																																																							

4.7.3.10.6.6. DDR3-x8 – Quad Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x8-QuadDie																																																																				
diagram	<p>The diagram illustrates the class structure for DDR3-x8-QuadDie-InterfaceFunctionType. It features a main class, DDR3-x8-QuadDie-InterfaceFunctionType, which contains two associations: one to StandardTerminalNameAssignment (multiplicity 1..oo) and another to constraints (multiplicity 0..3). The StandardTerminalNameAssignment association leads to a class named StandardTerminalNameAssignment, which further connects to MandatoryMapping (multiplicity 1..oo) and OptionalMapping (multiplicity 0..3). Both MandatoryMapping and OptionalMapping classes connect to StandardTerminalName (multiplicity 1..1).</p> <pre> classDiagram class DDR3_x8_QuadDie_InterfaceFunctionType { <<DDR3-x8-QuadDie-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR3-x8-QuadDieStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<DDR3-x8-QuadDieMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DDR3-x8-QuadDieOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<DDR3-x8-QuadDieMandatoryStandardTerminalNameType>> } DDR3_x8_QuadDie_InterfaceFunctionType "1..oo" -- "StandardTerminalNameAssignment" DDR3_x8_QuadDie_InterfaceFunctionType "0..3" -- "constraints" StandardTerminalNameAssignment "1..oo" -- "MandatoryMapping" StandardTerminalNameAssignment "0..3" -- "OptionalMapping" MandatoryMapping "1..oo" -- "StandardTerminalName" OptionalMapping "0..3" -- "StandardTerminalName" </pre>																																																																				
type	DDR3-x8-QuadDie-InterfaceFunctionType , DDR3-x8-QuadDieStandardTerminalNameAssignmentType , DDR3-x8-QuadDieMandatoryStandardTerminalMappingType , DDR3-x8-QuadDieMandatoryStandardTerminalNameType , DDR3-x8-QuadDieOptionalStandardTerminalMappingType , DDR3-x8-QuadDieOptionalStandardTerminalNameType .																																																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS</td><td>7. DQS#</td><td>8. VREFCA</td></tr> <tr><td>9. VREFDQ</td><td>10. DM/TDQS</td><td>11. NU/TDQS#</td><td>12. CK</td></tr> <tr><td>13. CK#</td><td>14. CKE0</td><td>15. CKE1</td><td>16. ZQ0</td></tr> <tr><td>17. ZQ1</td><td>18. ZQ2</td><td>19. ZQ3</td><td>20. RESET#</td></tr> <tr><td>21. ODT0</td><td>22. ODT1</td><td>23. CS0#</td><td>24. CS1#</td></tr> <tr><td>25. CS2#</td><td>26. CS3#</td><td>27. WE#</td><td>28. RAS#</td></tr> <tr><td>29. CAS#</td><td>30. BA0</td><td>31. BA1</td><td>32. BA2</td></tr> <tr><td>33. A0</td><td>34. A1</td><td>35. A2</td><td>36. A3</td></tr> <tr><td>37. A4</td><td>38. A5</td><td>39. A6</td><td>40. A7</td></tr> <tr><td>41. A8</td><td>42. A9</td><td>43. A10/AP</td><td>44. A11</td></tr> <tr><td>45. A12/BC#</td><td>46. DQ[0]</td><td>47. DQ[1]</td><td>48. DQ[2]</td></tr> <tr><td>49. DQ[3]</td><td>50. DQ[4]</td><td>51. DQ[5]</td><td>52. DQ[6]</td></tr> <tr><td>53. DQ[7]</td><td></td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. A13</td><td>2. A13</td><td>3. A13</td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS	7. DQS#	8. VREFCA	9. VREFDQ	10. DM/TDQS	11. NU/TDQS#	12. CK	13. CK#	14. CKE0	15. CKE1	16. ZQ0	17. ZQ1	18. ZQ2	19. ZQ3	20. RESET#	21. ODT0	22. ODT1	23. CS0#	24. CS1#	25. CS2#	26. CS3#	27. WE#	28. RAS#	29. CAS#	30. BA0	31. BA1	32. BA2	33. A0	34. A1	35. A2	36. A3	37. A4	38. A5	39. A6	40. A7	41. A8	42. A9	43. A10/AP	44. A11	45. A12/BC#	46. DQ[0]	47. DQ[1]	48. DQ[2]	49. DQ[3]	50. DQ[4]	51. DQ[5]	52. DQ[6]	53. DQ[7]				OptionalMapping/StandardTerminalName				1. A13	2. A13	3. A13	
MandatoryMapping/StandardTerminalName																																																																					
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																																		
5. VSS	6. DQS	7. DQS#	8. VREFCA																																																																		
9. VREFDQ	10. DM/TDQS	11. NU/TDQS#	12. CK																																																																		
13. CK#	14. CKE0	15. CKE1	16. ZQ0																																																																		
17. ZQ1	18. ZQ2	19. ZQ3	20. RESET#																																																																		
21. ODT0	22. ODT1	23. CS0#	24. CS1#																																																																		
25. CS2#	26. CS3#	27. WE#	28. RAS#																																																																		
29. CAS#	30. BA0	31. BA1	32. BA2																																																																		
33. A0	34. A1	35. A2	36. A3																																																																		
37. A4	38. A5	39. A6	40. A7																																																																		
41. A8	42. A9	43. A10/AP	44. A11																																																																		
45. A12/BC#	46. DQ[0]	47. DQ[1]	48. DQ[2]																																																																		
49. DQ[3]	50. DQ[4]	51. DQ[5]	52. DQ[6]																																																																		
53. DQ[7]																																																																					
OptionalMapping/StandardTerminalName																																																																					
1. A13	2. A13	3. A13																																																																			

4.7.3.10.6.7. DDR3-x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x16																																																																				
diagram	<pre> classDiagram DDR3-x16 "type DDR3-x16-InterfaceFunctionType" --> StandardTerminalNameAssignment "type DDR3-x16StandardTerminalNameAssignmentType" StandardTerminalNameAssignment --> MandatoryMapping "type DDR3-x16MandatoryStandardTerminalMappingType" StandardTerminalNameAssignment --> OptionalMapping "type DDR3-x16OptionalStandardTerminalMappingType" MandatoryMapping --> StandardTerminalName "type DDR3-x16StandardTerminalName" OptionalMapping --> StandardTerminalName "type DDR3-x16StandardTerminalName" constraints </pre>																																																																				
type	DDR3-x16-InterfaceFunctionType , DDR3-x16StandardTerminalNameAssignmentType , DDR3-x16MandatoryStandardTerminalMappingType , DDR3-x16MandatoryStandardTerminalNameType , DDR3-x16OptionalStandardTerminalMappingType , DDR3-x16OptionalStandardTerminalNameType .																																																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQSU</td><td>7. DQSU#</td><td>8. DQSL</td></tr> <tr><td>9. DQSL#</td><td>10. VREFCA</td><td>11. VREFDQ</td><td>12. DML</td></tr> <tr><td>13. DMU</td><td>14. CK</td><td>15. CK#</td><td>16. CKE</td></tr> <tr><td>17. ZQ</td><td>18. RESET#</td><td>19. ODT</td><td>20. CS#</td></tr> <tr><td>21. WE#</td><td>22. RAS#</td><td>23. CAS#</td><td>24. BA0</td></tr> <tr><td>25. BA1</td><td>26. BA2</td><td>27. A0</td><td>28. A1</td></tr> <tr><td>29. A2</td><td>30. A3</td><td>31. A4</td><td>32. A5</td></tr> <tr><td>33. A6</td><td>34. A7</td><td>35. A8</td><td>36. A9</td></tr> <tr><td>37. A10/AP</td><td>38. A11</td><td>39. A12/BC#</td><td>40. DQU[0]</td></tr> <tr><td>41. DQU[1]</td><td>42. DQU[2]</td><td>43. DQU[3]</td><td>44. DQU[4]</td></tr> <tr><td>45. DQU[5]</td><td>46. DQU[6]</td><td>47. DQU[7]</td><td>48. DQL[0]</td></tr> <tr><td>49. DQL[1]</td><td>50. DQL[2]</td><td>51. DQL[3]</td><td>52. DQL[4]</td></tr> <tr><td>53. DQL[5]</td><td>54. DQL[6]</td><td>55. DQL[7]</td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. A13</td><td>2. A13</td><td>3. A13</td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQSU	7. DQSU#	8. DQSL	9. DQSL#	10. VREFCA	11. VREFDQ	12. DML	13. DMU	14. CK	15. CK#	16. CKE	17. ZQ	18. RESET#	19. ODT	20. CS#	21. WE#	22. RAS#	23. CAS#	24. BA0	25. BA1	26. BA2	27. A0	28. A1	29. A2	30. A3	31. A4	32. A5	33. A6	34. A7	35. A8	36. A9	37. A10/AP	38. A11	39. A12/BC#	40. DQU[0]	41. DQU[1]	42. DQU[2]	43. DQU[3]	44. DQU[4]	45. DQU[5]	46. DQU[6]	47. DQU[7]	48. DQL[0]	49. DQL[1]	50. DQL[2]	51. DQL[3]	52. DQL[4]	53. DQL[5]	54. DQL[6]	55. DQL[7]		OptionalMapping/StandardTerminalName				1. A13	2. A13	3. A13	
MandatoryMapping/StandardTerminalName																																																																					
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																																		
5. VSS	6. DQSU	7. DQSU#	8. DQSL																																																																		
9. DQSL#	10. VREFCA	11. VREFDQ	12. DML																																																																		
13. DMU	14. CK	15. CK#	16. CKE																																																																		
17. ZQ	18. RESET#	19. ODT	20. CS#																																																																		
21. WE#	22. RAS#	23. CAS#	24. BA0																																																																		
25. BA1	26. BA2	27. A0	28. A1																																																																		
29. A2	30. A3	31. A4	32. A5																																																																		
33. A6	34. A7	35. A8	36. A9																																																																		
37. A10/AP	38. A11	39. A12/BC#	40. DQU[0]																																																																		
41. DQU[1]	42. DQU[2]	43. DQU[3]	44. DQU[4]																																																																		
45. DQU[5]	46. DQU[6]	47. DQU[7]	48. DQL[0]																																																																		
49. DQL[1]	50. DQL[2]	51. DQL[3]	52. DQL[4]																																																																		
53. DQL[5]	54. DQL[6]	55. DQL[7]																																																																			
OptionalMapping/StandardTerminalName																																																																					
1. A13	2. A13	3. A13																																																																			

4.7.3.10.6.8. DDR3-x16 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x16-DualDie																																																																								
diagram	<p>The diagram shows the class DDR3-x16-DualDie-InterfaceFunctionType with two associations:</p> <ul style="list-style-type: none"> An association from DDR3-x16-DualDie to StandardTerminalNameAssignment with multiplicity 1..00. An association from StandardTerminalNameAssignment to MandatoryMapping with multiplicity 1..00, and to OptionalMapping with multiplicity 0..3. <p>MandatoryMapping has an association to StandardTerminalName with multiplicity 1..1. OptionalMapping also has an association to StandardTerminalName with multiplicity 0..3.</p> <p>There is a constraint labeled constraints associated with the StandardTerminalNameAssignment class.</p>																																																																								
type	DDR3-x16–DualDie-InterfaceFunctionType , DDR3-x16-DualDieStandardTerminalNameAssignmentType , DDR3-x16-DualDieMandatoryStandardTerminalMappingType , DDR3-x16-DualDieMandatoryStandardTerminalNameType , DDR3-x16-DualDieOptionalStandardTerminalMappingType , DDR3-x16-DualDieOptionalStandardTerminalNameType .																																																																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQSU</td><td>7. DQSU#</td><td>8. DQL</td></tr> <tr><td>9. DQL#</td><td>10. VREFCA</td><td>11. VREFDQ</td><td>12. DML</td></tr> <tr><td>13. DMU</td><td>14. CK</td><td>15. CK#</td><td>16. CKE0</td></tr> <tr><td>17. CKE1</td><td>18. ZQ0</td><td>19. ZQ1</td><td>20. RESET#</td></tr> <tr><td>21. ODT0</td><td>22. ODT1</td><td>23. CS0#</td><td>24. CS1#</td></tr> <tr><td>25. WE#</td><td>26. RAS#</td><td>27. CAS#</td><td>28. BA0</td></tr> <tr><td>29. BA1</td><td>30. BA2</td><td>31. A0</td><td>32. A1</td></tr> <tr><td>33. A2</td><td>34. A3</td><td>35. A4</td><td>36. A5</td></tr> <tr><td>37. A6</td><td>38. A7</td><td>39. A8</td><td>40. A9</td></tr> <tr><td>41. A10/AP</td><td>42. A11</td><td>43. A12/BC#</td><td>44. DQU[0]</td></tr> <tr><td>45. DQU[1]</td><td>46. DQU[2]</td><td>47. DQU[3]</td><td>48. DQU[4]</td></tr> <tr><td>49. DQU[5]</td><td>50. DQU[6]</td><td>51. DQU[7]</td><td>52. DQL[0]</td></tr> <tr><td>53. DQL[1]</td><td>54. DQL[2]</td><td>55. DQL[3]</td><td>56. DQL[4]</td></tr> <tr><td>57. DQL[5]</td><td>58. DQL[6]</td><td>59. DQL[7]</td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. A13</td><td>2. A13</td><td>3. A13</td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQSU	7. DQSU#	8. DQL	9. DQL#	10. VREFCA	11. VREFDQ	12. DML	13. DMU	14. CK	15. CK#	16. CKE0	17. CKE1	18. ZQ0	19. ZQ1	20. RESET#	21. ODT0	22. ODT1	23. CS0#	24. CS1#	25. WE#	26. RAS#	27. CAS#	28. BA0	29. BA1	30. BA2	31. A0	32. A1	33. A2	34. A3	35. A4	36. A5	37. A6	38. A7	39. A8	40. A9	41. A10/AP	42. A11	43. A12/BC#	44. DQU[0]	45. DQU[1]	46. DQU[2]	47. DQU[3]	48. DQU[4]	49. DQU[5]	50. DQU[6]	51. DQU[7]	52. DQL[0]	53. DQL[1]	54. DQL[2]	55. DQL[3]	56. DQL[4]	57. DQL[5]	58. DQL[6]	59. DQL[7]		OptionalMapping/StandardTerminalName				1. A13	2. A13	3. A13	
MandatoryMapping/StandardTerminalName																																																																									
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																																						
5. VSS	6. DQSU	7. DQSU#	8. DQL																																																																						
9. DQL#	10. VREFCA	11. VREFDQ	12. DML																																																																						
13. DMU	14. CK	15. CK#	16. CKE0																																																																						
17. CKE1	18. ZQ0	19. ZQ1	20. RESET#																																																																						
21. ODT0	22. ODT1	23. CS0#	24. CS1#																																																																						
25. WE#	26. RAS#	27. CAS#	28. BA0																																																																						
29. BA1	30. BA2	31. A0	32. A1																																																																						
33. A2	34. A3	35. A4	36. A5																																																																						
37. A6	38. A7	39. A8	40. A9																																																																						
41. A10/AP	42. A11	43. A12/BC#	44. DQU[0]																																																																						
45. DQU[1]	46. DQU[2]	47. DQU[3]	48. DQU[4]																																																																						
49. DQU[5]	50. DQU[6]	51. DQU[7]	52. DQL[0]																																																																						
53. DQL[1]	54. DQL[2]	55. DQL[3]	56. DQL[4]																																																																						
57. DQL[5]	58. DQL[6]	59. DQL[7]																																																																							
OptionalMapping/StandardTerminalName																																																																									
1. A13	2. A13	3. A13																																																																							

4.7.3.10.6.9. DDR3-x16 – Quad Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x16-QuadDie																																																																				
diagram																																																																					
type	DDR3-x16-QuadDieInterfaceFunctionType , DDR3-x16-QuadDieStandardTerminalNameAssignmentType , DDR3-x16-QuadDieMandatoryStandardTerminalMappingType , DDR3-x16-QuadDieMandatoryStandardTerminalNameType , DDR3-x16-QuadDieOptionalStandardTerminalMappingType , DDR3-x16-QuadDieOptionalStandardTerminalNameType .																																																																				
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQSU</td><td>7. DQSU#</td><td>8. DQL</td></tr> <tr><td>9. DQL#</td><td>10. VREFCA</td><td>11. VREFDQ</td><td>12. DML</td></tr> <tr><td>13. DMU</td><td>14. CK</td><td>15. CK#</td><td>16. CKE0</td></tr> <tr><td>17. CKE1</td><td>18. ZQ0</td><td>19. ZQ1</td><td>20. ZQ2</td></tr> <tr><td>21. ZQ3</td><td>22. RESET#</td><td>23. ODT0</td><td>24. ODT1</td></tr> <tr><td>25. CS0#</td><td>26. CS1#</td><td>27. CS2#</td><td>28. CS3#</td></tr> <tr><td>29. WE#</td><td>30. RAS#</td><td>31. CAS#</td><td>32. BA0</td></tr> <tr><td>33. BA1</td><td>34. BA2</td><td>35. A0</td><td>36. A1</td></tr> <tr><td>37. A2</td><td>38. A3</td><td>39. A4</td><td>40. A5</td></tr> <tr><td>41. A6</td><td>42. A7</td><td>43. A8</td><td>44. A9</td></tr> <tr><td>45. A10/AP</td><td>46. A11</td><td>47. A12/BC#</td><td>48. DQU[0]</td></tr> <tr><td>49. DQU[1]</td><td>50. DQU[2]</td><td>51. DQU[3]</td><td>52. DQU[4]</td></tr> <tr><td>53. DQU[5]</td><td>54. DQU[6]</td><td>55. DQU[7]</td><td>56. DQL[0]</td></tr> <tr><td>57. DQL[1]</td><td>58. DQL[2]</td><td>59. DQL[3]</td><td>60. DQL[4]</td></tr> <tr><td>61. DQL[5]</td><td>62. DQL[6]</td><td>63. DQL[7]</td><td></td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. A13</td><td>2. A14</td><td>3. A15</td><td></td></tr> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQSU	7. DQSU#	8. DQL	9. DQL#	10. VREFCA	11. VREFDQ	12. DML	13. DMU	14. CK	15. CK#	16. CKE0	17. CKE1	18. ZQ0	19. ZQ1	20. ZQ2	21. ZQ3	22. RESET#	23. ODT0	24. ODT1	25. CS0#	26. CS1#	27. CS2#	28. CS3#	29. WE#	30. RAS#	31. CAS#	32. BA0	33. BA1	34. BA2	35. A0	36. A1	37. A2	38. A3	39. A4	40. A5	41. A6	42. A7	43. A8	44. A9	45. A10/AP	46. A11	47. A12/BC#	48. DQU[0]	49. DQU[1]	50. DQU[2]	51. DQU[3]	52. DQU[4]	53. DQU[5]	54. DQU[6]	55. DQU[7]	56. DQL[0]	57. DQL[1]	58. DQL[2]	59. DQL[3]	60. DQL[4]	61. DQL[5]	62. DQL[6]	63. DQL[7]		1. A13	2. A14	3. A15	
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																																		
5. VSS	6. DQSU	7. DQSU#	8. DQL																																																																		
9. DQL#	10. VREFCA	11. VREFDQ	12. DML																																																																		
13. DMU	14. CK	15. CK#	16. CKE0																																																																		
17. CKE1	18. ZQ0	19. ZQ1	20. ZQ2																																																																		
21. ZQ3	22. RESET#	23. ODT0	24. ODT1																																																																		
25. CS0#	26. CS1#	27. CS2#	28. CS3#																																																																		
29. WE#	30. RAS#	31. CAS#	32. BA0																																																																		
33. BA1	34. BA2	35. A0	36. A1																																																																		
37. A2	38. A3	39. A4	40. A5																																																																		
41. A6	42. A7	43. A8	44. A9																																																																		
45. A10/AP	46. A11	47. A12/BC#	48. DQU[0]																																																																		
49. DQU[1]	50. DQU[2]	51. DQU[3]	52. DQU[4]																																																																		
53. DQU[5]	54. DQU[6]	55. DQU[7]	56. DQL[0]																																																																		
57. DQL[1]	58. DQL[2]	59. DQL[3]	60. DQL[4]																																																																		
61. DQL[5]	62. DQL[6]	63. DQL[7]																																																																			
1. A13	2. A14	3. A15																																																																			

4.7.3.10.6.10. DDR3 Controller

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3-Controller																																																																												
diagram	<pre> classDiagram class DDR3_Controller { <<DDR3-Controller-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR3-ControllerStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<DDR3-ControllerMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DDR3-ControllerOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<DDR3-ControllerStandardTerminalNameType>> } DDR3_Controller "1..oo" -- "44" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..oo" -- "0..111" MandatoryMapping StandardTerminalNameAssignment "1..oo" -- "0..111" OptionalMapping MandatoryMapping "*" StandardTerminalName </pre>																																																																												
type	DDR3-Controller-InterfaceFunctionType , DDR3-ControllerStandardTerminalNameAssignmentType , DDR3-ControllerMandatoryStandardTerminalMappingType , DDR3-ControllerMandatoryStandardTerminalNameType , DDR3-ControllerOptionalStandardTerminalMappingType , DDR3-ControllerOptionalStandardTerminalNameType .																																																																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. A0</td><td>2. A1</td><td>3. A2</td><td>4. A3</td></tr> <tr> <td>5. A4</td><td>6. A5</td><td>7. A6</td><td>8. A7</td></tr> <tr> <td>9. A8</td><td>10. A9</td><td>11. A10/AP</td><td>12. A11</td></tr> <tr> <td>13. A12/BC#</td><td>14. BA0</td><td>15. BA1</td><td>16. BA2</td></tr> <tr> <td>17. CAS#</td><td>18. CK0</td><td>19. CK0#</td><td>20. CKE0</td></tr> <tr> <td>21. CS0#</td><td>22. DM0</td><td>23. DQ[0]</td><td>24. DQ[1]</td></tr> <tr> <td>25. DQ[2]</td><td>26. DQ[3]</td><td>27. DQ[4]</td><td>28. DQ[5]</td></tr> <tr> <td>29. DQ[6]</td><td>30. DQ[7]</td><td>31. DQ[8]</td><td>32. DQ[9]</td></tr> <tr> <td>33. DQ[10]</td><td>34. DQ[11]</td><td>35. DQ[12]</td><td>36. DQ[13]</td></tr> <tr> <td>37. DQ[14]</td><td>38. DQ[15]</td><td>39. DQS0</td><td>40. DQS0#</td></tr> <tr> <td>41. ODT0</td><td>42. RAS#</td><td>43. RESET#</td><td>44. WE#</td></tr> <tr> <td>45. A13</td><td>46. A14</td><td>47. A15</td><td>48. CB0</td></tr> <tr> <td>49. CB1</td><td>50. CB2</td><td>51. CB3</td><td>52. CB4</td></tr> <tr> <td>53. CB5</td><td>54. CB6</td><td>55. CB7</td><td>56. CK1</td></tr> <tr> <td>57. CK1#</td><td>58. CKE1</td><td>59. CS1#</td><td>60. CS2#</td></tr> <tr> <td>61. CS3#</td><td>62. DM1</td><td>63. DM2</td><td>64. DM3</td></tr> <tr> <td>65. DM4</td><td>66. DM5</td><td>67. DM6</td><td>68. DM7</td></tr> <tr> <td>69. DM8</td><td>70. DQ[16]</td><td>71. DQ[17]</td><td>72. DQ[18]</td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. A0	2. A1	3. A2	4. A3	5. A4	6. A5	7. A6	8. A7	9. A8	10. A9	11. A10/AP	12. A11	13. A12/BC#	14. BA0	15. BA1	16. BA2	17. CAS#	18. CK0	19. CK0#	20. CKE0	21. CS0#	22. DM0	23. DQ[0]	24. DQ[1]	25. DQ[2]	26. DQ[3]	27. DQ[4]	28. DQ[5]	29. DQ[6]	30. DQ[7]	31. DQ[8]	32. DQ[9]	33. DQ[10]	34. DQ[11]	35. DQ[12]	36. DQ[13]	37. DQ[14]	38. DQ[15]	39. DQS0	40. DQS0#	41. ODT0	42. RAS#	43. RESET#	44. WE#	45. A13	46. A14	47. A15	48. CB0	49. CB1	50. CB2	51. CB3	52. CB4	53. CB5	54. CB6	55. CB7	56. CK1	57. CK1#	58. CKE1	59. CS1#	60. CS2#	61. CS3#	62. DM1	63. DM2	64. DM3	65. DM4	66. DM5	67. DM6	68. DM7	69. DM8	70. DQ[16]	71. DQ[17]	72. DQ[18]
MandatoryMapping/StandardTerminalName																																																																													
1. A0	2. A1	3. A2	4. A3																																																																										
5. A4	6. A5	7. A6	8. A7																																																																										
9. A8	10. A9	11. A10/AP	12. A11																																																																										
13. A12/BC#	14. BA0	15. BA1	16. BA2																																																																										
17. CAS#	18. CK0	19. CK0#	20. CKE0																																																																										
21. CS0#	22. DM0	23. DQ[0]	24. DQ[1]																																																																										
25. DQ[2]	26. DQ[3]	27. DQ[4]	28. DQ[5]																																																																										
29. DQ[6]	30. DQ[7]	31. DQ[8]	32. DQ[9]																																																																										
33. DQ[10]	34. DQ[11]	35. DQ[12]	36. DQ[13]																																																																										
37. DQ[14]	38. DQ[15]	39. DQS0	40. DQS0#																																																																										
41. ODT0	42. RAS#	43. RESET#	44. WE#																																																																										
45. A13	46. A14	47. A15	48. CB0																																																																										
49. CB1	50. CB2	51. CB3	52. CB4																																																																										
53. CB5	54. CB6	55. CB7	56. CK1																																																																										
57. CK1#	58. CKE1	59. CS1#	60. CS2#																																																																										
61. CS3#	62. DM1	63. DM2	64. DM3																																																																										
65. DM4	66. DM5	67. DM6	68. DM7																																																																										
69. DM8	70. DQ[16]	71. DQ[17]	72. DQ[18]																																																																										

4.7.3.10.6.10 DDR3 Controller (cont'd)

list of enumerate values (cont.)	OptionalMapping/StandardTerminalName			
73. DQ[19]	74. DQ[20]	75. DQ[21]	76. DQ[22]	
77. DQ[23]	78. DQ[24]	79. DQ[25]	80. DQ[26]	
81. DQ[27]	82. DQ[28]	83. DQ[29]	84. DQ[30]	
85. DQ[31]	86. DQ[32]	87. DQ[33]	88. DQ[34]	
89. DQ[35]	90. DQ[36]	91. DQ[37]	92. DQ[38]	
93. DQ[39]	94. DQ[40]	95. DQ[41]	96. DQ[42]	
97. DQ[43]	98. DQ[44]	99. DQ[45]	100. DQ[46]	
101. DQ[47]	102. DQ[48]	103. DQ[49]	104. DQ[50]	
105. DQ[51]	106. DQ[52]	107. DQ[53]	108. DQ[54]	
109. DQ[55]	110. DQ[56]	111. DQ[57]	112. DQ[58]	
113. DQ[59]	114. DQ[60]	115. DQ[61]	116. DQ[62]	
117. DQ[63]	118. DQS1	119. DQS1#	120. DQS2	
121. DQS2#	122. DQS3	123. DQS3#	124. DQS4	
125. DQS4#	126. DQS5	127. DQS5#	128. DQS6	
129. DQS6#	130. DQS7	131. DQS7#	132. DQS8	
133. DQS8#	134. DQS9	135. DQS9#	136. DQS10	
137. DQS10#	138. DQS11	139. DQS11#	140. DQS12	
141. DQS12#	142. DQS13	143. DQS13#	144. DQS14	
145. DQS14#	146. DQS15	147. DQS15#	148. DQS16	
149. DQS16#	150. DQS17	151. DQS17#	152. ODT1	
153. VREF	154. VREFCA	155. VREFDQ		

4.7.3.10.7. DDR4 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4
diagram	<pre> graph LR DDR4[DDR4 type: DDR4-InterfaceFunctionType] --- DDR4IF[DDR4-InterfaceFunctionType] DDR4IF --- DDR4x4[DDR4-x4 type: DDR4-x4-InterfaceFunctionType] DDR4IF --- DDR4x4DD[DDR4-x4-DualDie type: DDR4-x4-DualDie-InterfaceFunctionType] DDR4IF --- DDR4x8[DDR4-x8 type: DDR4-x8-InterfaceFunctionType] DDR4IF --- DDR4x8DD[DDR4-x8-DualDie type: DDR4-x8-DualDie-InterfaceFunctionType] DDR4IF --- DDR4x16[DDR4-x16 type: DDR4-x16-InterfaceFunctionType] DDR4IF --- DDR4x16DD[DDR4-x16-DualDie type: DDR4-x16-DualDie-InterfaceFunctionType] DDR4IF --- DDR4x32[DDR4-x32 type: DDR4-x32-InterfaceFunctionType] DDR4IF --- DDR4x72[DDR4-x72 type: DDR4-x72-InterfaceFunctionType] DDR4IF --- DDR4DB02[DDR4DB02 type: DDR4DB02-InterfaceFunctionType] DDR4IF --- DDR4RCD02[DDR4RCD02 type: DDR4RCD02-InterfaceFunctionType] DDR4IF --- DDR4NVDIMM[DDR4-NVDIMM-N type: DDR4-NVDIMM-N-InterfaceFunctionType] DDR4IF --- LPDDR4SC[LPDDR4-SingleChannel type: LPDDR4-SingleChannelInterfaceFunctionType] DDR4IF --- LPDDR4DC[LPDDR4-DualChannel type: LPDDR4-DualChannelInterfaceFunctionType] </pre>
type	DDR4-InterfaceFunctionType , DDR4-x4-InterfaceFunctionType , DDR4-x4-DualDie-InterfaceFunctionType , DDR4-x8-InterfaceFunctionType , DDR4-x8-DualDie-InterfaceFunctionType , DDR4-x16-InterfaceFunctionType , DDR4-x16-DualDie-InterfaceFunctionType , DDR4-x32-InterfaceFunctionType , DDR4-x72-InterfaceFunctionType , DDR4DB02-InterfaceFunctionType , DDR4RCD02-InterfaceFunctionType , DDR4-NVDIMM-N-InterfaceFunctionType , DDR4-x72-InterfaceFunctionType , LPDDR4-DualChannelInterfaceFunctionType ,

4.7.3.10.7.1. DDR4-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x4																																																								
diagram	<pre> classDiagram class DDR4-x4 { <<InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<DDR4-x4MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DDR4-x4OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<StandardTerminalNameType>> } DDR4-x4 "1..40" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..40" --> MandatoryMapping StandardTerminalNameAssignment "1..40" --> OptionalMapping MandatoryMapping "1..43" --> StandardTerminalName OptionalMapping "0..2" --> StandardTerminalName </pre>																																																								
type	DDR4-x4-InterfaceFunctionType , DDR3-x4StandardTerminalNameAssignmentType , DDR4-x4MandatoryStandardTerminalMappingType , DDR4-x4MandatoryStandardTerminalNameType , DDR4-x4OptionalStandardTerminalMappingType , DDR4-x4OptionalStandardTerminalNameType .																																																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS_c</td><td>7. DQS_t</td><td>8. ODT</td></tr> <tr><td>9. CS_n</td><td>10. CK_t</td><td>11. CK_c</td><td>12. ZQ</td></tr> <tr><td>13. ALERT_n</td><td>14. ACT_n</td><td>15. CKE</td><td>16. PAR</td></tr> <tr><td>17. RESET_n</td><td>18. VREFCA</td><td>19. BA0</td><td>20. BA1</td></tr> <tr><td>21. BG0</td><td>22. BG1</td><td>23. A0</td><td>24. A1</td></tr> <tr><td>25. A2</td><td>26. A3</td><td>27. A4</td><td>28. A5</td></tr> <tr><td>29. A6</td><td>30. A7</td><td>31. A8</td><td>32. A9</td></tr> <tr><td>33. A10/AP</td><td>34. A11</td><td>35. A12/BC_n</td><td>36. A13</td></tr> <tr><td>37. WE_n/A14</td><td>38. CAS_n/A15</td><td>39. RAS_n/A16</td><td>40. DQ[0]</td></tr> <tr><td>41. DQ[1]</td><td>42. DQ[2]</td><td>43. DQ[3]</td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. TEN</td><td>2. A17</td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS_c	7. DQS_t	8. ODT	9. CS_n	10. CK_t	11. CK_c	12. ZQ	13. ALERT_n	14. ACT_n	15. CKE	16. PAR	17. RESET_n	18. VREFCA	19. BA0	20. BA1	21. BG0	22. BG1	23. A0	24. A1	25. A2	26. A3	27. A4	28. A5	29. A6	30. A7	31. A8	32. A9	33. A10/AP	34. A11	35. A12/BC_n	36. A13	37. WE_n/A14	38. CAS_n/A15	39. RAS_n/A16	40. DQ[0]	41. DQ[1]	42. DQ[2]	43. DQ[3]		OptionalMapping/StandardTerminalName				1. TEN	2. A17		
MandatoryMapping/StandardTerminalName																																																									
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																						
5. VSS	6. DQS_c	7. DQS_t	8. ODT																																																						
9. CS_n	10. CK_t	11. CK_c	12. ZQ																																																						
13. ALERT_n	14. ACT_n	15. CKE	16. PAR																																																						
17. RESET_n	18. VREFCA	19. BA0	20. BA1																																																						
21. BG0	22. BG1	23. A0	24. A1																																																						
25. A2	26. A3	27. A4	28. A5																																																						
29. A6	30. A7	31. A8	32. A9																																																						
33. A10/AP	34. A11	35. A12/BC_n	36. A13																																																						
37. WE_n/A14	38. CAS_n/A15	39. RAS_n/A16	40. DQ[0]																																																						
41. DQ[1]	42. DQ[2]	43. DQ[3]																																																							
OptionalMapping/StandardTerminalName																																																									
1. TEN	2. A17																																																								

For more information about the DDR4-x4 Interface, refer to the JEDEC standard JESD79-4B.

4.7.3.10.7.2. DDR4-x4 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x4-DualDie																																																				
diagram	<pre> classDiagram class DDR4-x4-DualDie { <<DDR4-x4-DualDie-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR4-x4-DualDie-StandardTerminalNameAssignmentType>> } class DDR4-x4-DualDie-InterfaceFunctionType { <<DDR4-x4-DualDie-InterfaceFunctionType>> } class MandatoryMapping { <<DDR4-x4-DualDie-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DDR4-x4-DualDie-OptionalStandardTerminalMappingType>> } class StandardTerminalNameAssignmentType { <<DDR4-x4-DualDie-StandardTerminalNameAssignmentType>> } class StandardTerminalName { <<DDR4-x4-DualDie-StandardTerminalName>> } DDR4-x4-DualDie "1..infinity" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..infinity" --> DDR4-x4-DualDie-InterfaceFunctionType DDR4-x4-DualDie-InterfaceFunctionType "1..infinity" --> MandatoryMapping DDR4-x4-DualDie-InterfaceFunctionType "0..2" --> OptionalMapping MandatoryMapping "1..infinity" --> StandardTerminalNameAssignmentType OptionalMapping "0..2" --> StandardTerminalNameAssignmentType StandardTerminalNameAssignmentType "1..infinity" --> StandardTerminalName </pre>																																																				
type	DDR4-x4-DualDie-InterfaceFunctionType , DDR4-x4-DualDieStandardTerminalNameAssignmentType , DDR4-x4-DualDieMandatoryStandardTerminalMappingType , DDR4-x4-DualDieMandatoryStandardTerminalNameType , DDR4-x4-DualDieOptionalStandardTerminalMappingType , DDR4-x4-DualDieOptionalStandardTerminalNameType .																																																				
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS_c</td><td>7. DQS_t</td><td>8. ODT</td></tr> <tr><td>9. CS_n</td><td>10. CK_t</td><td>11. CK_c</td><td>12. ZQ</td></tr> <tr><td>13. ALERT_n</td><td>14. ACT_n</td><td>15. CKE</td><td>16. PAR</td></tr> <tr><td>17. RESET_n</td><td>18. VREFCA</td><td>19. BA0</td><td>20. BA1</td></tr> <tr><td>21. BG0</td><td>22. BG1</td><td>23. CKE1</td><td>24. CS1_n</td></tr> <tr><td>25. ODT1</td><td>26. A0</td><td>27. A1</td><td>28. A2</td></tr> <tr><td>29. A3</td><td>30. A4</td><td>31. A5</td><td>32. A6</td></tr> <tr><td>33. A7</td><td>34. A8</td><td>35. A9</td><td>36. A10/AP</td></tr> <tr><td>37. A11</td><td>38. A12/BC_n</td><td>39. A13</td><td>40. WE_n/A14</td></tr> <tr><td>41. CAS_n/A15</td><td>42. RAS_n/A16</td><td>43. DQ[0]</td><td>44. DQ[1]</td></tr> <tr><td>45. DQ[2]</td><td>46. DQ[3]</td><td></td><td></td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. TEN</td><td>2. A17</td><td></td><td></td></tr> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS_c	7. DQS_t	8. ODT	9. CS_n	10. CK_t	11. CK_c	12. ZQ	13. ALERT_n	14. ACT_n	15. CKE	16. PAR	17. RESET_n	18. VREFCA	19. BA0	20. BA1	21. BG0	22. BG1	23. CKE1	24. CS1_n	25. ODT1	26. A0	27. A1	28. A2	29. A3	30. A4	31. A5	32. A6	33. A7	34. A8	35. A9	36. A10/AP	37. A11	38. A12/BC_n	39. A13	40. WE_n/A14	41. CAS_n/A15	42. RAS_n/A16	43. DQ[0]	44. DQ[1]	45. DQ[2]	46. DQ[3]			1. TEN	2. A17		
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																		
5. VSS	6. DQS_c	7. DQS_t	8. ODT																																																		
9. CS_n	10. CK_t	11. CK_c	12. ZQ																																																		
13. ALERT_n	14. ACT_n	15. CKE	16. PAR																																																		
17. RESET_n	18. VREFCA	19. BA0	20. BA1																																																		
21. BG0	22. BG1	23. CKE1	24. CS1_n																																																		
25. ODT1	26. A0	27. A1	28. A2																																																		
29. A3	30. A4	31. A5	32. A6																																																		
33. A7	34. A8	35. A9	36. A10/AP																																																		
37. A11	38. A12/BC_n	39. A13	40. WE_n/A14																																																		
41. CAS_n/A15	42. RAS_n/A16	43. DQ[0]	44. DQ[1]																																																		
45. DQ[2]	46. DQ[3]																																																				
1. TEN	2. A17																																																				

For more information about the DDR4-x4 Dual Die Interface, refer to the JEDEC standard JESD79-4B.

4.7.3.10.7.3. DDR4-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x8																																																																
diagram	<pre> classDiagram class DDR4_x8 { <<DDR4-x8-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR4-x8StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<DDR4-x8MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<DDR4-x8OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<DDR4-x8StandardTerminalNameType>> } DDR4_x8 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" MandatoryMapping StandardTerminalNameAssignment "*" OptionalMapping MandatoryMapping "49" StandardTerminalName OptionalMapping "49" StandardTerminalName </pre>																																																																
type	DDR4-x8-InterfaceFunctionType , DDR4-x8StandardTerminalNameAssignmentType , DDR4-x8MandatoryStandardTerminalMappingType , DDR4-x8MandatoryStandardTerminalNameType , DDR4-x8OptionalStandardTerminalMappingType , DDR4-x8OptionalStandardTerminalNameType .																																																																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DM_n/DBI_n/TDQS_t</td><td>7. TDQS_c</td><td>8. DQS_c</td></tr> <tr><td>9. DQS_t</td><td>10. ODT</td><td>11. CS_n</td><td>12. CK_t</td></tr> <tr><td>13. CK_c</td><td>14. ZQ</td><td>15. ALERT_n</td><td>16. ACT_n</td></tr> <tr><td>17. CKE</td><td>18. PAR</td><td>19. RESET_n</td><td>20. VREFCA</td></tr> <tr><td>21. BA0</td><td>22. BA1</td><td>23. BG0</td><td>24. BG1</td></tr> <tr><td>25. A0</td><td>26. A1</td><td>27. A2</td><td>28. A3</td></tr> <tr><td>29. A4</td><td>30. A5</td><td>31. A6</td><td>32. A7</td></tr> <tr><td>33. A8</td><td>34. A9</td><td>35. A10/AP</td><td>36. A11</td></tr> <tr><td>37. A12/BC_n</td><td>38. A13</td><td>39. WE_n/A14</td><td>40. CAS_n/A15</td></tr> <tr><td>41. RAS_n/A16</td><td>42. DQ[0]</td><td>43. DQ[1]</td><td>44. DQ[2]</td></tr> <tr><td>45. DQ[3]</td><td>46. DQ[4]</td><td>47. DQ[5]</td><td>48. DQ[6]</td></tr> <tr><td>49. DQ[7]</td><td></td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. TEN</td><td></td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DM_n/DBI_n/TDQS_t	7. TDQS_c	8. DQS_c	9. DQS_t	10. ODT	11. CS_n	12. CK_t	13. CK_c	14. ZQ	15. ALERT_n	16. ACT_n	17. CKE	18. PAR	19. RESET_n	20. VREFCA	21. BA0	22. BA1	23. BG0	24. BG1	25. A0	26. A1	27. A2	28. A3	29. A4	30. A5	31. A6	32. A7	33. A8	34. A9	35. A10/AP	36. A11	37. A12/BC_n	38. A13	39. WE_n/A14	40. CAS_n/A15	41. RAS_n/A16	42. DQ[0]	43. DQ[1]	44. DQ[2]	45. DQ[3]	46. DQ[4]	47. DQ[5]	48. DQ[6]	49. DQ[7]				OptionalMapping/StandardTerminalName				1. TEN			
MandatoryMapping/StandardTerminalName																																																																	
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																														
5. VSS	6. DM_n/DBI_n/TDQS_t	7. TDQS_c	8. DQS_c																																																														
9. DQS_t	10. ODT	11. CS_n	12. CK_t																																																														
13. CK_c	14. ZQ	15. ALERT_n	16. ACT_n																																																														
17. CKE	18. PAR	19. RESET_n	20. VREFCA																																																														
21. BA0	22. BA1	23. BG0	24. BG1																																																														
25. A0	26. A1	27. A2	28. A3																																																														
29. A4	30. A5	31. A6	32. A7																																																														
33. A8	34. A9	35. A10/AP	36. A11																																																														
37. A12/BC_n	38. A13	39. WE_n/A14	40. CAS_n/A15																																																														
41. RAS_n/A16	42. DQ[0]	43. DQ[1]	44. DQ[2]																																																														
45. DQ[3]	46. DQ[4]	47. DQ[5]	48. DQ[6]																																																														
49. DQ[7]																																																																	
OptionalMapping/StandardTerminalName																																																																	
1. TEN																																																																	

For more information about the DDR4-x8 Interface, refer to the JEDEC standard JESD79-4B.

4.7.3.10.7.4. DDR4-x8 Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x8-DualDie																																																								
diagram	<pre> classDiagram class DDR4-x8-DualDie { <<DDR4-x8-DualDie>> <<type DDR4-x8-DualDie-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<StandardTerminalNameAssignment>> <<type DDR4-x8-DualDie-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<MandatoryMapping>> <<type DDR4-x8-DualDie-MandatoryStandardTerminalNameAssignmentType>> } class OptionalMapping { <<OptionalMapping>> <<type DDR4-x8-DualDie-OptionalStandardTerminalNameAssignmentType>> } class StandardTerminalName { <<StandardTerminalName>> <<type DDR4-x8-DualDie-StandardTerminalNameType>> } DDR4-x8-DualDie "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" MandatoryMapping StandardTerminalNameAssignment "*" OptionalMapping MandatoryMapping "1..>" StandardTerminalName OptionalMapping "1..>" StandardTerminalName </pre>																																																								
type	DDR4-x8-DualDie-InterfaceFunctionType , DDR4-x8-DualDieStandardTerminalNameAssignmentType , DDR4-x8-DualDieMandatoryStandardTerminalMappingType , DDR4-x8-DualDieMandatoryStandardTerminalNameType , DDR4-x8-DualDieOptionalStandardTerminalMappingType , DDR4-x8-DualDieOptionalStandardTerminalNameType .																																																								
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DM_n/DBI_n/TDQS_t</td><td>7. TDQS_c</td><td>8. DQS_c</td></tr> <tr><td>9. DQS_t</td><td>10. ODT</td><td>11. CS_n</td><td>12. CK_t</td></tr> <tr><td>13. CK_c</td><td>14. ZQ</td><td>15. ALERT_n</td><td>16. ACT_n</td></tr> <tr><td>17. CKE</td><td>18. PAR</td><td>19. RESET_n</td><td>20. VREFCA</td></tr> <tr><td>21. BA0</td><td>22. BA1</td><td>23. BG0</td><td>24. BG1</td></tr> <tr><td>25. CKE1</td><td>26. CS1_n</td><td>27. ODT1</td><td>28. A0</td></tr> <tr><td>29. A1</td><td>30. A2</td><td>31. A3</td><td>32. A4</td></tr> <tr><td>33. A5</td><td>34. A6</td><td>35. A7</td><td>36. A8</td></tr> <tr><td>37. A9</td><td>38. A10/AP</td><td>39. A11</td><td>40. A12/BC_n</td></tr> <tr><td>41. A13</td><td>42. WE_n/A14</td><td>43. CAS_n/A15</td><td>44. RAS_n/A16</td></tr> <tr><td>45. DQ[0]</td><td>46. DQ[1]</td><td>47. DQ[2]</td><td>48. DQ[3]</td></tr> <tr><td>49. DQ[4]</td><td>50. DQ[5]</td><td>51. DQ[6]</td><td>52. DQ[7]</td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. TEN</td><td></td><td></td><td></td></tr> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DM_n/DBI_n/TDQS_t	7. TDQS_c	8. DQS_c	9. DQS_t	10. ODT	11. CS_n	12. CK_t	13. CK_c	14. ZQ	15. ALERT_n	16. ACT_n	17. CKE	18. PAR	19. RESET_n	20. VREFCA	21. BA0	22. BA1	23. BG0	24. BG1	25. CKE1	26. CS1_n	27. ODT1	28. A0	29. A1	30. A2	31. A3	32. A4	33. A5	34. A6	35. A7	36. A8	37. A9	38. A10/AP	39. A11	40. A12/BC_n	41. A13	42. WE_n/A14	43. CAS_n/A15	44. RAS_n/A16	45. DQ[0]	46. DQ[1]	47. DQ[2]	48. DQ[3]	49. DQ[4]	50. DQ[5]	51. DQ[6]	52. DQ[7]	1. TEN			
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																						
5. VSS	6. DM_n/DBI_n/TDQS_t	7. TDQS_c	8. DQS_c																																																						
9. DQS_t	10. ODT	11. CS_n	12. CK_t																																																						
13. CK_c	14. ZQ	15. ALERT_n	16. ACT_n																																																						
17. CKE	18. PAR	19. RESET_n	20. VREFCA																																																						
21. BA0	22. BA1	23. BG0	24. BG1																																																						
25. CKE1	26. CS1_n	27. ODT1	28. A0																																																						
29. A1	30. A2	31. A3	32. A4																																																						
33. A5	34. A6	35. A7	36. A8																																																						
37. A9	38. A10/AP	39. A11	40. A12/BC_n																																																						
41. A13	42. WE_n/A14	43. CAS_n/A15	44. RAS_n/A16																																																						
45. DQ[0]	46. DQ[1]	47. DQ[2]	48. DQ[3]																																																						
49. DQ[4]	50. DQ[5]	51. DQ[6]	52. DQ[7]																																																						
1. TEN																																																									

For more information about the DDR4-x8 Dual Die Interface, refer to the JEDEC standard JESD79-4B.

4.7.3.10.7.5. DDR4-x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x16																																																												
diagram	<pre> classDiagram class DDR4-x16 { <<DDR4-x16-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR4-x16StandardTerminalNameAssignmentType>> } class Mapping { <<DDR4-x16StandardTerminalMappingType>> } class StandardTerminalName { <<DDR4-x16StandardTerminalNameType>> } DDR4-x16 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>																																																												
type	DDR4-x16-InterfaceFunctionType , DDR4-x16StandardTerminalNameAssignmentType , DDR4-x16StandardTerminalMappingType , DDR4-x16StandardTerminalNameType .																																																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQSU_c</td><td>7. DQSU_t</td><td>8. DQU[0]</td></tr> <tr><td>9. DQU[1]</td><td>10. DQU[2]</td><td>11. DQU[3]</td><td>12. DQU[4]</td></tr> <tr><td>13. DQU[5]</td><td>14. DQU[6]</td><td>15. DQU[7]</td><td>16. DQL[0]</td></tr> <tr><td>17. DQL[1]</td><td>18. DQL[2]</td><td>19. DQL[3]</td><td>20. DQL[4]</td></tr> <tr><td>21. DQL[5]</td><td>22. DQL[6]</td><td>23. DQL[7]</td><td>24. DMU_n/DBIU_n</td></tr> <tr><td>25. DML_n/DBIL_n</td><td>26. DQSL_c</td><td>27. DQSL_t</td><td>28. CKE</td></tr> <tr><td>29. ODT</td><td>30. CK_t</td><td>31. CK_c</td><td>32. A0</td></tr> <tr><td>33. A1</td><td>34. A2</td><td>35. A3</td><td>36. A4</td></tr> <tr><td>37. A5</td><td>38. A6</td><td>39. A7</td><td>40. A8</td></tr> <tr><td>41. A9</td><td>42. A10/AP</td><td>43. A11</td><td>44. A12/BC_n</td></tr> <tr><td>45. A13</td><td>46. WE_n/A14</td><td>47. CAS_n/A15</td><td>48. RAS_n/A16</td></tr> <tr><td>49. ACT_n</td><td>50. CS_n</td><td>51. VREFCA</td><td>52. BG0</td></tr> <tr><td>53. BA0</td><td>54. BA1</td><td>55. TEN</td><td>56. RESET_n</td></tr> <tr><td>57. ALERT_n</td><td>58. PAR</td><td>59. ZQ</td><td></td></tr> </tbody> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQSU_c	7. DQSU_t	8. DQU[0]	9. DQU[1]	10. DQU[2]	11. DQU[3]	12. DQU[4]	13. DQU[5]	14. DQU[6]	15. DQU[7]	16. DQL[0]	17. DQL[1]	18. DQL[2]	19. DQL[3]	20. DQL[4]	21. DQL[5]	22. DQL[6]	23. DQL[7]	24. DMU_n/DBIU_n	25. DML_n/DBIL_n	26. DQSL_c	27. DQSL_t	28. CKE	29. ODT	30. CK_t	31. CK_c	32. A0	33. A1	34. A2	35. A3	36. A4	37. A5	38. A6	39. A7	40. A8	41. A9	42. A10/AP	43. A11	44. A12/BC_n	45. A13	46. WE_n/A14	47. CAS_n/A15	48. RAS_n/A16	49. ACT_n	50. CS_n	51. VREFCA	52. BG0	53. BA0	54. BA1	55. TEN	56. RESET_n	57. ALERT_n	58. PAR	59. ZQ	
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																										
5. VSS	6. DQSU_c	7. DQSU_t	8. DQU[0]																																																										
9. DQU[1]	10. DQU[2]	11. DQU[3]	12. DQU[4]																																																										
13. DQU[5]	14. DQU[6]	15. DQU[7]	16. DQL[0]																																																										
17. DQL[1]	18. DQL[2]	19. DQL[3]	20. DQL[4]																																																										
21. DQL[5]	22. DQL[6]	23. DQL[7]	24. DMU_n/DBIU_n																																																										
25. DML_n/DBIL_n	26. DQSL_c	27. DQSL_t	28. CKE																																																										
29. ODT	30. CK_t	31. CK_c	32. A0																																																										
33. A1	34. A2	35. A3	36. A4																																																										
37. A5	38. A6	39. A7	40. A8																																																										
41. A9	42. A10/AP	43. A11	44. A12/BC_n																																																										
45. A13	46. WE_n/A14	47. CAS_n/A15	48. RAS_n/A16																																																										
49. ACT_n	50. CS_n	51. VREFCA	52. BG0																																																										
53. BA0	54. BA1	55. TEN	56. RESET_n																																																										
57. ALERT_n	58. PAR	59. ZQ																																																											

For more information about the DDR4-x16 Interface, refer to the JEDEC standard JESD79-4B.

4.7.3.10.7.6. DDR4-x16 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x16-DualDie																																																																
diagram	<pre> classDiagram class DDR4-x16-DualDie-InterfaceFunctionType class StandardTerminalNameAssignment { <<DDR4-x16-DualDie-StandardTerminalNameAssignmentType>> } class Mapping { <<DDR4-x16-DualDie-StandardTerminalMappingType>> } class StandardTerminalName { <<DDR4-x16-DualDie-StandardTerminalNameType>> } DDR4-x16-DualDie-InterfaceFunctionType "1..∞" -- "61" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" -- "61" Mapping Mapping -- "61" StandardTerminalName </pre>																																																																
type	DDR4-x16-DualDie-InterfaceFunctionType , DDR4-x16-DualDieStandardTerminalNameAssignmentType , DDR4-x16-DualDieStandardTerminalMappingType , DDR4-x16-DualDieStandardTerminalNameType .																																																																
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQSU_c</td><td>7. DQSU_t</td><td>8. DQU[0]</td></tr> <tr><td>9. DQU[1]</td><td>10. DQU[2]</td><td>11. DQU[3]</td><td>12. DQU[4]</td></tr> <tr><td>13. DQU[5]</td><td>14. DQU[6]</td><td>15. DQU[7]</td><td>16. DQL[0]</td></tr> <tr><td>17. DQL[1]</td><td>18. DQL[2]</td><td>19. DQL[3]</td><td>20. DQL[4]</td></tr> <tr><td>21. DQL[5]</td><td>22. DQL[6]</td><td>23. DQL[7]</td><td>24. DMU_n/DBIU_n</td></tr> <tr><td>25. DML_n/DBIL_n</td><td>26. DQSL_c</td><td>27. DQSL_t</td><td>28. CKE</td></tr> <tr><td>29. ODT</td><td>30. CK_t</td><td>31. CK_c</td><td>32. A0</td></tr> <tr><td>33. A1</td><td>34. A2</td><td>35. A3</td><td>36. A4</td></tr> <tr><td>37. A5</td><td>38. A6</td><td>39. A7</td><td>40. A8</td></tr> <tr><td>41. A9</td><td>42. A10/AP</td><td>43. A11</td><td>44. A12/BC_n</td></tr> <tr><td>45. A13</td><td>46. WE_n/A14</td><td>47. CAS_n/A15</td><td>48. RAS_n/A16</td></tr> <tr><td>49. ACT_n</td><td>50. CS_n</td><td>51. VREFCA</td><td>52. BG0</td></tr> <tr><td>53. BG1</td><td>54. BA0</td><td>55. BA1</td><td>56. TEN</td></tr> <tr><td>57. RESET_n</td><td>58. ALERT_n</td><td>59. PAR</td><td>60. UZQ</td></tr> <tr><td>61. LZQ</td><td></td><td></td><td></td></tr> </tbody> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQSU_c	7. DQSU_t	8. DQU[0]	9. DQU[1]	10. DQU[2]	11. DQU[3]	12. DQU[4]	13. DQU[5]	14. DQU[6]	15. DQU[7]	16. DQL[0]	17. DQL[1]	18. DQL[2]	19. DQL[3]	20. DQL[4]	21. DQL[5]	22. DQL[6]	23. DQL[7]	24. DMU_n/DBIU_n	25. DML_n/DBIL_n	26. DQSL_c	27. DQSL_t	28. CKE	29. ODT	30. CK_t	31. CK_c	32. A0	33. A1	34. A2	35. A3	36. A4	37. A5	38. A6	39. A7	40. A8	41. A9	42. A10/AP	43. A11	44. A12/BC_n	45. A13	46. WE_n/A14	47. CAS_n/A15	48. RAS_n/A16	49. ACT_n	50. CS_n	51. VREFCA	52. BG0	53. BG1	54. BA0	55. BA1	56. TEN	57. RESET_n	58. ALERT_n	59. PAR	60. UZQ	61. LZQ			
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																														
5. VSS	6. DQSU_c	7. DQSU_t	8. DQU[0]																																																														
9. DQU[1]	10. DQU[2]	11. DQU[3]	12. DQU[4]																																																														
13. DQU[5]	14. DQU[6]	15. DQU[7]	16. DQL[0]																																																														
17. DQL[1]	18. DQL[2]	19. DQL[3]	20. DQL[4]																																																														
21. DQL[5]	22. DQL[6]	23. DQL[7]	24. DMU_n/DBIU_n																																																														
25. DML_n/DBIL_n	26. DQSL_c	27. DQSL_t	28. CKE																																																														
29. ODT	30. CK_t	31. CK_c	32. A0																																																														
33. A1	34. A2	35. A3	36. A4																																																														
37. A5	38. A6	39. A7	40. A8																																																														
41. A9	42. A10/AP	43. A11	44. A12/BC_n																																																														
45. A13	46. WE_n/A14	47. CAS_n/A15	48. RAS_n/A16																																																														
49. ACT_n	50. CS_n	51. VREFCA	52. BG0																																																														
53. BG1	54. BA0	55. BA1	56. TEN																																																														
57. RESET_n	58. ALERT_n	59. PAR	60. UZQ																																																														
61. LZQ																																																																	

For more information about the DDR4-x16 Dual Die Interface, refer to the JEDEC standard JESD79-4B.

4.7.3.10.7.7. DDR4-x32

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x32																																																																																				
diagram																																																																																					
type	DDR4-x32-InterfaceFunctionType , DDR4-x32StandardTerminalNameAssignmentType , DDR4-x32StandardTerminalMappingType , DDR4-x32StandardTerminalNameType .																																																																																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. VDDQ</td><td>2. VSSQ</td><td>3. VPP</td><td>4. VDD</td></tr> <tr><td>5. VSS</td><td>6. DQS0_t</td><td>7. DQS0_c</td><td>8. DQS1_c</td></tr> <tr><td>9. DQS1_t</td><td>10. ODT</td><td>11. ODT1</td><td>12. CS_n</td></tr> <tr><td>13. CS1_n</td><td>14. CKE</td><td>15. CKE1</td><td>16. CK_t</td></tr> <tr><td>17. CK_c</td><td>18. TEN</td><td>19. VREFCA</td><td>20. BA0</td></tr> <tr><td>21. BA1</td><td>22. ACT_n</td><td>23. BG0</td><td>24. RESET_n</td></tr> <tr><td>25. RFU</td><td>26. ALERT_n</td><td>27. PAR</td><td>28. ZQ</td></tr> <tr><td>29. ZQ1</td><td>30. A0</td><td>31. A1</td><td>32. A2</td></tr> <tr><td>33. A3</td><td>34. A4</td><td>35. A5</td><td>36. A6</td></tr> <tr><td>37. A7</td><td>38. A8</td><td>39. A9</td><td>40. A10/AP</td></tr> <tr><td>41. A11</td><td>42. A12/BC_n</td><td>43. A13</td><td>44. WE_n/A14</td></tr> <tr><td>45. CAS_n/A15</td><td>46. RAS_n/A16</td><td>47. DM0_n/DBI0_n</td><td>48. DM1_n/DBI1_n</td></tr> <tr><td>49. DM2_n/DBI2_n</td><td>50. DM3_n/DBI3_n</td><td>51. DQ[0]</td><td>52. DQ[1]</td></tr> <tr><td>53. DQ[2]</td><td>54. DQ[3]</td><td>55. DQ[4]</td><td>56. DQ[5]</td></tr> <tr><td>57. DQ[6]</td><td>58. DQ[7]</td><td>59. DQ[8]</td><td>60. DQ[9]</td></tr> <tr><td>61. DQ[10]</td><td>62. DQ[11]</td><td>63. DQ[12]</td><td>64. DQ[13]</td></tr> <tr><td>65. DQ[14]</td><td>66. DQ[15]</td><td>67. DQ[16]</td><td>68. DQ[17]</td></tr> <tr><td>69. DQ[18]</td><td>70. DQ[19]</td><td>71. DQ[20]</td><td>72. DQ[21]</td></tr> <tr><td>73. DQ[22]</td><td>74. DQ[23]</td><td>75. DQ[24]</td><td>76. DQ[25]</td></tr> <tr><td>77. DQ[26]</td><td>78. DQ[27]</td><td>79. DQ[28]</td><td>80. DQ[29]</td></tr> <tr><td>81. DQ[30]</td><td>82. DQ[31]</td><td></td><td></td></tr> </tbody> </table>	1. VDDQ	2. VSSQ	3. VPP	4. VDD	5. VSS	6. DQS0_t	7. DQS0_c	8. DQS1_c	9. DQS1_t	10. ODT	11. ODT1	12. CS_n	13. CS1_n	14. CKE	15. CKE1	16. CK_t	17. CK_c	18. TEN	19. VREFCA	20. BA0	21. BA1	22. ACT_n	23. BG0	24. RESET_n	25. RFU	26. ALERT_n	27. PAR	28. ZQ	29. ZQ1	30. A0	31. A1	32. A2	33. A3	34. A4	35. A5	36. A6	37. A7	38. A8	39. A9	40. A10/AP	41. A11	42. A12/BC_n	43. A13	44. WE_n/A14	45. CAS_n/A15	46. RAS_n/A16	47. DM0_n/DBI0_n	48. DM1_n/DBI1_n	49. DM2_n/DBI2_n	50. DM3_n/DBI3_n	51. DQ[0]	52. DQ[1]	53. DQ[2]	54. DQ[3]	55. DQ[4]	56. DQ[5]	57. DQ[6]	58. DQ[7]	59. DQ[8]	60. DQ[9]	61. DQ[10]	62. DQ[11]	63. DQ[12]	64. DQ[13]	65. DQ[14]	66. DQ[15]	67. DQ[16]	68. DQ[17]	69. DQ[18]	70. DQ[19]	71. DQ[20]	72. DQ[21]	73. DQ[22]	74. DQ[23]	75. DQ[24]	76. DQ[25]	77. DQ[26]	78. DQ[27]	79. DQ[28]	80. DQ[29]	81. DQ[30]	82. DQ[31]		
1. VDDQ	2. VSSQ	3. VPP	4. VDD																																																																																		
5. VSS	6. DQS0_t	7. DQS0_c	8. DQS1_c																																																																																		
9. DQS1_t	10. ODT	11. ODT1	12. CS_n																																																																																		
13. CS1_n	14. CKE	15. CKE1	16. CK_t																																																																																		
17. CK_c	18. TEN	19. VREFCA	20. BA0																																																																																		
21. BA1	22. ACT_n	23. BG0	24. RESET_n																																																																																		
25. RFU	26. ALERT_n	27. PAR	28. ZQ																																																																																		
29. ZQ1	30. A0	31. A1	32. A2																																																																																		
33. A3	34. A4	35. A5	36. A6																																																																																		
37. A7	38. A8	39. A9	40. A10/AP																																																																																		
41. A11	42. A12/BC_n	43. A13	44. WE_n/A14																																																																																		
45. CAS_n/A15	46. RAS_n/A16	47. DM0_n/DBI0_n	48. DM1_n/DBI1_n																																																																																		
49. DM2_n/DBI2_n	50. DM3_n/DBI3_n	51. DQ[0]	52. DQ[1]																																																																																		
53. DQ[2]	54. DQ[3]	55. DQ[4]	56. DQ[5]																																																																																		
57. DQ[6]	58. DQ[7]	59. DQ[8]	60. DQ[9]																																																																																		
61. DQ[10]	62. DQ[11]	63. DQ[12]	64. DQ[13]																																																																																		
65. DQ[14]	66. DQ[15]	67. DQ[16]	68. DQ[17]																																																																																		
69. DQ[18]	70. DQ[19]	71. DQ[20]	72. DQ[21]																																																																																		
73. DQ[22]	74. DQ[23]	75. DQ[24]	76. DQ[25]																																																																																		
77. DQ[26]	78. DQ[27]	79. DQ[28]	80. DQ[29]																																																																																		
81. DQ[30]	82. DQ[31]																																																																																				

For more information about the DDR4-x32 Interface, refer to the JEDEC standard JESD79-4B.

4.7.3.10.7.8. DDR4-x72

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x72																																																																																																																												
diagram	<pre> classDiagram class DDR4-x72-InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName DDR4-x72-InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for DDR4-x72. It features four main classes: DDR4-x72-InterfaceFunctionType, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The DDR4-x72-InterfaceFunctionType class is associated with the StandardTerminalNameAssignment class via a multiplicity of 1..>. The StandardTerminalNameAssignment class is associated with the Mapping class via a multiplicity of 1..>. The Mapping class is associated with the StandardTerminalName class via a multiplicity of 1..>. A dashed green box labeled 'constraints' is positioned near the Mapping class.</p>																																																																																																																												
type	DDR4-x72-InterfaceFunctionType, DDR4-x72-StandardTerminalNameAssignmentType, DDR4-x72-StandardTerminalMappingType, DDR4-x72-StandardTerminalNameType.																																																																																																																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. A0</td><td>2. A1</td><td>3. A10/AP</td><td>4. A11</td></tr> <tr><td>5. A12/BC_n</td><td>6. A13</td><td>7. A17</td><td>8. A2</td></tr> <tr><td>9. A3</td><td>10. A4</td><td>11. A5</td><td>12. A6</td></tr> <tr><td>13. A7</td><td>14. A8</td><td>15. A9</td><td>16. ACT_n</td></tr> <tr><td>17. ALERT_n</td><td>18. BA0</td><td>19. BA1</td><td>20. BG0</td></tr> <tr><td>21. BG1</td><td>22. CAS_n/A15</td><td>23. CB0</td><td>24. CB1</td></tr> <tr><td>25. CB2</td><td>26. CB3</td><td>27. CB4</td><td>28. CB5</td></tr> <tr><td>29. CB6</td><td>30. CB7</td><td>31. CK0_c</td><td>32. CK0_t</td></tr> <tr><td>33. CK1_c</td><td>34. CK1_t</td><td>35. CKE0</td><td>36. CKE1/NC</td></tr> <tr><td>37. CS0_n</td><td>38. CS1_n/NC</td><td>39. CS2_n/C0</td><td>40. CS3_n/C1, NC</td></tr> <tr><td>41. DQ[0]</td><td>42. DQ[1]</td><td>43. DQ[2]</td><td>44. DQ[3]</td></tr> <tr><td>45. DQ[4]</td><td>46. DQ[5]</td><td>47. DQ[6]</td><td>48. DQ[7]</td></tr> <tr><td>49. DQ[8]</td><td>50. DQ[9]</td><td>51. DQ[10]</td><td>52. DQ[11]</td></tr> <tr><td>53. DQ[12]</td><td>54. DQ[13]</td><td>55. DQ[14]</td><td>56. DQ[15]</td></tr> <tr><td>57. DQ[16]</td><td>58. DQ[17]</td><td>59. DQ[18]</td><td>60. DQ[19]</td></tr> <tr><td>61. DQ[20]</td><td>62. DQ[21]</td><td>63. DQ[22]</td><td>64. DQ[23]</td></tr> <tr><td>65. DQ[24]</td><td>66. DQ[25]</td><td>67. DQ[26]</td><td>68. DQ[27]</td></tr> <tr><td>69. DQ[28]</td><td>70. DQ[29]</td><td>71. DQ[30]</td><td>72. DQ[31]</td></tr> <tr><td>73. DQ[32]</td><td>74. DQ[33]</td><td>75. DQ[34]</td><td>76. DQ[35]</td></tr> <tr><td>77. DQ[36]</td><td>78. DQ[37]</td><td>79. DQ[38]</td><td>80. DQ[39]</td></tr> <tr><td>81. DQ[40]</td><td>82. DQ[41]</td><td>83. DQ[42]</td><td>84. DQ[43]</td></tr> <tr><td>85. DQ[44]</td><td>86. DQ[45]</td><td>87. DQ[46]</td><td>88. DQ[47]</td></tr> <tr><td>89. DQ[48]</td><td>90. DQ[49]</td><td>91. DQ[50]</td><td>92. DQ[51]</td></tr> <tr><td>93. DQ[52]</td><td>94. DQ[53]</td><td>95. DQ[54]</td><td>96. DQ[55]</td></tr> <tr><td>97. DQ[56]</td><td>98. DQ[57]</td><td>99. DQ[58]</td><td>100. DQ[59]</td></tr> <tr><td>101. DQ[60]</td><td>102. DQ[61]</td><td>103. DQ[62]</td><td>104. DQ[63]</td></tr> <tr><td>105. DQS0_c</td><td>106. DQS0_t</td><td>107. DQS09_c/TDQS9_c</td><td>108. DQS1_c</td></tr> <tr><td>109. DQS1_t</td><td>110. DQS10_c/TDQS10_c</td><td>111. DQS10_t/TDQS10_t</td><td>112. DQS11_c/TDQS11_c</td></tr> <tr><td>113. DQS11_t/TDQS11_t</td><td>114. DQS12_c/TDQS12_c</td><td>115. DQS12_t/TDQS12_t</td><td>116. DQS13_c/TDQS13_c</td></tr> <tr><td>117. DQS13_t/TDQS13_t</td><td>118. DQS14_c/TDQS14_c</td><td>119. DQS14_t/TDQS14_t</td><td>120. DQS15_c/TDQS15_c</td></tr> <tr><td>121. DQS15_t/TDQS15_t</td><td>122. DQS16_c/TDQS16_c</td><td>123. DQS16_t/TDQS16_t</td><td>124. DQS17_c/TDQS17_c</td></tr> </tbody> </table>	1. A0	2. A1	3. A10/AP	4. A11	5. A12/BC_n	6. A13	7. A17	8. A2	9. A3	10. A4	11. A5	12. A6	13. A7	14. A8	15. A9	16. ACT_n	17. ALERT_n	18. BA0	19. BA1	20. BG0	21. BG1	22. CAS_n/A15	23. CB0	24. CB1	25. CB2	26. CB3	27. CB4	28. CB5	29. CB6	30. CB7	31. CK0_c	32. CK0_t	33. CK1_c	34. CK1_t	35. CKE0	36. CKE1/NC	37. CS0_n	38. CS1_n/NC	39. CS2_n/C0	40. CS3_n/C1, NC	41. DQ[0]	42. DQ[1]	43. DQ[2]	44. DQ[3]	45. DQ[4]	46. DQ[5]	47. DQ[6]	48. DQ[7]	49. DQ[8]	50. DQ[9]	51. DQ[10]	52. DQ[11]	53. DQ[12]	54. DQ[13]	55. DQ[14]	56. DQ[15]	57. DQ[16]	58. DQ[17]	59. DQ[18]	60. DQ[19]	61. DQ[20]	62. DQ[21]	63. DQ[22]	64. DQ[23]	65. DQ[24]	66. DQ[25]	67. DQ[26]	68. DQ[27]	69. DQ[28]	70. DQ[29]	71. DQ[30]	72. DQ[31]	73. DQ[32]	74. DQ[33]	75. DQ[34]	76. DQ[35]	77. DQ[36]	78. DQ[37]	79. DQ[38]	80. DQ[39]	81. DQ[40]	82. DQ[41]	83. DQ[42]	84. DQ[43]	85. DQ[44]	86. DQ[45]	87. DQ[46]	88. DQ[47]	89. DQ[48]	90. DQ[49]	91. DQ[50]	92. DQ[51]	93. DQ[52]	94. DQ[53]	95. DQ[54]	96. DQ[55]	97. DQ[56]	98. DQ[57]	99. DQ[58]	100. DQ[59]	101. DQ[60]	102. DQ[61]	103. DQ[62]	104. DQ[63]	105. DQS0_c	106. DQS0_t	107. DQS09_c/TDQS9_c	108. DQS1_c	109. DQS1_t	110. DQS10_c/TDQS10_c	111. DQS10_t/TDQS10_t	112. DQS11_c/TDQS11_c	113. DQS11_t/TDQS11_t	114. DQS12_c/TDQS12_c	115. DQS12_t/TDQS12_t	116. DQS13_c/TDQS13_c	117. DQS13_t/TDQS13_t	118. DQS14_c/TDQS14_c	119. DQS14_t/TDQS14_t	120. DQS15_c/TDQS15_c	121. DQS15_t/TDQS15_t	122. DQS16_c/TDQS16_c	123. DQS16_t/TDQS16_t	124. DQS17_c/TDQS17_c
1. A0	2. A1	3. A10/AP	4. A11																																																																																																																										
5. A12/BC_n	6. A13	7. A17	8. A2																																																																																																																										
9. A3	10. A4	11. A5	12. A6																																																																																																																										
13. A7	14. A8	15. A9	16. ACT_n																																																																																																																										
17. ALERT_n	18. BA0	19. BA1	20. BG0																																																																																																																										
21. BG1	22. CAS_n/A15	23. CB0	24. CB1																																																																																																																										
25. CB2	26. CB3	27. CB4	28. CB5																																																																																																																										
29. CB6	30. CB7	31. CK0_c	32. CK0_t																																																																																																																										
33. CK1_c	34. CK1_t	35. CKE0	36. CKE1/NC																																																																																																																										
37. CS0_n	38. CS1_n/NC	39. CS2_n/C0	40. CS3_n/C1, NC																																																																																																																										
41. DQ[0]	42. DQ[1]	43. DQ[2]	44. DQ[3]																																																																																																																										
45. DQ[4]	46. DQ[5]	47. DQ[6]	48. DQ[7]																																																																																																																										
49. DQ[8]	50. DQ[9]	51. DQ[10]	52. DQ[11]																																																																																																																										
53. DQ[12]	54. DQ[13]	55. DQ[14]	56. DQ[15]																																																																																																																										
57. DQ[16]	58. DQ[17]	59. DQ[18]	60. DQ[19]																																																																																																																										
61. DQ[20]	62. DQ[21]	63. DQ[22]	64. DQ[23]																																																																																																																										
65. DQ[24]	66. DQ[25]	67. DQ[26]	68. DQ[27]																																																																																																																										
69. DQ[28]	70. DQ[29]	71. DQ[30]	72. DQ[31]																																																																																																																										
73. DQ[32]	74. DQ[33]	75. DQ[34]	76. DQ[35]																																																																																																																										
77. DQ[36]	78. DQ[37]	79. DQ[38]	80. DQ[39]																																																																																																																										
81. DQ[40]	82. DQ[41]	83. DQ[42]	84. DQ[43]																																																																																																																										
85. DQ[44]	86. DQ[45]	87. DQ[46]	88. DQ[47]																																																																																																																										
89. DQ[48]	90. DQ[49]	91. DQ[50]	92. DQ[51]																																																																																																																										
93. DQ[52]	94. DQ[53]	95. DQ[54]	96. DQ[55]																																																																																																																										
97. DQ[56]	98. DQ[57]	99. DQ[58]	100. DQ[59]																																																																																																																										
101. DQ[60]	102. DQ[61]	103. DQ[62]	104. DQ[63]																																																																																																																										
105. DQS0_c	106. DQS0_t	107. DQS09_c/TDQS9_c	108. DQS1_c																																																																																																																										
109. DQS1_t	110. DQS10_c/TDQS10_c	111. DQS10_t/TDQS10_t	112. DQS11_c/TDQS11_c																																																																																																																										
113. DQS11_t/TDQS11_t	114. DQS12_c/TDQS12_c	115. DQS12_t/TDQS12_t	116. DQS13_c/TDQS13_c																																																																																																																										
117. DQS13_t/TDQS13_t	118. DQS14_c/TDQS14_c	119. DQS14_t/TDQS14_t	120. DQS15_c/TDQS15_c																																																																																																																										
121. DQS15_t/TDQS15_t	122. DQS16_c/TDQS16_c	123. DQS16_t/TDQS16_t	124. DQS17_c/TDQS17_c																																																																																																																										

4.7.3.10.7.8 DDR4-x72 (cont'd)

list of enumerate values	Mapping/StandardTerminalName			
125. DQS17_t/TDQS17_t	126. DQS2_c	127. DQS2_t	128. DQS3_c	
129. DQS3_t	130. DQS4_c	131. DQS4_t	132. DQS5_c	
133. DQS5_t	134. DQS6_c	135. DQS6_t	136. DQS7_c	
137. DQS7_t	138. DQS8_c	139. DQS8_t	140. DQS9_t/TDQS9_t	
141. EVENT_n	142. NC	143. NC/C2	144. ODT0	
145. ODT1/NC	146. PARITY	147. RAS_n/A16	148. RESET_n	
149. SA0	150. SA1	151. SA2	152. SCL	
153. SDA	154. VDD	155. VDDSPD	156. VPP	
157. VREFCA	158. VSS	159. VTT	160. WE_n/A14	
161. DQS1_t	162. DQS10_c/TDQS10_c	163. DQS10_t/TDQS10_t	164. DQS11_c/TDQS11_c	

4.7.3.10.7.9. DDR4DB02 - 53 Ball Configuration

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4DB02																																												
diagram	<pre> classDiagram class StandardTerminalNameAssignment { <<type>> DDR4DB02-53BallConfig-StandardTerminalNameAssignmentType } class Mapping { <<type>> DDR4DB02-53BallConfig-StandardTerminalMappingType } class StandardTerminalName { <<type>> DDR4DB02-53BallConfig-StandardTerminalNameType } StandardTerminalNameAssignment "1..oo" -- "1..oo" Mapping : StandardTerminalNameAssignment "1..oo" -- "1..oo" StandardTerminalName : Mapping "1..oo" -- "1..oo" StandardTerminalNameAssignment : </pre>																																												
type	DDR4DB02-53BallConfig-InterfaceFunctionType, DDR4DB02-53BallConfig-StandardTerminalNameAssignmentType, DDR4DB02-53BallConfig-StandardTerminalMappingType, DDR4DB02-53BallConfig-StandardTerminalNameType.																																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. ALERT_n</td><td>2. BCK_c</td><td>3. BCK_t</td><td>4. BCKE</td></tr> <tr> <td>5. BCOM0</td><td>6. BCOM1</td><td>7. BCOM2</td><td>8. BCOM3</td></tr> <tr> <td>9. BODT</td><td>10. BVrefCA</td><td>11. DQ0</td><td>12. DQ1</td></tr> <tr> <td>13. DQ2</td><td>14. DQ3</td><td>15. DQ4</td><td>16. DQ5</td></tr> <tr> <td>17. DQ6</td><td>18. DQ7</td><td>19. DQS0_c</td><td>20. DQS0_t</td></tr> <tr> <td>21. DQS1_c</td><td>22. DQS1_t</td><td>23. MDQ0</td><td>24. MDQ1</td></tr> <tr> <td>25. MDQ2</td><td>26. MDQ3</td><td>27. MDQ4</td><td>28. MDQ5</td></tr> <tr> <td>29. MDQ6</td><td>30. MDQ7</td><td>31. MDQS0_c</td><td>32. MDQS0_t</td></tr> <tr> <td>33. MDQS1_c</td><td>34. MDQS1_t</td><td>35. VDD</td><td>36. VSS</td></tr> <tr> <td>37. ZQCAL</td><td></td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. ALERT_n	2. BCK_c	3. BCK_t	4. BCKE	5. BCOM0	6. BCOM1	7. BCOM2	8. BCOM3	9. BODT	10. BVrefCA	11. DQ0	12. DQ1	13. DQ2	14. DQ3	15. DQ4	16. DQ5	17. DQ6	18. DQ7	19. DQS0_c	20. DQS0_t	21. DQS1_c	22. DQS1_t	23. MDQ0	24. MDQ1	25. MDQ2	26. MDQ3	27. MDQ4	28. MDQ5	29. MDQ6	30. MDQ7	31. MDQS0_c	32. MDQS0_t	33. MDQS1_c	34. MDQS1_t	35. VDD	36. VSS	37. ZQCAL			
Mapping/StandardTerminalName																																													
1. ALERT_n	2. BCK_c	3. BCK_t	4. BCKE																																										
5. BCOM0	6. BCOM1	7. BCOM2	8. BCOM3																																										
9. BODT	10. BVrefCA	11. DQ0	12. DQ1																																										
13. DQ2	14. DQ3	15. DQ4	16. DQ5																																										
17. DQ6	18. DQ7	19. DQS0_c	20. DQS0_t																																										
21. DQS1_c	22. DQS1_t	23. MDQ0	24. MDQ1																																										
25. MDQ2	26. MDQ3	27. MDQ4	28. MDQ5																																										
29. MDQ6	30. MDQ7	31. MDQS0_c	32. MDQS0_t																																										
33. MDQS1_c	34. MDQS1_t	35. VDD	36. VSS																																										
37. ZQCAL																																													

For more information about the DDR4DB02 Interface, refer to the JEDEC standard JESD82-32A.

4.7.3.10.7.10. DDR4DB02 - 56 Ball Configuration

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4DB02																																								
diagram	<pre> classDiagram class DDR4DB02-56BallConfig-InterfaceFunctionType { <<DDR4DB02-56BallConfig-InterfaceFunctionType>> } class DDR4DB02-56BallConfig-StandardTerminalNameAssignmentType { <<DDR4DB02-56BallConfig-StandardTerminalNameAssignmentType>> } class DDR4DB02-56BallConfig-StandardTerminalMappingType { <<DDR4DB02-56BallConfig-StandardTerminalMappingType>> } class DDR4DB02-56BallConfig-StandardTerminalNameType { <<DDR4DB02-56BallConfig-StandardTerminalNameType>> } DDR4DB02-56BallConfig-InterfaceFunctionType "1..oo" --> DDR4DB02-56BallConfig-StandardTerminalNameAssignmentType : DDR4DB02-56BallConfig-StandardTerminalNameAssignmentType "1..oo" --> DDR4DB02-56BallConfig-StandardTerminalMappingType : DDR4DB02-56BallConfig-StandardTerminalMappingType "1..oo" --> DDR4DB02-56BallConfig-StandardTerminalNameType : DDR4DB02-56BallConfig-StandardTerminalNameAssignmentType "1..oo" --> constraints : DDR4DB02-56BallConfig-StandardTerminalNameType "1..oo" --> constraints : </pre>																																								
type	DDR4DB02-56BallConfig-InterfaceFunctionType , DDR4DB02-56BallConfig-StandardTerminalNameAssignmentType , DDR4DB02-56BallConfig-StandardTerminalMappingType , DDR4DB02-56BallConfig-StandardTerminalNameType .																																								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. ALERT_n</td><td>2. BCK_c</td><td>3. BCK_t</td><td>4. BCKE</td></tr> <tr><td>5. BCOM0</td><td>6. BCOM1</td><td>7. BCOM2</td><td>8. BCOM3</td></tr> <tr><td>9. BODT</td><td>10. BVrefCA</td><td>11. DQ0</td><td>12. DQ1</td></tr> <tr><td>13. DQ2</td><td>14. DQ3</td><td>15. DQ4</td><td>16. DQ5</td></tr> <tr><td>17. DQ6</td><td>18. DQ7</td><td>19. DQS0_c</td><td>20. DQS0_t</td></tr> <tr><td>21. DQS1_c</td><td>22. DQS1_t</td><td>23. LDQ0</td><td>24. LDQ1</td></tr> <tr><td>25. LDQS</td><td>26. MDQ0</td><td>27. MDQ1</td><td>28. MDQ2</td></tr> <tr><td>29. MDQ3</td><td>30. MDQ4</td><td>31. MDQ5</td><td>32. MDQ6</td></tr> <tr><td>33. MDQ7</td><td>34. MDQS0_c</td><td>35. MDQS0_t</td><td>36. MDQS1_c</td></tr> <tr><td>37. MDQS1_t</td><td>38. VDD</td><td>39. VSS</td><td>40. ZQCAL</td></tr> </table>	1. ALERT_n	2. BCK_c	3. BCK_t	4. BCKE	5. BCOM0	6. BCOM1	7. BCOM2	8. BCOM3	9. BODT	10. BVrefCA	11. DQ0	12. DQ1	13. DQ2	14. DQ3	15. DQ4	16. DQ5	17. DQ6	18. DQ7	19. DQS0_c	20. DQS0_t	21. DQS1_c	22. DQS1_t	23. LDQ0	24. LDQ1	25. LDQS	26. MDQ0	27. MDQ1	28. MDQ2	29. MDQ3	30. MDQ4	31. MDQ5	32. MDQ6	33. MDQ7	34. MDQS0_c	35. MDQS0_t	36. MDQS1_c	37. MDQS1_t	38. VDD	39. VSS	40. ZQCAL
1. ALERT_n	2. BCK_c	3. BCK_t	4. BCKE																																						
5. BCOM0	6. BCOM1	7. BCOM2	8. BCOM3																																						
9. BODT	10. BVrefCA	11. DQ0	12. DQ1																																						
13. DQ2	14. DQ3	15. DQ4	16. DQ5																																						
17. DQ6	18. DQ7	19. DQS0_c	20. DQS0_t																																						
21. DQS1_c	22. DQS1_t	23. LDQ0	24. LDQ1																																						
25. LDQS	26. MDQ0	27. MDQ1	28. MDQ2																																						
29. MDQ3	30. MDQ4	31. MDQ5	32. MDQ6																																						
33. MDQ7	34. MDQS0_c	35. MDQS0_t	36. MDQS1_c																																						
37. MDQS1_t	38. VDD	39. VSS	40. ZQCAL																																						

For more information about the DDR4DB02 Interface, refer to the JEDEC standard JESD82-32A.

4.7.3.10.7.11. DDR4RCD02

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4RCD02												
diagram	<pre> classDiagram class DDR4RCD02 { <<DDR4RCD02-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR4RCD02-StandardTerminalNameAssignmentType>> } class Mapping { <<DDR4RCD02-StandardTerminalMappingType>> } class StandardTerminalName { <<DDR4RCD02-StandardTerminalNameType>> } DDR4RCD02 "1..∞" --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>												
type	DDR4RCD02-InterfaceFunctionType , DDR4RCD02-StandardTerminalNameAssignmentType , DDR4RCD02-StandardTerminalMappingType , DDR4RCD02-StandardTerminalNameType .												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. BCOM0</td> <td>2. BCOM1</td> <td>3. BCOM2</td> <td>4. BCOM3</td> </tr> <tr> <td>5. BCKE</td> <td>6. BODT</td> <td>7. BCK_t</td> <td>8. BCK_c</td> </tr> <tr> <td>9. BVrefCA</td> <td></td> <td></td> <td></td> </tr> </table>	1. BCOM0	2. BCOM1	3. BCOM2	4. BCOM3	5. BCKE	6. BODT	7. BCK_t	8. BCK_c	9. BVrefCA			
1. BCOM0	2. BCOM1	3. BCOM2	4. BCOM3										
5. BCKE	6. BODT	7. BCK_t	8. BCK_c										
9. BVrefCA													

For more information about the DDR4CD02 Interface, refer to the JEDEC standard JESD82-31A.

4.7.3.10.7.12. DDR4-NVDIMM-N

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-NVDIMM-N																																																																												
diagram	<pre> classDiagram DDR4-NVDIMM-NInterfaceFunctionType "1..xx" -- "1..zz" StandardTerminalNameAssignment StandardTerminalNameAssignment "2..11" -- "1..zz" MandatoryMapping StandardTerminalNameAssignment "0..2" -- "0..2" OptionalMapping MandatoryMapping "1..zz" -- "1..zz" StandardTerminalName </pre>																																																																												
type	DDR4-NVDIMM-NInterfaceFunctionType , DDR4-NVDIMM-N-StandardTerminalNameAssignmentType , DDR4-NVDIMM-N-MandatoryStandardTerminalMappingType , DDR4-NVDIMM-N-MandatoryStandardTerminalNameType , DDR4-NVDIMM-N-OptionalStandardTerminalMappingType , DDR4-NVDIMM-N-OptionalStandardTerminalNameType .																																																																												
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. A0</td><td>2. A1</td><td>3. A2</td><td>4. A3</td></tr> <tr><td>5. A4</td><td>6. A5</td><td>7. A6</td><td>8. A7</td></tr> <tr><td>9. A8</td><td>10. A9</td><td>11. A10</td><td>12. A11</td></tr> <tr><td>13. A12</td><td>14. A13</td><td>15. A14</td><td>16. A15</td></tr> <tr><td>17. A16</td><td>18. BA0</td><td>19. BA1</td><td>20. BG0</td></tr> <tr><td>21. BG1</td><td>22. RAS_n</td><td>23. CAS_n</td><td>24. WE_n</td></tr> <tr><td>25. CS0_n</td><td>26. CS01_n</td><td>27. CS2_n</td><td>28. CS3_n</td></tr> <tr><td>29. CKE0</td><td>30. CKE1</td><td>31. ODT0</td><td>32. ODT1</td></tr> <tr><td>33. ACT_n</td><td>34. DQ[0]</td><td>35. DQ[1]</td><td>36. DQ[2]</td></tr> <tr><td>37. DQ[3]</td><td>38. DQ[4]</td><td>39. DQ[5]</td><td>40. DQ[6]</td></tr> <tr><td>41. DQ[7]</td><td>42. DQ[8]</td><td>43. DQ[9]</td><td>44. DQ[10]</td></tr> <tr><td>45. DQ[11]</td><td>46. DQ[12]</td><td>47. DQ[13]</td><td>48. DQ[14]</td></tr> <tr><td>49. DQ[15]</td><td>50. DQ[16]</td><td>51. DQ[17]</td><td>52. DQ[18]</td></tr> <tr><td>53. DQ[19]</td><td>54. DQ[20]</td><td>55. DQ[21]</td><td>56. DQ[22]</td></tr> <tr><td>57. DQ[23]</td><td>58. DQ[24]</td><td>59. DQ[25]</td><td>60. DQ[26]</td></tr> <tr><td>61. DQ[27]</td><td>62. DQ[28]</td><td>63. DQ[29]</td><td>64. DQ[30]</td></tr> <tr><td>65. DQ[31]</td><td>66. DQ[32]</td><td>67. DQ[33]</td><td>68. DQ[34]</td></tr> <tr><td>69. DQ[35]</td><td>70. DQ[36]</td><td>71. DQ[37]</td><td>72. DQ[38]</td></tr> <tr><td>73. DQ[39]</td><td>74. DQ[40]</td><td>75. DQ[41]</td><td>76. DQ[42]</td></tr> </tbody> </table>	1. A0	2. A1	3. A2	4. A3	5. A4	6. A5	7. A6	8. A7	9. A8	10. A9	11. A10	12. A11	13. A12	14. A13	15. A14	16. A15	17. A16	18. BA0	19. BA1	20. BG0	21. BG1	22. RAS_n	23. CAS_n	24. WE_n	25. CS0_n	26. CS01_n	27. CS2_n	28. CS3_n	29. CKE0	30. CKE1	31. ODT0	32. ODT1	33. ACT_n	34. DQ[0]	35. DQ[1]	36. DQ[2]	37. DQ[3]	38. DQ[4]	39. DQ[5]	40. DQ[6]	41. DQ[7]	42. DQ[8]	43. DQ[9]	44. DQ[10]	45. DQ[11]	46. DQ[12]	47. DQ[13]	48. DQ[14]	49. DQ[15]	50. DQ[16]	51. DQ[17]	52. DQ[18]	53. DQ[19]	54. DQ[20]	55. DQ[21]	56. DQ[22]	57. DQ[23]	58. DQ[24]	59. DQ[25]	60. DQ[26]	61. DQ[27]	62. DQ[28]	63. DQ[29]	64. DQ[30]	65. DQ[31]	66. DQ[32]	67. DQ[33]	68. DQ[34]	69. DQ[35]	70. DQ[36]	71. DQ[37]	72. DQ[38]	73. DQ[39]	74. DQ[40]	75. DQ[41]	76. DQ[42]
1. A0	2. A1	3. A2	4. A3																																																																										
5. A4	6. A5	7. A6	8. A7																																																																										
9. A8	10. A9	11. A10	12. A11																																																																										
13. A12	14. A13	15. A14	16. A15																																																																										
17. A16	18. BA0	19. BA1	20. BG0																																																																										
21. BG1	22. RAS_n	23. CAS_n	24. WE_n																																																																										
25. CS0_n	26. CS01_n	27. CS2_n	28. CS3_n																																																																										
29. CKE0	30. CKE1	31. ODT0	32. ODT1																																																																										
33. ACT_n	34. DQ[0]	35. DQ[1]	36. DQ[2]																																																																										
37. DQ[3]	38. DQ[4]	39. DQ[5]	40. DQ[6]																																																																										
41. DQ[7]	42. DQ[8]	43. DQ[9]	44. DQ[10]																																																																										
45. DQ[11]	46. DQ[12]	47. DQ[13]	48. DQ[14]																																																																										
49. DQ[15]	50. DQ[16]	51. DQ[17]	52. DQ[18]																																																																										
53. DQ[19]	54. DQ[20]	55. DQ[21]	56. DQ[22]																																																																										
57. DQ[23]	58. DQ[24]	59. DQ[25]	60. DQ[26]																																																																										
61. DQ[27]	62. DQ[28]	63. DQ[29]	64. DQ[30]																																																																										
65. DQ[31]	66. DQ[32]	67. DQ[33]	68. DQ[34]																																																																										
69. DQ[35]	70. DQ[36]	71. DQ[37]	72. DQ[38]																																																																										
73. DQ[39]	74. DQ[40]	75. DQ[41]	76. DQ[42]																																																																										

4.7.3.10.7.12 DDR4-NVDIMM-N (cont'd)

list of enumerate values (cont.)	MandatoryMapping/StandardTerminalName			
	77. DQ[43]	78. DQ[44]	79. DQ[45]	80. DQ[46]
	81. DQ[47]	82. DQ[48]	83. DQ[49]	84. DQ[50]
	85. DQ[51]	86. DQ[52]	87. DQ[53]	88. DQ[54]
	89. DQ[55]	90. DQ[56]	91. DQ[57]	92. DQ[58]
	93. DQ[59]	94. DQ[60]	95. DQ[61]	96. DQ[62]
	97. DQ[63]	98. CB0	99. CB1	100. CB2
	101. CB3	102. CB4	103. CB5	104. CB6
	105. CB7	106. TQDS9_t	107. TQDS10_t	108. TQDS11_t
	109. TQDS12_t	110. TQDS13_t	111. TQDS14_t	112. TQDS15_t
	113. TQDS16_t	114. TQDS17_t	115. TQDS9_c	116. TQDS10_c
	117. TQDS11_c	118. TQDS12_c	119. TQDS13_c	120. TQDS14_c
	121. TQDS15_c	122. TQDS16_c	123. TQDS17_c	124. DQS0_t
	125. DQS1_t	126. DQS2_t	127. DQS3_t	128. DQS4_t
	129. DQS5_t	130. DQS6_t	131. DQS7_t	132. DQS8_t
	133. DQS9_t	134. DQS10_t	135. DQS11_t	136. DQS12_t
	137. DQS13_t	138. DQS14_t	139. DQS15_t	140. DQS16_t
	141. DQS17_t	142. DQS0_c	143. DQS1_c	144. DQS2_c
	145. DQS3_c	146. DQS4_c	147. DQS5_c	148. DQS6_c
	149. DQS7_c	150. DQS8_c	151. DQS9_c	152. DQS10_c
	153. DQS11_c	154. DQS12_c	155. DQS13_c	156. DQS14_c
	157. DQS15_c	158. DQS16_c	159. DQS17_c	160. DBI0_n
	161. DBI1_n	162. DBI2_n	163. DBI3_n	164. DBI4_n
	165. DBI5_n	166. DBI6_n	167. DBI7_n	168. DBI8_n
	169. DBI9_n	170. DBI10_n	171. DBI11_n	172. DBI12_n
	173. DBI13_n	174. DBI14_n	175. DBI15_n	176. DBI16_n
	177. DBI17_n	178. DBI18_n	179. CK0_t	180. CK1_t
	181. CK0_c	182. CK1_c	183. SCL	184. SDA
	185. SA0	186. SA1	187. SA2	188. PAR
	189. VDD	190. C0	191. C1	192. C2
	193. 12V	194. VREFCA	195. VSS	196. VSDDSPD
	197. ALERT_n	198. VPP	199. SAVE_n	200. DM0_n
	201. DM1_n	202. DM2_n	203. DM3_n	204. DM4_n
	205. DM5_n	206. DM6_n	207. DM7_n	208. DM8_n
	209. RESET_n	210. EVENT_n	211. VTT	
list of enumerate values	OptionalMapping/StandardTerminalName			
	1. RFU	2. A17		

4.7.3.10.7.13. LPDDR4 – Single Channel

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/LPDDR4-SingleChannel																																								
diagram	<pre> classDiagram class LPDDR4-SingleChannel { <<LPDDR4-SingleChannelInterface>> } class StandardTerminalNameAssignment { <<LPDDR4-SingleChannelStandardTerminalNameAssignmentType>> } class Mapping { <<LPDDR4-SingleChannelStandardTerminalMappingType>> } class StandardTerminalName { <<LPDDR4-SingleChannelStandardTerminalNameType>> } LPDDR4-SingleChannel "1..xx" -- "1..xx" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..xx" -- "40" Mapping Mapping "40" -- "40" StandardTerminalName </pre>																																								
type	LPDDR4-SingleChannel-InterfaceFunctionType , LPDDR4-SingleChannelStandardTerminalNameAssignmentType , LPDDR4-SingleChannelStandardTerminalMappingType , LPDDR4-SingleChannelStandardTerminalNameType .																																								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. CK_t</td> <td>2. CK_c</td> <td>3. CKE</td> <td>4. CS</td> </tr> <tr> <td>5. CA0</td> <td>6. CA1</td> <td>7. CA2</td> <td>8. CA3</td> </tr> <tr> <td>9. CA4</td> <td>10. CA5</td> <td>11. ODT(ca)</td> <td>12. DQ[0]</td> </tr> <tr> <td>13. DQ[1]</td> <td>14. DQ[2]</td> <td>15. DQ[3]</td> <td>16. DQ[4]</td> </tr> <tr> <td>17. DQ[5]</td> <td>18. DQ[6]</td> <td>19. DQ[7]</td> <td>20. DQ[8]</td> </tr> <tr> <td>21. DQ[9]</td> <td>22. DQ[10]</td> <td>23. DQ[11]</td> <td>24. DQ[12]</td> </tr> <tr> <td>25. DQ[13]</td> <td>26. DQ[14]</td> <td>27. DQ[15]</td> <td>28. DQS0_t</td> </tr> <tr> <td>29. DQS1_t</td> <td>30. DQS0_c</td> <td>31. DQS1_c</td> <td>32. DMI0</td> </tr> <tr> <td>33. DMI1</td> <td>34. ZQ</td> <td>35. VDDQ</td> <td>36. VDD1</td> </tr> <tr> <td>37. VDD2</td> <td>38. VSS</td> <td>39. VSSQ</td> <td>40. RESET_n</td> </tr> </table>	1. CK_t	2. CK_c	3. CKE	4. CS	5. CA0	6. CA1	7. CA2	8. CA3	9. CA4	10. CA5	11. ODT(ca)	12. DQ[0]	13. DQ[1]	14. DQ[2]	15. DQ[3]	16. DQ[4]	17. DQ[5]	18. DQ[6]	19. DQ[7]	20. DQ[8]	21. DQ[9]	22. DQ[10]	23. DQ[11]	24. DQ[12]	25. DQ[13]	26. DQ[14]	27. DQ[15]	28. DQS0_t	29. DQS1_t	30. DQS0_c	31. DQS1_c	32. DMI0	33. DMI1	34. ZQ	35. VDDQ	36. VDD1	37. VDD2	38. VSS	39. VSSQ	40. RESET_n
1. CK_t	2. CK_c	3. CKE	4. CS																																						
5. CA0	6. CA1	7. CA2	8. CA3																																						
9. CA4	10. CA5	11. ODT(ca)	12. DQ[0]																																						
13. DQ[1]	14. DQ[2]	15. DQ[3]	16. DQ[4]																																						
17. DQ[5]	18. DQ[6]	19. DQ[7]	20. DQ[8]																																						
21. DQ[9]	22. DQ[10]	23. DQ[11]	24. DQ[12]																																						
25. DQ[13]	26. DQ[14]	27. DQ[15]	28. DQS0_t																																						
29. DQS1_t	30. DQS0_c	31. DQS1_c	32. DMI0																																						
33. DMI1	34. ZQ	35. VDDQ	36. VDD1																																						
37. VDD2	38. VSS	39. VSSQ	40. RESET_n																																						

For more information about the LPDDR4 – Single Channel Interface, refer to the JEDEC standard JESD209-4D.

4.7.3.10.7.14. LPDDR4 – Dual Channel

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/LPDDR4-DualChannel																																																																												
diagram	<pre> classDiagram class LPDDR4-DualChannel { <<LPDDR4-DualChannelInterfaceFunctionType>> } class StandardTerminalNameAssignment { <<LPDDR4-DualChannelStandardTerminalNameAssignmentType>> } class Mapping { <<LPDDR4-DualChannelStandardTerminalMappingType>> } class StandardTerminalName { <<LPDDR4-DualChannelStandardTerminalNameType>> } LPDDR4-DualChannel "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for the LPDDR4-DualChannel interface. It features four main classes: LPDDR4-DualChannel, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The LPDDR4-DualChannel class is associated with the StandardTerminalNameAssignment class via a multiplicity of 1..>. The StandardTerminalNameAssignment class is associated with the Mapping class via a multiplicity of 1..>. The Mapping class is associated with the StandardTerminalName class via a multiplicity of 1..>. A dashed box labeled 'constraints' encloses the multiplicity '1..>' for the association between StandardTerminalNameAssignment and Mapping.</p>																																																																												
type	LPDDR4-DualChannelInterfaceFunctionType , LPDDR4-DualChannelStandardTerminalNameAssignmentType , LPDDR4-DualChannelStandardTerminalMappingType , LPDDR4-DualChannelStandardTerminalNameType .																																																																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. CK_t_A</td><td>2. CK_c_A</td><td>3. CK_t_B</td><td>4. CK_c_B</td></tr> <tr><td>5. CKE_A</td><td>6. CKE_B</td><td>7. CS_A</td><td>8. CS_B</td></tr> <tr><td>9. CA0_A</td><td>10. CA1_A</td><td>11. CA2_A</td><td>12. CA3_A</td></tr> <tr><td>13. CA4_A</td><td>14. CA5_A</td><td>15. CA0_B</td><td>16. CA1_B</td></tr> <tr><td>17. CA2_B</td><td>18. CA3_B</td><td>19. CA4_B</td><td>20. CA5_B</td></tr> <tr><td>21. ODT(ca)_A</td><td>22. ODT(ca)_B</td><td>23. DQ[0]_A</td><td>24. DQ[1]_A</td></tr> <tr><td>25. DQ[2]_A</td><td>26. DQ[3]_A</td><td>27. DQ[4]_A</td><td>28. DQ[5]_A</td></tr> <tr><td>29. DQ[6]_A</td><td>30. DQ[7]_A</td><td>31. DQ[8]_A</td><td>32. DQ[9]_A</td></tr> <tr><td>33. DQ[10]_A</td><td>34. DQ[11]_A</td><td>35. DQ[12]_A</td><td>36. DQ[13]_A</td></tr> <tr><td>37. DQ[14]_A</td><td>38. DQ[15]_A</td><td>39. DQ[0]_B</td><td>40. DQ[1]_B</td></tr> <tr><td>41. DQ[2]_B</td><td>42. DQ[3]_B</td><td>43. DQ[4]_B</td><td>44. DQ[5]_B</td></tr> <tr><td>45. DQ[6]_B</td><td>46. DQ[7]_B</td><td>47. DQ[8]_B</td><td>48. DQ[9]_B</td></tr> <tr><td>49. DQ[10]_B</td><td>50. DQ[11]_B</td><td>51. DQ[12]_B</td><td>52. DQ[13]_B</td></tr> <tr><td>53. DQ[14]_B</td><td>54. DQ[15]_B</td><td>55. DQS0_t_A</td><td>56. DQS1_t_A</td></tr> <tr><td>57. DQS0_c_A</td><td>58. DQS1_c_A</td><td>59. DQS0_t_B</td><td>60. DQS1_t_B</td></tr> <tr><td>61. DQS0_c_B</td><td>62. DQS1_c_B</td><td>63. DMI0_A</td><td>64. DMI1_A</td></tr> <tr><td>65. DMI0_B</td><td>66. DMI1_B</td><td>67. ZQ</td><td>68. VDDQ</td></tr> <tr><td>69. VDD1</td><td>70. VDD2</td><td>71. VSS</td><td>72. VSSQ</td></tr> <tr><td>73. RESET_n</td><td></td><td></td><td></td></tr> </tbody> </table>	1. CK_t_A	2. CK_c_A	3. CK_t_B	4. CK_c_B	5. CKE_A	6. CKE_B	7. CS_A	8. CS_B	9. CA0_A	10. CA1_A	11. CA2_A	12. CA3_A	13. CA4_A	14. CA5_A	15. CA0_B	16. CA1_B	17. CA2_B	18. CA3_B	19. CA4_B	20. CA5_B	21. ODT(ca)_A	22. ODT(ca)_B	23. DQ[0]_A	24. DQ[1]_A	25. DQ[2]_A	26. DQ[3]_A	27. DQ[4]_A	28. DQ[5]_A	29. DQ[6]_A	30. DQ[7]_A	31. DQ[8]_A	32. DQ[9]_A	33. DQ[10]_A	34. DQ[11]_A	35. DQ[12]_A	36. DQ[13]_A	37. DQ[14]_A	38. DQ[15]_A	39. DQ[0]_B	40. DQ[1]_B	41. DQ[2]_B	42. DQ[3]_B	43. DQ[4]_B	44. DQ[5]_B	45. DQ[6]_B	46. DQ[7]_B	47. DQ[8]_B	48. DQ[9]_B	49. DQ[10]_B	50. DQ[11]_B	51. DQ[12]_B	52. DQ[13]_B	53. DQ[14]_B	54. DQ[15]_B	55. DQS0_t_A	56. DQS1_t_A	57. DQS0_c_A	58. DQS1_c_A	59. DQS0_t_B	60. DQS1_t_B	61. DQS0_c_B	62. DQS1_c_B	63. DMI0_A	64. DMI1_A	65. DMI0_B	66. DMI1_B	67. ZQ	68. VDDQ	69. VDD1	70. VDD2	71. VSS	72. VSSQ	73. RESET_n			
1. CK_t_A	2. CK_c_A	3. CK_t_B	4. CK_c_B																																																																										
5. CKE_A	6. CKE_B	7. CS_A	8. CS_B																																																																										
9. CA0_A	10. CA1_A	11. CA2_A	12. CA3_A																																																																										
13. CA4_A	14. CA5_A	15. CA0_B	16. CA1_B																																																																										
17. CA2_B	18. CA3_B	19. CA4_B	20. CA5_B																																																																										
21. ODT(ca)_A	22. ODT(ca)_B	23. DQ[0]_A	24. DQ[1]_A																																																																										
25. DQ[2]_A	26. DQ[3]_A	27. DQ[4]_A	28. DQ[5]_A																																																																										
29. DQ[6]_A	30. DQ[7]_A	31. DQ[8]_A	32. DQ[9]_A																																																																										
33. DQ[10]_A	34. DQ[11]_A	35. DQ[12]_A	36. DQ[13]_A																																																																										
37. DQ[14]_A	38. DQ[15]_A	39. DQ[0]_B	40. DQ[1]_B																																																																										
41. DQ[2]_B	42. DQ[3]_B	43. DQ[4]_B	44. DQ[5]_B																																																																										
45. DQ[6]_B	46. DQ[7]_B	47. DQ[8]_B	48. DQ[9]_B																																																																										
49. DQ[10]_B	50. DQ[11]_B	51. DQ[12]_B	52. DQ[13]_B																																																																										
53. DQ[14]_B	54. DQ[15]_B	55. DQS0_t_A	56. DQS1_t_A																																																																										
57. DQS0_c_A	58. DQS1_c_A	59. DQS0_t_B	60. DQS1_t_B																																																																										
61. DQS0_c_B	62. DQS1_c_B	63. DMI0_A	64. DMI1_A																																																																										
65. DMI0_B	66. DMI1_B	67. ZQ	68. VDDQ																																																																										
69. VDD1	70. VDD2	71. VSS	72. VSSQ																																																																										
73. RESET_n																																																																													

For more information about the LPDDR4 – Dual Channel Interface, refer to the JEDEC standard JESD209-4D.

4.7.3.10.8. DDR5 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5
diagram	
type	DDR5-InterfaceFunctionType , DDR5-x4-InterfaceFunctionType , DDR5-x8-InterfaceFunctionType , DDR5-x16-InterfaceFunctionType , DDR5DB01-InterfaceFunctionType , LPDDR5-InterfaceFunctionType , GDDR5-InterfaceFunctionType GDDR5X-InterfaceFunctionType .

4.7.3.10.8.1. DDR5-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5-x4																																								
diagram	<pre> classDiagram class DDR5-x4 { <<DDR5-x4-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR5-x4-StandardTerminalNameAssignmentType>> } class Mapping { <<DDR5-x4-StandardTerminalMappingType>> } class StandardTerminalName { <<DDR5-x4-StandardTerminalNameType>> } DDR5-x4 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for the DDR5-x4 interface. It consists of four main classes: DDR5-x4, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The DDR5-x4 class is associated with the StandardTerminalNameAssignment class via a directed association with multiplicity "1..>". The StandardTerminalNameAssignment class is associated with the Mapping class via a directed association with multiplicity "1..>". The Mapping class is associated with the StandardTerminalName class via a directed association with multiplicity "1..>". There are also dashed lines labeled "constraints" connecting the StandardTerminalNameAssignment and Mapping classes.</p>																																								
type	DDR5-x4-InterfaceFunctionType , DDR5-x4-StandardTerminalNameAssignmentType , DDR5-x4-StandardTerminalMappingType , DDR5-x4-StandardTerminalNameType .																																								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. ALERT_n</td><td>2. CA_ODT</td><td>3. CA0</td><td>4. CA1</td></tr> <tr><td>5. CA2</td><td>6. CA3</td><td>7. CA4</td><td>8. CA5</td></tr> <tr><td>9. CA6</td><td>10. CA7</td><td>11. CA8</td><td>12. CA9</td></tr> <tr><td>13. CA10</td><td>14. CA11</td><td>15. CA12</td><td>16. CA13</td></tr> <tr><td>17. CAI</td><td>18. CK_c</td><td>19. CK_t</td><td>20. CS_n</td></tr> <tr><td>21. DM_n</td><td>22. DQ[0]</td><td>23. DQ[1]</td><td>24. DQ[2]</td></tr> <tr><td>25. DQ[3]</td><td>26. DQS_c</td><td>27. DQS_t</td><td>28. LBDQ</td></tr> <tr><td>29. LBDQS</td><td>30. MIR</td><td>31. RESET_n</td><td>32. TEN</td></tr> <tr><td>33. VDD</td><td>34. VDDQ</td><td>35. VPP</td><td>36. VSS</td></tr> <tr><td>37. ZQ</td><td></td><td></td><td></td></tr> </table>	1. ALERT_n	2. CA_ODT	3. CA0	4. CA1	5. CA2	6. CA3	7. CA4	8. CA5	9. CA6	10. CA7	11. CA8	12. CA9	13. CA10	14. CA11	15. CA12	16. CA13	17. CAI	18. CK_c	19. CK_t	20. CS_n	21. DM_n	22. DQ[0]	23. DQ[1]	24. DQ[2]	25. DQ[3]	26. DQS_c	27. DQS_t	28. LBDQ	29. LBDQS	30. MIR	31. RESET_n	32. TEN	33. VDD	34. VDDQ	35. VPP	36. VSS	37. ZQ			
1. ALERT_n	2. CA_ODT	3. CA0	4. CA1																																						
5. CA2	6. CA3	7. CA4	8. CA5																																						
9. CA6	10. CA7	11. CA8	12. CA9																																						
13. CA10	14. CA11	15. CA12	16. CA13																																						
17. CAI	18. CK_c	19. CK_t	20. CS_n																																						
21. DM_n	22. DQ[0]	23. DQ[1]	24. DQ[2]																																						
25. DQ[3]	26. DQS_c	27. DQS_t	28. LBDQ																																						
29. LBDQS	30. MIR	31. RESET_n	32. TEN																																						
33. VDD	34. VDDQ	35. VPP	36. VSS																																						
37. ZQ																																									

For more information about the DDR5-x4 Interface, refer to the JEDEC standard JESD79-5A.

4.7.3.10.8.2. DDR5-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5-x8																																												
diagram	<pre> classDiagram class DDR5_x8 { <<DDR5-x8-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR5-x8-StandardTerminalNameAssignmentType>> } class Mapping { <<DDR5-x8-StandardTerminalMappingType>> } class StandardTerminalName { <<DDR5-x8-StandardTerminalNameType>> } DDR5_x8 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for the DDR5-x8 interface. It features four main classes: DDR5_x8, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The DDR5_x8 class is associated with the StandardTerminalNameAssignment class via a multiplicity of 1..>. The StandardTerminalNameAssignment class is associated with the Mapping class via a multiplicity of 1..>. The Mapping class is associated with the StandardTerminalName class via a multiplicity of 1..>. A constraint labeled 'constraints' is present between the StandardTerminalNameAssignment and Mapping classes.</p>																																												
type	DDR5-x8-InterfaceFunctionType , DDR5-x8-StandardTerminalNameAssignmentType , DDR5-x8-StandardTerminalMappingType , DDR5-x8-StandardTerminalNameType .																																												
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. ALERT_n</td><td>2. CA_ODT</td><td>3. CA0</td><td>4. CA1</td></tr> <tr><td>5. CA2</td><td>6. CA3</td><td>7. CA4</td><td>8. CA5</td></tr> <tr><td>9. CA6</td><td>10. CA7</td><td>11. CA8</td><td>12. CA9</td></tr> <tr><td>13. CA10</td><td>14. CA11</td><td>15. CA12</td><td>16. CA13</td></tr> <tr><td>17. CAI</td><td>18. CK_c</td><td>19. CK_t</td><td>20. CS_n</td></tr> <tr><td>21. DM_n</td><td>22. DQ[0]</td><td>23. DQ[1]</td><td>24. DQ[2]</td></tr> <tr><td>25. DQ[3]</td><td>26. DQ[4]</td><td>27. DQ[5]</td><td>28. DQ[6]</td></tr> <tr><td>29. DQ[7]</td><td>30. DQS_c</td><td>31. DQS_t</td><td>32. LBDQ</td></tr> <tr><td>33. LBDQS</td><td>34. MIR</td><td>35. RESET_n</td><td>36. TDQS_c</td></tr> <tr><td>37. TDQS_t</td><td>38. TEN</td><td>39. VDD</td><td>40. VDDQ</td></tr> <tr><td>41. VPP</td><td>42. VSS</td><td>43. ZQ</td><td></td></tr> </table>	1. ALERT_n	2. CA_ODT	3. CA0	4. CA1	5. CA2	6. CA3	7. CA4	8. CA5	9. CA6	10. CA7	11. CA8	12. CA9	13. CA10	14. CA11	15. CA12	16. CA13	17. CAI	18. CK_c	19. CK_t	20. CS_n	21. DM_n	22. DQ[0]	23. DQ[1]	24. DQ[2]	25. DQ[3]	26. DQ[4]	27. DQ[5]	28. DQ[6]	29. DQ[7]	30. DQS_c	31. DQS_t	32. LBDQ	33. LBDQS	34. MIR	35. RESET_n	36. TDQS_c	37. TDQS_t	38. TEN	39. VDD	40. VDDQ	41. VPP	42. VSS	43. ZQ	
1. ALERT_n	2. CA_ODT	3. CA0	4. CA1																																										
5. CA2	6. CA3	7. CA4	8. CA5																																										
9. CA6	10. CA7	11. CA8	12. CA9																																										
13. CA10	14. CA11	15. CA12	16. CA13																																										
17. CAI	18. CK_c	19. CK_t	20. CS_n																																										
21. DM_n	22. DQ[0]	23. DQ[1]	24. DQ[2]																																										
25. DQ[3]	26. DQ[4]	27. DQ[5]	28. DQ[6]																																										
29. DQ[7]	30. DQS_c	31. DQS_t	32. LBDQ																																										
33. LBDQS	34. MIR	35. RESET_n	36. TDQS_c																																										
37. TDQS_t	38. TEN	39. VDD	40. VDDQ																																										
41. VPP	42. VSS	43. ZQ																																											

For more information about the DDR5-x8 Interface, refer to the JEDEC standard JESD79-5A.

4.7.3.10.8.3. DDR5-x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5-x16																																																				
diagram	<pre> classDiagram class DDR5-x16 { <<DDR5-x16-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR5-x16-StandardTerminalNameAssignmentType>> } class Mapping { <<DDR5-x16-StandardTerminalMappingType>> } class StandardTerminalName { <<DDR5-x16-StandardTerminalNameType>> } DDR5-x16 "1..x" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..x" Mapping Mapping "1..x" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for the DDR5-x16 interface. It features four main classes: DDR5-x16, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The DDR5-x16 class is associated with the StandardTerminalNameAssignment class via a multiplicity of 1..x. The StandardTerminalNameAssignment class is associated with the Mapping class via a multiplicity of 1..x. The Mapping class is associated with the StandardTerminalName class via a multiplicity of 1..x. There is also a dashed association labeled 'constraints' between the StandardTerminalNameAssignment and Mapping classes.</p>																																																				
type	DDR5-x16-InterfaceFunctionType , DDR5-x16-StandardTerminalNameAssignmentType , DDR5-x16-StandardTerminalMappingType , DDR5-x16-StandardTerminalNameType .																																																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. ALERT_n</td><td>2. CA_ODT</td><td>3. CA0</td><td>4. CA1</td></tr> <tr><td>5. CA2</td><td>6. CA3</td><td>7. CA4</td><td>8. CA5</td></tr> <tr><td>9. CA6</td><td>10. CA7</td><td>11. CA8</td><td>12. CA9</td></tr> <tr><td>13. CA10</td><td>14. CA11</td><td>15. CA12</td><td>16. CA13</td></tr> <tr><td>17. CAI</td><td>18. CK_c</td><td>19. CK_t</td><td>20. CS_n</td></tr> <tr><td>21. DML_n</td><td>22. DMU_n</td><td>23. DQL[0]</td><td>24. DQL[1]</td></tr> <tr><td>25. DQL[2]</td><td>26. DQL[3]</td><td>27. DQL[4]</td><td>28. DQL[5]</td></tr> <tr><td>29. DQL[6]</td><td>30. DQL[7]</td><td>31. DQSL_c</td><td>32. DQSL_t</td></tr> <tr><td>33. DQSU_c</td><td>34. DQSU_t</td><td>35. DQU[0]</td><td>36. DQU[1]</td></tr> <tr><td>37. DQU[2]</td><td>38. DQU[3]</td><td>39. DQU[4]</td><td>40. DQU[5]</td></tr> <tr><td>41. DQU[6]</td><td>42. DQU[7]</td><td>43. LBDQ</td><td>44. LBDQS</td></tr> <tr><td>45. MIR</td><td>46. RESET_n</td><td>47. TEN</td><td>48. VDD</td></tr> <tr><td>49. VDDQ</td><td>50. VPP</td><td>51. VSS</td><td>52. ZQ</td></tr> </tbody> </table>	1. ALERT_n	2. CA_ODT	3. CA0	4. CA1	5. CA2	6. CA3	7. CA4	8. CA5	9. CA6	10. CA7	11. CA8	12. CA9	13. CA10	14. CA11	15. CA12	16. CA13	17. CAI	18. CK_c	19. CK_t	20. CS_n	21. DML_n	22. DMU_n	23. DQL[0]	24. DQL[1]	25. DQL[2]	26. DQL[3]	27. DQL[4]	28. DQL[5]	29. DQL[6]	30. DQL[7]	31. DQSL_c	32. DQSL_t	33. DQSU_c	34. DQSU_t	35. DQU[0]	36. DQU[1]	37. DQU[2]	38. DQU[3]	39. DQU[4]	40. DQU[5]	41. DQU[6]	42. DQU[7]	43. LBDQ	44. LBDQS	45. MIR	46. RESET_n	47. TEN	48. VDD	49. VDDQ	50. VPP	51. VSS	52. ZQ
1. ALERT_n	2. CA_ODT	3. CA0	4. CA1																																																		
5. CA2	6. CA3	7. CA4	8. CA5																																																		
9. CA6	10. CA7	11. CA8	12. CA9																																																		
13. CA10	14. CA11	15. CA12	16. CA13																																																		
17. CAI	18. CK_c	19. CK_t	20. CS_n																																																		
21. DML_n	22. DMU_n	23. DQL[0]	24. DQL[1]																																																		
25. DQL[2]	26. DQL[3]	27. DQL[4]	28. DQL[5]																																																		
29. DQL[6]	30. DQL[7]	31. DQSL_c	32. DQSL_t																																																		
33. DQSU_c	34. DQSU_t	35. DQU[0]	36. DQU[1]																																																		
37. DQU[2]	38. DQU[3]	39. DQU[4]	40. DQU[5]																																																		
41. DQU[6]	42. DQU[7]	43. LBDQ	44. LBDQS																																																		
45. MIR	46. RESET_n	47. TEN	48. VDD																																																		
49. VDDQ	50. VPP	51. VSS	52. ZQ																																																		

For more information about the DDR5-x16 Interface, refer to the JEDEC standard JESD79-5A.

4.7.3.10.8.4. DDR5DB01

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5DB01																																				
diagram	<pre> classDiagram class DDR5DB01 { <<DDR5DB01-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DDR5DB01-StandardTerminalNameAssignmentType>> } class Mapping { <<DDR5DB01-StandardTerminalMappingType>> } class StandardTerminalName { <<DDR5DB01-StandardTerminalNameType>> } DDR5DB01 "1..oo" -- "1..oo" StandardTerminalNameAssignment : <<DDR5DB01-InterfaceFunctionType>> StandardTerminalNameAssignment "1..oo" -- "36" Mapping : <<DDR5DB01-StandardTerminalMappingType>> Mapping "36" -- "36" StandardTerminalName : <<DDR5DB01-StandardTerminalNameType>> </pre>																																				
type	DDR5DB01-InterfaceFunctionType , DDR5DB01-StandardTerminalNameAssignmentType , DDR5DB01-StandardTerminalMappingType , DDR5DB01-StandardTerminalNameType .																																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr><td>1. BCS_n</td><td>2. BRST_n</td><td>3. BCOM0</td><td>4. BCOM1</td></tr> <tr><td>5. BCOM2</td><td>6. BCK_t</td><td>7. BCK_c</td><td>8. DQ[0]</td></tr> <tr><td>9. DQ[1]</td><td>10. DQ[2]</td><td>11. DQ[3]</td><td>12. DQ[4]</td></tr> <tr><td>13. DQ[5]</td><td>14. DQ[6]</td><td>15. DQ[7]</td><td>16. DQS0_t</td></tr> <tr><td>17. DQS0_c</td><td>18. DQS1_t</td><td>19. DQS1_c</td><td>20. MDQ0</td></tr> <tr><td>21. MDQ1</td><td>22. MDQ2</td><td>23. MDQ3</td><td>24. MDQ4</td></tr> <tr><td>25. MDQ5</td><td>26. MDQ6</td><td>27. MDQ7</td><td>28. MDQS0_t</td></tr> <tr><td>29. MDQS0_c</td><td>30. MDQS1_t</td><td>31. MDQS1_c</td><td>32. LBTXDQ</td></tr> <tr><td>33. LBTXDQS</td><td>34. VDD</td><td>35. VSS</td><td>36. ZQCAL</td></tr> </table>	1. BCS_n	2. BRST_n	3. BCOM0	4. BCOM1	5. BCOM2	6. BCK_t	7. BCK_c	8. DQ[0]	9. DQ[1]	10. DQ[2]	11. DQ[3]	12. DQ[4]	13. DQ[5]	14. DQ[6]	15. DQ[7]	16. DQS0_t	17. DQS0_c	18. DQS1_t	19. DQS1_c	20. MDQ0	21. MDQ1	22. MDQ2	23. MDQ3	24. MDQ4	25. MDQ5	26. MDQ6	27. MDQ7	28. MDQS0_t	29. MDQS0_c	30. MDQS1_t	31. MDQS1_c	32. LBTXDQ	33. LBTXDQS	34. VDD	35. VSS	36. ZQCAL
1. BCS_n	2. BRST_n	3. BCOM0	4. BCOM1																																		
5. BCOM2	6. BCK_t	7. BCK_c	8. DQ[0]																																		
9. DQ[1]	10. DQ[2]	11. DQ[3]	12. DQ[4]																																		
13. DQ[5]	14. DQ[6]	15. DQ[7]	16. DQS0_t																																		
17. DQS0_c	18. DQS1_t	19. DQS1_c	20. MDQ0																																		
21. MDQ1	22. MDQ2	23. MDQ3	24. MDQ4																																		
25. MDQ5	26. MDQ6	27. MDQ7	28. MDQS0_t																																		
29. MDQS0_c	30. MDQS1_t	31. MDQS1_c	32. LBTXDQ																																		
33. LBTXDQS	34. VDD	35. VSS	36. ZQCAL																																		

For more information about the DDR5DB01 Interface, refer to the JEDEC standard JESD82-521.

4.7.3.10.8.5. LPDDR5

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/LPDDR5																																												
diagram	<pre> classDiagram class LPDDR5 { <<LPDDR5-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<LPDDR5-StandardTerminalNameAssignmentType>> } class Mapping { <<LPDDR5-StandardTerminalMappingType>> } class StandardTerminalName { <<LPDDR5-StandardTerminalNameType>> } LPDDR5 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>																																												
type	LDDR5-InterfaceFunctionType, LDDR5-StandardTerminalNameAssignmentType, LDDR5-StandardTerminalMappingType, LDDR5-StandardTerminalNameType.																																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. CK_t</td><td>2. CK_c</td><td>3. CS</td><td>4. CA0</td></tr> <tr><td>5. CA1</td><td>6. CA2</td><td>7. CA3</td><td>8. CA4</td></tr> <tr><td>9. CA5</td><td>10. CA6</td><td>11. DQ[0]</td><td>12. DQ[1]</td></tr> <tr><td>13. DQ[2]</td><td>14. DQ[3]</td><td>15. DQ[4]</td><td>16. DQ[5]</td></tr> <tr><td>17. DQ[6]</td><td>18. DQ[7]</td><td>19. DQ[8]</td><td>20. DQ[9]</td></tr> <tr><td>21. DQ[10]</td><td>22. DQ[11]</td><td>23. DQ[12]</td><td>24. DQ[13]</td></tr> <tr><td>25. DQ[14]</td><td>26. DQ[15]</td><td>27. WCK0_t</td><td>28. WCK0_c</td></tr> <tr><td>29. WCK1_t</td><td>30. WCK1_c</td><td>31. RDQS0_t</td><td>32. RDQS0_c</td></tr> <tr><td>33. RDQS1_t</td><td>34. RDQS1_c</td><td>35. DMIO</td><td>36. DMI1</td></tr> <tr><td>37. ZQ</td><td>38. VDDQ</td><td>39. VDD1</td><td>40. VDD2H</td></tr> <tr><td>41. VDD2L</td><td>42. VSS</td><td>43. RESET_n</td><td></td></tr> </tbody> </table>	1. CK_t	2. CK_c	3. CS	4. CA0	5. CA1	6. CA2	7. CA3	8. CA4	9. CA5	10. CA6	11. DQ[0]	12. DQ[1]	13. DQ[2]	14. DQ[3]	15. DQ[4]	16. DQ[5]	17. DQ[6]	18. DQ[7]	19. DQ[8]	20. DQ[9]	21. DQ[10]	22. DQ[11]	23. DQ[12]	24. DQ[13]	25. DQ[14]	26. DQ[15]	27. WCK0_t	28. WCK0_c	29. WCK1_t	30. WCK1_c	31. RDQS0_t	32. RDQS0_c	33. RDQS1_t	34. RDQS1_c	35. DMIO	36. DMI1	37. ZQ	38. VDDQ	39. VDD1	40. VDD2H	41. VDD2L	42. VSS	43. RESET_n	
1. CK_t	2. CK_c	3. CS	4. CA0																																										
5. CA1	6. CA2	7. CA3	8. CA4																																										
9. CA5	10. CA6	11. DQ[0]	12. DQ[1]																																										
13. DQ[2]	14. DQ[3]	15. DQ[4]	16. DQ[5]																																										
17. DQ[6]	18. DQ[7]	19. DQ[8]	20. DQ[9]																																										
21. DQ[10]	22. DQ[11]	23. DQ[12]	24. DQ[13]																																										
25. DQ[14]	26. DQ[15]	27. WCK0_t	28. WCK0_c																																										
29. WCK1_t	30. WCK1_c	31. RDQS0_t	32. RDQS0_c																																										
33. RDQS1_t	34. RDQS1_c	35. DMIO	36. DMI1																																										
37. ZQ	38. VDDQ	39. VDD1	40. VDD2H																																										
41. VDD2L	42. VSS	43. RESET_n																																											

For more information about the LPDDR5 Interface, refer to the JEDEC standard JESD209-5B.

4.7.3.10.8.6. GDDR5

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/GDDR5																																																																																				
diagram	<pre> classDiagram class GDDR5 { <<GDDR5-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<GDDR5-StandardTerminalNameAssignmentType>> } class Mapping { <<GDDR5-StandardTerminalMappingType>> } class StandardTerminalName { <<GDDR5-StandardTerminalNameType>> } GDDR5 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the GDDR5 interface structure. It consists of four main classes: GDDR5-InterfaceFunctionType, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The GDDR5 class has a multiplicity of 1..> to StandardTerminalNameAssignment. The StandardTerminalNameAssignment class has a multiplicity of 1..> to Mapping. The Mapping class has a multiplicity of 1..> to StandardTerminalName. There are also dashed boxes labeled 'constraints' and 'GDDR5-StandardTerminalMappingType'.</p>																																																																																				
type	GDDR5-InterfaceFunctionType , GDDR5-StandardTerminalNameAssignmentType , GDDR5-StandardTerminalMappingType , GDDR5-StandardTerminalNameType .																																																																																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. A0</td><td>2. A1</td><td>3. A2</td><td>4. A3</td></tr> <tr><td>5. A4</td><td>6. A5</td><td>7. A6</td><td>8. A7</td></tr> <tr><td>9. A8</td><td>10. A9</td><td>11. A10</td><td>12. A11</td></tr> <tr><td>13. A12</td><td>14. A13</td><td>15. ABI_n</td><td>16. BA0</td></tr> <tr><td>17. BA1</td><td>18. BA2</td><td>19. BA3</td><td>20. CAS_n</td></tr> <tr><td>21. CK_c</td><td>22. CK_t</td><td>23. CKE_n</td><td>24. CS_n</td></tr> <tr><td>25. DBI0_n</td><td>26. DBI1_n</td><td>27. DBI2_n</td><td>28. DBI3_n</td></tr> <tr><td>29. DQ[0]</td><td>30. DQ[1]</td><td>31. DQ[2]</td><td>32. DQ[3]</td></tr> <tr><td>33. DQ[4]</td><td>34. DQ[5]</td><td>35. DQ[6]</td><td>36. DQ[7]</td></tr> <tr><td>37. DQ[8]</td><td>38. DQ[9]</td><td>39. DQ[10]</td><td>40. DQ[11]</td></tr> <tr><td>41. DQ[12]</td><td>42. DQ[13]</td><td>43. DQ[14]</td><td>44. DQ[15]</td></tr> <tr><td>45. DQ[16]</td><td>46. DQ[17]</td><td>47. DQ[18]</td><td>48. DQ[19]</td></tr> <tr><td>49. DQ[20]</td><td>50. DQ[21]</td><td>51. DQ[22]</td><td>52. DQ[23]</td></tr> <tr><td>53. DQ[24]</td><td>54. DQ[25]</td><td>55. DQ[26]</td><td>56. DQ[27]</td></tr> <tr><td>57. DQ[28]</td><td>58. DQ[29]</td><td>59. DQ[30]</td><td>60. DQ[31]</td></tr> <tr><td>61. EDC0</td><td>62. EDC1</td><td>63. EDC2</td><td>64. EDC3</td></tr> <tr><td>65. MF</td><td>66. RAS_n</td><td>67. RESET_n</td><td>68. SEN</td></tr> <tr><td>69. VDD</td><td>70. VDDQ</td><td>71. VPP</td><td>72. VREFC</td></tr> <tr><td>73. VREFD</td><td>74. VSS</td><td>75. VSSQ</td><td>76. WCK01_c</td></tr> <tr><td>77. WCK01_t</td><td>78. WCK23_c</td><td>79. WCK23_t</td><td>80. WE_n</td></tr> <tr><td></td><td></td><td></td><td>81. ZQ</td></tr> </tbody> </table>	1. A0	2. A1	3. A2	4. A3	5. A4	6. A5	7. A6	8. A7	9. A8	10. A9	11. A10	12. A11	13. A12	14. A13	15. ABI_n	16. BA0	17. BA1	18. BA2	19. BA3	20. CAS_n	21. CK_c	22. CK_t	23. CKE_n	24. CS_n	25. DBI0_n	26. DBI1_n	27. DBI2_n	28. DBI3_n	29. DQ[0]	30. DQ[1]	31. DQ[2]	32. DQ[3]	33. DQ[4]	34. DQ[5]	35. DQ[6]	36. DQ[7]	37. DQ[8]	38. DQ[9]	39. DQ[10]	40. DQ[11]	41. DQ[12]	42. DQ[13]	43. DQ[14]	44. DQ[15]	45. DQ[16]	46. DQ[17]	47. DQ[18]	48. DQ[19]	49. DQ[20]	50. DQ[21]	51. DQ[22]	52. DQ[23]	53. DQ[24]	54. DQ[25]	55. DQ[26]	56. DQ[27]	57. DQ[28]	58. DQ[29]	59. DQ[30]	60. DQ[31]	61. EDC0	62. EDC1	63. EDC2	64. EDC3	65. MF	66. RAS_n	67. RESET_n	68. SEN	69. VDD	70. VDDQ	71. VPP	72. VREFC	73. VREFD	74. VSS	75. VSSQ	76. WCK01_c	77. WCK01_t	78. WCK23_c	79. WCK23_t	80. WE_n				81. ZQ
1. A0	2. A1	3. A2	4. A3																																																																																		
5. A4	6. A5	7. A6	8. A7																																																																																		
9. A8	10. A9	11. A10	12. A11																																																																																		
13. A12	14. A13	15. ABI_n	16. BA0																																																																																		
17. BA1	18. BA2	19. BA3	20. CAS_n																																																																																		
21. CK_c	22. CK_t	23. CKE_n	24. CS_n																																																																																		
25. DBI0_n	26. DBI1_n	27. DBI2_n	28. DBI3_n																																																																																		
29. DQ[0]	30. DQ[1]	31. DQ[2]	32. DQ[3]																																																																																		
33. DQ[4]	34. DQ[5]	35. DQ[6]	36. DQ[7]																																																																																		
37. DQ[8]	38. DQ[9]	39. DQ[10]	40. DQ[11]																																																																																		
41. DQ[12]	42. DQ[13]	43. DQ[14]	44. DQ[15]																																																																																		
45. DQ[16]	46. DQ[17]	47. DQ[18]	48. DQ[19]																																																																																		
49. DQ[20]	50. DQ[21]	51. DQ[22]	52. DQ[23]																																																																																		
53. DQ[24]	54. DQ[25]	55. DQ[26]	56. DQ[27]																																																																																		
57. DQ[28]	58. DQ[29]	59. DQ[30]	60. DQ[31]																																																																																		
61. EDC0	62. EDC1	63. EDC2	64. EDC3																																																																																		
65. MF	66. RAS_n	67. RESET_n	68. SEN																																																																																		
69. VDD	70. VDDQ	71. VPP	72. VREFC																																																																																		
73. VREFD	74. VSS	75. VSSQ	76. WCK01_c																																																																																		
77. WCK01_t	78. WCK23_c	79. WCK23_t	80. WE_n																																																																																		
			81. ZQ																																																																																		

For more information about the GDDR5 Interface, refer to the JEDEC standard JESD212C.

4.7.3.10.8.7. GDDR5X

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/GDDR5X																																																																																				
diagram	<pre> classDiagram class GDDR5X { <<GDDR5X-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<GDDR5X-StandardTerminalNameAssignmentType>> } class Mapping { <<GDDR5X-StandardTerminalMappingType>> } class StandardTerminalName { <<GDDR5X-StandardTerminalNameType>> } GDDR5X "1..*" --> "84" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..*" --> "84" Mapping Mapping "1..*" --> "84" StandardTerminalName </pre>																																																																																				
type	GDDR5X-InterfaceFunctionType , GDDR5X-StandardTerminalNameAssignmentType , GDDR5X-StandardTerminalMappingType , GDDR5X-StandardTerminalNameType .																																																																																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. A0</td><td>2. A1</td><td>3. A2</td><td>4. A3</td></tr> <tr><td>5. A4</td><td>6. A5</td><td>7. A6</td><td>8. A7</td></tr> <tr><td>9. A8</td><td>10. A9</td><td>11. A10</td><td>12. A11</td></tr> <tr><td>13. A12</td><td>14. A13</td><td>15. A14</td><td>16. A15</td></tr> <tr><td>17. ABI_n</td><td>18. BA0</td><td>19. BA1</td><td>20. BA2</td></tr> <tr><td>21. BA3</td><td>22. CAS_n</td><td>23. CK_c</td><td>24. CK_t</td></tr> <tr><td>25. CKE_n</td><td>26. DBI0_n</td><td>27. DBI1_n</td><td>28. DBI2_n</td></tr> <tr><td>29. DBI3_n</td><td>30. DQ[0]</td><td>31. DQ[1]</td><td>32. DQ[2]</td></tr> <tr><td>33. DQ[3]</td><td>34. DQ[4]</td><td>35. DQ[5]</td><td>36. DQ[6]</td></tr> <tr><td>37. DQ[7]</td><td>38. DQ[8]</td><td>39. DQ[9]</td><td>40. DQ[10]</td></tr> <tr><td>41. DQ[11]</td><td>42. DQ[12]</td><td>43. DQ[13]</td><td>44. DQ[14]</td></tr> <tr><td>45. DQ[15]</td><td>46. DQ[16]</td><td>47. DQ[17]</td><td>48. DQ[18]</td></tr> <tr><td>49. DQ[19]</td><td>50. DQ[20]</td><td>51. DQ[21]</td><td>52. DQ[22]</td></tr> <tr><td>53. DQ[23]</td><td>54. DQ[24]</td><td>55. DQ[25]</td><td>56. DQ[26]</td></tr> <tr><td>57. DQ[27]</td><td>58. DQ[28]</td><td>59. DQ[29]</td><td>60. DQ[30]</td></tr> <tr><td>61. DQ[31]</td><td>62. EDC0</td><td>63. EDC1</td><td>64. EDC2</td></tr> <tr><td>65. EDC3</td><td>66. MF</td><td>67. RAS_n</td><td>68. RESET_n</td></tr> <tr><td>69. TCK</td><td>70. TDI</td><td>71. TDO</td><td>72. TMS</td></tr> <tr><td>73. VDD</td><td>74. VDDQ</td><td>75. VPP</td><td>76. VREFC</td></tr> <tr><td>77. VSS</td><td>78. VSSQ</td><td>79. WCK01_c</td><td>80. WCK01_t</td></tr> <tr><td>81. WCK23_c</td><td>82. WCK23_t</td><td>83. WE_n</td><td>84. ZQ</td></tr> </tbody> </table>	1. A0	2. A1	3. A2	4. A3	5. A4	6. A5	7. A6	8. A7	9. A8	10. A9	11. A10	12. A11	13. A12	14. A13	15. A14	16. A15	17. ABI_n	18. BA0	19. BA1	20. BA2	21. BA3	22. CAS_n	23. CK_c	24. CK_t	25. CKE_n	26. DBI0_n	27. DBI1_n	28. DBI2_n	29. DBI3_n	30. DQ[0]	31. DQ[1]	32. DQ[2]	33. DQ[3]	34. DQ[4]	35. DQ[5]	36. DQ[6]	37. DQ[7]	38. DQ[8]	39. DQ[9]	40. DQ[10]	41. DQ[11]	42. DQ[12]	43. DQ[13]	44. DQ[14]	45. DQ[15]	46. DQ[16]	47. DQ[17]	48. DQ[18]	49. DQ[19]	50. DQ[20]	51. DQ[21]	52. DQ[22]	53. DQ[23]	54. DQ[24]	55. DQ[25]	56. DQ[26]	57. DQ[27]	58. DQ[28]	59. DQ[29]	60. DQ[30]	61. DQ[31]	62. EDC0	63. EDC1	64. EDC2	65. EDC3	66. MF	67. RAS_n	68. RESET_n	69. TCK	70. TDI	71. TDO	72. TMS	73. VDD	74. VDDQ	75. VPP	76. VREFC	77. VSS	78. VSSQ	79. WCK01_c	80. WCK01_t	81. WCK23_c	82. WCK23_t	83. WE_n	84. ZQ
1. A0	2. A1	3. A2	4. A3																																																																																		
5. A4	6. A5	7. A6	8. A7																																																																																		
9. A8	10. A9	11. A10	12. A11																																																																																		
13. A12	14. A13	15. A14	16. A15																																																																																		
17. ABI_n	18. BA0	19. BA1	20. BA2																																																																																		
21. BA3	22. CAS_n	23. CK_c	24. CK_t																																																																																		
25. CKE_n	26. DBI0_n	27. DBI1_n	28. DBI2_n																																																																																		
29. DBI3_n	30. DQ[0]	31. DQ[1]	32. DQ[2]																																																																																		
33. DQ[3]	34. DQ[4]	35. DQ[5]	36. DQ[6]																																																																																		
37. DQ[7]	38. DQ[8]	39. DQ[9]	40. DQ[10]																																																																																		
41. DQ[11]	42. DQ[12]	43. DQ[13]	44. DQ[14]																																																																																		
45. DQ[15]	46. DQ[16]	47. DQ[17]	48. DQ[18]																																																																																		
49. DQ[19]	50. DQ[20]	51. DQ[21]	52. DQ[22]																																																																																		
53. DQ[23]	54. DQ[24]	55. DQ[25]	56. DQ[26]																																																																																		
57. DQ[27]	58. DQ[28]	59. DQ[29]	60. DQ[30]																																																																																		
61. DQ[31]	62. EDC0	63. EDC1	64. EDC2																																																																																		
65. EDC3	66. MF	67. RAS_n	68. RESET_n																																																																																		
69. TCK	70. TDI	71. TDO	72. TMS																																																																																		
73. VDD	74. VDDQ	75. VPP	76. VREFC																																																																																		
77. VSS	78. VSSQ	79. WCK01_c	80. WCK01_t																																																																																		
81. WCK23_c	82. WCK23_t	83. WE_n	84. ZQ																																																																																		

For more information about the GDDR5X Interface, refer to the JEDEC standard JESD232A.

4.7.3.10.9. DDR6 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5
diagram	<pre> graph LR DDR6[DDR6
type DDR6-InterfaceFunctionType] <--> GDDR6[GDDR6
type GDDR6-InterfaceFunctionType] </pre>
type	DDR6-InterfaceFunctionType , GDDR6-InterfaceFunctionType

4.7.3.10.9.1. GDDR6

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/LPDDR4-DualChannel																																																								
diagram	<pre> graph TD DDR6[DDR6
type DDR6-InterfaceFunctionType] --> GDDR6[GDDR6
type GDDR6-InterfaceFunctionType] GDDR6 --> STA[StandardTerminalNameAssignment
type GDDR6-StandardTerminalNameAssignmentType] STA --> M[Mapping
type GDDR6-StandardTerminalMappingType] M --> STN[GDDR6-StandardTerminalNameType
constraints] </pre>																																																								
type	GDDR6-InterfaceFunctionType , GDDR6-StandardTerminalNameAssignmentType , GDDR6-StandardTerminalMappingType , GDDR6-StandardTerminalNameType .																																																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. CA0</td><td>2. CA1</td><td>3. CA2</td><td>4. CA3</td></tr> <tr><td>5. CA4</td><td>6. CA5</td><td>7. CA6</td><td>8. CA7</td></tr> <tr><td>9. CA8</td><td>10. CA9</td><td>11. CA10</td><td>12. CABI_n</td></tr> <tr><td>13. CK_c</td><td>14. CK_t</td><td>15. CKE_n</td><td>16. DBI0_n</td></tr> <tr><td>17. DBI1_n</td><td>18. DQ[0]</td><td>19. DQ[1]</td><td>20. DQ[2]</td></tr> <tr><td>21. DQ[3]</td><td>22. DQ[4]</td><td>23. DQ[5]</td><td>24. DQ[6]</td></tr> <tr><td>25. DQ[7]</td><td>26. DQ[8]</td><td>27. DQ[9]</td><td>28. DQ[10]</td></tr> <tr><td>29. DQ[11]</td><td>30. DQ[12]</td><td>31. DQ[13]</td><td>32. DQ[14]</td></tr> <tr><td>33. DQ[15]</td><td>34. EDC0</td><td>35. EDC1</td><td>36. RESET_n</td></tr> <tr><td>37. TCK</td><td>38. TDI</td><td>39. TDO</td><td>40. TMS</td></tr> <tr><td>41. VDD</td><td>42. VDDQ</td><td>43. VPP</td><td>44. VREFC</td></tr> <tr><td>45. VSS</td><td>46. WCK0_c</td><td>47. WCK0_t</td><td>48. WCK1_c</td></tr> <tr><td>49. WCK1_t</td><td>50. ZQ</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. CA0	2. CA1	3. CA2	4. CA3	5. CA4	6. CA5	7. CA6	8. CA7	9. CA8	10. CA9	11. CA10	12. CABI_n	13. CK_c	14. CK_t	15. CKE_n	16. DBI0_n	17. DBI1_n	18. DQ[0]	19. DQ[1]	20. DQ[2]	21. DQ[3]	22. DQ[4]	23. DQ[5]	24. DQ[6]	25. DQ[7]	26. DQ[8]	27. DQ[9]	28. DQ[10]	29. DQ[11]	30. DQ[12]	31. DQ[13]	32. DQ[14]	33. DQ[15]	34. EDC0	35. EDC1	36. RESET_n	37. TCK	38. TDI	39. TDO	40. TMS	41. VDD	42. VDDQ	43. VPP	44. VREFC	45. VSS	46. WCK0_c	47. WCK0_t	48. WCK1_c	49. WCK1_t	50. ZQ		
Mapping/StandardTerminalName																																																									
1. CA0	2. CA1	3. CA2	4. CA3																																																						
5. CA4	6. CA5	7. CA6	8. CA7																																																						
9. CA8	10. CA9	11. CA10	12. CABI_n																																																						
13. CK_c	14. CK_t	15. CKE_n	16. DBI0_n																																																						
17. DBI1_n	18. DQ[0]	19. DQ[1]	20. DQ[2]																																																						
21. DQ[3]	22. DQ[4]	23. DQ[5]	24. DQ[6]																																																						
25. DQ[7]	26. DQ[8]	27. DQ[9]	28. DQ[10]																																																						
29. DQ[11]	30. DQ[12]	31. DQ[13]	32. DQ[14]																																																						
33. DQ[15]	34. EDC0	35. EDC1	36. RESET_n																																																						
37. TCK	38. TDI	39. TDO	40. TMS																																																						
41. VDD	42. VDDQ	43. VPP	44. VREFC																																																						
45. VSS	46. WCK0_c	47. WCK0_t	48. WCK1_c																																																						
49. WCK1_t	50. ZQ																																																								

For more information about the GDDR6 Interface, refer to the JEDEC standard JESD250C.

4.7.3.10.10. DigRF3G Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DigRF3G												
diagram	<pre> classDiagram class DigRF3G { <<DigRF3G-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DigRF3G-StandardTerminalNameAssignmentType>> } class Mapping { <<DigRF3G-StandardTerminalMappingType>> } class StandardTerminalName { <<DigRF3G-StandardTerminalNameType>> } DigRF3G "1..x" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "6" --> Mapping Mapping "6" --> StandardTerminalName </pre>												
type	DigRF3G-InterfaceFunctionType , DigRF3G-StandardTerminalNameAssignmentType , DigRF3G-StandardTerminalMappingType , DigRF3G-StandardTerminalNameType .												
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. SysClk</td> <td>2. SysClkEn</td> <td>3. TxDataP</td> <td>4. TxDataN</td> </tr> <tr> <td>5. RxDataP</td> <td>6. RxDataN</td> <td></td> <td></td> </tr> </table>					1. SysClk	2. SysClkEn	3. TxDataP	4. TxDataN	5. RxDataP	6. RxDataN		
1. SysClk	2. SysClkEn	3. TxDataP	4. TxDataN										
5. RxDataP	6. RxDataN												

For more information about the DigRF3G Interface, refer to the MIPI Alliance standard Specification for Dual Mode 2.5G/3G Baseband/RFIC Interface Version 3.09.06.

4.7.3.10.11. DigRFv4 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DigRFv4												
diagram	<pre> classDiagram class DigRFv4 { <<DigRFv4-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<DigRFv4-StandardTerminalNameAssignmentType>> } class Mapping { <<DigRFv4-StandardTerminalMappingType>> } class StandardTerminalName { <<DigRFv4-StandardTerminalNameType>> } DigRFv4 "1..x" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "7" --> Mapping Mapping "7" --> StandardTerminalName </pre>												
type	DigRFv4-InterfaceFunctionType , DigRFv4-StandardTerminalNameAssignmentType , DigRFv4-StandardTerminalMappingType , DigRFv4-StandardTerminalNameType .												
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. TxDataP</td> <td>2. TxDataN</td> <td>3. RxDataP</td> <td>4. RxDataN</td> </tr> <tr> <td>5. DigRFEN</td> <td>6. RefClkEn</td> <td>7. RefClk</td> <td></td> </tr> </table>					1. TxDataP	2. TxDataN	3. RxDataP	4. RxDataN	5. DigRFEN	6. RefClkEn	7. RefClk	
1. TxDataP	2. TxDataN	3. RxDataP	4. RxDataN										
5. DigRFEN	6. RefClkEn	7. RefClk											

For more information about the DigRFv4 Interface, refer to the MIPI Alliance standard Specification for DigiRFv4 Version 1.2.

4.7.3.10.12. EE1002-SPD-EEPROM

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/EE1002-SPD-EEPROM								
diagram	<pre> classDiagram class EE1002-SPD-EEPROM { <<type>> EE1002-SPD-EEPROM-InterfaceFunctionType } class StandardTerminalNameAssignment { <<type>> EE1002-SPD-EEPROM-StandardTerminalNameAssignmentType } class Mapping { <<type>> EE1002-SPD-EEPROM-StandardTerminalMappingType } class StandardTerminalName { <<type>> EE1002-SPD-EEPROM-StandardTerminalNameType } EE1002-SPD-EEPROM "1..oo" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "8" --> Mapping Mapping --> StandardTerminalName </pre>								
type	EE1002-SPD-EEPROM-InterfaceFunctionType , EE1002-SPD-EEPROM-StandardTerminalNameAssignmentType , EE1002-SPD-EEPROM-StandardTerminalMappingType , EE1002-SPD-EEPROM-StandardTerminalNameType .								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. SA0</td> <td>2. SA1</td> <td>3. SA2</td> <td>4. VSSSPD</td> </tr> <tr> <td>5. SDA</td> <td>6. SCL</td> <td>7. WC#</td> <td>8. VDDSPD</td> </tr> </table>	1. SA0	2. SA1	3. SA2	4. VSSSPD	5. SDA	6. SCL	7. WC#	8. VDDSPD
1. SA0	2. SA1	3. SA2	4. VSSSPD						
5. SDA	6. SCL	7. WC#	8. VDDSPD						

For more information about the EE1002-SPD-EEPROM Interface, refer to the JEDEC standard No. 21-C.

4.7.3.10.13. Embedded Display Port Interface

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/eTrak																																								
diagram	<pre> classDiagram class EmbeddedDisplayPort { <<EmbeddedDisplayPort-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<EmbeddedDisplayPort-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<EmbeddedDisplayPort-MandatoryMappingType>> } class OptionalMapping { <<EmbeddedDisplayPort-OptionalMappingType>> } class StandardTerminalName { <<EmbeddedDisplayPort-MandatoryStandardTerminalNameType>> } class StandardTerminalName { <<EmbeddedDisplayPort-OptionalStandardTerminalNameType>> } EmbeddedDisplayPort "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping StandardTerminalNameAssignment "0..19" OptionalMapping MandatoryMapping "11" StandardTerminalName OptionalMapping "0..19" StandardTerminalName </pre>																																								
type	EmbeddedDisplayPort-InterfaceFunctionType , EmbeddedDisplayPort-StandardTerminalNameAssignmentType , EmbeddedDisplayPort-MandatoryMappingType , EmbeddedDisplayPort-OptionalMappingType , EmbeddedDisplayPort-MandatoryStandardTerminalNameType , EmbeddedDisplayPort-OptionalStandardTerminalNameType .																																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. ML_Lane_0_P</td><td>2. ML_Lane_0_N</td><td>3. ML_Lane_1_P</td><td>4. ML_Lane_1_N</td></tr> <tr> <td>5. AUX_CH_P</td><td>6. AUX_CH_N</td><td>7. HPD</td><td>8. LCDVCC</td></tr> <tr> <td>9. H_GND</td><td>10. LCD_GND</td><td>11. NC-RESERVED</td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. ML_Lane_2_P</td><td>2. ML_Lane_2_N</td><td>3. ML_Lane_3_P</td><td>4. ML_Lane_3_N</td></tr> <tr> <td>5. VDC1</td><td>6. VDC2</td><td>7. VDC3</td><td>8. VDC4</td></tr> <tr> <td>9. VDC5</td><td>10. VDC6</td><td>11. VDC</td><td>12. BL_GND</td></tr> <tr> <td>13. BL_ENABLE</td><td>14. NC</td><td>15. ON/OFF</td><td>16. BL_PWM_DIM</td></tr> <tr> <td>17. PWM</td><td>18. BL_PWR</td><td>19. LCD_Self_Test</td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. ML_Lane_0_P	2. ML_Lane_0_N	3. ML_Lane_1_P	4. ML_Lane_1_N	5. AUX_CH_P	6. AUX_CH_N	7. HPD	8. LCDVCC	9. H_GND	10. LCD_GND	11. NC-RESERVED		OptionalMapping/StandardTerminalName				1. ML_Lane_2_P	2. ML_Lane_2_N	3. ML_Lane_3_P	4. ML_Lane_3_N	5. VDC1	6. VDC2	7. VDC3	8. VDC4	9. VDC5	10. VDC6	11. VDC	12. BL_GND	13. BL_ENABLE	14. NC	15. ON/OFF	16. BL_PWM_DIM	17. PWM	18. BL_PWR	19. LCD_Self_Test	
MandatoryMapping/StandardTerminalName																																									
1. ML_Lane_0_P	2. ML_Lane_0_N	3. ML_Lane_1_P	4. ML_Lane_1_N																																						
5. AUX_CH_P	6. AUX_CH_N	7. HPD	8. LCDVCC																																						
9. H_GND	10. LCD_GND	11. NC-RESERVED																																							
OptionalMapping/StandardTerminalName																																									
1. ML_Lane_2_P	2. ML_Lane_2_N	3. ML_Lane_3_P	4. ML_Lane_3_N																																						
5. VDC1	6. VDC2	7. VDC3	8. VDC4																																						
9. VDC5	10. VDC6	11. VDC	12. BL_GND																																						
13. BL_ENABLE	14. NC	15. ON/OFF	16. BL_PWM_DIM																																						
17. PWM	18. BL_PWR	19. LCD_Self_Test																																							

For more information about the Embedded Display Port (eDP) Interface, refer to the VESA Embedded DisplayPort (eDP) Standard.

4.7.3.10.14. Ethernet

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/Ethernet
diagram	<pre> classDiagram class Ethernet { <<EthernetInterfaceFunctionType>> } class Ethernet1000BASELX { <<Ethernet1000BASELX-InterfaceFunctionType>> } class Ethernet1000BASESX { <<Ethernet1000BASESX-InterfaceFunctionType>> } class Ethernet1000BASECX { <<Ethernet1000BASECX-InterfaceFunctionType>> } Ethernet < -- Ethernet1000BASELX Ethernet < -- Ethernet1000BASESX Ethernet < -- Ethernet1000BASECX </pre>
type	EthernetInterfaceFunctionType , Ethernet1000BASE-LX-InterfaceFunctionType , Ethernet1000BASE-SX-InterfaceFunctionType , Ethernet1000BASE-CX-InterfaceFunctionType .

For more information about the Ethernet Interface, refer to the IEEE Standard 802.3.

4.7.3.10.14.1. Ethernet1000BASE-LX

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ Ethernet1000BASE-LX																												
diagram	<pre> classDiagram class Ethernet1000BASELX { <<Ethernet1000BASELX-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<Ethernet1000BASELX-StandardTerminalNameAssignmentType>> } class Mapping { <<Ethernet1000BASELX-StandardTerminalMappingType>> } class StandardTerminalName { <<Ethernet1000BASELX-StandardTerminalNameType>> } Ethernet1000BASELX "1..∞" -- "24" StandardTerminalNameAssignment StandardTerminalNameAssignment -- "24" Mapping Mapping -- "24" StandardTerminalName </pre>																												
type	Ethernet1000BASE-LX-InterfaceFunctionType , Ethernet1000BASE-LX-StandardTerminalNameAssignmentType , Ethernet1000BASE-LX-StandardTerminalMappingType , Ethernet1000BASE-LX-StandardTerminalNameType .																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. TXD[0]</td><td>2. TXD[1]</td><td>3. TXD[2]</td><td>4. TXD[3]</td></tr> <tr> <td>5. TXD[4]</td><td>6. TXD[5]</td><td>7. TXD[6]</td><td>8. TXD[7]</td></tr> <tr> <td>9. TX_EN</td><td>10. TX_ER</td><td>11. GTX_CLK</td><td>12. RXD[0]</td></tr> <tr> <td>13. RXD[1]</td><td>14. RXD[2]</td><td>15. RXD[3]</td><td>16. RXD[4]</td></tr> <tr> <td>17. RXD[5]</td><td>18. RXD[6]</td><td>19. RXD[7]</td><td>20. RX_DV</td></tr> <tr> <td>21. RX_ER</td><td>22. RX_CLK</td><td>23. COL</td><td>24. CRS</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]	5. TXD[4]	6. TXD[5]	7. TXD[6]	8. TXD[7]	9. TX_EN	10. TX_ER	11. GTX_CLK	12. RXD[0]	13. RXD[1]	14. RXD[2]	15. RXD[3]	16. RXD[4]	17. RXD[5]	18. RXD[6]	19. RXD[7]	20. RX_DV	21. RX_ER	22. RX_CLK	23. COL	24. CRS
Mapping/StandardTerminalName																													
1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]																										
5. TXD[4]	6. TXD[5]	7. TXD[6]	8. TXD[7]																										
9. TX_EN	10. TX_ER	11. GTX_CLK	12. RXD[0]																										
13. RXD[1]	14. RXD[2]	15. RXD[3]	16. RXD[4]																										
17. RXD[5]	18. RXD[6]	19. RXD[7]	20. RX_DV																										
21. RX_ER	22. RX_CLK	23. COL	24. CRS																										

4.7.3.10.14.2. Ethernet1000BASE-SX

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ Ethernet1000BASE-SX																								
diagram	<pre> classDiagram class Ethernet1000BASE-SX-InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName Ethernet1000BASE-SX-InterfaceFunctionType "1..∞" -- "24" StandardTerminalNameAssignment StandardTerminalNameAssignment "24" -- "24" Mapping class Ethernet1000BASE-SX-StandardTerminalNameAssignmentType { StandardTerminalNameAssignment Mapping } class Ethernet1000BASE-SX-StandardTerminalMappingType { StandardTerminalName } class Ethernet1000BASE-SX-StandardTerminalNameType { StandardTerminalName } Ethernet1000BASE-SX-InterfaceFunctionType --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>																								
type	Ethernet1000BASE-SX-InterfaceFunctionType, Ethernet1000BASE-SX-StandardTerminalNameAssignmentType, Ethernet1000BASE-SX-StandardTerminalMappingType, Ethernet1000BASE-SX-StandardTerminalNameType.																								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. TXD[0]</td> <td>2. TXD[1]</td> <td>3. TXD[2]</td> <td>4. TXD[3]</td> </tr> <tr> <td>5. TXD[4]</td> <td>6. TXD[5]</td> <td>7. TXD[6]</td> <td>8. TXD[7]</td> </tr> <tr> <td>9. TX_EN</td> <td>10. TX_ER</td> <td>11. GTX_CLK</td> <td>12. RXD[0]</td> </tr> <tr> <td>13. RXD[1]</td> <td>14. RXD[2]</td> <td>15. RXD[3]</td> <td>16. RXD[4]</td> </tr> <tr> <td>17. RXD[5]</td> <td>18. RXD[6]</td> <td>19. RXD[7]</td> <td>20. RX_DV</td> </tr> <tr> <td>21. RX_ER</td> <td>22. RX_CLK</td> <td>23. COL</td> <td>24. CRS</td> </tr> </table>	1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]	5. TXD[4]	6. TXD[5]	7. TXD[6]	8. TXD[7]	9. TX_EN	10. TX_ER	11. GTX_CLK	12. RXD[0]	13. RXD[1]	14. RXD[2]	15. RXD[3]	16. RXD[4]	17. RXD[5]	18. RXD[6]	19. RXD[7]	20. RX_DV	21. RX_ER	22. RX_CLK	23. COL	24. CRS
1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]																						
5. TXD[4]	6. TXD[5]	7. TXD[6]	8. TXD[7]																						
9. TX_EN	10. TX_ER	11. GTX_CLK	12. RXD[0]																						
13. RXD[1]	14. RXD[2]	15. RXD[3]	16. RXD[4]																						
17. RXD[5]	18. RXD[6]	19. RXD[7]	20. RX_DV																						
21. RX_ER	22. RX_CLK	23. COL	24. CRS																						

4.7.3.10.14.3. Ethernet1000BASE-CX

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ Ethernet1000BASE-CX																								
diagram	<pre> classDiagram class Ethernet1000BASE-CX-InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName Ethernet1000BASE-CX-InterfaceFunctionType "1..∞" -- "24" StandardTerminalNameAssignment StandardTerminalNameAssignment "24" -- "24" Mapping class Ethernet1000BASE-CX-StandardTerminalNameAssignmentType { StandardTerminalNameAssignment Mapping } class Ethernet1000BASE-CX-StandardTerminalMappingType { StandardTerminalName } class Ethernet1000BASE-CX-StandardTerminalNameType { StandardTerminalName } Ethernet1000BASE-CX-InterfaceFunctionType --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>																								
type	Ethernet1000BASE-CX-InterfaceFunctionType, Ethernet1000BASE-CX-StandardTerminalNameAssignmentType, Ethernet1000BASE-CX-StandardTerminalMappingType, Ethernet1000BASE-CX-StandardTerminalNameType.																								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. TXD[0]</td> <td>2. TXD[1]</td> <td>3. TXD[2]</td> <td>4. TXD[3]</td> </tr> <tr> <td>5. TXD[4]</td> <td>6. TXD[5]</td> <td>7. TXD[6]</td> <td>8. TXD[7]</td> </tr> <tr> <td>9. TX_EN</td> <td>10. TX_ER</td> <td>11. GTX_CLK</td> <td>12. RXD[0]</td> </tr> <tr> <td>13. RXD[1]</td> <td>14. RXD[2]</td> <td>15. RXD[3]</td> <td>16. RXD[4]</td> </tr> <tr> <td>17. RXD[5]</td> <td>18. RXD[6]</td> <td>19. RXD[7]</td> <td>20. RX_DV</td> </tr> <tr> <td>21. RX_ER</td> <td>22. RX_CLK</td> <td>23. COL</td> <td>24. CRS</td> </tr> </table>	1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]	5. TXD[4]	6. TXD[5]	7. TXD[6]	8. TXD[7]	9. TX_EN	10. TX_ER	11. GTX_CLK	12. RXD[0]	13. RXD[1]	14. RXD[2]	15. RXD[3]	16. RXD[4]	17. RXD[5]	18. RXD[6]	19. RXD[7]	20. RX_DV	21. RX_ER	22. RX_CLK	23. COL	24. CRS
1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]																						
5. TXD[4]	6. TXD[5]	7. TXD[6]	8. TXD[7]																						
9. TX_EN	10. TX_ER	11. GTX_CLK	12. RXD[0]																						
13. RXD[1]	14. RXD[2]	15. RXD[3]	16. RXD[4]																						
17. RXD[5]	18. RXD[6]	19. RXD[7]	20. RX_DV																						
21. RX_ER	22. RX_CLK	23. COL	24. CRS																						

4.7.3.10.15. eTrak Interface

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/eTrak			
diagram				
type	eTrakInterfaceFunctionType , eTrakStandardTerminalNameAssignmentType , eTrakStandardTerminalMappingType , eTrakStandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. I 2. Q 3. VREF			

For more information about the eTrak Interface, refer to the MIPI Alliance standard Specification for Analog Interface for Envelope Tracking (eTrak) Version 1.1.

4.7.3.10.16. FC-PI-6 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/FC-PI-6			
diagram				
type	FC-PI-6-InterfaceFunctionType , FC-PI-6-StandardTerminalNameAssignmentType , FC-PI-6-StandardTerminalMappingType , FC-PI-6-StandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. Tx 2. Rx			

For more information about the FC-PI-6 Interface, refer to the INCITS standard 2221-D.

4.7.3.10.17. HBM Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM
diagram	
type	HBM-InterfaceFunctionType , HBM1-2-2E- FootprintA-InterfaceFunctionType , HBM1-2-2E- FootprintB-InterfaceFunctionType , HBM3-InterfaceFunctionType , HBM4-InterfaceFunctionType .

4.7.3.10.17.1. HBM1, HBM2 and HBM2E Footprint A Interface Functions

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM1-FootprintA 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2-FootprintA 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2E-FootprintA 																																																																																																												
diagram	<pre> classDiagram class HBM1-2-2E-FootprintA-InterfaceFunctionType { HBM1-FootprintA StandardTerminalNameAssignment Mapping StandardTerminalName } HBM1-FootprintA "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName StandardTerminalNameAssignment "constraints" </pre>																																																																																																												
type	HBM1-2-2E-FootprintA-InterfaceFunctionType, HBM1-2-2E-FootprintA-StandardTerminalNameAssignmentType, HBM1-2-2E-FootprintA-StandardTerminalMappingType, HBM1-2-2E-FootprintA-StandardTerminalNameType																																																																																																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. AERRa</td><td>2. AERRb</td><td>3. AERRc</td><td>4. AERRd</td></tr> <tr><td>5. AERRe</td><td>6. AERRf</td><td>7. AERRg</td><td>8. AERRh</td></tr> <tr><td>9. ARFUa[0]</td><td>10. ARFUa[1]</td><td>11. ARFUa[2]</td><td>12. ARFUa[3]</td></tr> <tr><td>13. ARFUb[0]</td><td>14. ARFUb[1]</td><td>15. ARFUb[2]</td><td>16. ARFUb[3]</td></tr> <tr><td>17. ARFUC[0]</td><td>18. ARFUC[1]</td><td>19. ARFUC[2]</td><td>20. ARFUC[3]</td></tr> <tr><td>21. ARFUD[0]</td><td>22. ARFUD[1]</td><td>23. ARFUD[2]</td><td>24. ARFUD[3]</td></tr> <tr><td>25. ARFUE[0]</td><td>26. ARFUE[1]</td><td>27. ARFUE[2]</td><td>28. ARFUE[3]</td></tr> <tr><td>29. ARFUF[0]</td><td>30. ARFUF[1]</td><td>31. ARFUF[2]</td><td>32. ARFUF[3]</td></tr> <tr><td>33. ARFUG[0]</td><td>34. ARFUG[1]</td><td>35. ARFUG[2]</td><td>36. ARFUG[3]</td></tr> <tr><td>37. ARFUh[0]</td><td>38. ARFUh[1]</td><td>39. ARFUh[2]</td><td>40. ARFUh[3]</td></tr> <tr><td>41. CAPTUREWR</td><td>42. CATTRIP</td><td>43. Ca[0]</td><td>44. Ca[1]</td></tr> <tr><td>45. Ca[2]</td><td>46. Ca[3]</td><td>47. Ca[4]</td><td>48. Ca[5]</td></tr> <tr><td>49. Ca[6]</td><td>50. Ca[7]</td><td>51. Cb[0]</td><td>52. Cb[1]</td></tr> <tr><td>53. Cb[2]</td><td>54. Cb[3]</td><td>55. Cb[4]</td><td>56. Cb[5]</td></tr> <tr><td>57. Cb[6]</td><td>58. Cb[7]</td><td>59. Cc[0]</td><td>60. Cc[1]</td></tr> <tr><td>61. Cc[2]</td><td>62. Cc[3]</td><td>63. Cc[4]</td><td>64. Cc[5]</td></tr> <tr><td>65. Cc[6]</td><td>66. Cc[7]</td><td>67. Cd[0]</td><td>68. Cd[1]</td></tr> <tr><td>69. Cd[2]</td><td>70. Cd[3]</td><td>71. Cd[4]</td><td>72. Cd[5]</td></tr> <tr><td>73. Cd[6]</td><td>74. Cd[7]</td><td>75. Ce[0]</td><td>76. Ce[1]</td></tr> <tr><td>77. Ce[2]</td><td>78. Ce[3]</td><td>79. Ce[4]</td><td>80. Ce[5]</td></tr> <tr><td>81. Ce[6]</td><td>82. Ce[7]</td><td>83. Cf[0]</td><td>84. Cf[1]</td></tr> <tr><td>85. Cf[2]</td><td>86. Cf[3]</td><td>87. Cf[4]</td><td>88. Cf[5]</td></tr> <tr><td>89. Cf[6]</td><td>90. Cf[7]</td><td>91. Cg[0]</td><td>92. Cg[1]</td></tr> <tr><td>93. Cg[2]</td><td>94. Cg[3]</td><td>95. Cg[4]</td><td>96. Cg[5]</td></tr> <tr><td>97. Cg[6]</td><td>98. Cg[7]</td><td>99. Ch[0]</td><td>100. Ch[1]</td></tr> <tr><td>101. Ch[2]</td><td>102. Ch[3]</td><td>103. Ch[4]</td><td>104. Ch[5]</td></tr> <tr><td>105. Ch[6]</td><td>106. Ch[7]</td><td>107. CKa_c</td><td>108. CKa_t</td></tr> </tbody> </table>	1. AERRa	2. AERRb	3. AERRc	4. AERRd	5. AERRe	6. AERRf	7. AERRg	8. AERRh	9. ARFUa[0]	10. ARFUa[1]	11. ARFUa[2]	12. ARFUa[3]	13. ARFUb[0]	14. ARFUb[1]	15. ARFUb[2]	16. ARFUb[3]	17. ARFUC[0]	18. ARFUC[1]	19. ARFUC[2]	20. ARFUC[3]	21. ARFUD[0]	22. ARFUD[1]	23. ARFUD[2]	24. ARFUD[3]	25. ARFUE[0]	26. ARFUE[1]	27. ARFUE[2]	28. ARFUE[3]	29. ARFUF[0]	30. ARFUF[1]	31. ARFUF[2]	32. ARFUF[3]	33. ARFUG[0]	34. ARFUG[1]	35. ARFUG[2]	36. ARFUG[3]	37. ARFUh[0]	38. ARFUh[1]	39. ARFUh[2]	40. ARFUh[3]	41. CAPTUREWR	42. CATTRIP	43. Ca[0]	44. Ca[1]	45. Ca[2]	46. Ca[3]	47. Ca[4]	48. Ca[5]	49. Ca[6]	50. Ca[7]	51. Cb[0]	52. Cb[1]	53. Cb[2]	54. Cb[3]	55. Cb[4]	56. Cb[5]	57. Cb[6]	58. Cb[7]	59. Cc[0]	60. Cc[1]	61. Cc[2]	62. Cc[3]	63. Cc[4]	64. Cc[5]	65. Cc[6]	66. Cc[7]	67. Cd[0]	68. Cd[1]	69. Cd[2]	70. Cd[3]	71. Cd[4]	72. Cd[5]	73. Cd[6]	74. Cd[7]	75. Ce[0]	76. Ce[1]	77. Ce[2]	78. Ce[3]	79. Ce[4]	80. Ce[5]	81. Ce[6]	82. Ce[7]	83. Cf[0]	84. Cf[1]	85. Cf[2]	86. Cf[3]	87. Cf[4]	88. Cf[5]	89. Cf[6]	90. Cf[7]	91. Cg[0]	92. Cg[1]	93. Cg[2]	94. Cg[3]	95. Cg[4]	96. Cg[5]	97. Cg[6]	98. Cg[7]	99. Ch[0]	100. Ch[1]	101. Ch[2]	102. Ch[3]	103. Ch[4]	104. Ch[5]	105. Ch[6]	106. Ch[7]	107. CKa_c	108. CKa_t
1. AERRa	2. AERRb	3. AERRc	4. AERRd																																																																																																										
5. AERRe	6. AERRf	7. AERRg	8. AERRh																																																																																																										
9. ARFUa[0]	10. ARFUa[1]	11. ARFUa[2]	12. ARFUa[3]																																																																																																										
13. ARFUb[0]	14. ARFUb[1]	15. ARFUb[2]	16. ARFUb[3]																																																																																																										
17. ARFUC[0]	18. ARFUC[1]	19. ARFUC[2]	20. ARFUC[3]																																																																																																										
21. ARFUD[0]	22. ARFUD[1]	23. ARFUD[2]	24. ARFUD[3]																																																																																																										
25. ARFUE[0]	26. ARFUE[1]	27. ARFUE[2]	28. ARFUE[3]																																																																																																										
29. ARFUF[0]	30. ARFUF[1]	31. ARFUF[2]	32. ARFUF[3]																																																																																																										
33. ARFUG[0]	34. ARFUG[1]	35. ARFUG[2]	36. ARFUG[3]																																																																																																										
37. ARFUh[0]	38. ARFUh[1]	39. ARFUh[2]	40. ARFUh[3]																																																																																																										
41. CAPTUREWR	42. CATTRIP	43. Ca[0]	44. Ca[1]																																																																																																										
45. Ca[2]	46. Ca[3]	47. Ca[4]	48. Ca[5]																																																																																																										
49. Ca[6]	50. Ca[7]	51. Cb[0]	52. Cb[1]																																																																																																										
53. Cb[2]	54. Cb[3]	55. Cb[4]	56. Cb[5]																																																																																																										
57. Cb[6]	58. Cb[7]	59. Cc[0]	60. Cc[1]																																																																																																										
61. Cc[2]	62. Cc[3]	63. Cc[4]	64. Cc[5]																																																																																																										
65. Cc[6]	66. Cc[7]	67. Cd[0]	68. Cd[1]																																																																																																										
69. Cd[2]	70. Cd[3]	71. Cd[4]	72. Cd[5]																																																																																																										
73. Cd[6]	74. Cd[7]	75. Ce[0]	76. Ce[1]																																																																																																										
77. Ce[2]	78. Ce[3]	79. Ce[4]	80. Ce[5]																																																																																																										
81. Ce[6]	82. Ce[7]	83. Cf[0]	84. Cf[1]																																																																																																										
85. Cf[2]	86. Cf[3]	87. Cf[4]	88. Cf[5]																																																																																																										
89. Cf[6]	90. Cf[7]	91. Cg[0]	92. Cg[1]																																																																																																										
93. Cg[2]	94. Cg[3]	95. Cg[4]	96. Cg[5]																																																																																																										
97. Cg[6]	98. Cg[7]	99. Ch[0]	100. Ch[1]																																																																																																										
101. Ch[2]	102. Ch[3]	103. Ch[4]	104. Ch[5]																																																																																																										
105. Ch[6]	106. Ch[7]	107. CKa_c	108. CKa_t																																																																																																										

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	109. CKb_c	110. CKb_t	111. CKc_c	112. CKc_t
	113. CKd_c	114. CKd_t	115. CKe_c	116. CKe_t
	117. CKEa	118. CKEb	119. CKEc	120. CKEd
	121. CKEe	122. CKEf	123. CKEg	124. CKEh
	125. CKf_c	126. CKf_t	127. CKg_c	128. CKg_t
	129. CKh_c	130. CKh_t	131. DA[0]	132. DA[1]
	133. DA[2]	134. DA[3]	135. DA[4]	136. DA[5]
	137. DA[6]	138. DA[7]	139. DA[8]	140. DA[9]
	141. DA[10]	142. DA[11]	143. DA[12]	144. DA[13]
	145. DA[14]	146. DA[15]	147. DA[16]	148. DA[17]
	149. DA[18]	150. DA[19]	151. DA[20]	152. DA[21]
	153. DA[22]	154. DA[23]	155. DA[24]	156. DA[25]
	157. DA[26]	158. DA[27]	159. DA[28]	160. DA[29]
	161. DA[30]	162. DA[31]	163. DA[32]	164. DA[33]
	165. DA[34]	166. DA[35]	167. DA[36]	168. DA[37]
	169. DA[38]	170. DA[39]	171. DA[40]	172. DA[41]
	173. DA[42]	174. DA[43]	175. DA[44]	176. DA[45]
	177. DA[46]	178. DA[47]	179. DA[48]	180. DA[49]
	181. DA[50]	182. DA[51]	183. DA[52]	184. DA[53]
	185. DA[54]	186. DA[55]	187. DA[56]	188. DA[57]
	189. DA[58]	190. DA[59]	191. DBla[0]	192. DBla[1]
	193. DBla[2]	194. DBla[3]	195. DBla[4]	196. DBla[5]
	197. DBla[6]	198. DBla[7]	199. DBla[8]	200. DBla[9]
	201. DBla[10]	202. DBla[11]	203. DBla[12]	204. DBla[13]
	205. DBla[14]	206. DBla[15]	207. DBlb[0]	208. DBlb[1]
	209. DBlb[2]	210. DBlb[3]	211. DBlb[4]	212. DBlb[5]
	213. DBlb[6]	214. DBlb[7]	215. DBlb[8]	216. DBlb[9]
	217. DBlb[10]	218. DBlb[11]	219. DBlb[12]	220. DBlb[13]
	221. DBlb[14]	222. DBlb[15]	223. DBlc[0]	224. DBlc[1]
	225. DBlc[2]	226. DBlc[3]	227. DBlc[4]	228. DBlc[5]
	229. DBlc[6]	230. DBlc[7]	231. DBlc[8]	232. DBlc[9]
	233. DBlc[10]	234. DBlc[11]	235. DBlc[12]	236. DBlc[13]
	237. DBlc[14]	238. DBlc[15]	239. DBld[0]	240. DBld[1]
	241. DBld[2]	242. DBld[3]	243. DBld[4]	244. DBld[5]
	245. DBld[6]	246. DBld[7]	247. DBld[8]	248. DBld[9]
	249. DBld[10]	250. DBld[11]	251. DBld[12]	252. DBld[13]
	253. DBld[14]	254. DBld[15]	255. DBle[0]	256. DBle[1]
	257. DBle[2]	258. DBle[3]	259. DBle[4]	260. DBle[5]
	261. DBle[6]	262. DBle[7]	263. DBle[8]	264. DBle[9]
	265. DBle[10]	266. DBle[11]	267. DBle[12]	268. DBle[13]
	269. DBle[14]	270. DBle[15]	271. DBlf[0]	272. DBlf[1]
	273. DBlf[2]	274. DBlf[3]	275. DBlf[4]	276. DBlf[5]
	277. DBlf[6]	278. DBlf[7]	279. DBlf[8]	280. DBlf[9]
	281. DBlf[10]	282. DBlf[11]	283. DBlf[12]	284. DBlf[13]
	285. DBlf[14]	286. DBlf[15]	287. DBlg[0]	288. DBlg[1]
	289. DBlg[2]	290. DBlg[3]	291. DBlg[4]	292. DBlg[5]
	293. DBlg[6]	294. DBlg[7]	295. DBlg[8]	296. DBlg[9]
	297. DBlg[10]	298. DBlg[11]	299. DBlg[12]	300. DBlg[13]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	301. DBIg[14]	302. DBIg[15]	303. DBIh[0]	304. DBIh[1]
	305. DBIh[2]	306. DBIh[3]	307. DBIh[4]	308. DBIh[5]
	309. DBIh[6]	310. DBIh[7]	311. DBIh[8]	312. DBIh[9]
	313. DBIh[10]	314. DBIh[11]	315. DBIh[12]	316. DBIh[13]
	317. DBIh[14]	318. DBIh[15]	319. DERRa[0]	320. DERRa[1]
	321. DERRa[2]	322. DERRa[3]	323. DERRb[0]	324. DERRb[1]
	325. DERRb[2]	326. DERRb[3]	327. DERRc[0]	328. DERRc[1]
	329. DERRc[2]	330. DERRc[3]	331. DERRd[0]	332. DERRd[1]
	333. DERRd[2]	334. DERRd[3]	335. DERRe[0]	336. DERRe[1]
	337. DERRe[2]	338. DERRe[3]	339. DERRf[0]	340. DERRf[1]
	341. DERRf[2]	342. DERRf[3]	343. DERRg[0]	344. DERRg[1]
	345. DERRg[2]	346. DERRg[3]	347. DERRh[0]	348. DERRh[1]
	349. DERRh[2]	350. DERRh[3]	351. DMa[0]	352. DMa[1]
	353. DMa[2]	354. DMa[3]	355. DMa[4]	356. DMa[5]
	357. DMa[6]	358. DMa[7]	359. DMa[8]	360. DMa[9]
	361. DMa[10]	362. DMa[11]	363. DMa[12]	364. DMa[13]
	365. DMa[14]	366. DMa[15]	367. DMb[0]	368. DMb[1]
	369. DMb[2]	370. DMb[3]	371. DMb[4]	372. DMb[5]
	373. DMb[6]	374. DMb[7]	375. DMb[8]	376. DMb[9]
	377. DMb[10]	378. DMb[11]	379. DMb[12]	380. DMb[13]
	381. DMb[14]	382. DMb[15]	383. DMc[0]	384. DMc[1]
	385. DMc[2]	386. DMc[3]	387. DMc[4]	388. DMc[5]
	389. DMc[6]	390. DMc[7]	391. DMc[8]	392. DMc[9]
	393. DMc[10]	394. DMc[11]	395. DMc[12]	396. DMc[13]
	397. DMc[14]	398. DMc[15]	399. DMd[0]	400. DMd[1]
	401. DMd[2]	402. DMd[3]	403. DMd[4]	404. DMd[5]
	405. DMd[6]	406. DMd[7]	407. DMd[8]	408. DMd[9]
	409. DMd[10]	410. DMd[11]	411. DMd[12]	412. DMd[13]
	413. DMd[14]	414. DMd[15]	415. DMe[0]	416. DMe[1]
	417. DMe[2]	418. DMe[3]	419. DMe[4]	420. DMe[5]
	421. DMe[6]	422. DMe[7]	423. DMe[8]	424. DMe[9]
	425. DMe[10]	426. DMe[11]	427. DMe[12]	428. DMe[13]
	429. DMe[14]	430. DMe[15]	431. DMf[0]	432. DMf[1]
	433. DMf[2]	434. DMf[3]	435. DMf[4]	436. DMf[5]
	437. DMf[6]	438. DMf[7]	439. DMf[8]	440. DMf[9]
	441. DMf[10]	442. DMf[11]	443. DMf[12]	444. DMf[13]
	445. DMf[14]	446. DMf[15]	447. DMg[0]	448. DMg[1]
	449. DMg[2]	450. DMg[3]	451. DMg[4]	452. DMg[5]
	453. DMg[6]	454. DMg[7]	455. DMg[8]	456. DMg[9]
	457. DMg[10]	458. DMg[11]	459. DMg[12]	460. DMg[13]
	461. DMg[14]	462. DMg[15]	463. DMh[0]	464. DMh[1]
	465. DMh[2]	466. DMh[3]	467. DMh[4]	468. DMh[5]
	469. DMh[6]	470. DMh[7]	471. DMh[8]	472. DMh[9]
	473. DMh[10]	474. DMh[11]	475. DMh[12]	476. DMh[13]
	477. DMh[14]	478. DMh[15]	479. DQa[0]	480. DQa[1]
	481. DQa[2]	482. DQa[3]	483. DQa[4]	484. DQa[5]
	485. DQa[6]	486. DQa[7]	487. DQa[8]	488. DQa[9]
	489. DQa[10]	490. DQa[11]	491. DQa[12]	492. DQa[13]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	493. DQa[14]	494. DQa[15]	495. DQa[16]	496. DQa[17]
	497. DQa[18]	498. DQa[19]	499. DQa[20]	500. DQa[21]
	501. DQa[22]	502. DQa[23]	503. DQa[24]	504. DQa[25]
	505. DQa[26]	506. DQa[27]	507. DQa[28]	508. DQa[29]
	509. DQa[30]	510. DQa[31]	511. DQa[32]	512. DQa[33]
	513. DQa[34]	514. DQa[35]	515. DQa[36]	516. DQa[37]
	517. DQa[38]	518. DQa[39]	519. DQa[40]	520. DQa[41]
	521. DQa[42]	522. DQa[43]	523. DQa[44]	524. DQa[45]
	525. DQa[46]	526. DQa[47]	527. DQa[48]	528. DQa[49]
	529. DQa[50]	530. DQa[51]	531. DQa[52]	532. DQa[53]
	533. DQa[54]	534. DQa[55]	535. DQa[56]	536. DQa[57]
	537. DQa[58]	538. DQa[59]	539. DQa[60]	540. DQa[61]
	541. DQa[62]	542. DQa[63]	543. DQa[64]	544. DQa[65]
	545. DQa[66]	546. DQa[67]	547. DQa[68]	548. DQa[69]
	549. DQa[70]	550. DQa[71]	551. DQa[72]	552. DQa[73]
	553. DQa[74]	554. DQa[75]	555. DQa[76]	556. DQa[77]
	557. DQa[78]	558. DQa[79]	559. DQa[80]	560. DQa[81]
	561. DQa[82]	562. DQa[83]	563. DQa[84]	564. DQa[85]
	565. DQa[86]	566. DQa[87]	567. DQa[88]	568. DQa[89]
	569. DQa[90]	570. DQa[91]	571. DQa[92]	572. DQa[93]
	573. DQa[94]	574. DQa[95]	575. DQa[96]	576. DQa[97]
	577. DQa[98]	578. DQa[99]	579. DQa[100]	580. DQa[101]
	581. DQa[102]	582. DQa[103]	583. DQa[104]	584. DQa[105]
	585. DQa[106]	586. DQa[107]	587. DQa[108]	588. DQa[109]
	589. DQa[110]	590. DQa[111]	591. DQa[112]	592. DQa[113]
	593. DQa[114]	594. DQa[115]	595. DQa[116]	596. DQa[117]
	597. DQa[118]	598. DQa[119]	599. DQa[120]	600. DQa[121]
	601. DQa[122]	602. DQa[123]	603. DQa[124]	604. DQa[125]
	605. DQa[126]	606. DQa[127]	607. DQb[0]	608. DQb[1]
	609. DQb[2]	610. DQb[3]	611. DQb[4]	612. DQb[5]
	613. DQb[6]	614. DQb[7]	615. DQb[8]	616. DQb[9]
	617. DQb[10]	618. DQb[11]	619. DQb[12]	620. DQb[13]
	621. DQb[14]	622. DQb[15]	623. DQb[16]	624. DQb[17]
	625. DQb[18]	626. DQb[19]	627. DQb[20]	628. DQb[21]
	629. DQb[22]	630. DQb[23]	631. DQb[24]	632. DQb[25]
	633. DQb[26]	634. DQb[27]	635. DQb[28]	636. DQb[29]
	637. DQb[30]	638. DQb[31]	639. DQb[32]	640. DQb[33]
	641. DQb[34]	642. DQb[35]	643. DQb[36]	644. DQb[37]
	645. DQb[38]	646. DQb[39]	647. DQb[40]	648. DQb[41]
	649. DQb[42]	650. DQb[43]	651. DQb[44]	652. DQb[45]
	653. DQb[46]	654. DQb[47]	655. DQb[48]	656. DQb[49]
	657. DQb[50]	658. DQb[51]	659. DQb[52]	660. DQb[53]
	661. DQb[54]	662. DQb[55]	663. DQb[56]	664. DQb[57]
	665. DQb[58]	666. DQb[59]	667. DQb[60]	668. DQb[61]
	669. DQb[62]	670. DQb[63]	671. DQb[64]	672. DQb[65]
	673. DQb[66]	674. DQb[67]	675. DQb[68]	676. DQb[69]
	677. DQb[70]	678. DQb[71]	679. DQb[72]	680. DQb[73]
	681. DQb[74]	682. DQb[75]	683. DQb[76]	684. DQb[77]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	685. DQb[78]	686. DQb[79]	687. DQb[80]	688. DQb[81]
	689. DQb[82]	690. DQb[83]	691. DQb[84]	692. DQb[85]
	693. DQb[86]	694. DQb[87]	695. DQb[88]	696. DQb[89]
	697. DQb[90]	698. DQb[91]	699. DQb[92]	700. DQb[93]
	701. DQb[94]	702. DQb[95]	703. DQb[96]	704. DQb[97]
	705. DQb[98]	706. DQb[99]	707. DQb[100]	708. DQb[101]
	709. DQb[102]	710. DQb[103]	711. DQb[104]	712. DQb[105]
	713. DQb[106]	714. DQb[107]	715. DQb[108]	716. DQb[109]
	717. DQb[110]	718. DQb[111]	719. DQb[112]	720. DQb[113]
	721. DQb[114]	722. DQb[115]	723. DQb[116]	724. DQb[117]
	725. DQb[118]	726. DQb[119]	727. DQb[120]	728. DQb[121]
	729. DQb[122]	730. DQb[123]	731. DQb[124]	732. DQb[125]
	733. DQb[126]	734. DQb[127]	735. DQc[0]	736. DQc[1]
	737. DQc[2]	738. DQc[3]	739. DQc[4]	740. DQc[5]
	741. DQc[6]	742. DQc[7]	743. DQc[8]	744. DQc[9]
	745. DQc[10]	746. DQc[11]	747. DQc[12]	748. DQc[13]
	749. DQc[14]	750. DQc[15]	751. DQc[16]	752. DQc[17]
	753. DQc[18]	754. DQc[19]	755. DQc[20]	756. DQc[21]
	757. DQc[22]	758. DQc[23]	759. DQc[24]	760. DQc[25]
	761. DQc[26]	762. DQc[27]	763. DQc[28]	764. DQc[29]
	765. DQc[30]	766. DQc[31]	767. DQc[32]	768. DQc[33]
	769. DQc[34]	770. DQc[35]	771. DQc[36]	772. DQc[37]
	773. DQc[38]	774. DQc[39]	775. DQc[40]	776. DQc[41]
	777. DQc[42]	778. DQc[43]	779. DQc[44]	780. DQc[45]
	781. DQc[46]	782. DQc[47]	783. DQc[48]	784. DQc[49]
	785. DQc[50]	786. DQc[51]	787. DQc[52]	788. DQc[53]
	789. DQc[54]	790. DQc[55]	791. DQc[56]	792. DQc[57]
	793. DQc[58]	794. DQc[59]	795. DQc[60]	796. DQc[61]
	797. DQc[62]	798. DQc[63]	799. DQc[64]	800. DQc[65]
	801. DQc[66]	802. DQc[67]	803. DQc[68]	804. DQc[69]
	805. DQc[70]	806. DQc[71]	807. DQc[72]	808. DQc[73]
	809. DQc[74]	810. DQc[75]	811. DQc[76]	812. DQc[77]
	813. DQc[78]	814. DQc[79]	815. DQc[80]	816. DQc[81]
	817. DQc[82]	818. DQc[83]	819. DQc[84]	820. DQc[85]
	821. DQc[86]	822. DQc[87]	823. DQc[88]	824. DQc[89]
	825. DQc[90]	826. DQc[91]	827. DQc[92]	828. DQc[93]
	829. DQc[94]	830. DQc[95]	831. DQc[96]	832. DQc[97]
	833. DQc[98]	834. DQc[99]	835. DQc[100]	836. DQc[101]
	837. DQc[102]	838. DQc[103]	839. DQc[104]	840. DQc[105]
	841. DQc[106]	842. DQc[107]	843. DQc[108]	844. DQc[109]
	845. DQc[110]	846. DQc[111]	847. DQc[112]	848. DQc[113]
	849. DQc[114]	850. DQc[115]	851. DQc[116]	852. DQc[117]
	853. DQc[118]	854. DQc[119]	855. DQc[120]	856. DQc[121]
	857. DQc[122]	858. DQc[123]	859. DQc[124]	860. DQc[125]
	861. DQc[126]	862. DQc[127]	863. DQd[0]	864. DQd[1]
	865. DQd[2]	866. DQd[3]	867. DQd[4]	868. DQd[5]
	869. DQd[6]	870. DQd[7]	871. DQd[8]	872. DQd[9]
	873. DQd[10]	874. DQd[11]	875. DQd[12]	876. DQd[13]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	877. DQd[14]	878. DQd[15]	879. DQd[16]	880. DQd[17]
	881. DQd[18]	882. DQd[19]	883. DQd[20]	884. DQd[21]
	885. DQd[22]	886. DQd[23]	887. DQd[24]	888. DQd[25]
	889. DQd[26]	890. DQd[27]	891. DQd[28]	892. DQd[29]
	893. DQd[30]	894. DQd[31]	895. DQd[32]	896. DQd[33]
	897. DQd[34]	898. DQd[35]	899. DQd[36]	900. DQd[37]
	901. DQd[38]	902. DQd[39]	903. DQd[40]	904. DQd[41]
	905. DQd[42]	906. DQd[43]	907. DQd[44]	908. DQd[45]
	909. DQd[46]	910. DQd[47]	911. DQd[48]	912. DQd[49]
	913. DQd[50]	914. DQd[51]	915. DQd[52]	916. DQd[53]
	917. DQd[54]	918. DQd[55]	919. DQd[56]	920. DQd[57]
	921. DQd[58]	922. DQd[59]	923. DQd[60]	924. DQd[61]
	925. DQd[62]	926. DQd[63]	927. DQd[64]	928. DQd[65]
	929. DQd[66]	930. DQd[67]	931. DQd[68]	932. DQd[69]
	933. DQd[70]	934. DQd[71]	935. DQd[72]	936. DQd[73]
	937. DQd[74]	938. DQd[75]	939. DQd[76]	940. DQd[77]
	941. DQd[78]	942. DQd[79]	943. DQd[80]	944. DQd[81]
	945. DQd[82]	946. DQd[83]	947. DQd[84]	948. DQd[85]
	949. DQd[86]	950. DQd[87]	951. DQd[88]	952. DQd[89]
	953. DQd[90]	954. DQd[91]	955. DQd[92]	956. DQd[93]
	957. DQd[94]	958. DQd[95]	959. DQd[96]	960. DQd[97]
	961. DQd[98]	962. DQd[99]	963. DQd[100]	964. DQd[101]
	965. DQd[102]	966. DQd[103]	967. DQd[104]	968. DQd[105]
	969. DQd[106]	970. DQd[107]	971. DQd[108]	972. DQd[109]
	973. DQd[110]	974. DQd[111]	975. DQd[112]	976. DQd[113]
	977. DQd[114]	978. DQd[115]	979. DQd[116]	980. DQd[117]
	981. DQd[118]	982. DQd[119]	983. DQd[120]	984. DQd[121]
	985. DQd[122]	986. DQd[123]	987. DQd[124]	988. DQd[125]
	989. DQd[126]	990. DQd[127]	991. DQe[0]	992. DQe[1]
	993. DQe[2]	994. DQe[3]	995. DQe[4]	996. DQe[5]
	997. DQe[6]	998. DQe[7]	999. DQe[8]	1000. DQe[9]
	1001. DQe[10]	1002. DQe[11]	1003. DQe[12]	1004. DQe[13]
	1005. DQe[14]	1006. DQe[15]	1007. DQe[16]	1008. DQe[17]
	1009. DQe[18]	1010. DQe[19]	1011. DQe[20]	1012. DQe[21]
	1013. DQe[22]	1014. DQe[23]	1015. DQe[24]	1016. DQe[25]
	1017. DQe[26]	1018. DQe[27]	1019. DQe[28]	1020. DQe[29]
	1021. DQe[30]	1022. DQe[31]	1023. DQe[32]	1024. DQe[33]
	1025. DQe[34]	1026. DQe[35]	1027. DQe[36]	1028. DQe[37]
	1029. DQe[38]	1030. DQe[39]	1031. DQe[40]	1032. DQe[41]
	1033. DQe[42]	1034. DQe[43]	1035. DQe[44]	1036. DQe[45]
	1037. DQe[46]	1038. DQe[47]	1039. DQe[48]	1040. DQe[49]
	1041. DQe[50]	1042. DQe[51]	1043. DQe[52]	1044. DQe[53]
	1045. DQe[54]	1046. DQe[55]	1047. DQe[56]	1048. DQe[57]
	1049. DQe[58]	1050. DQe[59]	1051. DQe[60]	1052. DQe[61]
	1053. DQe[62]	1054. DQe[63]	1055. DQe[64]	1056. DQe[65]
	1057. DQe[66]	1058. DQe[67]	1059. DQe[68]	1060. DQe[69]
	1061. DQe[70]	1062. DQe[71]	1063. DQe[72]	1064. DQe[73]
	1065. DQe[74]	1066. DQe[75]	1067. DQe[76]	1068. DQe[77]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	1069. DQe[78]	1070. DQe[79]	1071. DQe[80]	1072. DQe[81]
	1073. DQe[82]	1074. DQe[83]	1075. DQe[84]	1076. DQe[85]
	1077. DQe[86]	1078. DQe[87]	1079. DQe[88]	1080. DQe[89]
	1081. DQe[90]	1082. DQe[91]	1083. DQe[92]	1084. DQe[93]
	1085. DQe[94]	1086. DQe[95]	1087. DQe[96]	1088. DQe[97]
	1089. DQe[98]	1090. DQe[99]	1091. DQe[100]	1092. DQe[101]
	1093. DQe[102]	1094. DQe[103]	1095. DQe[104]	1096. DQe[105]
	1097. DQe[106]	1098. DQe[107]	1099. DQe[108]	1100. DQe[109]
	1101. DQe[110]	1102. DQe[111]	1103. DQe[112]	1104. DQe[113]
	1105. DQe[114]	1106. DQe[115]	1107. DQe[116]	1108. DQe[117]
	1109. DQe[118]	1110. DQe[119]	1111. DQe[120]	1112. DQe[121]
	1113. DQe[122]	1114. DQe[123]	1115. DQe[124]	1116. DQe[125]
	1117. DQe[126]	1118. DQe[127]	1119. DQf[0]	1120. DQf[1]
	1121. DQf[2]	1122. DQf[3]	1123. DQf[4]	1124. DQf[5]
	1125. DQf[6]	1126. DQf[7]	1127. DQf[8]	1128. DQf[9]
	1129. DQf[10]	1130. DQf[11]	1131. DQf[12]	1132. DQf[13]
	1133. DQf[14]	1134. DQf[15]	1135. DQf[16]	1136. DQf[17]
	1137. DQf[18]	1138. DQf[19]	1139. DQf[20]	1140. DQf[21]
	1141. DQf[22]	1142. DQf[23]	1143. DQf[24]	1144. DQf[25]
	1145. DQf[26]	1146. DQf[27]	1147. DQf[28]	1148. DQf[29]
	1149. DQf[30]	1150. DQf[31]	1151. DQf[32]	1152. DQf[33]
	1153. DQf[34]	1154. DQf[35]	1155. DQf[36]	1156. DQf[37]
	1157. DQf[38]	1158. DQf[39]	1159. DQf[40]	1160. DQf[41]
	1161. DQf[42]	1162. DQf[43]	1163. DQf[44]	1164. DQf[45]
	1165. DQf[46]	1166. DQf[47]	1167. DQf[48]	1168. DQf[49]
	1169. DQf[50]	1170. DQf[51]	1171. DQf[52]	1172. DQf[53]
	1173. DQf[54]	1174. DQf[55]	1175. DQf[56]	1176. DQf[57]
	1177. DQf[58]	1178. DQf[59]	1179. DQf[60]	1180. DQf[61]
	1181. DQf[62]	1182. DQf[63]	1183. DQf[64]	1184. DQf[65]
	1185. DQf[66]	1186. DQf[67]	1187. DQf[68]	1188. DQf[69]
	1189. DQf[70]	1190. DQf[71]	1191. DQf[72]	1192. DQf[73]
	1193. DQf[74]	1194. DQf[75]	1195. DQf[76]	1196. DQf[77]
	1197. DQf[78]	1198. DQf[79]	1199. DQf[80]	1200. DQf[81]
	1201. DQf[82]	1202. DQf[83]	1203. DQf[84]	1204. DQf[85]
	1205. DQf[86]	1206. DQf[87]	1207. DQf[88]	1208. DQf[89]
	1209. DQf[90]	1210. DQf[91]	1211. DQf[92]	1212. DQf[93]
	1213. DQf[94]	1214. DQf[95]	1215. DQf[96]	1216. DQf[97]
	1217. DQf[98]	1218. DQf[99]	1219. DQf[100]	1220. DQf[101]
	1221. DQf[102]	1222. DQf[103]	1223. DQf[104]	1224. DQf[105]
	1225. DQf[106]	1226. DQf[107]	1227. DQf[108]	1228. DQf[109]
	1229. DQf[110]	1230. DQf[111]	1231. DQf[112]	1232. DQf[113]
	1233. DQf[114]	1234. DQf[115]	1235. DQf[116]	1236. DQf[117]
	1237. DQf[118]	1238. DQf[119]	1239. DQf[120]	1240. DQf[121]
	1241. DQf[122]	1242. DQf[123]	1243. DQf[124]	1244. DQf[125]
	1245. DQf[126]	1246. DQf[127]	1247. DQg[0]	1248. DQg[1]
	1249. DQg[2]	1250. DQg[3]	1251. DQg[4]	1252. DQg[5]
	1253. DQg[6]	1254. DQg[7]	1255. DQg[8]	1256. DQg[9]
	1257. DQg[10]	1258. DQg[11]	1259. DQg[12]	1260. DQg[13]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	1261. DQg[14]	1262. DQg[15]	1263. DQg[16]	1264. DQg[17]
	1265. DQg[18]	1266. DQg[19]	1267. DQg[20]	1268. DQg[21]
	1269. DQg[22]	1270. DQg[23]	1271. DQg[24]	1272. DQg[25]
	1273. DQg[26]	1274. DQg[27]	1275. DQg[28]	1276. DQg[29]
	1277. DQg[30]	1278. DQg[31]	1279. DQg[32]	1280. DQg[33]
	1281. DQg[34]	1282. DQg[35]	1283. DQg[36]	1284. DQg[37]
	1285. DQg[38]	1286. DQg[39]	1287. DQg[40]	1288. DQg[41]
	1289. DQg[42]	1290. DQg[43]	1291. DQg[44]	1292. DQg[45]
	1293. DQg[46]	1294. DQg[47]	1295. DQg[48]	1296. DQg[49]
	1297. DQg[50]	1298. DQg[51]	1299. DQg[52]	1300. DQg[53]
	1301. DQg[54]	1302. DQg[55]	1303. DQg[56]	1304. DQg[57]
	1305. DQg[58]	1306. DQg[59]	1307. DQg[60]	1308. DQg[61]
	1309. DQg[62]	1310. DQg[63]	1311. DQg[64]	1312. DQg[65]
	1313. DQg[66]	1314. DQg[67]	1315. DQg[68]	1316. DQg[69]
	1317. DQg[70]	1318. DQg[71]	1319. DQg[72]	1320. DQg[73]
	1321. DQg[74]	1322. DQg[75]	1323. DQg[76]	1324. DQg[77]
	1325. DQg[78]	1326. DQg[79]	1327. DQg[80]	1328. DQg[81]
	1329. DQg[82]	1330. DQg[83]	1331. DQg[84]	1332. DQg[85]
	1333. DQg[86]	1334. DQg[87]	1335. DQg[88]	1336. DQg[89]
	1337. DQg[90]	1338. DQg[91]	1339. DQg[92]	1340. DQg[93]
	1341. DQg[94]	1342. DQg[95]	1343. DQg[96]	1344. DQg[97]
	1345. DQg[98]	1346. DQg[99]	1347. DQg[100]	1348. DQg[101]
	1349. DQg[102]	1350. DQg[103]	1351. DQg[104]	1352. DQg[105]
	1353. DQg[106]	1354. DQg[107]	1355. DQg[108]	1356. DQg[109]
	1357. DQg[110]	1358. DQg[111]	1359. DQg[112]	1360. DQg[113]
	1361. DQg[114]	1362. DQg[115]	1363. DQg[116]	1364. DQg[117]
	1365. DQg[118]	1366. DQg[119]	1367. DQg[120]	1368. DQg[121]
	1369. DQg[122]	1370. DQg[123]	1371. DQg[124]	1372. DQg[125]
	1373. DQg[126]	1374. DQg[127]	1375. DQh[0]	1376. DQh[1]
	1377. DQh[2]	1378. DQh[3]	1379. DQh[4]	1380. DQh[5]
	1381. DQh[6]	1382. DQh[7]	1383. DQh[8]	1384. DQh[9]
	1385. DQh[10]	1386. DQh[11]	1387. DQh[12]	1388. DQh[13]
	1389. DQh[14]	1390. DQh[15]	1391. DQh[16]	1392. DQh[17]
	1393. DQh[18]	1394. DQh[19]	1395. DQh[20]	1396. DQh[21]
	1397. DQh[22]	1398. DQh[23]	1399. DQh[24]	1400. DQh[25]
	1401. DQh[26]	1402. DQh[27]	1403. DQh[28]	1404. DQh[29]
	1405. DQh[30]	1406. DQh[31]	1407. DQh[32]	1408. DQh[33]
	1409. DQh[34]	1410. DQh[35]	1411. DQh[36]	1412. DQh[37]
	1413. DQh[38]	1414. DQh[39]	1415. DQh[40]	1416. DQh[41]
	1417. DQh[42]	1418. DQh[43]	1419. DQh[44]	1420. DQh[45]
	1421. DQh[46]	1422. DQh[47]	1423. DQh[48]	1424. DQh[49]
	1425. DQh[50]	1426. DQh[51]	1427. DQh[52]	1428. DQh[53]
	1429. DQh[54]	1430. DQh[55]	1431. DQh[56]	1432. DQh[57]
	1433. DQh[58]	1434. DQh[59]	1435. DQh[60]	1436. DQh[61]
	1437. DQh[62]	1438. DQh[63]	1439. DQh[64]	1440. DQh[65]
	1441. DQh[66]	1442. DQh[67]	1443. DQh[68]	1444. DQh[69]
	1445. DQh[70]	1446. DQh[71]	1447. DQh[72]	1448. DQh[73]
	1449. DQh[74]	1450. DQh[75]	1451. DQh[76]	1452. DQh[77]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	1453. DQh[78]	1454. DQh[79]	1455. DQh[80]	1456. DQh[81]
	1457. DQh[82]	1458. DQh[83]	1459. DQh[84]	1460. DQh[85]
	1461. DQh[86]	1462. DQh[87]	1463. DQh[88]	1464. DQh[89]
	1465. DQh[90]	1466. DQh[91]	1467. DQh[92]	1468. DQh[93]
	1469. DQh[94]	1470. DQh[95]	1471. DQh[96]	1472. DQh[97]
	1473. DQh[98]	1474. DQh[99]	1475. DQh[100]	1476. DQh[101]
	1477. DQh[102]	1478. DQh[103]	1479. DQh[104]	1480. DQh[105]
	1481. DQh[106]	1482. DQh[107]	1483. DQh[108]	1484. DQh[109]
	1485. DQh[110]	1486. DQh[111]	1487. DQh[112]	1488. DQh[113]
	1489. DQh[114]	1490. DQh[115]	1491. DQh[116]	1492. DQh[117]
	1493. DQh[118]	1494. DQh[119]	1495. DQh[120]	1496. DQh[121]
	1497. DQh[122]	1498. DQh[123]	1499. DQh[124]	1500. DQh[125]
	1501. DQh[126]	1502. DQh[127]	1503. MRFU[0]	1504. MRFU[1]
	1505. MRFU[2]	1506. MRFU[3]	1507. MRFU[4]	1508. MRFU[5]
	1509. MRFU[6]	1510. MRFU[7]	1511. MRFU[8]	1512. MRFU[9]
	1513. MRFU[10]	1514. MRFU[11]	1515. MRFU[12]	1516. MRFU[13]
	1517. MRFU[14]	1518. MRFU[15]	1519. MRFU[16]	1520. MRFU[17]
	1521. MRFU[18]	1522. MRFU[19]	1523. MRFU[20]	1524. MRFU[21]
	1525. MRFU[22]	1526. MRFU[23]	1527. PARa[0]	1528. PARa[1]
	1529. PARa[2]	1530. PARa[3]	1531. PARb[0]	1532. PARb[1]
	1533. PARb[2]	1534. PARb[3]	1535. PARc[0]	1536. PARc[1]
	1537. PARc[2]	1538. PARc[3]	1539. PARd[0]	1540. PARd[1]
	1541. PARd[2]	1542. PARd[3]	1543. PARe[0]	1544. PARe[1]
	1545. PARe[2]	1546. PARe[3]	1547. PARf[0]	1548. PARf[1]
	1549. PARf[2]	1550. PARf[3]	1551. PARg[0]	1552. PARg[1]
	1553. PARg[2]	1554. PARg[3]	1555. PARh[0]	1556. PARh[1]
	1557. PARh[2]	1558. PARh[3]	1559. Ra[0]	1560. Ra[1]
	1561. Ra[2]	1562. Ra[3]	1563. Ra[4]	1564. Ra[5]
	1565. Rb[0]	1566. Rb[1]	1567. Rb[2]	1568. Rb[3]
	1569. Rb[4]	1570. Rb[5]	1571. Rc[0]	1572. Rc[1]
	1573. Rc[2]	1574. Rc[3]	1575. Rc[4]	1576. Rc[5]
	1577. RC[a]	1578. RC[b]	1579. RC[c]	1580. RC[d]
	1581. RC[e]	1582. RC[f]	1583. RC[g]	1584. RC[h]
	1585. Rd[0]	1586. Rd[1]	1587. Rd[2]	1588. Rd[3]
	1589. Rd[4]	1590. Rd[5]	1591. RDa[0]	1592. RDa[1]
	1593. RDa[2]	1594. RDa[3]	1595. RDa[4]	1596. RDa[5]
	1597. RDa[6]	1598. RDa[7]	1599. RDb[0]	1600. RDb[1]
	1601. RDb[2]	1602. RDb[3]	1603. RDb[4]	1604. RDb[5]
	1605. RDb[6]	1606. RDb[7]	1607. RDc[0]	1608. RDc[1]
	1609. RDc[2]	1610. RDc[3]	1611. RDc[4]	1612. RDc[5]
	1613. RDc[6]	1614. RDc[7]	1615. RDd[0]	1616. RDd[1]
	1617. RDd[2]	1618. RDd[3]	1619. RDd[4]	1620. RDd[5]
	1621. RDd[6]	1622. RDd[7]	1623. RDe[0]	1624. RDe[1]
	1625. RDe[2]	1626. RDe[3]	1627. RDe[4]	1628. RDe[5]
	1629. RDe[6]	1630. RDe[7]	1631. RDf[0]	1632. RDf[1]
	1633. RDf[2]	1634. RDf[3]	1635. RDf[4]	1636. RDf[5]
	1637. RDf[6]	1638. RDf[7]	1639. RDg[0]	1640. RDg[1]
	1641. RDg[2]	1642. RDg[3]	1643. RDg[4]	1644. RDg[5]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	1645. DQh[78]	1646. DQh[79]	1647. DQh[80]	1648. DQh[81]
	1649. DQh[82]	1650. DQh[83]	1651. DQh[84]	1652. DQh[85]
	1653. DQh[86]	1654. DQh[87]	1655. DQh[88]	1656. DQh[89]
	1657. DQh[90]	1658. DQh[91]	1659. DQh[92]	1660. DQh[93]
	1661. DQh[94]	1662. DQh[95]	1663. DQh[96]	1664. DQh[97]
	1665. DQh[98]	1666. DQh[99]	1667. DQh[100]	1668. DQh[101]
	1669. DQh[102]	1670. DQh[103]	1671. DQh[104]	1672. DQh[105]
	1673. DQh[106]	1674. DQh[107]	1675. DQh[108]	1676. DQh[109]
	1677. DQh[110]	1678. DQh[111]	1679. DQh[112]	1680. DQh[113]
	1681. DQh[114]	1682. DQh[115]	1683. DQh[116]	1684. DQh[117]
	1685. DQh[118]	1686. DQh[119]	1687. DQh[120]	1688. DQh[121]
	1689. DQh[122]	1690. DQh[123]	1691. DQh[124]	1692. DQh[125]
	1693. DQh[126]	1694. DQh[127]	1695. MRFU[0]	1696. MRFU[1]
	1697. MRFU[2]	1698. MRFU[3]	1699. MRFU[4]	1700. MRFU[5]
	1701. MRFU[6]	1702. MRFU[7]	1703. MRFU[8]	1704. MRFU[9]
	1705. MRFU[10]	1706. MRFU[11]	1707. MRFU[12]	1708. MRFU[13]
	1709. MRFU[14]	1710. MRFU[15]	1711. MRFU[16]	1712. MRFU[17]
	1713. MRFU[18]	1714. MRFU[19]	1715. MRFU[20]	1716. MRFU[21]
	1717. MRFU[22]	1718. MRFU[23]	1719. PARa[0]	1720. PARa[1]
	1721. PARa[2]	1722. PARa[3]	1723. PARb[0]	1724. PARb[1]
	1725. PARb[2]	1726. PARb[3]	1727. PARc[0]	1728. PARc[1]
	1729. PARc[2]	1730. PARc[3]	1731. PARd[0]	1732. PARd[1]
	1733. PARd[2]	1734. PARd[3]	1735. PARe[0]	1736. PARe[1]
	1737. PARe[2]	1738. PARe[3]	1739. PARf[0]	1740. PARf[1]
	1741. PARf[2]	1742. PARf[3]	1743. PARg[0]	1744. PARg[1]
	1745. PARg[2]	1746. PARg[3]	1747. PARh[0]	1748. PARh[1]
	1749. PARh[2]	1750. PARh[3]	1751. Ra[0]	1752. Ra[1]
	1753. Ra[2]	1754. Ra[3]	1755. Ra[4]	1756. Ra[5]
	1757. Rb[0]	1758. Rb[1]	1759. Rb[2]	1760. Rb[3]
	1761. Rb[4]	1762. Rb[5]	1763. Rc[0]	1764. Rc[1]
	1765. Rc[2]	1766. Rc[3]	1767. Rc[4]	1768. Rc[5]
	1769. RC[a]	1770. RC[b]	1771. RC[c]	1772. RC[d]
	1773. RC[e]	1774. RC[f]	1775. RC[g]	1776. RC[h]
	1777. Rd[0]	1778. Rd[1]	1779. Rd[2]	1780. Rd[3]
	1781. Rd[4]	1782. Rd[5]	1783. RDa[0]	1784. RDa[1]
	1785. RDa[2]	1786. RDa[3]	1787. RDa[4]	1788. RDa[5]
	1789. RDa[6]	1790. RDa[7]	1791. RDb[0]	1792. RDb[1]
	1793. RDb[2]	1794. RDb[3]	1795. RDb[4]	1796. RDb[5]
	1797. RDb[6]	1798. RDb[7]	1799. RDc[0]	1800. RDc[1]
	1801. RDc[2]	1802. RDc[3]	1803. RDc[4]	1804. RDc[5]
	1805. RDc[6]	1806. RDc[7]	1807. RDd[0]	1808. RDd[1]
	1809. RDd[2]	1810. RDd[3]	1811. RDd[4]	1812. RDd[5]
	1813. RDd[6]	1814. RDd[7]	1815. RDe[0]	1816. RDe[1]
	1817. RDe[2]	1818. RDe[3]	1819. RDe[4]	1820. RDe[5]
	1821. RDe[6]	1822. RDe[7]	1823. RDf[0]	1824. RDf[1]
	1825. RDf[2]	1826. RDf[3]	1827. RDf[4]	1828. RDf[5]
	1829. RDf[6]	1830. RDf[7]	1831. RDg[0]	1832. RDg[1]
	1833. RDg[2]	1834. RDg[3]	1835. RDg[4]	1836. RDg[5]

4.7.3.10.17.1 HBM1, HBM2 and HBM2E Footprint A Interface Function (cont'd)

list of enumerate values (cont.)	1837. WSOc	1838. WSOd	1839. WSOe	1840. WSOf
	1841. WSOg	1842. WSOh		

4.7.3.10.17.2. HBM1, HBM2 and HBM2E Footprint B Interface Functions

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM1-FootprintB 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2-FootprintB 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2E-FootprintB 																																																																																																												
diagram	<pre> classDiagram class HBM1-2-2E-FootprintB-InterfaceFunctionType class HBM1-FootprintB class StandardTerminalNameAssignment class Mapping class StandardTerminalName class HBM1-2-2E-FootprintB-StandardTerminalNameAssignmentType class constraints HBM1-2-2E-FootprintB-InterfaceFunctionType "1..>" HBM1-FootprintB HBM1-2-2E-FootprintB-InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping StandardTerminalNameAssignment "1..>" StandardTerminalName Mapping "1..>" StandardTerminalName StandardTerminalNameAssignment "1..>" constraints constraints "1..>" StandardTerminalNameAssignment </pre>																																																																																																												
type	HBM1-2-2E-FootprintB-InterfaceFunctionType, HBM1-2-2E-FootprintB-StandardTerminalNameAssignmentType, HBM1-2-2E-FootprintB-StandardTerminalMappingType, HBM1-2-2E-FootprintB-StandardTerminalNameType																																																																																																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. AERRa</td><td>2. AERRb</td><td>3. AERRc</td><td>4. AERRd</td></tr> <tr><td>5. AERRe</td><td>6. AERRf</td><td>7. AERRg</td><td>8. AERRh</td></tr> <tr><td>9. ARFUa[1]</td><td>10. ARFUa[3]</td><td>11. ARFUb[1]</td><td>12. ARFUb[3]</td></tr> <tr><td>13. ARFUC[1]</td><td>14. ARFUC[3]</td><td>15. ARFUD[1]</td><td>16. ARFUD[3]</td></tr> <tr><td>17. ARFUE[1]</td><td>18. ARFUE[3]</td><td>19. ARFUF[1]</td><td>20. ARFUF[3]</td></tr> <tr><td>21. ARFUG[1]</td><td>22. ARFUG[3]</td><td>23. ARFUh[1]</td><td>24. ARFUh[3]</td></tr> <tr><td>25. CAPTUREWR</td><td>26. CATTRIP</td><td>27. Ca[0]</td><td>28. Ca[1]</td></tr> <tr><td>29. Ca[2]</td><td>30. Ca[3]</td><td>31. Ca[4]</td><td>32. Ca[5]</td></tr> <tr><td>33. Ca[6]</td><td>34. Ca[7]</td><td>35. Ca[8]</td><td>36. Cb[0]</td></tr> <tr><td>37. Cb[1]</td><td>38. Cb[2]</td><td>39. Cb[3]</td><td>40. Cb[4]</td></tr> <tr><td>41. Cb[5]</td><td>42. Cb[6]</td><td>43. Cb[7]</td><td>44. Cb[8]</td></tr> <tr><td>45. Cc[0]</td><td>46. Cc[1]</td><td>47. Cc[2]</td><td>48. Cc[3]</td></tr> <tr><td>49. Cc[4]</td><td>50. Cc[5]</td><td>51. Cc[6]</td><td>52. Cc[7]</td></tr> <tr><td>53. Cc[8]</td><td>54. Cd[0]</td><td>55. Cd[1]</td><td>56. Cd[2]</td></tr> <tr><td>57. Cd[3]</td><td>58. Cd[4]</td><td>59. Cd[5]</td><td>60. Cd[6]</td></tr> <tr><td>61. Cd[7]</td><td>62. Cd[8]</td><td>63. Ce[0]</td><td>64. Ce[1]</td></tr> <tr><td>65. Ce[2]</td><td>66. Ce[3]</td><td>67. Ce[4]</td><td>68. Ce[5]</td></tr> <tr><td>69. Ce[6]</td><td>70. Ce[7]</td><td>71. Ce[8]</td><td>72. Cf[0]</td></tr> <tr><td>73. Cf[1]</td><td>74. Cf[2]</td><td>75. Cf[3]</td><td>76. Cf[4]</td></tr> <tr><td>77. Cf[5]</td><td>78. Cf[6]</td><td>79. Cf[7]</td><td>80. Cf[8]</td></tr> <tr><td>81. Cg[0]</td><td>82. Cg[1]</td><td>83. Cg[2]</td><td>84. Cg[3]</td></tr> <tr><td>85. Cg[4]</td><td>86. Cg[5]</td><td>87. Cg[6]</td><td>88. Cg[7]</td></tr> <tr><td>89. Cg[8]</td><td>90. Ch[0]</td><td>91. Ch[1]</td><td>92. Ch[2]</td></tr> <tr><td>93. Ch[3]</td><td>94. Ch[4]</td><td>95. Ch[5]</td><td>96. Ch[6]</td></tr> <tr><td>97. Ch[7]</td><td>98. Ch[8]</td><td>99. CKa_c</td><td>100. CKa_t</td></tr> <tr><td>101. CKb_c</td><td>102. CKb_t</td><td>103. CKc_c</td><td>104. CKc_t</td></tr> <tr><td>105. CKd_c</td><td>106. CKd_t</td><td>107. CKe_c</td><td>108. CKe_t</td></tr> </tbody> </table>	1. AERRa	2. AERRb	3. AERRc	4. AERRd	5. AERRe	6. AERRf	7. AERRg	8. AERRh	9. ARFUa[1]	10. ARFUa[3]	11. ARFUb[1]	12. ARFUb[3]	13. ARFUC[1]	14. ARFUC[3]	15. ARFUD[1]	16. ARFUD[3]	17. ARFUE[1]	18. ARFUE[3]	19. ARFUF[1]	20. ARFUF[3]	21. ARFUG[1]	22. ARFUG[3]	23. ARFUh[1]	24. ARFUh[3]	25. CAPTUREWR	26. CATTRIP	27. Ca[0]	28. Ca[1]	29. Ca[2]	30. Ca[3]	31. Ca[4]	32. Ca[5]	33. Ca[6]	34. Ca[7]	35. Ca[8]	36. Cb[0]	37. Cb[1]	38. Cb[2]	39. Cb[3]	40. Cb[4]	41. Cb[5]	42. Cb[6]	43. Cb[7]	44. Cb[8]	45. Cc[0]	46. Cc[1]	47. Cc[2]	48. Cc[3]	49. Cc[4]	50. Cc[5]	51. Cc[6]	52. Cc[7]	53. Cc[8]	54. Cd[0]	55. Cd[1]	56. Cd[2]	57. Cd[3]	58. Cd[4]	59. Cd[5]	60. Cd[6]	61. Cd[7]	62. Cd[8]	63. Ce[0]	64. Ce[1]	65. Ce[2]	66. Ce[3]	67. Ce[4]	68. Ce[5]	69. Ce[6]	70. Ce[7]	71. Ce[8]	72. Cf[0]	73. Cf[1]	74. Cf[2]	75. Cf[3]	76. Cf[4]	77. Cf[5]	78. Cf[6]	79. Cf[7]	80. Cf[8]	81. Cg[0]	82. Cg[1]	83. Cg[2]	84. Cg[3]	85. Cg[4]	86. Cg[5]	87. Cg[6]	88. Cg[7]	89. Cg[8]	90. Ch[0]	91. Ch[1]	92. Ch[2]	93. Ch[3]	94. Ch[4]	95. Ch[5]	96. Ch[6]	97. Ch[7]	98. Ch[8]	99. CKa_c	100. CKa_t	101. CKb_c	102. CKb_t	103. CKc_c	104. CKc_t	105. CKd_c	106. CKd_t	107. CKe_c	108. CKe_t
1. AERRa	2. AERRb	3. AERRc	4. AERRd																																																																																																										
5. AERRe	6. AERRf	7. AERRg	8. AERRh																																																																																																										
9. ARFUa[1]	10. ARFUa[3]	11. ARFUb[1]	12. ARFUb[3]																																																																																																										
13. ARFUC[1]	14. ARFUC[3]	15. ARFUD[1]	16. ARFUD[3]																																																																																																										
17. ARFUE[1]	18. ARFUE[3]	19. ARFUF[1]	20. ARFUF[3]																																																																																																										
21. ARFUG[1]	22. ARFUG[3]	23. ARFUh[1]	24. ARFUh[3]																																																																																																										
25. CAPTUREWR	26. CATTRIP	27. Ca[0]	28. Ca[1]																																																																																																										
29. Ca[2]	30. Ca[3]	31. Ca[4]	32. Ca[5]																																																																																																										
33. Ca[6]	34. Ca[7]	35. Ca[8]	36. Cb[0]																																																																																																										
37. Cb[1]	38. Cb[2]	39. Cb[3]	40. Cb[4]																																																																																																										
41. Cb[5]	42. Cb[6]	43. Cb[7]	44. Cb[8]																																																																																																										
45. Cc[0]	46. Cc[1]	47. Cc[2]	48. Cc[3]																																																																																																										
49. Cc[4]	50. Cc[5]	51. Cc[6]	52. Cc[7]																																																																																																										
53. Cc[8]	54. Cd[0]	55. Cd[1]	56. Cd[2]																																																																																																										
57. Cd[3]	58. Cd[4]	59. Cd[5]	60. Cd[6]																																																																																																										
61. Cd[7]	62. Cd[8]	63. Ce[0]	64. Ce[1]																																																																																																										
65. Ce[2]	66. Ce[3]	67. Ce[4]	68. Ce[5]																																																																																																										
69. Ce[6]	70. Ce[7]	71. Ce[8]	72. Cf[0]																																																																																																										
73. Cf[1]	74. Cf[2]	75. Cf[3]	76. Cf[4]																																																																																																										
77. Cf[5]	78. Cf[6]	79. Cf[7]	80. Cf[8]																																																																																																										
81. Cg[0]	82. Cg[1]	83. Cg[2]	84. Cg[3]																																																																																																										
85. Cg[4]	86. Cg[5]	87. Cg[6]	88. Cg[7]																																																																																																										
89. Cg[8]	90. Ch[0]	91. Ch[1]	92. Ch[2]																																																																																																										
93. Ch[3]	94. Ch[4]	95. Ch[5]	96. Ch[6]																																																																																																										
97. Ch[7]	98. Ch[8]	99. CKa_c	100. CKa_t																																																																																																										
101. CKb_c	102. CKb_t	103. CKc_c	104. CKc_t																																																																																																										
105. CKd_c	106. CKd_t	107. CKe_c	108. CKe_t																																																																																																										
path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM1-FootprintB 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2-FootprintB 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2E-FootprintB 																																																																																																												
diagram	<pre> classDiagram class HBM1-2-2E-FootprintB-InterfaceFunctionType class HBM1-FootprintB class StandardTerminalNameAssignment class Mapping class StandardTerminalName class HBM1-2-2E-FootprintB-StandardTerminalNameAssignmentType class constraints HBM1-2-2E-FootprintB-InterfaceFunctionType "1..>" HBM1-FootprintB HBM1-2-2E-FootprintB-InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping StandardTerminalNameAssignment "1..>" StandardTerminalName Mapping "1..>" StandardTerminalName StandardTerminalNameAssignment "1..>" constraints constraints "1..>" StandardTerminalNameAssignment </pre>																																																																																																												
type	HBM1-2-2E-FootprintB-InterfaceFunctionType, HBM1-2-2E-FootprintB-StandardTerminalNameAssignmentType, HBM1-2-2E-FootprintB-StandardTerminalMappingType, HBM1-2-2E-FootprintB-StandardTerminalNameType																																																																																																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. AERRa</td><td>2. AERRb</td><td>3. AERRc</td><td>4. AERRd</td></tr> <tr><td>5. AERRe</td><td>6. AERRf</td><td>7. AERRg</td><td>8. AERRh</td></tr> <tr><td>9. ARFUa[1]</td><td>10. ARFUa[3]</td><td>11. ARFUb[1]</td><td>12. ARFUb[3]</td></tr> <tr><td>13. ARFUC[1]</td><td>14. ARFUC[3]</td><td>15. ARFUD[1]</td><td>16. ARFUD[3]</td></tr> <tr><td>17. ARFUE[1]</td><td>18. ARFUE[3]</td><td>19. ARFUF[1]</td><td>20. ARFUF[3]</td></tr> <tr><td>21. ARFUG[1]</td><td>22. ARFUG[3]</td><td>23. ARFUh[1]</td><td>24. ARFUh[3]</td></tr> <tr><td>25. CAPTUREWR</td><td>26. CATTRIP</td><td>27. Ca[0]</td><td>28. Ca[1]</td></tr> <tr><td>29. Ca[2]</td><td>30. Ca[3]</td><td>31. Ca[4]</td><td>32. Ca[5]</td></tr> <tr><td>33. Ca[6]</td><td>34. Ca[7]</td><td>35. Ca[8]</td><td>36. Cb[0]</td></tr> <tr><td>37. Cb[1]</td><td>38. Cb[2]</td><td>39. Cb[3]</td><td>40. Cb[4]</td></tr> <tr><td>41. Cb[5]</td><td>42. Cb[6]</td><td>43. Cb[7]</td><td>44. Cb[8]</td></tr> <tr><td>45. Cc[0]</td><td>46. Cc[1]</td><td>47. Cc[2]</td><td>48. Cc[3]</td></tr> <tr><td>49. Cc[4]</td><td>50. Cc[5]</td><td>51. Cc[6]</td><td>52. Cc[7]</td></tr> <tr><td>53. Cc[8]</td><td>54. Cd[0]</td><td>55. Cd[1]</td><td>56. Cd[2]</td></tr> <tr><td>57. Cd[3]</td><td>58. Cd[4]</td><td>59. Cd[5]</td><td>60. Cd[6]</td></tr> <tr><td>61. Cd[7]</td><td>62. Cd[8]</td><td>63. Ce[0]</td><td>64. Ce[1]</td></tr> <tr><td>65. Ce[2]</td><td>66. Ce[3]</td><td>67. Ce[4]</td><td>68. Ce[5]</td></tr> <tr><td>69. Ce[6]</td><td>70. Ce[7]</td><td>71. Ce[8]</td><td>72. Cf[0]</td></tr> <tr><td>73. Cf[1]</td><td>74. Cf[2]</td><td>75. Cf[3]</td><td>76. Cf[4]</td></tr> <tr><td>77. Cf[5]</td><td>78. Cf[6]</td><td>79. Cf[7]</td><td>80. Cf[8]</td></tr> <tr><td>81. Cg[0]</td><td>82. Cg[1]</td><td>83. Cg[2]</td><td>84. Cg[3]</td></tr> <tr><td>85. Cg[4]</td><td>86. Cg[5]</td><td>87. Cg[6]</td><td>88. Cg[7]</td></tr> <tr><td>89. Cg[8]</td><td>90. Ch[0]</td><td>91. Ch[1]</td><td>92. Ch[2]</td></tr> <tr><td>93. Ch[3]</td><td>94. Ch[4]</td><td>95. Ch[5]</td><td>96. Ch[6]</td></tr> <tr><td>97. Ch[7]</td><td>98. Ch[8]</td><td>99. CKa_c</td><td>100. CKa_t</td></tr> <tr><td>101. CKb_c</td><td>102. CKb_t</td><td>103. CKc_c</td><td>104. CKc_t</td></tr> <tr><td>105. CKd_c</td><td>106. CKd_t</td><td>107. CKe_c</td><td>108. CKe_t</td></tr> </tbody> </table>	1. AERRa	2. AERRb	3. AERRc	4. AERRd	5. AERRe	6. AERRf	7. AERRg	8. AERRh	9. ARFUa[1]	10. ARFUa[3]	11. ARFUb[1]	12. ARFUb[3]	13. ARFUC[1]	14. ARFUC[3]	15. ARFUD[1]	16. ARFUD[3]	17. ARFUE[1]	18. ARFUE[3]	19. ARFUF[1]	20. ARFUF[3]	21. ARFUG[1]	22. ARFUG[3]	23. ARFUh[1]	24. ARFUh[3]	25. CAPTUREWR	26. CATTRIP	27. Ca[0]	28. Ca[1]	29. Ca[2]	30. Ca[3]	31. Ca[4]	32. Ca[5]	33. Ca[6]	34. Ca[7]	35. Ca[8]	36. Cb[0]	37. Cb[1]	38. Cb[2]	39. Cb[3]	40. Cb[4]	41. Cb[5]	42. Cb[6]	43. Cb[7]	44. Cb[8]	45. Cc[0]	46. Cc[1]	47. Cc[2]	48. Cc[3]	49. Cc[4]	50. Cc[5]	51. Cc[6]	52. Cc[7]	53. Cc[8]	54. Cd[0]	55. Cd[1]	56. Cd[2]	57. Cd[3]	58. Cd[4]	59. Cd[5]	60. Cd[6]	61. Cd[7]	62. Cd[8]	63. Ce[0]	64. Ce[1]	65. Ce[2]	66. Ce[3]	67. Ce[4]	68. Ce[5]	69. Ce[6]	70. Ce[7]	71. Ce[8]	72. Cf[0]	73. Cf[1]	74. Cf[2]	75. Cf[3]	76. Cf[4]	77. Cf[5]	78. Cf[6]	79. Cf[7]	80. Cf[8]	81. Cg[0]	82. Cg[1]	83. Cg[2]	84. Cg[3]	85. Cg[4]	86. Cg[5]	87. Cg[6]	88. Cg[7]	89. Cg[8]	90. Ch[0]	91. Ch[1]	92. Ch[2]	93. Ch[3]	94. Ch[4]	95. Ch[5]	96. Ch[6]	97. Ch[7]	98. Ch[8]	99. CKa_c	100. CKa_t	101. CKb_c	102. CKb_t	103. CKc_c	104. CKc_t	105. CKd_c	106. CKd_t	107. CKe_c	108. CKe_t
1. AERRa	2. AERRb	3. AERRc	4. AERRd																																																																																																										
5. AERRe	6. AERRf	7. AERRg	8. AERRh																																																																																																										
9. ARFUa[1]	10. ARFUa[3]	11. ARFUb[1]	12. ARFUb[3]																																																																																																										
13. ARFUC[1]	14. ARFUC[3]	15. ARFUD[1]	16. ARFUD[3]																																																																																																										
17. ARFUE[1]	18. ARFUE[3]	19. ARFUF[1]	20. ARFUF[3]																																																																																																										
21. ARFUG[1]	22. ARFUG[3]	23. ARFUh[1]	24. ARFUh[3]																																																																																																										
25. CAPTUREWR	26. CATTRIP	27. Ca[0]	28. Ca[1]																																																																																																										
29. Ca[2]	30. Ca[3]	31. Ca[4]	32. Ca[5]																																																																																																										
33. Ca[6]	34. Ca[7]	35. Ca[8]	36. Cb[0]																																																																																																										
37. Cb[1]	38. Cb[2]	39. Cb[3]	40. Cb[4]																																																																																																										
41. Cb[5]	42. Cb[6]	43. Cb[7]	44. Cb[8]																																																																																																										
45. Cc[0]	46. Cc[1]	47. Cc[2]	48. Cc[3]																																																																																																										
49. Cc[4]	50. Cc[5]	51. Cc[6]	52. Cc[7]																																																																																																										
53. Cc[8]	54. Cd[0]	55. Cd[1]	56. Cd[2]																																																																																																										
57. Cd[3]	58. Cd[4]	59. Cd[5]	60. Cd[6]																																																																																																										
61. Cd[7]	62. Cd[8]	63. Ce[0]	64. Ce[1]																																																																																																										
65. Ce[2]	66. Ce[3]	67. Ce[4]	68. Ce[5]																																																																																																										
69. Ce[6]	70. Ce[7]	71. Ce[8]	72. Cf[0]																																																																																																										
73. Cf[1]	74. Cf[2]	75. Cf[3]	76. Cf[4]																																																																																																										
77. Cf[5]	78. Cf[6]	79. Cf[7]	80. Cf[8]																																																																																																										
81. Cg[0]	82. Cg[1]	83. Cg[2]	84. Cg[3]																																																																																																										
85. Cg[4]	86. Cg[5]	87. Cg[6]	88. Cg[7]																																																																																																										
89. Cg[8]	90. Ch[0]	91. Ch[1]	92. Ch[2]																																																																																																										
93. Ch[3]	94. Ch[4]	95. Ch[5]	96. Ch[6]																																																																																																										
97. Ch[7]	98. Ch[8]	99. CKa_c	100. CKa_t																																																																																																										
101. CKb_c	102. CKb_t	103. CKc_c	104. CKc_t																																																																																																										
105. CKd_c	106. CKd_t	107. CKe_c	108. CKe_t																																																																																																										

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	109. CKEa	110. CKEb	111. CKEc	112. CKEd
	113. CKEe	114. CKEf	115. CKEg	116. CKEh
	117. CKf_c	118. CKf_t	119. CKg_c	120. CKg_t
	121. CKh_c	122. CKh_t	123. DA[0]	124. DA[1]
	125. DA[2]	126. DA[3]	127. DA[4]	128. DA[5]
	129. DA[6]	130. DA[7]	131. DA[8]	132. DA[9]
	133. DA[10]	134. DA[11]	135. DA[12]	136. DA[13]
	137. DA[14]	138. DA[15]	139. DA[16]	140. DA[17]
	141. DA[18]	142. DA[19]	143. DA[20]	144. DA[21]
	145. DA[22]	146. DA[23]	147. DA[24]	148. DA[25]
	149. DA[26]	150. DA[27]	151. DA[28]	152. DA[29]
	153. DA[30]	154. DA[31]	155. DA[32]	156. DA[33]
	157. DA[34]	158. DA[35]	159. DA[36]	160. DA[37]
	161. DA[38]	162. DA[39]	163. DA[40]	164. DA[41]
	165. DA[42]	166. DA[43]	167. DA[44]	168. DA[45]
	169. DA[46]	170. DA[47]	171. DA[48]	172. DA[49]
	173. DA[50]	174. DA[51]	175. DA[52]	176. DA[53]
	177. DA[54]	178. DA[55]	179. DA[56]	180. DA[57]
	181. DA[58]	182. DA[59]	183. DBla[0]	184. DBla[1]
	185. DBla[2]	186. DBla[3]	187. DBla[4]	188. DBla[5]
	189. DBla[6]	190. DBla[7]	191. DBla[8]	192. DBla[9]
	193. DBla[10]	194. DBla[11]	195. DBla[12]	196. DBla[13]
	197. DBla[14]	198. DBla[15]	199. DBlb[0]	200. DBlb[1]
	201. DBlb[2]	202. DBlb[3]	203. DBlb[4]	204. DBlb[5]
	205. DBlb[6]	206. DBlb[7]	207. DBlb[8]	208. DBlb[9]
	209. DBlb[10]	210. DBlb[11]	211. DBlb[12]	212. DBlb[13]
	213. DBlb[14]	214. DBlb[15]	215. DBlc[0]	216. DBlc[1]
	217. DBlc[2]	218. DBlc[3]	219. DBlc[4]	220. DBlc[5]
	221. DBlc[6]	222. DBlc[7]	223. DBlc[8]	224. DBlc[9]
	225. DBlc[10]	226. DBlc[11]	227. DBlc[12]	228. DBlc[13]
	229. DBlc[14]	230. DBlc[15]	231. DBld[0]	232. DBld[1]
	233. DBld[2]	234. DBld[3]	235. DBld[4]	236. DBld[5]
	237. DBld[6]	238. DBld[7]	239. DBld[8]	240. DBld[9]
	241. DBld[10]	242. DBld[11]	243. DBld[12]	244. DBld[13]
	245. DBld[14]	246. DBld[15]	247. DBle[0]	248. DBle[1]
	249. DBle[2]	250. DBle[3]	251. DBle[4]	252. DBle[5]
	253. DBle[6]	254. DBle[7]	255. DBle[8]	256. DBle[9]
	257. DBle[10]	258. DBle[11]	259. DBle[12]	260. DBle[13]
	261. DBle[14]	262. DBle[15]	263. DBlf[0]	264. DBlf[1]
	265. DBlf[2]	266. DBlf[3]	267. DBlf[4]	268. DBlf[5]
	269. DBlf[6]	270. DBlf[7]	271. DBlf[8]	272. DBlf[9]
	273. DBlf[10]	274. DBlf[11]	275. DBlf[12]	276. DBlf[13]
	277. DBlf[14]	278. DBlf[15]	279. DBlg[0]	280. DBlg[1]
	281. DBlg[2]	282. DBlg[3]	283. DBlg[4]	284. DBlg[5]
	285. DBlg[6]	286. DBlg[7]	287. DBlg[8]	288. DBlg[9]
	289. DBlg[10]	290. DBlg[11]	291. DBlg[12]	292. DBlg[13]
	293. DBlg[14]	294. DBlg[15]	295. DBlh[0]	296. DBlh[1]
	297. DBlh[2]	298. DBlh[3]	299. DBlh[4]	300. DBlh[5]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	301. DBIh[6]	302. DBIh[7]	303. DBIh[8]	304. DBIh[9]
	305. DBIh[10]	306. DBIh[11]	307. DBIh[12]	308. DBIh[13]
	309. DBIh[14]	310. DBIh[15]	311. DERRa[0]	312. DERRa[1]
	313. DERRa[2]	314. DERRa[3]	315. DERRb[0]	316. DERRb[1]
	317. DERRb[2]	318. DERRb[3]	319. DERRc[0]	320. DERRc[1]
	321. DERRc[2]	322. DERRc[3]	323. DERRd[0]	324. DERRd[1]
	325. DERRd[2]	326. DERRd[3]	327. DERRe[0]	328. DERRe[1]
	329. DERRe[2]	330. DERRe[3]	331. DERRf[0]	332. DERRf[1]
	333. DERRf[2]	334. DERRf[3]	335. DERRg[0]	336. DERRg[1]
	337. DERRg[2]	338. DERRg[3]	339. DERRh[0]	340. DERRh[1]
	341. DERRh[2]	342. DERRh[3]	343. DMA[0]	344. DMA[1]
	345. DMA[2]	346. DMA[3]	347. DMA[4]	348. DMA[5]
	349. DMA[6]	350. DMA[7]	351. DMA[8]	352. DMA[9]
	353. DMA[10]	354. DMA[11]	355. DMA[12]	356. DMA[13]
	357. DMA[14]	358. DMA[15]	359. DMb[0]	360. DMb[1]
	361. DMb[2]	362. DMb[3]	363. DMb[4]	364. DMb[5]
	365. DMb[6]	366. DMb[7]	367. DMb[8]	368. DMb[9]
	369. DMb[10]	370. DMb[11]	371. DMb[12]	372. DMb[13]
	373. DMb[14]	374. DMb[15]	375. DMc[0]	376. DMc[1]
	377. DMc[2]	378. DMc[3]	379. DMc[4]	380. DMc[5]
	381. DMc[6]	382. DMc[7]	383. DMc[8]	384. DMc[9]
	385. DMc[10]	386. DMc[11]	387. DMc[12]	388. DMc[13]
	389. DMc[14]	390. DMc[15]	391. DMd[0]	392. DMd[1]
	393. DMd[2]	394. DMd[3]	395. DMd[4]	396. DMd[5]
	397. DMd[6]	398. DMd[7]	399. DMd[8]	400. DMd[9]
	401. DMd[10]	402. DMd[11]	403. DMd[12]	404. DMd[13]
	405. DMd[14]	406. DMd[15]	407. DMe[0]	408. DMe[1]
	409. DMe[2]	410. DMe[3]	411. DMe[4]	412. DMe[5]
	413. DMe[6]	414. DMe[7]	415. DMe[8]	416. DMe[9]
	417. DMe[10]	418. DMe[11]	419. DMe[12]	420. DMe[13]
	421. DMe[14]	422. DMe[15]	423. DMf[0]	424. DMf[1]
	425. DMf[2]	426. DMf[3]	427. DMf[4]	428. DMf[5]
	429. DMf[6]	430. DMf[7]	431. DMf[8]	432. DMf[9]
	433. DMf[10]	434. DMf[11]	435. DMf[12]	436. DMf[13]
	437. DMf[14]	438. DMf[15]	439. DMg[0]	440. DMg[1]
	441. DMg[2]	442. DMg[3]	443. DMg[4]	444. DMg[5]
	445. DMg[6]	446. DMg[7]	447. DMg[8]	448. DMg[9]
	449. DMg[10]	450. DMg[11]	451. DMg[12]	452. DMg[13]
	453. DMg[14]	454. DMg[15]	455. DMh[0]	456. DMh[1]
	457. DMh[2]	458. DMh[3]	459. DMh[4]	460. DMh[5]
	461. DMh[6]	462. DMh[7]	463. DMh[8]	464. DMh[9]
	465. DMh[10]	466. DMh[11]	467. DMh[12]	468. DMh[13]
	469. DMh[14]	470. DMh[15]	471. DQa[0]	472. DQa[1]
	473. DQa[2]	474. DQa[3]	475. DQa[4]	476. DQa[5]
	477. DQa[6]	478. DQa[7]	479. DQa[8]	480. DQa[9]
	481. DQa[10]	482. DQa[11]	483. DQa[12]	484. DQa[13]
	485. DQa[14]	486. DQa[15]	487. DQa[16]	488. DQa[17]
	489. DQa[18]	490. DQa[19]	491. DQa[20]	492. DQa[21]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	493. DQa[22]	494. DQa[23]	495. DQa[24]	496. DQa[25]
	497. DQa[26]	498. DQa[27]	499. DQa[28]	500. DQa[29]
	501. DQa[30]	502. DQa[31]	503. DQa[32]	504. DQa[33]
	505. DQa[34]	506. DQa[35]	507. DQa[36]	508. DQa[37]
	509. DQa[38]	510. DQa[39]	511. DQa[40]	512. DQa[41]
	513. DQa[42]	514. DQa[43]	515. DQa[44]	516. DQa[45]
	517. DQa[46]	518. DQa[47]	519. DQa[48]	520. DQa[49]
	521. DQa[50]	522. DQa[51]	523. DQa[52]	524. DQa[53]
	525. DQa[54]	526. DQa[55]	527. DQa[56]	528. DQa[57]
	529. DQa[58]	530. DQa[59]	531. DQa[60]	532. DQa[61]
	533. DQa[62]	534. DQa[63]	535. DQa[64]	536. DQa[65]
	537. DQa[66]	538. DQa[67]	539. DQa[68]	540. DQa[69]
	541. DQa[70]	542. DQa[71]	543. DQa[72]	544. DQa[73]
	545. DQa[74]	546. DQa[75]	547. DQa[76]	548. DQa[77]
	549. DQa[78]	550. DQa[79]	551. DQa[80]	552. DQa[81]
	553. DQa[82]	554. DQa[83]	555. DQa[84]	556. DQa[85]
	557. DQa[86]	558. DQa[87]	559. DQa[88]	560. DQa[89]
	561. DQa[90]	562. DQa[91]	563. DQa[92]	564. DQa[93]
	565. DQa[94]	566. DQa[95]	567. DQa[96]	568. DQa[97]
	569. DQa[98]	570. DQa[99]	571. DQa[100]	572. DQa[101]
	573. DQa[102]	574. DQa[103]	575. DQa[104]	576. DQa[105]
	577. DQa[106]	578. DQa[107]	579. DQa[108]	580. DQa[109]
	581. DQa[110]	582. DQa[111]	583. DQa[112]	584. DQa[113]
	585. DQa[114]	586. DQa[115]	587. DQa[116]	588. DQa[117]
	589. DQa[118]	590. DQa[119]	591. DQa[120]	592. DQa[121]
	593. DQa[122]	594. DQa[123]	595. DQa[124]	596. DQa[125]
	597. DQa[126]	598. DQa[127]	599. DQb[0]	600. DQb[1]
	601. DQb[2]	602. DQb[3]	603. DQb[4]	604. DQb[5]
	605. DQb[6]	606. DQb[7]	607. DQb[8]	608. DQb[9]
	609. DQb[10]	610. DQb[11]	611. DQb[12]	612. DQb[13]
	613. DQb[14]	614. DQb[15]	615. DQb[16]	616. DQb[17]
	617. DQb[18]	618. DQb[19]	619. DQb[20]	620. DQb[21]
	621. DQb[22]	622. DQb[23]	623. DQb[24]	624. DQb[25]
	625. DQb[26]	626. DQb[27]	627. DQb[28]	628. DQb[29]
	629. DQb[30]	630. DQb[31]	631. DQb[32]	632. DQb[33]
	633. DQb[34]	634. DQb[35]	635. DQb[36]	636. DQb[37]
	637. DQb[38]	638. DQb[39]	639. DQb[40]	640. DQb[41]
	641. DQb[42]	642. DQb[43]	643. DQb[44]	644. DQb[45]
	645. DQb[46]	646. DQb[47]	647. DQb[48]	648. DQb[49]
	649. DQb[50]	650. DQb[51]	651. DQb[52]	652. DQb[53]
	653. DQb[54]	654. DQb[55]	655. DQb[56]	656. DQb[57]
	657. DQb[58]	658. DQb[59]	659. DQb[60]	660. DQb[61]
	661. DQb[62]	662. DQb[63]	663. DQb[64]	664. DQb[65]
	665. DQb[66]	666. DQb[67]	667. DQb[68]	668. DQb[69]
	669. DQb[70]	670. DQb[71]	671. DQb[72]	672. DQb[73]
	673. DQb[74]	674. DQb[75]	675. DQb[76]	676. DQb[77]
	677. DQb[78]	678. DQb[79]	679. DQb[80]	680. DQb[81]
	681. DQb[82]	682. DQb[83]	683. DQb[84]	684. DQb[85]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	685. DQb[86]	686. DQb[87]	687. DQb[88]	688. DQb[89]
	689. DQb[90]	690. DQb[91]	691. DQb[92]	692. DQb[93]
	693. DQb[94]	694. DQb[95]	695. DQb[96]	696. DQb[97]
	697. DQb[98]	698. DQb[99]	699. DQb[100]	700. DQb[101]
	701. DQb[102]	702. DQb[103]	703. DQb[104]	704. DQb[105]
	705. DQb[106]	706. DQb[107]	707. DQb[108]	708. DQb[109]
	709. DQb[110]	710. DQb[111]	711. DQb[112]	712. DQb[113]
	713. DQb[114]	714. DQb[115]	715. DQb[116]	716. DQb[117]
	717. DQb[118]	718. DQb[119]	719. DQb[120]	720. DQb[121]
	721. DQb[122]	722. DQb[123]	723. DQb[124]	724. DQb[125]
	725. DQb[126]	726. DQb[127]	727. DQc[0]	728. DQc[1]
	729. DQc[2]	730. DQc[3]	731. DQc[4]	732. DQc[5]
	733. DQc[6]	734. DQc[7]	735. DQc[8]	736. DQc[9]
	737. DQc[10]	738. DQc[11]	739. DQc[12]	740. DQc[13]
	741. DQc[14]	742. DQc[15]	743. DQc[16]	744. DQc[17]
	745. DQc[18]	746. DQc[19]	747. DQc[20]	748. DQc[21]
	749. DQc[22]	750. DQc[23]	751. DQc[24]	752. DQc[25]
	753. DQc[26]	754. DQc[27]	755. DQc[28]	756. DQc[29]
	757. DQc[30]	758. DQc[31]	759. DQc[32]	760. DQc[33]
	761. DQc[34]	762. DQc[35]	763. DQc[36]	764. DQc[37]
	765. DQc[38]	766. DQc[39]	767. DQc[40]	768. DQc[41]
	769. DQc[42]	770. DQc[43]	771. DQc[44]	772. DQc[45]
	773. DQc[46]	774. DQc[47]	775. DQc[48]	776. DQc[49]
	777. DQc[50]	778. DQc[51]	779. DQc[52]	780. DQc[53]
	781. DQc[54]	782. DQc[55]	783. DQc[56]	784. DQc[57]
	785. DQc[58]	786. DQc[59]	787. DQc[60]	788. DQc[61]
	789. DQc[62]	790. DQc[63]	791. DQc[64]	792. DQc[65]
	793. DQc[66]	794. DQc[67]	795. DQc[68]	796. DQc[69]
	797. DQc[70]	798. DQc[71]	799. DQc[72]	800. DQc[73]
	801. DQc[74]	802. DQc[75]	803. DQc[76]	804. DQc[77]
	805. DQc[78]	806. DQc[79]	807. DQc[80]	808. DQc[81]
	809. DQc[82]	810. DQc[83]	811. DQc[84]	812. DQc[85]
	813. DQc[86]	814. DQc[87]	815. DQc[88]	816. DQc[89]
	817. DQc[90]	818. DQc[91]	819. DQc[92]	820. DQc[93]
	821. DQc[94]	822. DQc[95]	823. DQc[96]	824. DQc[97]
	825. DQc[98]	826. DQc[99]	827. DQc[100]	828. DQc[101]
	829. DQc[102]	830. DQc[103]	831. DQc[104]	832. DQc[105]
	833. DQc[106]	834. DQc[107]	835. DQc[108]	836. DQc[109]
	837. DQc[110]	838. DQc[111]	839. DQc[112]	840. DQc[113]
	841. DQc[114]	842. DQc[115]	843. DQc[116]	844. DQc[117]
	845. DQc[118]	846. DQc[119]	847. DQc[120]	848. DQc[121]
	849. DQc[122]	850. DQc[123]	851. DQc[124]	852. DQc[125]
	853. DQc[126]	854. DQc[127]	855. DQd[0]	856. DQd[1]
	857. DQd[2]	858. DQd[3]	859. DQd[4]	860. DQd[5]
	861. DQd[6]	862. DQd[7]	863. DQd[8]	864. DQd[9]
	865. DQd[10]	866. DQd[11]	867. DQd[12]	868. DQd[13]
	869. DQd[14]	870. DQd[15]	871. DQd[16]	872. DQd[17]
	873. DQd[18]	874. DQd[19]	875. DQd[20]	876. DQd[21]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	877. DQd[22]	878. DQd[23]	879. DQd[24]	880. DQd[25]
	881. DQd[26]	882. DQd[27]	883. DQd[28]	884. DQd[29]
	885. DQd[30]	886. DQd[31]	887. DQd[32]	888. DQd[33]
	889. DQd[34]	890. DQd[35]	891. DQd[36]	892. DQd[37]
	893. DQd[38]	894. DQd[39]	895. DQd[40]	896. DQd[41]
	897. DQd[42]	898. DQd[43]	899. DQd[44]	900. DQd[45]
	901. DQd[46]	902. DQd[47]	903. DQd[48]	904. DQd[49]
	905. DQd[50]	906. DQd[51]	907. DQd[52]	908. DQd[53]
	909. DQd[54]	910. DQd[55]	911. DQd[56]	912. DQd[57]
	913. DQd[58]	914. DQd[59]	915. DQd[60]	916. DQd[61]
	917. DQd[62]	918. DQd[63]	919. DQd[64]	920. DQd[65]
	921. DQd[66]	922. DQd[67]	923. DQd[68]	924. DQd[69]
	925. DQd[70]	926. DQd[71]	927. DQd[72]	928. DQd[73]
	929. DQd[74]	930. DQd[75]	931. DQd[76]	932. DQd[77]
	933. DQd[78]	934. DQd[79]	935. DQd[80]	936. DQd[81]
	937. DQd[82]	938. DQd[83]	939. DQd[84]	940. DQd[85]
	941. DQd[86]	942. DQd[87]	943. DQd[88]	944. DQd[89]
	945. DQd[90]	946. DQd[91]	947. DQd[92]	948. DQd[93]
	949. DQd[94]	950. DQd[95]	951. DQd[96]	952. DQd[97]
	953. DQd[98]	954. DQd[99]	955. DQd[100]	956. DQd[101]
	957. DQd[102]	958. DQd[103]	959. DQd[104]	960. DQd[105]
	961. DQd[106]	962. DQd[107]	963. DQd[108]	964. DQd[109]
	965. DQd[110]	966. DQd[111]	967. DQd[112]	968. DQd[113]
	969. DQd[114]	970. DQd[115]	971. DQd[116]	972. DQd[117]
	973. DQd[118]	974. DQd[119]	975. DQd[120]	976. DQd[121]
	977. DQd[122]	978. DQd[123]	979. DQd[124]	980. DQd[125]
	981. DQd[126]	982. DQd[127]	983. DQe[0]	984. DQe[1]
	985. DQe[2]	986. DQe[3]	987. DQe[4]	988. DQe[5]
	989. DQe[6]	990. DQe[7]	991. DQe[8]	992. DQe[9]
	993. DQe[10]	994. DQe[11]	995. DQe[12]	996. DQe[13]
	997. DQe[14]	998. DQe[15]	999. DQe[16]	1000. DQe[17]
	1001. DQe[18]	1002. DQe[19]	1003. DQe[20]	1004. DQe[21]
	1005. DQe[22]	1006. DQe[23]	1007. DQe[24]	1008. DQe[25]
	1009. DQe[26]	1010. DQe[27]	1011. DQe[28]	1012. DQe[29]
	1013. DQe[30]	1014. DQe[31]	1015. DQe[32]	1016. DQe[33]
	1017. DQe[34]	1018. DQe[35]	1019. DQe[36]	1020. DQe[37]
	1021. DQe[38]	1022. DQe[39]	1023. DQe[40]	1024. DQe[41]
	1025. DQe[42]	1026. DQe[43]	1027. DQe[44]	1028. DQe[45]
	1029. DQe[46]	1030. DQe[47]	1031. DQe[48]	1032. DQe[49]
	1033. DQe[50]	1034. DQe[51]	1035. DQe[52]	1036. DQe[53]
	1037. DQe[54]	1038. DQe[55]	1039. DQe[56]	1040. DQe[57]
	1041. DQe[58]	1042. DQe[59]	1043. DQe[60]	1044. DQe[61]
	1045. DQe[62]	1046. DQe[63]	1047. DQe[64]	1048. DQe[65]
	1049. DQe[66]	1050. DQe[67]	1051. DQe[68]	1052. DQe[69]
	1053. DQe[70]	1054. DQe[71]	1055. DQe[72]	1056. DQe[73]
	1057. DQe[74]	1058. DQe[75]	1059. DQe[76]	1060. DQe[77]
	1061. DQe[78]	1062. DQe[79]	1063. DQe[80]	1064. DQe[81]
	1065. DQe[82]	1066. DQe[83]	1067. DQe[84]	1068. DQe[85]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	1069. DQe[86]	1070. DQe[87]	1071. DQe[88]	1072. DQe[89]
	1073. DQe[90]	1074. DQe[91]	1075. DQe[92]	1076. DQe[93]
	1077. DQe[94]	1078. DQe[95]	1079. DQe[96]	1080. DQe[97]
	1081. DQe[98]	1082. DQe[99]	1083. DQe[100]	1084. DQe[101]
	1085. DQe[102]	1086. DQe[103]	1087. DQe[104]	1088. DQe[105]
	1089. DQe[106]	1090. DQe[107]	1091. DQe[108]	1092. DQe[109]
	1093. DQe[110]	1094. DQe[111]	1095. DQe[112]	1096. DQe[113]
	1097. DQe[114]	1098. DQe[115]	1099. DQe[116]	1100. DQe[117]
	1101. DQe[118]	1102. DQe[119]	1103. DQe[120]	1104. DQe[121]
	1105. DQe[122]	1106. DQe[123]	1107. DQe[124]	1108. DQe[125]
	1109. DQe[126]	1110. DQe[127]	1111. DQf[0]	1112. DQf[1]
	1113. DQf[2]	1114. DQf[3]	1115. DQf[4]	1116. DQf[5]
	1117. DQf[6]	1118. DQf[7]	1119. DQf[8]	1120. DQf[9]
	1121. DQf[10]	1122. DQf[11]	1123. DQf[12]	1124. DQf[13]
	1125. DQf[14]	1126. DQf[15]	1127. DQf[16]	1128. DQf[17]
	1129. DQf[18]	1130. DQf[19]	1131. DQf[20]	1132. DQf[21]
	1133. DQf[22]	1134. DQf[23]	1135. DQf[24]	1136. DQf[25]
	1137. DQf[26]	1138. DQf[27]	1139. DQf[28]	1140. DQf[29]
	1141. DQf[30]	1142. DQf[31]	1143. DQf[32]	1144. DQf[33]
	1145. DQf[34]	1146. DQf[35]	1147. DQf[36]	1148. DQf[37]
	1149. DQf[38]	1150. DQf[39]	1151. DQf[40]	1152. DQf[41]
	1153. DQf[42]	1154. DQf[43]	1155. DQf[44]	1156. DQf[45]
	1157. DQf[46]	1158. DQf[47]	1159. DQf[48]	1160. DQf[49]
	1161. DQf[50]	1162. DQf[51]	1163. DQf[52]	1164. DQf[53]
	1165. DQf[54]	1166. DQf[55]	1167. DQf[56]	1168. DQf[57]
	1169. DQf[58]	1170. DQf[59]	1171. DQf[60]	1172. DQf[61]
	1173. DQf[62]	1174. DQf[63]	1175. DQf[64]	1176. DQf[65]
	1177. DQf[66]	1178. DQf[67]	1179. DQf[68]	1180. DQf[69]
	1181. DQf[70]	1182. DQf[71]	1183. DQf[72]	1184. DQf[73]
	1185. DQf[74]	1186. DQf[75]	1187. DQf[76]	1188. DQf[77]
	1189. DQf[78]	1190. DQf[79]	1191. DQf[80]	1192. DQf[81]
	1193. DQf[82]	1194. DQf[83]	1195. DQf[84]	1196. DQf[85]
	1197. DQf[86]	1198. DQf[87]	1199. DQf[88]	1200. DQf[89]
	1201. DQf[90]	1202. DQf[91]	1203. DQf[92]	1204. DQf[93]
	1205. DQf[94]	1206. DQf[95]	1207. DQf[96]	1208. DQf[97]
	1209. DQf[98]	1210. DQf[99]	1211. DQf[100]	1212. DQf[101]
	1213. DQf[102]	1214. DQf[103]	1215. DQf[104]	1216. DQf[105]
	1217. DQf[106]	1218. DQf[107]	1219. DQf[108]	1220. DQf[109]
	1221. DQf[110]	1222. DQf[111]	1223. DQf[112]	1224. DQf[113]
	1225. DQf[114]	1226. DQf[115]	1227. DQf[116]	1228. DQf[117]
	1229. DQf[118]	1230. DQf[119]	1231. DQf[120]	1232. DQf[121]
	1233. DQf[122]	1234. DQf[123]	1235. DQf[124]	1236. DQf[125]
	1237. DQf[126]	1238. DQf[127]	1239. DQg[0]	1240. DQg[1]
	1241. DQg[2]	1242. DQg[3]	1243. DQg[4]	1244. DQg[5]
	1245. DQg[6]	1246. DQg[7]	1247. DQg[8]	1248. DQg[9]
	1249. DQg[10]	1250. DQg[11]	1251. DQg[12]	1252. DQg[13]
	1253. DQg[14]	1254. DQg[15]	1255. DQg[16]	1256. DQg[17]
	1257. DQg[18]	1258. DQg[19]	1259. DQg[20]	1260. DQg[21]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	1261. DQg[22]	1262. DQg[23]	1263. DQg[24]	1264. DQg[25]
	1265. DQg[26]	1266. DQg[27]	1267. DQg[28]	1268. DQg[29]
	1269. DQg[30]	1270. DQg[31]	1271. DQg[32]	1272. DQg[33]
	1273. DQg[34]	1274. DQg[35]	1275. DQg[36]	1276. DQg[37]
	1277. DQg[38]	1278. DQg[39]	1279. DQg[40]	1280. DQg[41]
	1281. DQg[42]	1282. DQg[43]	1283. DQg[44]	1284. DQg[45]
	1285. DQg[46]	1286. DQg[47]	1287. DQg[48]	1288. DQg[49]
	1289. DQg[50]	1290. DQg[51]	1291. DQg[52]	1292. DQg[53]
	1293. DQg[54]	1294. DQg[55]	1295. DQg[56]	1296. DQg[57]
	1297. DQg[58]	1298. DQg[59]	1299. DQg[60]	1300. DQg[61]
	1301. DQg[62]	1302. DQg[63]	1303. DQg[64]	1304. DQg[65]
	1305. DQg[66]	1306. DQg[67]	1307. DQg[68]	1308. DQg[69]
	1309. DQg[70]	1310. DQg[71]	1311. DQg[72]	1312. DQg[73]
	1313. DQg[74]	1314. DQg[75]	1315. DQg[76]	1316. DQg[77]
	1317. DQg[78]	1318. DQg[79]	1319. DQg[80]	1320. DQg[81]
	1321. DQg[82]	1322. DQg[83]	1323. DQg[84]	1324. DQg[85]
	1325. DQg[86]	1326. DQg[87]	1327. DQg[88]	1328. DQg[89]
	1329. DQg[90]	1330. DQg[91]	1331. DQg[92]	1332. DQg[93]
	1333. DQg[94]	1334. DQg[95]	1335. DQg[96]	1336. DQg[97]
	1337. DQg[98]	1338. DQg[99]	1339. DQg[100]	1340. DQg[101]
	1341. DQg[102]	1342. DQg[103]	1343. DQg[104]	1344. DQg[105]
	1345. DQg[106]	1346. DQg[107]	1347. DQg[108]	1348. DQg[109]
	1349. DQg[110]	1350. DQg[111]	1351. DQg[112]	1352. DQg[113]
	1353. DQg[114]	1354. DQg[115]	1355. DQg[116]	1356. DQg[117]
	1357. DQg[118]	1358. DQg[119]	1359. DQg[120]	1360. DQg[121]
	1361. DQg[122]	1362. DQg[123]	1363. DQg[124]	1364. DQg[125]
	1365. DQg[126]	1366. DQg[127]	1367. DQh[0]	1368. DQh[1]
	1369. DQh[2]	1370. DQh[3]	1371. DQh[4]	1372. DQh[5]
	1373. DQh[6]	1374. DQh[7]	1375. DQh[8]	1376. DQh[9]
	1377. DQh[10]	1378. DQh[11]	1379. DQh[12]	1380. DQh[13]
	1381. DQh[14]	1382. DQh[15]	1383. DQh[16]	1384. DQh[17]
	1385. DQh[18]	1386. DQh[19]	1387. DQh[20]	1388. DQh[21]
	1389. DQh[22]	1390. DQh[23]	1391. DQh[24]	1392. DQh[25]
	1393. DQh[26]	1394. DQh[27]	1395. DQh[28]	1396. DQh[29]
	1397. DQh[30]	1398. DQh[31]	1399. DQh[32]	1400. DQh[33]
	1401. DQh[34]	1402. DQh[35]	1403. DQh[36]	1404. DQh[37]
	1405. DQh[38]	1406. DQh[39]	1407. DQh[40]	1408. DQh[41]
	1409. DQh[42]	1410. DQh[43]	1411. DQh[44]	1412. DQh[45]
	1413. DQh[46]	1414. DQh[47]	1415. DQh[48]	1416. DQh[49]
	1417. DQh[50]	1418. DQh[51]	1419. DQh[52]	1420. DQh[53]
	1421. DQh[54]	1422. DQh[55]	1423. DQh[56]	1424. DQh[57]
	1425. DQh[58]	1426. DQh[59]	1427. DQh[60]	1428. DQh[61]
	1429. DQh[62]	1430. DQh[63]	1431. DQh[64]	1432. DQh[65]
	1433. DQh[66]	1434. DQh[67]	1435. DQh[68]	1436. DQh[69]
	1437. DQh[70]	1438. DQh[71]	1439. DQh[72]	1440. DQh[73]
	1441. DQh[74]	1442. DQh[75]	1443. DQh[76]	1444. DQh[77]
	1445. DQh[78]	1446. DQh[79]	1447. DQh[80]	1448. DQh[81]
	1449. DQh[82]	1450. DQh[83]	1451. DQh[84]	1452. DQh[85]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	1453. DQh[86]	1454. DQh[87]	1455. DQh[88]	1456. DQh[89]
	1457. DQh[90]	1458. DQh[91]	1459. DQh[92]	1460. DQh[93]
	1461. DQh[94]	1462. DQh[95]	1463. DQh[96]	1464. DQh[97]
	1465. DQh[98]	1466. DQh[99]	1467. DQh[100]	1468. DQh[101]
	1469. DQh[102]	1470. DQh[103]	1471. DQh[104]	1472. DQh[105]
	1473. DQh[106]	1474. DQh[107]	1475. DQh[108]	1476. DQh[109]
	1477. DQh[110]	1478. DQh[111]	1479. DQh[112]	1480. DQh[113]
	1481. DQh[114]	1482. DQh[115]	1483. DQh[116]	1484. DQh[117]
	1485. DQh[118]	1486. DQh[119]	1487. DQh[120]	1488. DQh[121]
	1489. DQh[122]	1490. DQh[123]	1491. DQh[124]	1492. DQh[125]
	1493. DQh[126]	1494. DQh[127]	1495. MRFU[0]	1496. MRFU[1]
	1497. MRFU[2]	1498. MRFU[3]	1499. MRFU[4]	1500. MRFU[5]
	1501. MRFU[6]	1502. MRFU[7]	1503. MRFU[8]	1504. MRFU[9]
	1505. MRFU[10]	1506. MRFU[11]	1507. MRFU[12]	1508. MRFU[13]
	1509. MRFU[14]	1510. MRFU[15]	1511. MRFU[16]	1512. MRFU[17]
	1513. MRFU[18]	1514. MRFU[19]	1515. MRFU[20]	1516. MRFU[21]
	1517. MRFU[22]	1518. MRFU[23]	1519. PARa[0]	1520. PARa[1]
	1521. PARa[2]	1522. PARa[3]	1523. PARb[0]	1524. PARb[1]
	1525. PARb[2]	1526. PARb[3]	1527. PARc[0]	1528. PARc[1]
	1529. PARc[2]	1530. PARc[3]	1531. PARd[0]	1532. PARd[1]
	1533. PARd[2]	1534. PARd[3]	1535. PARe[0]	1536. PARe[1]
	1537. PARe[2]	1538. PARe[3]	1539. PARf[0]	1540. PARf[1]
	1541. PARf[2]	1542. PARf[3]	1543. PARg[0]	1544. PARg[1]
	1545. PARg[2]	1546. PARg[3]	1547. PARh[0]	1548. PARh[1]
	1549. PARh[2]	1550. PARh[3]	1551. Ra[0]	1552. Ra[1]
	1553. Ra[2]	1554. Ra[3]	1555. Ra[4]	1556. Ra[5]
	1557. Ra[6]	1558. Rb[0]	1559. Rb[1]	1560. Rb[2]
	1561. Rb[3]	1562. Rb[4]	1563. Rb[5]	1564. Rb[6]
	1565. Rc[0]	1566. Rc[1]	1567. Rc[2]	1568. Rc[3]
	1569. Rc[4]	1570. Rc[5]	1571. Rc[6]	1572. RC[a]
	1573. RC[b]	1574. RC[c]	1575. RC[d]	1576. RC[e]
	1577. RC[f]	1578. RC[g]	1579. RC[h]	1580. Rd[0]
	1581. Rd[1]	1582. Rd[2]	1583. Rd[3]	1584. Rd[4]
	1585. Rd[5]	1586. Rd[6]	1587. RDa[0]	1588. RDa[1]
	1589. RDa[2]	1590. RDa[3]	1591. RDa[4]	1592. RDa[5]
	1593. RDa[6]	1594. RDa[7]	1595. RDb[0]	1596. RDb[1]
	1597. RDb[2]	1598. RDb[3]	1599. RDb[4]	1600. RDb[5]
	1601. RDb[6]	1602. RDb[7]	1603. RDc[0]	1604. RDc[1]
	1605. RDc[2]	1606. RDc[3]	1607. RDc[4]	1608. RDc[5]
	1609. RDc[6]	1610. RDc[7]	1611. RDd[0]	1612. RDd[1]
	1613. RDd[2]	1614. RDd[3]	1615. RDd[4]	1616. RDd[5]
	1617. RDd[6]	1618. RDd[7]	1619. RDe[0]	1620. RDe[1]
	1621. RDe[2]	1622. RDe[3]	1623. RDe[4]	1624. RDe[5]
	1625. RDe[6]	1626. RDe[7]	1627. RDf[0]	1628. RDf[1]
	1629. RDf[2]	1630. RDf[3]	1631. RDf[4]	1632. RDf[5]
	1633. RDf[6]	1634. RDf[7]	1635. RDg[0]	1636. RDg[1]
	1637. RDg[2]	1638. RDg[3]	1639. RDg[4]	1640. RDg[5]
	1641. RDg[6]	1642. RDg[7]	1643. RDh[0]	1644. RDh[1]

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	1645. RDH[2]	1646. RDH[3]	1647. RDH[4]	1648. RDH[5]
	1649. RDH[6]	1650. RDH[7]	1651. RDQSa0_c	1652. RDQSa0_t
	1653. RDQSa1_c	1654. RDQSa1_t	1655. RDQSa2_c	1656. RDQSa2_t
	1657. RDQSa3_c	1658. RDQSa3_t	1659. RDQSB0_c	1660. RDQSB0_t
	1661. RDQSB1_c	1662. RDQSB1_t	1663. RDQSB2_c	1664. RDQSB2_t
	1665. RDQSB3_c	1666. RDQSB3_t	1667. RDQSc0_c	1668. RDQSc0_t
	1669. RDQSc1_c	1670. RDQSc1_t	1671. RDQSc2_c	1672. RDQSc2_t
	1673. RDQSc3_c	1674. RDQSc3_t	1675. RDQSd0_c	1676. RDQSd0_t
	1677. RDQSd1_c	1678. RDQSd1_t	1679. RDQSd2_c	1680. RDQSd2_t
	1681. RDQSd3_c	1682. RDQSd3_t	1683. RDQSe0_c	1684. RDQSe0_t
	1685. RDQSe1_c	1686. RDQSe1_t	1687. RDQSe2_c	1688. RDQSe2_t
	1689. RDQSe3_c	1690. RDQSe3_t	1691. RDQSF0_c	1692. RDQSF0_t
	1693. RDQSF1_c	1694. RDQSF1_t	1695. RDQSF2_c	1696. RDQSF2_t
	1697. RDQSF3_c	1698. RDQSF3_t	1699. RDQSG0_c	1700. RDQSG0_t
	1701. RDQSG1_c	1702. RDQSG1_t	1703. RDQSG2_c	1704. RDQSG2_t
	1705. RDQSG3_c	1706. RDQSG3_t	1707. RDQSh0_c	1708. RDQSh0_t
	1709. RDQSh1_c	1710. RDQSh1_t	1711. RDQSh2_c	1712. RDQSh2_t
	1713. RDQSh3_c	1714. RDQSh3_t	1715. Re[0]	1716. Re[1]
	1717. Re[2]	1718. Re[3]	1719. Re[4]	1720. Re[5]
	1721. Re[6]	1722. RESET_n	1723. Rf[0]	1724. Rf[1]
	1725. Rf[2]	1726. Rf[3]	1727. Rf[4]	1728. Rf[5]
	1729. Rf[6]	1730. Rg[0]	1731. Rg[1]	1732. Rg[2]
	1733. Rg[3]	1734. Rg[4]	1735. Rg[5]	1736. Rg[6]
	1737. Rh[0]	1738. Rh[1]	1739. Rh[2]	1740. Rh[3]
	1741. Rh[4]	1742. Rh[5]	1743. Rh[6]	1744. RRa
	1745. RRb	1746. RRc	1747. RRD	1748. RRe
	1749. RRf	1750. RRg	1751. RRh	1752. RSVD[0]
	1753. RSVD[1]	1754. RSVD[2]	1755. RSVD[3]	1756. RSVD[4]
	1757. RSVD[5]	1758. SELECTWIR	1759. SHIFTWR	1760. TEMP[0]
	1761. TEMP[1]	1762. TEMP[2]	1763. UPDATEWR	1764. VDDC
	1765. VDDQ	1766. VPP	1767. VSS	1768. WDQSA0_c
	1769. WDQSA0_t	1770. WDQSA1_c	1771. WDQSA1_t	1772. WDQSA2_c
	1773. WDQSA2_t	1774. WDQSA3_c	1775. WDQSA3_t	1776. WDQSB0_c
	1777. WDQSB0_t	1778. WDQSB1_c	1779. WDQSB1_t	1780. WDQSB2_c
	1781. WDQSB2_t	1782. WDQSB3_c	1783. WDQSB3_t	1784. WDQSC0_c
	1785. WDQSC0_t	1786. WDQSC1_c	1787. WDQSC1_t	1788. WDQSC2_c
	1789. WDQSC2_t	1790. WDQSC3_c	1791. WDQSC3_t	1792. WDQSd0_c
	1793. WDQSd0_t	1794. WDQSd1_c	1795. WDQSd1_t	1796. WDQSd2_c
	1797. WDQSd2_t	1798. WDQSd3_c	1799. WDQSd3_t	1800. WDQSe0_c
	1801. WDQSe0_t	1802. WDQSe1_c	1803. WDQSe1_t	1804. WDQSe2_c
	1805. WDQSe2_t	1806. WDQSe3_c	1807. WDQSe3_t	1808. WDQSF0_c
	1809. WDQSF0_t	1810. WDQSF1_c	1811. WDQSF1_t	1812. WDQSF2_c
	1813. WDQSF2_t	1814. WDQSF3_c	1815. WDQSF3_t	1816. WDQSG0_c
	1817. WDQSG0_t	1818. WDQSG1_c	1819. WDQSG1_t	1820. WDQSG2_c
	1821. WDQSG2_t	1822. WDQSG3_c	1823. WDQSG3_t	1824. WDQSH0_c
	1825. WDQSH0_t	1826. WDQSH1_c	1827. WDQSH1_t	1828. WDQSH2_c
	1829. WDQSH2_t	1830. WDQSH3_c	1831. WDQSH3_t	1832. WRCK
	1833. WRST_n	1834. WSI	1835. WSOa	1836. WSOb

4.7.3.10.17.2 HBM1, HBM2 and HBM2E Footprint B Interface Function (cont'd)

list of enumerate values (cont.)	1837. WSOc	1838. WSOd	1839. WSOe	1840. WSOf
	1841. WSOg	1842. WSOh		

4.7.3.10.17.3. HBM3 Interface Functions

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM/HBM3																																																																																																																
diagram	<pre> classDiagram class HBM3 { <<HBM3-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HBM3-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<HBM3-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<HBM3-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<HBM3-MandatoryStandardTerminalNameType>> or <<HBM3-OptionalStandardTerminalNameType>> } HBM3 "1..∞" --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> MandatoryMapping StandardTerminalNameAssignment --> OptionalMapping MandatoryMapping --> StandardTerminalName OptionalMapping --> StandardTerminalName StandardTerminalName "1..∞" --> constraints </pre>																																																																																																																
type	HBM3-InterfaceFunctionType , HBM3-StandardTerminalNameAssignmentType , HBM3-MandatoryStandardTerminalMappingType , HBM3-MandatoryStandardTerminalNameType , HBM3-OptionalStandardTerminalMappingType , HBM3-OptionalStandardTerminalNameType .																																																																																																																
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. CKa_t</td><td>2. CKb_t</td><td>3. CKc_t</td><td>4. CKd_t</td></tr> <tr><td>5. CKe_t</td><td>6. CKf_t</td><td>7. CKg_t</td><td>8. CKh_t</td></tr> <tr><td>9. CKi_t</td><td>10. CKj_t</td><td>11. CKk_t</td><td>12. CKl_t</td></tr> <tr><td>13. CKm_t</td><td>14. CKn_t</td><td>15. CKo_t</td><td>16. CKp_t</td></tr> <tr><td>17. CKa_c</td><td>18. CKb_c</td><td>19. CKc_c</td><td>20. CKd_c</td></tr> <tr><td>21. CKe_c</td><td>22. CKf_c</td><td>23. CKg_c</td><td>24. CKh_c</td></tr> <tr><td>25. CKi_c</td><td>26. CKj_c</td><td>27. CKk_c</td><td>28. CKl_c</td></tr> <tr><td>29. CKm_c</td><td>30. CKn_c</td><td>31. CKo_c</td><td>32. CKp_c</td></tr> <tr><td>33. Ca0</td><td>34. Ca1</td><td>35. Ca2</td><td>36. Ca3</td></tr> <tr><td>37. Ca4</td><td>38. Ca5</td><td>39. Ca6</td><td>40. Ca7</td></tr> <tr><td>41. Cb0</td><td>42. Cb1</td><td>43. Cb2</td><td>44. Cb3</td></tr> <tr><td>45. Cb4</td><td>46. Cb5</td><td>47. Cb6</td><td>48. Cb7</td></tr> <tr><td>49. Cc0</td><td>50. Cc1</td><td>51. Cc2</td><td>52. Cc3</td></tr> <tr><td>53. Cc4</td><td>54. Cc5</td><td>55. Cc6</td><td>56. Cc7</td></tr> <tr><td>57. Cd0</td><td>58. Cd1</td><td>59. Cd2</td><td>60. Cd3</td></tr> <tr><td>61. Cd4</td><td>62. Cd5</td><td>63. Cd6</td><td>64. Cd7</td></tr> <tr><td>65. Ce0</td><td>66. Ce1</td><td>67. Ce2</td><td>68. Ce3</td></tr> <tr><td>69. Ce4</td><td>70. Ce5</td><td>71. Ce6</td><td>72. Ce7</td></tr> <tr><td>73. Cf0</td><td>74. Cf1</td><td>75. Cf2</td><td>76. Cf3</td></tr> <tr><td>77. Cf4</td><td>78. Cf5</td><td>79. Cf6</td><td>80. Cf7</td></tr> <tr><td>81. Cg0</td><td>82. Cg1</td><td>83. Cg2</td><td>84. Cg3</td></tr> <tr><td>85. Cg4</td><td>86. Cg5</td><td>87. Cg6</td><td>88. Cg7</td></tr> <tr><td>89. Ch0</td><td>90. Ch1</td><td>91. Ch2</td><td>92. Ch3</td></tr> <tr><td>93. Ch4</td><td>94. Ch5</td><td>95. Ch6</td><td>96. Ch7</td></tr> <tr><td>97. Ci0</td><td>98. Ci1</td><td>99. Ci2</td><td>100. Ci3</td></tr> <tr><td>101. Ci4</td><td>102. Ci5</td><td>103. Ci6</td><td>104. Ci7</td></tr> <tr><td>105. Cj0</td><td>106. Cj1</td><td>107. Cj2</td><td>108. Cj3</td></tr> <tr><td>109. Cj4</td><td>110. Cj5</td><td>111. Cj6</td><td>112. Cj7</td></tr> </tbody> </table>	1. CKa_t	2. CKb_t	3. CKc_t	4. CKd_t	5. CKe_t	6. CKf_t	7. CKg_t	8. CKh_t	9. CKi_t	10. CKj_t	11. CKk_t	12. CKl_t	13. CKm_t	14. CKn_t	15. CKo_t	16. CKp_t	17. CKa_c	18. CKb_c	19. CKc_c	20. CKd_c	21. CKe_c	22. CKf_c	23. CKg_c	24. CKh_c	25. CKi_c	26. CKj_c	27. CKk_c	28. CKl_c	29. CKm_c	30. CKn_c	31. CKo_c	32. CKp_c	33. Ca0	34. Ca1	35. Ca2	36. Ca3	37. Ca4	38. Ca5	39. Ca6	40. Ca7	41. Cb0	42. Cb1	43. Cb2	44. Cb3	45. Cb4	46. Cb5	47. Cb6	48. Cb7	49. Cc0	50. Cc1	51. Cc2	52. Cc3	53. Cc4	54. Cc5	55. Cc6	56. Cc7	57. Cd0	58. Cd1	59. Cd2	60. Cd3	61. Cd4	62. Cd5	63. Cd6	64. Cd7	65. Ce0	66. Ce1	67. Ce2	68. Ce3	69. Ce4	70. Ce5	71. Ce6	72. Ce7	73. Cf0	74. Cf1	75. Cf2	76. Cf3	77. Cf4	78. Cf5	79. Cf6	80. Cf7	81. Cg0	82. Cg1	83. Cg2	84. Cg3	85. Cg4	86. Cg5	87. Cg6	88. Cg7	89. Ch0	90. Ch1	91. Ch2	92. Ch3	93. Ch4	94. Ch5	95. Ch6	96. Ch7	97. Ci0	98. Ci1	99. Ci2	100. Ci3	101. Ci4	102. Ci5	103. Ci6	104. Ci7	105. Cj0	106. Cj1	107. Cj2	108. Cj3	109. Cj4	110. Cj5	111. Cj6	112. Cj7
1. CKa_t	2. CKb_t	3. CKc_t	4. CKd_t																																																																																																														
5. CKe_t	6. CKf_t	7. CKg_t	8. CKh_t																																																																																																														
9. CKi_t	10. CKj_t	11. CKk_t	12. CKl_t																																																																																																														
13. CKm_t	14. CKn_t	15. CKo_t	16. CKp_t																																																																																																														
17. CKa_c	18. CKb_c	19. CKc_c	20. CKd_c																																																																																																														
21. CKe_c	22. CKf_c	23. CKg_c	24. CKh_c																																																																																																														
25. CKi_c	26. CKj_c	27. CKk_c	28. CKl_c																																																																																																														
29. CKm_c	30. CKn_c	31. CKo_c	32. CKp_c																																																																																																														
33. Ca0	34. Ca1	35. Ca2	36. Ca3																																																																																																														
37. Ca4	38. Ca5	39. Ca6	40. Ca7																																																																																																														
41. Cb0	42. Cb1	43. Cb2	44. Cb3																																																																																																														
45. Cb4	46. Cb5	47. Cb6	48. Cb7																																																																																																														
49. Cc0	50. Cc1	51. Cc2	52. Cc3																																																																																																														
53. Cc4	54. Cc5	55. Cc6	56. Cc7																																																																																																														
57. Cd0	58. Cd1	59. Cd2	60. Cd3																																																																																																														
61. Cd4	62. Cd5	63. Cd6	64. Cd7																																																																																																														
65. Ce0	66. Ce1	67. Ce2	68. Ce3																																																																																																														
69. Ce4	70. Ce5	71. Ce6	72. Ce7																																																																																																														
73. Cf0	74. Cf1	75. Cf2	76. Cf3																																																																																																														
77. Cf4	78. Cf5	79. Cf6	80. Cf7																																																																																																														
81. Cg0	82. Cg1	83. Cg2	84. Cg3																																																																																																														
85. Cg4	86. Cg5	87. Cg6	88. Cg7																																																																																																														
89. Ch0	90. Ch1	91. Ch2	92. Ch3																																																																																																														
93. Ch4	94. Ch5	95. Ch6	96. Ch7																																																																																																														
97. Ci0	98. Ci1	99. Ci2	100. Ci3																																																																																																														
101. Ci4	102. Ci5	103. Ci6	104. Ci7																																																																																																														
105. Cj0	106. Cj1	107. Cj2	108. Cj3																																																																																																														
109. Cj4	110. Cj5	111. Cj6	112. Cj7																																																																																																														

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	113. Ck0	114. Ck1	115. Ck2	116. Ck3
	117. Ck4	118. Ck5	119. Ck6	120. Ck7
	121. Cl0	122. Cl1	123. Cl2	124. Cl3
	125. Cl4	126. Cl5	127. Cl6	128. Cl7
	129. Cm0	130. Cm1	131. Cm2	132. Cm3
	133. Cm4	134. Cm5	135. Cm6	136. Cm7
	137. Cn0	138. Cn1	139. Cn2	140. Cn3
	141. Cn4	142. Cn5	143. Cn6	144. Cn7
	145. Co0	146. Co1	147. Co2	148. Co3
	149. Co4	150. Co5	151. Co6	152. Co7
	153. Cp0	154. Cp1	155. Cp2	156. Cp3
	157. Cp4	158. Cp5	159. Cp6	160. Cp7
	161. Ra0	162. Ra1	163. Ra2	164. Ra3
	165. Ra4	166. Ra5	167. Ra6	168. Ra7
	169. Ra8	170. Ra9	171. Rb0	172. Rb1
	173. Rb2	174. Rb3	175. Rb4	176. Rb5
	177. Rb6	178. Rb7	179. Rb8	180. Rb9
	181. Rc0	182. Rc1	183. Rc2	184. Rc3
	185. Rc4	186. Rc5	187. Rc6	188. Rc7
	189. Rc8	190. Rc9	191. Rd0	192. Rd1
	193. Rd2	194. Rd3	195. Rd4	196. Rd5
	197. Rd6	198. Rd7	199. Rd8	200. Rd9
	201. Re0	202. Re1	203. Re2	204. Re3
	205. Re4	206. Re5	207. Re6	208. Re7
	209. Re8	210. Re9	211. Rf0	212. Rf1
	213. Rf2	214. Rf3	215. Rf4	216. Rf5
	217. Rf6	218. Rf7	219. Rf8	220. Rf9
	221. Rg0	222. Rg1	223. Rg2	224. Rg3
	225. Rg4	226. Rg5	227. Rg6	228. Rg7
	229. Rg8	230. Rg9	231. Rh0	232. Rh1
	233. Rh2	234. Rh3	235. Rh4	236. Rh5
	237. Rh6	238. Rh7	239. Rh8	240. Rh9
	241. Ri0	242. Ri1	243. Ri2	244. Ri3
	245. Ri4	246. Ri5	247. Ri6	248. Ri7
	249. Ri8	250. Ri9	251. Rj0	252. Rj1
	253. Rj2	254. Rj3	255. Rj4	256. Rj5
	257. Rj6	258. Rj7	259. Rj8	260. Rj9
	261. Rk0	262. Rk1	263. Rk2	264. Rk3
	265. Rk4	266. Rk5	267. Rk6	268. Rk7
	269. Rk8	270. Rk9	271. RI0	272. RI1
	273. RI2	274. RI3	275. RI4	276. RI5
	277. RI6	278. RI7	279. RI8	280. RI9
	281. Rm0	282. Rm1	283. Rm2	284. Rm3
	285. Rm4	286. Rm5	287. Rm6	288. Rm7
	289. Rm8	290. Rm9	291. Rn0	292. Rn1
	293. Rn2	294. Rn3	295. Rn4	296. Rn5
	297. Rn6	298. Rn7	299. Rn8	300. Rn9

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	301. Ro0	302. Ro1	303. Ro2	304. Ro3
	305. Ro4	306. Ro5	307. Ro6	308. Ro7
	309. Ro8	310. Ro9	311. Rp0	312. Rp1
	313. Rp2	314. Rp3	315. Rp4	316. Rp5
	317. Rp6	318. Rp7	319. Rp8	320. Rp9
	321. ARFUa	322. ARFUb	323. ARFUc	324. ARFUD
	325. ARFUe	326. ARFUf	327. ARFUG	328. ARFUH
	329. ARFUj	330. ARFUj	331. ARFUk	332. ARFUI
	333. ARFUm	334. ARFUn	335. ARFUo	336. ARFUp
	337. APARa	338. APARb	339. APARc	340. APARd
	341. APARe	342. APARf	343. APARg	344. APARh
	345. APARI	346. APARj	347. APARK	348. APARI
	349. APARM	350. APARN	351. APARo	352. APARP
	353. DQa[0]	354. DQa[1]	355. DQa[2]	356. DQa[3]
	357. DQa[4]	358. DQa[5]	359. DQa[6]	360. DQa[7]
	361. DQa[8]	362. DQa[9]	363. DQa[10]	364. DQa[11]
	365. DQa[12]	366. DQa[13]	367. DQa[14]	368. DQa[15]
	369. DQa[16]	370. DQa[17]	371. DQa[18]	372. DQa[19]
	373. DQa[20]	374. DQa[21]	375. DQa[22]	376. DQa[23]
	377. DQa[24]	378. DQa[25]	379. DQa[26]	380. DQa[27]
	381. DQa[28]	382. DQa[29]	383. DQa[30]	384. DQa[31]
	385. DQa[32]	386. DQa[33]	387. DQa[34]	388. DQa[35]
	389. DQa[36]	390. DQa[37]	391. DQa[38]	392. DQa[39]
	393. DQa[40]	394. DQa[41]	395. DQa[42]	396. DQa[43]
	397. DQa[44]	398. DQa[45]	399. DQa[46]	400. DQa[47]
	401. DQa[48]	402. DQa[49]	403. DQa[50]	404. DQa[51]
	405. DQa[52]	406. DQa[53]	407. DQa[54]	408. DQa[55]
	409. DQa[56]	410. DQa[57]	411. DQa[58]	412. DQa[59]
	413. DQa[60]	414. DQa[61]	415. DQa[62]	416. DQa[63]
	417. DQb[0]	418. DQb[1]	419. DQb[2]	420. DQb[3]
	421. DQb[4]	422. DQb[5]	423. DQb[6]	424. DQb[7]
	425. DQb[8]	426. DQb[9]	427. DQb[10]	428. DQb[11]
	429. DQb[12]	430. DQb[13]	431. DQb[14]	432. DQb[15]
	433. DQb[16]	434. DQb[17]	435. DQb[18]	436. DQb[19]
	437. DQb[20]	438. DQb[21]	439. DQb[22]	440. DQb[23]
	441. DQb[24]	442. DQb[25]	443. DQb[26]	444. DQb[27]
	445. DQb[28]	446. DQb[29]	447. DQb[30]	448. DQb[31]
	449. DQb[32]	450. DQb[33]	451. DQb[34]	452. DQb[35]
	453. DQb[36]	454. DQb[37]	455. DQb[38]	456. DQb[39]
	457. DQb[40]	458. DQb[41]	459. DQb[42]	460. DQb[43]
	461. DQb[44]	462. DQb[45]	463. DQb[46]	464. DQb[47]
	465. DQb[48]	466. DQb[49]	467. DQb[50]	468. DQb[51]
	469. DQb[52]	470. DQb[53]	471. DQb[54]	472. DQb[55]
	473. DQb[56]	474. DQb[57]	475. DQb[58]	476. DQb[59]
	477. DQb[60]	478. DQb[61]	479. DQb[62]	480. DQb[63]
	481. DQc[0]	482. DQc[1]	483. DQc[2]	484. DQc[3]
	485. DQc[4]	486. DQc[5]	487. DQc[6]	488. DQc[7]

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	489. DQc[8]	490. DQc[9]	491. DQc[10]	492. DQc[11]
	493. DQc[12]	494. DQc[13]	495. DQc[14]	496. DQc[15]
	497. DQc[16]	498. DQc[17]	499. DQc[18]	500. DQc[19]
	501. DQc[20]	502. DQc[21]	503. DQc[22]	504. DQc[23]
	505. DQc[24]	506. DQc[25]	507. DQc[26]	508. DQc[27]
	509. DQc[28]	510. DQc[29]	511. DQc[30]	512. DQc[31]
	513. DQc[32]	514. DQc[33]	515. DQc[34]	516. DQc[35]
	517. DQc[36]	518. DQc[37]	519. DQc[38]	520. DQc[39]
	521. DQc[40]	522. DQc[41]	523. DQc[42]	524. DQc[43]
	525. DQc[44]	526. DQc[45]	527. DQc[46]	528. DQc[47]
	529. DQc[48]	530. DQc[49]	531. DQc[50]	532. DQc[51]
	533. DQc[52]	534. DQc[53]	535. DQc[54]	536. DQc[55]
	537. DQc[56]	538. DQc[57]	539. DQc[58]	540. DQc[59]
	541. DQc[60]	542. DQc[61]	543. DQc[62]	544. DQc[63]
	545. DQd[0]	546. DQd[1]	547. DQd[2]	548. DQd[3]
	549. DQd[4]	550. DQd[5]	551. DQd[6]	552. DQd[7]
	553. DQd[8]	554. DQd[9]	555. DQd[10]	556. DQd[11]
	557. DQd[12]	558. DQd[13]	559. DQd[14]	560. DQd[15]
	561. DQd[16]	562. DQd[17]	563. DQd[18]	564. DQd[19]
	565. DQd[20]	566. DQd[21]	567. DQd[22]	568. DQd[23]
	569. DQd[24]	570. DQd[25]	571. DQd[26]	572. DQd[27]
	573. DQd[28]	574. DQd[29]	575. DQd[30]	576. DQd[31]
	577. DQd[32]	578. DQd[33]	579. DQd[34]	580. DQd[35]
	581. DQd[36]	582. DQd[37]	583. DQd[38]	584. DQd[39]
	585. DQd[40]	586. DQd[41]	587. DQd[42]	588. DQd[43]
	589. DQd[44]	590. DQd[45]	591. DQd[46]	592. DQd[47]
	593. DQd[48]	594. DQd[49]	595. DQd[50]	596. DQd[51]
	597. DQd[52]	598. DQd[53]	599. DQd[54]	600. DQd[55]
	601. DQd[56]	602. DQd[57]	603. DQd[58]	604. DQd[59]
	605. DQd[60]	606. DQd[61]	607. DQd[62]	608. DQd[63]
	609. DQe[0]	610. DQe[1]	611. DQe[2]	612. DQe[3]

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	613. DQe[4]	614. DQe[5]	615. DQe[6]	616. DQe[7]
	617. DQe[8]	618. DQe[9]	619. DQe[10]	620. DQe[11]
	621. DQe[12]	622. DQe[13]	623. DQe[14]	624. DQe[15]
	625. DQe[16]	626. DQe[17]	627. DQe[18]	628. DQe[19]
	629. DQe[20]	630. DQe[21]	631. DQe[22]	632. DQe[23]
	633. DQe[24]	634. DQe[25]	635. DQe[26]	636. DQe[27]
	637. DQe[28]	638. DQe[29]	639. DQe[30]	640. DQe[31]
	641. DQe[32]	642. DQe[33]	643. DQe[34]	644. DQe[35]
	645. DQe[36]	646. DQe[37]	647. DQe[38]	648. DQe[39]
	649. DQe[40]	650. DQe[41]	651. DQe[42]	652. DQe[43]
	653. DQe[44]	654. DQe[45]	655. DQe[46]	656. DQe[47]
	657. DQe[48]	658. DQe[49]	659. DQe[50]	660. DQe[51]
	661. DQe[52]	662. DQe[53]	663. DQe[54]	664. DQe[55]
	665. DQe[56]	666. DQe[57]	667. DQe[58]	668. DQe[59]
	669. DQe[60]	670. DQe[61]	671. DQe[62]	672. DQe[63]
	673. DQf[0]	674. DQf[1]	675. DQf[2]	676. DQf[3]
	677. DQf[4]	678. DQf[5]	679. DQf[6]	680. DQf[7]
	681. DQf[8]	682. DQf[9]	683. DQf[10]	684. DQf[11]
	685. DQf[12]	686. DQf[13]	687. DQf[14]	688. DQf[15]
	689. DQf[16]	690. DQf[17]	691. DQf[18]	692. DQf[19]
	693. DQf[20]	694. DQf[21]	695. DQf[22]	696. DQf[23]
	697. DQf[24]	698. DQf[25]	699. DQf[26]	700. DQf[27]
	701. DQf[28]	702. DQf[29]	703. DQf[30]	704. DQf[31]
	705. DQf[32]	706. DQf[33]	707. DQf[34]	708. DQf[35]
	709. DQf[36]	710. DQf[37]	711. DQf[38]	712. DQf[39]
	713. DQf[40]	714. DQf[41]	715. DQf[42]	716. DQf[43]
	717. DQf[44]	718. DQf[45]	719. DQf[46]	720. DQf[47]
	721. DQf[48]	722. DQf[49]	723. DQf[50]	724. DQf[51]
	725. DQf[52]	726. DQf[53]	727. DQf[54]	728. DQf[55]
	729. DQf[56]	730. DQf[57]	731. DQf[58]	732. DQf[59]
	733. DQf[60]	734. DQf[61]	735. DQf[62]	736. DQf[63]
	737. DQg[0]	738. DQg[1]	739. DQg[2]	740. DQg[3]
	741. DQg[4]	742. DQg[5]	743. DQg[6]	744. DQg[7]
	745. DQg[8]	746. DQg[9]	747. DQg[10]	748. DQg[11]
	749. DQg[12]	750. DQg[13]	751. DQg[14]	752. DQg[15]
	753. DQg[16]	754. DQg[17]	755. DQg[18]	756. DQg[19]
	757. DQg[20]	758. DQg[21]	759. DQg[22]	760. DQg[23]
	761. DQg[24]	762. DQg[25]	763. DQg[26]	764. DQg[27]
	765. DQg[28]	766. DQg[29]	767. DQg[30]	768. DQg[31]
	769. DQg[32]	770. DQg[33]	771. DQg[34]	772. DQg[35]
	773. DQg[36]	774. DQg[37]	775. DQg[38]	776. DQg[39]
	777. DQg[40]	778. DQg[41]	779. DQg[42]	780. DQg[43]
	781. DQg[44]	782. DQg[45]	783. DQg[46]	784. DQg[47]

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	785. DQg[48]	786. DQg[49]	787. DQg[50]	788. DQg[51]
	789. DQg[52]	790. DQg[53]	791. DQg[54]	792. DQg[55]
	793. DQg[56]	794. DQg[57]	795. DQg[58]	796. DQg[59]
	797. DQg[60]	798. DQg[61]	799. DQg[62]	800. DQg[63]
	801. DQh[0]	802. DQh[1]	803. DQh[2]	804. DQh[3]
	805. DQh[4]	806. DQh[5]	807. DQh[6]	808. DQh[7]
	809. DQh[8]	810. DQh[9]	811. DQh[10]	812. DQh[11]
	813. DQh[12]	814. DQh[13]	815. DQh[14]	816. DQh[15]
	817. DQh[16]	818. DQh[17]	819. DQh[18]	820. DQh[19]
	821. DQh[20]	822. DQh[21]	823. DQh[22]	824. DQh[23]
	825. DQh[24]	826. DQh[25]	827. DQh[26]	828. DQh[27]
	829. DQh[28]	830. DQh[29]	831. DQh[30]	832. DQh[31]
	833. DQh[32]	834. DQh[33]	835. DQh[34]	836. DQh[35]
	837. DQh[36]	838. DQh[37]	839. DQh[38]	840. DQh[39]
	841. DQh[40]	842. DQh[41]	843. DQh[42]	844. DQh[43]
	845. DQh[44]	846. DQh[45]	847. DQh[46]	848. DQh[47]
	849. DQh[48]	850. DQh[49]	851. DQh[50]	852. DQh[51]
	853. DQh[52]	854. DQh[53]	855. DQh[54]	856. DQh[55]
	857. DQh[56]	858. DQh[57]	859. DQh[58]	860. DQh[59]
	861. DQh[60]	862. DQh[61]	863. DQh[62]	864. DQh[63]
	865. DQi[0]	866. DQi[1]	867. DQi[2]	868. DQi[3]
	869. DQi[4]	870. DQi[5]	871. DQi[6]	872. DQi[7]
	873. DQi[8]	874. DQi[9]	875. DQi[10]	876. DQi[11]
	877. DQi[12]	878. DQi[13]	879. DQi[14]	880. DQi[15]
	881. DQi[16]	882. DQi[17]	883. DQi[18]	884. DQi[19]
	885. DQi[20]	886. DQi[21]	887. DQi[22]	888. DQi[23]
	889. DQi[24]	890. DQi[25]	891. DQi[26]	892. DQi[27]
	893. DQi[28]	894. DQi[29]	895. DQi[30]	896. DQi[31]
	897. DQi[32]	898. DQi[33]	899. DQi[34]	900. DQi[35]
	901. DQi[36]	902. DQi[37]	903. DQi[38]	904. DQi[39]
	905. DQi[40]	906. DQi[41]	907. DQi[42]	908. DQi[43]
	909. DQi[44]	910. DQi[45]	911. DQi[46]	912. DQi[47]
	913. DQi[48]	914. DQi[49]	915. DQi[50]	916. DQi[51]
	917. DQi[52]	918. DQi[53]	919. DQi[54]	920. DQi[55]
	921. DQi[56]	922. DQi[57]	923. DQi[58]	924. DQi[59]
	925. DQi[60]	926. DQi[61]	927. DQi[62]	928. DQi[63]
	929. DQj[0]	930. DQj[1]	931. DQj[2]	932. DQj[3]
	933. DQj[4]	934. DQj[5]	935. DQj[6]	936. DQj[7]
	937. DQj[8]	938. DQj[9]	939. DQj[10]	940. DQj[11]
	941. DQj[12]	942. DQj[13]	943. DQj[14]	944. DQj[15]
	945. DQj[16]	946. DQj[17]	947. DQj[18]	948. DQj[19]
	949. DQj[20]	950. DQj[21]	951. DQj[22]	952. DQj[23]
	953. DQj[24]	954. DQj[25]	955. DQj[26]	956. DQj[27]

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	957. DQj[28]	958. DQj[29]	959. DQj[30]	960. DQj[31]
	961. DQj[32]	962. DQj[33]	963. DQj[34]	964. DQj[35]
	965. DQj[36]	966. DQj[37]	967. DQj[38]	968. DQj[39]
	969. DQj[40]	970. DQj[41]	971. DQj[42]	972. DQj[43]
	973. DQj[44]	974. DQj[45]	975. DQj[46]	976. DQj[47]
	977. DQj[48]	978. DQj[49]	979. DQj[50]	980. DQj[51]
	981. DQj[52]	982. DQj[53]	983. DQj[54]	984. DQj[55]
	985. DQj[56]	986. DQj[57]	987. DQj[58]	988. DQj[59]
	989. DQj[60]	990. DQj[61]	991. DQj[62]	992. DQj[63]
	993. DQk[0]	994. DQk[1]	995. DQk[2]	996. DQk[3]
	997. DQk[4]	998. DQk[5]	999. DQk[6]	1000. DQk[7]
	1001. DQk[8]	1002. DQk[9]	1003. DQk[10]	1004. DQk[11]
	1005. DQk[12]	1006. DQk[13]	1007. DQk[14]	1008. DQk[15]
	1009. DQk[16]	1010. DQk[17]	1011. DQk[18]	1012. DQk[19]
	1013. DQk[20]	1014. DQk[21]	1015. DQk[22]	1016. DQk[23]
	1017. DQk[24]	1018. DQk[25]	1019. DQk[26]	1020. DQk[27]
	1021. DQk[28]	1022. DQk[29]	1023. DQk[30]	1024. DQk[31]
	1025. DQk[32]	1026. DQk[33]	1027. DQk[34]	1028. DQk[35]
	1029. DQk[36]	1030. DQk[37]	1031. DQk[38]	1032. DQk[39]
	1033. DQk[40]	1034. DQk[41]	1035. DQk[42]	1036. DQk[43]
	1037. DQk[44]	1038. DQk[45]	1039. DQk[46]	1040. DQk[47]
	1041. DQk[48]	1042. DQk[49]	1043. DQk[50]	1044. DQk[51]
	1045. DQk[52]	1046. DQk[53]	1047. DQk[54]	1048. DQk[55]
	1049. DQk[56]	1050. DQk[57]	1051. DQk[58]	1052. DQk[59]
	1053. DQk[60]	1054. DQk[61]	1055. DQk[62]	1056. DQk[63]
	1057. DQI[0]	1058. DQI[1]	1059. DQI[2]	1060. DQI[3]
	1061. DQI[4]	1062. DQI[5]	1063. DQI[6]	1064. DQI[7]
	1065. DQI[8]	1066. DQI[9]	1067. DQI[10]	1068. DQI[11]
	1069. DQI[12]	1070. DQI[13]	1071. DQI[14]	1072. DQI[15]
	1073. DQI[16]	1074. DQI[17]	1075. DQI[18]	1076. DQI[19]
	1077. DQI[20]	1078. DQI[21]	1079. DQI[22]	1080. DQI[23]
	1081. DQI[24]	1082. DQI[25]	1083. DQI[26]	1084. DQI[27]
	1085. DQI[28]	1086. DQI[29]	1087. DQI[30]	1088. DQI[31]
	1089. DQI[32]	1090. DQI[33]	1091. DQI[34]	1092. DQI[35]
	1093. DQI[36]	1094. DQI[37]	1095. DQI[38]	1096. DQI[39]
	1097. DQI[40]	1098. DQI[41]	1099. DQI[42]	1100. DQI[43]
	1101. DQI[44]	1102. DQI[45]	1103. DQI[46]	1104. DQI[47]
	1105. DQI[48]	1106. DQI[49]	1107. DQI[50]	1108. DQI[51]
	1109. DQI[52]	1110. DQI[53]	1111. DQI[54]	1112. DQI[55]
	1113. DQI[56]	1114. DQI[57]	1115. DQI[58]	1116. DQI[59]
	1117. DQI[60]	1118. DQI[61]	1119. DQI[62]	1120. DQI[63]
	1121. DQm[0]	1122. DQm[1]	1123. DQm[2]	1124. DQm[3]
	1125. DQm[4]	1126. DQm[5]	1127. DQm[6]	1128. DQm[7]

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	1129. DQm[8]	1130. DQm[9]	1131. DQm[10]	1132. DQm[11]
	1133. DQm[12]	1134. DQm[13]	1135. DQm[14]	1136. DQm[15]
	1137. DQm[16]	1138. DQm[17]	1139. DQm[18]	1140. DQm[19]
	1141. DQm[20]	1142. DQm[21]	1143. DQm[22]	1144. DQm[23]
	1145. DQm[24]	1146. DQm[25]	1147. DQm[26]	1148. DQm[27]
	1149. DQm[28]	1150. DQm[29]	1151. DQm[30]	1152. DQm[31]
	1153. DQm[32]	1154. DQm[33]	1155. DQm[34]	1156. DQm[35]
	1157. DQm[36]	1158. DQm[37]	1159. DQm[38]	1160. DQm[39]
	1161. DQm[40]	1162. DQm[41]	1163. DQm[42]	1164. DQm[43]
	1165. DQm[44]	1166. DQm[45]	1167. DQm[46]	1168. DQm[47]
	1169. DQm[48]	1170. DQm[49]	1171. DQm[50]	1172. DQm[51]
	1173. DQm[52]	1174. DQm[53]	1175. DQm[54]	1176. DQm[55]
	1177. DQm[56]	1178. DQm[57]	1179. DQm[58]	1180. DQm[59]
	1181. DQm[60]	1182. DQm[61]	1183. DQm[62]	1184. DQm[63]
	1185. DQn[0]	1186. DQn[1]	1187. DQn[2]	1188. DQn[3]
	1189. DQn[4]	1190. DQn[5]	1191. DQn[6]	1192. DQn[7]
	1193. DQn[8]	1194. DQn[9]	1195. DQn[10]	1196. DQn[11]
	1197. DQn[12]	1198. DQn[13]	1199. DQn[14]	1200. DQn[15]
	1201. DQn[16]	1202. DQn[17]	1203. DQn[18]	1204. DQn[19]
	1205. DQn[20]	1206. DQn[21]	1207. DQn[22]	1208. DQn[23]
	1209. DQn[24]	1210. DQn[25]	1211. DQn[26]	1212. DQn[27]
	1213. DQn[28]	1214. DQn[29]	1215. DQn[30]	1216. DQn[31]
	1217. DQn[32]	1218. DQn[33]	1219. DQn[34]	1220. DQn[35]
	1221. DQn[36]	1222. DQn[37]	1223. DQn[38]	1224. DQn[39]
	1225. DQn[40]	1226. DQn[41]	1227. DQn[42]	1228. DQn[43]
	1229. DQn[44]	1230. DQn[45]	1231. DQn[46]	1232. DQn[47]
	1233. DQn[48]	1234. DQn[49]	1235. DQn[50]	1236. DQn[51]
	1237. DQn[52]	1238. DQn[53]	1239. DQn[54]	1240. DQn[55]
	1241. DQn[56]	1242. DQn[57]	1243. DQn[58]	1244. DQn[59]
	1245. DQn[60]	1246. DQn[61]	1247. DQn[62]	1248. DQn[63]
	1249. DQo[0]	1250. DQo[1]	1251. DQo[2]	1252. DQo[3]
	1253. DQo[4]	1254. DQo[5]	1255. DQo[6]	1256. DQo[7]
	1257. DQo[8]	1258. DQo[9]	1259. DQo[10]	1260. DQo[11]
	1261. DQo[12]	1262. DQo[13]	1263. DQo[14]	1264. DQo[15]
	1265. DQo[16]	1266. DQo[17]	1267. DQo[18]	1268. DQo[19]
	1269. DQo[20]	1270. DQo[21]	1271. DQo[22]	1272. DQo[23]
	1273. DQo[24]	1274. DQo[25]	1275. DQo[26]	1276. DQo[27]
	1277. DQo[28]	1278. DQo[29]	1279. DQo[30]	1280. DQo[31]
	1281. DQo[32]	1282. DQo[33]	1283. DQo[34]	1284. DQo[35]
	1285. DQo[36]	1286. DQo[37]	1287. DQo[38]	1288. DQo[39]
	1289. DQo[40]	1290. DQo[41]	1291. DQo[42]	1292. DQo[43]
	1293. DQo[44]	1294. DQo[45]	1295. DQo[46]	1296. DQo[47]
	1297. DQo[48]	1298. DQo[49]	1299. DQo[50]	1300. DQo[51]

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	1301. DQo[52]	1302. DQo[53]	1303. DQo[54]	1304. DQo[55]
	1305. DQo[56]	1306. DQo[57]	1307. DQo[58]	1308. DQo[59]
	1309. DQo[60]	1310. DQo[61]	1311. DQo[62]	1312. DQo[63]
	1313. DQp[0]	1314. DQp[1]	1315. DQp[2]	1316. DQp[3]
	1317. DQp[4]	1318. DQp[5]	1319. DQp[6]	1320. DQp[7]
	1321. DQp[8]	1322. DQp[9]	1323. DQp[10]	1324. DQp[11]
	1325. DQp[12]	1326. DQp[13]	1327. DQp[14]	1328. DQp[15]
	1329. DQp[16]	1330. DQp[17]	1331. DQp[18]	1332. DQp[19]
	1333. DQp[20]	1334. DQp[21]	1335. DQp[22]	1336. DQp[23]
	1337. DQp[24]	1338. DQp[25]	1339. DQp[26]	1340. DQp[27]
	1341. DQp[28]	1342. DQp[29]	1343. DQp[30]	1344. DQp[31]
	1345. DQp[32]	1346. DQp[33]	1347. DQp[34]	1348. DQp[35]
	1349. DQp[36]	1350. DQp[37]	1351. DQp[38]	1352. DQp[39]
	1353. DQp[40]	1354. DQp[41]	1355. DQp[42]	1356. DQp[43]
	1357. DQp[44]	1358. DQp[45]	1359. DQp[46]	1360. DQp[47]
	1361. DQp[48]	1362. DQp[49]	1363. DQp[50]	1364. DQp[51]
	1365. DQp[52]	1366. DQp[53]	1367. DQp[54]	1368. DQp[55]
	1369. DQp[56]	1370. DQp[57]	1371. DQp[58]	1372. DQp[59]
	1373. DQp[60]	1374. DQp[61]	1375. DQp[62]	1376. DQp[63]
	1377. DBla0	1378. DBla1	1379. DBla2	1380. DBla3
	1381. DBla4	1382. DBla5	1383. DBla6	1384. DBla7
	1385. DBlb0	1386. DBlb1	1387. DBlb2	1388. DBlb3
	1389. DBlb4	1390. DBlb5	1391. DBlb6	1392. DBlb7
	1393. DBlc0	1394. DBlc1	1395. DBlc2	1396. DBlc3
	1397. DBlc4	1398. DBlc5	1399. DBlc6	1400. DBlc7
	1401. DBld0	1402. DBld1	1403. DBld2	1404. DBld3
	1405. DBld4	1406. DBld5	1407. DBld6	1408. DBld7
	1409. DBle0	1410. DBle1	1411. DBle2	1412. DBle3
	1413. DBle4	1414. DBle5	1415. DBle6	1416. DBle7
	1417. DBlf0	1418. DBlf1	1419. DBlf2	1420. DBlf3
	1421. DBlf4	1422. DBlf5	1423. DBlf6	1424. DBlf7
	1425. DBlg0	1426. DBlg1	1427. DBlg2	1428. DBlg3
	1429. DBlg4	1430. DBlg5	1431. DBlg6	1432. DBlg7
	1433. DBlh0	1434. DBlh1	1435. DBlh2	1436. DBlh3
	1437. DBlh4	1438. DBlh5	1439. DBlh6	1440. DBlh7
	1441. DBli0	1442. DBli1	1443. DBli2	1444. DBli3
	1445. DBli4	1446. DBli5	1447. DBli6	1448. DBli7
	1449. DBlj0	1450. DBlj1	1451. DBlj2	1452. DBlj3
	1453. DBlj4	1454. DBlj5	1455. DBlj6	1456. DBlj7
	1457. DBlk0	1458. DBlk1	1459. DBlk2	1460. DBlk3
	1461. DBlk4	1462. DBlk5	1463. DBlk6	1464. DBlk7
	1465. DBli0	1466. DBli1	1467. DBli2	1468. DBli3
	1469. DBli4	1470. DBli5	1471. DBli6	1472. DBli7

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	1473. DBIm0	1474. DBIm1	1475. DBIm2	1476. DBIm3
	1477. DBIm4	1478. DBIm5	1479. DBIm6	1480. DBIm7
	1481. DBIn0	1482. DBIn1	1483. DBIn2	1484. DBIn3
	1485. DBIn4	1486. DBIn5	1487. DBIn6	1488. DBIn7
	1489. DBIo0	1490. DBIo1	1491. DBIo2	1492. DBIo3
	1493. DBIo4	1494. DBIo5	1495. DBIo6	1496. DBIo7
	1497. DBIp0	1498. DBIp1	1499. DBIp2	1500. DBIp3
	1501. DBIp4	1502. DBIp5	1503. DBIp6	1504. DBIp7
	1505. ECCa0	1506. ECCa1	1507. ECCa2	1508. ECCa3
	1509. ECCb0	1510. ECCb1	1511. ECCb2	1512. ECCb3
	1513. ECCc0	1514. ECCc1	1515. ECCc2	1516. ECCc3
	1517. ECCd0	1518. ECCd1	1519. ECCd2	1520. ECCd3
	1521. ECCe0	1522. ECCe1	1523. ECCe2	1524. ECCe3
	1525. ECCf0	1526. ECCf1	1527. ECCf2	1528. ECCf3
	1529. ECCg0	1530. ECCg1	1531. ECCg2	1532. ECCg3
	1533. ECCh0	1534. ECCh1	1535. ECCh2	1536. ECCh3
	1537. ECCi0	1538. ECCi1	1539. ECCi2	1540. ECCi3
	1541. ECCj0	1542. ECCj1	1543. ECCj2	1544. ECCj3
	1545. ECCk0	1546. ECCk1	1547. ECCk2	1548. ECCk3
	1549. ECCl0	1550. ECCl1	1551. ECCl2	1552. ECCl3
	1553. ECCm0	1554. ECCm1	1555. ECCm2	1556. ECCm3
	1557. ECCn0	1558. ECCn1	1559. ECCn2	1560. ECCn3
	1561. ECCo0	1562. ECCo1	1563. ECCo2	1564. ECCo3
	1565. ECCp0	1566. ECCp1	1567. ECCp2	1568. ECCp3
	1569. SEVa0	1570. SEVa1	1571. SEVa2	1572. SEVa3
	1573. SEVb0	1574. SEVb1	1575. SEVb2	1576. SEVb3
	1577. SEVc0	1578. SEVc1	1579. SEVc2	1580. SEVc3
	1581. SEVd0	1582. SEVd1	1583. SEVd2	1584. SEVd3
	1585. SEVe0	1586. SEVe1	1587. SEVe2	1588. SEVe3
	1589. SEVf0	1590. SEVf1	1591. SEVf2	1592. SEVf3
	1593. SEVg0	1594. SEVg1	1595. SEVg2	1596. SEVg3
	1597. SEVh0	1598. SEVh1	1599. SEVh2	1600. SEVh3
	1601. SEVi0	1602. SEVi1	1603. SEVi2	1604. SEVi3
	1605. SEVj0	1606. SEVj1	1607. SEVj2	1608. SEVj3
	1609. SEVk0	1610. SEVk1	1611. SEVk2	1612. SEVk3
	1613. SEVI0	1614. SEVI1	1615. SEVI2	1616. SEVI3
	1617. SEVm0	1618. SEVm1	1619. SEVm2	1620. SEVm3
	1621. SEVn0	1622. SEVn1	1623. SEVn2	1624. SEVn3
	1625. SEVo0	1626. SEVo1	1627. SEVo2	1628. SEVo3
	1629. SEVp0	1630. SEVp1	1631. SEVp2	1632. SEVp3
	1633. DPARa0	1634. DPARb0	1635. DPARc0	1636. DPARd0
	1637. DPARe0	1638. DPARf0	1639. DPARg0	1640. DPARh0
	1641. DPARi0	1642. DPARj0	1643. DPARk0	1644. DPARl0

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	1645. DPARm0	1646. DPARn0	1647. DPARo0	1648. DPARp0
	1649. DPARa1	1650. DPARb1	1651. DPARc1	1652. DPARd1
	1653. DPARe1	1654. DPARf1	1655. DPARg1	1656. DPARh1
	1657. DPARi1	1658. DPARj1	1659. DPARk1	1660. DPARl1
	1661. DPARm1	1662. DPARn1	1663. DPARo1	1664. DPARp1
	1665. DERRa0	1666. DERRb0	1667. DERRc0	1668. DERRd0
	1669. DERRe0	1670. DERRf0	1671. DERRg0	1672. DERRh0
	1673. DERRi0	1674. DERRj0	1675. DERRk0	1676. DERRl0
	1677. DERRm0	1678. DERRn0	1679. DERRo0	1680. DERRp0
	1681. DERRa1	1682. DERRb1	1683. DERRc1	1684. DERRd1
	1685. DERRe1	1686. DERRf1	1687. DERRg1	1688. DERRh1
	1689. DERRi1	1690. DERRj1	1691. DERRk1	1692. DERRl1
	1693. DERRm1	1694. DERRn1	1695. DERRo1	1696. DERRp1
	1697. AERRa	1698. AERRb	1699. AERRc	1700. AERRd
	1701. AERRe	1702. AERRf	1703. AERRg	1704. AERRh
	1705. AERRi	1706. AERRj	1707. AERRk	1708. AERRl
	1709. AERRm	1710. AERRn	1711. AERRo	1712. AERRp
	1713. WDQSa0_t	1714. WDQSb0_t	1715. WDQSc0_t	1716. WDQSD0_t
	1717. WDQSe0_t	1718. WDQSf0_t	1719. WDQSG0_t	1720. WDQSh0_t
	1721. WDQSi0_t	1722. WDQSp0_t	1723. WDQSk0_t	1724. WDQSI0_t
	1725. WDQSm0_t	1726. WDQSn0_t	1727. WDQSo0_t	1728. WDQSp0_t
	1729. WDQSa1_t	1730. WDQSb1_t	1731. WDQSc1_t	1732. WDQSD1_t
	1733. WDQSe1_t	1734. WDQSp1_t	1735. WDQSG1_t	1736. WDQSh1_t
	1737. WDQSi1_t	1738. WDQSp1_t	1739. WDQSk1_t	1740. WDQSI1_t
	1741. WDQSm1_t	1742. WDQSn1_t	1743. WDQSo1_t	1744. WDQSp1_t
	1745. WDQSa0_c	1746. WDQSb0_c	1747. WDQSc0_c	1748. WDQSD0_c
	1749. WDQSe0_c	1750. WDQSf0_c	1751. WDQSG0_c	1752. WDQSh0_c
	1753. WDQSi0_c	1754. WDQSp0_c	1755. WDQSk0_c	1756. WDQSI0_c
	1757. WDQSm0_c	1758. WDQSn0_c	1759. WDQSo0_c	1760. WDQSp0_c
	1761. WDQSa1_c	1762. WDQSb1_c	1763. WDQSc1_c	1764. WDQSD1_c
	1765. WDQSe1_c	1766. WDQSp1_c	1767. WDQSG1_c	1768. WDQSh1_c
	1769. WDQSi1_c	1770. WDQSp1_c	1771. WDQSk1_c	1772. WDQSI1_c
	1773. WDQSm1_c	1774. WDQSn1_c	1775. WDQSo1_c	1776. WDQSp1_c
	1777. RDQSa0_t	1778. RDQSb0_t	1779. RDQSc0_t	1780. RDQSD0_t
	1781. RDQSe0_t	1782. RDQSf0_t	1783. RDQSG0_t	1784. RDQSh0_t
	1785. RDQSi0_t	1786. RDQSp0_t	1787. RDQSk0_t	1788. RDQSI0_t
	1789. RDQSm0_t	1790. RDQSn0_t	1791. RDQSo0_t	1792. RDQSp0_t
	1793. RDQSa1_t	1794. RDQSb1_t	1795. RDQSc1_t	1796. RDQSD1_t
	1797. RDQSe1_t	1798. RDQSp1_t	1799. RDQSG1_t	1800. RDQSh1_t
	1801. RDQSi1_t	1802. RDQSp1_t	1803. RDQSk1_t	1804. RDQSI1_t
	1805. RDQSm1_t	1806. RDQSn1_t	1807. RDQSo1_t	1808. RDQSp1_t
	1809. RDQSa0_c	1810. RDQSb0_c	1811. RDQSc0_c	1812. RDQSD0_c
	1813. RDQSe0_c	1814. RDQSp0_c	1815. RDQSG0_c	1816. RDQSh0_c

4.7.3.10.17.3 HBM3 Interface Function (cont'd)

list of enumerate values (cont.)	1817. RDQSi0_c	1818. RDQSj0_c	1819. RDQSk0_c	1820. RDQSI0_c
	1821. RDQSm0_c	1822. RDQSn0_c	1823. RDQSs0_c	1824. RDQSp0_c
	1825. RDQSa1_c	1826. RDQSb1_c	1827. RDQSc1_c	1828. RDQSD1_c
	1829. RDQSe1_c	1830. RDQSf1_c	1831. RDQSg1_c	1832. RDQSh1_c
	1833. RDQSi1_c	1834. RDQSj1_c	1835. RDQSk1_c	1836. RDQSI1_c
	1837. RDQSm1_c	1838. RDQSn1_c	1839. RDQSs1_c	1840. RDQSp1_c
	1841. DA0	1842. DA1	1843. DA2	1844. DA3
	1845. DA4	1846. DA5	1847. DA6	1848. DA7
	1849. DA8	1850. DA9	1851. DA10	1852. DA11
	1853. DA12	1854. DA13	1855. DA14	1856. DA15
	1857. DA16	1858. DA17	1859. DA18	1860. DA19
	1861. DA20	1862. DA21	1863. DA22	1864. DA23
	1865. DA24	1866. DA25	1867. DA26	1868. DA27
	1869. DA28	1870. DA29	1871. DA30	1872. DA31
	1873. DA32	1874. DA33	1875. DA34	1876. DA35
	1877. DA36	1878. DA37	1879. DA38	1880. DA39
	1881. RESET_n	1882. WRCK	1883. WRST_n	1884. SelectWIR
	1885. ShiftWR	1886. CaptureWR	1887. UpdateWR	1888. WSI
	1889. WSOa	1890. WSOb	1891. WSOc	1892. WSOd
	1893. WSOe	1894. WSOf	1895. WSOg	1896. WSOh
	1897. WSOi	1898. WSOj	1899. WSOk	1900. WSOl
	1901. WSOm	1902. WSOn	1903. WSOo	1904. WSOp
	1905. TEMP0	1906. TEMP1	1907. CATTRIP	1908. VSS
	1909. VDDC	1910. VDDQ	1911. VPP	1912. VDDQL
OptionalMapping/StandardTerminalName				
1. RDa0 2. RDb0 3. RDc0 4. RDd0				
5. RDe0 6. RDf0 7. RDg0 8. RDh0				
9. RDi0 10. RDj0 11. RDk0 12. RDl0				
13. RDm0 14. RDn0 15. RDo0 16. RDp0				
17. RDa1 18. RDb1 19. RDc1 20. RDd1				
21. RDe1 22. RDf1 23. RDg1 24. RDh1				
25. RDi1 26. RDj1 27. RDk1 28. RDl1				
29. RDm1 30. RDn1 31. RDo1 32. RDp1				
33. RDa2 34. RDb2 35. RDc2 36. RDd2				
37. RDe2 38. RDf2 39. RDg2 40. RDh2				
41. RDi2 42. RDj2 43. RDk2 44. RDl2				
45. RDm2 46. RDn2 47. RDo2 48. RDp2				
49. RDa3 50. RDb3 51. RDc3 52. RDd3				
53. RDe3 54. RDf3 55. RDg3 56. RDh3				
57. RDi3 58. RDj3 59. RDk3 60. RDl3				
61. RDm3 62. RDn3 63. RDo3 64. RDp3				
65. RAa 66. RAb 67. RAc 68. Rad				
69. RAe 70. RAf 71. RAg 72. RAh				
73. RAI 74. RAj 75. RAk 76. RAI				
77. RAM 78. RAn 79. RAo 80. RAp				

For more information about the HBM3 Interface, refer to the JEDEC standard JESD238.

4.7.3.10.17.4. HBM4 Interface Functions

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM/HBM4																																												
diagram	<pre> classDiagram class HBM4 { <<HBM4>> <<HBM4-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HBM4-StandardTerminalNameAssignmentType>> } class Mapping { <<HBM4-StandardTerminalMappingType>> } class StandardTerminalName { <<HBM4-StandardTerminalNameType>> } class StructuredMapping_Array { <<HBM4-StandardStructuredTerminalMappingType>> } HBM4 "1..∞" -- "1..∞" StandardTerminalNameAssignment : StandardTerminalNameAssignment "1..∞" -- "1..∞" Mapping : Mapping "1..∞" -- "1..∞" StandardTerminalName : StandardTerminalNameAssignment --> StructuredMapping_Array : Mapping --> assertions : Mapping --> constraints : </pre>																																												
type	HBM4-InterfaceFunctionType , HBM4-StandardTerminalNameAssignmentType , HBM4-StandardTerminalMappingType , HBM4-StandardTerminalNameType , HBM4-StandardStructuredTerminalMappingArrayType																																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CAPTUREWR</td><td>2. CATTRIP</td><td>3. NC</td><td>4. RESET_n</td></tr> <tr> <td>5. SELECTWIR</td><td>6. SHIFTWR</td><td>7. UPDATEWR</td><td>8. VDDC</td></tr> <tr> <td>9. VDDQ</td><td>10. VDDQL</td><td>11. VPP</td><td>12. VSS</td></tr> <tr> <td>13. WRCK</td><td>14. WRST_n</td><td>15. WSI</td><td>16. RM[0..1]</td></tr> <tr> <td>17. MRFU[0..3]</td><td>18. ARFU[0..31]</td><td>19. APAR[0..31]</td><td>20. AERR[0..31]</td></tr> <tr> <td>21. WSO[0..31]</td><td>22. RA[0..31]</td><td>23. DA[0..39]</td><td>24. DPAR[0..31][0..1]</td></tr> <tr> <td>25. DERR[0..31][0..1]</td><td>26. CK[0..31]_t</td><td>27. CK[0..31]_c</td><td>28. EEC[0..31][0..4]</td></tr> <tr> <td>29. SEV[0..31][0..4]</td><td>30. RD[0..31][0..4]</td><td>31. C[0..31][0..8]</td><td>32. DBI[0..31][0..8]</td></tr> <tr> <td>33. R[0..31][0..9]</td><td>34. DQ[0..31][0..64]</td><td>35. WDQS[0..31][0..1]_t</td><td>36. WDQS[0..31][0..1]_c</td></tr> <tr> <td>37. RDQS[0..31][0..1]_t</td><td>38. RDQS[0..31][0..1]_c</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. CAPTUREWR	2. CATTRIP	3. NC	4. RESET_n	5. SELECTWIR	6. SHIFTWR	7. UPDATEWR	8. VDDC	9. VDDQ	10. VDDQL	11. VPP	12. VSS	13. WRCK	14. WRST_n	15. WSI	16. RM[0..1]	17. MRFU[0..3]	18. ARFU[0..31]	19. APAR[0..31]	20. AERR[0..31]	21. WSO[0..31]	22. RA[0..31]	23. DA[0..39]	24. DPAR[0..31][0..1]	25. DERR[0..31][0..1]	26. CK[0..31]_t	27. CK[0..31]_c	28. EEC[0..31][0..4]	29. SEV[0..31][0..4]	30. RD[0..31][0..4]	31. C[0..31][0..8]	32. DBI[0..31][0..8]	33. R[0..31][0..9]	34. DQ[0..31][0..64]	35. WDQS[0..31][0..1]_t	36. WDQS[0..31][0..1]_c	37. RDQS[0..31][0..1]_t	38. RDQS[0..31][0..1]_c		
Mapping/StandardTerminalName																																													
1. CAPTUREWR	2. CATTRIP	3. NC	4. RESET_n																																										
5. SELECTWIR	6. SHIFTWR	7. UPDATEWR	8. VDDC																																										
9. VDDQ	10. VDDQL	11. VPP	12. VSS																																										
13. WRCK	14. WRST_n	15. WSI	16. RM[0..1]																																										
17. MRFU[0..3]	18. ARFU[0..31]	19. APAR[0..31]	20. AERR[0..31]																																										
21. WSO[0..31]	22. RA[0..31]	23. DA[0..39]	24. DPAR[0..31][0..1]																																										
25. DERR[0..31][0..1]	26. CK[0..31]_t	27. CK[0..31]_c	28. EEC[0..31][0..4]																																										
29. SEV[0..31][0..4]	30. RD[0..31][0..4]	31. C[0..31][0..8]	32. DBI[0..31][0..8]																																										
33. R[0..31][0..9]	34. DQ[0..31][0..64]	35. WDQS[0..31][0..1]_t	36. WDQS[0..31][0..1]_c																																										
37. RDQS[0..31][0..1]_t	38. RDQS[0..31][0..1]_c																																												

For more information about the HBM4 Interface, refer to the JEDEC standard JESD270-4. For the expanded list of Standard Terminal Names, refer to the JEDEC JESD271-4 Bump Matrix Spreadsheet.

4.7.3.10.17.4.1. HBM4 - Structured Mapping - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM/HBM4/StandardTerminalNameAssignment/StructuredMapping-Array
diagram 1 of 13	<p>The diagram illustrates the structure of the <code>HBM4-StandardStructuredTerminalMapping-ArrayType</code>. It features a main container class <code>HBM4-StandardStructuredTerminalMapping-ArrayType</code> which contains several fields, each represented by a box with a plus sign (+) icon in the top right corner. The fields are:</p> <ul style="list-style-type: none"> <code>Field1</code> (<code>HBM4-StandardStructuredTerminalMapping-Field1Type</code>) <code>Field2x2i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field2x2iType</code>) <code>Field2x4i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field2x4iType</code>) <code>Field2x32i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field2x32iType</code>) <code>Field2x40i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field2x40iType</code>) <code>Field4x32i.2i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field4x32i.2iType</code>) <code>Field4x32i.2s</code> (<code>HBM4-StandardStructuredTerminalMapping-Field4x32i.2sType</code>) <code>Field4x32i.4i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field4x32i.4iType</code>) <code>Field4x32i.8i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field4x32i.8iType</code>) <code>Field4x32i.10i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field4x32i.10iType</code>) <code>Field4x32i.64i</code> (<code>HBM4-StandardStructuredTerminalMapping-Field4x32i.64iType</code>) <code>Field6x32i.2i.2s</code> (<code>HBM4-StandardStructuredTerminalMapping-Field6x32i.2i.2sType</code>) <p>A connector line links the <code>StructuredMapping-Array</code> class to the <code>HBM4-StandardStructuredTerminalMapping-ArrayType</code> container.</p>
type	<p><code>HBM4-StandardStructuredTerminalMapping-ArrayType</code>, <code>HBM4-Field1-StandardTerminalBaseNameType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field1Type</code>, <code>HBM4-StandardStructuredTerminalMapping-Field2x2iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field2x4iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field2x32iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field2x40iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field4x32i.2iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field4x32i.2sType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field4x32i.4iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field4x32i.8iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field4x32i.10iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field4x32i.64iType</code>, <code>HBM4-StandardStructuredTerminalMapping-Field6x32i.2i.2sType</code></p>

4.7.3.10.17.4.1 HBM4 - Structured Mapping - Array (cont'd)

diagram 2 of 13	<p>HBM4-StandardStructuredTerminalMapping-Field1Type</p> <p>Field1 type HBM4-StandardStructuredTerminalMapping-Field1Type</p> <p>StandardTerminalBaseName type HBM4-Field1-StandardTerminalBaseNameType</p> <p>15</p>			
type	HBM4-StandardStructuredTerminalMapping-ArrayType, HBM4-Field1-StandardTerminalBaseNameType.			
list of enumerate values	Standard Terminal Base Name -> HBM4-Field1-StandardTerminalBaseNameType			
	1. CAPTUREWR	2. CATTRIP	3. NC	4. RESET_n
	5. SELECTWIR	6. SHIFTWR	7. UPDATEWR	8. VDDC
	9. VDDQ	10. VDDQL	11. VPP	12. VSS
	13. WRCK	14. WRST_n	15. WSI	
diagram 3 of 13	<p>HBM4-StandardStructuredTerminalMapping-Field2x2iType</p> <p>Field2x2i type HBM4-StandardStructuredTerminalMapping-Field2x2iType</p> <p>StandardTerminalBaseName type HBM4-Field2x2i-StandardTerminalBaseNameType</p> <p>Code type HBM4-Field2x2i-CodeType minIncl/maxIncl 0 1</p> <p>2</p>			
type	HBM4-StandardStructuredTerminalMapping-Field2x2iType, HBM4-Field2x2i-StandardTerminalBaseNameType, HBM4-Field2x2i-CodeType			
list of enumerate values	Standard Terminal Base Name -> HBM4-Field2x2i-StandardTerminalBaseNameType			
	1. RM			
diagram 4 of 13	<p>HBM4-StandardStructuredTerminalMapping-Field2x4iType</p> <p>Field2x4i type HBM4-StandardStructuredTerminalMapping-Field2x4iType</p> <p>StandardTerminalBaseName type HBM4-Field2x4i-StandardTerminalBaseNameType</p> <p>Code type HBM4-Field2x4i-CodeType minIncl/maxIncl 0 3</p> <p>4</p>			
type	HBM4-StandardStructuredTerminalMapping-Field2x4iType, HBM4-Field2x4i-StandardTerminalBaseNameType, HBM4-Field2x4i-CodeType			
list of enumerate values	Standard Terminal Base Name -> HBM4-Field2x4i-StandardTerminalBaseNameType			
	1. MRFU			

4.7.3.10.17.4.1 HBM4 - Structured Mapping - Array (cont'd)

diagram 5 of 13	<p>HBM4-StandardStructuredTerminalMapping-Field2x32iType</p> <p>StandardTerminalBaseName type HBM4-Field2x32i-StandardTerminalBaseNameType</p> <p>Code type HBM4-ChannelCodeType minIncl/maxIncl 0 31</p>	
type	HBM4-StandardStructuredTerminalMapping-Field2x32iType, HBM4-Field2x32i-StandardTerminalBaseNameType, HBM4-ChannelCodeType.	
list of enumerate values	Standard Terminal Base Name -> HBM4-Field2x32i-StandardTerminalBaseNameType 1. ARFU 2. APAR 3. AERR 4. WSO 5. RA	
diagram 6 of 13	<p>HBM4-StandardStructuredTerminalMapping-Field2x40iType</p> <p>StandardTerminalBaseName type HBM4-Field2x40i-StandardTerminalBaseNameType</p> <p>Code type HBM4-DirectAccessCodeType minIncl/maxIncl 0 39</p>	
type	HBM4-StandardStructuredTerminalMapping-Field2x40iType, HBM4-Field2x40i-StandardTerminalBaseNameType, HBM4-DirectAccessCodeType.	
list of enumerate values	Standard Terminal Base Name -> HBM4-Field2x40i-StandardTerminalBaseNameType 1. DA	
diagram 7 of 13	<p>HBM4-StandardStructuredTerminalMapping-Field4x32i.2iType</p> <p>StandardTerminalBaseName type HBM4-Field4x32i.2i-StandardTerminalBaseNameType</p> <p>Code1 type HBM4-ChannelCodeType minIncl/maxIncl 0 31</p> <p>Separator type xs:string</p> <p>Code2 type HBM4-Field4x32i.2i-Code2Type minIncl/maxIncl 0 1</p>	
type	HBM4-StandardStructuredTerminalMapping-Field4x32i.2iType, HBM4-Field4x32i.2i-StandardTerminalBaseNameType, HBM4-ChannelCodeType, HBM4-Field4x32i.2i-Code2Type	
list of enumerate values	Standard Terminal Base Name -> HBM4-Field2x32i-StandardTerminalBaseNameType 1. DPAR 2. DERR	

4.7.3.10.17.4.1 HBM4 - Structured Mapping - Array (cont'd)

diagram 8 of 13	<pre> classDiagram class Field4x32i.2s { type HBM4-StandardStructuredTerminalMapping-Field4x32i.2sType } class StandardTerminalBaseName { type HBM4-Field4x32i.2s-StandardTerminalBaseNameType } class Code1 { type HBM4-ChannelCodeType minIncl/maxIncl 0..31 } class Separator { type xs:string } class Code2 { type HBM4-Field4x32i.2s-Code2Type minIncl/maxIncl 2..32 } Field4x32i.2s --> StandardTerminalBaseName StandardTerminalBaseName --> Code1 Code1 --> Separator Separator --> Code2 </pre>				
type	HBM4-StandardStructuredTerminalMapping-Field4x32i.2sType, HBM4-Field4x32i.2s-StandardTerminalBaseNameType, HBM4-ChannelCodeType, HBM4-Field4x32i.2s-Code2Type				
list of enumerate values	Standard Terminal Base Name -> HBM4-Field4x32i.2s-StandardTerminalBaseNameType 1. CK Code 2 1. t 2. c				
diagram 9 of 13	<pre> classDiagram class Field4x32i.4i { type HBM4-StandardStructuredTerminalMapping-Field4x32i.4iType } class StandardTerminalBaseName { type HBM4-Field4x32i.4i-StandardTerminalBaseNameType } class Code1 { type HBM4-ChannelCodeType minIncl/maxIncl 0..31 } class Separator { type xs:string } class Code2 { type HBM4-Field4x32i.4i-Code2Type minIncl/maxIncl 0..4 } Field4x32i.4i --> StandardTerminalBaseName StandardTerminalBaseName --> Code1 Code1 --> Separator Separator --> Code2 </pre>				
type	HBM4-StandardStructuredTerminalMapping-Field4x32i.4iType, HBM4-Field4x32i.4i-StandardTerminalBaseNameType, HBM4-ChannelCodeType, HBM4-Field4x32i.4i-Code2Type				
list of enumerate values	Standard Terminal Base Name -> HBM4-Field4x32i.4i-StandardTerminalBaseNameType 1. ECC 2. SEV 3. RD				

4.7.3.10.17.4.1 HBM4 - Structured Mapping - Array (cont'd)

diagram 10 of 13	<pre> classDiagram class Field4x32i_8i { type HBM4-StandardStructuredTerminalMapping-Field4x32i.8iType } class StandardTerminalBaseName { type HBM4-Field4x32i.8i-StandardTerminalBaseNameType } class Code1 { type HBM4-ChannelCodeType minIncl/maxIncl 0 .. 31 } class Separator { type xs:string } class Code2 { type HBM4-Field4x32i.8i-Code2Type minIncl/maxIncl 0 .. 7 } Field4x32i_8i --> StandardTerminalBaseName StandardTerminalBaseName --> Code1 Code1 --> Separator Separator --> Code2 </pre>	
type	HBM4-StandardStructuredTerminalMapping-Field4x32i.8iType, HBM4-Field4x32i.8i-StandardTerminalBaseNameType, HBM4-ChannelCodeType, HBM4-Field4x32i.8i-Code2Type	
list of enumerate values	Standard Terminal Base Name -> HBM4-Field4x32i.8i-StandardTerminalBaseNameType	
1. C 2. DBI		
diagram 11 of 13	<pre> classDiagram class Field4x32i_10i { type HBM4-StandardStructuredTerminalMapping-Field4x32i.10iType } class StandardTerminalBaseName { type HBM4-Field4x32i.10i-StandardTerminalBaseNameType } class Code1 { type HBM4-ChannelCodeType minIncl/maxIncl 0 .. 31 } class Separator { type xs:string } class Code2 { type HBM4-Field4x32i.10i-Code2Type minIncl/maxIncl 0 .. 9 } Field4x32i_10i --> StandardTerminalBaseName StandardTerminalBaseName --> Code1 Code1 --> Separator Separator --> Code2 </pre>	
type	HBM4-StandardStructuredTerminalMapping-Field4x32i.10iType, HBM4-Field4x32i.10i-StandardTerminalBaseNameType, HBM4-ChannelCodeType, HBM4-Field4x32i.10i-Code2Type.	
list of enumerate values	Standard Terminal Base Name ->HBM4-Field4x32i.10i-StandardTerminalBaseNameType	
1. R		

4.7.3.10.17.4.1 HBM4 - Structured Mapping - Array (cont'd)

diagram 12 of 13		
type	HBM4-StandardStructuredTerminalMapping-Field4x32i.64iType, HBM4-Field4x32i.64i-StandardTerminalBaseNameType, HBM4-ChannelCodeType, HBM4-Field4x32i.64i-Code2Type	
list of enumerate values	Standard Terminal Base Name -> HBM4-Field4x32i.64i-StandardTerminalBaseNameType	
1. DQ		
diagram 13 of 13		
type	HBM4-StandardStructuredTerminalMapping-Field6x32i.2i.2sType, HBM4-Field6x32i.2i.2s-StandardTerminalBaseNameType, HBM4-ChannelCodeType, HBM4-Field6x32i.2i.2s-Code2Type, HBM4-Field6x32i.2i.2s-Code3Type	
list of enumerate values	Standard Terminal Base Name -> HBM4-Field6x32i.2i.2s-StandardTerminalBaseNameType	
1. WDQS	2. RDQS	
Code 3		
1. t	2. c	

4.7.3.10.18. HDMI Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI
diagram	<pre> classDiagram class HDMI-InterfaceFunctionType { <<HDMI-InterfaceFunctionType>> <<HDMI-TypeA>> <<HDMI-TypeB>> <<HDMI-TypeC>> <<HDMI-Source>> <<HDMI-Sink>> } class HDMI { <<HDMI-InterfaceFunctionType>> } HDMI < -- HDMI-InterfaceFunctionType HDMI < -- HDMI-TypeA HDMI < -- HDMI-TypeB HDMI < -- HDMI-TypeC HDMI < -- HDMI-Source HDMI < -- HDMI-Sink </pre> <p>The diagram illustrates the class hierarchy for the HDMI interface. At the top is a dashed-line box labeled "HDMI-InterfaceFunctionType". Inside this box are five sub-classes: "HDMI-TypeA", "HDMI-TypeB", "HDMI-TypeC", "HDMI-Source", and "HDMI-Sink". To the left of the box is a separate class "HDMI" with the same name and type. A line connects "HDMI" to the "HDMI-InterfaceFunctionType" box, indicating that "HDMI" is a specialization of "HDMI-InterfaceFunctionType".</p>
type	HDMI-InterfaceFunctionType , HDMI-TypeA-InterfaceFunctionType , HDMI-TypeB-InterfaceFunctionType , HDMI-TypeC-InterfaceFunctionType , HDMI-Source-InterfaceFunctionType , HDMI-Sink-InterfaceFunctionType

For more information about the HDMI Interface, refer to the “High-Definition Multimedia Interface Specification Version 1.3a” on the HDMI website.

4.7.3.10.18.1. HDMI-TypeA

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-TypeA																				
diagram	<pre> classDiagram class HDMI_TypeA { <<HDMI-TypeA>> <<HDMI-TypeA-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HDMI-TypeA-StandardTerminalNameAssignmentType>> } class Mapping { <<HDMI-TypeA-StandardTerminalMappingType>> } class StandardTerminalName { <<HDMI-TypeA-StandardTerminalNameType>> } HDMI_TypeA "1..x" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..x" --> Mapping Mapping "1..x" --> StandardTerminalName </pre>																				
type	HDMI-TypeA-InterfaceFunctionType , HDMI-TypeA-StandardTerminalNameAssignmentType , HDMI-TypeA-StandardTerminalMappingType , HDMI-TypeA-StandardTerminalNameType .																				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. TMDS Data2+</td><td>2. TMDS Data2 Shield</td><td>3. TMDS Data2-</td><td>4. TMDS Data1+</td></tr> <tr><td>5. TMDS Data1 Shield</td><td>6. TMDS Data1-</td><td>7. TMDS Data0+</td><td>8. TMDS Data0 Shield</td></tr> <tr><td>9. TMDS Data0-</td><td>10. TMDS Clock+</td><td>11. TMDS Clock Shield</td><td>12. TMDS Clock-</td></tr> <tr><td>13. CEC</td><td>14. SCL</td><td>15. SDA</td><td>16. DDC/CEC Ground</td></tr> <tr><td>17. +5V Power</td><td>18. Hot Plug Detect</td><td></td><td></td></tr> </table>	1. TMDS Data2+	2. TMDS Data2 Shield	3. TMDS Data2-	4. TMDS Data1+	5. TMDS Data1 Shield	6. TMDS Data1-	7. TMDS Data0+	8. TMDS Data0 Shield	9. TMDS Data0-	10. TMDS Clock+	11. TMDS Clock Shield	12. TMDS Clock-	13. CEC	14. SCL	15. SDA	16. DDC/CEC Ground	17. +5V Power	18. Hot Plug Detect		
1. TMDS Data2+	2. TMDS Data2 Shield	3. TMDS Data2-	4. TMDS Data1+																		
5. TMDS Data1 Shield	6. TMDS Data1-	7. TMDS Data0+	8. TMDS Data0 Shield																		
9. TMDS Data0-	10. TMDS Clock+	11. TMDS Clock Shield	12. TMDS Clock-																		
13. CEC	14. SCL	15. SDA	16. DDC/CEC Ground																		
17. +5V Power	18. Hot Plug Detect																				

4.7.3.10.18.2. HDMI-TypeB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-TypeB																				
diagram	<pre> classDiagram class HDMI_TypeB { <<HDMI-TypeB>> <<HDMI-TypeB-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HDMI-TypeB-StandardTerminalNameAssignmentType>> } class Mapping { <<HDMI-TypeB-StandardTerminalMappingType>> } class StandardTerminalName { <<HDMI-TypeB-StandardTerminalNameType>> } HDMI_TypeB "1..x" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..x" --> Mapping Mapping "1..x" --> StandardTerminalName </pre>																				
type	HDMI-TypeB-InterfaceFunctionType , HDMI-TypeB-StandardTerminalNameAssignmentType , HDMI-TypeB-StandardTerminalMappingType , HDMI-TypeB-StandardTerminalNameType .																				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. TMDS Data2+</td><td>2. TMDS Data2 Shield</td><td>3. TMDS Data2-</td><td>4. TMDS Data1+</td></tr> <tr><td>5. TMDS Data1 Shield</td><td>6. TMDS Data1-</td><td>7. TMDS Data0+</td><td>8. TMDS Data0 Shield</td></tr> <tr><td>9. TMDS Data0-</td><td>10. TMDS Clock+</td><td>11. TMDS Clock Shield</td><td>12. TMDS Clock-</td></tr> <tr><td>13. CEC</td><td>14. SCL</td><td>15. SDA</td><td>16. DDC/CEC Ground</td></tr> <tr><td>17. +5V Power</td><td>18. Hot Plug Detect</td><td></td><td></td></tr> </table>	1. TMDS Data2+	2. TMDS Data2 Shield	3. TMDS Data2-	4. TMDS Data1+	5. TMDS Data1 Shield	6. TMDS Data1-	7. TMDS Data0+	8. TMDS Data0 Shield	9. TMDS Data0-	10. TMDS Clock+	11. TMDS Clock Shield	12. TMDS Clock-	13. CEC	14. SCL	15. SDA	16. DDC/CEC Ground	17. +5V Power	18. Hot Plug Detect		
1. TMDS Data2+	2. TMDS Data2 Shield	3. TMDS Data2-	4. TMDS Data1+																		
5. TMDS Data1 Shield	6. TMDS Data1-	7. TMDS Data0+	8. TMDS Data0 Shield																		
9. TMDS Data0-	10. TMDS Clock+	11. TMDS Clock Shield	12. TMDS Clock-																		
13. CEC	14. SCL	15. SDA	16. DDC/CEC Ground																		
17. +5V Power	18. Hot Plug Detect																				

4.7.3.10.18.3. HDMI-TypeC

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-TypeC</code>																												
diagram	<pre> classDiagram class HDMI_TypeC { <<HDMI-TypeC>> <<HDMI-TypeC-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HDMI-TypeC-StandardTerminalNameAssignmentType>> } class Mapping { <<HDMI-TypeC-StandardTerminalMappingType>> } class StandardTerminalName { <<HDMI-TypeC-StandardTerminalNameType>> } HDMI_TypeC "1..oo" -- "1..oo" StandardTerminalNameAssignment StandardTerminalNameAssignment "27" -- "27" Mapping Mapping "27" -- "27" StandardTerminalName </pre>																												
type	<code>HDMI-TypeC-InterfaceFunctionType</code> , <code>HDMI-TypeC-StandardTerminalNameAssignmentType</code> , <code>HDMI-TypeC-StandardTerminalMappingType</code> , <code>HDMI-TypeC-StandardTerminalNameType</code> .																												
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr><td>1. TMDS Data2+</td><td>2. TMDS Data2 Shield</td><td>3. TMDS Data2-</td><td>4. TMDS Data1+</td></tr> <tr><td>5. TMDS Data1 Shield</td><td>6. TMDS Data1-</td><td>7. TMDS Data0+</td><td>8. TMDS Data0 Shield</td></tr> <tr><td>9. TMDS Data0-</td><td>10. TMDS Clock+</td><td>11. TMDS Clock Shield</td><td>12. TMDS Clock-</td></tr> <tr><td>13. TMDS Data5+</td><td>14. TMDS Data5 Shield</td><td>15. TMDS Data5-</td><td>16. TMDS Data4+</td></tr> <tr><td>17. TMDS Data4 Shield</td><td>18. TMDS Data4-</td><td>19. TMDS Data3+</td><td>20. TMDS Data3 Shield</td></tr> <tr><td>21. TMDS Data3-</td><td>22. CEC</td><td>23. SCL</td><td>24. SDA</td></tr> <tr><td>25. DDC/CEC Ground</td><td>26. +5V Power</td><td>27. Hot Plug Detect</td><td></td></tr> </table>	1. TMDS Data2+	2. TMDS Data2 Shield	3. TMDS Data2-	4. TMDS Data1+	5. TMDS Data1 Shield	6. TMDS Data1-	7. TMDS Data0+	8. TMDS Data0 Shield	9. TMDS Data0-	10. TMDS Clock+	11. TMDS Clock Shield	12. TMDS Clock-	13. TMDS Data5+	14. TMDS Data5 Shield	15. TMDS Data5-	16. TMDS Data4+	17. TMDS Data4 Shield	18. TMDS Data4-	19. TMDS Data3+	20. TMDS Data3 Shield	21. TMDS Data3-	22. CEC	23. SCL	24. SDA	25. DDC/CEC Ground	26. +5V Power	27. Hot Plug Detect	
1. TMDS Data2+	2. TMDS Data2 Shield	3. TMDS Data2-	4. TMDS Data1+																										
5. TMDS Data1 Shield	6. TMDS Data1-	7. TMDS Data0+	8. TMDS Data0 Shield																										
9. TMDS Data0-	10. TMDS Clock+	11. TMDS Clock Shield	12. TMDS Clock-																										
13. TMDS Data5+	14. TMDS Data5 Shield	15. TMDS Data5-	16. TMDS Data4+																										
17. TMDS Data4 Shield	18. TMDS Data4-	19. TMDS Data3+	20. TMDS Data3 Shield																										
21. TMDS Data3-	22. CEC	23. SCL	24. SDA																										
25. DDC/CEC Ground	26. +5V Power	27. Hot Plug Detect																											

4.7.3.10.18.4. HDMI-Source

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-Source</code>																				
diagram	<pre> classDiagram class HDMI_Source { <<HDMI-Source>> <<HDMI-Source-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HDMI-SourceStandardTerminalNameAssignmentType>> } class Mapping { <<HDMI-SourceStandardTerminalMappingType>> } class StandardTerminalName { <<HDMI-SourceStandardTerminalNameType>> } HDMI_Source "1..oo" -- "1..oo" StandardTerminalNameAssignment StandardTerminalNameAssignment "18" -- "18" Mapping Mapping "18" -- "18" StandardTerminalName </pre>																				
type	<code>HDMI-Source-InterfaceFunctionType</code> , <code>HDMI-SourceStandardTerminalNameAssignmentType</code> , <code>HDMI-SourceStandardTerminalMappingType</code> , <code>HDMI-SourceStandardTerminalNameType</code> .																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr><td>1. TMDS Data2+</td><td>2. TMDS Data2-</td><td>3. TMDS Data1+</td><td>4. TMDS Data1-</td></tr> <tr><td>5. TMDS Data0+</td><td>6. TMDS Data0-</td><td>7. TMDS Clock+</td><td>8. TMDS Clock-</td></tr> <tr><td>9. TMDS Data5+</td><td>10. TMDS Data5-</td><td>11. TMDS Data4+</td><td>12. TMDS Data4-</td></tr> <tr><td>13. TMDS Data3+</td><td>14. TMDS Data3-</td><td>15. CEC</td><td>16. SCL</td></tr> <tr><td>17. SDA</td><td>18. Hot Plug Detect</td><td></td><td></td></tr> </table>	1. TMDS Data2+	2. TMDS Data2-	3. TMDS Data1+	4. TMDS Data1-	5. TMDS Data0+	6. TMDS Data0-	7. TMDS Clock+	8. TMDS Clock-	9. TMDS Data5+	10. TMDS Data5-	11. TMDS Data4+	12. TMDS Data4-	13. TMDS Data3+	14. TMDS Data3-	15. CEC	16. SCL	17. SDA	18. Hot Plug Detect		
1. TMDS Data2+	2. TMDS Data2-	3. TMDS Data1+	4. TMDS Data1-																		
5. TMDS Data0+	6. TMDS Data0-	7. TMDS Clock+	8. TMDS Clock-																		
9. TMDS Data5+	10. TMDS Data5-	11. TMDS Data4+	12. TMDS Data4-																		
13. TMDS Data3+	14. TMDS Data3-	15. CEC	16. SCL																		
17. SDA	18. Hot Plug Detect																				

4.7.3.10.18.5. HDMI-Sink

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-Sink																								
diagram	<pre> classDiagram class HDMI_Sink { <<HDMI-Sink-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HDMI-SinkStandardTerminalNameAssignmentType>> } class Mapping { <<HDMI-SinkStandardTerminalMappingType>> } class StandardTerminalName { <<HDMI-SinkStandardTerminalNameType>> } HDMI_Sink "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the structure of the HDMI-Sink interface. It starts with an 'HDMI-Sink' object, which is associated with a 'StandardTerminalNameAssignment' object (multiplicity 1..infinity). This assignment is further refined by a 'Mapping' object (multiplicity 1..infinity), which then maps to a 'StandardTerminalName' object (multiplicity 1..infinity). A dashed box labeled 'constraints' is shown near the bottom right.</p>																								
type	HDMI-Sink-InterfaceFunctionType, HDMI-SinkStandardTerminalNameAssignmentType, HDMI-SinkStandardTerminalMappingType, HDMI-SinkStandardTerminalNameType.																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. TMDS Data2+</td> <td>2. TMDS Data2-</td> <td>3. TMDS Data1+</td> <td>4. TMDS Data1-</td> </tr> <tr> <td>5. TMDS Data0+</td> <td>6. TMDS Data0-</td> <td>7. TMDS Clock+</td> <td>8. TMDS Clock-</td> </tr> <tr> <td>9. TMDS Data5+</td> <td>10. TMDS Data5-</td> <td>11. TMDS Data4+</td> <td>12. TMDS Data4-</td> </tr> <tr> <td>13. TMDS Data3+</td> <td>14. TMDS Data3-</td> <td>15. CEC</td> <td>16. SCL</td> </tr> <tr> <td>17. SDA</td> <td>18. Hot Plug Detect</td> <td></td> <td></td> </tr> </tbody> </table>	Mapping/StandardTerminalName				1. TMDS Data2+	2. TMDS Data2-	3. TMDS Data1+	4. TMDS Data1-	5. TMDS Data0+	6. TMDS Data0-	7. TMDS Clock+	8. TMDS Clock-	9. TMDS Data5+	10. TMDS Data5-	11. TMDS Data4+	12. TMDS Data4-	13. TMDS Data3+	14. TMDS Data3-	15. CEC	16. SCL	17. SDA	18. Hot Plug Detect		
Mapping/StandardTerminalName																									
1. TMDS Data2+	2. TMDS Data2-	3. TMDS Data1+	4. TMDS Data1-																						
5. TMDS Data0+	6. TMDS Data0-	7. TMDS Clock+	8. TMDS Clock-																						
9. TMDS Data5+	10. TMDS Data5-	11. TMDS Data4+	12. TMDS Data4-																						
13. TMDS Data3+	14. TMDS Data3-	15. CEC	16. SCL																						
17. SDA	18. Hot Plug Detect																								

4.7.3.10.19. HSI Interface

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HSI																				
diagram	<pre> classDiagram class HSI { <<HSI-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<HSI-StandardTerminalNameAssignmentType>> } class HSI_MandatoryStandardTerminalMapping { <<HSI-MandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<HSI-StandardTerminalNameType>> } HSI "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" HSI_MandatoryStandardTerminalMapping HSI_MandatoryStandardTerminalMapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the structure of the HSI interface. It starts with an 'HSI' object, which is associated with a 'StandardTerminalNameAssignment' object (multiplicity 1..infinity). This assignment is further refined by a 'HSI-MandatoryStandardTerminalMapping' object (multiplicity 1..infinity), which then maps to a 'StandardTerminalName' object (multiplicity 1..infinity). A dashed box labeled 'constraints' is shown near the bottom right.</p>																				
type	HSI-InterfaceFunctionType, HSI-StandardTerminalNameAssignmentType, HSI-MandatoryStandardTerminalMappingType, HSI-MandatoryStandardTerminalNameType, HSI-OptionalStandardTerminalMappingType, HSI-OptionalStandardTerminalNameType.																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CADATA</td> <td>2. CAFLAG</td> <td>3. ACREADY</td> <td>4. ACDATA</td> </tr> <tr> <td>5. ACFLAG</td> <td>6. CAREADY</td> <td></td> <td></td> </tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. CAWAKE</td> <td>2. ACWAKE</td> <td></td> <td></td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. CADATA	2. CAFLAG	3. ACREADY	4. ACDATA	5. ACFLAG	6. CAREADY			OptionalMapping/StandardTerminalName				1. CAWAKE	2. ACWAKE		
MandatoryMapping/StandardTerminalName																					
1. CADATA	2. CAFLAG	3. ACREADY	4. ACDATA																		
5. ACFLAG	6. CAREADY																				
OptionalMapping/StandardTerminalName																					
1. CAWAKE	2. ACWAKE																				

For more information about the HSI Interface, refer to the MIPI Alliance standard Specification for High-Speed Synchronous Serial Interface Version 1.01.

4.7.3.10.20. HTI Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HTI								
diagram	<pre> classDiagram HTI "type HTI-InterfaceFunctionType" --> StandardTerminalNameAssignment "type HTI-StandardTerminalNameAssignmentType" StandardTerminalNameAssignment "type HTI-StandardTerminalNameAssignmentType" --> Mapping "type HTI-StandardTerminalMappingType" Mapping "type HTI-StandardTerminalMappingType" --> StandardTerminalName "type HTI-StandardTerminalNameType" StandardTerminalName "type HTI-StandardTerminalNameType" *--> StandardTerminalNameAssignment "type HTI-StandardTerminalNameAssignmentType" </pre>								
type	HTI-InterfaceFunctionType , HTI-StandardTerminalNameAssignmentType , HTI-StandardTerminalMappingType , HTI-StandardTerminalNameType .								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Lane1</td> <td>2. Lane2</td> <td>3. Lane3</td> <td>4. Lane4</td> </tr> <tr> <td>5. Lane5</td> <td>6. Lane6</td> <td></td> <td></td> </tr> </table>	1. Lane1	2. Lane2	3. Lane3	4. Lane4	5. Lane5	6. Lane6		
1. Lane1	2. Lane2	3. Lane3	4. Lane4						
5. Lane5	6. Lane6								

For more information about the HTI Interface, refer to the MIPI Alliance standard Specification for High-Speed Trace Interface (HTI) Version 1.1.

4.7.3.10.21. HTIv1 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HTIv1								
diagram	<pre> classDiagram HTIv1 "type HTIv1-InterfaceFunctionType" --> StandardTerminalNameAssignment "type HTIv1-StandardTerminalNameAssignmentType" StandardTerminalNameAssignment "type HTIv1-StandardTerminalNameAssignmentType" --> Mapping "type HTIv1-StandardTerminalMappingType" Mapping "type HTIv1-StandardTerminalMappingType" --> StandardTerminalName "type HTIv1-StandardTerminalNameType" StandardTerminalName "type HTIv1-StandardTerminalNameType" *--> StandardTerminalNameAssignment "type HTIv1-StandardTerminalNameAssignmentType" </pre>								
type	HTIv1-InterfaceFunctionType , HTIv1-StandardTerminalNameAssignmentType , HTIv1-StandardTerminalMappingType , HTIv1-StandardTerminalNameType .								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Lane1</td> <td>2. Lane2</td> <td>3. Lane3</td> <td>4. Lane4</td> </tr> <tr> <td>5. Lane5</td> <td>6. Lane6</td> <td>7. Lane7</td> <td>8. Lane8</td> </tr> </table>	1. Lane1	2. Lane2	3. Lane3	4. Lane4	5. Lane5	6. Lane6	7. Lane7	8. Lane8
1. Lane1	2. Lane2	3. Lane3	4. Lane4						
5. Lane5	6. Lane6	7. Lane7	8. Lane8						

4.7.3.10.22. I2C Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/I2C							
diagram	<pre> classDiagram class I2C { <<I2C>> } class I2C_InterfaceFunctionType { <<I2C-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<I2C-StandardTerminalNameAssignmentType>> } class Mapping { <<I2C-StandardTerminalMappingType>> } class StandardTerminalName { <<I2C-StandardTerminalNameType>> } I2C "1" -- "*" I2C_InterfaceFunctionType I2C_InterfaceFunctionType "*" -- "1..*" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" -- "1..*" Mapping Mapping "*" -- "2" StandardTerminalName </pre>							
type	I2C-InterfaceFunctionType , I2C-StandardTerminalNameAssignmentType , I2C-StandardTerminalMappingType , I2C-StandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. SCL</td> <td style="padding: 2px;">2. SDA</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>				1. SCL	2. SDA		
1. SCL	2. SDA							

4.7.3.10.23. I3C Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/I3C							
diagram	<pre> classDiagram class I3C { <<I3C>> } class I3C_InterfaceFunctionType { <<I3C-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<I3C-StandardTerminalNameAssignmentType>> } class Mapping { <<I3C-StandardTerminalMappingType>> } class StandardTerminalName { <<I3C-StandardTerminalNameType>> } I3C "1" -- "*" I3C_InterfaceFunctionType I3C_InterfaceFunctionType "*" -- "1..*" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" -- "1..*" Mapping Mapping "*" -- "2" StandardTerminalName </pre>							
type	I3CType , I3C-StandardTerminalNameAssignmentType , I3C-StandardTerminalMappingType , I3C-StandardTerminalNameType							
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. SDA</td> <td style="padding: 2px;">2. SCL</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>				1. SDA	2. SCL		
1. SDA	2. SCL							

4.7.3.10.24. LLI-Serial Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/LLI-Serial							
diagram	<pre> classDiagram class LLI_Serial { <<LLI-Serial>> <<type LLI-Serial-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<StandardTerminalNameAssignment>> <<type LLI-Serial-StandardTerminalNameAssignmentType>> } class Mapping { <<Mapping>> <<type LLI-Serial-StandardTerminalMappingType>> } class StandardTerminalName { <<StandardTerminalName>> <<type LLI-Serial-StandardTerminalNameType>> } LLI_Serial "1..x" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "4" --> Mapping Mapping --> StandardTerminalName </pre>							
type	LLI-Serial-InterfaceFunctionType , LLI-Serial-StandardTerminalNameAssignmentType , LLI-Serial-StandardTerminalMappingType , LLI-Serial-StandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. TXDP</td> <td>2. TXDN</td> <td>3. RXDP</td> <td>4. RXDN</td> </tr> </table>				1. TXDP	2. TXDN	3. RXDP	4. RXDN
1. TXDP	2. TXDN	3. RXDP	4. RXDN					

For more information about the LLI-Serial Interface, refer to the MIPI Alliance standard Specification for Low Latency Interface (LLI) Version 2.1.

4.7.3.10.25. LVSTL06 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/LVSTL06							
diagram	<pre> classDiagram class LVSTL06 { <<LVSTL06>> <<type LVSTL06-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<StandardTerminalNameAssignment>> <<type LVSTL06-StandardTerminalNameAssignmentType>> } class Mapping { <<Mapping>> <<type LVSTL06-StandardTerminalMappingType>> } class StandardTerminalName { <<StandardTerminalName>> <<type LVSTL06-StandardTerminalNameType>> } LVSTL06 "1..x" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "3" --> Mapping Mapping --> StandardTerminalName </pre>							
type	LVSTL06-InterfaceFunctionType , LVSTL06-StandardTerminalNameAssignmentType , LVSTL06-StandardTerminalMappingType , LVSTL06-StandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. VDDQ</td> <td>2. VSSQ</td> <td>3. DQ</td> <td></td> </tr> </table>				1. VDDQ	2. VSSQ	3. DQ	
1. VDDQ	2. VSSQ	3. DQ						

For more information about the LVSTL06 Interface, refer to the JEDEC Standard JESD8-29.

4.7.3.10.26. MMC Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/MultiMediaCard
diagram	<pre> classDiagram class MultiMediaCard { type MultiMediaCard-InterfaceFunctionsType } class MultiMediaCard-InterfaceFunctionsType { eMMC type eMMC-InterfaceFunctionType MMC type MMC-InterfaceFunctionType SPI-Mode type SPI-Mode-InterfaceFunctionType } MultiMediaCard "1" -- "1" MultiMediaCard-InterfaceFunctionsType </pre>
type	MultiMediaCard-InterfaceFunctionsType , eMMC-InterfaceFunctionType , MMC-InterfaceFunctionType , SPI-Mode-InterfaceFunctionType .

4.7.3.10.26.1. eMMC Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/MultiMediaCard/MMC																				
diagram	<pre> classDiagram class eMMC { type eMMC-InterfaceFunctionType } class eMMC-InterfaceFunctionType { StandardTerminalNameAssignment type eMMC-StandardTerminalNameAssignmentType Mapping type eMMC-StandardTerminalMappingType StandardTerminalName type eMMC-StandardTerminalNameType } eMMC "1" -- "1" eMMC-InterfaceFunctionType eMMC-InterfaceFunctionType "1..oo" -- "1" StandardTerminalNameAssignment eMMC-InterfaceFunctionType "1" -- "1" Mapping eMMC-InterfaceFunctionType "1" -- "1" StandardTerminalName </pre>																				
type	eMMC-InterfaceFunctionType , eMMC-StandardTerminalNameAssignmentType , eMMC-StandardTerminalMappingType , eMMC-StandardTerminalNameType																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CLK</td><td>2. DS</td><td>3. DAT[0]</td><td>4. DAT[1]</td></tr> <tr> <td>5. DAT[2]</td><td>6. DAT[3]</td><td>7. DAT[4]</td><td>8. DAT[5]</td></tr> <tr> <td>9. DAT[6]</td><td>10. DAT[7]</td><td>11. CMD</td><td>12. RST_n</td></tr> <tr> <td>13. VCC</td><td>14. VCCQ</td><td>15. VSS</td><td>16. VSSQ</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. CLK	2. DS	3. DAT[0]	4. DAT[1]	5. DAT[2]	6. DAT[3]	7. DAT[4]	8. DAT[5]	9. DAT[6]	10. DAT[7]	11. CMD	12. RST_n	13. VCC	14. VCCQ	15. VSS	16. VSSQ
Mapping/StandardTerminalName																					
1. CLK	2. DS	3. DAT[0]	4. DAT[1]																		
5. DAT[2]	6. DAT[3]	7. DAT[4]	8. DAT[5]																		
9. DAT[6]	10. DAT[7]	11. CMD	12. RST_n																		
13. VCC	14. VCCQ	15. VSS	16. VSSQ																		

For more information about the eMMC Interface, refer to the JEDEC standard JESD84-B51A.

4.7.3.10.26.2. MMC Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MultiMediaCardMMC																
diagram	<pre> classDiagram class MMC { <<MMC-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<MMC-StandardTerminalNameAssignmentType>> } class Mapping { <<MMC-StandardTerminalMappingType>> } class StandardTerminalName { <<MMC-StandardTerminalNameType>> } MMC "1..oo" --> "1..oo" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..oo" --> "1..13" Mapping Mapping "1..13" --> StandardTerminalName </pre>																
type	MMC-InterfaceFunctionType , MMC-StandardTerminalNameAssignmentType , MMC-StandardTerminalMappingType , MMC-StandardTerminalNameType																
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. CLK</td> <td>2. DAT[0]</td> <td>3. DAT[1]</td> <td>4. DAT[2]</td> </tr> <tr> <td>5. DAT[3]</td> <td>6. DAT[4]</td> <td>7. DAT[5]</td> <td>8. DAT[6]</td> </tr> <tr> <td>9. DAT[7]</td> <td>10. CMD</td> <td>11. VSS1</td> <td>12. VSS2</td> </tr> <tr> <td>13. VDD</td> <td></td> <td></td> <td></td> </tr> </table>	1. CLK	2. DAT[0]	3. DAT[1]	4. DAT[2]	5. DAT[3]	6. DAT[4]	7. DAT[5]	8. DAT[6]	9. DAT[7]	10. CMD	11. VSS1	12. VSS2	13. VDD			
1. CLK	2. DAT[0]	3. DAT[1]	4. DAT[2]														
5. DAT[3]	6. DAT[4]	7. DAT[5]	8. DAT[6]														
9. DAT[7]	10. CMD	11. VSS1	12. VSS2														
13. VDD																	

For more information about the MMC Interface, refer to the JEDEC standard JESD84-B42.

4.7.3.10.26.3. SPI-Mode Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MultiMediaCard/SPI-Mode								
diagram	<pre> classDiagram class SPI_Mode { <<SPI-Mode-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<SPI-Mode-StandardTerminalNameAssignmentType>> } class Mapping { <<SPI-Mode-StandardTerminalMappingType>> } class StandardTerminalName { <<SPI-Mode-StandardTerminalNameType>> } SPI_Mode "1..oo" --> "1..oo" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..oo" --> "1..7" Mapping Mapping "1..7" --> StandardTerminalName </pre>								
type	SPI-Mode-InterfaceFunctionType , SPI-Mode-StandardTerminalNameAssignmentType , SPI-Mode-StandardTerminalMappingType , SPI-Mode-StandardTerminalNameType								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. CS</td> <td>2. DI</td> <td>3. VSS</td> <td>4. VDD</td> </tr> <tr> <td>5. SCLK</td> <td>6. VSS2</td> <td>7. DO</td> <td></td> </tr> </table>	1. CS	2. DI	3. VSS	4. VDD	5. SCLK	6. VSS2	7. DO	
1. CS	2. DI	3. VSS	4. VDD						
5. SCLK	6. VSS2	7. DO							

For more information about the SPI Mode Interface, refer to the JEDEC standard JESD84-B42.

4.7.3.10.27. MII Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI
diagram	<pre> graph LR MII[MII type MII-InterfaceFunctionsType] --- MIIfunc[MII-InterfaceFunctionsType] MIIfunc --- CGMII[CGMII type CGMII-InterfaceFunctionType] MIIfunc --- GMII[GMII type GMII-InterfaceFunctionType] MIIfunc --- MII[MII type MII-InterfaceFunctionType] MIIfunc --- RGMII[RGMII type RGMII-InterfaceFunctionType] MIIfunc --- RMII[RMII type RMII-InterfaceFunctionType] MIIfunc --- SMII[SMII type SMII-InterfaceFunctionType] MIIfunc --- XGMII[XGMII type XGMII-InterfaceFunctionType] MIIfunc --- XLGMII[XLGMII type XLGMII-InterfaceFunctionType] </pre>
type	MII-InterfaceFunctionsType , CGMII-InterfaceFunctionType , GMII-InterfaceFunctionType , MII-InterfaceFunctionType , RGMII-InterfaceFunctionType , RMII-InterfaceFunctionType , SMII-InterfaceFunctionType , XGMII-InterfaceFunctionType , XLGMII-InterfaceFunctionType .

4.7.3.10.27.1. CGMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/CGMII																																																																																																								
diagram	<pre> classDiagram CGMII < -- CGMII-InterfaceFunctionType CGMII-InterfaceFunctionType --> StandardTerminalNameAssignment : 1..∞ StandardTerminalNameAssignment --> Mapping : 1..1 Mapping --> CGMII-StandardTerminalNameAssignmentType : 1..1 Mapping --> CGMII-StandardTerminalMappingType : 1..1 CGMII-StandardTerminalNameAssignmentType --> CGMII-StandardTerminalNameType : 146 CGMII-StandardTerminalNameType --> constraints </pre>																																																																																																								
type	CGMIIType, CGMII-StandardTerminalNameAssignmentType, CGMII-StandardTerminalMappingType, CGMII-StandardTerminalNameType																																																																																																								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. TX_CLK</td><td>2. TXC0</td><td>3. TXC1</td><td>4. TXC2</td></tr> <tr><td>5. TXC3</td><td>6. TXC4</td><td>7. TXC5</td><td>8. TXC6</td></tr> <tr><td>9. TXC7</td><td>10. TXD[0]</td><td>11. TXD[1]</td><td>12. TXD[2]</td></tr> <tr><td>13. TXD[3]</td><td>14. TXD[4]</td><td>15. TXD[5]</td><td>16. TXD[6]</td></tr> <tr><td>17. TXD[7]</td><td>18. TXD[8]</td><td>19. TXD[9]</td><td>20. TXD[10]</td></tr> <tr><td>21. TXD[11]</td><td>22. TXD[12]</td><td>23. TXD[13]</td><td>24. TXD[14]</td></tr> <tr><td>25. TXD[15]</td><td>26. TXD[16]</td><td>27. TXD[17]</td><td>28. TXD[18]</td></tr> <tr><td>29. TXD[19]</td><td>30. TXD[20]</td><td>31. TXD[21]</td><td>32. TXD[22]</td></tr> <tr><td>33. TXD[23]</td><td>34. TXD[24]</td><td>35. TXD[25]</td><td>36. TXD[26]</td></tr> <tr><td>37. TXD[27]</td><td>38. TXD[28]</td><td>39. TXD[29]</td><td>40. TXD[30]</td></tr> <tr><td>41. TXD[31]</td><td>42. TXD[32]</td><td>43. TXD[33]</td><td>44. TXD[34]</td></tr> <tr><td>45. TXD[35]</td><td>46. TXD[36]</td><td>47. TXD[37]</td><td>48. TXD[38]</td></tr> <tr><td>49. TXD[39]</td><td>50. TXD[40]</td><td>51. TXD[41]</td><td>52. TXD[42]</td></tr> <tr><td>53. TXD[43]</td><td>54. TXD[44]</td><td>55. TXD[45]</td><td>56. TXD[46]</td></tr> <tr><td>57. TXD[47]</td><td>58. TXD[48]</td><td>59. TXD[49]</td><td>60. TXD[50]</td></tr> <tr><td>61. TXD[51]</td><td>62. TXD[52]</td><td>63. TXD[53]</td><td>64. TXD[54]</td></tr> <tr><td>65. TXD[55]</td><td>66. TXD[56]</td><td>67. TXD[57]</td><td>68. TXD[58]</td></tr> <tr><td>69. TXD[59]</td><td>70. TXD[60]</td><td>71. TXD[61]</td><td>72. TXD[62]</td></tr> <tr><td>73. TXD[63]</td><td>74. RX_CLK</td><td>75. RXC0</td><td>76. RXC1</td></tr> <tr><td>77. RXC2</td><td>78. RXC3</td><td>79. RXC4</td><td>80. RXC5</td></tr> <tr><td>81. RXC6</td><td>82. RXC7</td><td>83. RXD[0]</td><td>84. RXD[1]</td></tr> <tr><td>85. RXD[2]</td><td>86. RXD[3]</td><td>87. RXD[4]</td><td>88. RXD[5]</td></tr> <tr><td>89. RXD[6]</td><td>90. RXD[7]</td><td>91. RXD[8]</td><td>92. RXD[9]</td></tr> <tr><td>93. RXD[10]</td><td>94. RXD[11]</td><td>95. RXD[12]</td><td>96. RXD[13]</td></tr> <tr><td>97. RXD[14]</td><td>98. RXD[15]</td><td>99. RXD[16]</td><td>100. RXD[17]</td></tr> <tr><td>101. RXD[18]</td><td>102. RXD[19]</td><td>103. RXD[20]</td><td>104. RXD[21]</td></tr> </tbody> </table>	1. TX_CLK	2. TXC0	3. TXC1	4. TXC2	5. TXC3	6. TXC4	7. TXC5	8. TXC6	9. TXC7	10. TXD[0]	11. TXD[1]	12. TXD[2]	13. TXD[3]	14. TXD[4]	15. TXD[5]	16. TXD[6]	17. TXD[7]	18. TXD[8]	19. TXD[9]	20. TXD[10]	21. TXD[11]	22. TXD[12]	23. TXD[13]	24. TXD[14]	25. TXD[15]	26. TXD[16]	27. TXD[17]	28. TXD[18]	29. TXD[19]	30. TXD[20]	31. TXD[21]	32. TXD[22]	33. TXD[23]	34. TXD[24]	35. TXD[25]	36. TXD[26]	37. TXD[27]	38. TXD[28]	39. TXD[29]	40. TXD[30]	41. TXD[31]	42. TXD[32]	43. TXD[33]	44. TXD[34]	45. TXD[35]	46. TXD[36]	47. TXD[37]	48. TXD[38]	49. TXD[39]	50. TXD[40]	51. TXD[41]	52. TXD[42]	53. TXD[43]	54. TXD[44]	55. TXD[45]	56. TXD[46]	57. TXD[47]	58. TXD[48]	59. TXD[49]	60. TXD[50]	61. TXD[51]	62. TXD[52]	63. TXD[53]	64. TXD[54]	65. TXD[55]	66. TXD[56]	67. TXD[57]	68. TXD[58]	69. TXD[59]	70. TXD[60]	71. TXD[61]	72. TXD[62]	73. TXD[63]	74. RX_CLK	75. RXC0	76. RXC1	77. RXC2	78. RXC3	79. RXC4	80. RXC5	81. RXC6	82. RXC7	83. RXD[0]	84. RXD[1]	85. RXD[2]	86. RXD[3]	87. RXD[4]	88. RXD[5]	89. RXD[6]	90. RXD[7]	91. RXD[8]	92. RXD[9]	93. RXD[10]	94. RXD[11]	95. RXD[12]	96. RXD[13]	97. RXD[14]	98. RXD[15]	99. RXD[16]	100. RXD[17]	101. RXD[18]	102. RXD[19]	103. RXD[20]	104. RXD[21]
1. TX_CLK	2. TXC0	3. TXC1	4. TXC2																																																																																																						
5. TXC3	6. TXC4	7. TXC5	8. TXC6																																																																																																						
9. TXC7	10. TXD[0]	11. TXD[1]	12. TXD[2]																																																																																																						
13. TXD[3]	14. TXD[4]	15. TXD[5]	16. TXD[6]																																																																																																						
17. TXD[7]	18. TXD[8]	19. TXD[9]	20. TXD[10]																																																																																																						
21. TXD[11]	22. TXD[12]	23. TXD[13]	24. TXD[14]																																																																																																						
25. TXD[15]	26. TXD[16]	27. TXD[17]	28. TXD[18]																																																																																																						
29. TXD[19]	30. TXD[20]	31. TXD[21]	32. TXD[22]																																																																																																						
33. TXD[23]	34. TXD[24]	35. TXD[25]	36. TXD[26]																																																																																																						
37. TXD[27]	38. TXD[28]	39. TXD[29]	40. TXD[30]																																																																																																						
41. TXD[31]	42. TXD[32]	43. TXD[33]	44. TXD[34]																																																																																																						
45. TXD[35]	46. TXD[36]	47. TXD[37]	48. TXD[38]																																																																																																						
49. TXD[39]	50. TXD[40]	51. TXD[41]	52. TXD[42]																																																																																																						
53. TXD[43]	54. TXD[44]	55. TXD[45]	56. TXD[46]																																																																																																						
57. TXD[47]	58. TXD[48]	59. TXD[49]	60. TXD[50]																																																																																																						
61. TXD[51]	62. TXD[52]	63. TXD[53]	64. TXD[54]																																																																																																						
65. TXD[55]	66. TXD[56]	67. TXD[57]	68. TXD[58]																																																																																																						
69. TXD[59]	70. TXD[60]	71. TXD[61]	72. TXD[62]																																																																																																						
73. TXD[63]	74. RX_CLK	75. RXC0	76. RXC1																																																																																																						
77. RXC2	78. RXC3	79. RXC4	80. RXC5																																																																																																						
81. RXC6	82. RXC7	83. RXD[0]	84. RXD[1]																																																																																																						
85. RXD[2]	86. RXD[3]	87. RXD[4]	88. RXD[5]																																																																																																						
89. RXD[6]	90. RXD[7]	91. RXD[8]	92. RXD[9]																																																																																																						
93. RXD[10]	94. RXD[11]	95. RXD[12]	96. RXD[13]																																																																																																						
97. RXD[14]	98. RXD[15]	99. RXD[16]	100. RXD[17]																																																																																																						
101. RXD[18]	102. RXD[19]	103. RXD[20]	104. RXD[21]																																																																																																						

4.7.3.10.27.1 CGMII – Mandatory Mapping (cont'd)

list of enumerate values (cont.)	105. RXD[22]	106. RXD[23]	107. RXD[24]	108. RXD[25]
	109. RXD[26]	110. RXD[27]	111. RXD[28]	112. RXD[29]
	113. RXD[30]	114. RXD[31]	115. RXD[32]	116. RXD[33]
	117. RXD[34]	118. RXD[35]	119. RXD[36]	120. RXD[37]
	121. RXD[38]	122. RXD[39]	123. RXD[40]	124. RXD[41]
	125. RXD[42]	126. RXD[43]	127. RXD[44]	128. RXD[45]
	129. RXD[46]	130. RXD[47]	131. RXD[48]	132. RXD[49]
	133. RXD[50]	134. RXD[51]	135. RXD[52]	136. RXD[53]
	137. RXD[54]	138. RXD[55]	139. RXD[56]	140. RXD[57]
	141. RXD[58]	142. RXD[59]	143. RXD[60]	144. RXD[61]
	145. RXD[62]	146. RXD[63]		

For more information about the CGMII Interface, refer to the IEEE standard IEEE 802.3ba-2010.

4.7.3.10.27.2. GMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/GMII																																
diagram	<pre> classDiagram class GMII { <<GMII-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<GMII-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<GMII-MandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<GMII-MandatoryStandardTerminalNameType>> } class OptionalMapping { <<GMII-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<GMII-OptionalStandardTerminalNameType>> } GMII "1..n" -- "n" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..n" -- "n" MandatoryMapping StandardTerminalNameAssignment "1..n" -- "n" OptionalMapping MandatoryMapping "*" -- "n" StandardTerminalName OptionalMapping "*" -- "n" StandardTerminalName </pre>																																
type	GMIIType, GMII-StandardTerminalNameAssignmentType, GMII-MandatoryStandardTerminalMappingType, GMII-MandatoryStandardTerminalNameType, GMII-OptionalStandardTerminalMappingType, GMII-OptionalStandardTerminalNameType																																
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. GTX_CLK</td><td>2. TXD[0]</td><td>3. TXD[1]</td><td>4. TXD[2]</td></tr> <tr><td>5. TXD[3]</td><td>6. TXD[4]</td><td>7. TXD[5]</td><td>8. TXD[6]</td></tr> <tr><td>9. TXD[7]</td><td>10. TX_EN</td><td>11. TX_ER</td><td>12. RX_CLK</td></tr> <tr><td>13. RXD[0]</td><td>14. RXD[1]</td><td>15. RXD[2]</td><td>16. RXD[3]</td></tr> <tr><td>17. RXD[4]</td><td>18. RXD[5]</td><td>19. RXD[6]</td><td>20. RXD[7]</td></tr> <tr><td>21. RX_DV</td><td>22. RX_ER</td><td>23. COL</td><td>24. CRS</td></tr> <tr><td>25. MDC</td><td>26. MDIO</td><td></td><td></td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. TX_CLK</td><td></td><td></td><td></td></tr> </table>	1. GTX_CLK	2. TXD[0]	3. TXD[1]	4. TXD[2]	5. TXD[3]	6. TXD[4]	7. TXD[5]	8. TXD[6]	9. TXD[7]	10. TX_EN	11. TX_ER	12. RX_CLK	13. RXD[0]	14. RXD[1]	15. RXD[2]	16. RXD[3]	17. RXD[4]	18. RXD[5]	19. RXD[6]	20. RXD[7]	21. RX_DV	22. RX_ER	23. COL	24. CRS	25. MDC	26. MDIO			1. TX_CLK			
1. GTX_CLK	2. TXD[0]	3. TXD[1]	4. TXD[2]																														
5. TXD[3]	6. TXD[4]	7. TXD[5]	8. TXD[6]																														
9. TXD[7]	10. TX_EN	11. TX_ER	12. RX_CLK																														
13. RXD[0]	14. RXD[1]	15. RXD[2]	16. RXD[3]																														
17. RXD[4]	18. RXD[5]	19. RXD[6]	20. RXD[7]																														
21. RX_DV	22. RX_ER	23. COL	24. CRS																														
25. MDC	26. MDIO																																
1. TX_CLK																																	

For more information about the GMII Interface, refer to the IEEE standard IEEE 802.3z-1998.

4.7.3.10.27.3. MII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/MII																												
diagram	<pre> classDiagram class MII { <<MII-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<MII-StandardTerminalNameAssignment>> } class MII_StandardTerminalNameAssignmentType { <<MII-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<MII-MandatoryStandardTerminalMapping>> } class OptionalMapping { <<MII-OptionalStandardTerminalMapping>> } class StandardTerminalName { <<MII-StandardTerminalName>> } MII "1..∞" -- "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" -- "1..∞" MII_StandardTerminalNameAssignmentType MII_StandardTerminalNameAssignmentType "1..∞" -- "1..16" MandatoryMapping MII_StandardTerminalNameAssignmentType "1..∞" -- "0..3" OptionalMapping MandatoryMapping "1..16" -- "1..16" StandardTerminalName OptionalMapping "0..3" -- "1..16" StandardTerminalName </pre>																												
type	MIIType, RMII-StandardTerminalNameAssignmentType, MII-MandatoryStandardTerminalMappingType, MII-MandatoryStandardTerminalNameType, MII-OptionalStandardTerminalMappingType, MII-OptionalStandardTerminalNameType.																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. TXD[0]</td> <td>2. TXD[1]</td> <td>3. TXD[2]</td> <td>4. TXD[3]</td> </tr> <tr> <td>5. TX_EN</td> <td>6. TX_ERR</td> <td>7. TX_CLK</td> <td>8. RXD[0]</td> </tr> <tr> <td>9. RXD[1]</td> <td>10. RXD[2]</td> <td>11. RXD[3]</td> <td>12. RX_DV</td> </tr> <tr> <td>13. RX_ERR</td> <td>14. RX_CLK</td> <td>15. COLL</td> <td>16. CRS</td> </tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. SPEED</td> <td>2. DUPLEX</td> <td>3. LINK</td> <td></td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]	5. TX_EN	6. TX_ERR	7. TX_CLK	8. RXD[0]	9. RXD[1]	10. RXD[2]	11. RXD[3]	12. RX_DV	13. RX_ERR	14. RX_CLK	15. COLL	16. CRS	OptionalMapping/StandardTerminalName				1. SPEED	2. DUPLEX	3. LINK	
MandatoryMapping/StandardTerminalName																													
1. TXD[0]	2. TXD[1]	3. TXD[2]	4. TXD[3]																										
5. TX_EN	6. TX_ERR	7. TX_CLK	8. RXD[0]																										
9. RXD[1]	10. RXD[2]	11. RXD[3]	12. RX_DV																										
13. RX_ERR	14. RX_CLK	15. COLL	16. CRS																										
OptionalMapping/StandardTerminalName																													
1. SPEED	2. DUPLEX	3. LINK																											

For more information about the MII Interface, refer to the IEEE standard IEEE 802.3u-1995.

4.7.3.10.27.4. RGMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/RGMII																
diagram	<pre> classDiagram class RGMII-InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName RGMII-InterfaceFunctionType "1..*" -- "1..*" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..*" -- "1..*" Mapping Mapping "1..*" -- "1..*" StandardTerminalName constraints </pre>																
type	RGMIIType, RGMII-StandardTerminalNameAssignmentType, RGMII-StandardTerminalMappingType, RGMII-StandardTerminalNameType																
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. TXC</td> <td>2. TD[0]</td> <td>3. TD[1]</td> <td>4. TD[2]</td> </tr> <tr> <td>5. TD[3]</td> <td>6. TX_CTL</td> <td>7. RXC</td> <td>8. RD[0]</td> </tr> <tr> <td>9. RD[1]</td> <td>10. RD[2]</td> <td>11. RD[3]</td> <td>12. RX_CTL</td> </tr> <tr> <td>13. MDIO</td> <td>14. MDC</td> <td></td> <td></td> </tr> </table>	1. TXC	2. TD[0]	3. TD[1]	4. TD[2]	5. TD[3]	6. TX_CTL	7. RXC	8. RD[0]	9. RD[1]	10. RD[2]	11. RD[3]	12. RX_CTL	13. MDIO	14. MDC		
1. TXC	2. TD[0]	3. TD[1]	4. TD[2]														
5. TD[3]	6. TX_CTL	7. RXC	8. RD[0]														
9. RD[1]	10. RD[2]	11. RD[3]	12. RX_CTL														
13. MDIO	14. MDC																

For more information about the RGMII Interface, refer to the document released by HP at http://www.hp.com/rnd/pdfs/RGMIIv2_0_final_hp.pdf.

4.7.3.10.27.5. RMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/RMII																								
diagram	<pre> classDiagram class RMII { <<RMII-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<RMII-StandardTerminalNameAssignmentType>> } class RMII_StandardTerminalNameAssignmentType { <<RMII-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<RMII-MandatoryStandardTerminalMappingType>> } class RMII_MandatoryStandardTerminalMappingType { <<RMII-MandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<RMII-MandatoryStandardTerminalNameType>> } class RMII_OptionalStandardTerminalMappingType { <<RMII-OptionalStandardTerminalMappingType>> } class OptionalMapping { <<RMII-OptionalStandardTerminalMappingType>> } class RMII_OptionalStandardTerminalNameType { <<RMII-OptionalStandardTerminalNameType>> } RMII "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" RMII_StandardTerminalNameAssignmentType RMII_StandardTerminalNameAssignmentType "1..>" MandatoryMapping RMII_StandardTerminalNameAssignmentType "1..>" OptionalMapping MandatoryMapping "1..>" StandardTerminalName OptionalMapping "1..>" StandardTerminalName </pre>																								
type	RMIIType, RMII-StandardTerminalNameAssignmentType, RMII-MandatoryStandardTerminalMappingType, RMII-MandatoryStandardTerminalNameType, RMII-OptionalStandardTerminalMappingType, RMII-OptionalStandardTerminalNameType.																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. REF_CLK</td><td>2. TXD[0]</td><td>3. TXD[1]</td><td>4. TX_EN</td></tr> <tr> <td>5. RXD[0]</td><td>6. RXD[1]</td><td>7. CRS_DV</td><td>8. MDIO</td></tr> <tr> <td>9. MDC</td><td></td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th></tr> <tr> <td>1. RX_ER</td><td></td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. REF_CLK	2. TXD[0]	3. TXD[1]	4. TX_EN	5. RXD[0]	6. RXD[1]	7. CRS_DV	8. MDIO	9. MDC				OptionalMapping/StandardTerminalName				1. RX_ER			
MandatoryMapping/StandardTerminalName																									
1. REF_CLK	2. TXD[0]	3. TXD[1]	4. TX_EN																						
5. RXD[0]	6. RXD[1]	7. CRS_DV	8. MDIO																						
9. MDC																									
OptionalMapping/StandardTerminalName																									
1. RX_ER																									

For more information about the RMII Interface, refer to https://en.wikipedia.org/wiki/Media-independent_interface#RMII.

4.7.3.10.27.6. SMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/SMII								
diagram	<pre> classDiagram class SMII { <<SMII-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<SMII-StandardTerminalNameAssignmentType>> } class SMII_StandardTerminalNameAssignmentType { <<SMII-StandardTerminalNameAssignmentType>> } class Mapping { <<SMII-StandardTerminalMappingType>> } class SMII_StandardTerminalMappingType { <<SMII-StandardTerminalMappingType>> } class StandardTerminalName { <<SMII-StandardTerminalNameType>> } SMII "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" SMII_StandardTerminalNameAssignmentType SMII_StandardTerminalNameAssignmentType "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>								
type	SMIIType, SMII-StandardTerminalNameAssignmentType, SMII-StandardTerminalMappingType, SMII-StandardTerminalNameType								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. RX</td><td>2. TX</td><td>3. SYNC</td><td>4. CLOCK</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. RX	2. TX	3. SYNC	4. CLOCK
Mapping/StandardTerminalName									
1. RX	2. TX	3. SYNC	4. CLOCK						

4.7.3.10.27.7. XGMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/XGMII																																																																				
diagram	<pre> classDiagram class XGMII { <<XGMII-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<XGMII-StandardTerminalNameAssignmentType>> } class Mapping { <<XGMII-StandardTerminalMappingType>> } class StandardTerminalName { <<XGMII-StandardTerminalNameType>> } XGMII "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName StandardTerminalName "1..>" constraints constraints <<constraints>> </pre>																																																																				
type	XGMIIType , XGMII-StandardTerminalNameAssignmentType , XGMII-StandardTerminalMappingType , XGMII-StandardTerminalNameType																																																																				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. TX_CLK</td><td>2. TXC</td><td>3. TXD[0]</td><td>4. TXD[1]</td></tr> <tr><td>5. TXD[2]</td><td>6. TXD[3]</td><td>7. TXD[4]</td><td>8. TXD[5]</td></tr> <tr><td>9. TXD[6]</td><td>10. TXD[7]</td><td>11. TXD[8]</td><td>12. TXD[9]</td></tr> <tr><td>13. TXD[10]</td><td>14. TXD[11]</td><td>15. TXD[12]</td><td>16. TXD[13]</td></tr> <tr><td>17. TXD[14]</td><td>18. TXD[15]</td><td>19. TXD[16]</td><td>20. TXD[17]</td></tr> <tr><td>21. TXD[18]</td><td>22. TXD[19]</td><td>23. TXD[20]</td><td>24. TXD[21]</td></tr> <tr><td>25. TXD[22]</td><td>26. TXD[23]</td><td>27. TXD[24]</td><td>28. TXD[25]</td></tr> <tr><td>29. TXD[26]</td><td>30. TXD[27]</td><td>31. TXD[28]</td><td>32. TXD[29]</td></tr> <tr><td>33. TXD[30]</td><td>34. TXD[31]</td><td>35. RX_CLK</td><td>36. RXC</td></tr> <tr><td>37. RXD[0]</td><td>38. RXD[1]</td><td>39. RXD[2]</td><td>40. RXD[3]</td></tr> <tr><td>41. RXD[4]</td><td>42. RXD[5]</td><td>43. RXD[6]</td><td>44. RXD[7]</td></tr> <tr><td>45. RXD[8]</td><td>46. RXD[9]</td><td>47. RXD[10]</td><td>48. RXD[11]</td></tr> <tr><td>49. RXD[12]</td><td>50. RXD[13]</td><td>51. RXD[14]</td><td>52. RXD[15]</td></tr> <tr><td>53. RXD[16]</td><td>54. RXD[17]</td><td>55. RXD[18]</td><td>56. RXD[19]</td></tr> <tr><td>57. RXD[20]</td><td>58. RXD[21]</td><td>59. RXD[22]</td><td>60. RXD[23]</td></tr> <tr><td>61. RXD[24]</td><td>62. RXD[25]</td><td>63. RXD[26]</td><td>64. RXD[27]</td></tr> <tr><td>65. RXD[28]</td><td>66. RXD[29]</td><td>67. RXD[30]</td><td>68. RXD[31]</td></tr> </tbody> </table>	1. TX_CLK	2. TXC	3. TXD[0]	4. TXD[1]	5. TXD[2]	6. TXD[3]	7. TXD[4]	8. TXD[5]	9. TXD[6]	10. TXD[7]	11. TXD[8]	12. TXD[9]	13. TXD[10]	14. TXD[11]	15. TXD[12]	16. TXD[13]	17. TXD[14]	18. TXD[15]	19. TXD[16]	20. TXD[17]	21. TXD[18]	22. TXD[19]	23. TXD[20]	24. TXD[21]	25. TXD[22]	26. TXD[23]	27. TXD[24]	28. TXD[25]	29. TXD[26]	30. TXD[27]	31. TXD[28]	32. TXD[29]	33. TXD[30]	34. TXD[31]	35. RX_CLK	36. RXC	37. RXD[0]	38. RXD[1]	39. RXD[2]	40. RXD[3]	41. RXD[4]	42. RXD[5]	43. RXD[6]	44. RXD[7]	45. RXD[8]	46. RXD[9]	47. RXD[10]	48. RXD[11]	49. RXD[12]	50. RXD[13]	51. RXD[14]	52. RXD[15]	53. RXD[16]	54. RXD[17]	55. RXD[18]	56. RXD[19]	57. RXD[20]	58. RXD[21]	59. RXD[22]	60. RXD[23]	61. RXD[24]	62. RXD[25]	63. RXD[26]	64. RXD[27]	65. RXD[28]	66. RXD[29]	67. RXD[30]	68. RXD[31]
1. TX_CLK	2. TXC	3. TXD[0]	4. TXD[1]																																																																		
5. TXD[2]	6. TXD[3]	7. TXD[4]	8. TXD[5]																																																																		
9. TXD[6]	10. TXD[7]	11. TXD[8]	12. TXD[9]																																																																		
13. TXD[10]	14. TXD[11]	15. TXD[12]	16. TXD[13]																																																																		
17. TXD[14]	18. TXD[15]	19. TXD[16]	20. TXD[17]																																																																		
21. TXD[18]	22. TXD[19]	23. TXD[20]	24. TXD[21]																																																																		
25. TXD[22]	26. TXD[23]	27. TXD[24]	28. TXD[25]																																																																		
29. TXD[26]	30. TXD[27]	31. TXD[28]	32. TXD[29]																																																																		
33. TXD[30]	34. TXD[31]	35. RX_CLK	36. RXC																																																																		
37. RXD[0]	38. RXD[1]	39. RXD[2]	40. RXD[3]																																																																		
41. RXD[4]	42. RXD[5]	43. RXD[6]	44. RXD[7]																																																																		
45. RXD[8]	46. RXD[9]	47. RXD[10]	48. RXD[11]																																																																		
49. RXD[12]	50. RXD[13]	51. RXD[14]	52. RXD[15]																																																																		
53. RXD[16]	54. RXD[17]	55. RXD[18]	56. RXD[19]																																																																		
57. RXD[20]	58. RXD[21]	59. RXD[22]	60. RXD[23]																																																																		
61. RXD[24]	62. RXD[25]	63. RXD[26]	64. RXD[27]																																																																		
65. RXD[28]	66. RXD[29]	67. RXD[30]	68. RXD[31]																																																																		

For more information about the XGMII Interface, refer to the IEEE standard IEEE 802.3ae-2002.

4.7.3.10.27.8. XLGMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/XLGMII																																																																																																								
diagram	<pre> classDiagram class XLGMII { <<XLGMII-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<XLGMII-StandardTerminalNameAssignmentType>> } class Mapping { <<XLGMII-StandardTerminalMappingType>> } class StandardTerminalName { <<XLGMII-StandardTerminalNameType>> } XLGMII "1..oo" -- "1..oo" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..oo" -- "1..oo" Mapping Mapping "1..oo" -- "1..oo" StandardTerminalName </pre>																																																																																																								
type	XLGMIIType, XLGMII-StandardTerminalNameAssignmentType, XLGMII-StandardTerminalMappingType, XLGMII-StandardTerminalNameType																																																																																																								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. TX_CLK</td><td>2. TXC0</td><td>3. TXC1</td><td>4. TXC2</td></tr> <tr><td>5. TXC3</td><td>6. TXC4</td><td>7. TXC5</td><td>8. TXC6</td></tr> <tr><td>9. TXC7</td><td>10. TXD[0]</td><td>11. TXD[1]</td><td>12. TXD[2]</td></tr> <tr><td>13. TXD[3]</td><td>14. TXD[4]</td><td>15. TXD[5]</td><td>16. TXD[6]</td></tr> <tr><td>17. TXD[7]</td><td>18. TXD[8]</td><td>19. TXD[9]</td><td>20. TXD[10]</td></tr> <tr><td>21. TXD[11]</td><td>22. TXD[12]</td><td>23. TXD[13]</td><td>24. TXD[14]</td></tr> <tr><td>25. TXD[15]</td><td>26. TXD[16]</td><td>27. TXD[17]</td><td>28. TXD[18]</td></tr> <tr><td>29. TXD[19]</td><td>30. TXD[20]</td><td>31. TXD[21]</td><td>32. TXD[22]</td></tr> <tr><td>33. TXD[23]</td><td>34. TXD[24]</td><td>35. TXD[25]</td><td>36. TXD[26]</td></tr> <tr><td>37. TXD[27]</td><td>38. TXD[28]</td><td>39. TXD[29]</td><td>40. TXD[30]</td></tr> <tr><td>41. TXD[31]</td><td>42. TXD[32]</td><td>43. TXD[33]</td><td>44. TXD[34]</td></tr> <tr><td>45. TXD[35]</td><td>46. TXD[36]</td><td>47. TXD[37]</td><td>48. TXD[38]</td></tr> <tr><td>49. TXD[39]</td><td>50. TXD[40]</td><td>51. TXD[41]</td><td>52. TXD[42]</td></tr> <tr><td>53. TXD[43]</td><td>54. TXD[44]</td><td>55. TXD[45]</td><td>56. TXD[46]</td></tr> <tr><td>57. TXD[47]</td><td>58. TXD[48]</td><td>59. TXD[49]</td><td>60. TXD[50]</td></tr> <tr><td>61. TXD[51]</td><td>62. TXD[52]</td><td>63. TXD[53]</td><td>64. TXD[54]</td></tr> <tr><td>65. TXD[55]</td><td>66. TXD[56]</td><td>67. TXD[57]</td><td>68. TXD[58]</td></tr> <tr><td>69. TXD[59]</td><td>70. TXD[60]</td><td>71. TXD[61]</td><td>72. TXD[62]</td></tr> <tr><td>73. TXD[63]</td><td>74. RX_CLK</td><td>75. RXC0</td><td>76. RXC1</td></tr> <tr><td>77. RXC2</td><td>78. RXC3</td><td>79. RXC4</td><td>80. RXC5</td></tr> <tr><td>81. RXC6</td><td>82. RXC7</td><td>83. RXD[0]</td><td>84. RXD[1]</td></tr> <tr><td>85. RXD[2]</td><td>86. RXD[3]</td><td>87. RXD[4]</td><td>88. RXD[5]</td></tr> <tr><td>89. RXD[6]</td><td>90. RXD[7]</td><td>91. RXD[8]</td><td>92. RXD[9]</td></tr> <tr><td>93. RXD[10]</td><td>94. RXD[11]</td><td>95. RXD[12]</td><td>96. RXD[13]</td></tr> <tr><td>97. RXD[14]</td><td>98. RXD[15]</td><td>99. RXD[16]</td><td>100. RXD[17]</td></tr> <tr><td>101. RXD[18]</td><td>102. RXD[19]</td><td>103. RXD[20]</td><td>104. RXD[21]</td></tr> </tbody> </table>	1. TX_CLK	2. TXC0	3. TXC1	4. TXC2	5. TXC3	6. TXC4	7. TXC5	8. TXC6	9. TXC7	10. TXD[0]	11. TXD[1]	12. TXD[2]	13. TXD[3]	14. TXD[4]	15. TXD[5]	16. TXD[6]	17. TXD[7]	18. TXD[8]	19. TXD[9]	20. TXD[10]	21. TXD[11]	22. TXD[12]	23. TXD[13]	24. TXD[14]	25. TXD[15]	26. TXD[16]	27. TXD[17]	28. TXD[18]	29. TXD[19]	30. TXD[20]	31. TXD[21]	32. TXD[22]	33. TXD[23]	34. TXD[24]	35. TXD[25]	36. TXD[26]	37. TXD[27]	38. TXD[28]	39. TXD[29]	40. TXD[30]	41. TXD[31]	42. TXD[32]	43. TXD[33]	44. TXD[34]	45. TXD[35]	46. TXD[36]	47. TXD[37]	48. TXD[38]	49. TXD[39]	50. TXD[40]	51. TXD[41]	52. TXD[42]	53. TXD[43]	54. TXD[44]	55. TXD[45]	56. TXD[46]	57. TXD[47]	58. TXD[48]	59. TXD[49]	60. TXD[50]	61. TXD[51]	62. TXD[52]	63. TXD[53]	64. TXD[54]	65. TXD[55]	66. TXD[56]	67. TXD[57]	68. TXD[58]	69. TXD[59]	70. TXD[60]	71. TXD[61]	72. TXD[62]	73. TXD[63]	74. RX_CLK	75. RXC0	76. RXC1	77. RXC2	78. RXC3	79. RXC4	80. RXC5	81. RXC6	82. RXC7	83. RXD[0]	84. RXD[1]	85. RXD[2]	86. RXD[3]	87. RXD[4]	88. RXD[5]	89. RXD[6]	90. RXD[7]	91. RXD[8]	92. RXD[9]	93. RXD[10]	94. RXD[11]	95. RXD[12]	96. RXD[13]	97. RXD[14]	98. RXD[15]	99. RXD[16]	100. RXD[17]	101. RXD[18]	102. RXD[19]	103. RXD[20]	104. RXD[21]
1. TX_CLK	2. TXC0	3. TXC1	4. TXC2																																																																																																						
5. TXC3	6. TXC4	7. TXC5	8. TXC6																																																																																																						
9. TXC7	10. TXD[0]	11. TXD[1]	12. TXD[2]																																																																																																						
13. TXD[3]	14. TXD[4]	15. TXD[5]	16. TXD[6]																																																																																																						
17. TXD[7]	18. TXD[8]	19. TXD[9]	20. TXD[10]																																																																																																						
21. TXD[11]	22. TXD[12]	23. TXD[13]	24. TXD[14]																																																																																																						
25. TXD[15]	26. TXD[16]	27. TXD[17]	28. TXD[18]																																																																																																						
29. TXD[19]	30. TXD[20]	31. TXD[21]	32. TXD[22]																																																																																																						
33. TXD[23]	34. TXD[24]	35. TXD[25]	36. TXD[26]																																																																																																						
37. TXD[27]	38. TXD[28]	39. TXD[29]	40. TXD[30]																																																																																																						
41. TXD[31]	42. TXD[32]	43. TXD[33]	44. TXD[34]																																																																																																						
45. TXD[35]	46. TXD[36]	47. TXD[37]	48. TXD[38]																																																																																																						
49. TXD[39]	50. TXD[40]	51. TXD[41]	52. TXD[42]																																																																																																						
53. TXD[43]	54. TXD[44]	55. TXD[45]	56. TXD[46]																																																																																																						
57. TXD[47]	58. TXD[48]	59. TXD[49]	60. TXD[50]																																																																																																						
61. TXD[51]	62. TXD[52]	63. TXD[53]	64. TXD[54]																																																																																																						
65. TXD[55]	66. TXD[56]	67. TXD[57]	68. TXD[58]																																																																																																						
69. TXD[59]	70. TXD[60]	71. TXD[61]	72. TXD[62]																																																																																																						
73. TXD[63]	74. RX_CLK	75. RXC0	76. RXC1																																																																																																						
77. RXC2	78. RXC3	79. RXC4	80. RXC5																																																																																																						
81. RXC6	82. RXC7	83. RXD[0]	84. RXD[1]																																																																																																						
85. RXD[2]	86. RXD[3]	87. RXD[4]	88. RXD[5]																																																																																																						
89. RXD[6]	90. RXD[7]	91. RXD[8]	92. RXD[9]																																																																																																						
93. RXD[10]	94. RXD[11]	95. RXD[12]	96. RXD[13]																																																																																																						
97. RXD[14]	98. RXD[15]	99. RXD[16]	100. RXD[17]																																																																																																						
101. RXD[18]	102. RXD[19]	103. RXD[20]	104. RXD[21]																																																																																																						

4.7.3.10.27.8 XLGMII – Mandatory Mapping (cont'd)

105. RXD[22]	106. RXD[23]	107. RXD[24]	108. RXD[25]
109. RXD[26]	110. RXD[27]	111. RXD[28]	112. RXD[29]
113. RXD[30]	114. RXD[31]	115. RXD[32]	116. RXD[33]
117. RXD[34]	118. RXD[35]	119. RXD[36]	120. RXD[37]
121. RXD[38]	122. RXD[39]	123. RXD[40]	124. RXD[41]
125. RXD[42]	126. RXD[43]	127. RXD[44]	128. RXD[45]
129. RXD[46]	130. RXD[47]	131. RXD[48]	132. RXD[49]
133. RXD[50]	134. RXD[51]	135. RXD[52]	136. RXD[53]
137. RXD[54]	138. RXD[55]	139. RXD[56]	140. RXD[57]
141. RXD[58]	142. RXD[59]	143. RXD[60]	144. RXD[61]
145. RXD[62]	146. RXD[63]		

For more information about the XLGMII Interface, refer to the IEEE standard IEEE std 802.3.

4.7.3.10.28. OIF-CEI-04.0 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/OIF-CEI-04.0																
diagram	<pre> classDiagram class OIF_CEI_04_0 { <<OIF-CEI-04.0>> <<OIF-CEI-04.0-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<OIF-CEI-04.0-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<OIF-CEI-04.0-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<OIF-CEI-04.0-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<OIF-CEI-04.0-MandatoryStandardTerminalNameType>> } OIF_CEI_04_0 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping StandardTerminalNameAssignment "1..>" OptionalMapping MandatoryMapping "1..>" StandardTerminalName OptionalMapping "1..>" StandardTerminalName </pre>																
type	OIF-CEI-04.0-InterfaceFunctionType, OIF-CEI-04.0-StandardTerminalNameAssignmentType, OIF-CEI-04.0-StandardTerminalMappingType, OIF-CEI-04.0-StandardTerminalNameType.																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. TxData0P</td><td>2. TxData0N</td><td>3. RxData0P</td><td>4. RxData0N</td> </tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>5. RefClk</td><td></td><td></td><td></td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. TxData0P	2. TxData0N	3. RxData0P	4. RxData0N	OptionalMapping/StandardTerminalName				5. RefClk			
MandatoryMapping/StandardTerminalName																	
1. TxData0P	2. TxData0N	3. RxData0P	4. RxData0N														
OptionalMapping/StandardTerminalName																	
5. RefClk																	

For more information about the XGMII Interface, refer to the OIF standard OIF-CEI-04.0.

4.7.3.10.29. PCIe Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe
diagram	<pre> graph LR PCIe[PCIe type: PCIe-InterfaceFunctionType] --- PCIeIF[PCIe-InterfaceFunctionType] PCIeIF --- PCIe_x1[PCIe-x1 type: PCIe-x1-InterfaceFunctionType] PCIeIF --- PCIe_x2[PCIe-x2 type: PCIe-x2-InterfaceFunctionType] PCIeIF --- PCIe_x4[PCIe-x4 type: PCIe-x4-InterfaceFunctionType] PCIeIF --- PCIe_x8[PCIe-x8 type: PCIe-x8-InterfaceFunctionType] PCIeIF --- PCIe_x16[PCIe-x16 type: PCIe-x16-InterfaceFunctionType] PCIeIF --- PCIe_x32[PCIe-x32 type: PCIe-x32-InterfaceFunctionType] PCIeIF --- SFF8639[SFF-8639Connector type: SFF-8639ConnectorInterfaceFunctionType] PCIeIF --- ATXPWR[ATX-PowerConnector150W type: ATX-PowerConnector150W-InterfaceFunctionType] </pre>
type	PCIe-InterfaceFunctionType , PCIe-x1-InterfaceFunctionType , PCIe-x2-InterfaceFunctionType , PCIe-x4-InterfaceFunctionType , PCIe-x8-InterfaceFunctionType , PCIe-x16-InterfaceFunctionType , PCIe-x32-InterfaceFunctionType , SFF-8639ConnectorInterfaceFunctionType , ATX-PowerConnector150W-InterfaceFunctionType , AuxiliaryPowerConnector2x4-InterfaceFunctionType .

For more information about the PCIe Interfaces, refer to the PCI-SIG standard PCI Express Card Electromechanical Specification Rev 2.0.

4.7.3.10.29.1. PCIe-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x1																								
diagram	<pre> classDiagram class PCIe_x1 { <<PCIe-x1-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<PCIe-x1-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<PCIe-x1-MandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<PCIe-x1-MandatoryStandardTerminalNameType>> } class OptionalMapping { <<PCIe-x1-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<PCIe-x1-OptionalStandardTerminalNameType>> } PCIe_x1 --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> MandatoryMapping StandardTerminalNameAssignment --> OptionalMapping MandatoryMapping --> StandardTerminalName OptionalMapping --> StandardTerminalName </pre>																								
type	PCIe-x1-InterfaceFunctionType, PCIe-x1-StandardTerminalNameAssignmentType, PCIe-x1-MandatoryStandardTerminalMappingType, PCIe-x1-MandatoryStandardTerminalNameType, PCIe-x1-OptionalStandardTerminalMappingType, PCIe-x1-OptionalStandardTerminalNameType.																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. REFCLK+</td> <td>2. REFCLK-</td> <td>3. PETp0</td> <td>4. PETn0</td> </tr> <tr> <td>5. PERp0</td> <td>6. PERn0</td> <td></td> <td></td> </tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. WAKE#</td> <td>2. CLKREQ#</td> <td>3. PERST#</td> <td>4. PRSNT1#</td> </tr> <tr> <td>5. PRSNT2#</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. REFCLK+	2. REFCLK-	3. PETp0	4. PETn0	5. PERp0	6. PERn0			OptionalMapping/StandardTerminalName				1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#	5. PRSNT2#			
MandatoryMapping/StandardTerminalName																									
1. REFCLK+	2. REFCLK-	3. PETp0	4. PETn0																						
5. PERp0	6. PERn0																								
OptionalMapping/StandardTerminalName																									
1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#																						
5. PRSNT2#																									

4.7.3.10.29.2. PCIe-x2

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x2																												
diagram	<pre> classDiagram class PCIe_x2_InterfaceFunctionType class StandardTerminalNameAssignment class MandatoryMapping class OptionalMapping class StandardTerminalName PCIe_x2_InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" MandatoryMapping StandardTerminalNameAssignment "*" OptionalMapping MandatoryMapping "1..>" StandardTerminalName OptionalMapping "0..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for PCIe-x2. It starts with a main class, PCIe-x2-InterfaceFunctionType, which has a multiplicity of 1..> associated with StandardTerminalNameAssignment. StandardTerminalNameAssignment has a multiplicity of * associated with both MandatoryMapping and OptionalMapping. MandatoryMapping and OptionalMapping both have a multiplicity of 1..> associated with StandardTerminalName. There is also a constraint labeled 'constraints'.</p>																												
type	PCIe-x2-InterfaceFunctionType, PCIe-x2-StandardTerminalNameAssignmentType, PCIe-x2-MandatoryStandardTerminalMappingType, PCIe-x2-MandatoryStandardTerminalNameType, PCIe-x2-OptionalStandardTerminalMappingType, PCIe-x2-OptionalStandardTerminalNameType.																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. REFCLK+</td> <td>2. REFCLK-</td> <td>3. PETp0</td> <td>4. PETp1</td> </tr> <tr> <td>5. PETn0</td> <td>6. PETn1</td> <td>7. PERp0</td> <td>8. PERp1</td> </tr> <tr> <td>9. PERn0</td> <td>10. PERn1</td> <td></td> <td></td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. WAKE#</td> <td>2. CLKREQ#</td> <td>3. PERST#</td> <td>4. PRSNT1#</td> </tr> <tr> <td>5. PRSNT2#</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1	5. PETn0	6. PETn1	7. PERp0	8. PERp1	9. PERn0	10. PERn1			OptionalMapping/StandardTerminalName				1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#	5. PRSNT2#			
MandatoryMapping/StandardTerminalName																													
1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1																										
5. PETn0	6. PETn1	7. PERp0	8. PERp1																										
9. PERn0	10. PERn1																												
OptionalMapping/StandardTerminalName																													
1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#																										
5. PRSNT2#																													

4.7.3.10.29.3. PCIe-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x4																																				
diagram	<pre> classDiagram class PCIe_x4 { <<PCIe-x4-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<PCIe-x4-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<PCIe-x4-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<PCIe-x4-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<PCIe-x4-MandatoryStandardTerminalNameType>> } class StandardTerminalName { <<PCIe-x4-OptionalStandardTerminalNameType>> } PCIe_x4 "1..oo" --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> MandatoryMapping StandardTerminalNameAssignment --> OptionalMapping MandatoryMapping "1..18" --> StandardTerminalName OptionalMapping "0..5" --> StandardTerminalName Note over MandatoryMapping, OptionalMapping: constraints </pre>																																				
type	PCIe-x4-InterfaceFunctionType, PCIe-x4-StandardTerminalNameAssignmentType, PCIe-x4-MandatoryStandardTerminalMappingType, PCIe-x4-MandatoryStandardTerminalNameType, PCIe-x4-OptionalStandardTerminalMappingType, PCIe-x4-OptionalStandardTerminalNameType.																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. REFCLK+</td><td>2. REFCLK-</td><td>3. PETp0</td><td>4. PETp1</td></tr> <tr><td>5. PETp2</td><td>6. PETp3</td><td>7. PETn0</td><td>8. PETn1</td></tr> <tr><td>9. PETn2</td><td>10. PETn3</td><td>11. PERp0</td><td>12. PERp1</td></tr> <tr><td>13. PERp2</td><td>14. PERp3</td><td>15. PERn0</td><td>16. PERn1</td></tr> <tr><td>17. PERn2</td><td>18. PERn3</td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. WAKE#</td><td>2. CLKREQ#</td><td>3. PERST#</td><td>4. PRSNT1#</td></tr> <tr><td>5. PRSNT2#</td><td></td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1	5. PETp2	6. PETp3	7. PETn0	8. PETn1	9. PETn2	10. PETn3	11. PERp0	12. PERp1	13. PERp2	14. PERp3	15. PERn0	16. PERn1	17. PERn2	18. PERn3			OptionalMapping/StandardTerminalName				1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#	5. PRSNT2#			
MandatoryMapping/StandardTerminalName																																					
1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1																																		
5. PETp2	6. PETp3	7. PETn0	8. PETn1																																		
9. PETn2	10. PETn3	11. PERp0	12. PERp1																																		
13. PERp2	14. PERp3	15. PERn0	16. PERn1																																		
17. PERn2	18. PERn3																																				
OptionalMapping/StandardTerminalName																																					
1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#																																		
5. PRSNT2#																																					

4.7.3.10.29.4. PCIe-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x8																																																				
diagram	<pre> classDiagram class PCIe_x8 { <<PCIe-x8-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<PCIe-x8-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<PCIe-x8-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<PCIe-x8-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<PCIe-x8-MandatoryStandardTerminalNameType>> } class StandardTerminalName { <<PCIe-x8-OptionalStandardTerminalNameType>> } PCIe_x8 "1..20" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "*" --> MandatoryMapping StandardTerminalNameAssignment "*" --> OptionalMapping MandatoryMapping "34" --> StandardTerminalName OptionalMapping "0..5" --> StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for PCIe-x8. It features a main class, PCIe-x8, which has a multiplicity of 1..20 instances associated with StandardTerminalNameAssignment. This association is marked with an asterisk (*), indicating many-to-many. From each StandardTerminalNameAssignment, there are two outgoing associations: one to MandatoryMapping (multiplicity *) and another to OptionalMapping (multiplicity *). Finally, both MandatoryMapping and OptionalMapping have associations to StandardTerminalName, with specific multiplicities: 34 for MandatoryMapping and 0..5 for OptionalMapping.</p>																																																				
type	PCIe-x8-InterfaceFunctionType, PCIe-x8-StandardTerminalNameAssignmentType, PCIe-x8-MandatoryStandardTerminalMappingType, PCIe-x8-MandatoryStandardTerminalNameType, PCIe-x8-OptionalStandardTerminalMappingType, PCIe-x8-OptionalStandardTerminalNameType.																																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. REFCLK+</td><td>2. REFCLK-</td><td>3. PETp0</td><td>4. PETp1</td></tr> <tr><td>5. PETp2</td><td>6. PETp3</td><td>7. PETp4</td><td>8. PETp5</td></tr> <tr><td>9. PETp6</td><td>10. PETp7</td><td>11. PETn0</td><td>12. PETn1</td></tr> <tr><td>13. PETn2</td><td>14. PETn3</td><td>15. PETn4</td><td>16. PETn5</td></tr> <tr><td>17. PETn6</td><td>18. PETn7</td><td>19. PERp0</td><td>20. PERp1</td></tr> <tr><td>21. PERp2</td><td>22. PERp3</td><td>23. PERp4</td><td>24. PERp5</td></tr> <tr><td>25. PERp6</td><td>26. PERp7</td><td>27. PERn0</td><td>28. PERn1</td></tr> <tr><td>29. PERn2</td><td>30. PERn3</td><td>31. PERn4</td><td>32. PERn5</td></tr> <tr><td>33. PERn6</td><td>34. PERn7</td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. WAKE#</td><td>2. CLKREQ#</td><td>3. PERST#</td><td>4. PRSNT1#</td></tr> <tr> <td>5. PRSNT2#</td><td></td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1	5. PETp2	6. PETp3	7. PETp4	8. PETp5	9. PETp6	10. PETp7	11. PETn0	12. PETn1	13. PETn2	14. PETn3	15. PETn4	16. PETn5	17. PETn6	18. PETn7	19. PERp0	20. PERp1	21. PERp2	22. PERp3	23. PERp4	24. PERp5	25. PERp6	26. PERp7	27. PERn0	28. PERn1	29. PERn2	30. PERn3	31. PERn4	32. PERn5	33. PERn6	34. PERn7			OptionalMapping/StandardTerminalName				1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#	5. PRSNT2#			
MandatoryMapping/StandardTerminalName																																																					
1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1																																																		
5. PETp2	6. PETp3	7. PETp4	8. PETp5																																																		
9. PETp6	10. PETp7	11. PETn0	12. PETn1																																																		
13. PETn2	14. PETn3	15. PETn4	16. PETn5																																																		
17. PETn6	18. PETn7	19. PERp0	20. PERp1																																																		
21. PERp2	22. PERp3	23. PERp4	24. PERp5																																																		
25. PERp6	26. PERp7	27. PERn0	28. PERn1																																																		
29. PERn2	30. PERn3	31. PERn4	32. PERn5																																																		
33. PERn6	34. PERn7																																																				
OptionalMapping/StandardTerminalName																																																					
1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#																																																		
5. PRSNT2#																																																					

4.7.3.10.29.5. PCIe-x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x16																																																																																				
diagram																																																																																					
type	PCIe-x16-InterfaceFunctionType , PCIe-x16-StandardTerminalNameAssignmentType , PCIe-x16-MandatoryStandardTerminalMappingType , PCIe-x16-MandatoryStandardTerminalNameType , PCIe-x16-OptionalStandardTerminalMappingType , PCIe-x16-OptionalStandardTerminalNameType .																																																																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. REFCLK+</td><td>2. REFCLK-</td><td>3. PETp0</td><td>4. PETp1</td></tr> <tr><td>5. PETp2</td><td>6. PETp3</td><td>7. PETp4</td><td>8. PETp5</td></tr> <tr><td>9. PETp6</td><td>10. PETp7</td><td>11. PETp8</td><td>12. PETp9</td></tr> <tr><td>13. PETp10</td><td>14. PETp11</td><td>15. PETp12</td><td>16. PETp13</td></tr> <tr><td>17. PETp14</td><td>18. PETp15</td><td>19. PETn0</td><td>20. PETn1</td></tr> <tr><td>21. PETn2</td><td>22. PETn3</td><td>23. PETn4</td><td>24. PETn5</td></tr> <tr><td>25. PETn6</td><td>26. PETn7</td><td>27. PETn8</td><td>28. PETn9</td></tr> <tr><td>29. PETn10</td><td>30. PETn11</td><td>31. PETn12</td><td>32. PETn13</td></tr> <tr><td>33. PETn14</td><td>34. PETn15</td><td>35. PERp0</td><td>36. PERp1</td></tr> <tr><td>37. PERp2</td><td>38. PERp3</td><td>39. PERp4</td><td>40. PERp5</td></tr> <tr><td>41. PERp6</td><td>42. PERp7</td><td>43. PERp8</td><td>44. PERp9</td></tr> <tr><td>45. PERp10</td><td>46. PERp11</td><td>47. PERp12</td><td>48. PERp13</td></tr> <tr><td>49. PERp14</td><td>50. PERp15</td><td>51. PERn0</td><td>52. PERn1</td></tr> <tr><td>53. PERn2</td><td>54. PERn3</td><td>55. PERn4</td><td>56. PERn5</td></tr> <tr><td>57. PERn6</td><td>58. PERn7</td><td>59. PERn8</td><td>60. PERn9</td></tr> <tr><td>61. PERn10</td><td>62. PERn11</td><td>63. PERn12</td><td>64. PERn13</td></tr> <tr><td>65. PERn14</td><td>66. PERn15</td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. WAKE#</td><td>2. CLKREQ#</td><td>3. PERST#</td><td>4. PRSNT1#</td></tr> <tr><td>5. PRSNT2#</td><td></td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1	5. PETp2	6. PETp3	7. PETp4	8. PETp5	9. PETp6	10. PETp7	11. PETp8	12. PETp9	13. PETp10	14. PETp11	15. PETp12	16. PETp13	17. PETp14	18. PETp15	19. PETn0	20. PETn1	21. PETn2	22. PETn3	23. PETn4	24. PETn5	25. PETn6	26. PETn7	27. PETn8	28. PETn9	29. PETn10	30. PETn11	31. PETn12	32. PETn13	33. PETn14	34. PETn15	35. PERp0	36. PERp1	37. PERp2	38. PERp3	39. PERp4	40. PERp5	41. PERp6	42. PERp7	43. PERp8	44. PERp9	45. PERp10	46. PERp11	47. PERp12	48. PERp13	49. PERp14	50. PERp15	51. PERn0	52. PERn1	53. PERn2	54. PERn3	55. PERn4	56. PERn5	57. PERn6	58. PERn7	59. PERn8	60. PERn9	61. PERn10	62. PERn11	63. PERn12	64. PERn13	65. PERn14	66. PERn15			OptionalMapping/StandardTerminalName				1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#	5. PRSNT2#			
MandatoryMapping/StandardTerminalName																																																																																					
1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1																																																																																		
5. PETp2	6. PETp3	7. PETp4	8. PETp5																																																																																		
9. PETp6	10. PETp7	11. PETp8	12. PETp9																																																																																		
13. PETp10	14. PETp11	15. PETp12	16. PETp13																																																																																		
17. PETp14	18. PETp15	19. PETn0	20. PETn1																																																																																		
21. PETn2	22. PETn3	23. PETn4	24. PETn5																																																																																		
25. PETn6	26. PETn7	27. PETn8	28. PETn9																																																																																		
29. PETn10	30. PETn11	31. PETn12	32. PETn13																																																																																		
33. PETn14	34. PETn15	35. PERp0	36. PERp1																																																																																		
37. PERp2	38. PERp3	39. PERp4	40. PERp5																																																																																		
41. PERp6	42. PERp7	43. PERp8	44. PERp9																																																																																		
45. PERp10	46. PERp11	47. PERp12	48. PERp13																																																																																		
49. PERp14	50. PERp15	51. PERn0	52. PERn1																																																																																		
53. PERn2	54. PERn3	55. PERn4	56. PERn5																																																																																		
57. PERn6	58. PERn7	59. PERn8	60. PERn9																																																																																		
61. PERn10	62. PERn11	63. PERn12	64. PERn13																																																																																		
65. PERn14	66. PERn15																																																																																				
OptionalMapping/StandardTerminalName																																																																																					
1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#																																																																																		
5. PRSNT2#																																																																																					

4.7.3.10.29.6. PCIe-x32

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCI3/PCIe-x32																																																																																																
diagram																																																																																																	
type	PCIe-x32-InterfaceFunctionType , PCIe-x32-StandardTerminalNameAssignmentType , PCIe-x32-MandatoryStandardTerminalMappingType , PCIe-x32-MandatoryStandardTerminalNameType , PCIe-x32-OptionalStandardTerminalMappingType , PCIe-x32-OptionalStandardTerminalNameType .																																																																																																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. REFCLK+</td><td>2. REFCLK-</td><td>3. PETp1</td><td>4. PETp1</td></tr> <tr><td>5. PETp2</td><td>6. PETp3</td><td>7. PETp4</td><td>8. PETp5</td></tr> <tr><td>9. PETp6</td><td>10. PETp7</td><td>11. PETp8</td><td>12. PETp9</td></tr> <tr><td>13. PETp10</td><td>14. PETp11</td><td>15. PETp12</td><td>16. PETp13</td></tr> <tr><td>17. PETp14</td><td>18. PETp15</td><td>19. PETp16</td><td>20. PETp17</td></tr> <tr><td>21. PETp18</td><td>22. PETp19</td><td>23. PETp20</td><td>24. PETp21</td></tr> <tr><td>25. PETp22</td><td>26. PETp23</td><td>27. PETp24</td><td>28. PETp25</td></tr> <tr><td>29. PETp26</td><td>30. PETp27</td><td>31. PETp28</td><td>32. PETp29</td></tr> <tr><td>33. PETp30</td><td>34. PETp31</td><td>35. PETn0</td><td>36. PETn1</td></tr> <tr><td>37. PETn2</td><td>38. PETn3</td><td>39. PETn4</td><td>40. PETn5</td></tr> <tr><td>41. PETn6</td><td>42. PETn7</td><td>43. PETn8</td><td>44. PETn9</td></tr> <tr><td>45. PETn10</td><td>46. PETn11</td><td>47. PETn12</td><td>48. PETn13</td></tr> <tr><td>49. PETn14</td><td>50. PETn15</td><td>51. PETn16</td><td>52. PETn17</td></tr> <tr><td>53. PETn18</td><td>54. PETn19</td><td>55. PETn20</td><td>56. PETn21</td></tr> <tr><td>57. PETn22</td><td>58. PETn23</td><td>59. PETn24</td><td>60. PETn25</td></tr> <tr><td>61. PETn26</td><td>62. PETn27</td><td>63. PETn28</td><td>64. PETn29</td></tr> <tr><td>65. PETn30</td><td>66. PETn31</td><td>67. PERp0</td><td>68. PERp1</td></tr> <tr><td>69. PERp2</td><td>70. PERp3</td><td>71. PERp4</td><td>72. PERp5</td></tr> <tr><td>73. PERp6</td><td>74. PERp7</td><td>75. PERp8</td><td>76. PERp9</td></tr> <tr><td>77. PERp10</td><td>78. PERp11</td><td>79. PERp12</td><td>80. PERp13</td></tr> <tr><td>81. PERp14</td><td>82. PERp15</td><td>83. PERp16</td><td>84. PERp17</td></tr> <tr><td>85. PERp18</td><td>86. PERp19</td><td>87. PERp20</td><td>88. PERp21</td></tr> <tr><td>89. PERp22</td><td>90. PERp23</td><td>91. PERp24</td><td>92. PERp25</td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. REFCLK+	2. REFCLK-	3. PETp1	4. PETp1	5. PETp2	6. PETp3	7. PETp4	8. PETp5	9. PETp6	10. PETp7	11. PETp8	12. PETp9	13. PETp10	14. PETp11	15. PETp12	16. PETp13	17. PETp14	18. PETp15	19. PETp16	20. PETp17	21. PETp18	22. PETp19	23. PETp20	24. PETp21	25. PETp22	26. PETp23	27. PETp24	28. PETp25	29. PETp26	30. PETp27	31. PETp28	32. PETp29	33. PETp30	34. PETp31	35. PETn0	36. PETn1	37. PETn2	38. PETn3	39. PETn4	40. PETn5	41. PETn6	42. PETn7	43. PETn8	44. PETn9	45. PETn10	46. PETn11	47. PETn12	48. PETn13	49. PETn14	50. PETn15	51. PETn16	52. PETn17	53. PETn18	54. PETn19	55. PETn20	56. PETn21	57. PETn22	58. PETn23	59. PETn24	60. PETn25	61. PETn26	62. PETn27	63. PETn28	64. PETn29	65. PETn30	66. PETn31	67. PERp0	68. PERp1	69. PERp2	70. PERp3	71. PERp4	72. PERp5	73. PERp6	74. PERp7	75. PERp8	76. PERp9	77. PERp10	78. PERp11	79. PERp12	80. PERp13	81. PERp14	82. PERp15	83. PERp16	84. PERp17	85. PERp18	86. PERp19	87. PERp20	88. PERp21	89. PERp22	90. PERp23	91. PERp24	92. PERp25
MandatoryMapping/StandardTerminalName																																																																																																	
1. REFCLK+	2. REFCLK-	3. PETp1	4. PETp1																																																																																														
5. PETp2	6. PETp3	7. PETp4	8. PETp5																																																																																														
9. PETp6	10. PETp7	11. PETp8	12. PETp9																																																																																														
13. PETp10	14. PETp11	15. PETp12	16. PETp13																																																																																														
17. PETp14	18. PETp15	19. PETp16	20. PETp17																																																																																														
21. PETp18	22. PETp19	23. PETp20	24. PETp21																																																																																														
25. PETp22	26. PETp23	27. PETp24	28. PETp25																																																																																														
29. PETp26	30. PETp27	31. PETp28	32. PETp29																																																																																														
33. PETp30	34. PETp31	35. PETn0	36. PETn1																																																																																														
37. PETn2	38. PETn3	39. PETn4	40. PETn5																																																																																														
41. PETn6	42. PETn7	43. PETn8	44. PETn9																																																																																														
45. PETn10	46. PETn11	47. PETn12	48. PETn13																																																																																														
49. PETn14	50. PETn15	51. PETn16	52. PETn17																																																																																														
53. PETn18	54. PETn19	55. PETn20	56. PETn21																																																																																														
57. PETn22	58. PETn23	59. PETn24	60. PETn25																																																																																														
61. PETn26	62. PETn27	63. PETn28	64. PETn29																																																																																														
65. PETn30	66. PETn31	67. PERp0	68. PERp1																																																																																														
69. PERp2	70. PERp3	71. PERp4	72. PERp5																																																																																														
73. PERp6	74. PERp7	75. PERp8	76. PERp9																																																																																														
77. PERp10	78. PERp11	79. PERp12	80. PERp13																																																																																														
81. PERp14	82. PERp15	83. PERp16	84. PERp17																																																																																														
85. PERp18	86. PERp19	87. PERp20	88. PERp21																																																																																														
89. PERp22	90. PERp23	91. PERp24	92. PERp25																																																																																														

4.7.3.10.29.6 PCIe-x32 – Mandatory Mapping (cont'd)

93. PERp26	94. PERp27	95. PERp28	96. PERp29
97. PERp30	98. PERp31	99. PERn0	100. PERn1
101. PERn2	102. PERn3	103. PERn4	104. PERn5
105. PERn6	106. PERn7	107. PERn8	108. PERn9
109. PERn10	110. PERn11	111. PERn12	112. PERn13
113. PERn14	114. PERn15	115. PERn16	116. PERn17
117. PERn18	118. PERn19	119. PERn20	120. PERn21
121. PERn22	122. PERn23	123. PERn24	124. PERn25
125. PERn26	126. PERn27	127. PERn28	128. PERn29
129. PERn30	130. PERn31		
OptionalMapping/StandardTerminalName			
1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#
5. PRSNT2#			

4.7.3.10.29.7. SFF-8639 Connector

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/PCIe/SFF-8639Connector																																																
diagram	<pre> classDiagram class SFF_8639Connector { <<SFF-8639ConnectorInterfaceFunctionType>> } class StandardTerminalNameAssignment { <<SFF-8639ConnectorStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<SFF-8639ConnectorMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<SFF-8639ConnectorOptionalStandardTerminalMappingType>> } SFF_8639Connector "1..x" -- "1..x" StandardTerminalNameAssignment StandardTerminalNameAssignment "35" -- "35" MandatoryMapping StandardTerminalNameAssignment "0..2" -- "0..2" OptionalMapping </pre>																																																
type	SFF-8639ConnectorInterfaceFunctionType , SFF-8639ConnectorStandardTerminalNameAssignmentType , SFF-8639ConnectorMandatoryStandardTerminalMappingType , SFF-8639ConnectorMandatoryStandardTerminalNameType , SFF-8639ConnectorOptionalStandardTerminalMappingType , SFF-8639ConnectorOptionalStandardTerminalNameType .																																																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. WAKE#</td><td>2. PWRDIS</td><td>3. IfDet#</td><td>4. Ground</td></tr> <tr><td>5. PRSNT#</td><td>6. ACTIVITY#</td><td>7. +12 V Precharge</td><td>8. +12 V</td></tr> <tr><td>9. HPT0</td><td>10. PETp1</td><td>11. PETn1</td><td>12. PERn1</td></tr> <tr><td>13. PERP1</td><td>14. PETp2</td><td>15. PETn2</td><td>16. PERn2</td></tr> <tr><td>17. PERp2</td><td>18. REFCLKB+</td><td>19. REFCLKB-</td><td>20. +3.3 Vaux</td></tr> <tr><td>21. PERST#</td><td>22. REFCLK+</td><td>23. REFCLK-</td><td>24. PETp0</td></tr> <tr><td>25. PETn0</td><td>26. PERn0</td><td>27. PERp0</td><td>28. HPT1</td></tr> <tr><td>29. PETp3</td><td>30. PETn3</td><td>31. PERn3</td><td>32. PERp3</td></tr> <tr><td>33. SMBCLK</td><td>34. SMBDAT</td><td>35. DUALPORTEN#</td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. CLKREQ#</td><td>2. PERSTB#</td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. WAKE#	2. PWRDIS	3. IfDet#	4. Ground	5. PRSNT#	6. ACTIVITY#	7. +12 V Precharge	8. +12 V	9. HPT0	10. PETp1	11. PETn1	12. PERn1	13. PERP1	14. PETp2	15. PETn2	16. PERn2	17. PERp2	18. REFCLKB+	19. REFCLKB-	20. +3.3 Vaux	21. PERST#	22. REFCLK+	23. REFCLK-	24. PETp0	25. PETn0	26. PERn0	27. PERp0	28. HPT1	29. PETp3	30. PETn3	31. PERn3	32. PERp3	33. SMBCLK	34. SMBDAT	35. DUALPORTEN#		OptionalMapping/StandardTerminalName				1. CLKREQ#	2. PERSTB#		
MandatoryMapping/StandardTerminalName																																																	
1. WAKE#	2. PWRDIS	3. IfDet#	4. Ground																																														
5. PRSNT#	6. ACTIVITY#	7. +12 V Precharge	8. +12 V																																														
9. HPT0	10. PETp1	11. PETn1	12. PERn1																																														
13. PERP1	14. PETp2	15. PETn2	16. PERn2																																														
17. PERp2	18. REFCLKB+	19. REFCLKB-	20. +3.3 Vaux																																														
21. PERST#	22. REFCLK+	23. REFCLK-	24. PETp0																																														
25. PETn0	26. PERn0	27. PERp0	28. HPT1																																														
29. PETp3	30. PETn3	31. PERn3	32. PERp3																																														
33. SMBCLK	34. SMBDAT	35. DUALPORTEN#																																															
OptionalMapping/StandardTerminalName																																																	
1. CLKREQ#	2. PERSTB#																																																

For more information about the SFF-8639 Connector Interface, refer to the PCI Express standard PCI Express SFF-8639 Module Revision 4, Version 1.0

4.7.3.10.29.8. ATX - Power Connector 150W

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/PCIe/ATX-PowerConnector150W							
diagram	<pre> classDiagram class ATX_PowerConnector150W { <<ATX-PowerConnector150W-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<ATX_PowerConnector150W-StandardTerminalNameAssignmentType>> } class Mapping { <<ATX_PowerConnector150W-StandardTerminalMappingType>> } class StandardTerminalName { <<ATX_PowerConnector150W-StandardTerminalNameType>> } ATX_PowerConnector150W "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>							
type	ATX-PowerConnector150W-InterfaceFunctionType, ATX-PowerConnector150W-StandardTerminalNameAssignmentType, ATX-PowerConnector150W-StandardTerminalMappingType, ATX-PowerConnector150W-StandardTerminalNameType.							
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. +12V</td> <td>2. Ground</td> <td>3. Sense</td> <td></td> </tr> </table>				1. +12V	2. Ground	3. Sense	
1. +12V	2. Ground	3. Sense						

For more information about the ATX-Power Connector 150W Interface, refer to the PCI Express standard Specification PCI Express x16 Graphics 150W-ATX Revision 1.0

4.7.3.10.29.9. AuxiliaryPowerConnector2x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/PCIe/AuxiliaryPowerConnector2x4							
diagram	<pre> classDiagram class AuxiliaryPowerConnector2x4 { <<AuxiliaryPowerConnector2x4-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<AuxiliaryPowerConnector2x4-StandardTerminalNameAssignmentType>> } class Mapping { <<AuxiliaryPowerConnector2x4-StandardTerminalMappingType>> } class StandardTerminalName { <<AuxiliaryPowerConnector2x4-StandardTerminalNameType>> } AuxiliaryPowerConnector2x4 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>							
type	AuxiliaryPowerConnector2x4-InterfaceFunctionType, AuxiliaryPowerConnector2x4-StandardTerminalNameAssignmentType, AuxiliaryPowerConnector2x4-StandardTerminalMappingType, AuxiliaryPowerConnector2x4-StandardTerminalNameType.							
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. +12V</td> <td>2. Sense0</td> <td>3. Sense1</td> <td>4. Ground</td> </tr> </table>				1. +12V	2. Sense0	3. Sense1	4. Ground
1. +12V	2. Sense0	3. Sense1	4. Ground					

For more information about the Auxiliary Power Connector 2x4 Interface, refer to the PCI Express standard Specification PCI Express 225 W/ 300 W High Power Card Electromechanical Specification Revision 1.0.

4.7.3.10.30. Cabling PCIe Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCle
diagram	<pre> classDiagram class CablingPCIe { <<CablingPCIe-InterfaceFunctionType>> } class CablingPCIe_x1 { <<CablingPCIe-x1-InterfaceFunctionType>> } class CablingPCIe_x4 { <<CablingPCIe-x4-InterfaceFunctionType>> } class CablingPCIe_x8 { <<CablingPCIe-x8-InterfaceFunctionType>> } class CablingPCIe_x16 { <<CablingPCIe-x16-InterfaceFunctionType>> } CablingPCIe < -- CablingPCIe_x1 CablingPCIe < -- CablingPCIe_x4 CablingPCIe < -- CablingPCIe_x8 CablingPCIe < -- CablingPCIe_x16 </pre>
type	CablingPCIe-InterfaceFunctionType , CablingPCIe-x1-InterfaceFunctionType , CablingPCIe-x2-InterfaceFunctionType , CablingPCIe-x4-InterfaceFunctionType , CablingPCIe-x8-InterfaceFunctionType , CablingPCIe-x16-InterfaceFunctionType , CablingPCIe-x32-InterfaceFunctionType .

For more information about the PCIe Interfaces, refer to the PCI-SIG standard PCI Express Card Electromechanical Specification Revision 2.0.

4.7.3.10.30.1. CablingPCIe-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCle/CablingPCIe-x1												
diagram	<pre> classDiagram class CablingPCIe_x1 { <<CablingPCIe-x1-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<CablingPCIe-x1-StandardTerminalNameAssignmentType>> } class Mapping { <<CablingPCIe-x1-StandardTerminalMappingType>> } class StandardTerminalName { <<CablingPCIe-x1-StandardTerminalNameType>> } CablingPCIe_x1 --> StandardTerminalNameAssignment : 1..> StandardTerminalNameAssignment --> Mapping : 1..> Mapping --> StandardTerminalName : 1..> </pre>												
type	CablingPCIe-x1-InterfaceFunctionType , CablingPCIe-x1-StandardTerminalNameAssignmentType , CablingPCIe-x1-StandardTerminalMappingType , CablingPCIe-x1-StandardTerminalNameType .												
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. SB_RTN</td> <td>2. CREFCLKp</td> <td>3. CREFCLKn</td> <td>4. CPWRON</td> </tr> <tr> <td>5. CWAKE#</td> <td>6. CPRSNT#</td> <td>7. CPERST#</td> <td>8. PETp0</td> </tr> <tr> <td>9. PETn0</td> <td>10. PERp0</td> <td>11. PERn0</td> <td></td> </tr> </table>	1. SB_RTN	2. CREFCLKp	3. CREFCLKn	4. CPWRON	5. CWAKE#	6. CPRSNT#	7. CPERST#	8. PETp0	9. PETn0	10. PERp0	11. PERn0	
1. SB_RTN	2. CREFCLKp	3. CREFCLKn	4. CPWRON										
5. CWAKE#	6. CPRSNT#	7. CPERST#	8. PETp0										
9. PETn0	10. PERp0	11. PERn0											

4.7.3.10.30.2. CablingPCIe-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCIe/CablingPCIe-x4																								
diagram	<pre> classDiagram class CablingPCIe-x4 { <<CablingPCIe-x4-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<CablingPCIe-x4-StandardTerminalNameAssignmentType>> } class Mapping { <<CablingPCIe-x4-StandardTerminalMappingType>> } class StandardTerminalName { <<CablingPCIe-x4-StandardTerminalNameType>> } CablingPCIe-x4 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for CablingPCIe-x4. It features four main classes: CablingPCIe-x4, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The CablingPCIe-x4 class is associated with the StandardTerminalNameAssignment class via a multiplicity of 1..>. The StandardTerminalNameAssignment class is associated with the Mapping class via a multiplicity of 1..>. The Mapping class is associated with the StandardTerminalName class via a multiplicity of 1..>. A dashed box labeled 'constraints' is present at the bottom right.</p>																								
type	CablingPCIe-x4-InterfaceFunctionType , CablingPCIe-x4-StandardTerminalNameAssignmentType , CablingPCIe-x4-StandardTerminalMappingType , CablingPCIe-x4-StandardTerminalNameType .																								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr><td>1. SB_RTN</td><td>2. CREFCLKp</td><td>3. CREFCLKn</td><td>4. CPWRON</td></tr> <tr><td>5. CWAKE#</td><td>6. CPRSNT#</td><td>7. CPERST#</td><td>8. PETp0</td></tr> <tr><td>9. PETp1</td><td>10. PETp2</td><td>11. PETp3</td><td>12. PETn0</td></tr> <tr><td>13. PETn1</td><td>14. PETn2</td><td>15. PETn3</td><td>16. PERp0</td></tr> <tr><td>17. PERp1</td><td>18. PERp2</td><td>19. PERp3</td><td>20. PERn0</td></tr> <tr><td>21. PERn1</td><td>22. PERn2</td><td>23. PERn3</td><td></td></tr> </table>	1. SB_RTN	2. CREFCLKp	3. CREFCLKn	4. CPWRON	5. CWAKE#	6. CPRSNT#	7. CPERST#	8. PETp0	9. PETp1	10. PETp2	11. PETp3	12. PETn0	13. PETn1	14. PETn2	15. PETn3	16. PERp0	17. PERp1	18. PERp2	19. PERp3	20. PERn0	21. PERn1	22. PERn2	23. PERn3	
1. SB_RTN	2. CREFCLKp	3. CREFCLKn	4. CPWRON																						
5. CWAKE#	6. CPRSNT#	7. CPERST#	8. PETp0																						
9. PETp1	10. PETp2	11. PETp3	12. PETn0																						
13. PETn1	14. PETn2	15. PETn3	16. PERp0																						
17. PERp1	18. PERp2	19. PERp3	20. PERn0																						
21. PERn1	22. PERn2	23. PERn3																							

4.7.3.10.30.3. CablingPCIe-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCIe/CablingPCIe-x8																																								
diagram	<pre> classDiagram class CablingPCIe-x8 { <<CablingPCIe-x8-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<CablingPCIe-x8-StandardTerminalNameAssignmentType>> } class Mapping { <<CablingPCIe-x8-StandardTerminalMappingType>> } class StandardTerminalName { <<CablingPCIe-x8-StandardTerminalNameType>> } CablingPCIe-x8 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for CablingPCIe-x8. It features four main classes: CablingPCIe-x8, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The CablingPCIe-x8 class is associated with the StandardTerminalNameAssignment class via a multiplicity of 1..>. The StandardTerminalNameAssignment class is associated with the Mapping class via a multiplicity of 1..>. The Mapping class is associated with the StandardTerminalName class via a multiplicity of 1..>. A dashed box labeled 'constraints' is present at the bottom right.</p>																																								
type	CablingPCIe-x8-InterfaceFunctionType , CablingPCIe-x8-StandardTerminalNameAssignmentType , CablingPCIe-x8-StandardTerminalMappingType , CablingPCIe-x8-StandardTerminalNameType .																																								
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr><td>1. SB_RTN</td><td>2. CREFCLKp</td><td>3. CREFCLKn</td><td>4. CPWRON</td></tr> <tr><td>5. CWAKE#</td><td>6. CPRSNT#</td><td>7. CPERST#</td><td>8. PETp0</td></tr> <tr><td>9. PETp1</td><td>10. PETp2</td><td>11. PETp3</td><td>12. PETp4</td></tr> <tr><td>13. PETp5</td><td>14. PETp6</td><td>15. PETp7</td><td>16. PETn0</td></tr> <tr><td>17. PETn1</td><td>18. PETn2</td><td>19. PETn3</td><td>20. PETn4</td></tr> <tr><td>21. PETn5</td><td>22. PETn6</td><td>23. PETn7</td><td>24. PERp0</td></tr> <tr><td>25. PERp1</td><td>26. PERp2</td><td>27. PERp3</td><td>28. PERp4</td></tr> <tr><td>29. PERp5</td><td>30. PERp6</td><td>31. PERp7</td><td>32. PERn0</td></tr> <tr><td>33. PERn1</td><td>34. PERn2</td><td>35. PERn3</td><td>36. PERn4</td></tr> <tr><td>37. PERn5</td><td>38. PERn6</td><td>39. PERn7</td><td></td></tr> </table>	1. SB_RTN	2. CREFCLKp	3. CREFCLKn	4. CPWRON	5. CWAKE#	6. CPRSNT#	7. CPERST#	8. PETp0	9. PETp1	10. PETp2	11. PETp3	12. PETp4	13. PETp5	14. PETp6	15. PETp7	16. PETn0	17. PETn1	18. PETn2	19. PETn3	20. PETn4	21. PETn5	22. PETn6	23. PETn7	24. PERp0	25. PERp1	26. PERp2	27. PERp3	28. PERp4	29. PERp5	30. PERp6	31. PERp7	32. PERn0	33. PERn1	34. PERn2	35. PERn3	36. PERn4	37. PERn5	38. PERn6	39. PERn7	
1. SB_RTN	2. CREFCLKp	3. CREFCLKn	4. CPWRON																																						
5. CWAKE#	6. CPRSNT#	7. CPERST#	8. PETp0																																						
9. PETp1	10. PETp2	11. PETp3	12. PETp4																																						
13. PETp5	14. PETp6	15. PETp7	16. PETn0																																						
17. PETn1	18. PETn2	19. PETn3	20. PETn4																																						
21. PETn5	22. PETn6	23. PETn7	24. PERp0																																						
25. PERp1	26. PERp2	27. PERp3	28. PERp4																																						
29. PERp5	30. PERp6	31. PERp7	32. PERn0																																						
33. PERn1	34. PERn2	35. PERn3	36. PERn4																																						
37. PERn5	38. PERn6	39. PERn7																																							

4.7.3.10.30.4. CablingPCIe-x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCIe/CablingPCIe-x16																																																																																
diagram	<pre> classDiagram class CablingPCIe_x16_InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName CablingPCIe_x16_InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for the CablingPCIe-x16 interface. It features four main classes: CablingPCIe-x16-InterfaceFunctionType, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The CablingPCIe-x16-InterfaceFunctionType class has a multiplicity of 1..> at its end of the association with StandardTerminalNameAssignment. The StandardTerminalNameAssignment class has a multiplicity of 1..> at its end of the association with Mapping. The Mapping class has a multiplicity of 1..> at its end of the association with StandardTerminalName. A dashed yellow box labeled 'constraints' encloses the three association lines.</p>																																																																																
type	CablingPCIe-x16-InterfaceFunctionType , CablingPCIe-x16-StandardTerminalNameAssignmentType , CablingPCIe-x16-StandardTerminalMappingType , CablingPCIe-x16-StandardTerminalNameType .																																																																																
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tbody> <tr><td>1. SB_RTN1</td><td>2. CREFCLKp1</td><td>3. CREFCLKn1</td><td>4. CPWRON1</td></tr> <tr><td>5. CWAKE#1</td><td>6. CPRSNT#1</td><td>7. CPERST#1</td><td>8. SB_RTN2</td></tr> <tr><td>9. CREFCLKp2</td><td>10. CREFCLKn2</td><td>11. CPWRON2</td><td>12. CWAKE#2</td></tr> <tr><td>13. CPRSNT#2</td><td>14. CPERST#2</td><td>15. PETp0</td><td>16. PETp1</td></tr> <tr><td>17. PETp2</td><td>18. PETp3</td><td>19. PETp4</td><td>20. PETp5</td></tr> <tr><td>21. PETp6</td><td>22. PETp7</td><td>23. PETp8</td><td>24. PETp9</td></tr> <tr><td>25. PETp10</td><td>26. PETp11</td><td>27. PETp12</td><td>28. PETp13</td></tr> <tr><td>29. PETp14</td><td>30. PETp15</td><td>31. PETn0</td><td>32. PETn1</td></tr> <tr><td>33. PETn2</td><td>34. PETn3</td><td>35. PETn4</td><td>36. PETn5</td></tr> <tr><td>37. PETn6</td><td>38. PETn7</td><td>39. PETn8</td><td>40. PETn9</td></tr> <tr><td>41. PETn10</td><td>42. PETn11</td><td>43. PETn12</td><td>44. PETn13</td></tr> <tr><td>45. PETn14</td><td>46. PETn15</td><td>47. PERp0</td><td>48. PERp1</td></tr> <tr><td>49. PERp2</td><td>50. PERp3</td><td>51. PERp4</td><td>52. PERp5</td></tr> <tr><td>53. PERp6</td><td>54. PERp7</td><td>55. PERp8</td><td>56. PERp9</td></tr> <tr><td>57. PERp10</td><td>58. PERp11</td><td>59. PERp12</td><td>60. PERp13</td></tr> <tr><td>61. PERp14</td><td>62. PERp15</td><td>63. PERn0</td><td>64. PERn1</td></tr> <tr><td>65. PERn2</td><td>66. PERn3</td><td>67. PERn4</td><td>68. PERn5</td></tr> <tr><td>69. PERn6</td><td>70. PERn7</td><td>71. PERn8</td><td>72. PERn9</td></tr> <tr><td>73. PERn10</td><td>74. PERn11</td><td>75. PERn12</td><td>76. PERn13</td></tr> <tr><td>77. PERn14</td><td>78. PERn15</td><td></td><td></td></tr> </tbody> </table>	1. SB_RTN1	2. CREFCLKp1	3. CREFCLKn1	4. CPWRON1	5. CWAKE#1	6. CPRSNT#1	7. CPERST#1	8. SB_RTN2	9. CREFCLKp2	10. CREFCLKn2	11. CPWRON2	12. CWAKE#2	13. CPRSNT#2	14. CPERST#2	15. PETp0	16. PETp1	17. PETp2	18. PETp3	19. PETp4	20. PETp5	21. PETp6	22. PETp7	23. PETp8	24. PETp9	25. PETp10	26. PETp11	27. PETp12	28. PETp13	29. PETp14	30. PETp15	31. PETn0	32. PETn1	33. PETn2	34. PETn3	35. PETn4	36. PETn5	37. PETn6	38. PETn7	39. PETn8	40. PETn9	41. PETn10	42. PETn11	43. PETn12	44. PETn13	45. PETn14	46. PETn15	47. PERp0	48. PERp1	49. PERp2	50. PERp3	51. PERp4	52. PERp5	53. PERp6	54. PERp7	55. PERp8	56. PERp9	57. PERp10	58. PERp11	59. PERp12	60. PERp13	61. PERp14	62. PERp15	63. PERn0	64. PERn1	65. PERn2	66. PERn3	67. PERn4	68. PERn5	69. PERn6	70. PERn7	71. PERn8	72. PERn9	73. PERn10	74. PERn11	75. PERn12	76. PERn13	77. PERn14	78. PERn15		
1. SB_RTN1	2. CREFCLKp1	3. CREFCLKn1	4. CPWRON1																																																																														
5. CWAKE#1	6. CPRSNT#1	7. CPERST#1	8. SB_RTN2																																																																														
9. CREFCLKp2	10. CREFCLKn2	11. CPWRON2	12. CWAKE#2																																																																														
13. CPRSNT#2	14. CPERST#2	15. PETp0	16. PETp1																																																																														
17. PETp2	18. PETp3	19. PETp4	20. PETp5																																																																														
21. PETp6	22. PETp7	23. PETp8	24. PETp9																																																																														
25. PETp10	26. PETp11	27. PETp12	28. PETp13																																																																														
29. PETp14	30. PETp15	31. PETn0	32. PETn1																																																																														
33. PETn2	34. PETn3	35. PETn4	36. PETn5																																																																														
37. PETn6	38. PETn7	39. PETn8	40. PETn9																																																																														
41. PETn10	42. PETn11	43. PETn12	44. PETn13																																																																														
45. PETn14	46. PETn15	47. PERp0	48. PERp1																																																																														
49. PERp2	50. PERp3	51. PERp4	52. PERp5																																																																														
53. PERp6	54. PERp7	55. PERp8	56. PERp9																																																																														
57. PERp10	58. PERp11	59. PERp12	60. PERp13																																																																														
61. PERp14	62. PERp15	63. PERn0	64. PERn1																																																																														
65. PERn2	66. PERn3	67. PERn4	68. PERn5																																																																														
69. PERn6	70. PERn7	71. PERn8	72. PERn9																																																																														
73. PERn10	74. PERn11	75. PERn12	76. PERn13																																																																														
77. PERn14	78. PERn15																																																																																

4.7.3.10.31. A-PHY

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/A-PHY							
diagram	<pre> classDiagram class A-PHY-InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName A-PHY-InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "3..>" StandardTerminalName </pre>							
type	A-PHY-InterfaceFunctionType , A-PHY-StandardTerminalNameAssignmentType , A-PHY-StandardTerminalMappingType , A-PHY-StandardTerminalNameType							
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. DP</td> <td>2. DN</td> <td>3. GND</td> <td></td> </tr> </table>				1. DP	2. DN	3. GND	
1. DP	2. DN	3. GND						

For more information about the A-PHY Interface, refer to the IEEE standard IEEE 2977-2021.

4.7.3.10.32. BoW-PHY

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/BoW-PHY																											
diagram	<pre> classDiagram class BoW-PHY-InterfaceFunctionType class StandardTerminalNameAssignment class MandatoryMapping class OptionalMapping class StandardTerminalName BoW-PHY-InterfaceFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping MandatoryMapping "18..>" StandardTerminalName StandardTerminalNameAssignment "0..2" OptionalMapping OptionalMapping "0..2" StandardTerminalName </pre>																											
type	BoW-PHY-InterfaceFunctionType , BoW-PHY-StandardTerminalNameAssignmentType , BoW-PHY-MandatoryStandardTerminalMappingType , BoW-PHY-MandatoryStandardTerminalNameType , BoW-PHY-OptionalStandardTerminalMappingType , BoW-PHY-OptionalStandardTerminalNameType .																											
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. CLK+</td> <td>2. CLK-</td> <td>3. D[0]</td> <td>4. D[1]</td> </tr> <tr> <td>5. D[2]</td> <td>6. D[3]</td> <td>7. D[4]</td> <td>8. D[5]</td> </tr> <tr> <td>9. D[6]</td> <td>10. D[7]</td> <td>11. D[8]</td> <td>12. D[9]</td> </tr> <tr> <td>13. D[10]</td> <td>14. D[11]</td> <td>15. D[12]</td> <td>16. D[13]</td> </tr> <tr> <td>17. D[14]</td> <td>18. D[15]</td> <td></td> <td></td> </tr> </table> OptionalMapping/StandardTerminalName <table border="1"> <tr> <td>1. FEC</td> <td>2. AUX</td> <td></td> <td></td> </tr> </table>				1. CLK+	2. CLK-	3. D[0]	4. D[1]	5. D[2]	6. D[3]	7. D[4]	8. D[5]	9. D[6]	10. D[7]	11. D[8]	12. D[9]	13. D[10]	14. D[11]	15. D[12]	16. D[13]	17. D[14]	18. D[15]			1. FEC	2. AUX		
1. CLK+	2. CLK-	3. D[0]	4. D[1]																									
5. D[2]	6. D[3]	7. D[4]	8. D[5]																									
9. D[6]	10. D[7]	11. D[8]	12. D[9]																									
13. D[10]	14. D[11]	15. D[12]	16. D[13]																									
17. D[14]	18. D[15]																											
1. FEC	2. AUX																											

For more information about the BoW-PHY Interface, refer to the OPEN Compute Project standard Bunch of Wires (BoW) PHY Specification.

4.7.3.10.33. C-PHY Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/C-PHY
diagram	<pre> classDiagram class C-PHY-InterfaceFunctionType { <<C-PHY-InterfaceFunctionType>> } class C-PHY-x1 { <<C-PHY-x1-InterfaceFunctionType>> } class C-PHY-x2 { <<C-PHY-x2-InterfaceFunctionType>> } class C-PHY-x3 { <<C-PHY-x3-InterfaceFunctionType>> } class C-PHY-x4 { <<C-PHY-x4-InterfaceFunctionType>> } class C-PHY-x5 { <<C-PHY-x5-InterfaceFunctionType>> } class C-PHY-x6 { <<C-PHY-x6-InterfaceFunctionType>> } C-PHY-InterfaceFunctionType < -- C-PHY-x1 C-PHY-InterfaceFunctionType < -- C-PHY-x2 C-PHY-InterfaceFunctionType < -- C-PHY-x3 C-PHY-InterfaceFunctionType < -- C-PHY-x4 C-PHY-InterfaceFunctionType < -- C-PHY-x5 C-PHY-InterfaceFunctionType < -- C-PHY-x6 </pre>
type	C-PHY-InterfaceFunctionType , C-PHY-x1-InterfaceFunctionType , C-PHY-x2-InterfaceFunctionType , C-PHY-x3-InterfaceFunctionType , C-PHY-x4-InterfaceFunctionType , C-PHY-x5-InterfaceFunctionType , C-PHY-x6-InterfaceFunctionType

For more information about the C-PHY Interfaces, refer to the MIPI Alliance standard Specification for C-PHY.

4.7.3.10.33.1. C-PHY-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/C-PHY/C-PHY-x1				
diagram	<pre> classDiagram class C-PHY-x1-InterfaceFunctionType { <<C-PHY-x1-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<C-PHY-x1-StandardTerminalNameAssignmentType>> } class Mapping { <<C-PHY-x1-StandardTerminalMappingType>> } class StandardTerminalName { <<C-PHY-x1-StandardTerminalNameType>> } C-PHY-x1-InterfaceFunctionType < -- StandardTerminalNameAssignment C-PHY-x1-InterfaceFunctionType < -- Mapping C-PHY-x1-InterfaceFunctionType < -- StandardTerminalName </pre>				
type	C-PHY-x1-InterfaceFunctionType , C-PHY-x1-StandardTerminalNameAssignmentType , C-PHY-x1-StandardTerminalMappingType , C-PHY-x1-StandardTerminalNameType				
list of enumerative values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. A</td> <td>2. B</td> <td>3. C</td> <td></td> </tr> </table>	1. A	2. B	3. C	
1. A	2. B	3. C			

4.7.3.10.33.2. C-PHY-x2

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x2												
diagram	<pre> classDiagram class C_PHY_x2_InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName C_PHY_x2_InterfaceFunctionType "1..2" -- "1..2" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..2" -- "1..2" Mapping Mapping "1..2" -- "1..2" StandardTerminalName </pre> <p>The diagram illustrates the structure of the C-PHY-x2 interface. It features a central 'Mapping' element connected to two 'StandardTerminalNameAssignment' elements, which in turn are connected to two 'C-PHY-x2-InterfaceFunctionType' elements. A dashed yellow box encloses the 'StandardTerminalNameAssignment' and 'Mapping' components, indicating they are part of the same functional group. A green dashed box encloses the 'StandardTerminalName' and its association with 'Mapping', further defining a sub-group.</p>												
type	C-PHY-x2-InterfaceFunctionType , C-PHY-x2-StandardTerminalNameAssignmentType , C-PHY-x2-StandardTerminalMappingType , C-PHY-x2-StandardTerminalNameType												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. A1</td><td>2. B1</td><td>3. C1</td><td>4. A2</td></tr> <tr> <td>5. B2</td><td>6. C2</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. A1	2. B1	3. C1	4. A2	5. B2	6. C2		
Mapping/StandardTerminalName													
1. A1	2. B1	3. C1	4. A2										
5. B2	6. C2												

4.7.3.10.33.3. C-PHY-x3

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x3																
diagram	<pre> classDiagram class C_PHY_x3_InterfaceFunctionType class StandardTerminalNameAssignment class Mapping class StandardTerminalName C_PHY_x3_InterfaceFunctionType "1..2" -- "1..2" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..2" -- "1..2" Mapping Mapping "1..2" -- "1..2" StandardTerminalName </pre> <p>The diagram illustrates the structure of the C-PHY-x3 interface, similar to the C-PHY-x2 diagram. It features a central 'Mapping' element connected to two 'StandardTerminalNameAssignment' elements, which in turn are connected to two 'C-PHY-x3-InterfaceFunctionType' elements. A dashed yellow box encloses the 'StandardTerminalNameAssignment' and 'Mapping' components, and a green dashed box encloses the 'StandardTerminalName' and its association with 'Mapping'.</p>																
type	C-PHY-x3-InterfaceFunctionType , C-PHY-x3-StandardTerminalNameAssignmentType , C-PHY-x3-StandardTerminalMappingType , C-PHY-x3-StandardTerminalNameType																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. A1</td><td>2. B1</td><td>3. C1</td><td>4. A2</td></tr> <tr> <td>5. B2</td><td>6. C2</td><td>7. A3</td><td>8. B3</td></tr> <tr> <td>9. C3</td><td></td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. A1	2. B1	3. C1	4. A2	5. B2	6. C2	7. A3	8. B3	9. C3			
Mapping/StandardTerminalName																	
1. A1	2. B1	3. C1	4. A2														
5. B2	6. C2	7. A3	8. B3														
9. C3																	

4.7.3.10.33.4. C-PHY-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x4																
diagram	<pre> classDiagram class C_PHY_x4 { <<C-PHY-x4>> <<type C-PHY-x4-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<C-PHY-x4-StandardTerminalNameAssignmentType>> <<type C-PHY-x4-StandardTerminalNameAssignmentType>> } class Mapping { <<C-PHY-x4-StandardTerminalMappingType>> <<type C-PHY-x4-StandardTerminalMappingType>> } class StandardTerminalName { <<C-PHY-x4-StandardTerminalNameType>> <<type C-PHY-x4-StandardTerminalNameType>> } C_PHY_x4 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for C-PHY-x4. It features four main classes: C-PHY-x4-InterfaceFunctionType, C-PHY-x4-StandardTerminalNameAssignmentType, C-PHY-x4-StandardTerminalMappingType, and C-PHY-x4-StandardTerminalNameType. The C-PHY-x4-InterfaceFunctionType class has a multiplicity of 1..> to the C-PHY-x4-StandardTerminalNameAssignmentType class. The C-PHY-x4-StandardTerminalNameAssignmentType class has a multiplicity of 1..> to the C-PHY-x4-StandardTerminalMappingType class. The C-PHY-x4-StandardTerminalMappingType class has a multiplicity of 1..> to the C-PHY-x4-StandardTerminalNameType class. A dashed box labeled 'constraints' is present.</p>																
type	C-PHY-x4-InterfaceFunctionType , C-PHY-x4-StandardTerminalNameAssignmentType , C-PHY-x4-StandardTerminalMappingType , C-PHY-x4-StandardTerminalNameType																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. A1</td><td>2. B1</td><td>3. C1</td><td>4. A2</td></tr> <tr> <td>5. B2</td><td>6. C2</td><td>7. A3</td><td>8. B3</td></tr> <tr> <td>9. C3</td><td>10. A4</td><td>11. B4</td><td>12. C4</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. A1	2. B1	3. C1	4. A2	5. B2	6. C2	7. A3	8. B3	9. C3	10. A4	11. B4	12. C4
Mapping/StandardTerminalName																	
1. A1	2. B1	3. C1	4. A2														
5. B2	6. C2	7. A3	8. B3														
9. C3	10. A4	11. B4	12. C4														

4.7.3.10.33.5. C-PHY-x5

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x5																				
diagram	<pre> classDiagram class C_PHY_x5 { <<C-PHY-x5>> <<type C-PHY-x5-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<C-PHY-x5-StandardTerminalNameAssignmentType>> <<type C-PHY-x5-StandardTerminalNameAssignmentType>> } class Mapping { <<C-PHY-x5-StandardTerminalMappingType>> <<type C-PHY-x5-StandardTerminalMappingType>> } class StandardTerminalName { <<C-PHY-x5-StandardTerminalNameType>> <<type C-PHY-x5-StandardTerminalNameType>> } C_PHY_x5 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for C-PHY-x5. It features four main classes: C-PHY-x5-InterfaceFunctionType, C-PHY-x5-StandardTerminalNameAssignmentType, C-PHY-x5-StandardTerminalMappingType, and C-PHY-x5-StandardTerminalNameType. The C-PHY-x5-InterfaceFunctionType class has a multiplicity of 1..> to the C-PHY-x5-StandardTerminalNameAssignmentType class. The C-PHY-x5-StandardTerminalNameAssignmentType class has a multiplicity of 1..> to the C-PHY-x5-StandardTerminalMappingType class. The C-PHY-x5-StandardTerminalMappingType class has a multiplicity of 1..> to the C-PHY-x5-StandardTerminalNameType class. A dashed box labeled 'constraints' is present.</p>																				
type	C-PHY-x5-InterfaceFunctionType , C-PHY-x5-StandardTerminalNameAssignmentType , C-PHY-x5-StandardTerminalMappingType , C-PHY-x5-StandardTerminalNameType																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. A1</td><td>2. B1</td><td>3. C1</td><td>4. A2</td></tr> <tr> <td>5. B2</td><td>6. C2</td><td>7. A3</td><td>8. B3</td></tr> <tr> <td>9. C3</td><td>10. A4</td><td>11. B4</td><td>12. C4</td></tr> <tr> <td>13. A5</td><td>14. B5</td><td>15. C5</td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. A1	2. B1	3. C1	4. A2	5. B2	6. C2	7. A3	8. B3	9. C3	10. A4	11. B4	12. C4	13. A5	14. B5	15. C5	
Mapping/StandardTerminalName																					
1. A1	2. B1	3. C1	4. A2																		
5. B2	6. C2	7. A3	8. B3																		
9. C3	10. A4	11. B4	12. C4																		
13. A5	14. B5	15. C5																			

4.7.3.10.33.6. C-PHY-x6

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x6																								
diagram	<pre> classDiagram class C_PHY_x6 { <<C-PHY-x6<> <<type C-PHY-x6-InterfaceFunctionType<> } class StandardTerminalNameAssignment { <<C-PHY-x6-StandardTerminalNameAssignmentType<> <<type C-PHY-x6-StandardTerminalNameAssignmentType<> } class Mapping { <<C-PHY-x6-StandardTerminalMappingType<> <<type C-PHY-x6-StandardTerminalMappingType<> } class StandardTerminalName { <<C-PHY-x6-StandardTerminalNameType<> <<type C-PHY-x6-StandardTerminalNameType<> } C_PHY_x6 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for the C-PHY-x6 interface. It features four main classes: C-PHY-x6, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The C-PHY-x6 class is associated with the StandardTerminalNameAssignment class via a multiplicity of 1..>. The StandardTerminalNameAssignment class is associated with the Mapping class via a multiplicity of 1..>. The Mapping class is associated with the StandardTerminalName class via a multiplicity of 1..>. A dashed yellow box labeled 'constraints' encloses the entire sequence of associations.</p>																								
type	C-PHY-x6-InterfaceFunctionType , C-PHY-x6-StandardTerminalNameAssignmentType , C-PHY-x6-StandardTerminalMappingType , C-PHY-x6-StandardTerminalNameType																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. A1</td><td>2. B1</td><td>3. C1</td><td>4. A2</td></tr> <tr> <td>5. B2</td><td>6. C2</td><td>7. A3</td><td>8. B3</td></tr> <tr> <td>9. C3</td><td>10. A4</td><td>11. B4</td><td>12. C4</td></tr> <tr> <td>13. A5</td><td>14. B5</td><td>15. C5</td><td>16. A6</td></tr> <tr> <td>17. B6</td><td>18. C6</td><td></td><td></td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. A1	2. B1	3. C1	4. A2	5. B2	6. C2	7. A3	8. B3	9. C3	10. A4	11. B4	12. C4	13. A5	14. B5	15. C5	16. A6	17. B6	18. C6		
Mapping/StandardTerminalName																									
1. A1	2. B1	3. C1	4. A2																						
5. B2	6. C2	7. A3	8. B3																						
9. C3	10. A4	11. B4	12. C4																						
13. A5	14. B5	15. C5	16. A6																						
17. B6	18. C6																								

4.7.3.10.34. D-PHY Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/D-PHY
diagram	<pre> classDiagram class D-PHY { <<D-PHY-InterfaceFunctionType>> } class D-PHY-InterfaceFunctionType { D-PHY-x1 D-PHY-x2 D-PHY-x3 D-PHY-x4 D-PHY-x5 D-PHY-x6 D-PHY-x7 D-PHY-x8 } D-PHY "1" -- "1" D-PHY-InterfaceFunctionType </pre>
type	D-PHY-InterfaceFunctionType , D-PHY-x1-InterfaceFunctionType , D-PHY-x2-InterfaceFunctionType , D-PHY-x3-InterfaceFunctionType , D-PHY-x4-InterfaceFunctionType , D-PHY-x5-InterfaceFunctionType , D-PHY-x6-InterfaceFunctionType , D-PHY-x7-InterfaceFunctionType , D-PHY-x8-InterfaceFunctionType

For more information about the D-PHY Interfaces, refer to the MIPI Alliance standard Specification for D-PHY Version 1.00.00.

4.7.3.10.34.1. D-PHY-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x1								
diagram	<pre> classDiagram class D_PHY_x1 { <<D-PHY-x1-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<D-PHY-x1-StandardTerminalNameAssignmentType>> } class Mapping { <<D-PHY-x1-StandardTerminalMappingType>> } class StandardTerminalName { <<D-PHY-x1-StandardTerminalNameType>> } D_PHY_x1 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the mapping process for D-PHY-x1. It starts with a D-PHY-x1 object (type D-PHY-x1-InterfaceFunctionType) which is associated with a StandardTerminalNameAssignment object (type D-PHY-x1-StandardTerminalNameAssignmentType). This assignment is then mapped to a StandardTerminalName object (type D-PHY-x1-StandardTerminalNameType) via a Mapping object (type D-PHY-x1-StandardTerminalMappingType). A dashed yellow box labeled 'constraints' encloses the entire sequence.</p>								
type	D-PHY-x1-InterfaceFunctionType, D-PHY-x1-StandardTerminalNameAssignmentType, D-PHY-x1-StandardTerminalMappingType, D-PHY-x1-StandardTerminalNameType								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td></td> <td></td> </tr> </table>	1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-		
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-						
5. DATA1+	6. DATA1-								

4.7.3.10.34.2. D-PHY-x2

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x2								
diagram	<pre> classDiagram class D_PHY_x2 { <<D-PHY-x2-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<D-PHY-x2-StandardTerminalNameAssignmentType>> } class Mapping { <<D-PHY-x2-StandardTerminalMappingType>> } class StandardTerminalName { <<D-PHY-x2-StandardTerminalNameType>> } D_PHY_x2 "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the mapping process for D-PHY-x2. It starts with a D-PHY-x2 object (type D-PHY-x2-InterfaceFunctionType) which is associated with a StandardTerminalNameAssignment object (type D-PHY-x2-StandardTerminalNameAssignmentType). This assignment is then mapped to a StandardTerminalName object (type D-PHY-x2-StandardTerminalNameType) via a Mapping object (type D-PHY-x2-StandardTerminalMappingType). A dashed yellow box labeled 'constraints' encloses the entire sequence.</p>								
type	D-PHY-x2-InterfaceFunctionType, D-PHY-x2-StandardTerminalNameAssignmentType, D-PHY-x2-StandardTerminalMappingType, D-PHY-x2-StandardTerminalNameType								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td></td> <td></td> </tr> </table>	1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-		
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-						
5. DATA1+	6. DATA1-								

4.7.3.10.34.3. D-PHY-x3

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x3</code>												
diagram	<p>The diagram illustrates the class structure for D-PHY-x3. It features a central <code>D-PHY-x3</code> class connected via an association labeled <code>1..oo</code> to a <code>StandardTerminalNameAssignment</code> class. This association is part of a larger sequence: <code>D-PHY-x3</code> → <code>StandardTerminalNameAssignment</code> → <code>Mapping</code> → <code>StandardTerminalName</code>. The <code>Mapping</code> class is associated with <code>StandardTerminalName</code> via an association labeled <code>8</code>. A dashed box labeled <code>constraints</code> encloses the <code>StandardTerminalNameAssignment</code>, <code>Mapping</code>, and <code>StandardTerminalName</code> classes. Above this box, a dashed box labeled <code>D-PHY-x3-StandardTerminalNameAssignmentType</code> contains the <code>StandardTerminalNameAssignment</code> class. Above that, another dashed box labeled <code>D-PHY-x3-InterfaceFunctionType</code> contains the <code>D-PHY-x3</code> class.</p>												
type	<code>D-PHY-x3-InterfaceFunctionType</code> , <code>D-PHY-x3-StandardTerminalNameAssignmentType</code> , <code>D-PHY-x3-StandardTerminalMappingType</code> , <code>D-PHY-x3-StandardTerminalNameType</code>												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4"><code>Mapping/StandardTerminalName</code></th> </tr> </thead> <tbody> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td>7. DATA2+</td> <td>8. DATA2-</td> </tr> </tbody> </table>	<code>Mapping/StandardTerminalName</code>				1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-
<code>Mapping/StandardTerminalName</code>													
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-										
5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-										

4.7.3.10.34.4. D-PHY-x4

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x4</code>																
diagram	<p>The diagram illustrates the class structure for D-PHY-x4. It features a central <code>D-PHY-x4</code> class connected via an association labeled <code>1..oo</code> to a <code>StandardTerminalNameAssignment</code> class. This association is part of a larger sequence: <code>D-PHY-x4</code> → <code>StandardTerminalNameAssignment</code> → <code>Mapping</code> → <code>StandardTerminalName</code>. The <code>Mapping</code> class is associated with <code>StandardTerminalName</code> via an association labeled <code>10</code>. A dashed box labeled <code>constraints</code> encloses the <code>StandardTerminalNameAssignment</code>, <code>Mapping</code>, and <code>StandardTerminalName</code> classes. Above this box, a dashed box labeled <code>D-PHY-x4-StandardTerminalNameAssignmentType</code> contains the <code>StandardTerminalNameAssignment</code> class. Above that, another dashed box labeled <code>D-PHY-x4-InterfaceFunctionType</code> contains the <code>D-PHY-x4</code> class.</p>																
type	<code>D-PHY-x4-InterfaceFunctionType</code> , <code>D-PHY-x4-StandardTerminalNameAssignmentType</code> , <code>D-PHY-x4-StandardTerminalMappingType</code> , <code>D-PHY-x4-StandardTerminalNameType</code>																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4"><code>Mapping/StandardTerminalName</code></th> </tr> </thead> <tbody> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td>7. DATA2+</td> <td>8. DATA2-</td> </tr> <tr> <td>9. DATA3+</td> <td>10. DATA3-</td> <td></td> <td></td> </tr> </tbody> </table>	<code>Mapping/StandardTerminalName</code>				1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-	9. DATA3+	10. DATA3-		
<code>Mapping/StandardTerminalName</code>																	
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-														
5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-														
9. DATA3+	10. DATA3-																

4.7.3.10.34.5. D-PHY-x5

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x5																
diagram	<pre> classDiagram class D_PHY_x5 { <<D-PHY-x5-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<D-PHY-x5-StandardTerminalNameAssignmentType>> } class Mapping { <<D-PHY-x5-StandardTerminalMap...>> } class StandardTerminalName { <<D-PHY-x5-StandardTerminalName...>> } D_PHY_x5 --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre> <p>The diagram illustrates the functional hierarchy for D-PHY-x5. It starts with a box labeled "D-PHY-x5-InterfaceFunctionType" containing a "D-PHY-x5" object. This connects via a directed association to a "StandardTerminalNameAssignment" object, which is contained within a box labeled "D-PHY-x5-StandardTerminalNameAssignmentType". From there, another directed association leads to a "Mapping" object, which is contained within a box labeled "D-PHY-x5-StandardTerminalMappingType". Finally, a directed association leads to a "StandardTerminalName" object, which is contained within a box labeled "D-PHY-x5-StandardTerminalNameType". A dashed green line labeled "constraints" connects the "Mapping" and "StandardTerminalName" boxes.</p>																
type	D-PHY-x5-InterfaceFunctionType , D-PHY-x5-StandardTerminalNameAssignmentType , D-PHY-x5-StandardTerminalMappingType , D-PHY-x5-StandardTerminalNameType																
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td>7. DATA2+</td> <td>8. DATA2-</td> </tr> <tr> <td>9. DATA3+</td> <td>10. DATA3-</td> <td>11. DATA4+</td> <td>12. DATA4-</td> </tr> </tbody> </table>	Mapping/StandardTerminalName				1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-	9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-
Mapping/StandardTerminalName																	
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-														
5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-														
9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-														

4.7.3.10.34.6. D-PHY-x6

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x6																				
diagram	<pre> classDiagram class D_PHY_x6 { <<D-PHY-x6-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<D-PHY-x6-StandardTerminalNameAssignmentType>> } class Mapping { <<D-PHY-x6-StandardTerminalMap...>> } class StandardTerminalName { <<D-PHY-x6-StandardTerminalName...>> } D_PHY_x6 --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre> <p>The diagram illustrates the functional hierarchy for D-PHY-x6. It starts with a box labeled "D-PHY-x6-InterfaceFunctionType" containing a "D-PHY-x6" object. This connects via a directed association to a "StandardTerminalNameAssignment" object, which is contained within a box labeled "D-PHY-x6-StandardTerminalNameAssignmentType". From there, another directed association leads to a "Mapping" object, which is contained within a box labeled "D-PHY-x6-StandardTerminalMappingType". Finally, a directed association leads to a "StandardTerminalName" object, which is contained within a box labeled "D-PHY-x6-StandardTerminalNameType". A dashed green line labeled "constraints" connects the "Mapping" and "StandardTerminalName" boxes.</p>																				
type	D-PHY-x6-InterfaceFunctionType , D-PHY-x6-StandardTerminalNameAssignmentType , D-PHY-x6-StandardTerminalMappingType , D-PHY-x6-StandardTerminalNameType																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td>7. DATA2+</td> <td>8. DATA2-</td> </tr> <tr> <td>9. DATA3+</td> <td>10. DATA3-</td> <td>11. DATA4+</td> <td>12. DATA4-</td> </tr> <tr> <td>13. DATA5+</td> <td>14. DATA5-</td> <td></td> <td></td> </tr> </tbody> </table>	Mapping/StandardTerminalName				1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-	9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-	13. DATA5+	14. DATA5-		
Mapping/StandardTerminalName																					
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-																		
5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-																		
9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-																		
13. DATA5+	14. DATA5-																				

4.7.3.10.34.7. D-PHY-x7

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x7																				
diagram	<pre> classDiagram class D_PHY_x7_InterfaceFunctionType { <<D-PHY-x7-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<D-PHY-x7-StandardTerminalNameAssignmentType>> } class Mapping { <<D-PHY-x7-StandardTerminalMappingType>> } class StandardTerminalName { <<D-PHY-x7-StandardTerminalNameType>> } D_PHY_x7_InterfaceFunctionType "1..oo" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..oo" --> Mapping Mapping "1..oo" --> StandardTerminalName </pre> <p>The diagram illustrates the mapping of D-PHY-x7 interface functions to standard terminal names. It shows three main components: D-PHY-x7-InterfaceFunctionType, D-PHY-x7-StandardTerminalNameAssignmentType, and D-PHY-x7-StandardTerminalMappingType. The D-PHY-x7-InterfaceFunctionType is connected to the D-PHY-x7-StandardTerminalNameAssignmentType via a multiplicity of 1..oo. The D-PHY-x7-StandardTerminalNameAssignmentType is connected to the D-PHY-x7-StandardTerminalMappingType via a multiplicity of 1..oo. The D-PHY-x7-StandardTerminalMappingType is connected to the D-PHY-x7-StandardTerminalNameType via a multiplicity of 1..oo. A constraint labeled 'constraints' is associated with the mapping.</p>																				
type	D-PHY-x7-InterfaceFunctionType , D-PHY-x7-StandardTerminalNameAssignmentType , D-PHY-x7-StandardTerminalMappingType , D-PHY-x7-StandardTerminalNameType																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td>7. DATA2+</td> <td>8. DATA2-</td> </tr> <tr> <td>9. DATA3+</td> <td>10. DATA3-</td> <td>11. DATA4+</td> <td>12. DATA4-</td> </tr> <tr> <td>13. DATA5+</td> <td>14. DATA5-</td> <td>15. DATA6+</td> <td>16. DATA6-</td> </tr> </tbody> </table>	Mapping/StandardTerminalName				1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-	9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-	13. DATA5+	14. DATA5-	15. DATA6+	16. DATA6-
Mapping/StandardTerminalName																					
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-																		
5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-																		
9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-																		
13. DATA5+	14. DATA5-	15. DATA6+	16. DATA6-																		

4.7.3.10.34.8. D-PHY-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x8																								
diagram	<pre> classDiagram class D_PHY_x8_InterfaceFunctionType { <<D-PHY-x8-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<D-PHY-x8-StandardTerminalNameAssignmentType>> } class Mapping { <<D-PHY-x8-StandardTerminalMappingType>> } class StandardTerminalName { <<D-PHY-x8-StandardTerminalNameType>> } D_PHY_x8_InterfaceFunctionType "1..oo" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..oo" --> Mapping Mapping "1..oo" --> StandardTerminalName </pre> <p>The diagram illustrates the mapping of D-PHY-x8 interface functions to standard terminal names. It shows three main components: D-PHY-x8-InterfaceFunctionType, D-PHY-x8-StandardTerminalNameAssignmentType, and D-PHY-x8-StandardTerminalMappingType. The D-PHY-x8-InterfaceFunctionType is connected to the D-PHY-x8-StandardTerminalNameAssignmentType via a multiplicity of 1..oo. The D-PHY-x8-StandardTerminalNameAssignmentType is connected to the D-PHY-x8-StandardTerminalMappingType via a multiplicity of 1..oo. The D-PHY-x8-StandardTerminalMappingType is connected to the D-PHY-x8-StandardTerminalNameType via a multiplicity of 1..oo. A constraint labeled 'constraints' is associated with the mapping.</p>																								
type	D-PHY-x8-InterfaceFunctionType , D-PHY-x8-StandardTerminalNameAssignmentType , D-PHY-x8-StandardTerminalMappingType , D-PHY-x8-StandardTerminalNameType																								
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CLOCK+</td> <td>2. CLOCK-</td> <td>3. DATA0+</td> <td>4. DATA0-</td> </tr> <tr> <td>5. DATA1+</td> <td>6. DATA1-</td> <td>7. DATA2+</td> <td>8. DATA2-</td> </tr> <tr> <td>9. DATA3+</td> <td>10. DATA3-</td> <td>11. DATA4+</td> <td>12. DATA4-</td> </tr> <tr> <td>13. DATA5+</td> <td>14. DATA5-</td> <td>15. DATA6+</td> <td>16. DATA6-</td> </tr> <tr> <td>17. DATA7+</td> <td>18. DATA7-</td> <td></td> <td></td> </tr> </tbody> </table>	Mapping/StandardTerminalName				1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-	5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-	9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-	13. DATA5+	14. DATA5-	15. DATA6+	16. DATA6-	17. DATA7+	18. DATA7-		
Mapping/StandardTerminalName																									
1. CLOCK+	2. CLOCK-	3. DATA0+	4. DATA0-																						
5. DATA1+	6. DATA1-	7. DATA2+	8. DATA2-																						
9. DATA3+	10. DATA3-	11. DATA4+	12. DATA4-																						
13. DATA5+	14. DATA5-	15. DATA6+	16. DATA6-																						
17. DATA7+	18. DATA7-																								

4.7.3.10.35. M-PHY

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/M-PHY			
diagram				
type	M-PHY-InterfaceFunctionType, M-PHY-StandardTerminalNameAssignmentType, M-PHY-StandardTerminalMappingType, M-PHY-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName 1. TXDP 2. TXDN 3. RXDP 4. RXDN			

For more information about the M-PHY Interfaces, refer to the MIPI Alliance standard Specification for M-PHY Version 4.1.

4.7.3.10.36. OpenHBI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/OpenHBI			
diagram				
type	OpenHBI-InterfaceFunctionType, OpenHBI -StandardTerminalNameAssignmentType, OpenHBI -StandardTerminalMappingType, OpenHBI -StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName 1. D[0] 2. D[1] 3. D[2] 4. D[3] 5. D[4] 6. D[5] 7. D[6] 8. D[7] 9. D[8] 10. D[9] 11. D[10] 12. D[11] 13. D[12] 14. D[13] 15. D[14] 16. D[15] 17. D[16] 18. D[17] 19. D[18] 20. D[19] 21. D[20] 22. D[21] 23. D[22] 24. D[23] 25. D[24] 26. D[25] 27. D[26] 28. D[27] 29. D[28] 30. D[29] 31. D[30] 32. D[31] 33. D[32] 34. D[33] 35. D[34] 36. D[35] 37. D[36] 38. D[37] 39. D[38] 40. D[39] 41. D[40] 42. D[41] 43. WDQS_t 44. WDQS_c 45. RDQS_t 46. WDQS_c 47. RD0 48. RD1			

For more information about the OpenHBI Interface, refer to the OPEN Compute Project standard OpenHBI Specification Version 1.0

4.7.3.10.37. PTI Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/PTI			
diagram				
type	PTI-InterfaceFunctionType, PTI -StandardTerminalNameAssignmentType, PTI-StandardTerminalMappingType, PTI-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. TRC_CLK	2. DATA[0]	3. DATA[1]	4. DATA[2]
	5. DATA[3]			

For more information about the PTI Interface, refer to the MIPI Alliance standard Specification for Parallel Trace Interface (PTI) Version 2.0.1.

4.7.3.10.38. Radio Front End Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/RadioFrontEnd			
diagram				
type	RadioFrontEnd-interfaceFunctionType, RBDP-InterfaceFunctionType, RF-BB-InterfaceFunctionType.			

4.7.3.10.38.1. RBDP

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/RadioFrontEnd/RBDP																												
diagram	<pre> classDiagram class RBDP-InterfaceFunctionType class StandardTerminalNameAssignment class RBDP-StandardTerminalNameAssignmentType { class RBDP-MandatoryStandardTerminalMappingType { class StandardTerminalName } class RBDP-OptionalStandardTerminalMappingType { class StandardTerminalName } } RBDP-InterfaceFunctionType "1..oo" --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> RBDP-StandardTerminalNameAssignmentType RBDP-StandardTerminalNameAssignmentType "1..14" --> RBDP-MandatoryStandardTerminalMappingType RBDP-StandardTerminalNameAssignmentType "0..2" --> RBDP-OptionalStandardTerminalMappingType RBDP-MandatoryStandardTerminalMappingType --> StandardTerminalName RBDP-OptionalStandardTerminalMappingType --> StandardTerminalName </pre>																												
type	RBDP-InterfaceFunctionType, RBDP-StandardTerminalNameAssignmentType, RBDP-MandatoryStandardTerminalMappingType, RBDP-MandatoryStandardTerminalNameType, RBDP-OptionalStandardTerminalMappingType, RBDP-OptionalStandardTerminalNameType.																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. MCLK</td><td>2. FCLK</td><td>3. TXNRX</td><td>4. ENABLE</td></tr> <tr><td>5. DIQ[0]</td><td>6. DIQ[1]</td><td>7. DIQ[2]</td><td>8. DIQ[3]</td></tr> <tr><td>9. DIQ[4]</td><td>10. DIQ[5]</td><td>11. DIQ[6]</td><td>12. DIQ[7]</td></tr> <tr><td>13. D1Q[8]</td><td>14. D1Q[9]</td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. D1Q[10]</td><td>2. D1Q[11]</td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. MCLK	2. FCLK	3. TXNRX	4. ENABLE	5. DIQ[0]	6. DIQ[1]	7. DIQ[2]	8. DIQ[3]	9. DIQ[4]	10. DIQ[5]	11. DIQ[6]	12. DIQ[7]	13. D1Q[8]	14. D1Q[9]			OptionalMapping/StandardTerminalName				1. D1Q[10]	2. D1Q[11]		
MandatoryMapping/StandardTerminalName																													
1. MCLK	2. FCLK	3. TXNRX	4. ENABLE																										
5. DIQ[0]	6. DIQ[1]	7. DIQ[2]	8. DIQ[3]																										
9. DIQ[4]	10. DIQ[5]	11. DIQ[6]	12. DIQ[7]																										
13. D1Q[8]	14. D1Q[9]																												
OptionalMapping/StandardTerminalName																													
1. D1Q[10]	2. D1Q[11]																												

For more information about the RBDP Interface, refer to JEDEC Standard JESD207.

4.7.3.10.38.2. RF-BB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/RadioFrontEnd/RF-BB																				
diagram	<pre> classDiagram class RF-BB-InterfaceFunctionType class StandardTerminalNameAssignment class RF-BB-StandardTerminalNameAssignmentType { class RF-BB-MandatoryStandardTerminalMappingType { class StandardTerminalName } class RF-BB-OptionalStandardTerminalMappingType { class StandardTerminalName } } RF-BB-InterfaceFunctionType "1..oo" --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> RF-BB-StandardTerminalNameAssignmentType RF-BB-StandardTerminalNameAssignmentType "1..6" --> RF-BB-MandatoryStandardTerminalMappingType RF-BB-StandardTerminalNameAssignmentType "0..2" --> RF-BB-OptionalStandardTerminalMappingType RF-BB-MandatoryStandardTerminalMappingType --> StandardTerminalName RF-BB-OptionalStandardTerminalMappingType --> StandardTerminalName </pre>																				
type	RF-BB-InterfaceFunctionType, RF-BB-StandardTerminalNameAssignmentType, RF-BB-MandatoryStandardTerminalMappingType, RF-BB-MandatoryStandardTerminalNameType, RF-BB-OptionalStandardTerminalMappingType, RF-BB-OptionalStandardTerminalNameType.																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr><td>1. CLK</td><td>2. CLKN</td><td>3. TX</td><td>4. TXN</td></tr> <tr><td>5. RX</td><td>6. RXN</td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr><td>1. Tx_Clock</td><td>2. Tx_ClockN</td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. CLK	2. CLKN	3. TX	4. TXN	5. RX	6. RXN			OptionalMapping/StandardTerminalName				1. Tx_Clock	2. Tx_ClockN		
MandatoryMapping/StandardTerminalName																					
1. CLK	2. CLKN	3. TX	4. TXN																		
5. RX	6. RXN																				
OptionalMapping/StandardTerminalName																					
1. Tx_Clock	2. Tx_ClockN																				

For more information about the RBDP Interface, refer to JEDEC Standard JESD96A.

4.7.3.10.39. RFFE

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/RFFE							
diagram								
type	RFFE-InterfaceFunctionType , RFFE-StandardTerminalNameAssignmentType , RFFE-StandardTerminalMappingType , RFFE-StandardTerminalNameType							
list of enumerative values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. SCLK</td><td style="padding: 2px;">2. SDATA</td><td style="padding: 2px;">3. VIO</td><td style="padding: 2px;"></td></tr> </table>				1. SCLK	2. SDATA	3. VIO	
1. SCLK	2. SDATA	3. VIO						

For more information about the M-PHY Interfaces, refer to the MIPI Alliance standard Specification for RF Front-End Control Interface Version 1.10.

4.7.3.10.40. SD

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SD											
diagram												
type	SD-InterfaceFunctionType , SD-StandardTerminalNameAssignmentType , SD-StandardTerminalMappingType , SD-StandardTerminalNameType											
list of enumerative values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. CLK</td><td style="padding: 2px;">2. DAT[0]</td><td style="padding: 2px;">3. DAT[1]</td><td style="padding: 2px;">4. DAT[2]</td></tr> <tr> <td style="padding: 2px;">5. DAT[3]</td><td style="padding: 2px;">6. CMD</td><td style="padding: 2px;"></td><td style="padding: 2px;"></td></tr> </table>				1. CLK	2. DAT[0]	3. DAT[1]	4. DAT[2]	5. DAT[3]	6. CMD		
1. CLK	2. DAT[0]	3. DAT[1]	4. DAT[2]									
5. DAT[3]	6. CMD											

4.7.3.10.41. SD-UHS-II

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SD-UHS-II																
diagram	<pre> classDiagram SD-UHS-II "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the structure of the SD-UHS-II interface. It starts with a 'SD-UHS-II' object, which has a multiplicity of 1..∞ to 'StandardTerminalNameAssignment'. This leads to a 'Mapping' object, which in turn has a multiplicity of 1..∞ to 'StandardTerminalName'. A dashed box labeled 'constraints' is shown near the 'Mapping' object.</p>																
type	SD-UHS-II-InterfaceFunctionType , SD-UHS-II-StandardTerminalNameAssignmentType , SD-UHS-II-StandardTerminalMappingType , SD-UHS-II-StandardTerminalNameType																
list of enumerative values	<table border="1"> <thead> <tr> <th colspan="4">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. CLK</td><td>2. DAT[0]</td><td>3. DAT[1]</td><td>4. DAT[2]</td></tr> <tr> <td>5. DAT[3]</td><td>6. CMD</td><td>7. RCLK+</td><td>8. RCLK-</td></tr> <tr> <td>9. D0+</td><td>10. D0-</td><td>11. D1+</td><td>12. D1-</td></tr> </tbody> </table>	Mapping/StandardTerminalName				1. CLK	2. DAT[0]	3. DAT[1]	4. DAT[2]	5. DAT[3]	6. CMD	7. RCLK+	8. RCLK-	9. D0+	10. D0-	11. D1+	12. D1-
Mapping/StandardTerminalName																	
1. CLK	2. DAT[0]	3. DAT[1]	4. DAT[2]														
5. DAT[3]	6. CMD	7. RCLK+	8. RCLK-														
9. D0+	10. D0-	11. D1+	12. D1-														

4.7.3.10.42. Serial Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface
diagram	<pre> classDiagram SerialInterface "1..>" SI SI "1..>" SPI SPI "1..>" eSPI eSPI "1..>" xSPI xSPI "1..>" SPD5118Hub </pre> <p>The diagram shows the inheritance structure of serial interface functions. It starts with a 'SerialInterface' object, which has a multiplicity of 1..∞ to 'SI'. 'SI' has a multiplicity of 1..∞ to 'SPI'. 'SPI' has a multiplicity of 1..∞ to 'eSPI'. 'eSPI' has a multiplicity of 1..∞ to 'xSPI'. Finally, 'xSPI' has a multiplicity of 1..∞ to 'SPD5118Hub'.</p>
type	SerialInterfaceFunctionType , SI-InterfaceFunctionType , SPI-InterfaceFunctionType , eSPI-InterfaceFunctionType , xSPI-InterfaceFunctionType , SPD5118Hub-InterfaceFunctionType .

4.7.3.10.42.1. SI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/SI			
diagram				
type	SI-InterfaceFunctionType , SI-StandardTerminalNameAssignmentType , SI-StandardTerminalMappingType , SI-StandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. TX 2. RX 3. SYSREF 4. Device Clock 5. SYNC-			

For more information about the RBDP Interface, refer to JEDEC Standard JESD204C.1.

4.7.3.10.42.2. SPI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/SPI			
diagram				
type	SPI-InterfaceFunctionType , SPI-StandardTerminalNameAssignmentType , SPI-StandardTerminalMappingType , SPI-StandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. SCK 2. MOSI 3. MISO 4. ~SS			

4.7.3.10.42.3. eSPI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/eSPI												
diagram	<pre> classDiagram class eSPI-InterfaceFunctionType class StandardTerminalNameAssignment class eSPI-StandardTerminalNameAssignmentType class MandatoryMapping class eSPI-MandatoryStandardTerminalMappingType class StandardTerminalName class OptionalMapping class eSPI-OptionalStandardTerminalMappingType eSPI-InterfaceFunctionType "1..∞" -- "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment -- "1..∞" eSPI-StandardTerminalNameAssignmentType eSPI-StandardTerminalNameAssignmentType "6" -- "6" MandatoryMapping eSPI-StandardTerminalNameAssignmentType "6" -- "6" OptionalMapping MandatoryMapping -- "6" eSPI-MandatoryStandardTerminalMappingType eSPI-MandatoryStandardTerminalMappingType -- "6" StandardTerminalName OptionalMapping -- "6" eSPI-OptionalStandardTerminalMappingType eSPI-OptionalStandardTerminalMappingType -- "6" StandardTerminalName </pre>												
type	eSPI-InterfaceFunctionType, eSPI-StandardTerminalNameAssignmentType, eSPI-MandatoryStandardTerminalMappingType, eSPI-MandatoryStandardTerminalNameType, eSPI-OptionalStandardTerminalMappingType, eSPI-OptionalStandardTerminalNameType.												
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. eSPI Reset#</td> <td>2. Chip Select#</td> <td>3. Serial Clock</td> <td>4. Alert#</td> </tr> <tr> <td>5. I/O[0]</td> <td></td> <td></td> <td></td> </tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. I/O[1]</td> <td>2. I/O[2]</td> <td>3. I/O[3]</td> <td></td> </tr> </table>	1. eSPI Reset#	2. Chip Select#	3. Serial Clock	4. Alert#	5. I/O[0]				1. I/O[1]	2. I/O[2]	3. I/O[3]	
1. eSPI Reset#	2. Chip Select#	3. Serial Clock	4. Alert#										
5. I/O[0]													
1. I/O[1]	2. I/O[2]	3. I/O[3]											

For more information about the eSPI Interface, refer to INTEL Specification 327432-004 Enhanced Serial Peripheral Interface (eSPI) Revision 1.0.

4.7.3.10.42.4. xSPI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/xSPI																
diagram	<pre> classDiagram class xSPI-InterfaceFunctionType class StandardTerminalNameAssignment class xSPI-StandardTerminalNameAssignmentType class Mapping class xSPI-StandardTerminalMappingType class StandardTerminalName xSPI-InterfaceFunctionType "1..∞" -- "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment -- "1..∞" xSPI-StandardTerminalNameAssignmentType xSPI-StandardTerminalNameAssignmentType "16" -- "16" Mapping Mapping -- "16" xSPI-StandardTerminalMappingType xSPI-StandardTerminalMappingType -- "16" StandardTerminalName </pre>																
type	xSPI-InterfaceFunctionType, xSPI-StandardTerminalNameAssignmentType, xSPI-StandardTerminalMappingType, xSPI-StandardTerminalNameType.																
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. CS0#</td> <td>2. CS1#</td> <td>3. CK</td> <td>4. IO[0]</td> </tr> <tr> <td>5. IO[1]</td> <td>6. IO[2]</td> <td>7. IO[3]</td> <td>8. IO[4]</td> </tr> <tr> <td>9. IO[5]</td> <td>10. IO[6]</td> <td>11. IO[7]</td> <td>12. DS</td> </tr> <tr> <td>13. VDD</td> <td>14. VDDQ</td> <td>15. VSS</td> <td>16. VSSQ</td> </tr> </table>	1. CS0#	2. CS1#	3. CK	4. IO[0]	5. IO[1]	6. IO[2]	7. IO[3]	8. IO[4]	9. IO[5]	10. IO[6]	11. IO[7]	12. DS	13. VDD	14. VDDQ	15. VSS	16. VSSQ
1. CS0#	2. CS1#	3. CK	4. IO[0]														
5. IO[1]	6. IO[2]	7. IO[3]	8. IO[4]														
9. IO[5]	10. IO[6]	11. IO[7]	12. DS														
13. VDD	14. VDDQ	15. VSS	16. VSSQ														

For more information about the RBDP Interface, refer to JEDEC Standard JESD251C.

4.7.3.10.42.5. SPD5118 Hub

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/SPD5118Hub								
diagram	<pre> classDiagram class SPD5118Hub { <<SPD5118Hub-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<SPD5118Hub-StandardTerminalNameAssignmentType>> } class Mapping { <<SPD5118Hub-StandardTerminalMappingType>> } class StandardTerminalName { <<SPD5118Hub-StandardTerminalNameType>> } SPD5118Hub "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>								
type	SPD5118Hub-InterfaceFunctionType, SPD5118Hub-StandardTerminalNameAssignmentType, SPD5118Hub-StandardTerminalMappingType, SPD5118Hub-StandardTerminalNameType.								
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. 1.8VVDDSPD</td> <td>2. VDDIO</td> <td>3. HSCL</td> <td>4. HSDA</td> </tr> <tr> <td>5. LSCL</td> <td>6. LSDA</td> <td></td> <td></td> </tr> </table>	1. 1.8VVDDSPD	2. VDDIO	3. HSCL	4. HSDA	5. LSCL	6. LSDA		
1. 1.8VVDDSPD	2. VDDIO	3. HSCL	4. HSDA						
5. LSCL	6. LSDA								

For more information about the RBDP Interface, refer to JEDEC Standard JESD300-5B.

4.7.3.10.43. SLIMbus

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SLIMbus												
diagram	<pre> classDiagram class SLIMbus { <<SLIMbus-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<SLIMbus-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<SLIMbus-MandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<SLIMbus-MandatoryStandardTerminalNameType>> } SLIMbus "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping MandatoryMapping "2..>" StandardTerminalName </pre> <p>OptionalMapping is also shown with a 0..7 multiplicity.</p>												
type	SLIMbus-InterfaceFunctionType, SLIMbus-StandardTerminalNameAssignmentType, SLIMbus-MandatoryStandardTerminalMappingType, SLIMbus-MandatoryStandardTerminalNameType, SLIMbus-OptionalStandardTerminalMappingType, SLIMbus-OptionalStandardTerminalNameType.												
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. CLK</td> <td>2. DP0</td> <td></td> <td></td> </tr> </table> OptionalMapping/StandardTerminalName <table border="1"> <tr> <td>1. DP[1]</td> <td>2. DP[2]</td> <td>3. DP[3]</td> <td>4. DP[4]</td> </tr> <tr> <td>5. DP[5]</td> <td>6. DP[6]</td> <td>7. DP[7]</td> <td></td> </tr> </table>	1. CLK	2. DP0			1. DP[1]	2. DP[2]	3. DP[3]	4. DP[4]	5. DP[5]	6. DP[6]	7. DP[7]	
1. CLK	2. DP0												
1. DP[1]	2. DP[2]	3. DP[3]	4. DP[4]										
5. DP[5]	6. DP[6]	7. DP[7]											

For more information about the SLIMbus Interface, refer to the MIPI Alliance standard Specification for Serial Low-Power Inter-Chip Media Bus (SLIMbus) Version 2.0.

4.7.3.10.44. SMB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/SMB	
diagram		
type	SMB-InterfaceFunctionType, SMB-StandardTerminalNameAssignmentType, SMB-StandardTerminalMappingType, SMB-StandardTerminalNameType.	
list of enumerate values	Mapping/StandardTerminalName 1. SMBCLK 2. SMBDAT	

For more information about the SMB Interfaces, refer to the System Management Interface Forum SMI for System Management Bus (SMBus) Specification Version 3.0.

4.7.3.10.45. SoundWire Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/SoundWire	
diagram		
type	SoundWire-InterfaceFunctionType, SoundWire -StandardTerminalNameAssignmentType, SoundWire - StandardTerminalMappingType, SoundWire -StandardTerminalNameType.	
list of enumerate values	Mapping/StandardTerminalName 1. Clock 2. Data	

For more information about the SoundWire Interface, refer to the MIPI Alliance standard Specification for SoundWire Version 1.1.

4.7.3.10.46. SPMI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SPMI			
diagram				
type	SPMI-InterfaceFunctionType, SPMI-StandardTerminalNameAssignmentType, SPMI-StandardTerminalMappingType, SPMI-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	1. SCLK	2. SDATA		

For more information about the SPMI Interfaces, refer to the MIPI Alliance standard System Power Management Interface V2.0.

4.7.3.10.47. UART

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UART																					
diagram																						
type	UART-InterfaceFunctionType, UART-StandardTerminalNameAssignmentType, UART-MandatoryStandardTerminalMappingType, UART-MandatoryStandardTerminalNameType, UART-OptionalStandardTerminalMappingType, UART-OptionalStandardTerminalNameType																					
list of enumerate values	<table border="1"> <tr> <td colspan="2">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. RX</td><td>2. TX</td> <td></td><td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. CTS</td><td>2. RTS</td><td>3. DSR</td><td>4. RI</td> </tr> <tr> <td>5. DCD</td><td>6. DTR</td><td></td><td></td> </tr> </table>				MandatoryMapping/StandardTerminalName		1. RX	2. TX			OptionalMapping/StandardTerminalName				1. CTS	2. RTS	3. DSR	4. RI	5. DCD	6. DTR		
MandatoryMapping/StandardTerminalName																						
1. RX	2. TX																					
OptionalMapping/StandardTerminalName																						
1. CTS	2. RTS	3. DSR	4. RI																			
5. DCD	6. DTR																					

4.7.3.10.48. UCIe

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UCle-InterfaceFunction
diagram	<pre>graph TD; UCle[UCle
type UCle-InterfaceFunctionType] --> UCleIF[UCle-InterfaceFunctionType]; UCleIF --> UCleAdvanced[UCle-Advanced
type UCle-AdvancedType]; UCleIF --> UCleStandardX16[UCleStandard_x16
type UCleStandard_x16Type]; UCleStandardX16 --> UCleStandardX32[UCleStandard_x32
type UCleStandard_x32Type];</pre>
type	UCle-InterfaceFunctionType , UCle-AdvancedType , UCleStandard_x16Type , UCleStandard_x32Type .

For more information about the UCIe Interfaces, refer to the UCIe Specification Universal Chiplet Interconnect Express (UCIe) Specification Revision 1.0

4.7.3.10.48.1. UCIe - Advanced

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UCIe-InterfaceFunction/UCIe-Advanced																																																																																																																																														
diagram																																																																																																																																															
type	UCIe-AdvancedType, UCIe-AdvancedStandardTerminalNameAssignmentType, UCIe-AdvancedStandardTerminalMappingType, UCIe-AdvancedStandardTerminalNameType.																																																																																																																																														
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. TXDATA[0]</td><td>2. TXDATA[1]</td><td>3. TXDATA[2]</td><td>4. TXDATA[3]</td></tr> <tr><td>5. TXDATA[4]</td><td>6. TXDATA[5]</td><td>7. TXDATA[6]</td><td>8. TXDATA[7]</td></tr> <tr><td>9. TXDATA[8]</td><td>10. TXDATA[9]</td><td>11. TXDATA[10]</td><td>12. TXDATA[11]</td></tr> <tr><td>13. TXDATA[12]</td><td>14. TXDATA[13]</td><td>15. TXDATA[14]</td><td>16. TXDATA[15]</td></tr> <tr><td>17. TXDATA[16]</td><td>18. TXDATA[17]</td><td>19. TXDATA[18]</td><td>20. TXDATA[19]</td></tr> <tr><td>21. TXDATA[20]</td><td>22. TXDATA[21]</td><td>23. TXDATA[22]</td><td>24. TXDATA[23]</td></tr> <tr><td>25. TXDATA[24]</td><td>26. TXDATA[25]</td><td>27. TXDATA[26]</td><td>28. TXDATA[27]</td></tr> <tr><td>29. TXDATA[28]</td><td>30. TXDATA[29]</td><td>31. TXDATA[30]</td><td>32. TXDATA[31]</td></tr> <tr><td>33. TXDATA[32]</td><td>34. TXDATA[33]</td><td>35. TXDATA[34]</td><td>36. TXDATA[35]</td></tr> <tr><td>37. TXDATA[36]</td><td>38. TXDATA[37]</td><td>39. TXDATA[38]</td><td>40. TXDATA[39]</td></tr> <tr><td>41. TXDATA[40]</td><td>42. TXDATA[41]</td><td>43. TXDATA[42]</td><td>44. TXDATA[43]</td></tr> <tr><td>45. TXDATA[44]</td><td>46. TXDATA[45]</td><td>47. TXDATA[46]</td><td>48. TXDATA[47]</td></tr> <tr><td>49. TXDATA[48]</td><td>50. TXDATA[49]</td><td>51. TXDATA[50]</td><td>52. TXDATA[51]</td></tr> <tr><td>53. TXDATA[52]</td><td>54. TXDATA[53]</td><td>55. TXDATA[54]</td><td>56. TXDATA[55]</td></tr> <tr><td>57. TXDATA[56]</td><td>58. TXDATA[57]</td><td>59. TXDATA[58]</td><td>60. TXDATA[59]</td></tr> <tr><td>61. TXDATA[60]</td><td>62. TXDATA[61]</td><td>63. TXDATA[62]</td><td>64. TXDATA[63]</td></tr> <tr><td>65. TXVLD</td><td>66. TXTRK</td><td>67. TXCKP</td><td>68. TXCKN</td></tr> <tr><td>69. TXCKRD</td><td>70. TXRD0</td><td>71. TXRD1</td><td>72. TXRD2</td></tr> <tr><td>73. RXRD3</td><td>74. RXDATA[0]</td><td>75. RXDATA[1]</td><td>76. RXDATA[2]</td></tr> <tr><td>77. RXDATA[3]</td><td>78. RXDATA[4]</td><td>79. RXDATA[5]</td><td>80. RXDATA[6]</td></tr> <tr><td>81. RXDATA[7]</td><td>82. RXDATA[8]</td><td>83. RXDATA[9]</td><td>84. RXDATA[10]</td></tr> <tr><td>85. RXDATA[11]</td><td>86. RXDATA[12]</td><td>87. RXDATA[13]</td><td>88. RXDATA[14]</td></tr> <tr><td>89. RXDATA[15]</td><td>90. RXDATA[16]</td><td>91. RXDATA[17]</td><td>92. RXDATA[18]</td></tr> <tr><td>93. RXDATA[19]</td><td>94. RXDATA[20]</td><td>95. RXDATA[21]</td><td>96. RXDATA[22]</td></tr> <tr><td>97. RXDATA[23]</td><td>98. RXDATA[24]</td><td>99. RXDATA[25]</td><td>100. RXDATA[26]</td></tr> <tr><td>101. RXDATA[27]</td><td>102. RXDATA[28]</td><td>103. RXDATA[29]</td><td>104. RXDATA[30]</td></tr> <tr><td>105. RXDATA[31]</td><td>106. RXDATA[32]</td><td>107. RXDATA[33]</td><td>108. RXDATA[34]</td></tr> <tr><td>109. RXDATA[35]</td><td>110. RXDATA[36]</td><td>111. RXDATA[37]</td><td>112. RXDATA[38]</td></tr> <tr><td>113. RXDATA[39]</td><td>114. RXDATA[40]</td><td>115. RXDATA[41]</td><td>116. RXDATA[42]</td></tr> <tr><td>117. RXDATA[43]</td><td>118. RXDATA[44]</td><td>119. RXDATA[45]</td><td>120. RXDATA[46]</td></tr> <tr><td>121. RXDATA[47]</td><td>122. RXDATA[48]</td><td>123. RXDATA[49]</td><td>124. RXDATA[50]</td></tr> <tr><td>125. RXDATA[51]</td><td>126. RXDATA[52]</td><td>127. RXDATA[53]</td><td>128. RXDATA[54]</td></tr> <tr><td>129. RXDATA[55]</td><td>130. RXDATA[56]</td><td>131. RXDATA[57]</td><td>132. RXDATA[58]</td></tr> <tr><td>133. RXDATA[59]</td><td>134. RXDATA[60]</td><td>135. RXDATA[61]</td><td>136. RXDATA[62]</td></tr> <tr><td>137. RXDATA[63]</td><td>138. RXVLD</td><td>139. RXTRK</td><td>140. RXCKP</td></tr> </table>			1. TXDATA[0]	2. TXDATA[1]	3. TXDATA[2]	4. TXDATA[3]	5. TXDATA[4]	6. TXDATA[5]	7. TXDATA[6]	8. TXDATA[7]	9. TXDATA[8]	10. TXDATA[9]	11. TXDATA[10]	12. TXDATA[11]	13. TXDATA[12]	14. TXDATA[13]	15. TXDATA[14]	16. TXDATA[15]	17. TXDATA[16]	18. TXDATA[17]	19. TXDATA[18]	20. TXDATA[19]	21. TXDATA[20]	22. TXDATA[21]	23. TXDATA[22]	24. TXDATA[23]	25. TXDATA[24]	26. TXDATA[25]	27. TXDATA[26]	28. TXDATA[27]	29. TXDATA[28]	30. TXDATA[29]	31. TXDATA[30]	32. TXDATA[31]	33. TXDATA[32]	34. TXDATA[33]	35. TXDATA[34]	36. TXDATA[35]	37. TXDATA[36]	38. TXDATA[37]	39. TXDATA[38]	40. TXDATA[39]	41. TXDATA[40]	42. TXDATA[41]	43. TXDATA[42]	44. TXDATA[43]	45. TXDATA[44]	46. TXDATA[45]	47. TXDATA[46]	48. TXDATA[47]	49. TXDATA[48]	50. TXDATA[49]	51. TXDATA[50]	52. TXDATA[51]	53. TXDATA[52]	54. TXDATA[53]	55. TXDATA[54]	56. TXDATA[55]	57. TXDATA[56]	58. TXDATA[57]	59. TXDATA[58]	60. TXDATA[59]	61. TXDATA[60]	62. TXDATA[61]	63. TXDATA[62]	64. TXDATA[63]	65. TXVLD	66. TXTRK	67. TXCKP	68. TXCKN	69. TXCKRD	70. TXRD0	71. TXRD1	72. TXRD2	73. RXRD3	74. RXDATA[0]	75. RXDATA[1]	76. RXDATA[2]	77. RXDATA[3]	78. RXDATA[4]	79. RXDATA[5]	80. RXDATA[6]	81. RXDATA[7]	82. RXDATA[8]	83. RXDATA[9]	84. RXDATA[10]	85. RXDATA[11]	86. RXDATA[12]	87. RXDATA[13]	88. RXDATA[14]	89. RXDATA[15]	90. RXDATA[16]	91. RXDATA[17]	92. RXDATA[18]	93. RXDATA[19]	94. RXDATA[20]	95. RXDATA[21]	96. RXDATA[22]	97. RXDATA[23]	98. RXDATA[24]	99. RXDATA[25]	100. RXDATA[26]	101. RXDATA[27]	102. RXDATA[28]	103. RXDATA[29]	104. RXDATA[30]	105. RXDATA[31]	106. RXDATA[32]	107. RXDATA[33]	108. RXDATA[34]	109. RXDATA[35]	110. RXDATA[36]	111. RXDATA[37]	112. RXDATA[38]	113. RXDATA[39]	114. RXDATA[40]	115. RXDATA[41]	116. RXDATA[42]	117. RXDATA[43]	118. RXDATA[44]	119. RXDATA[45]	120. RXDATA[46]	121. RXDATA[47]	122. RXDATA[48]	123. RXDATA[49]	124. RXDATA[50]	125. RXDATA[51]	126. RXDATA[52]	127. RXDATA[53]	128. RXDATA[54]	129. RXDATA[55]	130. RXDATA[56]	131. RXDATA[57]	132. RXDATA[58]	133. RXDATA[59]	134. RXDATA[60]	135. RXDATA[61]	136. RXDATA[62]	137. RXDATA[63]	138. RXVLD	139. RXTRK	140. RXCKP
1. TXDATA[0]	2. TXDATA[1]	3. TXDATA[2]	4. TXDATA[3]																																																																																																																																												
5. TXDATA[4]	6. TXDATA[5]	7. TXDATA[6]	8. TXDATA[7]																																																																																																																																												
9. TXDATA[8]	10. TXDATA[9]	11. TXDATA[10]	12. TXDATA[11]																																																																																																																																												
13. TXDATA[12]	14. TXDATA[13]	15. TXDATA[14]	16. TXDATA[15]																																																																																																																																												
17. TXDATA[16]	18. TXDATA[17]	19. TXDATA[18]	20. TXDATA[19]																																																																																																																																												
21. TXDATA[20]	22. TXDATA[21]	23. TXDATA[22]	24. TXDATA[23]																																																																																																																																												
25. TXDATA[24]	26. TXDATA[25]	27. TXDATA[26]	28. TXDATA[27]																																																																																																																																												
29. TXDATA[28]	30. TXDATA[29]	31. TXDATA[30]	32. TXDATA[31]																																																																																																																																												
33. TXDATA[32]	34. TXDATA[33]	35. TXDATA[34]	36. TXDATA[35]																																																																																																																																												
37. TXDATA[36]	38. TXDATA[37]	39. TXDATA[38]	40. TXDATA[39]																																																																																																																																												
41. TXDATA[40]	42. TXDATA[41]	43. TXDATA[42]	44. TXDATA[43]																																																																																																																																												
45. TXDATA[44]	46. TXDATA[45]	47. TXDATA[46]	48. TXDATA[47]																																																																																																																																												
49. TXDATA[48]	50. TXDATA[49]	51. TXDATA[50]	52. TXDATA[51]																																																																																																																																												
53. TXDATA[52]	54. TXDATA[53]	55. TXDATA[54]	56. TXDATA[55]																																																																																																																																												
57. TXDATA[56]	58. TXDATA[57]	59. TXDATA[58]	60. TXDATA[59]																																																																																																																																												
61. TXDATA[60]	62. TXDATA[61]	63. TXDATA[62]	64. TXDATA[63]																																																																																																																																												
65. TXVLD	66. TXTRK	67. TXCKP	68. TXCKN																																																																																																																																												
69. TXCKRD	70. TXRD0	71. TXRD1	72. TXRD2																																																																																																																																												
73. RXRD3	74. RXDATA[0]	75. RXDATA[1]	76. RXDATA[2]																																																																																																																																												
77. RXDATA[3]	78. RXDATA[4]	79. RXDATA[5]	80. RXDATA[6]																																																																																																																																												
81. RXDATA[7]	82. RXDATA[8]	83. RXDATA[9]	84. RXDATA[10]																																																																																																																																												
85. RXDATA[11]	86. RXDATA[12]	87. RXDATA[13]	88. RXDATA[14]																																																																																																																																												
89. RXDATA[15]	90. RXDATA[16]	91. RXDATA[17]	92. RXDATA[18]																																																																																																																																												
93. RXDATA[19]	94. RXDATA[20]	95. RXDATA[21]	96. RXDATA[22]																																																																																																																																												
97. RXDATA[23]	98. RXDATA[24]	99. RXDATA[25]	100. RXDATA[26]																																																																																																																																												
101. RXDATA[27]	102. RXDATA[28]	103. RXDATA[29]	104. RXDATA[30]																																																																																																																																												
105. RXDATA[31]	106. RXDATA[32]	107. RXDATA[33]	108. RXDATA[34]																																																																																																																																												
109. RXDATA[35]	110. RXDATA[36]	111. RXDATA[37]	112. RXDATA[38]																																																																																																																																												
113. RXDATA[39]	114. RXDATA[40]	115. RXDATA[41]	116. RXDATA[42]																																																																																																																																												
117. RXDATA[43]	118. RXDATA[44]	119. RXDATA[45]	120. RXDATA[46]																																																																																																																																												
121. RXDATA[47]	122. RXDATA[48]	123. RXDATA[49]	124. RXDATA[50]																																																																																																																																												
125. RXDATA[51]	126. RXDATA[52]	127. RXDATA[53]	128. RXDATA[54]																																																																																																																																												
129. RXDATA[55]	130. RXDATA[56]	131. RXDATA[57]	132. RXDATA[58]																																																																																																																																												
133. RXDATA[59]	134. RXDATA[60]	135. RXDATA[61]	136. RXDATA[62]																																																																																																																																												
137. RXDATA[63]	138. RXVLD	139. RXTRK	140. RXCKP																																																																																																																																												

4.7.3.10.48.1 UCIe - Advanced (cont'd)

list of enumerate values (cont.)	141. RXCKN	142. RXRD0	143. RXRD1	144. RXRD2
	145. RXRD3	146. RXCKRD	147. TXDATASB	148. RXDATASB
	149. TXCKSB	150. RXCKSB	151. TXDATASBRD	152. RXDATASBRD
	153. TXCKSBRD	154. RXCKSBRD	155. VSS	156. VCCIO
	157. VCCAON			

4.7.3.10.48.2. UCIeStandard_x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UCIe-InterfaceFunction/UCIeStandard_x16																																																			
diagram																																																				
type	UCIeStandard_x16Type , UCIeStandard_x16StandardTerminalNameAssignmentType , UCIeStandard_x16StandardTerminalMappingType , UCIeStandard_x16StandardTerminalNameType .																																																			
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. TXDATA[0]</td><td>2. TXDATA[1]</td><td>3. TXDATA[2]</td><td>4. TXDATA[3]</td></tr> <tr><td>5. TXDATA[4]</td><td>6. TXDATA[5]</td><td>7. TXDATA[6]</td><td>8. TXDATA[7]</td></tr> <tr><td>9. TXDATA[8]</td><td>10. TXDATA[9]</td><td>11. TXDATA[10]</td><td>12. TXDATA[11]</td></tr> <tr><td>13. TXDATA[12]</td><td>14. TXDATA[13]</td><td>15. TXDATA[14]</td><td>16. TXDATA[15]</td></tr> <tr><td>17. TXVLD</td><td>18. TXTRK</td><td>19. TXCP</td><td>20. TXCKN</td></tr> <tr><td>21. RXDATA[0]</td><td>22. RXDATA[1]</td><td>23. RXDATA[2]</td><td>24. RXDATA[3]</td></tr> <tr><td>25. RXDATA[4]</td><td>26. RXDATA[5]</td><td>27. RXDATA[6]</td><td>28. RXDATA[7]</td></tr> <tr><td>29. RXDATA[8]</td><td>30. RXDATA[9]</td><td>31. RXDATA[10]</td><td>32. RXDATA[11]</td></tr> <tr><td>33. RXDATA[12]</td><td>34. RXDATA[13]</td><td>35. RXDATA[14]</td><td>36. RXDATA[15]</td></tr> <tr><td>37. RXVLD</td><td>38. RXTRK</td><td>39. RXCKP</td><td>40. RXCKN</td></tr> <tr><td>41. TXDATASB</td><td>42. RXDATASB</td><td>43. TXCKSB</td><td>44. RXCKSB</td></tr> <tr><td>45. VSS</td><td>46. VCCIO</td><td>47. VCCAON</td><td></td></tr> </table>				1. TXDATA[0]	2. TXDATA[1]	3. TXDATA[2]	4. TXDATA[3]	5. TXDATA[4]	6. TXDATA[5]	7. TXDATA[6]	8. TXDATA[7]	9. TXDATA[8]	10. TXDATA[9]	11. TXDATA[10]	12. TXDATA[11]	13. TXDATA[12]	14. TXDATA[13]	15. TXDATA[14]	16. TXDATA[15]	17. TXVLD	18. TXTRK	19. TXCP	20. TXCKN	21. RXDATA[0]	22. RXDATA[1]	23. RXDATA[2]	24. RXDATA[3]	25. RXDATA[4]	26. RXDATA[5]	27. RXDATA[6]	28. RXDATA[7]	29. RXDATA[8]	30. RXDATA[9]	31. RXDATA[10]	32. RXDATA[11]	33. RXDATA[12]	34. RXDATA[13]	35. RXDATA[14]	36. RXDATA[15]	37. RXVLD	38. RXTRK	39. RXCKP	40. RXCKN	41. TXDATASB	42. RXDATASB	43. TXCKSB	44. RXCKSB	45. VSS	46. VCCIO	47. VCCAON	
1. TXDATA[0]	2. TXDATA[1]	3. TXDATA[2]	4. TXDATA[3]																																																	
5. TXDATA[4]	6. TXDATA[5]	7. TXDATA[6]	8. TXDATA[7]																																																	
9. TXDATA[8]	10. TXDATA[9]	11. TXDATA[10]	12. TXDATA[11]																																																	
13. TXDATA[12]	14. TXDATA[13]	15. TXDATA[14]	16. TXDATA[15]																																																	
17. TXVLD	18. TXTRK	19. TXCP	20. TXCKN																																																	
21. RXDATA[0]	22. RXDATA[1]	23. RXDATA[2]	24. RXDATA[3]																																																	
25. RXDATA[4]	26. RXDATA[5]	27. RXDATA[6]	28. RXDATA[7]																																																	
29. RXDATA[8]	30. RXDATA[9]	31. RXDATA[10]	32. RXDATA[11]																																																	
33. RXDATA[12]	34. RXDATA[13]	35. RXDATA[14]	36. RXDATA[15]																																																	
37. RXVLD	38. RXTRK	39. RXCKP	40. RXCKN																																																	
41. TXDATASB	42. RXDATASB	43. TXCKSB	44. RXCKSB																																																	
45. VSS	46. VCCIO	47. VCCAON																																																		

4.7.3.10.48.3. UCIeStandard_x32

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UCIe-InterfaceFunction/ UCIeStandard_x32																																																																																															
diagram	<pre> classDiagram class UCIeStandard_x32Type { <<UCIeStandard_x32Type>> } class StandardTerminalNameAssignment { <<UCIeStandard_x32StandardTerminalNameAssignmentType>> } class Mapping { <<UCIeStandard_x32StandardTerminalMappingType>> } class StandardTerminalName { <<UCIeStandard_x32StandardTerminalNameType>> } UCIeStandard_x32Type "1..∞" -- "91" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" -- "91" Mapping Mapping "91" -- "91" StandardTerminalName </pre>																																																																																															
type	UCIeStandard_x32Type , UCIeStandard_x32StandardTerminalNameAssignmentType , UCIeStandard_x32StandardTerminalMappingType , UCIeStandard_x32StandardTerminalNameType .																																																																																															
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr><td>1. M1RXCKN</td><td>2. M1RXCKP</td><td>3. M1RXCKSB</td><td>4. M1RXDATA[0]</td></tr> <tr><td>5. M1RXDATA[1]</td><td>6. M1RXDATA[2]</td><td>7. M1RXDATA[3]</td><td>8. M1RXDATA[4]</td></tr> <tr><td>9. M1RXDATA[5]</td><td>10. M1RXDATA[6]</td><td>11. M1RXDATA[7]</td><td>12. M1RXDATA[8]</td></tr> <tr><td>13. M1RXDATA[9]</td><td>14. M1RXDATA[10]</td><td>15. M1RXDATA[11]</td><td>16. M1RXDATA[12]</td></tr> <tr><td>17. M1RXDATA[13]</td><td>18. M1RXDATA[14]</td><td>19. M1RXDATA[15]</td><td>20. M1RXDATASB</td></tr> <tr><td>21. M1RXTRK</td><td>22. M1RXVLD</td><td>23. M1TXCKN</td><td>24. M1TXCKP</td></tr> <tr><td>25. M1TXCKSB</td><td>26. M1TXDATA[0]</td><td>27. M1TXDATA[1]</td><td>28. M1TXDATA[2]</td></tr> <tr><td>29. M1TXDATA[3]</td><td>30. M1TXDATA[4]</td><td>31. M1TXDATA[5]</td><td>32. M1TXDATA[6]</td></tr> <tr><td>33. M1TXDATA[7]</td><td>34. M1TXDATA[8]</td><td>35. M1TXDATA[9]</td><td>36. M1TXDATA[10]</td></tr> <tr><td>37. M1TXDATA[11]</td><td>38. M1TXDATA[12]</td><td>39. M1TXDATA[13]</td><td>40. M1TXDATA[14]</td></tr> <tr><td>41. M1TXDATA[15]</td><td>42. M1TXDATASB</td><td>43. M1TXTRK</td><td>44. M1TXVLD</td></tr> <tr><td>45. M2RXCKN</td><td>46. M2RXCKP</td><td>47. M2RXCKSB</td><td>48. M2RXDATA[0]</td></tr> <tr><td>49. M2RXDATA[1]</td><td>50. M2RXDATA[2]</td><td>51. M2RXDATA[3]</td><td>52. M2RXDATA[4]</td></tr> <tr><td>53. M2RXDATA[5]</td><td>54. M2RXDATA[6]</td><td>55. M2RXDATA[7]</td><td>56. M2RXDATA[8]</td></tr> <tr><td>57. M2RXDATA[9]</td><td>58. M2RXDATA[10]</td><td>59. M2RXDATA[11]</td><td>60. M2RXDATA[12]</td></tr> <tr><td>61. M2RXDATA[13]</td><td>62. M2RXDATA[14]</td><td>63. M2RXDATA[15]</td><td>64. M2RXDATASB</td></tr> <tr><td>65. M2RXTRK</td><td>66. M2RXVLD</td><td>67. M2TXCKN</td><td>68. M2TXCKP</td></tr> <tr><td>69. M2TXCKSB</td><td>70. M2TXDATA[0]</td><td>71. M2TXDATA[1]</td><td>72. M2TXDATA[2]</td></tr> <tr><td>73. M2TXDATA[3]</td><td>74. M2TXDATA[4]</td><td>75. M2TXDATA[5]</td><td>76. M2TXDATA[6]</td></tr> <tr><td>77. M2TXDATA[7]</td><td>78. M2TXDATA[8]</td><td>79. M2TXDATA[9]</td><td>80. M2TXDATA[10]</td></tr> <tr><td>81. M2TXDATA[11]</td><td>82. M2TXDATA[12]</td><td>83. M2TXDATA[13]</td><td>84. M2TXDATA[14]</td></tr> <tr><td>85. M2TXDATA[15]</td><td>86. M2TXDATASB</td><td>87. M2TXTRK</td><td>88. M2TXVLD</td></tr> <tr><td>89. VCCAON</td><td>90. VCCIO</td><td>91. VSS</td><td></td></tr> </table>				1. M1RXCKN	2. M1RXCKP	3. M1RXCKSB	4. M1RXDATA[0]	5. M1RXDATA[1]	6. M1RXDATA[2]	7. M1RXDATA[3]	8. M1RXDATA[4]	9. M1RXDATA[5]	10. M1RXDATA[6]	11. M1RXDATA[7]	12. M1RXDATA[8]	13. M1RXDATA[9]	14. M1RXDATA[10]	15. M1RXDATA[11]	16. M1RXDATA[12]	17. M1RXDATA[13]	18. M1RXDATA[14]	19. M1RXDATA[15]	20. M1RXDATASB	21. M1RXTRK	22. M1RXVLD	23. M1TXCKN	24. M1TXCKP	25. M1TXCKSB	26. M1TXDATA[0]	27. M1TXDATA[1]	28. M1TXDATA[2]	29. M1TXDATA[3]	30. M1TXDATA[4]	31. M1TXDATA[5]	32. M1TXDATA[6]	33. M1TXDATA[7]	34. M1TXDATA[8]	35. M1TXDATA[9]	36. M1TXDATA[10]	37. M1TXDATA[11]	38. M1TXDATA[12]	39. M1TXDATA[13]	40. M1TXDATA[14]	41. M1TXDATA[15]	42. M1TXDATASB	43. M1TXTRK	44. M1TXVLD	45. M2RXCKN	46. M2RXCKP	47. M2RXCKSB	48. M2RXDATA[0]	49. M2RXDATA[1]	50. M2RXDATA[2]	51. M2RXDATA[3]	52. M2RXDATA[4]	53. M2RXDATA[5]	54. M2RXDATA[6]	55. M2RXDATA[7]	56. M2RXDATA[8]	57. M2RXDATA[9]	58. M2RXDATA[10]	59. M2RXDATA[11]	60. M2RXDATA[12]	61. M2RXDATA[13]	62. M2RXDATA[14]	63. M2RXDATA[15]	64. M2RXDATASB	65. M2RXTRK	66. M2RXVLD	67. M2TXCKN	68. M2TXCKP	69. M2TXCKSB	70. M2TXDATA[0]	71. M2TXDATA[1]	72. M2TXDATA[2]	73. M2TXDATA[3]	74. M2TXDATA[4]	75. M2TXDATA[5]	76. M2TXDATA[6]	77. M2TXDATA[7]	78. M2TXDATA[8]	79. M2TXDATA[9]	80. M2TXDATA[10]	81. M2TXDATA[11]	82. M2TXDATA[12]	83. M2TXDATA[13]	84. M2TXDATA[14]	85. M2TXDATA[15]	86. M2TXDATASB	87. M2TXTRK	88. M2TXVLD	89. VCCAON	90. VCCIO	91. VSS	
1. M1RXCKN	2. M1RXCKP	3. M1RXCKSB	4. M1RXDATA[0]																																																																																													
5. M1RXDATA[1]	6. M1RXDATA[2]	7. M1RXDATA[3]	8. M1RXDATA[4]																																																																																													
9. M1RXDATA[5]	10. M1RXDATA[6]	11. M1RXDATA[7]	12. M1RXDATA[8]																																																																																													
13. M1RXDATA[9]	14. M1RXDATA[10]	15. M1RXDATA[11]	16. M1RXDATA[12]																																																																																													
17. M1RXDATA[13]	18. M1RXDATA[14]	19. M1RXDATA[15]	20. M1RXDATASB																																																																																													
21. M1RXTRK	22. M1RXVLD	23. M1TXCKN	24. M1TXCKP																																																																																													
25. M1TXCKSB	26. M1TXDATA[0]	27. M1TXDATA[1]	28. M1TXDATA[2]																																																																																													
29. M1TXDATA[3]	30. M1TXDATA[4]	31. M1TXDATA[5]	32. M1TXDATA[6]																																																																																													
33. M1TXDATA[7]	34. M1TXDATA[8]	35. M1TXDATA[9]	36. M1TXDATA[10]																																																																																													
37. M1TXDATA[11]	38. M1TXDATA[12]	39. M1TXDATA[13]	40. M1TXDATA[14]																																																																																													
41. M1TXDATA[15]	42. M1TXDATASB	43. M1TXTRK	44. M1TXVLD																																																																																													
45. M2RXCKN	46. M2RXCKP	47. M2RXCKSB	48. M2RXDATA[0]																																																																																													
49. M2RXDATA[1]	50. M2RXDATA[2]	51. M2RXDATA[3]	52. M2RXDATA[4]																																																																																													
53. M2RXDATA[5]	54. M2RXDATA[6]	55. M2RXDATA[7]	56. M2RXDATA[8]																																																																																													
57. M2RXDATA[9]	58. M2RXDATA[10]	59. M2RXDATA[11]	60. M2RXDATA[12]																																																																																													
61. M2RXDATA[13]	62. M2RXDATA[14]	63. M2RXDATA[15]	64. M2RXDATASB																																																																																													
65. M2RXTRK	66. M2RXVLD	67. M2TXCKN	68. M2TXCKP																																																																																													
69. M2TXCKSB	70. M2TXDATA[0]	71. M2TXDATA[1]	72. M2TXDATA[2]																																																																																													
73. M2TXDATA[3]	74. M2TXDATA[4]	75. M2TXDATA[5]	76. M2TXDATA[6]																																																																																													
77. M2TXDATA[7]	78. M2TXDATA[8]	79. M2TXDATA[9]	80. M2TXDATA[10]																																																																																													
81. M2TXDATA[11]	82. M2TXDATA[12]	83. M2TXDATA[13]	84. M2TXDATA[14]																																																																																													
85. M2TXDATA[15]	86. M2TXDATASB	87. M2TXTRK	88. M2TXVLD																																																																																													
89. VCCAON	90. VCCIO	91. VSS																																																																																														

4.7.3.10.49. UniPro

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ UniPro																				
diagram	<pre> classDiagram UniPro < -- UniPro-InterfaceFunctionType StandardTerminalNameAssignment < -- UniPro-StandardTerminalNameAssignmentType MandatoryMapping < -- UniPro-MandatoryStandardTerminalMappingType StandardTerminalName < -- UniPro-MandatoryStandardTerminalNameType OptionalMapping < -- UniPro-OptionalStandardTerminalMappingType StandardTerminalName < -- UniPro-OptionalStandardTerminalNameType UniPro "1..2" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..2" --> MandatoryMapping StandardTerminalNameAssignment "0..8" --> OptionalMapping MandatoryMapping "2" --> StandardTerminalName OptionalMapping "0..8" --> StandardTerminalName </pre> <p>constraints</p>																				
type	UniPro-InterfaceFunctionType , UniPro-StandardTerminalNameAssignmentType , UniPro-MandatoryStandardTerminalMappingType , UniPro-MandatoryStandardTerminalNameType , UniPro-OptionalStandardTerminalMappingType , UniPro-OptionalStandardTerminalNameType																				
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. T_SAP</td> <td>2. DME_SAP</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. N_SAP</td> <td>2. DL_SAP</td> <td>3. PA_SAP</td> <td>4. PHY_SAP</td> </tr> <tr> <td>5. T_LM_SAP</td> <td>6. N_LM_SAP</td> <td>7. DL_LM_SAP</td> <td>8. PA_LM_SAP</td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. T_SAP	2. DME_SAP			OptionalMapping/StandardTerminalName				1. N_SAP	2. DL_SAP	3. PA_SAP	4. PHY_SAP	5. T_LM_SAP	6. N_LM_SAP	7. DL_LM_SAP	8. PA_LM_SAP
MandatoryMapping/StandardTerminalName																					
1. T_SAP	2. DME_SAP																				
OptionalMapping/StandardTerminalName																					
1. N_SAP	2. DL_SAP	3. PA_SAP	4. PHY_SAP																		
5. T_LM_SAP	6. N_LM_SAP	7. DL_LM_SAP	8. PA_LM_SAP																		

For more information about the SPMI Interfaces, refer to the MIPI Alliance standard Specification for Unified Protocol (UniPro) Version 1.8.

4.7.3.10.50. Universal Flash Storage Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UniversalFlashStorage
diagram	<pre> classDiagram UniversalFlashStorage < -- UniversalFlashStorage-InterfaceFunctionType UFS-SingleChannel < -- UFS-SingleChannel-InterfaceFunctionType UFS-DualChannel < -- UFS-DualChannel-InterfaceFunctionType UFSHCI < -- UFSHCI-InterfaceFunctionType UniversalFlashStorage --> UFS-SingleChannel UniversalFlashStorage --> UFS-DualChannel UniversalFlashStorage --> UFSHCI </pre>
type	UniversalFlashStorage-InterfaceFunctionType , UFS-SingleChannel-InterfaceFunctionType , UFS-DualChannel-InterfaceFunctionType , UFSHCI-InterfaceFunctionType .

4.7.3.10.50.1. UFS – Single Channel

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UniversalFlashStorage/UFS-SingleChannel																								
diagram part 1 of 2	<pre> classDiagram class UFS-SingleChannel { <<UFS-SingleChannel-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<UFS-SingleChannel-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<UFS-SingleChannel-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<UFS-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<UFS-SingleChannel-MandatoryStandardTerminalNameType>> } class UFS-OptionalStandardTerminalNameType UFS-SingleChannel "1..0" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..0" --> MandatoryMapping StandardTerminalNameAssignment "1..0" --> OptionalMapping MandatoryMapping "1..1" --> StandardTerminalName OptionalMapping "0..1" --> StandardTerminalName </pre>																								
type	UFS-SingleChannel-InterfaceFunctionType , UFS-SingleChannel-StandardTerminalNameAssignmentType , UFS-SingleChannel-MandatoryStandardTerminalMappingType , UFS-SingleChannel-MandatoryStandardTerminalNameType , UFS-OptionalStandardTerminalMappingType , UFS-OptionalStandardTerminalNameType																								
list of enumerate values	<p>MandatoryMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. DIN_c</td><td>2. DIN_t</td><td>3. DOUT_c</td><td>4. DOUT_t</td></tr> <tr><td>5. LSS</td><td>6. REF_CLK</td><td>7. RST_n</td><td>8. VCC</td></tr> <tr><td>9. VCCQ</td><td>10. VCCQ2</td><td>11. VDDI</td><td>12. VDDIQ</td></tr> <tr><td>13. VDDIQ2</td><td>14. VSS</td><td></td><td></td></tr> </table> <p>OptionalMapping/StandardTerminalName</p> <table border="1"> <tr><td>1. C+</td><td>2. C-</td><td>3. CPOUT1</td><td>4. CPOUT2</td></tr> <tr><td>5. RZQ1</td><td>6. RZQ2</td><td></td><td></td></tr> </table>	1. DIN_c	2. DIN_t	3. DOUT_c	4. DOUT_t	5. LSS	6. REF_CLK	7. RST_n	8. VCC	9. VCCQ	10. VCCQ2	11. VDDI	12. VDDIQ	13. VDDIQ2	14. VSS			1. C+	2. C-	3. CPOUT1	4. CPOUT2	5. RZQ1	6. RZQ2		
1. DIN_c	2. DIN_t	3. DOUT_c	4. DOUT_t																						
5. LSS	6. REF_CLK	7. RST_n	8. VCC																						
9. VCCQ	10. VCCQ2	11. VDDI	12. VDDIQ																						
13. VDDIQ2	14. VSS																								
1. C+	2. C-	3. CPOUT1	4. CPOUT2																						
5. RZQ1	6. RZQ2																								

For more information about the UFS Interface, refer to the JEDEC standard JESD220E.

4.7.3.10.50.2. UFS – Dual Channel

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UniversalFlashStorage/UFS-DualChannel																																				
diagram part 1 of 2	<pre> classDiagram class UFS-DualChannel-InterfaceFunctionType { <<UFS-DualChannel-InterfaceFunctionType>> } class UFS-DualChannel { <<UFS-DualChannel-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<UFS-DualChannel-StandardTerminalNameAssignmentType>> } class MandatoryMapping { <<UFS-DualChannel-MandatoryStandardTerminalMappingType>> } class OptionalMapping { <<UFS-OptionalStandardTerminalMappingType>> } class StandardTerminalName { <<UFS-DualChannel-MandatoryStandardTerminalNameType>> } class UFS-OptionalStandardTerminalNameType { <<UFS-OptionalStandardTerminalNameType>> } UFS-DualChannel-InterfaceFunctionType "1..x" --> StandardTerminalNameAssignment : UFS-DualChannel-InterfaceFunctionType "1..x" --> MandatoryMapping : UFS-DualChannel-InterfaceFunctionType "0..1" --> OptionalMapping : StandardTerminalNameAssignment "1..x" --> StandardTerminalName : MandatoryMapping "1..x" --> StandardTerminalName : OptionalMapping "0..1" --> UFS-OptionalStandardTerminalNameType : StandardTerminalName "1..x" --> UFS-OptionalStandardTerminalNameType : </pre> <p>The diagram illustrates the UFS-DualChannel-InterfaceFunctionType class, which has three associations: one to StandardTerminalNameAssignment (multiplicity 1..x), one to MandatoryMapping (multiplicity 1..x), and one to OptionalMapping (multiplicity 0..1). The StandardTerminalNameAssignment class has an association to StandardTerminalName (multiplicity 1..x). The MandatoryMapping class also has an association to StandardTerminalName (multiplicity 1..x). The OptionalMapping class has an association to UFS-OptionalStandardTerminalNameType (multiplicity 0..1). The StandardTerminalName class has an association to UFS-OptionalStandardTerminalNameType (multiplicity 1..x).</p>																																				
type	UFS-DualChannel-InterfaceFunctionType , UFS-DualChannel-StandardTerminalNameAssignmentType , UFS-DualChannel-MandatoryStandardTerminalMappingType , UFS-DualChannel-MandatoryStandardTerminalNameType , UFS-OptionalStandardTerminalMappingType , UFS-OptionalStandardTerminalNameType																																				
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. DIN0_c</td><td>2. DIN0_t</td><td>3. DIN1_c</td><td>4. DIN1_t</td></tr> <tr> <td>5. DOUT0_c</td><td>6. DOUT0_t</td><td>7. DOUT1_c</td><td>8. DOUT1_t</td></tr> <tr> <td>9. LSS</td><td>10. REF_CLK</td><td>11. RST_n</td><td>12. VCC</td></tr> <tr> <td>13. VCCQ</td><td>14. VCCQ2</td><td>15. VDDi</td><td>16. VDDiQ</td></tr> <tr> <td>17. VDDiQ2</td><td>18. VSS</td><td></td><td></td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. C+</td><td>2. C-</td><td>3. CPOUT1</td><td>4. CPOUT2</td></tr> <tr> <td>5. RZQ1</td><td>6. RZQ2</td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. DIN0_c	2. DIN0_t	3. DIN1_c	4. DIN1_t	5. DOUT0_c	6. DOUT0_t	7. DOUT1_c	8. DOUT1_t	9. LSS	10. REF_CLK	11. RST_n	12. VCC	13. VCCQ	14. VCCQ2	15. VDDi	16. VDDiQ	17. VDDiQ2	18. VSS			OptionalMapping/StandardTerminalName				1. C+	2. C-	3. CPOUT1	4. CPOUT2	5. RZQ1	6. RZQ2		
MandatoryMapping/StandardTerminalName																																					
1. DIN0_c	2. DIN0_t	3. DIN1_c	4. DIN1_t																																		
5. DOUT0_c	6. DOUT0_t	7. DOUT1_c	8. DOUT1_t																																		
9. LSS	10. REF_CLK	11. RST_n	12. VCC																																		
13. VCCQ	14. VCCQ2	15. VDDi	16. VDDiQ																																		
17. VDDiQ2	18. VSS																																				
OptionalMapping/StandardTerminalName																																					
1. C+	2. C-	3. CPOUT1	4. CPOUT2																																		
5. RZQ1	6. RZQ2																																				

For more information about the UFS Interface, refer to the JEDEC standard JESD220E.

4.7.3.10.50.3. UFSHCI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UFSHCI										
diagram	<p>The diagram illustrates the UFSHCI Interface Function Type. It starts with a 'UFSHCI' element (type UFSHCI-InterfaceFunctionType) which connects to a 'StandardTerminalNameAssignment' element (type UFSHCI-StandardTerminalNameAssignmentType). This assignment is associated with a 'Mapping' element (type UFSHCI-StandardTerminalMappingType), which maps to a 'StandardTerminalName' element (type UFSHCI-StandardTerminalNameType). A constraint labeled 'constraints' is shown between the StandardTerminalNameAssignment and Mapping elements.</p>										
type	UFSHCI-InterfaceFunctionType , UFSHCI-StandardTerminalNameAssignmentType , UFSHCI-MandatoryStandardTerminalMappingType , UFSHCI-MandatoryStandardTerminalNameType ,										
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="2">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Tx+</td><td>2. Tx-</td></tr> <tr> <td>5. RX+</td><td>6. RX-</td></tr> <tr> <td></td><td>3. Ref_Clock</td></tr> <tr> <td></td><td>4. Reset</td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName		1. Tx+	2. Tx-	5. RX+	6. RX-		3. Ref_Clock		4. Reset
MandatoryMapping/StandardTerminalName											
1. Tx+	2. Tx-										
5. RX+	6. RX-										
	3. Ref_Clock										
	4. Reset										

For more information about the UFSHCI Interface, refer to the JEDEC standard JESD223C.

4.7.3.10.51. USB Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/USB
diagram	<p>The diagram illustrates the USB Interface Function Type. It starts with a 'USB' element (type USB-InterfaceFunctionType) which connects to a 'USB-InterfaceFunctionType' container. Inside this container are four specific interface types: 'USB2.0' (type USB2.0-InterfaceFunctionType), 'USB3.0' (type USB3.0-InterfaceFunctionType), 'USB_Type-C_Receptacle' (type USB_Type-C_ReceptacleInterfaceFunctionType), and 'USB_Type-C_Plug' (type USB_Type-C_PlugInterfaceFunctionType).</p>
type	USB-InterfaceFunctionType , USB2.0-InterfaceFunctionType , USB3.0-InterfaceFunctionType , USB_Type-C_ReceptacleInterfaceFunctionType , USB_Type-C_PlugInterfaceFunctionType .

4.7.3.10.51.1. USB2.0

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/USB2.0																
diagram	<pre> classDiagram class USB2.0 { type USB2.0-InterfaceFunctionType } class StandardTerminalNameAssignment { type USB2.0-StandardTerminalNameAssignmentType } class MandatoryMapping { type USB2.0-MandatoryStandardTerminalMappingType } class StandardTerminalName { type USB2.0-MandatoryStandardTerminalNameType } class OptionalMapping { type USB2.0-OptionalStandardTerminalMappingType } class StandardTerminalName { type USB2.0-OptionalStandardTerminalNameType } USB2.0 "1..∞" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "*" --> MandatoryMapping MandatoryMapping "2" --> StandardTerminalName StandardTerminalNameAssignment "*" --> OptionalMapping OptionalMapping "0..2" --> StandardTerminalName </pre>																
type	USB2.0-InterfaceFunctionType, USB2.0-StandardTerminalNameAssignmentType, USB2.0-MandatoryStandardTerminalMappingType, USB2.0-MandatoryStandardTerminalNameType, USB2.0-OptionalStandardTerminalMappingType, USB2.0-OptionalStandardTerminalNameType.																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. D-</td> <td>2. D+</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. VBUS</td> <td>2. ID</td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. D-	2. D+			OptionalMapping/StandardTerminalName				1. VBUS	2. ID		
MandatoryMapping/StandardTerminalName																	
1. D-	2. D+																
OptionalMapping/StandardTerminalName																	
1. VBUS	2. ID																

For more information about the USB2.0 Interface, refer to the USB-IF standard Universal Serial Bus Specification Revision 2.0.

4.7.3.10.51.2. USB3.0

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/USB3.0																				
diagram																					
type	USB3.0-InterfaceFunctionType, USB3.0-StandardTerminalNameAssignmentType, USB3.0-MandatoryStandardTerminalMappingType, USB3.0-MandatoryStandardTerminalNameType, USB3.0-OptionalStandardTerminalMappingType, USB3.0-OptionalStandardTerminalNameType.																				
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. SSTX+</td> <td>2. SSTX-</td> <td>3. SSRX+</td> <td>4. SSRX-</td> </tr> <tr> <td>5. D+</td> <td>6. D-</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. VBUS</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. SSTX+	2. SSTX-	3. SSRX+	4. SSRX-	5. D+	6. D-			OptionalMapping/StandardTerminalName				1. VBUS			
MandatoryMapping/StandardTerminalName																					
1. SSTX+	2. SSTX-	3. SSRX+	4. SSRX-																		
5. D+	6. D-																				
OptionalMapping/StandardTerminalName																					
1. VBUS																					

For more information about the USB3.0 Interface, refer to the USB-IF standard Universal Serial Bus 3.2 Specification Revision 1.0.

4.7.3.10.51.3. USB_Type-C_Receptacle

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/USB/USB_Type-C_Receptacle																
diagram	<pre> classDiagram class USB_Type-C_ReceptacleInterfaceFunctionType class StandardTerminalNameAssignment { <<USB_Type-C_ReceptacleStandardTerminalNameAssignmentType>> } class Mapping { <<USB_Type-C_ReceptacleStandardTerminalMappingType>> } class StandardTerminalName { <<USB_Type-C_ReceptacleStandardTerminalNameType>> } USB_Type-C_ReceptacleInterfaceFunctionType --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>																
type	USB_Type-C_Receptacle-InterfaceFunctionType , USB_Type-C_Receptacle-StandardTerminalNameAssignmentType , USB_Type-C_Receptacle-StandardTerminalMappingType , USB_Type-C_Receptacle-StandardTerminalNameType .																
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. CC1</td> <td>2. CC2</td> <td>3. D+</td> <td>4. D-</td> </tr> <tr> <td>5. GND</td> <td>6. RX1+</td> <td>7. RX1-</td> <td>8. RX2+</td> </tr> <tr> <td>9. RX2-</td> <td>10. SBU1</td> <td>11. SBU2</td> <td>12. TX1+</td> </tr> <tr> <td>13. TX1-</td> <td>14. TX2+</td> <td>15. TX2-</td> <td>16. VBUS</td> </tr> </table>	1. CC1	2. CC2	3. D+	4. D-	5. GND	6. RX1+	7. RX1-	8. RX2+	9. RX2-	10. SBU1	11. SBU2	12. TX1+	13. TX1-	14. TX2+	15. TX2-	16. VBUS
1. CC1	2. CC2	3. D+	4. D-														
5. GND	6. RX1+	7. RX1-	8. RX2+														
9. RX2-	10. SBU1	11. SBU2	12. TX1+														
13. TX1-	14. TX2+	15. TX2-	16. VBUS														

For more information about the USB Type-C Receptacle Interface, refer to the USB Type-C Specification 2.0.

4.7.3.10.51.4. USB_Type-C_Plug

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/USB/USB_Type-C_Plug																
diagram	<pre> classDiagram class USB_Type-C_PlugInterfaceFunctionType class StandardTerminalNameAssignment { <<USB_Type-C_PlugStandardTerminalNameAssignmentType>> } class Mapping { <<USB_Type-C_PlugStandardTerminalMappingType>> } class StandardTerminalName { <<USB_Type-C_PlugStandardTerminalNameType>> } USB_Type-C_PlugInterfaceFunctionType --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>																
type	USB_Type-C_Plug-InterfaceFunctionType , USB_Type-C_Plug-StandardTerminalNameAssignmentType , USB_Type-C_Plug-StandardTerminalMappingType , USB_Type-C_Plug-StandardTerminalNameType .																
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. CC</td> <td>2. D+</td> <td>3. D-</td> <td>4. GND</td> </tr> <tr> <td>5. RX1+</td> <td>6. RX1-</td> <td>7. RX2+</td> <td>8. RX2-</td> </tr> <tr> <td>9. SBU1</td> <td>10. SBU2</td> <td>11. TX1+</td> <td>12. TX1-</td> </tr> <tr> <td>13. TX2+</td> <td>14. TX2-</td> <td>15. VBUS</td> <td>16. VCONN</td> </tr> </table>	1. CC	2. D+	3. D-	4. GND	5. RX1+	6. RX1-	7. RX2+	8. RX2-	9. SBU1	10. SBU2	11. TX1+	12. TX1-	13. TX2+	14. TX2-	15. VBUS	16. VCONN
1. CC	2. D+	3. D-	4. GND														
5. RX1+	6. RX1-	7. RX2+	8. RX2-														
9. SBU1	10. SBU2	11. TX1+	12. TX1-														
13. TX2+	14. TX2-	15. VBUS	16. VCONN														

For more information about the USB Type-C Plug Interface, refer to the USB Type-C Specification 2.0.

4.7.3.10.51.5. XFM

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/XFM</code>																
diagram	<pre> classDiagram class XFM { <<XFM-InterfaceFunctionType>> } class StandardTerminalNameAssignment { <<XFM-StandardTerminalNameAssignmentType>> } class Mapping { <<XFM-StandardTerminalMappingType>> } class StandardTerminalName { <<XFM-StandardTerminalNameType>> } XFM "1" --> "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" --> "1..∞" Mapping Mapping "1..∞" --> "16" StandardTerminalName </pre>																
type	<code>XFM-InterfaceFunctionType</code> , <code>XFM -StandardTerminalNameAssignmentType</code> , <code>XFM -StandardTerminalMappingType</code> , <code>XFM -StandardTerminalNameType</code>																
list of enumerative values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. GND</td> <td>2. PERp0</td> <td>3. PERn0</td> <td>4. PERp1</td> </tr> <tr> <td>5. PERn1</td> <td>6. PETp0</td> <td>7. PETn0</td> <td>8. PETp1</td> </tr> <tr> <td>9. PETn1</td> <td>10. Reserved</td> <td>11.REFCLKp</td> <td>12.REFCLKn</td> </tr> <tr> <td>13. PERST#</td> <td>14. PWR_1</td> <td>15.PWR_2</td> <td>16.CLKREQ#</td> </tr> </table>	1. GND	2. PERp0	3. PERn0	4. PERp1	5. PERn1	6. PETp0	7. PETn0	8. PETp1	9. PETn1	10. Reserved	11.REFCLKp	12.REFCLKn	13. PERST#	14. PWR_1	15.PWR_2	16.CLKREQ#
1. GND	2. PERp0	3. PERn0	4. PERp1														
5. PERn1	6. PETp0	7. PETn0	8. PETp1														
9. PETn1	10. Reserved	11.REFCLKp	12.REFCLKn														
13. PERST#	14. PWR_1	15.PWR_2	16.CLKREQ#														

For more information about the XFM Interface, refer to the JEDEC Standard JESD233.

4.7.3.10.52. Other Interface Standard

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/OtherInterfaceStandard</code>
diagram	<pre> classDiagram class OtherInterfaceStandard { <<FunctionMap-to-StandardNameType>> } class StandardTerminalNameAssignment { <<OtherStandardFunctionStandardType>> } class Mapping { <<TerminalStandardType>> } class StandardTerminalName { <<xs:string>> } OtherInterfaceStandard "1" --> "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..∞" --> "1..∞" Mapping Mapping "1..∞" --> "1..∞" StandardTerminalName </pre>
type	<code>FunctionMap-to-StandardNameType</code> , <code>OtherStandardFunctionStandardType</code> , <code>TerminalStandardType</code> .

Specific interfaces that are not currently supported by the Schema can be captured under the “Other Interface Standard” section. Here the interface name can be captured along with an unbounded list of the Standard terminal Names mapped over to any one of the following `TerminalNumber`, `TerminalName` or the `InternalNodeName`.

4.7.3.11. Non Linear – Frequency Mixer

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear
diagram	<pre> classDiagram class NonLinear { <<NonLinearFunctionType>> } class FrequencyMixer { <<FrequencyMixerFunctionType>> } class OtherNonLinearFunctionStandard { <<FunctionMap-to-StandardNameType>> } class SingleBalanced { <<BalancedFrequencyMixerFunctionType>> } class DoubleBalanced { <<BalancedFrequencyMixerFunctionType>> } class TripleBalanced { <<BalancedFrequencyMixerFunctionType>> } NonLinear < -- FrequencyMixer FrequencyMixer < -- OtherNonLinearFunctionStandard FrequencyMixer < -- SingleBalanced FrequencyMixer < -- DoubleBalanced FrequencyMixer < -- TripleBalanced </pre> <p>The diagram illustrates the UML class hierarchy for Non Linear – Frequency Mixer. It starts with a general class NonLinear (type: NonLinearFunctionType) which inherits from a more specific class FrequencyMixer (type: FrequencyMixerFunctionType). The FrequencyMixer class then further branches into three subclasses: SingleBalanced, DoubleBalanced, and TripleBalanced, all of which are also type BalancedFrequencyMixerFunctionType. Additionally, there is a separate class OtherNonLinearFunctionStandard (type: FunctionMap-to-StandardNameType) that also inherits from FrequencyMixer.</p>
type	NonLinearFunctionType , FrequencyMixerFunctionType , BalancedFrequencyMixerFunctionType , FunctionMap-to-StandardNameType .

A **NonLinear** can be one of two types: **OtherNonLinearFunctionStandard** and **Frequency**, which itself can be **Balanced** or **Double**. Each of these types is elaborated on below.

4.7.3.11.1. Balanced

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/FrequencyMixer/SingleBalanced 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/FrequencyMixer/DoubleBalanced 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/FrequencyMixer/TripleBalanced 																
diagram																	
type	<p>BalancedFrequencyMixerFunctionType, FrequencyMixerStandardTerminalNameAssignmentType, BalancedFrequencyMixerMandatoryStandardTerminalMappingType, BalancedFrequencyMixerMandatoryStandardTerminalNameType, BalancedFrequencyMixerOptionalStandardTerminalMappingType, BalancedFrequencyMixerOptionalStandardTerminalNameType.</p>																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td></tr> <tr> <td>1. Output</td><td></td><td></td><td></td></tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td></tr> <tr> <td>1. Enable</td><td>2. Positive Rail</td><td>3. Negative Rail</td><td></td></tr> </table>	MandatoryMapping/StandardTerminalName				1. Output				OptionalMapping/StandardTerminalName				1. Enable	2. Positive Rail	3. Negative Rail	
MandatoryMapping/StandardTerminalName																	
1. Output																	
OptionalMapping/StandardTerminalName																	
1. Enable	2. Positive Rail	3. Negative Rail															

4.7.3.11.2. Other Non Linear Function Standard

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/OtherNonLinearFunctionStandard
diagram	
type	<p>FunctionMap-to-StandardNameType, OtherStandardFunctionStandardTerminalNameAssignmentType, TerminalStandardType.</p>

4.7.3.12. Optoelectronics

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic
diagram	<pre> graph LR Optoelectronic[Optoelectronic
type OptoelectronicFunctionType] --- OptoelectronicFunctionType[OptoelectronicFunctionType] subgraph OptoelectronicFunctionType [OptoelectronicFunctionType] Photoemitter[Photoemitter
type PhotoemitterFunctionType] PhotosensitiveDevice[PhotosensitiveDevice
type PhotosensitiveDeviceFunctionType...] Optocoupler[Optocoupler
type OptocouplerFunctionType] end </pre>
type	OptoelectronicFunctionType , PhotoemitterFunctionType , PhotosensitiveDeviceFunctionType , OptocouplerFunctionType .

An [Optoelectronic](#) can be one of the following three types: [Photoemitter](#), [PhotosensitiveDevice](#), and [Optocoupler](#). Each of these types is specified in further detail below.

4.7.3.12.1. Photoemitter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Photoemitter
diagram	<pre> graph LR Photoemitter[Photoemitter
type PhotoemitterFunctionType] --- PhotoemitterFunctionType[PhotoemitterFunctionType] subgraph PhotoemitterFunctionType [PhotoemitterFunctionType] InfraredEmittingDiode[InfraredEmittingDiode
type InfraredEmittingDiodeFunctionType...] LED[LED
type LEDFunctionType] Laser[Laser
type LaserFunctionType] end </pre>
type	PhotoemitterFunctionType , InfraredEmittingDiodeFunctionType , LEDFunctionType , LaserFunctionType .

A [Photoemitter](#) can be one of the following three types: [InfraredEmittingDiode](#), [LED](#), and [Laser](#). Each of these types is specified in further detail below.

4.7.3.12.1.1. Infrared Emitting Diode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Photoemitter/InfraredEmittingDiode			
diagram				
type	InfraredEmittingDiodeFunctionType , InfraredEmittingDiodeStandardTerminalNameAssignmentType , DiodeStandardTerminalMappingType , DiodeStandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. Anode 2. Cathode			

4.7.3.12.1.2. LED

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Photoemitter/LED			
diagram				
type	LEDFunctionType , LEDStandardTerminalNameAssignmentType , DiodeStandardTerminalMappingType , DiodeStandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. Anode 2. Cathode			

4.7.3.12.1.3. Laser

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Photoemitter/Laser								
diagram	<pre> classDiagram class LaserFunctionType class Laser class StandardTerminalNameAssignment class MandatoryMapping class OptionalMapping class StandardTerminalName class LaserStandardTerminalNameAssignmentType class LaserMandatoryStandardTerminalMappingType class LaserOptionalStandardTerminalMappingType Laser < -- LaserFunctionType StandardTerminalNameAssignment < -- LaserStandardTerminalNameAssignmentType MandatoryMapping < -- LaserMandatoryStandardTerminalMappingType OptionalMapping < -- LaserOptionalStandardTerminalMappingType StandardTerminalName < -- LaserOptionalStandardTerminalNameType Laser --> StandardTerminalNameAssignment : 1..> StandardTerminalNameAssignment --> MandatoryMapping : 1..> StandardTerminalNameAssignment --> OptionalMapping : 1..> MandatoryMapping --> StandardTerminalName : 2..> OptionalMapping --> StandardTerminalName : 2..> </pre> <p>The diagram illustrates the UML Class Diagram for the Laser Function Type. It features a central class, LaserFunctionType, which contains associations with other classes: Laser, StandardTerminalNameAssignment, MandatoryMapping, and OptionalMapping. The Laser class is associated with the StandardTerminalNameAssignment class (multiplicity 1..infinity). The StandardTerminalNameAssignment class is associated with both MandatoryMapping and OptionalMapping classes (multiplicity 1..infinity). Each of these mapping classes is associated with a StandardTerminalName class (multiplicity 2..infinity). The classes involved are categorized by their type: LaserFunctionType, StandardTerminalNameAssignmentType, MandatoryStandardTerminalMappingType, and OptionalStandardTerminalMappingType.</p>								
type	LaserFunctionType , LaserStandardTerminalNameAssignmentType , LaserMandatoryStandardTerminalMappingType , LaserMandatoryStandardTerminalNameType , LaserOptionalStandardTerminalMappingType , LaserOptionalStandardTerminalNameType .								
list of enumerate values	<table border="1"> <tr> <td colspan="2">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Anode</td> <td>2. Cathode</td> </tr> <tr> <td colspan="2">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Ground</td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName		1. Anode	2. Cathode	OptionalMapping/StandardTerminalName		1. Ground	
MandatoryMapping/StandardTerminalName									
1. Anode	2. Cathode								
OptionalMapping/StandardTerminalName									
1. Ground									

4.7.3.12.2. Photosensitive Device

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice
diagram	<pre> classDiagram class PhotosensitiveDeviceFunctionType { <<PhotosensitiveDeviceFunction...>> } class Photodiode { <<PhotodiodeType>> } class Phototyristor { <<PhototyristorType>> } class Phototriac { <<PhototriacType>> } class Phototransistor { <<PhototransistorType>> } class Photodarlington { <<PhotodarlingtonType>> } class PhotovoltaicDiode { <<PhotovoltaicDiodeType>> } PhotosensitiveDeviceFunctionType < -- PhotosensitiveDevice PhotosensitiveDevice < --> PhotosensitiveDeviceFunctionType PhotosensitiveDevice < --> Photodiode PhotosensitiveDevice < --> Phototyristor PhotosensitiveDevice < --> Phototriac PhotosensitiveDevice < --> Phototransistor PhotosensitiveDevice < --> Photodarlington PhotosensitiveDevice < --> PhotovoltaicDiode </pre>
type	PhotosensitiveDeviceFunctionType , PhotodiodeType , PhototyristorType , PhototriacType , PhototransistorType , PhotodarlingtonType , PhotovoltaicDiodeType .

A *PhotosensitiveDevice* can be one of the following six types: *Photodiode*, *Phototyristor*, *Phototriac*, *Phototransistor*, *Photodarlington*, and *PhotovoltaicDiode*. Each of these types is explained in further detail below.

4.7.3.12.2.1. Photodiode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Photodiode				
diagram	<pre> classDiagram class PhotodiodeType { <<PhotodiodeType>> } class StandardTerminalNameAssignment { <<DiodeStandardTerminalNameAssignmentType>> } class Mapping { <<DiodeStandardTerminalMappingType>> } class StandardTerminalName { <<DiodeStandardTerminalNameType>> } PhotodiodeType < --> StandardTerminalNameAssignment PhotodiodeType < --> Mapping PhotodiodeType < --> StandardTerminalName StandardTerminalNameAssignment < --> Mapping constraints </pre>				
type	PhotodiodeType , PhotodiodeStandardTerminalNameAssignmentType , PhotodiodeStandardTerminalMappingType , PhotodiodeStandardTerminalNameType .				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Anode</td> <td>2. Cathode</td> <td></td> <td></td> </tr> </table>	1. Anode	2. Cathode		
1. Anode	2. Cathode				

4.7.3.12.2.2. Photothyristor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Photothyristor						
diagram	<pre> classDiagram class PhotothyristorType { <<Photothyristor>> <<StandardTerminalNameAssignment>> } class PhotothyristorStandardTerminalNameAssignmentType { <<StandardTerminalNameAssignment>> <<MandatoryMapping>> <<OptionalMapping>> } class PhotothyristorMandatoryStandardTerminalMappingType { <<StandardTerminalName>> } class PhotothyristorOptionalStandardTerminalMappingType { <<StandardTerminalName>> } PhotothyristorType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping StandardTerminalNameAssignment "1..>" OptionalMapping MandatoryMapping "2..>" PhotothyristorMandatoryStandardTerminalMappingType OptionalMapping "2..>" PhotothyristorOptionalStandardTerminalMappingType </pre>						
type	PhotothyristorType , PhotothyristorStandardTerminalNameAssignmentType , PhotothyristorMandatoryStandardTerminalMappingType , PhotothyristorMandatoryStandardTerminalNameType , PhotothyristorOptionalStandardTerminalMappingType , PhotothyristorOptionalStandardTerminalNameType .						
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. Anode</td> <td>2. Cathode</td> <td></td> </tr> </table> OptionalMapping/StandardTerminalName <table border="1"> <tr> <td>1. Gate</td> <td></td> <td></td> </tr> </table>	1. Anode	2. Cathode		1. Gate		
1. Anode	2. Cathode						
1. Gate							

4.7.3.12.2.3. Phototriac

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Phototriac						
diagram	<pre> classDiagram class PhototriacType { <<Phototriac>> <<StandardTerminalNameAssignment>> } class PhototriacStandardTerminalNameAssignmentType { <<StandardTerminalNameAssignment>> <<MandatoryMapping>> <<OptionalMapping>> } class PhototriacMandatoryStandardTerminalMappingType { <<StandardTerminalName>> } class PhototriacOptionalStandardTerminalMappingType { <<StandardTerminalName>> } PhototriacType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping StandardTerminalNameAssignment "1..>" OptionalMapping MandatoryMapping "2..>" PhototriacMandatoryStandardTerminalMappingType OptionalMapping "2..>" PhototriacOptionalStandardTerminalMappingType </pre>						
type	PhototriacType , PhototriacStandardTerminalNameAssignmentType , PhototriacMandatoryStandardTerminalMappingType , PhototriacMandatoryStandardTerminalNameType , PhototriacOptionalStandardTerminalMappingType , PhototriacOptionalStandardTerminalNameType .						
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. Main Terminal 1</td> <td>2. Main Terminal 2</td> <td></td> </tr> </table> OptionalMapping/StandardTerminalName <table border="1"> <tr> <td>1. Gate</td> <td></td> <td></td> </tr> </table>	1. Main Terminal 1	2. Main Terminal 2		1. Gate		
1. Main Terminal 1	2. Main Terminal 2						
1. Gate							

4.7.3.12.2.4. Phototransistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Phototransistor								
diagram	<pre> classDiagram class PhototransistorType class StandardTerminalNameAssignment class PhototransistorStandardTerminalNameAssignmentType { <<PhototransistorStandardTerminalNameAssignment>> } class MandatoryMapping { <<PhototransistorMandatoryStandardTerminalMapping>> } class StandardTerminalName { <<PhototransistorMandatoryStandardTerminalName>> } class OptionalMapping { <<PhototransistorOptionalStandardTerminalMapping>> } class StandardTerminalName { <<PhototransistorOptionalStandardTerminalName>> } PhototransistorType "1..∞" *-- "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" *-- "1..∞" PhototransistorStandardTerminalNameAssignmentType PhototransistorStandardTerminalNameAssignmentType "*" *-- "1..∞" MandatoryMapping PhototransistorStandardTerminalNameAssignmentType "*" *-- "0..1" OptionalMapping MandatoryMapping "*" *-- "1..∞" StandardTerminalName OptionalMapping "*" *-- "0..1" StandardTerminalName </pre>								
type	PhototransistorType , PhototransistorStandardTerminalNameAssignmentType , PhototransistorMandatoryStandardTerminalMappingType , PhototransistorMandatoryStandardTerminalNameType , PhototransistorOptionalStandardTerminalMappingType , PhototransistorOptionalStandardTerminalNameType .								
list of enumerate values	<table border="1"> <tr> <td colspan="2">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Collector</td> <td>2. Emitter</td> </tr> <tr> <td colspan="2">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Base</td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName		1. Collector	2. Emitter	OptionalMapping/StandardTerminalName		1. Base	
MandatoryMapping/StandardTerminalName									
1. Collector	2. Emitter								
OptionalMapping/StandardTerminalName									
1. Base									

4.7.3.12.2.5. Photodarlington

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Photodarlington				
diagram	<pre> classDiagram class PhotodarlingtonType class StandardTerminalNameAssignment class PhotodarlingtonStandardTerminalNameAssignmentType { <<PhotodarlingtonStandardTerminalNameAssignment>> } class Mapping { <<PhotodarlingtonStandardTerminalMapping>> } class StandardTerminalName { <<PhotodarlingtonStandardTerminalName>> } PhotodarlingtonType "1..∞" *-- "1..∞" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" *-- "1..∞" PhotodarlingtonStandardTerminalNameAssignmentType PhotodarlingtonStandardTerminalNameAssignmentType "*" *-- "1..∞" Mapping Mapping "*" *-- "1..∞" StandardTerminalName </pre>				
type	PhotodarlingtonType , PhotodarlingtonStandardTerminalNameAssignmentType , PhotodarlingtonStandardTerminalMappingType , PhotodarlingtonStandardTerminalNameType .				
list of enumerate values	<table border="1"> <tr> <td colspan="2">Mapping/StandardTerminalName</td> </tr> <tr> <td>1. Collector</td> <td>2. Emitter</td> </tr> </table>	Mapping/StandardTerminalName		1. Collector	2. Emitter
Mapping/StandardTerminalName					
1. Collector	2. Emitter				

4.7.3.12.2.6. Photovoltaic Diode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Opto electronic/PhotosensitiveDevice/PhotovoltaicDiode				
diagram	<pre> classDiagram class PhotovoltaicDiode { type PhotovoltaicDiodeType } class StandardTerminalNameAssignment { type DiodeStandardTerminalNameAssignmentType } class Mapping { type DiodeStandardTerminalMappingType } class StandardTerminalName { type DiodeStandardTerminalNameType } PhotovoltaicDiode "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>				
type	PhotovoltaicDiodeType , PhotovoltaicDiodeStandardTerminalNameAssignmentType , PhotovoltaicDiodeStandardTerminalMappingType , PhotovoltaicDiodeStandardTerminalNameType .				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Anode</td> <td>2. Cathode</td> <td></td> <td></td> </tr> </table>	1. Anode	2. Cathode		
1. Anode	2. Cathode				

4.7.3.12.3. Optocoupler

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Opto electronic/Optocoupler
diagram	<pre> classDiagram class Optocoupler { type OptocouplerFunctionType } class Photodiode { type PhotodiodeOptocouplerType } class Photothyristor { type PhotothyristorOptocouplerType } class Phototriac { type PhototriacOptocouplerType } class Phototransistor { type PhototransistorOptocouplerType } class Photodarlington { type PhotodarlingtonOptocouplerType } Optocoupler "1..>" Photodiode Optocoupler "1..>" Photothyristor Optocoupler "1..>" Phototriac Optocoupler "1..>" Phototransistor Optocoupler "1..>" Photodarlington </pre>
type	OptocouplerFunctionType , PhotodiodeOptocouplerType , PhotothyristorOptocouplerType , PhototriacOptocouplerType , PhototransistorOptocouplerType , PhotodarlingtonOptocouplerType .

An *Optocoupler* can be one of the following five types: *Photodiode*, *Photothyristor*, *Phototriac*, *Phototransistor*, and *Photodarlington*. Each of these types is further specified below.

4.7.3.12.3.1. Photodiode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/OptoElectronic/Optocoupler/Photodiode							
diagram	<pre> classDiagram class Photodiode { <<PhotodiodeOptocouplerType>> } class StandardTerminalNameAssignment { <<PhotodiodeOptocouplerStandardTerminalNameAssignmentType>> } class Mapping { <<PhotodiodeOptocouplerStandardTerminalMappingType>> } class StandardTerminalName { <<PhotodiodeOptocouplerStandardTerminalNameType>> } Photodiode "1..0" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..0" --> Mapping Mapping "1..0" --> StandardTerminalName </pre>							
type	PhotodiodeOptocouplerType , PhotodiodeOptocouplerStandardTerminalNameAssignmentType , PhotodiodeOptocouplerStandardTerminalMappingType , PhotodiodeOptocouplerStandardTerminalNameType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. LED-Anode</td> <td style="padding: 2px;">2. LED-Cathode</td> <td style="padding: 2px;">3. Detector-Anode</td> <td style="padding: 2px;">4. Detector-Cathode</td> </tr> </table>				1. LED-Anode	2. LED-Cathode	3. Detector-Anode	4. Detector-Cathode
1. LED-Anode	2. LED-Cathode	3. Detector-Anode	4. Detector-Cathode					

4.7.3.12.3.2. Photothyristor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/OptoElectronic/Optocoupler/Photothyristor											
diagram	<pre> classDiagram class Photothyristor { <<PhotothyristorOptocouplerType>> } class StandardTerminalNameAssignment { <<PhotothyristorOptocouplerStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<PhotothyristorOptocouplerMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<PhotothyristorOptocouplerOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<PhotothyristorOptocouplerStandardTerminalNameType>> } Photothyristor "1..0" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..0" --> MandatoryMapping StandardTerminalNameAssignment "1..0" --> OptionalMapping MandatoryMapping "1..0" --> StandardTerminalName OptionalMapping "1..0" --> StandardTerminalName </pre>											
type	PhotothyristorOptocouplerType , PhotothyristorOptocouplerStandardTerminalNameAssignmentType , PhotothyristorOptocouplerMandatoryStandardTerminalMappingType , PhotothyristorOptocouplerMandatoryStandardTerminalNameType , PhotothyristorOptocouplerOptionalStandardTerminalMappingType , PhotothyristorOptocouplerOptionalStandardTerminalNameType .											
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. LED-Anode</td> <td style="padding: 2px;">2. LED-Cathode</td> <td style="padding: 2px;">3. Detector-Anode</td> <td style="padding: 2px;">4. Detector-Cathode</td> </tr> </table> OptionalMapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Detector-Gate</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>				1. LED-Anode	2. LED-Cathode	3. Detector-Anode	4. Detector-Cathode	1. Detector-Gate			
1. LED-Anode	2. LED-Cathode	3. Detector-Anode	4. Detector-Cathode									
1. Detector-Gate												

4.7.3.12.3.3. Phototriac

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optolectric/Optocoupler/Phototriac</code>																
diagram	<pre> classDiagram class Phototriac { type: PhototriacOptocouplerType } class StandardTerminalNameAssignment { type: PhototriacOptocouplerStandardTerminalNameAssignmentType } class MandatoryMapping { type: PhototriacOptocouplerMandatoryStandardTerminalMappingType } class OptionalMapping { type: PhototriacOptocouplerOptionalStandardTerminalMappingType } class StandardTerminalName { type: PhototriacOptocouplerStandardTerminalNameType } Phototriac "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping StandardTerminalNameAssignment "1..>" OptionalMapping MandatoryMapping "4..>" StandardTerminalName OptionalMapping "1..>" StandardTerminalName </pre>																
type	PhototriacOptocouplerType, PhototriacOptocouplerStandardTerminalNameAssignmentType, PhototriacOptocouplerMandatoryStandardTerminalMappingType, PhototriacOptocouplerMandatoryStandardTerminalNameType, PhototriacOptocouplerOptionalStandardTerminalMappingType, PhototriacOptocouplerOptionalStandardTerminalNameType.																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. LED-Anode</td> <td>2. LED-Cathode</td> <td>3. Detector – Main Terminal 1</td> <td>4. Detector – Main Terminal 2</td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Detector-Gate</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. LED-Anode	2. LED-Cathode	3. Detector – Main Terminal 1	4. Detector – Main Terminal 2	OptionalMapping/StandardTerminalName				1. Detector-Gate			
MandatoryMapping/StandardTerminalName																	
1. LED-Anode	2. LED-Cathode	3. Detector – Main Terminal 1	4. Detector – Main Terminal 2														
OptionalMapping/StandardTerminalName																	
1. Detector-Gate																	

4.7.3.12.3.4. Phototransistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optolectric/Optocoupler/Phototransistor																
diagram	<pre> classDiagram class Phototransistor { <<PhototransistorOptocouplerType>> } class StandardTerminalNameAssignment { <<PhototransistorOptocouplerStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<PhototransistorOptocouplerMandatoryStandardTerminalMappingType>> } class OptionalMapping { <<PhototransistorOptocouplerOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<PhototransistorOptocouplerMandatoryStandardTerminalNameType>> } Phototransistor "1..*" --> "1..1" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..1" --> MandatoryMapping StandardTerminalNameAssignment "1..1" --> OptionalMapping MandatoryMapping "1..1" --> StandardTerminalName OptionalMapping "1..1" --> StandardTerminalName </pre>																
type	PhototransistorOptocouplerType , PhototransistorOptocouplerStandardTerminalNameAssignmentType , PhototransistorOptocouplerMandatoryStandardTerminalMappingType , PhototransistorOptocouplerMandatoryStandardTerminalNameType , PhototransistorOptocouplerOptionalStandardTerminalMappingType , PhototransistorOptocouplerOptionalStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. LED-Anode</td> <td>2. LED-Cathode</td> <td>3. Detector-Anode,</td> <td>4. Detector-Cathode</td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Detector-Base</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. LED-Anode	2. LED-Cathode	3. Detector-Anode,	4. Detector-Cathode	OptionalMapping/StandardTerminalName				1. Detector-Base			
MandatoryMapping/StandardTerminalName																	
1. LED-Anode	2. LED-Cathode	3. Detector-Anode,	4. Detector-Cathode														
OptionalMapping/StandardTerminalName																	
1. Detector-Base																	

4.7.3.12.3.5. Photodarlington

path	2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Optocoupler/Photodarlington																								
diagram	<pre> classDiagram class Photodarlington { <<PhotodarlingtonOptocouplerType>> } class StandardTerminalNameAssignment { <<PhotodarlingtonOptocouplerStandardTerminalNameAssignmentType>> } class BiDirectionalMapping { <<PhotodarlingtonOptocouplerBi-directionalStandardTerminalMappingType>> } class UniDirectionalMapping { <<PhotodarlingtonOptocouplerUni-directionalStandardTerminalMappingType>> } class MandatoryMapping { <<PhotodarlingtonOptocouplerMandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<PhotodarlingtonOptocouplerStandardTerminalNameType>> } Photodarlington "1..e" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..e" --> BiDirectionalMapping StandardTerminalNameAssignment "1..e" --> UniDirectionalMapping StandardTerminalNameAssignment "1..e" --> MandatoryMapping BiDirectionalMapping "1..e" --> StandardTerminalName UniDirectionalMapping "1..e" --> StandardTerminalName MandatoryMapping "1..e" --> StandardTerminalName </pre>																								
type	PhotodarlingtonOptocouplerType, PhotodarlingtonOptocouplerStandardTerminalNameAssignmentType, PhotodarlingtonOptocouplerBi-directionalStandardTerminalMappingType, PhotodarlingtonOptocouplerBi-directionalStandardTerminalNameType, PhotodarlingtonOptocouplerUni-directionalStandardTerminalMappingType, PhotodarlingtonOptocouplerUni-directionalStandardTerminalNameType, PhotodarlingtonOptocouplerMandatoryStandardTerminalMappingType, PhotodarlingtonOptocouplerMandatoryStandardTerminalNameType.																								
list of enumerate values	<table border="1"> <tr> <td colspan="4">Bi-directionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Cathode-Anode</td> <td>2. Anode-Cathode</td> <td></td> <td></td> </tr> <tr> <td colspan="4">Uni-directionalMapping /StandardTerminalName</td> </tr> <tr> <td>1. Anode</td> <td>2. Cathode</td> <td></td> <td></td> </tr> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Detector-Collector</td> <td>2. Detector-Emitter</td> <td></td> <td></td> </tr> </table>	Bi-directionalMapping/StandardTerminalName				1. Cathode-Anode	2. Anode-Cathode			Uni-directionalMapping /StandardTerminalName				1. Anode	2. Cathode			MandatoryMapping/StandardTerminalName				1. Detector-Collector	2. Detector-Emitter		
Bi-directionalMapping/StandardTerminalName																									
1. Cathode-Anode	2. Anode-Cathode																								
Uni-directionalMapping /StandardTerminalName																									
1. Anode	2. Cathode																								
MandatoryMapping/StandardTerminalName																									
1. Detector-Collector	2. Detector-Emitter																								

4.7.3.13. Relay

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay</code>
diagram	<pre> classDiagram class Relay { type: RelayFunctionType } class SolidState { type: SolidStateRelayFunctionType } class Electromagnetic { type: ElectromagneticRelayFunctionType } class SingleThrow { type: SolidStateRelaySingleThrowType } class DoubleThrow { type: SolidStateRelayDoubleThrowType } class SingleThrow2 { type: ElectromagneticRelaySingleThrowType } class DoubleThrow2 { type: ElectromagneticRelayDoubleThrowType } Relay < -- SolidState Relay < -- Electromagnetic SolidState < -- SingleThrow SolidState < -- DoubleThrow Electromagnetic < -- SingleThrow2 Electromagnetic < -- DoubleThrow2 </pre>
type	<code>RelayFunctionType, SolidStateRelayFunctionType, SolidStateRelaySingleThrowType,</code> <code>SolidStateRelayDoubleThrowType, ElectromagneticRelayFunctionType,</code> <code>ElectromagneticRelaySingleThrowType, ElectromagneticRelayDoubleThrowType.</code>

4.7.3.13.1. Solid State Relay – Single Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/SolidState/SingleThrow																
diagram	<pre> classDiagram class SolidStateRelaySingleThrowType { <<SolidStateRelaySingleThrowType>> } class SolidStateRelayType { <<SolidStateRelayType>> } class StandardTerminalNameAssignment { <<StandardTerminalNameAssignment>> } class Bi-directionalMapping { <<Bi-directionalMapping>> } class StandardTerminalName { <<StandardTerminalName>> } class Uni-directionalMapping { <<Uni-directionalMapping>> } class SolidStateRelayUni-directionalStandardTerminalMappingType { <<SolidStateRelayUni-directionalStandardTerminalMappingType>> } class StandardTerminalName { <<StandardTerminalName>> } class ChoiceMapping { <<ChoiceMapping>> } class SolidStateRelaySingleThrowChoiceStandardTerminalMappingType { <<SolidStateRelaySingleThrowChoiceStandardTerminalMappingType>> } class StandardTerminalName { <<StandardTerminalName>> } class OptionalMapping { <<OptionalMapping>> } class SolidStateRelaySingleThrowOptionalStandardTerminalMappingType { <<SolidStateRelaySingleThrowOptionalStandardTerminalMappingType>> } class StandardTerminalName { <<StandardTerminalName>> } SolidStateRelaySingleThrowType < -- SolidStateRelayType StandardTerminalNameAssignment < -- SolidStateRelayType StandardTerminalNameAssignment < -- Bi-directionalMapping StandardTerminalNameAssignment < -- Uni-directionalMapping StandardTerminalNameAssignment < -- ChoiceMapping StandardTerminalNameAssignment < -- OptionalMapping </pre>																
type	SolidStateRelayFunctionType , SolidStateRelaySingleThrowType , SolidStateRelaySingleThrowStandardTerminalNameAssignmentType , SolidStateRelayBi-directionalStandardTerminalMappingType , SolidStateRelayBi-directionalStandardTerminalNameType , SolidStateRelayUni-directionalStandardTerminalMappingType , SolidStateRelayUni-directionalStandardTerminalNameType , SolidStateRelaySingleThrowChoiceStandardTerminalMappingType , SolidStateRelaySingleThrowChoiceStandardTerminalNameType , SolidStateRelaySingleThrowOptionalStandardTerminalMappingType , SolidStateRelaySingleThrowOptionalStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td colspan="2">Bi-directionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Cathode-Anode</td> <td>2. Anode-Cathode</td> </tr> <tr> <td colspan="2">Uni-directionalMapping /StandardTerminalName</td> </tr> <tr> <td>1. Anode</td> <td>2. Cathode</td> </tr> <tr> <td colspan="2">ChoiceMapping/StandardTerminalName</td> </tr> <tr> <td>1. Normally Closed</td> <td>2. Normally Open</td> </tr> <tr> <td colspan="2">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Pole</td> <td></td> </tr> </table>	Bi-directionalMapping/StandardTerminalName		1. Cathode-Anode	2. Anode-Cathode	Uni-directionalMapping /StandardTerminalName		1. Anode	2. Cathode	ChoiceMapping/StandardTerminalName		1. Normally Closed	2. Normally Open	OptionalMapping/StandardTerminalName		1. Pole	
Bi-directionalMapping/StandardTerminalName																	
1. Cathode-Anode	2. Anode-Cathode																
Uni-directionalMapping /StandardTerminalName																	
1. Anode	2. Cathode																
ChoiceMapping/StandardTerminalName																	
1. Normally Closed	2. Normally Open																
OptionalMapping/StandardTerminalName																	
1. Pole																	

4.7.3.13.2. Solid State Relay – Double Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/SolidState/DoubleThrow																								
diagram	<pre> classDiagram class DoubleThrow { <<SolidStateRelay.DoubleThrowType>> } class StandardTerminalNameAssignment { <<SolidStateRelay.DoubleThrowStandardTerminalNameAssignmentType>> } class Bi-directionalMapping { <<SolidStateRelay.Bi-directionalStandardTerminalMappingType>> } class Uni-directionalMapping { <<SolidStateRelay.Uni-directionalStandardTerminalMappingType>> } class MandatoryMapping { <<SolidStateRelay.DoubleThrowMandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<SolidStateRelay.Bi-directionalStandardTerminalNameType>> } class StandardTerminalName { <<SolidStateRelay.Uni-directionalStandardTerminalNameType>> } class StandardTerminalName { <<SolidStateRelay.DoubleThrowMandatoryStandardTerminalNameType>> } DoubleThrow "1..x" --> StandardTerminalNameAssignment DoubleThrow "1..x" --> Bi-directionalMapping DoubleThrow "1..x" --> Uni-directionalMapping DoubleThrow "1..x" --> MandatoryMapping StandardTerminalNameAssignment "1..x" --> StandardTerminalName Bi-directionalMapping "1..x" --> StandardTerminalName Uni-directionalMapping "1..x" --> StandardTerminalName MandatoryMapping "1..x" --> StandardTerminalName </pre>																								
type	<p>SolidStateRelayFunctionType,</p> <p>SolidStateRelayDoubleThrowStandardTerminalNameAssignmentType,</p> <p>SolidStateRelayBi-directionalStandardTerminalMappingType,</p> <p>SolidStateRelayBi-directionalStandardTerminalNameType,</p> <p>SolidStateRelayUni-directionalStandardTerminalMappingType,</p> <p>SolidStateRelayUni-directionalStandardTerminalNameType,</p> <p>SolidStateRelayDoubleThrowMandatoryStandardTerminalMappingType,</p> <p>SolidStateRelayDoubleThrowMandatoryStandardTerminalNameType.</p>																								
list of enumerate values	<table border="1"> <tr> <td colspan="4">Bi-directionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Cathode-Anode</td> <td>2. Anode-Cathode</td> <td></td> <td></td> </tr> <tr> <td colspan="4">Uni-directionalMapping /StandardTerminalName</td> </tr> <tr> <td>1. Anode</td> <td>2. Cathode</td> <td></td> <td></td> </tr> <tr> <td colspan="4">ChoiceMapping/StandardTerminalName</td> </tr> <tr> <td>1. Normally Closed</td> <td>2. Normally Open</td> <td>3. Pole</td> <td></td> </tr> </table>	Bi-directionalMapping/StandardTerminalName				1. Cathode-Anode	2. Anode-Cathode			Uni-directionalMapping /StandardTerminalName				1. Anode	2. Cathode			ChoiceMapping/StandardTerminalName				1. Normally Closed	2. Normally Open	3. Pole	
Bi-directionalMapping/StandardTerminalName																									
1. Cathode-Anode	2. Anode-Cathode																								
Uni-directionalMapping /StandardTerminalName																									
1. Anode	2. Cathode																								
ChoiceMapping/StandardTerminalName																									
1. Normally Closed	2. Normally Open	3. Pole																							

4.7.3.13.3. Electromagnetic Relay – Single Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/SingleThrow				
diagram	<pre> classDiagram class ElectromagneticRelaySingleThrowType { <<ElectromagneticRelaySingleThrowType>> } class SingleThrow { <<ElectromagneticRelaySingleThrowType>> } class StandardTerminalNameAssignment { <<ElectromagneticRelaySingleThrowType>> } class Mapping { <<ElectromagneticRelaySingleThrowType>> } class Contact { <<ElectromagneticRelaySingleThrowType>> } class ContactArray { <<ElectromagneticRelaySingleThrowType>> } ElectromagneticRelaySingleThrowType "1..x" --> SingleThrow : ElectromagneticRelaySingleThrowType "1..x" --> StandardTerminalNameAssignment : ElectromagneticRelaySingleThrowType "1..x" --> Mapping : ElectromagneticRelaySingleThrowType "1..x" --> ContactArray : ElectromagneticRelaySingleThrowType "1..x" --> Contact : StandardTerminalNameAssignment "1..x" --> Mapping : ContactArray "1..x" --> Contact : constraints </pre>				
type	ElectromagneticRelayFunctionType , ElectromagneticRelaySingleThrowType , ElectromagneticRelaySingleThrowStandardTerminalNameAssignmentType , ElectromagneticRelayMandatoryStandardTerminalMappingType , ElectromagneticRelayMandatoryStandardTerminalNameType , ElectromagneticRelaySingleThrowContactArrayType , ElectromagneticRelaySingleThrowContactType .				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Input 1</td> <td>2. Input 2</td> <td></td> <td></td> </tr> </table>	1. Input 1	2. Input 2		
1. Input 1	2. Input 2				

4.7.3.13.3.1. Electromagnetic Relay – Single Throw – Contact Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/SingleThrow/StandardTerminalNameAssignment/Contact-Array/Contact								
diagram	<pre> classDiagram class ElectromagneticRelaySingleThrowContactType { <<ElectromagneticRelaySingleThrowContactType>> } class Contact { <<ElectromagneticRelaySingleThrowContactType>> } class ChoiceMapping { <<ElectromagneticRelaySingleThrowContactType>> } class StandardTerminalName { <<ElectromagneticRelaySingleThrowContactType>> } class MandatoryMapping { <<ElectromagneticRelaySingleThrowContactType>> } ElectromagneticRelaySingleThrowContactType "1..x" --> Contact : ElectromagneticRelaySingleThrowContactType "1..x" --> ChoiceMapping : ElectromagneticRelaySingleThrowContactType "1..x" --> StandardTerminalName : ElectromagneticRelaySingleThrowContactType "1..x" --> MandatoryMapping : ChoiceMapping "1..x" --> StandardTerminalName : MandatoryMapping "1..x" --> StandardTerminalName : constraints </pre>								
type	ElectromagneticRelaySingleThrowContactType , ElectromagneticRelaySingleThrowChoiceStandardTerminalMappingType , ElectromagneticRelaySingleThrowChoiceStandardTerminalNameType , ElectromagneticRelaySingleThrowMandatoryStandardTerminalMappingType , ElectromagneticRelaySingleThrowMandatoryStandardTerminalNameType .								
list of enumerate values	ChoiceMapping/StandardTerminalName <table border="1"> <tr> <td>1. Normally Closed</td> <td>2. Normally Open</td> <td></td> <td></td> </tr> </table> MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. Pole</td> <td></td> <td></td> <td></td> </tr> </table>	1. Normally Closed	2. Normally Open			1. Pole			
1. Normally Closed	2. Normally Open								
1. Pole									

4.7.3.13.4. Electromagnetic Relay – Double Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/DoubleThrow				
diagram	<pre> classDiagram class ElectromagneticRelayDoubleThrowType { <<ElectromagneticRelayDoubleThrowType>> } class DoubleThrow { <<ElectromagneticRelayDoubleThrowType>> } class StandardTerminalNameAssignment { <<ElectromagneticRelayDoubleThrowStandardTerminalNameAssignmentType>> } class Contact { <<ElectromagneticRelayDoubleThrowContactType>> } class ContactArray { <<ElectromagneticRelayDoubleThrowContactArrayType>> } class Mapping { <<ElectromagneticRelayMandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<ElectromagneticRelayMandatoryStandardTerminalNameType>> } ElectromagneticRelayDoubleThrowType "1..>" DoubleThrow ElectromagneticRelayDoubleThrowType "1..>" StandardTerminalNameAssignment ElectromagneticRelayDoubleThrowType "1..>" ContactArray ElectromagneticRelayDoubleThrowType "1..>" Mapping DoubleThrow "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping ContactArray "1..>" Contact Mapping "1..>" StandardTerminalName </pre>				
type	ElectromagneticRelayFunctionType , ElectromagneticRelaySingleThrowType , ElectromagneticRelaySingleThrowStandardTerminalNameAssignmentType , ElectromagneticRelayMandatoryStandardTerminalMappingType , ElectromagneticRelayMandatoryStandardTerminalNameType , ElectromagneticRelaySingleThrowContactArrayType , ElectromagneticRelayDoubleThrowContactType .				
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. Input 1</td> <td>2. Input 2</td> <td></td> <td></td> </tr> </table>	1. Input 1	2. Input 2		
1. Input 1	2. Input 2				

4.7.3.13.5. Electromagnetic Relay – Double Throw – Contact Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/DoubleThrow/StandardTerminalNameAssignment/Contact-Array/Contact				
diagram	<pre> classDiagram class ElectromagneticRelayDoubleThrowContactType { <<ElectromagneticRelayDoubleThrowContactType>> } class Contact { <<ElectromagneticRelayDoubleThrowContactType>> } class Mapping { <<ElectromagneticRelayDoubleThrowMandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<ElectromagneticRelayMandatoryStandardTerminalNameType>> } ElectromagneticRelayDoubleThrowContactType "1..>" Contact ElectromagneticRelayDoubleThrowContactType "1..>" Mapping ElectromagneticRelayDoubleThrowContactType "1..>" StandardTerminalName Contact "1..>" Mapping Mapping "1..>" StandardTerminalName </pre>				
type	ElectromagneticRelayDoubleThrowContactType , ElectromagneticRelayDoubleThrowMandatoryStandardTerminalMappingType , ElectromagneticRelayDoubleThrowMandatoryStandardTerminalNameType .				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Normally Closed</td> <td>2. Normally Open</td> <td>3. Pole</td> <td></td> </tr> </table>	1. Normally Closed	2. Normally Open	3. Pole	
1. Normally Closed	2. Normally Open	3. Pole			

4.7.3.14. Resistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor
diagram	<pre> classDiagram class Resistor { type ResistorFunctionType } class Fixed { type FixedResistorFunctionType } class Variable { type VariableResistorFunctionType } class Photo { type PhotoResistorFunctionType } class PotentiometerFunction { type PotentiometerFunctionType } class Varistor { type VaristorResistorFunctionType } class Thermistor { type ThermistorFunctionType } class LightDependent { type LightDependentResistorFunctionType } class Shunt { type ShuntResistorFunctionType } class Magnetic { type MagneticResistorFunctionType } Resistor < -- Fixed Resistor < -- Variable Resistor < -- Photo Resistor < -- PotentiometerFunction Resistor < -- Varistor Resistor < -- Thermistor Resistor < -- LightDependent Resistor < -- Shunt Resistor < -- Magnetic Fixed --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType Variable --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType Photo --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType PotentiometerFunction --> StandardTerminalNameAssignmentType : type PotentiometerStandardTerminalType Varistor --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType Thermistor --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType LightDependent --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType Shunt --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType Magnetic --> StandardTerminalNameAssignmentType : type TwoTerminalResistorStandardType </pre>
type	ResistorFunctionType , FixedResistorFunctionType , VariableResistorFunctionType , PhotoResistorFunctionType , PotentiometerFunctionType , VaristorResistorFunctionType , ThermistorFunctionType , LightDependentResistorFunctionType , ShuntResistorFunctionType , MagneticResistorFunctionType , TwoTerminalResistorStandardTerminalNameAssignmentType , PotentiometerStandardTerminalNameAssignmentType .

4.7.3.14.1. Two Terminal Resistor Standard Terminal Name Assignment Type

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Fixed/StandardTerminalNameAssignment 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Variable/StandardTerminalNameAssignment 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Photo/StandardTerminalNameAssignment 4. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/PotentiometerFunction/StandardTerminalNameAssignment 5. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Varistor/StandardTerminalNameAssignment 6. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Termistor/StandardTerminalNameAssignment 7. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/LightDependent/StandardTerminalNameAssignment 8. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Shunt/StandardTerminalNameAssignment 9. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Magnetic/StandardTerminalNameAssignment 				
diagram	<pre> classDiagram class StandardTerminalNameAssignment { <<TwoTerminalResistorStandardTerminalNameAssignmentType>> } class Mapping { <<TwoTerminalResistorStandardTerminalMappingType>> } class StandardTerminalName { <<TwoTerminalResistorStandardTerminalNameType>> } StandardTerminalNameAssignment "1..>" -- "2..>" Mapping : "constraints" StandardTerminalNameAssignment "1..>" -- "2..>" StandardTerminalName : "constraints" StandardTerminalNameAssignment "1..>" -- "1..>" Mapping : "constraints" StandardTerminalNameAssignment "1..>" -- "1..>" StandardTerminalName : "constraints" StandardTerminalNameAssignment "1..>" -- "1..>" StandardTerminalName : "constraints" StandardTerminalNameAssignment "1..>" -- "1..>" StandardTerminalName : "constraints" </pre>				
type	TwoTerminalResistorStandardTerminalNameAssignmentType, TwoTerminalResistorStandardTerminalMappingType, TwoTerminalResistorStandardTerminalNameType.				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">1. Terminal 1</td> <td style="padding: 5px;">2. Terminal 2</td> <td style="padding: 5px;"></td> <td style="padding: 5px;"></td> </tr> </table>	1. Terminal 1	2. Terminal 2		
1. Terminal 1	2. Terminal 2				

4.7.3.14.2. Potentiometer Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/PotentiometerFunction						
diagram	<pre> classDiagram class StandardTerminalNameAssignment { <<PotentiometerStandardTerminalNameAssignmentType>> } class Mapping { <<PotentiometerStandardTerminalMappingType>> } class StandardTerminalName { <<PotentiometerStandardTerminalNameType>> } StandardTerminalNameAssignment "3" --> Mapping Mapping "3" --> StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for a PotentiometerFunction. It features three main classes: StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The StandardTerminalNameAssignment class is associated with the Mapping class, which in turn is associated with the StandardTerminalName class. A dashed box labeled 'constraints' surrounds the Mapping class.</p>						
type	PotentiometerFunctionType , PotentiometerStandardTerminalNameAssignmentType , PotentiometerStandardTerminalMappingType , PotentiometerStandardTerminalNameType .						
list of enumerate values	<table border="1"> <tr> <td>PotentiometerStandardTerminalName</td> <td></td> <td></td> </tr> <tr> <td>1. Terminal 1</td> <td>2. Terminal 2</td> <td>3. Wiper</td> </tr> </table>	PotentiometerStandardTerminalName			1. Terminal 1	2. Terminal 2	3. Wiper
PotentiometerStandardTerminalName							
1. Terminal 1	2. Terminal 2	3. Wiper					

Internal complex array functions as shown in Figure 41 can be described in the XML file as follows:

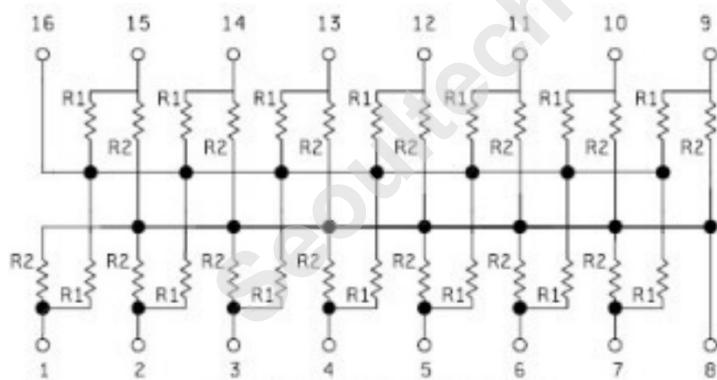


Figure 41 — Pull-up Pull-down Resistor Array

4.7.3.14.2 Potentiometer Function (cont'd)

```
<FunctionGroup-Array>
  <Function>
    <Resistor>
      <StandardTerminalNameAssignment>
        <Terminal1>
          <TerminalNumber>1</TerminalNumber>
        </Terminal1>
        <Terminal2>
          <TerminalNumber>16</TerminalNumber>
        </Terminal2>
      </StandardTerminalNameAssignment>
      :
      :
      <StandardTerminalNameAssignment>
        <Terminal1>
          <TerminalNumber>15</TerminalNumber>
        </Terminal1>
        <Terminal2>
          <TerminalNumber>16</TerminalNumber>
        </Terminal2>
      </StandardTerminalNameAssignment>
    </Resistor>
    <ElectricalSpecificationID>Res R1</ElectricalSpecificationID>
  </Function>
  <Function>
    <Resistor>
      <StandardTerminalNameAssignment>
        <Terminal1>
          <TerminalNumber>1</TerminalNumber>
        </Terminal1>
        <Terminal2>
          <TerminalNumber>8</TerminalNumber>
        </Terminal2>
      </StandardTerminalNameAssignment>
      :
      :
      <StandardTerminalNameAssignment>
        <Terminal1>
          <TerminalNumber>15</TerminalNumber>
        </Terminal1>
        <Terminal2>
          <TerminalNumber>8</TerminalNumber>
        </Terminal2>
      </StandardTerminalNameAssignment>
    </Resistor>
    <ElectricalSpecificationID>Res R2</ElectricalSpecificationID>
  </Function>
</FunctionGroup-Array>
```

4.7.3.15. RF

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF
diagram	<pre> graph TD RF[RF type RF-FunctionType] --- RF_FT[RF-FunctionType] subgraph RF_FT [RF-FunctionType] Antenna[Antenna type AntennaFunctionType] Attenuator[Attenuator type AttenuatorFunctionType] Balun[Balun type BalunFunctionType] Circulator[Circulator type CirculatorFunctionType] Coupler[Coupler type CouplerFunctionType] DCBlock[DCBlock type DCBlockType] Detector[Detector type DetectorFunctionType] Divider[Divider type DividerFunctionType] Isolator[Isolator type IsolatorFunctionType] Limiter[Limiter type LimiterFunctionType] Mixer[Mixer type MixerFunctionType] Modulator[Modulator type ModulatorType] Demodulator[Demodulator type DemodulatorType] Multiplier[Multiplier type MultiplierFunctionType] PhaseDetector[PhaseDetector type PhaseDetectorType] PhaseShifter[PhaseShifter type PhaseShifterFunctionType] end </pre>
type	FunctionType, AntennaFunctionType, AttenuatorFunctionType, BalunFunctionType, CirculatorFunctionType, CouplerFunctionType, DCBlockType, DetectorFunctionType, DividerFunctionType, IsolatorFunctionType, LimiterFunctionType, MixerFunctionType, ModulatorType, DemodulatorType, MultiplierFunctionType, PhaseDetectorType, PhaseShifterFunctionType.

4.7.3.15.1. Antenna

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Antenna												
diagram	<pre> classDiagram class AntennaFunctionType { <<Antenna>> <<AntennaStandardTerminalNameAssignmentType>> <<AntennaMandatoryStandardTerminalMappingType>> <<AntennaOptionalStandardTerminalMappingType>> <<StandardTerminalName>> <<OptionalMapping>> <<MandatoryMapping>> <<constraints>> } AntennaFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" Mapping Mapping "*" StandardTerminalName StandardTerminalNameAssignment "*" MandatoryMapping MandatoryMapping "*" StandardTerminalName StandardTerminalNameAssignment "*" OptionalMapping OptionalMapping "*" StandardTerminalName </pre>												
type	AntennaFunctionType , AntennaStandardTerminalNameAssignmentType , AntennaMandatoryStandardTerminalMappingType , AntennaMandatoryStandardTerminalNameType , AntennaOptionalStandardTerminalMappingType , AntennaOptionalStandardTerminalNameType .												
list of enumerate values	<table border="1"> <tr> <td>MandatoryMapping/StandardTerminalName</td> <td></td> <td></td> </tr> <tr> <td>1. Port 1</td> <td></td> <td></td> </tr> <tr> <td>OptionalMapping/StandardTerminalName</td> <td></td> <td></td> </tr> <tr> <td>1. Port2</td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName			1. Port 1			OptionalMapping/StandardTerminalName			1. Port2		
MandatoryMapping/StandardTerminalName													
1. Port 1													
OptionalMapping/StandardTerminalName													
1. Port2													

4.7.3.15.2. Attenuator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Attenuator						
diagram	<pre> classDiagram class AttenuatorFunctionType { <<Attenuator>> <<AttenuatorStandardTerminalNameAssignmentType>> <<AttenuatorStandardTerminalMappingType>> <<StandardTerminalName>> <<Mapping>> <<constraints>> } AttenuatorFunctionType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "*" Mapping Mapping "*" StandardTerminalName StandardTerminalNameAssignment "*" constraints </pre>						
type	AttenuatorFunctionType , AttenuatorStandardTerminalNameAssignmentType , AttenuatorStandardTerminalMappingType , AttenuatorStandardTerminalNameType .						
list of enumerate values	<table border="1"> <tr> <td>Mapping/StandardTerminalName</td> <td></td> <td></td> </tr> <tr> <td>1. Port 1</td> <td>2. Port 2</td> <td>3. Ground</td> </tr> </table>	Mapping/StandardTerminalName			1. Port 1	2. Port 2	3. Ground
Mapping/StandardTerminalName							
1. Port 1	2. Port 2	3. Ground					

4.7.3.15.3. Balun

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Balun			
diagram	<pre> classDiagram class Balun { <<BalunFunctionType>> } class StandardTerminalNameAssignment { <<BalunStandardTerminalNameAssignmentType>> } class Mapping { <<BalunStandardTerminalMappingType>> } class StandardTerminalName { <<BalunStandardTerminalNameType>> } Balun "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "4" StandardTerminalName </pre> <p>The diagram illustrates the structure of the BalunFunctionType. It consists of four main components: Balun, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The Balun component is associated with the StandardTerminalNameAssignment component (multiplicity 1..infinity). The StandardTerminalNameAssignment component is associated with the Mapping component (multiplicity 1..infinity). The Mapping component is associated with the StandardTerminalName component (multiplicity 4). A constraint labeled 'constraints' is present.</p>			
type	BalunFunctionType, BalunStandardTerminalNameAssignmentType, BalunStandardTerminalMappingType, BalunStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName 1. Balanced 1 2. Balanced 2 3. Unbalanced 4. Ground			

4.7.3.15.4. Circulator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Circulator			
diagram	<pre> classDiagram class Circulator { <<CirculatorFunctionType>> } class StandardTerminalNameAssignment { <<CirculatorStandardTerminalNameAssignmentType>> } class MandatoryMapping { <<CirculatorMandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<CirculatorMandatoryStandardTerminalNameType>> } Circulator "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" MandatoryMapping MandatoryMapping "4" StandardTerminalName </pre> <p>The diagram illustrates the structure of the CirculatorFunctionType. It consists of four main components: Circulator, StandardTerminalNameAssignment, MandatoryMapping, and StandardTerminalName. The Circulator component is associated with the StandardTerminalNameAssignment component (multiplicity 1..infinity). The StandardTerminalNameAssignment component is associated with the MandatoryMapping component (multiplicity 1..infinity). The MandatoryMapping component is associated with the StandardTerminalName component (multiplicity 4). A constraint labeled 'constraints' is present.</p>			
type	CirculatorFunctionType, CirculatorStandardTerminalNameAssignmentType, CirculatorMandatoryStandardTerminalMappingType, CirculatorMandatoryStandardTerminalNameType, CirculatorOptionalStandardTerminalMappingType, CirculatorOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName 1. Port 1 2. Port 2 3. Port 3 4. Ground OptionalMapping/StandardTerminalName 1. Port 4			

4.7.3.15.5. Coupler

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Coupler
diagram	<pre> classDiagram Coupler < -- CouplerFunctionType Properties < -- CouplerPropertiesType DCPassthrough < -- xs:boolean BiDirectional < -- BiDirectionalCouplerType Directional < -- DirectionalCouplerType Coupler "2..1" --> Properties Coupler "2..1" --> BiDirectional Coupler "2..1" --> Directional Properties "1..1" --> DCPassthrough </pre>
type	CouplerFunctionType , CouplerPropertiesType , BiDirectionalCouplerType , DirectionalCouplerType .

4.7.3.15.5.1. BiDirectional

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Coupler/BiDirectional															
diagram	<pre> classDiagram BiDirectional < -- BiDirectionalCouplerType StandardTerminalNameAssignment < -- BiDirectionalCouplerStandardTerminalNameAssignmentType Mapping < -- BiDirectionalCouplerStandardTerminalMappingType StandardTerminalName < -- BiDirectionalCouplerStandardTerminalNameType BiDirectional "1..1" --> StandardTerminalNameAssignment StandardTerminalNameAssignment "1..1" --> Mapping Mapping "1..1" --> StandardTerminalName </pre>															
type	BiDirectionalCouplerType , BiDirectionalCouplerStandardTerminalNameAssignmentType , BiDirectionalCouplerStandardTerminalMappingType , BiDirectionalCouplerStandardTerminalNameType .															
list of enumerate values	<table border="1"> <tr> <td colspan="5">Mapping/StandardTerminalName</td> </tr> <tr> <td>1. Input</td> <td>2. Output</td> <td>3. Coupled Forward</td> <td>4. Coupled Reverse</td> <td></td> </tr> <tr> <td>5. Ground</td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	Mapping/StandardTerminalName					1. Input	2. Output	3. Coupled Forward	4. Coupled Reverse		5. Ground				
Mapping/StandardTerminalName																
1. Input	2. Output	3. Coupled Forward	4. Coupled Reverse													
5. Ground																

4.7.3.15.5.2. Directional

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Coupler/Directional																
diagram																	
type	DirectionalCouplerType , DirectionalCouplerStandardTerminalNameAssignmentType , ConnectionType .																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Input</td> <td>2. Output</td> <td>3. Coupled Forward</td> <td>4. Ground</td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Termination</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Input	2. Output	3. Coupled Forward	4. Ground	OptionalMapping/StandardTerminalName				1. Termination			
MandatoryMapping/StandardTerminalName																	
1. Input	2. Output	3. Coupled Forward	4. Ground														
OptionalMapping/StandardTerminalName																	
1. Termination																	

4.7.3.15.6. DC Block

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/DCBlock																
diagram																	
type	DCBlockType , DCBlockStandardTerminalNameAssignmentType , DCBlockMandatoryStandardTerminalMappingType , DCBlockMandatoryStandardTerminalNameType , DCBlockOptionalStandardTerminalMappingType , DCBlockOptionalStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Port 1</td> <td>2. Port 2</td> <td></td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>1. Ground</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Port 1	2. Port 2			OptionalMapping/StandardTerminalName				1. Ground			
MandatoryMapping/StandardTerminalName																	
1. Port 1	2. Port 2																
OptionalMapping/StandardTerminalName																	
1. Ground																	

4.7.3.15.7. Detector

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Detector																
diagram	<p>The diagram illustrates the class structure for the Detector Function Type. It features a main class, DetectorFunctionType, which contains a multiplicity of StandardTerminalNameAssignment objects, indicated by the 1..* symbol. This association is labeled 1..*. Within each StandardTerminalNameAssignment object, there are two types of mappings: MandatoryMapping and OptionalMapping. Each mapping is associated with a StandardTerminalName object. The MandatoryMapping class has three associations labeled 1, 2, and 3, which correspond to the Input 1, Output, and Ground roles respectively. The OptionalMapping class also has three associations labeled 1, 2, and 3, corresponding to the Input 2, Output, and Ground roles. A constraint block labeled constraints is located at the bottom right of the diagram area.</p>																
type	DetectorFunctionType , DetectorStandardTerminalNameAssignmentType , DetectorMandatoryStandardTerminalMappingType , DetectorMandatoryStandardTerminalNameType , DetectorOptionalStandardTerminalMappingType , DetectorOptionalStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Input 1</td> <td>2. Output</td> <td>3. Ground</td> <td></td> </tr> <tr> <td colspan="4">OptionalMapping/StandardTerminalName</td> </tr> <tr> <td>4. Input 2</td> <td></td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Input 1	2. Output	3. Ground		OptionalMapping/StandardTerminalName				4. Input 2			
MandatoryMapping/StandardTerminalName																	
1. Input 1	2. Output	3. Ground															
OptionalMapping/StandardTerminalName																	
4. Input 2																	

4.7.3.15.8. Divider

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Divider</code>																												
diagram																													
type	DividerFunctionType , DividerPropertiesType , DividerStandardTerminalNameAssignmentType , DividerMandatoryStandardTerminalMappingType , DividerMandatoryStandardTerminalNameType , DividerOptionalStandardTerminalMappingType , DividerOptionalStandardTerminalNameType .																												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="4">MandatoryMapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. SUM</td><td>2. Port 1</td><td>3. Port 2</td><td>4. Ground</td></tr> <tr> <th colspan="4">OptionalMapping/StandardTerminalName</th> </tr> <tr> <td>1. Port 3</td><td>2. Port 4</td><td>3. Port 5</td><td>4. Port 6</td></tr> <tr> <td>5. Port 7</td><td>6. Port 8</td><td>7. Port 9</td><td>8. Port 10</td></tr> <tr> <td>9. Port 11</td><td>10. Port 12</td><td>11. Port 13</td><td>12. Port 14</td></tr> <tr> <td>13. Port 15</td><td>14. Port 16</td><td></td><td></td></tr> </tbody> </table>	MandatoryMapping/StandardTerminalName				1. SUM	2. Port 1	3. Port 2	4. Ground	OptionalMapping/StandardTerminalName				1. Port 3	2. Port 4	3. Port 5	4. Port 6	5. Port 7	6. Port 8	7. Port 9	8. Port 10	9. Port 11	10. Port 12	11. Port 13	12. Port 14	13. Port 15	14. Port 16		
MandatoryMapping/StandardTerminalName																													
1. SUM	2. Port 1	3. Port 2	4. Ground																										
OptionalMapping/StandardTerminalName																													
1. Port 3	2. Port 4	3. Port 5	4. Port 6																										
5. Port 7	6. Port 8	7. Port 9	8. Port 10																										
9. Port 11	10. Port 12	11. Port 13	12. Port 14																										
13. Port 15	14. Port 16																												

4.7.3.15.9. Isolator

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Isolator</code>						
diagram							
type	IsolatorFunctionType , IsolatorStandardTerminalNameAssignmentType , IsolatorStandardTerminalMappingType , IsolatorStandardTerminalNameType .						
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="3">Mapping/StandardTerminalName</th> </tr> </thead> <tbody> <tr> <td>1. Input</td><td>2. Output</td><td>3. Ground</td></tr> </tbody> </table>	Mapping/StandardTerminalName			1. Input	2. Output	3. Ground
Mapping/StandardTerminalName							
1. Input	2. Output	3. Ground					

4.7.3.15.10. Limiter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Limiter				
diagram	<pre> classDiagram class LimiterFunctionType { Limiter StandardTerminalNameAssignment Mapping StandardTerminalName } class LimiterStandardTerminalNameAssignmentType { LimiterStandardTerminalNameAssignment } class LimiterStandardTerminalMappingType { LimiterStandardTerminalMapping } class StandardTerminalNameType { StandardTerminalName } Limiter --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> StandardTerminalName </pre>				
type	LimiterFunctionType, LimiterStandardTerminalNameAssignmentType, LimiterStandardTerminalMappingType, LimiterStandardTerminalNameType.				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Input</td> <td>2. Output</td> <td>3. Ground</td> <td></td> </tr> </table>	1. Input	2. Output	3. Ground	
1. Input	2. Output	3. Ground			

4.7.3.15.11. Mixer

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Mixer								
diagram	<pre> classDiagram class MixerFunctionType { Mixer StandardTerminalNameAssignment MandatoryMapping StandardTerminalName OptionalMapping } class MixerStandardTerminalNameAssignmentType { MixerStandardTerminalNameAssignment } class MixerMandatoryStandardTerminalMappingType { MixerMandatoryStandardTerminalMapping } class MixerOptionalStandardTerminalMappingType { MixerOptionalStandardTerminalMapping } class StandardTerminalNameType { StandardTerminalName } Mixer --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> MandatoryMapping MandatoryMapping --> StandardTerminalName StandardTerminalNameAssignment --> OptionalMapping </pre>								
type	MixerFunctionType, MixerStandardTerminalNameAssignmentType, MixerMandatoryStandardTerminalMappingType, MixerMandatoryStandardTerminalNameType.								
list of enumerate values	MandatoryMapping/StandardTerminalName <table border="1"> <tr> <td>1. RF</td> <td>2. LO</td> <td>3. IF</td> <td>4. Ground</td> </tr> </table> OptionalMapping/StandardTerminalName <table border="1"> <tr> <td>1. PowerLO</td> <td></td> <td></td> <td></td> </tr> </table>	1. RF	2. LO	3. IF	4. Ground	1. PowerLO			
1. RF	2. LO	3. IF	4. Ground						
1. PowerLO									

4.7.3.15.12. Modulator

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator</code>
diagram	<pre> classDiagram class Modulator { <<ModulatorType>> } class IQ { <<IQ-ModulatorType>> } class QPSK { <<QPSK-ModulatorType>> } Modulator "1" --> IQ IQ "1" --> IQ IQ "1" --> QPSK </pre>
type	ModulatorType, IQ-ModulatorType, IQ-ModulatorStandardTerminalNameAssignmentType, QPSK-ModulatorType, QPSK-ModulatorStandardTerminalNameAssignmentType, ConnectionType.

4.7.3.15.12.1. IQ - Modulator Standard Terminal Name Assignment

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/IQ</code>				
diagram	<pre> classDiagram class Modulator { <<ModulatorType>> } class IQ { <<IQ-ModulatorType>> } class I { <<I-SingleDifferentialConnectionType>> } class Q { <<Q-SingleDifferentialConnectionType>> } class Mapping { <<IQ-ModulatorMandatoryStandardTerminalMappingType>> } class StandardTerminalName { <<IQ-ModulatorMandatoryStandardTerminalName>> } Modulator "1" --> IQ IQ "1" --> IQ IQ "1" --> I IQ "1" --> Q I "1" --> Mapping Q "1" --> Mapping Mapping "1" --> StandardTerminalName </pre>				
type	IQ-ModulatorStandardTerminalNameAssignmentType, I-SingleDifferentialConnectionType, Q-SingleDifferentialConnectionType, IQ-ModulatorMandatoryStandardTerminalMappingType, IQ-ModulatorMandatoryStandardTerminalNameType.				
list of enumerate values	IQ-ModulatorMandatoryStandardTerminalMapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">1. RF</td> <td style="padding: 5px;">2. LO</td> <td style="padding: 5px;">3. Ground</td> <td style="padding: 5px;"></td> </tr> </table>	1. RF	2. LO	3. Ground	
1. RF	2. LO	3. Ground			

4.7.3.15.12.1.1. I-SingleDifferentialConnectionType

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/IQ/StandardTerminalNameAssignment/I 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Demodulator/IQ/StandardTerminalNameAssignment/I 												
diagram	<pre> classDiagram class I_SingleDifferentialConnectionType { SingleEndedConnectionMapping DifferentialConnectionMapping } class I_DifferentialConnectionType { I_DifferentialConnectionMapping DifferentialPairname } I_SingleDifferentialConnectionType "1..1" --> "0..1" IQ_Modulator_I_SingleEndedConnectionStandardTerminalMappingType : StandardTerminalName I_SingleDifferentialConnectionType "1..1" --> "0..1" I_DifferentialConnectionType : I_DifferentialConnectionMapping I_DifferentialConnectionType "0..1" --> "1..1" IQ_Modulator_I_SingleEndedConnectionStandardTerminalMappingType : StandardTerminalName I_DifferentialConnectionType "0..1" --> "1..1" I_DifferentialConnectionMapping : DifferentialPairname </pre>												
type	I-SingleDifferentialConnectionType, IQ-Modulator-I-SingleEndedConnectionStandardTerminalMappingType, IQ-Modulator-I-SingleEndedConnectionStandardTerminalNameType, I-DifferentialConnectionType, I-DifferentialConnectionMappingType, I-DifferentialStandardTerminalNameType												
list of enumerate values	<p>SingleEndedConnectionMapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. I</td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="4">DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName</td> </tr> <tr> <td>1. I+</td> <td>2. I-</td> <td></td> <td></td> </tr> </table>	1. I				DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName				1. I+	2. I-		
1. I													
DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName													
1. I+	2. I-												

4.7.3.15.12.1.2. Q-SingleDifferentialConnectionType

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/IQ/StandardTerminalNameAssignment/Q 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Demodulator/IQ/StandardTerminalNameAssignment/Q 																
diagram	<pre> classDiagram class Q { <<Q-SingleDifferentialConnectionType>> } class IQModulatorQSingleEndedConnectionStandardTerminalMappingType { <<IQ-Modulator-Q-SingleEndedConnectionStandardTerminalMappingType>> } class StandardTerminalName { <<StandardTerminalName>> } class IQModulatorQSingleEndedConnectionStandardTerminalNameType { <<IQ-Modulator-Q-SingleEndedConnectionStandardTerminalNameType>> } class QDifferentialConnectionType { <<Q-DifferentialConnectionType>> } class QDifferentialConnectionMapping { <<Q-DifferentialConnectionMapping>> } class StandardTerminalName2 { <<StandardTerminalName>> } class DifferentialPairname { <<DifferentialPairname>> } Q "1" --> IQModulatorQSingleEndedConnectionStandardTerminalMappingType IQModulatorQSingleEndedConnectionStandardTerminalMappingType "1..>" StandardTerminalName IQModulatorQSingleEndedConnectionStandardTerminalMappingType "1..>" IQModulatorQSingleEndedConnectionStandardTerminalNameType IQModulatorQSingleEndedConnectionStandardTerminalNameType "1..>" StandardTerminalName2 IQModulatorQSingleEndedConnectionStandardTerminalNameType "1..>" QDifferentialConnectionType QDifferentialConnectionType "1..>" QDifferentialConnectionMapping QDifferentialConnectionMapping "1..>" DifferentialPairname </pre>																
type	SingleDifferentialConnectionType, IQ-Modulator-Q-SingleEndedConnectionStandardTerminalMappingType, IQ-Modulator-Q-SingleEndedConnectionStandardTerminalNameType, Q-DifferentialConnectionType, Q-DifferentialConnectionMappingType, Q-DifferentialStandardTerminalNameType																
list of enumerate values	<table border="1"> <tr> <td colspan="4">SingleEndedConnectionMapping/StandardTerminalName</td> </tr> <tr> <td colspan="4">1. Q</td> </tr> <tr> <td colspan="4">DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName</td> </tr> <tr> <td>1. Q+</td> <td>2. Q-</td> <td></td> <td></td> </tr> </table>	SingleEndedConnectionMapping/StandardTerminalName				1. Q				DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName				1. Q+	2. Q-		
SingleEndedConnectionMapping/StandardTerminalName																	
1. Q																	
DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName																	
1. Q+	2. Q-																

4.7.3.15.12.2. QPSK - Modulator Standard Terminal Name Assignment

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/QPSK												
diagram	<pre> classDiagram class QPSK { <<QPSK-ModulatorType>> } class StandardTerminalNameAssignment { <<StandardTerminalNameAssignment>> } class Mapping { <<Mapping>> } class StandardTerminalName { <<StandardTerminalName>> } QPSK "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName Note over StandardTerminalNameAssignment, Mapping: constraints </pre>												
type	QPSK-ModulatorType, QPSK-ModulatorStandardTerminalNameAssignmentType, QPSK-ModulatorStandardTerminalMappingType, QPSK-ModulatorStandardTerminalNameType.												
list of enumerate values	<table border="1"> <tr> <td colspan="4">Mapping/StandardTerminalName</td> </tr> <tr> <td>1. RF Input</td> <td>2. RF Output</td> <td>3. Ctrl 1</td> <td>4. Ctrl 2</td> </tr> <tr> <td>5. Ground</td> <td></td> <td></td> <td></td> </tr> </table>	Mapping/StandardTerminalName				1. RF Input	2. RF Output	3. Ctrl 1	4. Ctrl 2	5. Ground			
Mapping/StandardTerminalName													
1. RF Input	2. RF Output	3. Ctrl 1	4. Ctrl 2										
5. Ground													

4.7.3.15.13. Demodulator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Demodulator
diagram	
type	DemodulatorType, IQ-ModulatorType

4.7.3.15.14. Multiplier

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Multiplier				
diagram					
type	MultiplierFunctionType, MultiplierPropertiesType, MultiplierStandardTerminalNameAssignmentType, MultiplierStandardTerminalMappingType, MultiplierStandardTerminalNameType.				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. Input</td> <td>2. Output</td> <td>3. Ground</td> <td></td> </tr> </table>	1. Input	2. Output	3. Ground	
1. Input	2. Output	3. Ground			

4.7.3.15.15. Phase Detector

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/PhaseDetector				
diagram					
type	PhaseDetectorType, PhaseDetectorStandardTerminalNameAssignmentType, PhaseDetectorStandardTerminalMappingType, PhaseDetectorStandardTerminalNameType.				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. Input</td> <td>2. Reference</td> <td>3. Output</td> <td>4. Ground</td> </tr> </table>	1. Input	2. Reference	3. Output	4. Ground
1. Input	2. Reference	3. Output	4. Ground		

4.7.3.15.16. Phase Shifter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/PhaseShifter							
diagram	<pre> classDiagram class PhaseShifter { <<PhaseShifterFunctionType>> } class StandardTerminalNameAssignment { <<PhaseShifterStandardTerminalNameAssignmentType>> } class Mapping { <<PhaseShifterStandardTerminalMappingType>> } class StandardTerminalName { <<PhaseShifterStandardTerminalName>> } PhaseShifter "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" Mapping Mapping "1..>" StandardTerminalName </pre> <p>The diagram illustrates the UML class structure for a Phase Shifter. It consists of four classes: PhaseShifter, StandardTerminalNameAssignment, Mapping, and StandardTerminalName. The PhaseShifter class has a multiplicity of 1..> at its end of the association with StandardTerminalNameAssignment. The StandardTerminalNameAssignment class has a multiplicity of 1..> at its end of the association with Mapping. The Mapping class has a multiplicity of 1..> at its end of the association with StandardTerminalName. A constraint labeled 'constraints' is present on the association between StandardTerminalNameAssignment and Mapping.</p>							
type	PhaseShifterFunctionType , PhaseShifterStandardTerminalNameAssignmentType , ConnectionType .							
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Input</td> <td>2. Output</td> <td>3. Control</td> <td>4. Ground</td> </tr> </table>				1. Input	2. Output	3. Control	4. Ground
1. Input	2. Output	3. Control	4. Ground					

4.7.3.16. Source

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source			
diagram	<pre> classDiagram class Source { <<SourceFunctionType>> } class Battery { <<BatteryFunctionType>> } class Generic { <<GenericSourceFunctionType>> } class VoltageSource { <<VoltageSourceFunctionType>> } class CurrentSource { <<CurrentSourceFunctionType>> } class AC_VoltageSource { <<AC-SourceType>> } class DC_VoltageSource { <<DC-SourceType>> } class AC_CurrentSource { <<AC-SourceType>> } class DC_CurrentSource { <<DC-SourceType>> } Source "1..>" Battery Source "1..>" Generic Source "1..>" VoltageSource Source "1..>" CurrentSource </pre> <p>The diagram illustrates the UML class structure for a Source. It consists of eight classes: Source, Battery, Generic, VoltageSource, CurrentSource, AC_VoltageSource, DC_VoltageSource, AC_CurrentSource, and DC_CurrentSource. The Source class has a multiplicity of 1..> at its end of the associations with Battery, Generic, VoltageSource, and CurrentSource.</p>			
type	SourceFunctionType , BatteryFunctionType , GenericSourceFunctionType , VoltageSourceFunctionType , CurrentSourceFunctionType , AC-SourceType , DC-SourceType .			

4.7.3.16.1. Battery Source

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Battery			
diagram				
type	BatteryFunctionType , DC-SourceStandardTerminalNameAssignmentType , DC-SourceStandardTerminalMappingType , DC-SourceStandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. Positive 2. Negative			

4.7.3.16.2. AC - Source Type

path	1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/VoltageSource/AC-VoltageSource 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/CurrentSource/AC-VoltageSource			
diagram				
type	AC-SourceType , AC-SourceStandardTerminalNameAssignmentType , AC-SourceStandardTerminalMappingType , AC-SourceStandardTerminalNameType .			
list of enumerate values	Mapping/StandardTerminalName 1. Terminal 1 2. Terminal 2			

4.7.3.16.3. DC - Source Type

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/VoltageSource/DC-VoltageSource 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/CurrentSource/DC-VoltageSource 				
diagram	<pre> classDiagram class DC_VoltageSource { <<DC-SourceType>> } class StandardTerminalNameAssignmentType { <<DC-SourceStandardTerminalNameAssignmentType>> } class Mapping { <<DC-SourceStandardTerminalMappingType>> } class StandardTerminalName { <<DC-SourceStandardTerminalName>> <<constraints>> } class TerminalMapID { <<xs:string>> } DC_VoltageSource --> StandardTerminalNameAssignmentType : StandardTerminalNameAssignmentType --> Mapping : 1..x Mapping --> StandardTerminalName : 2 Mapping --> TerminalMapID : 2 </pre>				
type	DC-SourceType, DC-SourceStandardTerminalNameAssignmentType, DC-SourceStandardTerminalMappingType, DC-SourceStandardTerminalNameType.				
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Positive</td> <td style="padding: 2px;">2. Negative</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>	1. Positive	2. Negative		
1. Positive	2. Negative				

4.7.3.17. Switch

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch
diagram	<pre> classDiagram class Switch { <<SwitchFunctionType>> } class SwitchFunctionType { <<SwitchFunctionType>> SingleThrow DoubleThrow ManyThrow } Switch --> SwitchFunctionType : 1..x </pre>
type	SwitchFunctionType, SingleThrowSwitchType, DoubleThrowSwitchType, ManyThrowSwitchType.

4.7.3.17.1. Single Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch/SingleThrow																
diagram																	
type	SingleThrowSwitchType , SingleThrowSwitchStandardTerminalNameAssignmentType , SingleThrowSwitchChoiceStandardTerminalMappingType , SingleThrowSwitchChoiceStandardTerminalNameType SingleThrowSwitchMandatoryStandardTerminalMappingType SingleThrowSwitchMandatoryStandardTerminalNameType .																
list of enumerate values	<table border="1"> <tr> <td>ChoiceMapping/StandardTerminalName</td> <td></td> <td></td> <td></td> </tr> <tr> <td>1. Normally Closed</td> <td>2. Normally Open</td> <td></td> <td></td> </tr> <tr> <td>MandatoryMapping/StandardTerminalName</td> <td></td> <td></td> <td></td> </tr> <tr> <td>1. Pole</td> <td></td> <td></td> <td></td> </tr> </table>	ChoiceMapping/StandardTerminalName				1. Normally Closed	2. Normally Open			MandatoryMapping/StandardTerminalName				1. Pole			
ChoiceMapping/StandardTerminalName																	
1. Normally Closed	2. Normally Open																
MandatoryMapping/StandardTerminalName																	
1. Pole																	

4.7.3.17.2. Double Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch/DoubleThrow								
diagram									
type	DoubleThrowSwitchType , DoubleThrowSwitchStandardTerminalNameAssignmentType , DoubleThrowSwitchStandardTerminalMappingType , DoubleThrowSwitchStandardTerminalNameType .								
list of enumerate values	<table border="1"> <tr> <td>Mapping/StandardTerminalName</td> <td></td> <td></td> <td></td> </tr> <tr> <td>1. Normally Closed</td> <td>2. Normally Open</td> <td>3. Pole</td> <td></td> </tr> </table>	Mapping/StandardTerminalName				1. Normally Closed	2. Normally Open	3. Pole	
Mapping/StandardTerminalName									
1. Normally Closed	2. Normally Open	3. Pole							

4.7.3.17.3. Many Throw

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch/SingleThrow</code>												
diagram	<pre> classDiagram class ManyThrowSwitchType class StandardTerminalNameAssignment class ManyThrowSwitchStandardTerminalNameAssignmentType { MultiMapping MandatoryMapping } class StandardTerminalName ManyThrowSwitchType "1..cc" --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> ManyThrowSwitchStandardTerminalNameAssignmentType ManyThrowSwitchStandardTerminalNameAssignmentType "1..cc" --> MultiMapping ManyThrowSwitchStandardTerminalNameAssignmentType "2" --> MandatoryMapping MultiMapping --> StandardTerminalName MandatoryMapping --> StandardTerminalName </pre>												
type	<code>ManyThrowSwitchType</code> , <code>ManyThrowSwitchStandardTerminalNameAssignmentType</code> , <code>ManyThrowSwitchNormallyOpenStandardTerminalMappingType</code> , <code>ManyThrowSwitchNormallyOpenStandardTerminalNameType</code> , <code>ManyThrowSwitchMandatoryStandardTerminalMappingType</code> , <code>ManyThrowSwitchMandatoryStandardTerminalNameType</code> .												
list of enumerate values	<table border="1"> <tr> <td><code>MultiMapping/StandardTerminalName</code></td> <td></td> <td></td> </tr> <tr> <td>1. Normally Open</td> <td></td> <td></td> </tr> <tr> <td><code>MandatoryMapping/StandardTerminalName</code></td> <td></td> <td></td> </tr> <tr> <td>1. Normally Closed</td> <td>2. Pole</td> <td></td> </tr> </table>	<code>MultiMapping/StandardTerminalName</code>			1. Normally Open			<code>MandatoryMapping/StandardTerminalName</code>			1. Normally Closed	2. Pole	
<code>MultiMapping/StandardTerminalName</code>													
1. Normally Open													
<code>MandatoryMapping/StandardTerminalName</code>													
1. Normally Closed	2. Pole												

4.7.3.18. Thyristor

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor</code>
diagram	<pre> classDiagram class Thyristor class BidirectionalThyristorType { Diode Triode } class UnidirectionalThyristorType { ReverseBlocking ReverseConducting } Thyristor --> BidirectionalThyristorType Thyristor --> UnidirectionalThyristorType </pre>
type	<code>ThyristorFunctionType</code> , <code>BidirectionalThyristorType</code> , <code>BidirectionalThyristorDiodeStandardTerminalNameAssignmentType</code> , <code>BidirectionalThyristorTriodeType</code> , <code>UnidirectionalThyristorType</code> , <code>UnidirectionalReverseBlockingThyristorType</code> , <code>UnidirectionalReverseConductingThyristorType</code> .

4.7.3.18.1. Bidirectional Thyristor Diode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Bidirectional/Diode						
diagram							
type	BidirectionalThyristorDiodeType , BidirectionalThyristorDiodeStandardTerminalNameAssignmentType , BidirectionalThyristorDiodeStandardTerminalMappingType , BidirectionalThyristorDiodeStandardTerminalNameType .						
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Terminal 1</td> <td style="padding: 2px;">2. Terminal 2</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </table>			1. Terminal 1	2. Terminal 2		
1. Terminal 1	2. Terminal 2						

4.7.3.18.2. Bidirectional Thyristor Triode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Bidirectional/Triode						
diagram							
type	BidirectionalThyristorDiodeType , BidirectionalThyristorDiodeStandardTerminalNameAssignmentType , BidirectionalThyristorTriodeStandardTerminalMappingType , BidirectionalThyristorTriodeStandardTerminalNameType .						
list of enumerate values	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Terminal 1</td> <td style="padding: 2px;">2. Terminal 2</td> <td style="padding: 2px;">3. Gate</td> <td style="padding: 2px;"></td> </tr> </table>			1. Terminal 1	2. Terminal 2	3. Gate	
1. Terminal 1	2. Terminal 2	3. Gate					

4.7.3.18.3. Unidirectional

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional
diagram	<pre> classDiagram class UnidirectionalThyristorType { <<UnidirectionalThyristorType>> } class ReverseBlocking { <<UnidirectionalReverseBlockingThyristorType>> } class ReverseConducting { <<UnidirectionalReverseConductingThyristorType>> } class Unidirectional { <<UnidirectionalThyristorType>> } UnidirectionalThyristorType < -- ReverseBlocking UnidirectionalThyristorType < -- ReverseConducting UnidirectionalThyristorType < -- Unidirectional ReverseBlocking --> Diode ReverseBlocking --> Triode ReverseConducting --> Diode ReverseConducting --> Triode </pre>
type	UnidirectionalThyristorType , UnidirectionalReverseBlockingThyristorType , UnidirectionalReverseConductingThyristorType .

4.7.3.18.3.1. Reverse Blocking - Diode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseBlocking/Diode				
diagram	<pre> classDiagram class UnidirectionalReverseBlockingThyristorDiodeType { <<UnidirectionalReverseBlockingThyristorDiodeType>> } class StandardTerminalNameAssignment { <<DiodeStandardTerminalNameAssignmentType>> } class Mapping { <<DiodeStandardTerminalMappingType>> } class StandardTerminalName { <<DiodeStandardTerminalNameType>> } UnidirectionalReverseBlockingThyristorDiodeType --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> DiodeStandardTerminalName UnidirectionalReverseBlockingThyristorDiodeType --> Mapping Mapping --> StandardTerminalName Mapping --> constraints </pre>				
type	UnidirectionalReverseBlockingThyristorDiodeType , DiodeStandardTerminalNameAssignmentType , DiodeStandardTerminalMappingType , DiodeStandardTerminalNameType .				
list of enumerate values	<p>Mapping/StandardTerminalName</p> <table border="1"> <tr> <td>1. Anode</td> <td>2. Cathode</td> <td></td> <td></td> </tr> </table>	1. Anode	2. Cathode		
1. Anode	2. Cathode				

4.7.3.18.3.2. Reverse Blocking - Triode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseBlocking/Triode	
diagram		
type	UnidirectionalReverseBlockingThyristorTriodeType, UnidirectionalTriodeStandardTerminalNameAssignmentType, UnidirectionalTriodeStandardTerminalMappingType, UnidirectionalTriodeStandardTerminalNameType.	
list of enumerate values	Mapping/StandardTerminalName 1. Anode 2. Cathode 3. Gate	

4.7.3.18.3.3. Reverse Conducting - Diode

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseConducting/Diode	
diagram		
type	UnidirectionalReverseConductingThyristorDiodeType, DiodeStandardTerminalNameAssignmentType, DiodeStandardTerminalMappingType, DiodeStandardTerminalNameType.	
list of enumerate values	Mapping/StandardTerminalName 1. Anode 2. Cathode	

4.7.3.18.3.4. Reverse Conducting - Triode

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseConducting/Triode</code>						
diagram							
type	<code>UnidirectionalReverseConductingThyristorTriodeType,</code> <code>UnidirectionalTriodeStandardTerminalNameAssignmentType,</code> <code>UnidirectionalTriodeStandardTerminalMappingType, UnidirectionalTriodeStandardTerminalNameType.</code>						
list of enumerate values	<table border="1"> <tr> <td>Mapping/StandardTerminalName</td> <td></td> </tr> <tr> <td>1. Anode</td> <td>2. Cathode</td> </tr> <tr> <td>3. Gate</td> <td></td> </tr> </table>	Mapping/StandardTerminalName		1. Anode	2. Cathode	3. Gate	
Mapping/StandardTerminalName							
1. Anode	2. Cathode						
3. Gate							

4.7.3.19. Transformer

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transformer</code>
diagram	
type	<code>TransformerFunctionType, TransformerCoil-to-Coil-ArrayType, TransformerCoil-to-CoilFunctionType,</code> <code>TransformerFromCoilType, TransformerToCoilType, TransformerCoil-ArrayType,</code> <code>TransformerCoilType, TransformerStandardTerminalNameAssignmentType,</code> <code>TransformerStandardTerminalNameAssignmentType.</code>

4.7.3.19.1. Transformer Coil Mapping

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transformer			
diagram	<pre> classDiagram class TransformerCoilType { ID ElectricalSpecificationID } class StandardTerminalNameAssignment { StandardTerminalNameAssignment } class Mapping { constraints } class StandardTerminalName TransformerCoilType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "0..>" Mapping Mapping "1..>" StandardTerminalName </pre>			
type	TransformerCoilType, TransformerStandardTerminalNameAssignmentType, TransformerStandardTerminalMappingType, TransformerStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Start Wiring</td> <td>2. End Wiring</td> <td></td> </tr> </table>	1. Start Wiring	2. End Wiring	
1. Start Wiring	2. End Wiring			

4.7.3.20. Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor
diagram	<pre> classDiagram class Transistor class TransistorFunctionType { BipolarJunction Unijunction FieldEffect Insulated-GateBipolar ProgrammableUnijunction } Transistor "1..>" TransistorFunctionType </pre>
type	TransistorFunctionType, BipolarJunctionTransistorFunctionType, UnijunctionTransistorFunctionType, FieldEffectTransistorFunctionType, GateBipolarTransistorFunctionType, ProgrammableUnijunctionTransistorFunctionType.

4.7.3.20.1. Bipolar Junction Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction
diagram	<pre> classDiagram class BipolarJunction { <<BipolarJunctionTransistorFunctionType>> } class NPN { <<NPN-BipolarJunctionTransistorType>> } class PNP { <<PNP-BipolarJunctionTransistorType>> } class NPN_Darlington { <<NPN-DarlingtonFunctionType>> } class PNP_Darlington { <<PNP-DarlingtonFunctionType>> } class StandardTerminalNameAssignment { <<BipolarTransistorStandardTerminalNameAssignmentType>> } class BipolarTransistorStandardTerminalNameAssignmentType { <<PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction>> } class BJT_StandardTerminalMappingType { <<BJT-StandardTerminalMappingType>> } class BJT_StandardTerminalNameType { <<BJT-StandardTerminalNameType>> } BipolarJunction "1..∞" --> NPN BipolarJunction "1..∞" --> PNP BipolarJunction "1..∞" --> NPN_Darlington BipolarJunction "1..∞" --> PNP_Darlington NPN "1..∞" --> StandardTerminalNameAssignment PNP "1..∞" --> StandardTerminalNameAssignment NPN_Darlington "1..∞" --> StandardTerminalNameAssignment PNP_Darlington "1..∞" --> StandardTerminalNameAssignment </pre>
type	BipolarJunctionTransistorFunctionType , NPN-BipolarJunctionTransistorType , PNP-BipolarJunctionTransistorType , BipolarTransistorStandardTerminalNameAssignmentType , NPN-DarlingtonFunctionType , PNP-DarlingtonFunctionType , DarlingtonTransistorStandardTerminalNameAssignmentType .

A [BipolarJunction](#) can have one of the following four types: [NPN](#), [PNP](#), [NPN-Darlington](#), and [PNP-Darlington](#). These four types have two different terminal name assignment as shown below.

4.7.3.20.1.1. Bipolar Transistor Standard Terminal Name Assignment

path	<ol style="list-style-type: none"> PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/NPN/StandardTerminalNameAssignment PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/PNP/StandardTerminalNameAssignment 						
diagram	<pre> classDiagram class StandardTerminalNameAssignment { <<BipolarTransistorStandardTerminalNameAssignmentType>> } class Mapping { <<BJT-StandardTerminalMappingType>> } class StandardTerminalName { <<BJT-StandardTerminalNameType>> } class BJT_StandardTerminalMappingType { <<PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction>> } StandardTerminalNameAssignment "1..∞" --> Mapping Mapping "3" --> StandardTerminalName </pre>						
type	BipolarTransistorStandardTerminalNameAssignmentType , BJT-StandardTerminalMappingType , BJT-StandardTerminalNameType .						
list of enumerate values	<table border="1"> <tr> <td>Mapping/StandardTerminalName</td> <td></td> </tr> <tr> <td>1. Base</td> <td>2. Collector</td> <td>3. Emitter</td> <td></td> </tr> </table>	Mapping/StandardTerminalName		1. Base	2. Collector	3. Emitter	
Mapping/StandardTerminalName							
1. Base	2. Collector	3. Emitter					

4.7.3.20.1.2. Darlington Transistor Standard Terminal Name Assignment

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/NPN-Darlington 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/PNP-Darlington 																								
diagram	<pre> classDiagram DarlingtonTransistorStandardTerminalNameAssignmentType { <<DarlingtonTransistorStandardTerminalNameAssignmentType>> <<DarlingtonTransistorMandatoryStandardTerminalMappingType>> <<DarlingtonTransistorSingleEmitterStandardTerminalMappingType>> <<DarlingtonTransistorDoubleEmitterStandardTerminalMappingType>> <<StandardTerminalName>> <<StandardTerminalNameAssignment>> <<SingleEmitterMapping>> <<DoubleEmitterMapping>> <<MandatoryMapping>> <<constraints>> } DarlingtonTransistorStandardTerminalNameAssignmentType "1..>" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..>" DarlingtonTransistorMandatoryStandardTerminalMappingType DarlingtonTransistorMandatoryStandardTerminalMappingType "2" StandardTerminalName DarlingtonTransistorMandatoryStandardTerminalMappingType "2" DarlingtonTransistorSingleEmitterStandardTerminalMappingType DarlingtonTransistorMandatoryStandardTerminalMappingType "2" DarlingtonTransistorDoubleEmitterStandardTerminalMappingType DarlingtonTransistorSingleEmitterStandardTerminalMappingType "2" StandardTerminalName DarlingtonTransistorDoubleEmitterStandardTerminalMappingType "2" StandardTerminalName StandardTerminalNameAssignment "1..>" SingleEmitterMapping SingleEmitterMapping "1..>" DarlingtonTransistorSingleEmitterStandardTerminalMappingType DoubleEmitterMapping "1..>" DarlingtonTransistorDoubleEmitterStandardTerminalMappingType DarlingtonTransistorSingleEmitterStandardTerminalMappingType "2" StandardTerminalName DarlingtonTransistorDoubleEmitterStandardTerminalMappingType "2" StandardTerminalName MandatoryMapping "1..>" constraints constraints "2" StandardTerminalName </pre>																								
type	<p>DarlingtonTransistorStandardTerminalNameAssignmentType, DarlingtonTransistorMandatoryStandardTerminalMappingType, DarlingtonTransistorMandatoryStandardTerminalNameType, DarlingtonTransistorSingleEmitterStandardTerminalMappingType, DarlingtonTransistorSingleEmitterStandardTerminalNameType, DarlingtonTransistorDoubleEmitterStandardTerminalMappingType, DarlingtonTransistorDoubleEmitterStandardTerminalNameType.</p>																								
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Base</td> <td>2. Collector</td> <td></td> <td></td> </tr> <tr> <td colspan="4">SingleEmitterMapping/StandardTerminalName</td> </tr> <tr> <td>1. Emitter</td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="4">DoubleEmitterMapping/StandardTerminalName</td> </tr> <tr> <td>1. Emitter 1</td> <td>2. Emitter 2</td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Base	2. Collector			SingleEmitterMapping/StandardTerminalName				1. Emitter				DoubleEmitterMapping/StandardTerminalName				1. Emitter 1	2. Emitter 2		
MandatoryMapping/StandardTerminalName																									
1. Base	2. Collector																								
SingleEmitterMapping/StandardTerminalName																									
1. Emitter																									
DoubleEmitterMapping/StandardTerminalName																									
1. Emitter 1	2. Emitter 2																								

4.7.3.20.2. Unijunction Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction
diagram	<pre> classDiagram class UnijunctionTransistorFunctionType class Ntype { <<N-type UnijunctionTransistorFunctionType>> } class Ptype { <<P-type UnijunctionTransistorFunctionType>> } class StandardTerminalNameAssignmentType { <<StandardTerminalNameAssignmentType>> } UnijunctionTransistorFunctionType "1..∞" --> Ntype : UnijunctionTransistorFunctionType "1..∞" --> Ptype : Ntype --> StandardTerminalNameAssignmentType : Ptype --> StandardTerminalNameAssignmentType : </pre>
type	UnijunctionTransistorFunctionType , N-typeUnijunctionTransistorFunctionType , P-typeUnijunctionTransistorFunctionType , UniJunctionTransistorStandardTerminalNameAssignmentType .

A *Unijunction* can have one of the following two types: *N-type* and *P-type*. These types share the same standard terminal name assignment structure.

4.7.3.20.2.1. N-type Unijunction Transistor

path	<ol style="list-style-type: none"> PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction/N-type PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction/P-type 				
diagram	<pre> classDiagram class NtypeUnijunctionTransistorFunctionType class StandardTerminalNameAssignmentType class Mapping { <<Mapping>> } class StandardTerminalName { <<StandardTerminalName>> } class TerminalMapID { <<TerminalMapID>> } NtypeUnijunctionTransistorFunctionType "1..∞" --> StandardTerminalNameAssignmentType : StandardTerminalNameAssignmentType --> Mapping : Mapping --> StandardTerminalName : Mapping --> TerminalMapID : </pre>				
type	UnijunctionTransistorStandardTerminalNameAssignmentType , UnijunctionTransistorStandardTerminalMappingType , UnijunctionTransistorStandardTerminalNameType .				
list of enumerate values	<table border="1"> <tr> <td>1. Base 1</td> <td>2. Base 2</td> <td>3. Emitter</td> <td></td> </tr> </table>	1. Base 1	2. Base 2	3. Emitter	
1. Base 1	2. Base 2	3. Emitter			

4.7.3.20.2.2. P-type Unijunction Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction/P-type				
diagram	<pre> classDiagram class P-type { <<type P-typeUnijunctionTransistorFunctionType>> } class StandardTerminalNameAssignment { <<type UniJunctionTransistorStandardTerminalNameAssignmentType>> } class Mapping { <<type UniJunctionTransistorStandardTerminalMappingType>> } class StandardTerminalName { <<type UniJunctionTransistorStandardTerminalNameType>> } P-type "1..*" --> "1..*" StandardTerminalNameAssignment StandardTerminalNameAssignment "1..*" --> "1..*" Mapping Mapping "1..*" --> "1..*" StandardTerminalName constraints "constraints" --> Mapping </pre>				
type	P-typeUnijunctionTransistorFunctionType , UniJunctionTransistorStandardTerminalNameAssignmentType , UniJunctionTransistorStandardTerminalMappingType , UniJunctionTransistorStandardTerminalNameType .				
list of enumerate values	Mapping/StandardTerminalName <table border="1"> <tr> <td>1. Base 1</td> <td>2. Base 2</td> <td>3. Emitter</td> <td></td> </tr> </table>	1. Base 1	2. Base 2	3. Emitter	
1. Base 1	2. Base 2	3. Emitter			

4.7.3.20.3. Field Effect Transistor (FET)

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect
diagram	<pre> classDiagram class FieldEffect { <<FieldEffectTransistorFunctionType>> } class JunctionGate { <<JunctionGateFunctionType>> } class IGFET_EnhancementType { <<IGFET-EnhancementFunctionType>> } class IGFET_DepletionType { <<IGFET-DepletionFunctionType>> } FieldEffect < -- JunctionGate JunctionGate < -- N_channel_JunctionGateFunctionType JunctionGate < -- P_channel_JunctionGateFunctionType IGFET_EnhancementType < -- JunctionGate IGFET_DepletionType < --> JunctionGate N_channel_JunctionGateFunctionType < -- N_channel_JunctionGateFunctionType N_channel_JunctionGateFunctionType --> StandardTerminalNameAssignmentType P_channel_JunctionGateFunctionType < -- P_channel_JunctionGateFunctionType P_channel_JunctionGateFunctionType --> StandardTerminalNameAssignmentType N_channel_JGFET_EnhancementFunctionType < -- N_channel_JGFET_EnhancementFunctionType N_channel_JGFET_EnhancementFunctionType --> StandardTerminalNameAssignmentType P_channel_JGFET_EnhancementFunctionType < -- P_channel_JGFET_EnhancementFunctionType P_channel_JGFET_EnhancementFunctionType --> StandardTerminalNameAssignmentType N_channel_JGFET_DepletionFunctionType < -- N_channel_JGFET_DepletionFunctionType N_channel_JGFET_DepletionFunctionType --> StandardTerminalNameAssignmentType P_channel_JGFET_DepletionFunctionType < -- P_channel_JGFET_DepletionFunctionType P_channel_JGFET_DepletionFunctionType --> StandardTerminalNameAssignmentType </pre>
type	FieldEffectTransistorFunctionType, JunctionGateFunctionType, N-channelJunctionGateFunctionType, P-channelJunctionGateFunctionType, IGFET-EnhancementFunctionType, N-channel-IGFET-EnhancementFunctionType, P-channel-IGFET-EnhancementFunctionType, IGFET-DepletionFunctionType, N-channel-IGFET-DepletionFunctionType, P-channel-IGFET-DepletionFunctionType, FET-StandardTerminalNameAssignmentType.

4.7.3.20.3.1. FET – Standard Terminal Name Assignment

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/JunctionGate/N-channel/StandardTerminalNameAssignment 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/JunctionGate/P-channel/StandardTerminalNameAssignment 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-EnhancementType/N-channel/StandardTerminalNameAssignment 4. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-EnhancementType/P-channel/StandardTerminalNameAssignment 5. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-DepletionType/N-channelIGFET-DepletionType/StandardTerminalNameAssignment 6. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-DepletionType/P-channelIGFET-DepletionType/StandardTerminalNameAssignment 																								
diagram	<pre> classDiagram class StandardTerminalNameAssignment { <<FET-StandardTerminalNameAssignment>> } class MandatoryMapping { <<FET-MandatoryStandardTerminalMapping>> } class SingleGateMapping { <<FET-SingleGateStandardTerminalMapping>> } class DoubleGateMapping { <<FET-DoubleGateStandardTerminalMapping>> } class StandardTerminalName { <<FET-StandardTerminalName>> } StandardTerminalNameAssignment "1..∞" --> MandatoryMapping StandardTerminalNameAssignment "1..∞" --> SingleGateMapping StandardTerminalNameAssignment "1..∞" --> DoubleGateMapping MandatoryMapping "*" --> StandardTerminalName SingleGateMapping "*" --> StandardTerminalName DoubleGateMapping "*" --> StandardTerminalName class constraints { ... } </pre>																								
type	FET–StandardTerminalNameAssignmentType, FET–MandatoryStandardTerminalMappingType, FET–MandatoryStandardTerminalNameType, FET–SingleGateStandardTerminalMappingType, FET–SingleGateStandardTerminalNameType, FET–DoubleGateStandardTerminalMappingType, FET–DoubleGateStandardTerminalNameType.																								
list of enumerate values	<table border="1"> <tr> <td colspan="4">MandatoryMapping/StandardTerminalName</td> </tr> <tr> <td>1. Drain</td> <td>2. Source</td> <td></td> <td></td> </tr> <tr> <td colspan="4">SingleEmitterMapping/StandardTerminalName</td> </tr> <tr> <td>3. Gate</td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="4">DoubleEmitterMapping/StandardTerminalName</td> </tr> <tr> <td>1. Gate 1</td> <td>2. Gate 2</td> <td></td> <td></td> </tr> </table>	MandatoryMapping/StandardTerminalName				1. Drain	2. Source			SingleEmitterMapping/StandardTerminalName				3. Gate				DoubleEmitterMapping/StandardTerminalName				1. Gate 1	2. Gate 2		
MandatoryMapping/StandardTerminalName																									
1. Drain	2. Source																								
SingleEmitterMapping/StandardTerminalName																									
3. Gate																									
DoubleEmitterMapping/StandardTerminalName																									
1. Gate 1	2. Gate 2																								

4.7.3.20.4. Insulated Gate Bipolar Transistor (IGBT)

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar
diagram	<pre> classDiagram class Insulated-GateBipolar { <<Insulated-GateBipolarTransistor>> } class IGBT-EnhancementType { <<IGBT-EnhancementFunctionType>> } class IGBT-DepletionType { <<IGBT-DepletionFunctionType>> } class N-channel { <<N-channel-IGBT-EnhancementFunctionType>> } class P-channel { <<P-channel-IGBT-EnhancementFunctionType>> } class StandardTerminalNameAssignmentA { <<IGBT-StandardTerminalNameAssignment>> } class N-channel-IGBT-DepletionFunctionType { <<N-channel-IGBT-DepletionFunctionType>> } class P-channel-IGBT-DepletionFunctionType { <<P-channel-IGBT-DepletionFunctionType>> } class StandardTerminalNameAssignmentB { <<IGBT-StandardTerminalNameAssignment>> } Insulated-GateBipolar < -- IGBT-EnhancementType Insulated-GateBipolar < -- IGBT-DepletionType IGBT-EnhancementType < -- N-channel IGBT-EnhancementType < -- P-channel IGBT-DepletionType < -- N-channel-IGBT-DepletionFunctionType IGBT-DepletionType < -- P-channel-IGBT-DepletionFunctionType N-channel --> StandardTerminalNameAssignmentA P-channel --> StandardTerminalNameAssignmentA N-channel-IGBT-DepletionFunctionType --> StandardTerminalNameAssignmentB P-channel-IGBT-DepletionFunctionType --> StandardTerminalNameAssignmentB </pre>
type	Insulated-GateBipolarFunctionType, IGBT-EnhancementFunctionType, N-channel-IGBT-EnhancementType, P-channel-IGBT-EnhancementType, IGBT-DepletionFunctionType, N-channel-IGBT-DepletionFunctionType, P-channel-IGBT-DepletionFunctionType, IGBT-StandardTerminalNameAssignmentType.

4.7.3.20.4.1. IGBT – Standard Terminal Name Assignment Type

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-EnhancementType/N-channel/IGBT-StandardTerminalNameAssignment 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-EnhancementType/P-channel/IGBT-StandardTerminalNameAssignment 3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-DepletionType/N-channel/IGBT-StandardTerminalNameAssignment 4. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-DepletionType/P-channel/IGBT-StandardTerminalNameAssignment 				
diagram	<pre> classDiagram class StandardTerminalNameAssignment { <<IGBT-StandardTerminalNameAssignment>> } class Mapping { <<IGBT-StandardTerminalMappingType>> } class StandardTerminalName { <<IGBT-StandardTerminalName>> } StandardTerminalNameAssignment "1..∞" -- "1..∞" Mapping : Mapping Mapping "3" -- "1..∞" StandardTerminalName : constraints </pre>				
type	IGBT-StandardTerminalNameAssignmentType, IGBT-StandardTerminalMappingType, IGBT-StandardTerminalNameType.				
	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Gate</td> <td style="padding: 2px;">2. Collector</td> <td style="padding: 2px;">3. Emitter</td> <td style="padding: 2px;"></td> </tr> </table>	1. Gate	2. Collector	3. Emitter	
1. Gate	2. Collector	3. Emitter			

4.7.3.20.5. Programmable Unijunction Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/ProgrammableUnijunction
diagram	<pre> classDiagram class ProgrammableUnijunction { <<ProgrammableUnijunctionTransistor>> } class N-type { <<N-typeProgrammableUnijunctionTransistor>> } class P-type { <<P-typeProgrammableUnijunctionTransistor>> } class StandardTerminalNameAssignment... { <<ProgrammableUnijunctionTransistorStandardTerminalNameAssignmentType>> } ProgrammableUnijunction "1..∞" -- "1..∞" N-type : ProgrammableUnijunction "1..∞" -- "1..∞" P-type : N-type "1..∞" -- "1..∞" StandardTerminalNameAssignment... P-type "1..∞" -- "1..∞" StandardTerminalNameAssignment... </pre>
type	ProgrammableUnijunctionTransistorFunctionType, N-typeProgrammableUnijunctionTransistorFunctionType, P-typeProgrammableUnijunctionTransistorFunctionType, ProgrammableUnijunctionTransistorStandardTerminalNameAssignmentType, ProgrammableUnijunctionTransistorStandardTerminalNameMappingType.

4.7.3.20.5.1. Terminal Mapping

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/ProgrammableUnijunction/N-type/ProgrammableUnijunctionTransistorStandardTerminalNameAssignment 2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/ProgrammableUnijunction/P-type/ProgrammableUnijunctionTransistorStandardTerminalNameAssignment 				
diagram	<pre> classDiagram class StandardTerminalNameAssignment { <<ProgrammableUnijunctionTransistorStandardTerminalNameAssignment>> } class Mapping { <<ProgrammableUnijunctionTransistorStandardTerminalMappingType>> } class StandardTerminalName { <<ProgrammableUnijunctionTransistorStandardTerminalNameType>> } StandardTerminalNameAssignment "1..∞" -- "1" Mapping StandardTerminalName "1..∞" -- "1" Mapping </pre>				
type	ProgrammableUnijunctionTransistorStandardTerminalMappingType, ProgrammableUnijunctionTransistorStandardTerminalNameType.				
	Mapping/StandardTerminalName <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">1. Gate</td> <td style="padding: 2px;">2. Anode</td> <td style="padding: 2px;">3. Cathode</td> <td style="padding: 2px;"></td> </tr> </table>	1. Gate	2. Anode	3. Cathode	
1. Gate	2. Anode	3. Cathode			

4.7.3.21. Other Standard Electrical Functions

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/OtherStandard
diagram	<pre> classDiagram class OtherStandard { <<FunctionMap-to-StandardNameType>> } class FunctionMap-to-StandardNameType { <<StandardTerminalNameAssignment>> class FunctionName { <<xstring>> } } class StandardTerminalNameAssignment { <<OtherStandardFunctionStandardTerminalNameAssignmentType>> } class Mapping { <<TerminalStandardType>> } class TerminalStandardType { <<StandardTerminalName>> class StandardTerminalName { <<xstring>> } } OtherStandard --> FunctionMap-to-StandardNameType FunctionMap-to-StandardNameType --> StandardTerminalNameAssignment StandardTerminalNameAssignment --> Mapping Mapping --> TerminalStandardType </pre>
type	FunctionMap-to-StandardNameType, OtherStandardFunctionStandardTerminalNameAssignmentType, TerminalStandardType.

This section can be used to capture the standard terminal name assignments for functions not classified here. If other functions are desired to be captured by this XML standard, please contact the JEDEC committee JC-16.

4.7.4. Electrical Specification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array
diagram	<p>The diagram illustrates the UML class structure for the ElectricalSpecification-Array. It features a main class ElectricalSpecification-ArrayType containing two associations: one to ElectricalSpecification (multiplicity 0..∞) and another to TruthTable (multiplicity 0..∞). The ElectricalSpecification association connects to a class ElectricalSpecificationType, which has attributes ID (xs:string) and type (ElectricalSpecificationType). This type is further refined by TestCondition (multiplicity 0..∞), ParameterSet (multiplicity 1..∞), and ParameterGraph (multiplicity 1..∞). The TruthTable association connects to a class TruthTableType. Below the main array class, there is a separate section for Footnote definitions, enclosed in a dashed box labeled JEP30-D10:Footnote-ArrayType. This section contains a Footnote class with attributes ID (xs:string) and type (FootnoteType).</p>
type	ElectricalSpecification-ArrayType , ElectricalSpecificationType , ElectricalSpecificationTestConditionType , ElectricalSpecificationParameterSetType , ElectricalSpecificationParameterGraphType , TruthTableType , JEP30-D10:Footnote-ArrayType , FootnoteType .

The [ElectricalSpecification-Array](#) is split into 2 sections, namely the [ElectricalSpecification](#) which typically captures the analog specifications, and the [TruthTable](#) which captures the logical definition of the device operation.

Several [TestConditions](#) can be defined for any given [ParameterSet](#), or [ParameterGraph](#). [TestConditions](#) set at this level must be the same for all the content contained in the [ParameterSet](#), or the [ParameterGraph](#). An example here is the ambient temperature of a device ($T_A = 25^\circ\text{C}$) in which a set of parameters are defined.

4.7.4.1. Test Condition

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/TestCondition
diagram 1 of 3	<p>The diagram shows the ElectricalSpecificationTestConditionType class with the following associations:</p> <ul style="list-style-type: none"> TestCondition: Multiplicity 0..∞ JEP30-D10:ParameterIdentityGroup: Multiplicity 0..∞ Values: Multiplicity 0..∞ Rule: Multiplicity 0..∞ AsciiUnits: Multiplicity 0..∞ (Note: This branch will go obsolete in first release in 2028) Units: Multiplicity 0..∞ (Note: Use these Preferred and Standard compliant Units) TestMethod: Multiplicity 0..∞
diagram 2 of 3	<p>The diagram shows the JEP30-D10:ParameterValuesGroupType class with the following associations:</p> <ul style="list-style-type: none"> Values: Multiplicity 0..∞ ValueSetGroup: Multiplicity 0..∞ FootnoteID: Multiplicity 0..∞ <p>ValueSetGroup has the following associations:</p> <ul style="list-style-type: none"> SymbolDescription: Multiplicity 0..∞ Nominal: Multiplicity xs:decimal ValueText: Multiplicity xs:string (Note: Use same element names as in ValueSetGroup when defining ValueText for Nominal, Negative Tolerance, Positive Tolerance, TotalTolerance, Minimum and/or Maximum into Schema for ValueText.) FootnoteID: Multiplicity 0..∞ <p>Nominal has the following associations:</p> <ul style="list-style-type: none"> NegativeTolerance: Multiplicity xs:decimal PositiveTolerance: Multiplicity xs:decimal TotalTolerance: Multiplicity xs:decimal ToleranceUOM: Multiplicity ToleranceUOMType Minimum: Multiplicity xs:decimal Maximum: Multiplicity xs:decimal

4.5.4.1. Test Condition (cont'd)

diagram 3 of 3	<pre> classDiagram class Rule { <<JEP30-D10:ParameterRuleType>> } class RuleName { <<xs:string>> } class RuleAnnotation { <<xs:string>> } class RuleContext { <<MinNomMaxRuleContextType>> } class LaTeX-Rule { <<xs:string>> } class MathML-Rule { <<m:math.type>> } class FootnoteID { <<xs:string>> 0..∞ } Rule "0..∞" --> LaTeX-and-MathML-RuleGroup LaTeX-and-MathML-RuleGroup < -- RuleName LaTeX-and-MathML-RuleGroup < -- RuleAnnotation LaTeX-and-MathML-RuleGroup < -- RuleContext LaTeX-and-MathML-RuleGroup < -- LaTeX-Rule LaTeX-and-MathML-RuleGroup < -- MathML-Rule LaTeX-and-MathML-RuleGroup < -- FootnoteID </pre>
type	ElectricalSpecificationTestConditionType, m:math.type, JEP30-D10:ParameterValuesGroupType, JEP30-D10:ParameterRuleType, MinNomMaxRuleContextType, AsciiUnitsForElectricalSpecificationType, UnitsForElectricalSpecificationType.
group	JEP30-D10:ParameterIdentityGroup, ValueSetGroup, LaTeX-and-MathML-RuleGroup

An example formula ($dl/dt \leq 70 \text{ A}/\mu\text{s}$) is shown below in its XML representation.

```

<TestCondition>
  <TeX-Symbol>dl/dt</TeX-Symbol>
  <SymbolDescription>rate of rise of the current</SymbolDescription>
  <TeX-Rule>\leq 70 \text{ A}/\mu\text{s}</TeX-Rule>
</TestCondition>
  
```

NOTE The *TeX-Symbol* is specified because the rule applies to a specific symbol.

Another formula example in which more than one Symbol is defined ($V_{DD} \leq V_{DS}$) is shown below in its XML representation. In this case the Symbols are encoded directly into the rule syntax.

```

<TestCondition>
  <TeX-Rule>V_{DD} \leq V_{DS}</TeX-Rule>
</TestCondition>
  
```

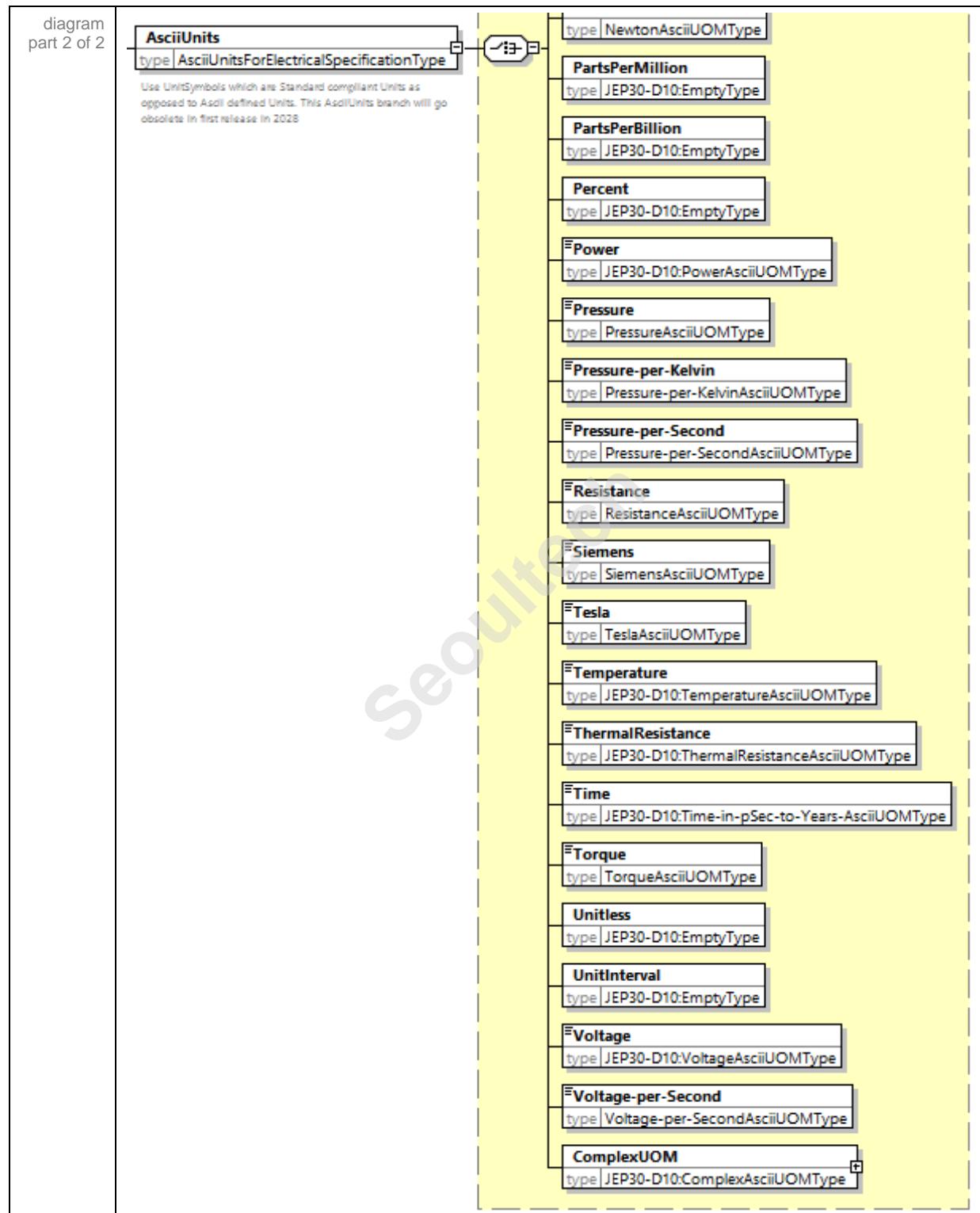
Alternatively, the rule may be expressed via MathML under the MathML-Rule element. This uses an existing MathML XML schema as defined by W3C.

The annotation under the *ValueText* is “Use same element names as in *ValueSetGroup* when defining *ValueText* for *Nominal*, *NegativeTolerance*, *PositiveTolerance*, *TotalTolerance*, *Minimum* and/or *Maximum* into Schema for *ValueText*.”

4.7.4.1.1. Ascii Units

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/TestCondition/AsciiUnits
diagram part 1 of 2	<pre> graph TD AsciiUnits[AsciiUnits] --> AsciiUnitsForElectricalSpecificationType[AsciiUnitsForElectricalSpecificationType] AsciiUnitsForElectricalSpecificationType --- AmpereHour[Ampere-Hour] AsciiUnitsForElectricalSpecificationType --- Bits[Bits] AsciiUnitsForElectricalSpecificationType --- BitsPerSecond[BitsPerSecond] AsciiUnitsForElectricalSpecificationType --- Candela[Candela] AsciiUnitsForElectricalSpecificationType --- Capacitance[Capacitance] AsciiUnitsForElectricalSpecificationType --- Coulomb[Coulomb] AsciiUnitsForElectricalSpecificationType --- Current[Current] AsciiUnitsForElectricalSpecificationType --- CurrentperSecond[Current-per-Second] AsciiUnitsForElectricalSpecificationType --- CurrentSquaredTime[CurrentSquaredTime] AsciiUnitsForElectricalSpecificationType --- Decibel[Decibel] AsciiUnitsForElectricalSpecificationType --- DecibelPerMeter[DecibelPerMeter] AsciiUnitsForElectricalSpecificationType --- Degrees[Degrees] AsciiUnitsForElectricalSpecificationType --- Dimension[Dimension] AsciiUnitsForElectricalSpecificationType --- Frequency[Frequency] AsciiUnitsForElectricalSpecificationType --- Gravity[Gravity] AsciiUnitsForElectricalSpecificationType --- Inductance[Inductance] AsciiUnitsForElectricalSpecificationType --- Joule[Joule] AsciiUnitsForElectricalSpecificationType --- LeastSignificantBit[Least-Significant-Bit] AsciiUnitsForElectricalSpecificationType --- Newton[Newton] AsciiUnitsForElectricalSpecificationType --- PartsPerMillion[PartsPerMillion] AsciiUnitsForElectricalSpecificationType --- PartsPerRillion[PartsPerRillion] AsciiUnits[AsciiUnits] --- AsciiUnitsForElectricalSpecificationType </pre> <p>Use UnitSymbols which are Standard compliant Units as opposed to Ascii defined Units. This AsciiUnits branch will go obsolete in first release in 2028.</p>

4.7.4.1.1. Ascii Units (cont'd)



4.7.4.1.1. Ascii Units (cont'd)

type	AsciiUnitsForElectricalSpecificationType , Ampere-HourAsciiUOMType , JEP30-D10:EmptyType , BitsPerSecondAsciiUOMType , CandelaAsciiUOMType , CapacitanceAsciiUOMType , CoulombAsciiUOMType , JEP30-D10:CurrentAsciiUOMType , Current-per-SecondAsciiUOMType , I2TAsciiUOMType , DecibelAsciiUOMType , Decibel-MeterAsciiUOMType , DimensionMetricAsciiUOMType , JEP30-D10:FrequencyAsciiUOMType , GravityAsciiUOMType , InductanceAsciiUOMType , JouleAsciiUOMType , NewtonAsciiUOMType , JEP30-D10:PowerAsciiUOMType , PressureAsciiUOMType , Pressure-per-KelvinAsciiUOMType , Pressure-per-SecondAsciiUOMType , ResistanceAsciiUOMType , SiemensAsciiUOMType , TelsaAsciiUOMType , JEP30-D10:TemperatureAsciiUOMType , JEP30-D10:ThermalResistanceAsciiUOMType , JEP30-D10:Time-in-uSec-to-Sec-AsciiUOMType , TorqueAsciiUOMType , JEP30-D10:VoltageAsciiUOMType , JEP30-D10:ComplexAsciiUOMType .
NOTE:	Use Units which are Standard compliant Units as opposed to Ascii defined Units. This AsciiUnits branch will go obsolete in first release in 2028.

4.7.4.1.1. Ascii Units (cont'd)

The enumerated list of values for each of the UOM's specified above are identified in Table 3.

Table 2 — UOM Enumerated Lists

Unit	Enumerated Values							
Ampere-Hour	uAh	mAh	Ah	kAh				
BitsPerSecond	bps	kbps	Mbps	Gbps				
Candela	cd							
Capacitance	aF	fF	pF	nF	uF	mF	F	
Coulomb	nC	uC	mC	C	kC			
Current	nA	uA	mA	A				
Current-per-Second	mA/s	A/us	A/ms	A/s				
Current Squared Time	A^2s							
Decibel	dB	dB(A)						
DecibelPerMeter	dB/m							
Dimension	nm	um	mm	m				
Frequency	Hz	kHz	MHz	GHz				
Gravity	gn							
Inductance	uH	mH	H					
Joule	uJ	mJ	J	kJ				
Newton	uN	mN	N	KN				
Power	mW	W	kW	dBm				
Pressure	mPa	Pa	kPa	MPa	atm	mbar	bar	lb/in^2
Pressure-per-Kelvin	Pa/K							

4.7.4.1.1. Ascii Units (cont'd)**Table 2 - UOM Enumerated Lists (cont'd)**

Unit	Enumerated Values									
Pressure-per-Second	Pa/s									
Resistance	mOhm	Ohm	KOhm	MOhm	GOhm					
Siemens	uS	mS	S	kS	MS					
Tesla	nT	uT	mT	T	kT					
Temperature	DegC	DegF	K							
ThermalResistance	K/W	DegC/W								
Time	ps	ns	us	ms	s	min	h	d	wk	y
Torque	Nm									
Voltage	uV	mV	V	kV	dBV	dBV(A)				
Voltage-per-Second	mV/us	mV/min	V/us	V/ms	V/s					
ComplexUOM										
Prefix	Femto	Pico	Nano	Micro	Milli	Unity	Kilo	Mega	Giga	Tera
	Ampere	Bits	Candela	Celsius	Decibel	Fahrenheit		Farad	Foot	Gram
	Henry	Hertz	Hour	Inch	Kelvin	Least-Significant-Bit		Lumen	Meter	Mil
	Min	Newton	Ohm	Ounce	Pascal	Percent	PartsPerMillion		PartsPerBillion	
	Pound	Second	Tesla	Volt	Watt					

4.7.4.1.1.1. Complex UOM

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/TestCondition/AsciiUnits/ComplexUOM</code>
diagram	<pre> classDiagram class ComplexUOM { type JEP30-D10:ComplexAsciiUOMType } class Factor { type AsciiFactorType } class Prefix { type AsciiPrefixType } class UOM { type AsciiUOMType } class Exponent { type xs:integer } class Root { type xs:integer } ComplexUOM "1..∞" --> Factor Factor "1..∞" --> Prefix Factor "1..∞" --> UOM Factor "1..∞" --> Exponent Factor "1..∞" --> Root </pre>
type	<code>ComplexAsciiUOMType, AsciiFactorType, AsciiPrefixType, AsciiUOMType.</code>

Other UOM can be defined under the `ComplexUOM` branch. An example formula ($dl/dt = 100$ A/ μ s) is shown below in its XML representation.

```

<TestCondition>
  <LaTeX-Symbol>dl/dt</LaTeX-Symbol>
  <SymbolDescription>rate of rise of the current</SymbolDescription>
  <Value>100</Value>
  <AsciiUnits>
    <ComplexUOM>
      <Factor>
        <Prefix>Unity</Prefix>
        <UOM>Ampere</UOM>
      </Factor>
      <Factor>
        <Prefix>Micro</Prefix>
        <UOM>Second</UOM>
        <Exponent>-1</Exponent>
      </Factor>
    </ComplexUOM>
  </AsciiUnits>
</TestCondition>
  
```

4.7.4.1.1. Complex UOM (cont'd)

An alternative Unit such as Decibel/Meter can be created such as.

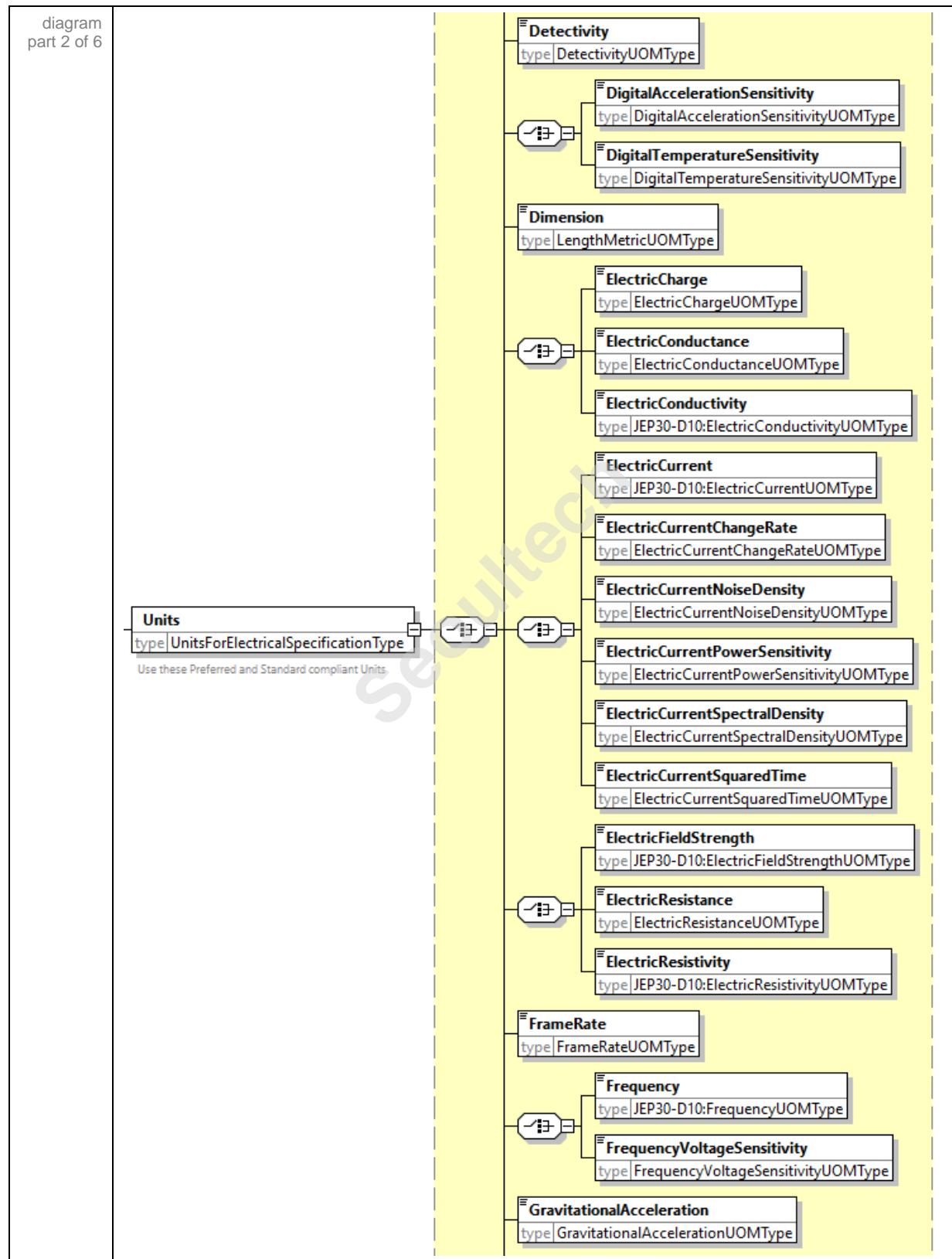
```
<AsciiUnits>
  <ComplexUOM>
    <Factor>
      <UOM>Decibel</UOM>
    </Factor>
    <Factor>
      <UOM>Meter</UOM>
      <Exponent>-1</Exponent>
    </Factor>
  </ComplexUOM>
</AsciiUnits>
```

The absence of *Prefix* defaults to *Unity*.

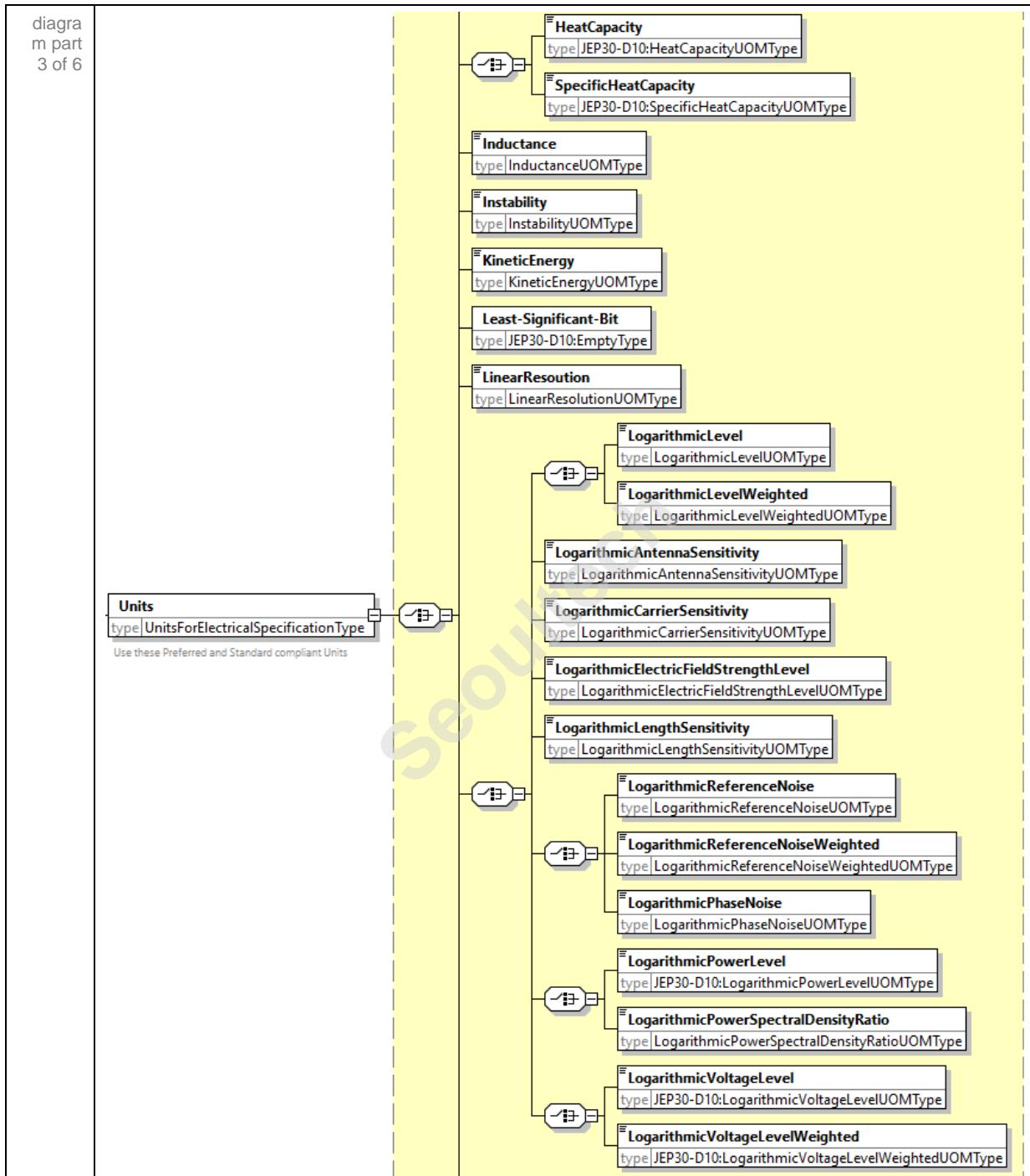
4.7.4.1.2. Units

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/TestCondition/Units
diagram part 1 of 6	<p>The diagram illustrates the <code>UnitsForElectricalSpecificationType</code> class and its associations. The <code>Units</code> class (type: <code>UnitsForElectricalSpecificationType</code>) is associated with the following units:</p> <ul style="list-style-type: none"> <code>StandardsAuthorityBodyID</code> (type: <code>xs:string</code>) <code>Unit-ID</code> (type: <code>xs:string</code>) <code>Area</code> (type: <code>JEP30-D10:AreaUOMType</code>) <code>BitRate</code> (type: <code>BitsRateUOMType</code>) <code>DatasetOfBits</code> (type: <code>DatasetOfBitsUOMType</code>) <code>DatasetOfBytes</code> (type: <code>DatasetOfBytesUOMType</code>) <code>Capacitance</code> (type: <code>CapacitanceUOMType</code>) <code>CapacitancePressureSensitivity</code> (type: <code>CapacitancePressureSensitivityUOMType</code>) <code>Capacity</code> (type: <code>CapacityUOMType</code>) <code>Cycle</code> (type: <code>CycleUOMType</code>) <code>Degrees</code> (type: <code>JEP30-D10:EmptyType</code>) <p>A note at the bottom left of the diagram states: "Use these Preferred and Standard compliant Units".</p>

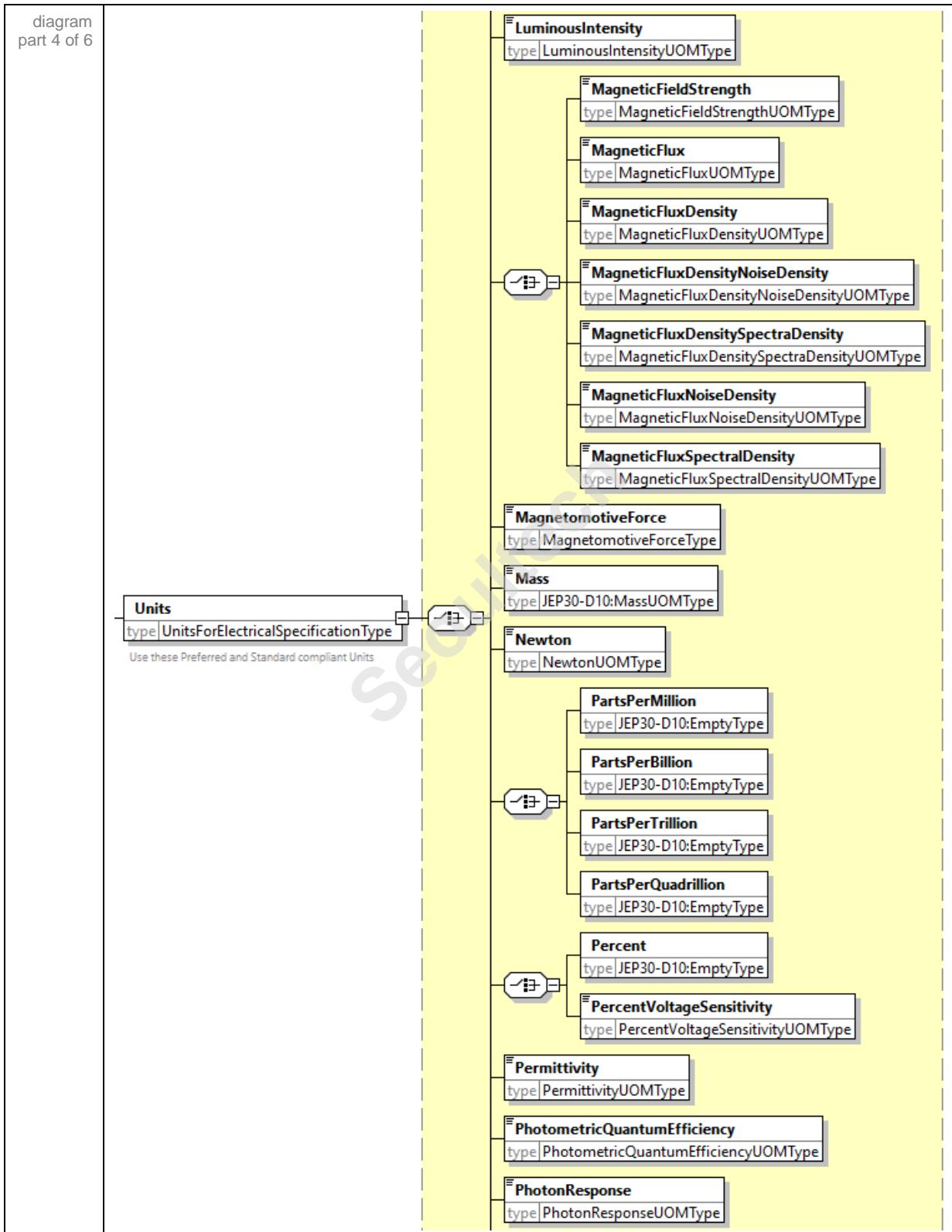
4.7.4.1.2. Units (cont'd)



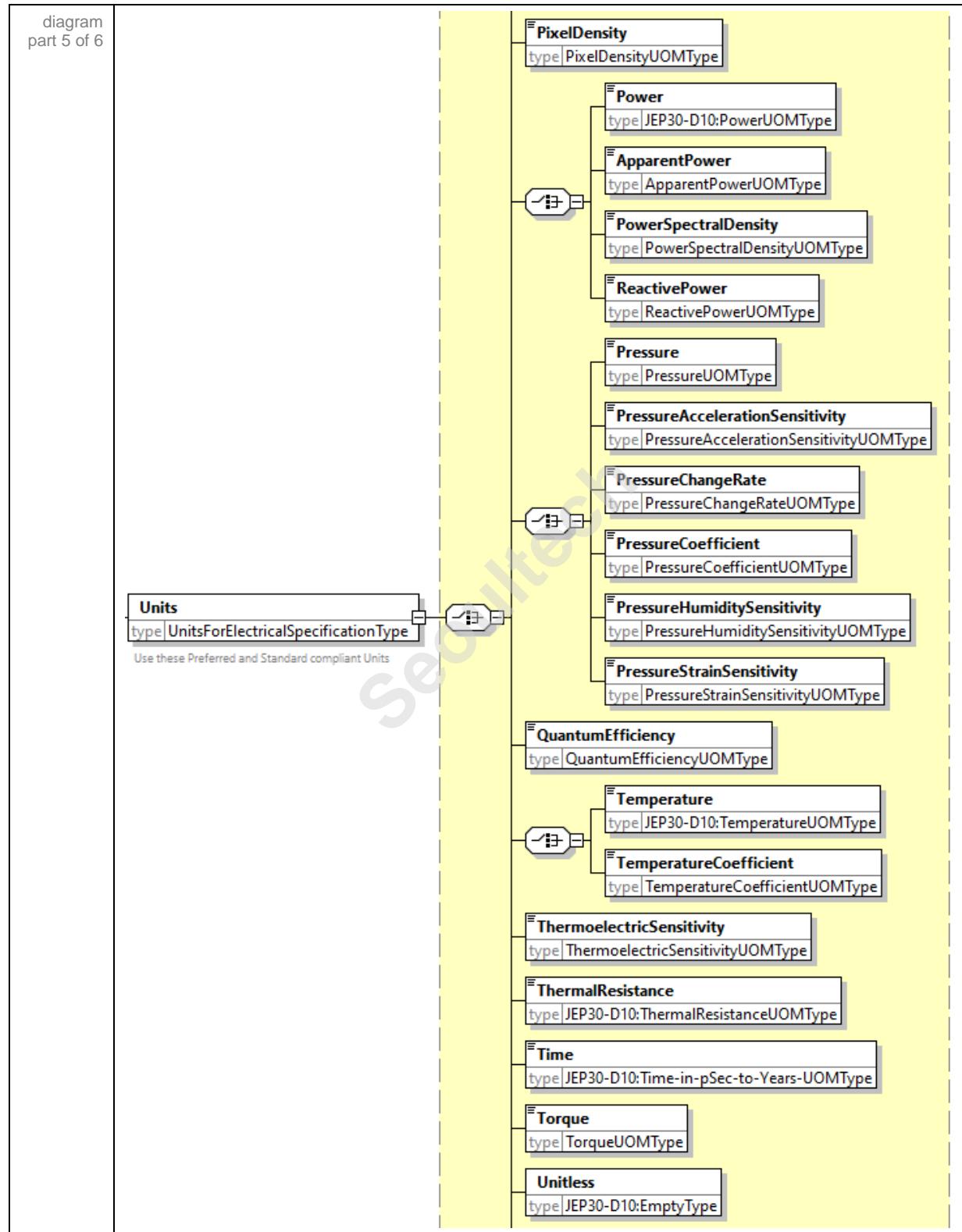
4.7.4.1.2. Units (cont'd)



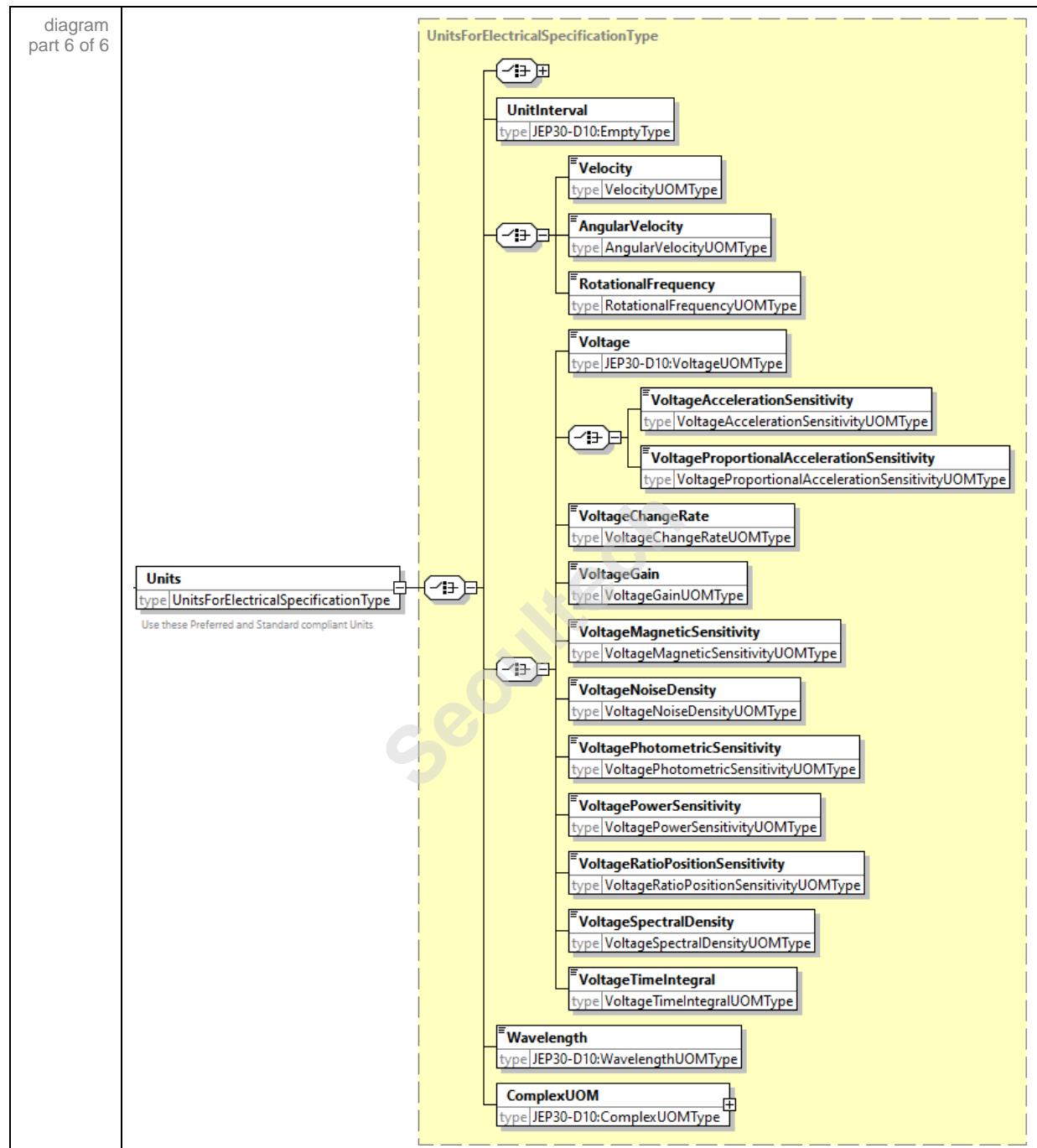
4.7.4.1.2. Units (cont'd)



4.7.4.1.2. Units (cont'd)



4.7.4.1.2. Units (cont'd)



4.7.4.1.2. Units (cont'd)

type	<p>UnitsForElectricalSpecificationType, JEP30-D10:AreaUOMType, BitsRateUOMType, DatesetOfBitsUOMType, DatesetOfBytesUOMType, CapacitanceUOMType, CapacitancePressureSensitivityUOMType, CapacityUOMType, CycleUOMType, JEP30-D10:EmptyType, DetectivityUOMType, DigitalAccelerationSensitivityUOMType, DigitalTemperatureSensitivityUOMType, LengthMetricUOMType, ElectricChargeUOMType, ElectricConductanceUOMType, JEP30-D10:ElectricConductivityUOMType, JEP30-D10:ElectricCurrentUOMType, ElectricCurrentChangeRateUOMType, ElectricCurrentNoiseDensityUOMType, ElectricCurrentPowerSensitivityUOMType, ElectricCurrentSpectralDensityUOMType, ElectricCurrentSquaredTimeUOMType, ElectricFieldStrengthUOMType, ElectricResistanceUOMType, JEP30-D10:ElectricResistivityUOMType, FrameRateUOMType, JEP30-D10:FrequencyUOMType, FrequencyVoltSensitivityUOMType, GravitationalAccelerationUOMType, JEP30-D10:HeatCapacityUOMType, JEP30-D10:SpecificHeatCapacityUOMType, InductanceUOMType, InstabilityUOMType, KineticEnergyUOMType, LinearResolutionUOMType, LogarithmicLevelUOMType, LogarithmicLevelWeightedUOMType, LogarithmicAntennaSensitivityUOMType, LogarithmicCarrierSensitivityUOMType, LogarithmicElectricFieldStrengthLevelUOMType, LogarithmicLengthSensitivityUOMType, LogarithmicReferenceNoiseUOMType, LogarithmicReferenceNoiseWeightedUOMType, LogarithmicPhaseNoiseUOMType, JEP30-D10:LogarithmicPowerLevelUOMType, LogarithmicPowerSpectralDensityRatioUOMType, JEP30-D10:LogarithmicVoltageLevelUOMType, JEP30-D10:LogarithmicVoltageLevelWeightedUOMType, LuminousIntensityUOMType, MagnetomotiveForceType, MagneticFieldNoiseDensityUOMType, MagneticFieldSpectralDensityUOMType, MagneticFieldStrengthUOMType, MagneticFluxUOMType, MagneticFieldDensityUOMType, MagneticFluxNoiseDensityUOMType, MagneticFieldSpectralDensityUOMType, JEP30-D10:MassUOMType, NewtonUOMType, PercentVoltSensitivityUOMType, PermittivityUOMType, PhotometricQuantumEfficiencyUOMType, PhotonResponseUOMType, PixelDensityUOMType, JEP30-D10:PowerUOMType, ApparentPowerUOMType, PowerSpectralDensityUOMType, ReactivePowerUOMType, PressureUOMType, PressureAccelerationSensitivityUOMType, PressureChangeRateUOMType, PressureCoefficientUOMType, PressureHumiditySensitivityUOMType, PressureStrainSensitivityUOMType, QuantumEfficiencyUOMType, JEP30-D10:TemperatureUOMType, TemperatureCoefficientUOMType, JEP30-D10:ThermalResistanceUOMType, JEP30-D10:Time-in-pSec-to-Years-UOMType, TorqueUOMType, VelocityUOMType, AngularVelocityUOMType, RotationalFrequencyUOMType, JEP30-D10:VoltageUOMType, VoltageAccelerationSensitivityUOMType, VoltageProportionalAccelerationSensitivityUOMType, VoltageChangeRateUOMType, VoltageGainUOMType, VoltageMagneticSensitivityUOMType, VoltageNoiseDensityUOMType, VoltagePhotometricSensitivityUOMType, VoltagePowerSensitivityUOMType, VoltageRatioPositionSensitivityUOMType, VoltageSpectralDensityUOMType, VoltageTemperatureSensitivityUOMType, VoltageTimeIntegralUOMType, WavelengthUOMType, JEP30-D10:ComplexUOMType.</p>
NOTE:	Use these Preferred and Standard compliant Units

4.7.4.1.2. Units (cont'd)

The enumerated list of values for each of the UOM's specified above are identified in Table 3.

Table 3 — UOM Enumerated Lists

Unit	Enumerated Values								
Area	nm ²	μm ²	mm ²	cm ²	m ²				
BitRate	bit/s	kbit/s	Mbit/s	Gbit/s	Tbit/s				
DatasetOfBits	bit	kbit	Mbit	Gbit	Tbit				
DatasetOfBytes	byte	kbyte	Mbyte	Gbyte	Tbyte				
Capacitance	aF	fF	pF	nF	μF	mF	F		
Capacitance Pressure Sensitivity	aF/Pa	aF/kPa	aF/MPa	fF/Pa	fFk/Pa	fF/MPa	pF/Pa	pF/kPa	pF/MPa
	nF/kPa	nF/MPa	aF/bar	aF/mbar	fF/bar	fF/mbar	pF/bar	pF/mbar	nF/bar
	aF/psi	fF/psi	pF/psi	nF/psi	aF/Torr	fF/Torr	pF/Torr	nF/Torr	
Capacity	μA·h	mA·h	A·h	kA·h					
Cycles	cycles	kcycles	Mcycles	Gcycles	Tcycles				
Detectivity	mm·√Hz/W		cm·√Hz/W		m·√Hz/W				
Digital Acceleration Sensitivity	LSB/gn								
Digital Temperature Sensitivity	LSB/mK	LSB/K	LSB/°C						
Dimension	nm	μm	mm	m					
Electric Charge	pC	nC	μC	mC	C	kC			
Electric Conductance	pS	nS	μS	mS	S	kS	MS		
Electric Conductivity	S/m								
Electric Current	pA	nA	μA	mA	A	kA			

4.7.4.1.2. Units (cont'd)**Table 3 - UOM Enumerated Lists (cont'd)**

Unit	Enumerated Values									
Electric Current Change Rate	µA/s	mA/µs	mA/ms	mA/s	A/µs	A/ms	A/s			
Electric Current Noise Density	pA/√Hz	nA/√Hz	µA/√Hz	mA/√Hz	A/√Hz					
Electric Current Power Sensitivity	mA/mW	A/mW	µA/W	mA/W	A/W					
Electric Current Spectral Density	pA²/Hz	nA²/Hz	µA²/Hz	mA²/Hz	A²/Hz					
Electric Current Squared Time	A²·ms	mA²·s	A²·s	kA²·s						
Electric Field Strength	µV/m	mV/m	V/m	kV/m	MV/m	V/cm	kV/cm	MV/cm		
Electric Resistance	nΩ	µΩ	mΩ	Ω	KΩ	MΩ	GΩ			
Electric Resistivity	Ω·m									
Force	pN	nN	µN	mN	N	kN				
Frame Rate	Hz									
Frequency	mHz	Hz	kHz	MHz	GHz	THz				
Frequency Voltage Sensitivity	Hz/mV	Hz/V	kHz/mV	kHz/V	MHz/mV	MHz/V	GHz/mV	GHz/V	THz/mV	
Gravitational Acceleration	mgn	gn	kgn							
Heat Capacity	mJ/K	J/K	mJ/°C	J/°C						
Specific Heat Capacity	J/(g·K)	J/(kg·K)	J/(g·°C)	J/(kg·°C)	callIT/(g·K)		kcallIT/(g·K)		callIT/(g·°C)	
	kcallIT/(g·°C)		calhth/(g·K)		kcalhth/(g·K)		calhth/(g·°C)		kcalhth/(g·°C)	
Inductance	pH	nH	µH	mH	H	kH				

4.7.4.1.2. Units (cont'd)

Table 3 - UOM Enumerated Lists (cont'd)

Unit	Enumerated Values									
Instability	ppm/s	ppm/min	ppm/h	ppm/d	ppm/y	ppb/s	ppb/min	ppb/h	ppb/d	ppb/y
	ppt/s	ppt/min	ppt/h	ppt/d	ppt/y	ppq/s	ppq/min	ppq/h	ppq/d	ppq/y
Kinetic Energy	fJ	pJ	nJ	μJ	mJ	J	kJ			
Linear Resolution	ppi	pp·cm	dpi							
Logarithmic Level	dB									
Logarithmic Level Weighted	dB(A)	dB(C)	dB(Z)							
Logarithmic Antenna Sensitivity	dBi	dBd								
Logarithmic Carrier Sensitivity	dBc									
Logarithmic Electric Field Strength Level	dB μ V/m	dBmV/m	dBV/m							
Logarithmic Length Sensitivity	dB/ μ m	dB/mm	dB/cm	dB/m						
Logarithmic Reference Noise	dBrn									
Logarithmic Reference Noise Weighted	dBrn(C)	dBrn(O)								
Logarithmic Phase Noise	dBc/Hz	dBc/kHz	dBc/MHz							
Logarithmic Power Level	dB μ W	dBm	dBW							
Logarithmic Power Spectral Density Ratio	dBm/Hz	dBm/kHz		dBm/MHz		dBW/Hz	dBW/kHz		dBW/MHz	

4.7.4.1.2. Units (cont'd)**Table 3 - UOM Enumerated Lists (cont'd)**

Unit	Enumerated Values									
Logarithmic Voltage Level	dB μ V	dBmV	dBV							
Logarithmic Voltage Level Weighted	dB μ V(A)	dBmV(A)		dBV(A)						
Luminous Intensity	μ cd	mcd	cd							
Magnetic Flux Density Noise Density	pT/ $\sqrt{\text{Hz}}$	nT/ $\sqrt{\text{Hz}}$	$\mu\text{T}/\sqrt{\text{Hz}}$	mT/ $\sqrt{\text{Hz}}$	T/ $\sqrt{\text{Hz}}$					
Magnetic Flux Density Spectral Density	pT ² /Hz	nT ² /Hz	$\mu\text{T}^2/\text{Hz}$	mT ² /Hz	T ² /Hz					
Magnetic Field Strength	pA/m	nA/m	$\mu\text{A}/\text{m}$	mA/m	A/m					
Magnetic Flux	pWb	nWb	μWb	mWb	Wb	kWb				
Magnetic Flux Density	pT	nT	μT	mT	T	kT				
Magnetic Flux Noise Density	pWb/ $\sqrt{\text{Hz}}$		nWb/ $\sqrt{\text{Hz}}$		$\mu\text{Wb}/\sqrt{\text{Hz}}$		mWb/ $\sqrt{\text{Hz}}$		Wb/ $\sqrt{\text{Hz}}$	
Magnetic Flux Spectral Density	pWb ² /Hz	nWb ² /Hz	$\mu\text{Wb}^2/\text{Hz}$	mWb ² /Hz		Wb ² /Hz				
Magnetomotive Force	A·t									
Mass	μg	mg	g	kg	oz	lb				
Percent Voltage Sensitivity	%/ μV	%/mV	%/V	%/kV						
Permittivity	pF/m	nF/m	$\mu\text{F}/\text{m}$	mF/m	F/m					
Photometric Quantum Efficiency	e ⁻ /(lx·s)									

4.7.4.1.2. Units (cont'd)

Table 3 - UOM Enumerated Lists (cont'd)

Unit	Enumerated Values								
Photon Response	$\text{γ}/(\text{lx}\cdot\text{s})$								
Pixel Density	pixel/mm ²		kpixel/mm ²		Mpixel/mm ²		Gpixel/mm ²		pixel/cm ²
	kpixel/cm ²		Mpixel/cm ²		Gpixel/cm ²				
Power	pW	nW	μW	mW	W	kW	MW		
Apparent Power	nV·A	μV·A	mV·A	V·A	kV·A	MV·A			
Power Spectral Density	pW/Hz	nW/Hz	μW/Hz	mW/Hz	W/Hz				
Reactive Power	pVAR	nVAR	μVAR	mVAR	VAR	kVAR	MVAR		
Pressure	μPa	mPa	Pa	kPa	MPa	mbar	bar	atm	psi
	mTorr	Torr							μTorr
Pressure Acceleration Sensitivity	mPa/gn	Pa/gn	hPa/gn	kPa/gn	mbar/gn	bar/gn	psi/gn	Torr/gn	
Pressure Change Rate	mPa/s	Pa/s	kPa/s						
Pressure Coefficient	mPa/K	Pa/K	kPa/K						
Pressure Humidity Sensitivity	mPa/%RH		Pa/%RH		hPa/%RH		kPa/%RH		mbar/%RH
	bar/%RH		psi/%RH		Torr/%RH				
Pressure Strain Sensitivity	Pa/με								
Quantum Efficiency	(lx·s) ⁻¹								
Temperature	°C	°F	K						
Temperature Coefficient	ppb/K	ppm/K	%/K	ppb/°C	ppm/°C	%/°C			
Thermal Resistance	K/mW	K/W	K/kW	°C/mW	°C/W	°C/kW			

4.7.4.1.2. Units (cont'd)**Table 3 - UOM Enumerated Lists (cont'd)**

Unit	Enumerated Values									
Thermoelectric Sensitivity	$\mu\text{V}/^\circ\text{C}$	$\text{V}/^\circ\text{C}$	$\text{MV}/^\circ\text{C}$	$\mu\text{V}/\text{K}$	V/K	MV/K				
Time	ps	ns	μs	ms	s	min	h	d	wk	mo
	y									
Torque	$\mu\text{N}\cdot\text{m}$	$\text{mN}\cdot\text{m}$	$\text{mN}\cdot\text{m}$							
Velocity	m/s	ft/s	km/h	mph						
Angular Velocity	$^\circ/\text{ms}$	$^\circ/\text{s}$	$^\circ/\text{min}$	rad/ms	rad/s					
Rotational Frequency	rev/s	rev/min								
Voltage	μV	mV	V	kV						
Voltage Acceleration Sensitivity	nV/gn	$\mu\text{V}/\text{gn}$	mV/gn	V/gn						
Voltage Proportional Acceleration Sensitivity	$\text{mV}/(\% \cdot \text{gn})$		$\text{V}/(\% \cdot \text{gn})$							
Voltage Change Rate	$\mu\text{V}/\text{min}$	mV/min	V/min	$\mu\text{V}/\text{s}$	mV/s	V/s	$\mu\text{V}/\text{ms}$	mV/ms	V/ms	$\mu\text{V}/\mu\text{s}$
	$\text{mV}/\mu\text{s}$	V/ μs	$\mu\text{V}/\text{ns}$	mV/ns	V/ns					
Voltage Gain	$\mu\text{V}/\text{V}$	$\mu\text{V}/\text{mV}$	$\mu\text{V}/\mu\text{V}$	mV/V	mV/mV	$\text{mV}/\mu\text{V}$	V/V	V/mV	V/ μV	
Voltage Magnetic Sensitivity	pV/T	pV/mT	pV/ μT	pV/nT	nV/T	nV/mT	nV/ μT	nV/nT	$\mu\text{V}/\text{T}$	$\mu\text{V}/\text{mT}$
	$\mu\text{V}/\mu\text{T}$	uV/nT	mV/T	mV/mT	$\text{mV}/\mu\text{T}$	mV/nT	V/T	V/mT	V/ μT	V/nT
Voltage Noise Density	pV/ $\sqrt{\text{Hz}}$	nV/ $\sqrt{\text{Hz}}$	$\mu\text{V}/\sqrt{\text{Hz}}$	$\text{mV}/\sqrt{\text{Hz}}$	V/ $\sqrt{\text{Hz}}$					
Voltage Photometric Sensitivity	$\mu\text{V}/(\text{l}x\cdot\text{s})$		$\text{mV}/(\text{l}x\cdot\text{s})$		V/(lx·s)					
Voltage Power Sensitivity	$\mu\text{V}/\mu\text{W}$	$\text{mV}/\mu\text{W}$	mV/mW	V/mW	V/W					

4.7.4.1.2. Units (cont'd)

Table 3 - UOM Enumerated Lists (cont'd)

Unit	Enumerated Values									
Voltage Ratio Position Sensitivity	(mV/V)/mm		(V/V)/mm		(mV/V)/cm		(V/V)/cm	(mV/V)/in		(V/V)/in
Voltage Spectral Density	pV ² /Hz	nV ² /Hz	μV ² /Hz	mV ² /Hz	V ² /Hz					
Voltage Time Integral	μV·ps	μV·ns	μV·μs	μV·ms	μV·s	mV·ps	mV·ns	mV·μs	mV·ms	mV·s
	V·ps	V·ns	V·μs	V·ms	V·s	kV·ps	kV·ns	kV·μs	kV·ms	kV·s
Wavelength	pm	nm	μm	mm	m					
Complex UOM										
- Prefix	a	f	p	n	μ	c	m	k	M	G
	T									
- UOM	%	°C	°F	A	bar	bit	byte	C	cd	cycles
	dB	dpi	e ⁻	F	fc	ft	g	gn	H	h
	Hz	in	J	K	lm	LSB	lx	m	min	N
	Pa	ppm	ppb	ppt	ppq	ppcm	ppi	px	rad	rev
	S	s	T	t	Torr	V	VAR	W	Wb	Y
	ε	Ω								
- Reference	c	d	i	rn						
- Weighted	(A)	(C)	(O)	(Z)						

4.7.4.1.2.1. Complex UOM

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/TestCondition/Units/ComplexUOM
diagram	
type	ComplexUOMType, FactorType, EmptyType, PrefixSymbolType, UOMSymbolType, UOMReferenceType, UOMWeightedType, m:math.type

Other UOM can be defined under the *ComplexUOM* branch. A simple example of a complexUOM such as (mV/V)/cm is shown below in its XML representation.

```
<ComplexUOM>
  <Factor>
    <UOM>dB</UOM>
  </Factor>
  <Factor>
    <UOM>m</UOM>
    <Exponent>-1</Exponent>
  </Factor>
</ComplexUOM>
```

4.7.4.1.2.1. Complex UOM (cont'd)

A more complex unit as in Voltage Ratio Position Sensitivity (mV/V)/cm can be expressed as

```
<ComplexUOM>
  <Factor>
    <OpenParenteses/>
    <Prefix>m</Prefix>
    <UOM>V</UOM>
  </Factor>
  <Factor>
    <UOM>V</UOM>
    <Exponent>-1</Exponent>
    <CloseParenteses/>
  </Factor>
  <Factor>
    <Prefix>c</Prefix>
    <UOM>m</UOM>
    <Exponent>-1</Exponent>
  </Factor>
</ComplexUOM>
```

Another complex UOM such as Voltage Photometric Sensitivity $\mu\text{V}/(\text{l}\cdot\text{s})$ can be expressed as

```
<ComplexUOM>
  <Factor>
    <Prefix>\u</Prefix>
    <UOM>V</UOM>
  </Factor>
  <Factor>
    <OpenParenteses/>
    <UOM>l\></UOM>
  </Factor>
  <Factor>
    <UOM>s</UOM>
    <CloseParenteses/>
    <Exponent>-1</Exponent>
  </Factor>
</ComplexUOM>
```

The absence of **Prefix** defaults to unity.

4.7.4.2. Parameter Set

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterSet
diagram	<pre> classDiagram class ParameterSet { <<ElectricalSpecificationParameterSetType>> } class ID { <<xs:string>> } class Title { <<xs:string>> } class TestCondition { <<ElectricalSpecificationTestConditionType>> } class SubTitle { <<xs:string>> } class Parameter { <<ElectricalSpecificationParameterType>> } class Footnote_Array { <<JEP30-D10:Footnote-ArrayType>> } ParameterSet "1..>" -- "0..>" ID ParameterSet "1..>" -- "1..>" Title ParameterSet "1..>" -- "0..>" TestCondition ParameterSet "1..>" -- "1..>" SubTitle ParameterSet "1..>" -- "1..>" Parameter ParameterSet "1..>" -- "1..>" Footnote_Array </pre> <p>The diagram illustrates the UML Class Diagram for <code>ElectricalSpecificationParameterSetType</code>. It consists of the following classes and their associations:</p> <ul style="list-style-type: none"> ParameterSet: The primary class, marked with <code><<ElectricalSpecificationParameterSetType>></code>. It has multiplicity <code>1..></code> at both ends of its associations. ID: Associated with <code>ParameterSet</code> with multiplicity <code>1..></code>. Title: Associated with <code>ParameterSet</code> with multiplicity <code>1..></code>. TestCondition: Associated with <code>ParameterSet</code> with multiplicity <code>0..></code>. SubTitle: Associated with <code>ParameterSet</code> with multiplicity <code>1..></code>. Parameter: Associated with <code>ParameterSet</code> with multiplicity <code>1..></code>. Footnote-Array: Associated with <code>ParameterSet</code> with multiplicity <code>1..></code>. <p>A constraint box labeled <code>+ constraints</code> is located at the bottom left of the diagram area.</p>
type	<code>ElectricalSpecificationParameterSetType</code> , <code>ElectricalSpecificationTestConditionType</code> , <code>ElectricalSpecificationParameterType</code> , <code>JEP30-D10:Footnote-ArrayType</code> .

The `TestCondition` defined under a `ParameterSet` is specific to the condition under which the Parameter is true. For example, when the `TestCondition` is set to $I_F = 1.5$ A, then the Parameter V_F has a typical (nominal) value of 0.41 V. When the `TestCondition` is set to $I_F = 3.0$ A, then the Parameter V_F has a nominal value of 0.46 V and a maximum value of 0.54 V. Both of these `TestConditions` occur at an ambient temperature ($T_A = 25$ °C), therefore the ambient temperature test condition is set at the `ElectricalSpecification/TestCondition` branch.

4.7.4.2.1. Parameter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterSet/Parameter
diagram	
type	ElectricalSpecificationParameterType, ElectricalSpecificationTestConditionType, ParameterValuesType, ParameterRuleType, AsciiUnitsForElectricalSpecificationType, UnitsForElectricalSpecificationType.
group	JEP30-D10:ParameterIdentityGroup

NOTE: The element Units has the following annotation “Use IEC compliant Units as opposed to Ascii defined Units. This Units branch will go obsolete in first release in 2028”.

4.7.4.2.1.1. Values

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterSet/Parameter/Values
diagram	<pre> classDiagram class ParameterValuesType { LaTeX-Symbol MathML-Symbol SymbolDescription Nominal NegativeTolerance PositiveTolerance TotalTolerance ToleranceUOM Minimum Maximum ValueText FootnoteID } class JEP30-D10:ValueSetGroup ParameterValuesType "1" --> "1" JEP30-D10:ValueSetGroup ParameterValuesType "1" --> "1" LaTeX-Symbol ParameterValuesType "1" --> "1" MathML-Symbol ParameterValuesType "1" --> "1" SymbolDescription ParameterValuesType "1" --> "1" Nominal ParameterValuesType "1" --> "1" NegativeTolerance ParameterValuesType "1" --> "1" PositiveTolerance ParameterValuesType "1" --> "1" TotalTolerance ParameterValuesType "1" --> "1" ToleranceUOM ParameterValuesType "1" --> "1" Minimum ParameterValuesType "1" --> "1" Maximum ParameterValuesType "1" --> "1" ValueText ParameterValuesType "1" --> "1" FootnoteID </pre> <p>Use same element names as in ValueSetGroup when defining ValueText for Nominal, NegativeTolerance, PositiveTolerance, TotalTolerance, Minimum and/or Maximum.</p>
type	ParameterValuesType , m.math.type , ToleranceUOMType , UnitsForElectricalSpecificationType .
group	JEP30-D10:ValueSetGroup ,

NOTE 1: The element Units has the following annotation “Use same element names as in ValueSetGroup when defining ValueText for Nominal, NegativeTolerance, PositiveTolerance, TotalTolerance, Minimum and/or Maximum”.

4.7.4.2.1.2. Rule

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterSet/Parameter/RuleContext
diagram	<pre> classDiagram class Rule { <<JEP30-D10:ParameterRuleType>> } class LaTeX-and-MathML-RuleGroup { <<JEP30-D10:LaTeX-and-MathML-RuleGroup>> } class RuleContext { <<MinNomMaxRuleContextType>> } class RuleName { <<xs:string>> } class RuleAnnotation { <<xs:string>> } class LaTeX-Rule { <<xs:string>> } class MathML-Rule { <<m:math.type>> } class FootnoteID { <<xs:string>> } Rule "0..∞" --> LaTeX-and-MathML-RuleGroup : LaTeX-and-MathML-RuleGroup --> RuleContext : RuleContext "0..∞" --> FootnoteID : RuleContext "0..∞" --> RuleName : RuleContext "0..∞" --> RuleAnnotation : RuleContext "0..∞" --> LaTeX-Rule : RuleContext "0..∞" --> MathML-Rule : </pre>
type	ParameterRuleType , MinNomMaxRuleContextType , m:math.type
group	JEP30-D10:LaTeX-and-MathML-RuleGroup

4.7.4.2.1.2.1. Rule Context

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterSet/Parameter/RuleContext
diagram	<pre> classDiagram class RuleContext { <<MinNomMaxRuleContextType>> } class Nominal { <<EmptyType>> } class Minimum { <<EmptyType>> } class Maximum { <<EmptyType>> } RuleContext --> Nominal : RuleContext --> Minimum : RuleContext --> Maximum : </pre>
type	MinNomMaxRuleContextType , EmptyType .

4.7.4.3. Parameter Graph

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph
diagram 1 of 3	<pre> classDiagram class ElectricalSpecificationParameterGraphType { GraphTitle TestConditionDefinition ParameterDefinition Data-Array GraphFormula Formatting Footnote-Array } class ParameterGraph { <<ElectricalSpecificationParameterGraphType>> } ParameterGraph "1..∞" --> "1..∞" ElectricalSpecificationParameterGraphType class constraints </pre> <p>The diagram shows the <code>ElectricalSpecificationParameterGraphType</code> class with the following attributes:</p> <ul style="list-style-type: none"> <code>GraphTitle</code>: type <code>xs:string</code> <code>TestConditionDefinition</code>: type <code>ElectricalParametricGraphChartXAxisType</code> <code>ParameterDefinition</code>: type <code>ElectricalParameterGraphChartYAxisType</code> <code>Data-Array</code>: type <code>ElectricalSpecificationParameterGraphData-ArrayType</code> <code>GraphFormula</code>: type <code>JEP30-D10:GraphFormulaType</code> <code>Formatting</code>: type <code>JEP30-D10:GraphFormattingType</code> <code>Footnote-Array</code>: type <code>JEP30-D10:Footnote-ArrayType</code> <p>A multiplicity of <code>1..∞</code> connects the <code>ParameterGraph</code> class to the <code>ElectricalSpecificationParameterGraphType</code> class. A constraint block is also present.</p>
diagram 2 of 3	<pre> classDiagram class ElectricalParametricGraphChartXAxisType { TestConditionDefinition JEP30-D10:AxisParameterIdentityGroup AsciiUnits Units Formatting } class TestConditionDefinition { <<ElectricalParametricGraphChartXAxisType>> } TestConditionDefinition --> JEP30-D10:AxisParameterIdentityGroup JEP30-D10:AxisParameterIdentityGroup --> AsciiUnits JEP30-D10:AxisParameterIdentityGroup --> Units AsciiUnits <<AsciiUnitsForElectricalSpecificationType>> Units <<UnitsForElectricalSpecificationType>> class constraints </pre> <p>The diagram shows the <code>ElectricalParametricGraphChartXAxisType</code> class with the following attributes:</p> <ul style="list-style-type: none"> <code>TestConditionDefinition</code>: type <code>ElectricalParametricGraphChartXAxisType</code> <code>JEP30-D10:AxisParameterIdentityGroup</code> <code>AsciiUnits</code>: type <code>AsciiUnitsForElectricalSpecificationType</code> <code>Units</code>: type <code>UnitsForElectricalSpecificationType</code> <code>Formatting</code>: type <code>JEP30-D10:GraphChartX-AxisFormattingType</code> <p>A constraint block is also present.</p>
diagram 3 of 3	<pre> classDiagram class ElectricalParameterGraphChartYAxisType { ID JEP30-D10:AxisParameterIdentityGroup AsciiUnits Units Formatting } class ParameterDefinition { <<ElectricalParameterGraphChartYAxisType>> } ParameterDefinition "1..∞" --> "1..∞" ElectricalParameterGraphChartYAxisType JEP30-D10:AxisParameterIdentityGroup --> AsciiUnits JEP30-D10:AxisParameterIdentityGroup --> Units AsciiUnits <<AsciiUnitsForElectricalSpecificationType>> Units <<UnitsForElectricalSpecificationType>> class constraints </pre> <p>The diagram shows the <code>ElectricalParameterGraphChartYAxisType</code> class with the following attributes:</p> <ul style="list-style-type: none"> <code>ID</code>: type <code>xs:string</code> <code>JEP30-D10:AxisParameterIdentityGroup</code> <code>AsciiUnits</code>: type <code>AsciiUnitsForElectricalSpecificationType</code> <code>Units</code>: type <code>UnitsForElectricalSpecificationType</code> <code>Formatting</code>: type <code>JEP30-D10:GraphChartY-AxisFormattingType</code> <p>A multiplicity of <code>1..∞</code> connects the <code>ParameterDefinition</code> class to the <code>ElectricalParameterGraphChartYAxisType</code> class. A constraint block is also present.</p>

4.7.4.3. Parameter Graph (cont'd)

type	ElectricalSpecificationParameterGraphType , ElectricalParameterGraphChartXAxisType , AsciiUnitsForElectricalSpecificationType , UnitsForElectricalSpecificationType , JEP30-D10:GraphChartX-AxisFormattingType , ElectricalParameterGraphChartYAxisType , JEP30-D10:GraphChartY-AxisFormattingType , ElectricalSpecificationParameterGraphData-ArrayType , JEP30-D10:GraphFormulaType , JEP30-D10:GraphFormattingType , JEP30-D10:Footnote-ArrayType .
group	JEP30-D10:AxisParameterIdentityGroup

A [ParameterGraph](#) has 2 axis that are defined by the [TestConditionType](#) (The X-axis definition), and the [ParameterType](#) (The Y-axis definition). Each axis is labelled by the [AxisTitle](#). When possible, the [LaTeX-Symbol](#) or the [MathML-Symbol](#) which represents the [AxisTitle](#) should be added to the PartModel file and should represent a standards-based symbol as defined in the appropriate Terms and Definitions standards. If appropriate, a more detailed [Description](#) can be used to describe the definition of the [AxisTitle](#). Each axis will also have a pre-defined set of [Units](#) but can be optionally excluded for those axis' which are unitless.

Note that the [ParameterDefinition](#) is unbounded whereas the [TestConditionDefinition](#) is bounded to a single instance. This is to cater for those graphs in which there are 2 or more y-axis, each with their own definition.

The graph can either be captured under the [Data-Array](#) or represented via a [GraphFormula](#) (*A string representing the equation of the ParameterDefinition relationship to the Test TestConditionDefinition*).

4.7.4.3.1. Formatting

The Formatting is an optional set of data that enables the user to re-create the graph for visualization purposes. Formatting applies to the following

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/TestConditionDefinition/Formatting.
diagram	<pre> classDiagram class JEP30-D10:GraphChartX-AxisFormattingType { Range Inverted Scale Position GraphAxisRangeType { Minimum Maximum } GraphAxisScaleType { Linear Logarithmic } GraphAxisScaleLinearType { Step } GraphAxisScaleLogarithmicType { Natural Base } } JEP30-D10 < -- GraphAxisRangeType JEP30-D10 < -- GraphAxisScaleType JEP30-D10 < -- GraphAxisScaleLinearType JEP30-D10 < -- GraphAxisScaleLogarithmicType Range < -- Inverted Range < -- Scale Range < -- Position Inverted < -- GraphAxisRangeType Scale < -- GraphAxisScaleType Position < -- GraphChartXAxisPositionType GraphAxisRangeType < -- Minimum GraphAxisRangeType < -- Maximum GraphAxisScaleType < -- Linear GraphAxisScaleType < -- Logarithmic GraphAxisScaleLinearType < -- Step GraphAxisScaleLogarithmicType < -- Natural GraphAxisScaleLogarithmicType < -- Base </pre>
type	GraphChartXAxisFormattingType, GraphAxisRangeType, JEP30-D10:EmptyType, GraphAxisScaleType, GraphAxisScaleLinearType, GraphAxisScaleLogarithmicType, GraphChartXAxisPositionType.

4.7.4.3.1. Formatting (cont'd)

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/ParameterDefinition/Formatting .
diagram	<pre> classDiagram class JEP30-D10:GraphChartY-AxisFormattingType { Range Inverted Scale Position } class GraphAxisRangeType { Minimum Maximum } class GraphAxisScaleType { Linear Logarithmic } class GraphAxisScaleLinearType { Step } class GraphAxisScaleLogarithmicType { Natural Base } JEP30-D10 < -- Range JEP30-D10 < -- Inverted JEP30-D10 < -- Scale JEP30-D10 < -- Position Range < -- GraphAxisRangeType Inverted < -- JEP30-D10:EmptyType Scale < -- GraphAxisScaleType Position < -- GraphChartYAxisPositionType GraphAxisRangeType < -- Minimum GraphAxisRangeType < -- Maximum GraphAxisScaleType < -- Linear GraphAxisScaleType < -- Logarithmic GraphAxisScaleLinearType < -- Step GraphAxisScaleLogarithmicType < -- Natural GraphAxisScaleLogarithmicType < -- Base </pre> <p>The diagram illustrates the structure of the <code>JEP30-D10:GraphChartY-AxisFormattingType</code> class. It includes attributes for <code>Range</code>, <code>Inverted</code>, <code>Scale</code>, and <code>Position</code>. The <code>Range</code> attribute is associated with the <code>GraphAxisRangeType</code> class, which contains <code>Minimum</code> and <code>Maximum</code> attributes. The <code>Inverted</code> attribute is associated with the <code>JEP30-D10:EmptyType</code>. The <code>Scale</code> attribute is associated with the <code>GraphAxisScaleType</code> class, which contains <code>Linear</code> and <code>Logarithmic</code> subtypes. The <code>Linear</code> subtype is associated with the <code>GraphAxisScaleLinearType</code> class, which contains a <code>Step</code> attribute. The <code>Logarithmic</code> subtype is associated with the <code>GraphAxisScaleLogarithmicType</code> class, which contains <code>Natural</code> and <code>Base</code> attributes, with <code>Base</code> having a default value of <code>10.0</code>. The <code>Position</code> attribute is associated with the <code>GraphChartYAxisPositionType</code> class.</p>
type	<code>GraphChartYAxisFormattingType</code> , <code>GraphAxisRangeType</code> , <code>JEP30-D10:EmptyType</code> , <code>GraphAxisScaleType</code> , <code>GraphAxisScaleLinearType</code> , <code>GraphAxisScaleLogarithmicType</code> , <code>GraphChartYAxisPositionType</code> .

The axis range which is usually defined from minimum to maximum can be inverted to show a graph going from maximum to minimum. The scale can be defined in either a linear step amount, a natural logarithm, or a logarithm of the specified base. The `Base` log is set to a default of `Base 10` but can be defined to any base number.

The `Position` enumerated list for the `GraphChartXAxisPositionType` is

- Top
- Bottom

And for the `GraphChartYAxisPositionType`, the enumerated values are

- Left
- Right

4.5.4.3.1. Formatting (cont'd)

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/Formatting</code>
diagram	<pre> classDiagram class GraphFormattingType { <<DisplayType>> <<Location>> <<Legend>> <<VerticalPosition>> <<HorizontalPosition>> } class GraphDisplayType { <<GraphFormattingType>> } class GraphLegendType { <<GraphFormattingType>> } class GraphLegendLocationType { <<Location>> } class GraphLegendVerticalPositionType { <<VerticalPosition>> } class GraphLegendHorizontalPositionType { <<HorizontalPosition>> } GraphFormattingType "3" --> "1" GraphDisplayType GraphFormattingType "3" --> "1" GraphLegendType GraphFormattingType "3" --> "1" GraphLegendLocationType GraphFormattingType "3" --> "1" GraphLegendVerticalPositionType GraphFormattingType "3" --> "1" GraphLegendHorizontalPositionType </pre>
type	<code>GraphFormattingType, GraphDisplayType, GraphLegendType, GraphLegendLocationType, GraphLegendVerticalPositionType, GraphLegendHorizontalPositionType.</code>

The body of the graph can be formatted under the `GraphFormattingType`. The `DisplayType` enumerated list is

- Line
- Bar

The graph Legend can also be positioned around the graph in any of the following locations:

- Location
 - Inside Graph,
 - Outside Graph,
- Vertical Position
 - Top,
 - Center,
 - Bottom,
- Horizontal Position
 - Left,
 - Center,
 - Right.

4.7.4.3.1.1. Linking the Data-Array to the Appropriate Parameter Definition

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph
diagram	<pre> classDiagram class ElectricalSpecificationParameterGraphType { GraphTitle xsstring TestConditionDefinition ElectricalParametricGraphChartXAxisType ParameterDefinition ElectricalParameterGraphChartYAxisType [1..oo] ParameterGraph ElectricalSpecificationParameterGraphType [1..oo] Data-Array ElectricalSpecificationParameterGraphData-ArrayType [1..oo] GraphFormula JEP30-D10:GraphFormulaType [1..oo] Formatting JEP30-D10:GraphFormattingType Footnote-Array JEP30-D10:Footnote-ArrayType } class ElectricalParameterGraphChartYAxisType { ID xsstring JEP30-D10:AxisParameterIdentityGroup AsciiUnits AsciiUnitsForElectricalSpecificationType Units UnitsForElectricalSpecificationType Formatting JEP30-D10:GraphChartYAxisFormattingType } class ElectricalSpecificationParameterGraphData-ArrayType { ParameterDefinitionID xsstring PlotTestCondition ElectricalSpecificationGraphPlotConditionType [0..oo] Data JEP30-D10:GraphDataType [1..oo] TestMethod xsstring Formatting JEP30-D10:GraphDataFormattingType } class ParameterGraph { constraints unique ElectricalParameterDefinitionID selector ParameterDefinition field ID keyref ElectricalParameterDefinitionKeyRef refer ElectricalParameterDefinitionID selector Data-Array field ParameterDefinitionID } </pre>
type	ElectricalSpecificationParameterGraphType , ElectricalParametricGraphChartXAxisType , ElectricalParameterGraphChartYAxisType , AsciiUnitsForElectricalSpecificationType , UnitsForElectricalSpecificationType , JEP30-D10:GraphChartYAxisFormattingType , ElectricalSpecificationParameterGraphData-ArrayType , ElectricalSpecificationGraphPlotConditionType , JEP30-D10:GraphDataType , JEP30-D10:GraphDataFormattingType , JEP30-D10:GraphFormulaType , JEP30-D10:GraphFormattingType ,
group	JEP30-D10:AxisParameterIdentityGroup

When populating the [Data-Array](#) for a given graph, the set of data is referenced to the specific [ParameterDefinition](#) via the [ParameterDefinitionID](#). The process is replicated for each [ParameterDefinition](#) if there are 2 or more vertical [ParameterDefinition](#) axis defined. Each set of data captured under the [Data-Array](#) is now tied to the appropriate [ParameterDefinition](#) axis.

4.7.4.3.1.1. Linking the Data-Array to the Appropriate Parameter Definition (cont'd)

The data plot on the graph, is a set of points associated with the intersection of these two axis and can be either captured under the *Data-Array*, or represented via a *GraphFormula*.

A rule may also be defined as a function on one electrical parameter against a second electrical parameter. The aging rate of the materials in a device is generally defined on a simple logarithmic curve. In this example, the Aging Rate, shown in Figure 42 can be captured with a *GraphFormula* under the *ParameterGraph* as shown below.

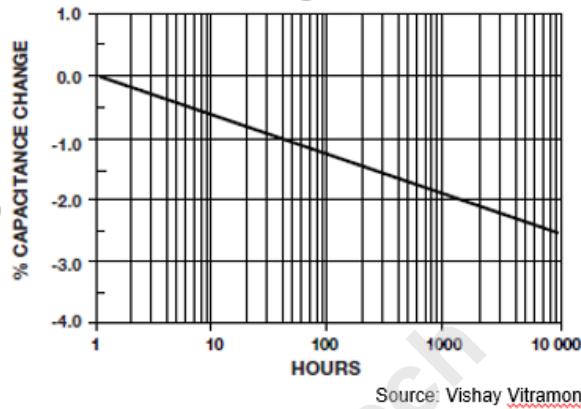


Figure 42 – Aging Rate

This is a straight line; however, the scale of the HOURS axis is logarithmic. HOURS is represented by the symbol "h" and the Capacitance change is represented by the symbol "\Delta C". The formula of the graph plot is $\Delta C = \text{LOG}(h^{-0.625})$. The xml fragment shown below represents the data plotted in Figure 44 — Aging Rate

```

<ParameterGraph>
  <GraphTitle>Aging Rate</GraphTitle>
  <TestConditionDefinition>
    <AxisTitle>HOURS</AxisTitle>
    <LaTeX-Symbol>h</LaTeX-Symbol>
    <SymbolDescription>Hours</SymbolDescription>
    <Units>
      <Time>h</Time>
    </Units>
    <Formatting>
      <Range>
        <Minimum>1</Minimum>
        <Maximum>10000</Maximum>
      </Range>
      <Scale>
        <Logarithmic>
          <Base>10</Base>
        </Logarithmic>
      </Scale>
      <Position>Bottom</Position>
    </Formatting>
  </TestConditionDefinition>
</ParameterGraph>

```

4.7.4.3.1.1. Linking the Data-Array to the Appropriate Parameter Definition (cont'd)

```
</Formatting>
</TestConditionDefinition>
<ParameterDefinition>
  <ID>Plot1</ID>
  <AxisTitle>% CAPACITANCE CHANGE</AxisTitle>
  <LaTeX-Symbol>\Delta C</LaTeX-Symbol>
  <Description>Capacitance Change</Description>
  <Units>
    <Percent/>
  </Units>
  <Formatting>
    <Range>
      <Minimum>-4</Minimum>
      <Maximum>1</Maximum>
    </Range>
    <Scale>
      <Linear>
        <Step>1</Step>
      </Linear>
    </Scale>
    <Position>Left</Position>
  </Formatting>
</ParameterDefinition>
<GraphFormula>
  <LaTeX-Rule> $\log_{10}(h^{-0.625})$ </LaTeX-Rule>
</GraphFormula>
</ParameterGraph>
```

4.7.4.3.2. Data-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/Data-Array
diagram	<pre> classDiagram class ElectricalSpecificationParameterGraphData-ArrayType { ParameterDefinitionID PlotTestCondition Data TestMethod Formatting } class PlotTestCondition { type ElectricalSpecificationGraphPlotConditionType } class Data { type JEP30-D10:GraphDataType } class ParameterValue { type MinNomMaxValueSetType } class MinNomMaxValueSetType { Minimum Nominal Maximum FootnoteID } class TestConditionValue class JEP30-D10:GraphDataType class TestMethod class Formatting ElectricalSpecificationParameterGraphData-ArrayType "1..>" Data ElectricalSpecificationParameterGraphData-ArrayType "0..>" PlotTestCondition Data "1..>" ParameterValue Data "0..>" TestConditionValue Data "0..>" JEP30-D10:GraphDataType Data "0..>" TestMethod Data "0..>" Formatting PlotTestCondition "0..>" TestConditionValue ParameterValue "0..>" MinNomMaxValueSetType MinNomMaxValueSetType "0..>" FootnoteID </pre> <p>The diagram illustrates the UML class structure for the <code>ElectricalSpecificationParameterGraphData-ArrayType</code>. It includes the following classes and their associations:</p> <ul style="list-style-type: none"> ElectricalSpecificationParameterGraphData-ArrayType (superclass): <ul style="list-style-type: none"> Associations: <code>Data</code> (multiplicity 1..∞), <code>PlotTestCondition</code> (multiplicity 0..∞). PlotTestCondition (sub-class of <code>ElectricalSpecificationParameterGraphData-ArrayType</code>): <ul style="list-style-type: none"> Associations: <code>TestConditionValue</code> (multiplicity 0..∞). Data (sub-class of <code>ElectricalSpecificationParameterGraphData-ArrayType</code>): <ul style="list-style-type: none"> Associations: <code>ParameterValue</code> (multiplicity 1..∞), <code>JEP30-D10:GraphDataType</code> (multiplicity 0..∞), <code>TestMethod</code> (multiplicity 0..∞), <code>Formatting</code> (multiplicity 0..∞). ParameterValue (sub-class of <code>Data</code>): <ul style="list-style-type: none"> Associations: <code>MinNomMaxValueSetType</code> (multiplicity 0..∞). MinNomMaxValueSetType (sub-class of <code>ParameterValue</code>): <ul style="list-style-type: none"> Associations: <code>FootnoteID</code> (multiplicity 0..∞).
type	ElectricalSpecificationParameterGraphData-ArrayType , ElectricalSpecificationGraphPlotConditionType , JEP30-D10:GraphDataType , MinNomMaxValueSetType , JEP30-D10:GraphDataFormattingType .

Each *Data* set consisting of the *TestConditionValue*, and the *ParameterValue* represents one point of the data plot on the graph. Various examples of Parameter Graphs are shown below, and representation of those parameter graph examples accompany each graph.

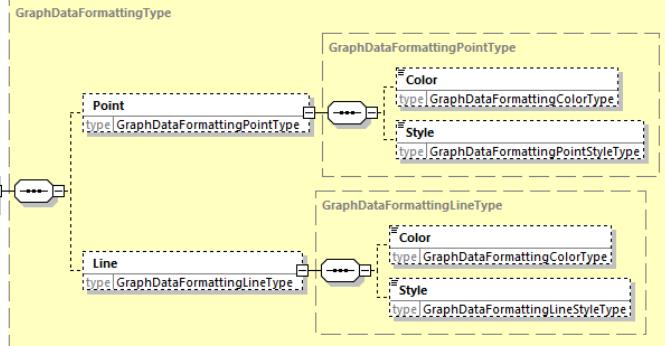
4.7.4.3.2. Data-Array (cont'd)

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/Data-Array/PlotTestCondition .
diagram	<pre> classDiagram class ElectricalSpecificationGraphPlotConditionType { <<ElectricalSpecificationGraphPlotConditionType>> <<JEP30-D10:ParameterIdentityGroup>> <<JEP30-D10:LaTeX-and-MathML-RuleGroup>> <<PlotTestCondition>> <<Legend>> } class Name { <<Name>> <<JEP30-D10:ParameterIdentityGroup>> } class StandardsAuthorityBodyID { <<StandardsAuthorityBodyID>> <<Property-ID>> } class PropertyID { <<Property-ID>> } class LaTeXSymbol { <<LaTeX-Symbol>> <<MathML-Symbol>> } class MathMLSymbol { <<MathML-Symbol>> } class SymbolDescription { <<SymbolDescription>> <<Definition>> <<FootnoteID>> } class Definition { <<Definition>> } class FootnoteID { <<FootnoteID>> 0..> } class Value { <<Value>> <<JEP30-D10:ValueSetType>> } class AsciiUnits { <<AsciiUnits>> <<UnitsForElectricalSpecificationType>> } class Units { <<Units>> <<UnitsForElectricalSpecificationType>> } class RuleName { <<RuleName>> <<RuleAnnotation>> <<RuleContext>> <<MinNomMaxRuleContextType>> } class RuleAnnotation { <<RuleAnnotation>> } class RuleContext { <<RuleContext>> <<MinNomMaxRuleContextType>> } class MinNomMaxRuleContextType { <<MinNomMaxRuleContextType>> } class LaTeXRule { <<LaTeX-Rule>> <<MathML-Rule>> } class MathMLRule { <<MathML-Rule>> <<m:math.type>> } class FootnoteID { <<FootnoteID>> 0..> } class PlotTestCondition { <<PlotTestCondition>> <<ElectricalSpecificationGraphPlotConditionType>> 0..> } class Legend { <<Legend>> <<JEP30-D10:GraphDataFormattingLegendType>> } </pre>
type	ElectricalSpecificationGraphPlotConditionType , m.math.type , JEP30-D10:ValueSetType , AsciiUnitsForElectricalSpecificationType , UnitsForElectricalSpecificationType , MinNomMaxRuleContextType , JEP30-D10:GraphDataFormattingLegendType .
group	JEP30-D10:ParameterIdentityGroup , JEP30-D10:LaTeX-and-MathML-RuleGroup

Depending upon the type of test condition, its value may be a string or label that describes the [PlotTestCondition](#), in which case the value is populated under the [Legend](#) branch.

If, however, the [PlotTestCondition](#) can be represented by a [Symbol](#) or a [Rule](#), then this should be written to the top branch under the [PlotTestCondition](#) as opposed to the [Legend](#) branch as a string. The [Symbol](#) should represent a standards-based symbol as defined in the appropriate Terms and Definitions standards. If appropriate, a more detailed [SymbolDescription](#) can be defined to describe the definition of the [PlotTestCondition](#). The [Symbol](#) can have a pre-defined set of [Units](#) but can be optionally excluded for those [PlotTestCondition](#) which are unitless.

4.7.4.3.2. Data-Array (cont'd)

path	PartModel/ThermalSection/ThermalFamily-Array/ThermalFamily/ThermalData/ThermalMetrics-Array/ThermalMetricGraph/Data-Array/Formatting .
diagram	
type	GraphDataFormattingType , GraphDataFormattingPointType , GraphDataFormattingLineType , GraphDataFormattingColorType , GraphDataFormattingPointStyleType , GraphDataFormattingLineStyleType .

The data points can also be formatted. Individual data points can have the following styles

- Point Styles are
 - Circle,
 - Square,
 - Triangle,
 - None.
- Line Style are
 - Solid,
 - Dash,
 - Dot,
 - Dash-dot,
 - Dash-dash-dot,
 - None.
- Colors are
 - Red,
 - Green,
 - Blue,
 - Orange,
 - Brown,
 - Pink,
 - Purple,
 - Yellow,
 - Black.

The xml fragment shown below represents the data plotted in Figure 44 — Capacitance Value versus Temperature.

4.7.4.3.2. Data-Array (cont'd)

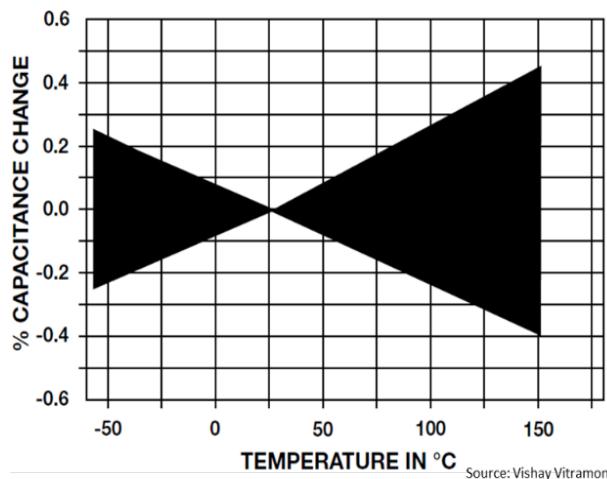


Figure 43 — Capacitance Value versus Temperature

```

<ParameterGraph>
  <GraphTitle>Capacitance Value versus Temperature</GraphTitle>
  <TestConditionDefinition>
    <AxisTitle>Temperature</AxisTitle>
    <LaTeX-Symbol>T</LaTeX-Symbol>
    <Description>Temperature in DegC</Description>
    <Units>
      <Temperature>DegC</Temperature>
    </Units>
    <Formatting>
      <Range>
        <Minimum>-50</Minimum>
        <Maximum>175</Maximum>
      </Range >
      <Scale>
        <Linear>
          <Step>25</Step>
        </Linear>
      </Scale>
      <Position>Bottom</ Position >
    </Formatting>
  </TestConditionDefinition>
  <ParameterDefinition>
    <ID>Plot1</ID >
    <AxisTitle>% CAPACITANCE CHANGE</AxisTitle>
    <LaTeX-Symbol> $\Delta C$ </LaTeX-Symbol>
    <Description>Capacitance change</Description>
    <Units>
      <Percent/>
    </Units>
  </ParameterDefinition>

```

4.7.4.3.2. Data-Array (cont'd)

```
<Formatting>
  <Range>
    <Minimum>-0.6</Minimum>
    <Maximum>0.6</Maximum>
  </Range >
  <Scale>
    <Linear>
      <Step>0.1</Step>
    </Linear>
  </Scale>
  <Position>Left</ Position >
</Formatting >
</ParameterType>
<Data-Array>
  <ParameterDefinitionID> Plot1</ParameterDefinitionID>
  <Data>
    <TestConditionValue>-55</TestConditionValue>
    <ParameterValue>
      <Minimum>-0.25</Minimum>
      <Maximum>0.25</Maximum>
    </ParameterValue>
  </Data>
  <Data>
    <TestConditionValue>25</TestConditionValue>
    <ParameterValue>
      <Minimum>0.0</Minimum>
      <Maximum>0.0</Maximum>
    </ParameterValue>
  </Data>
  <Data>
    <TestConditionValue>150</TestConditionValue>
    <ParameterValue>
      <Minimum>-0.4</Minimum>
      <Maximum>0.45</Maximum>
    </ParameterValue>
  </Data>
  <Formatting>
    <Point>
      <Color>Black</Color>
      <Style>None</Style>
    </Point>
    <Line>
      <Color>Black</Color>
      <Style>Solid</Style>
    </ Line>
  </Formatting>
</Data-Array>
```

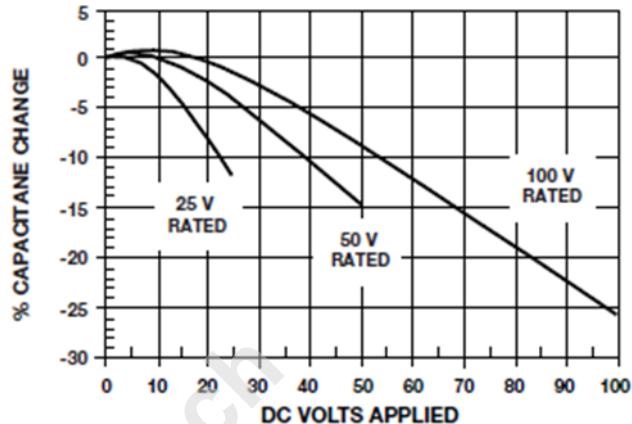
4.7.4.3.2. Data-Array (cont'd)

```

<Formatting>
  <DisplayType>Line</DisplayType>
  <Legend>
    <Location>Outside Graph</Location>
    <VerticalPosition>Center</VerticalPosition>
    <HorizontalPosition>Right</HorizontalPosition>
  </Legend>
</Formatting>
</ParameterGraph>

```

The *TestCondition* under the *Data*, represents different plots on the same graph, as shown in Figure 45 and captured below in the XML file.



Source: Vishay Vitramon

Figure 44 — Voltage Coefficient of Capacitance

```

<ParameterGraph>
  <GraphTitle>Voltage Coefficient of Capacitance</GraphTitle>
  <TestConditionDefinition>
    <AxisTitle>DC Volts Applied</AxisTitle>
    <LaTeX-Symbol>V_{DC}</LaTeX-Symbol>
    <Description>DC Volts Applied</Description>
    <Units>
      <Voltage>V</Voltage>
    </Units>
    <Formatting>
      <Range>
        <Minimum>0</Minimum>
        <Maximum>100</Maximum>
      </Range>
      <Scale>
        <Linear>
          <Step>10</Step>
        </Linear>
      </Scale>
      <Position>Bottom</Position>
    </Formatting>
  </TestConditionType>
  <ParameterType>
    <ID>Plot1</ID>
    <AxisTitle>% CAPACITANCE CHANGE</AxisTitle>

```

4.7.4.3.2. Data-Array (cont'd)

```
<LaTeX-Symbol>\Delta C</LaTeX-Symbol>
<Description>Capacitance change</Description>
<Units>
    <Percent/>
</Units>
<Formatting>
    <Range>
        <Minimum>-30</Minimum>
        <Maximum>5</Maximum>
    </Range >
    <Scale>
        <Linear>
            <Step>5</Step>
        </Linear>
    </Scale>
    <Position>Left</ Position >
</Formatting >
</ParameterType>
<Data-Array>
    <PlotTestCondition>
        <LaTeX-Symbol>VDC</LaTeX-Symbol>
        <SymbolDescription>Rated Voltage DC</SymbolDescription>
        <Value>25</Value>
        <Units>
            <Voltage>V</Voltage>
        </Units>
    </PlotTestCondition>
    <Data>
        <TestConditionValue>0</TestConditionValue>
        <ParameterValue>
            <Nominal>0</Nominal>
        </ParameterValue>
    </Data>
    <Data>
        <TestConditionValue>5</TestConditionValue>
        <ParameterValue>
            <Nominal>0</Nominal>
        </ParameterValue>
    </Data>
    <Data>
        <TestConditionValue>10</TestConditionValue>
        <ParameterValue>
            <Nominal>-2</Nominal>
        </ParameterValue>
    </Data>
    <Data>
        <TestConditionValue>15</TestConditionValue>
        <ParameterValue>
            <Nominal>-5</Nominal>
        </ParameterValue>
    </Data>
```

4.7.4.3.2. Data-Array (cont'd)

```
<Data>
  <TestConditionValue>25</TestConditionValue>
  <ParameterValue>
    <Nominal>-12</Nominal>
  </ParameterValue>
</Data>
</Data-Array>
<Data-Array>
  <PlotTestCondition>
    <LaTeX-Symbol>VDC</LaTeX-Symbol>
    <SymbolDescription>Rated Voltage DC</SymbolDescription>
    <Value>50</Value>
    <Units>
      <Voltage>V</Voltage>
    </Units>
  </PlotTestCondition>
  <Data>
    <TestConditionValue>0</TestConditionValue>
    <ParameterValue>
      <Nominal>0</Nominal>
    </ParameterValue>
  </Data>
  <Data>
    <TestConditionValue>4</TestConditionValue>
    <ParameterValue>
      <Nominal>1</Nominal>
    </ParameterValue>
  </Data>
  <Data>
    <TestConditionValue>10</TestConditionValue>
    <ParameterValue>
      <Nominal>0</Nominal>
    </ParameterValue>
  </Data>
  <Data>
    <TestConditionValue>20</TestConditionValue>
    <ParameterValue>
      <Nominal>-2</Nominal>
    </ParameterValue>
  </Data>
  <Data>
    <TestConditionValue>50</TestConditionValue>
    <ParameterValue>
      <Nominal>-15</Nominal>
    </ParameterValue>
  </Data>
</Data-Array>
<Data-Array>
  <PlotTestCondition>
    <LaTeX-Symbol>VDC</LaTeX-Symbol>
    <SymbolDescription>Rated Voltage DC</SymbolDescription>
    <Value>100</Value>
```

4.7.4.3.2. Data-Array (cont'd)

```
<Units>
    <Voltage>V</Voltage>
</Units>
</PlotTestCondition>
<Data>
    <TestConditionValue>0</TestConditionValue>
    <ParameterValue>
        <Nominal>0</Nominal>
    </ParameterValue>
</Data>
<Data>
    <TestConditionValue>4</TestConditionValue>
    <ParameterValue>
        <Nominal>1</Nominal>
    </ParameterValue>
</Data>
<Data>
    <TestConditionValue>10</TestConditionValue>
    <ParameterValue>
        <Nominal>1</Nominal>
    </ParameterValue>
</Data>
<Data>
    <TestConditionValue>20</TestConditionValue>
    <ParameterValue>
        <Nominal>-1</Nominal>
    </ParameterValue>
</Data>
<Data>
    <TestConditionValue>50</TestConditionValue>
    <ParameterValue>
        <Nominal>-9</Nominal>
    </ParameterValue>
</Data>
<Data>
    <TestConditionValue>100</TestConditionValue>
    <ParameterValue>
        <Nominal>-26</Nominal>
    </ParameterValue>
</Data>
</Data-Array>
</ParameterGraph>
```

4.7.4.3.3. Graph Formula

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph
diagram	<pre> classDiagram class GraphFormula { <<JEP30-D10:GraphFormulaType>> } class RuleName { <<xs:string>> } class RuleAnnotation { <<xs:string>> } class RuleContext { <<MinNomMaxRuleContextType>> } class MinNomMaxRuleContextType { class Nominal { <<EmptyType>> } class Minimum { <<EmptyType>> } class Maximum { <<EmptyType>> } } class LaTeX-and-MathML-RuleGroup { <<MinNomMaxRuleContextType>> } class LaTeX-Rule { <<xs:string>> } class MathML-Rule { <<m:math.type>> } class FootnoteID { <<xs:string>> } class TestMethod { <<xs:string>> } GraphFormula "1..>" -- "0..>" RuleName GraphFormula "1..>" -- "0..>" RuleAnnotation GraphFormula "1..>" -- "0..>" RuleContext RuleContext "*" -- "1..>" Nominal RuleContext "*" -- "1..>" Minimum RuleContext "*" -- "1..>" Maximum RuleName "*" -- "1..>" LaTeX-and-MathML-RuleGroup RuleAnnotation "*" -- "1..>" LaTeX-and-MathML-RuleGroup RuleContext "*" -- "1..>" LaTeX-and-MathML-RuleGroup LaTeX-and-MathML-RuleGroup "*" -- "1..>" LaTeX-Rule LaTeX-and-MathML-RuleGroup "*" -- "1..>" MathML-Rule LaTeX-and-MathML-RuleGroup "*" -- "1..>" FootnoteID LaTeX-and-MathML-RuleGroup "*" -- "1..>" TestMethod </pre> <p>The diagram illustrates the UML Class Diagram for the JEP30-D10:GraphFormulaType. It defines the following classes and their associations:</p> <ul style="list-style-type: none"> GraphFormula (type JEP30-D10:GraphFormulaType) has multiplicity 1..> and is associated with RuleName, RuleAnnotation, and RuleContext. RuleName (type xs:string) and RuleAnnotation (type xs:string) have multiplicity 1..> and are associated with GraphFormula. RuleContext (type MinNomMaxRuleContextType) has multiplicity 1..> and is associated with GraphFormula. MinNomMaxRuleContextType contains three subclasses: Nominal (type EmptyType), Minimum (type EmptyType), and Maximum (type EmptyType). RuleContext has multiplicity * and is associated with Nominal, Minimum, and Maximum. LaTeX-and-MathML-RuleGroup (type MinNomMaxRuleContextType) has multiplicity 1..> and is associated with GraphFormula. LaTeX-and-MathML-RuleGroup has multiplicity * and is associated with LaTeX-Rule, MathML-Rule, FootnoteID, and TestMethod. LaTeX-Rule (type xs:string) and MathML-Rule (type m:math.type) have multiplicity 1..> and are associated with LaTeX-and-MathML-RuleGroup. FootnoteID (type xs:string) and TestMethod (type xs:string) have multiplicity 1..> and are associated with LaTeX-and-MathML-RuleGroup.
type	JEP30-D10:GraphFormulaType, MinNomMaxRuleContextType, m:math.type
group	LaTeX-and-MathML-RuleGroup

4.7.5. Truth Table

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/TruthTable
diagram	<pre> classDiagram class TruthTableType { ID : xs:string TerminalName : xs:string DataRows : DataRowLogicStatesType } class TruthTableHeaderRowType { HeaderRow : TruthTableHeaderRowType Cell : TruthTableCellType } class TruthTableCellType { Label : xs:string From : positiveInteger To : positiveInteger } class DataRowLogicStatesType { HighState : JEP30-D10:EmptyType LowState : JEP30-D10:EmptyType NegativeEdgeTrigger : JEP30-D10:EmptyType PositiveEdgetrigger : JEP30-D10:EmptyType HighImpedance : JEP30-D10:EmptyType Toggle : JEP30-D10:EmptyType TerminalName : xs:string Other : xs:string } TruthTableType "0..∞" -- "0..∞" TerminalName TruthTableType "1..∞" -- "1..∞" DataRows TruthTableType "0..∞" -- "0..∞" HeaderRow TruthTableHeaderRowType "1..∞" -- "1..∞" Cell TruthTableHeaderRowType "1..∞" -- "1..∞" DataRows </pre>
type	TruthTableType , TruthTableHeaderRowType , TruthTableCellType , DataRowLogicStatesType

A [TruthTable](#) is a breakdown of a logic function by listing all possible values the function can attain. Such a table typically contains several rows and columns, with the top row representing the logical variables and combinations, in increasing complexity leading up to the final function.

The [HeaderRow](#) branch captures the multi row header structure of a table for ease of human readability. It is not required for the capture of the actual logic steps that represents the function of the device and is therefore an optional branch.

The columns of the table are made up of the [TerminalNames](#) and the table rows are captured under the [DataRowLogicStates](#) branch. The value under the data row represent the values for each column in sequence to match the sequence of the [TerminalNames](#). If there are 3 terminal names, then there should be 3 values within each data row container.

4.7.5 Truth Table (cont'd)

Table 4 — NOR logic States

Input		Output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Table 4 can be represented in the xml file as follows:

```

<TruthTable>
  <ID>Logic Function ID 1</ID>
  <HeaderRow>
    <Cell>
      <Label>Input</Label>
      <From>1</From>
      <To>2</To>
    </Cell>
    <Cell>
      <Label>Output</Label>
    </Cell>
  </HeaderRow>
  <TerminalName>A</TerminalName>
  <TerminalName>B</TerminalName>
  <TerminalName>Y</TerminalName>
  <DataRowLogicStates>
    <LowState/>
    <LowState/>
    <HighState/>
  </DataRowLogicStates>
  <DataRowLogicStates>
    <LowState/>
    <HighState/>
    <LowState/>
  </DataRowLogicStates>
  <DataRowLogicStates>
    <HighState/>
    <LowState/>
    <LowState/>
  </DataRowLogicStates>
  <DataRowLogicStates>
    <HighState/>
    <HighState/>
    <HighState/>
  </DataRowLogicStates>
</TruthTable>
```

4.7.5 Truth Table (cont'd)

The enumerated values for the Truth Table Value element are:

1. "1"
2. "0"
3. "Negative Edge trigger"
4. "Positive Edge trigger"
5. "↓"
6. "↑"
7. "Z"
8. "High Impedance"
9. "Toggle"
10. "N/A"

4.7.6 ESD

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array
diagram	<pre> classDiagram class ESD { type ESD-ArrayType } class ESD-ArrayType class ESDType class ID { type xs:string } class HBM-HumanBodyModel { type HBM-Type } class CDM-ChargedDeviceModel { type CDM-Type } ESD "0..1" *-- "1..1" ESD-ArrayType ESD "0..1" *-- "1..1" ESDType ESDType "1..1" *-- "1..1" ID ESDType "0..1" *-- "1..1" HBM-HumanBodyModel HBM-HumanBodyModel "0..1" *-- "1..1" CDM-ChargedDeviceModel </pre>
type	ESD-ArrayType, ESDType, HBM-Type, CDM-Type.

4.7.6.1. HBM – Human Body Model

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array/ESD/HBM-HumanBodyModel</code>
diagram	<pre> classDiagram class HBM-Type { Classification } class HBM-HumanBodyModel { <<HBM-Type>> } HBM-HumanBodyModel --> HBM-Type : HBM-Type --> ESD-VoltageRange ESD-VoltageRange --> ESD-VoltageRangeType ESD-VoltageRangeType { LowerLimit xs:decimal UpperLimit xs:decimal Value xs:decimal UOM StaticVoltageUOMType } </pre>
type	<code>HBM-Type, HBM-ClassificationType, ESD-VoltageRangeType, StaticVoltageUOMType.</code>

ESD sensitive components are classified according to their *HBM-HumanBodyModel* withstand voltage, regardless of polarity, as defined in ANSI/ESDA/JEDEC JS-001-2017, Human Body Model (HBM) – Component Level. The enumerated values of the *HBM-Classification* are defined in the “Component Classification” section of this publication. The *UOM* enumerated values are V and kV.

4.7.6.2. CDM – Charged Device Model

path	<code>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array/ESD/CDM-ChargedDeviceModel</code>
diagram	<pre> classDiagram class CDM-Type { Classification } class CDM-ChargedDeviceModel { <<CDM-Type>> } CDM-ChargedDeviceModel --> CDM-Type : CDM-Type --> ESD-VoltageRange ESD-VoltageRange --> ESD-VoltageRangeType ESD-VoltageRangeType { LowerLimit xs:decimal UpperLimit xs:decimal Value xs:decimal UOM StaticVoltageUOMType } </pre>
type	<code>CDM-Type, CDM-ClassificationType, ESD-VoltageRangeType, StaticVoltageUOMType.</code>

ESD sensitive components are also classified according to their *CDM-ChargedDeviceModel* withstand voltage in accordance with ANSI/ESDA/JEDEC JS-002-2018, Charged Device Model (CDM) - Device Level. The enumerated values of the *CDM-Classification* are defined in the “CDM Classification Criteria” section of this publication.

4.8. Schematic Data - Array

path	PartModel/ElectricalSection/SchematicData-Array
diagram	<p>SchematicData-ArrayType</p> <p>SchematicData-Array type SchematicData-ArrayType</p> <p>SchematicData-Type type SchematicDataType</p> <p>ID type xs:string</p> <p>Symbol-Array type JEP30-D10:Symbol-ArrayType</p> <p>RequiredCircuitry-Array type RequiredCircuitry-ArrayType</p> <p>ds:Signature type ds:SignatureType</p> <p>constraints</p>
type	SchematicData-Array , SchematicDataType , Symbol-ArrayType , RequiredCircuitry-ArrayType , ds:SignatureType

4.8.1. Symbol - Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array
diagram	<p>JEP30-D10:Symbol-ArrayType</p> <p>Symbol-Array type JEP30-D10:Symbol-ArrayType</p> <p>Symbol-Type type SymbolType</p> <p>ID type xs:string</p> <p>SymbolGraphics-Array type SymbolGraphics-ArrayType</p> <p>ds:Signature type ds:SignatureType</p> <p>constraints</p>
type	Symbol-ArrayType , SymbolType , SymbolGraphics-ArrayType , ds:SignatureType

Symbol data used in Schematic design is not normally provided by the component manufacturer. However, its provision can enhance the efficiency of utilizing that Part within a design process. There are various drafting standards to which a symbol can be generated to, such as ANSI, IEEE or IEC Drafting Standard.

Throughout this section as defined by the path [PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array](#), all dimensions and xy coordinates refers to grid spacing as opposed to any dimensional units of measure. This is because, customers defining their schematic grid spacing to be a multiple of a metric unit versus a multiple of an imperial unit (i.e., 2.5mm versus 100 mil). The grid spacing is defined in the software tool that absorbs this *Symbol-Array*

section of the PartModel xml file.

4.8.1.1. Symbol Graphics – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array
diagram	<pre> classDiagram class SymbolGraphics-Array { * --> AlternativeSymbolGraphics } class AlternativeSymbolGraphics { 1..∞ --> SymbolGraphics-ArrayType } class SymbolGraphics-ArrayType { Fracture-Array PartSymbolVersionType PartReferenceDesignationType Fracture-ArrayType } class AlternativeSymbolGraphicsType { PartSymbolVersionType PartReferenceDesignationType Fracture-ArrayType } </pre>
type	SymbolGraphics-ArrayType, AlternativeSymbolGraphicsType, PartSymbolVersionType, PartReferenceDesignationType, Fracture-ArrayType.

The *SymbolGraphics-Array* captures the necessary data required by a software tool to generate a graphical symbol for an electronic part. The *AlternativeSymbolGraphics* is unbounded because there are several different graphical representations that can be used to represent the same part as shown in the diagram here.

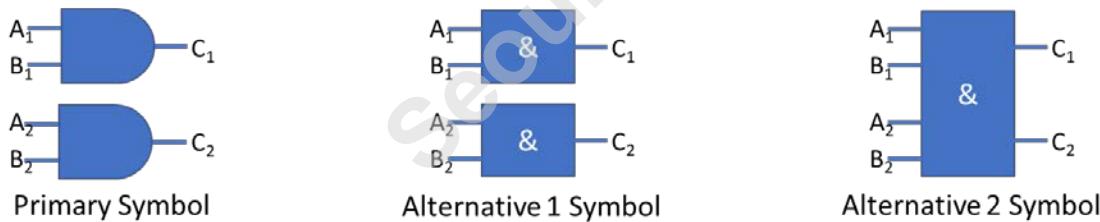


Figure 45 – Alternative Symbols

The data is grouped into the following sections:

- *PartSymbolVersion*
- *PartReferenceDesignation*, and
- *Fracture-Array*

4.8.1.1.1. Part Symbol Version

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/PartSymbolVersion
diagram	<pre> classDiagram class PartSymbolVersion { <<PartSymbolVersionType>> } class PartSymbolVersionType { <<VersionNumber>> type xs:string <<Timestamp-of-Change>> type xs:dateTime <<ChangeDescription>> type xs:string <<Reason-for-Change>> type xs:string } PartSymbolVersion "2" --> "1..1" PartSymbolVersionType </pre> <p>The diagram illustrates the UML class PartSymbolVersion, which is defined as PartSymbolVersionType. The class PartSymbolVersionType contains four attributes: VersionNumber (type xs:string), Timestamp-of-Change (type xs:dateTime), ChangeDescription (type xs:string), and Reason-for-Change (type xs:string). A multiplicity of 2 is associated with PartSymbolVersion, indicating it can have multiple instances of PartSymbolVersionType.</p>
type	PartSymbolVersionType .

PartSymbolVersion captures the version number of the symbol, timestamp of the latest changes, change description, and reason for change.

4.8.1.1.2. Part Reference Designation

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/PartReferenceDesignation																																																			
diagram	<pre> classDiagram class SymbolReferenceDesignationType { <<SymbolReferenceDesignationType>> <<ANSI-Standard>> : type Symbol-ANSI-RefDesType <<IEC-Standard>> : type Symbol-IEC-RefDesType <<Other>> : type xs:string } class PartReferenceDesignation { <<PartReferenceDesignation>> <<SymbolReferenceDesignationT...>> } PartReferenceDesignation --> SymbolReferenceDesignationType </pre>																																																			
type	SymbolReferenceDesignationType , Symbol-ANSI-RefDesType , Symbol-IEC-RefDesType .																																																			
list of enumerate values	<p>ANSI-Standard</p> <table border="1"> <tr><td>1. A</td><td>2. AR</td><td>3. AT</td></tr> <tr><td>4. BT</td><td>5. C</td><td>6. CB</td></tr> <tr><td>7. CR</td><td>8. D</td><td>9. DC</td></tr> <tr><td>10. DS</td><td>11. E</td><td>12. F</td></tr> <tr><td>13. FL</td><td>14. G</td><td>15. H</td></tr> <tr><td>16. J</td><td>17. K</td><td>18. L</td></tr> <tr><td>19. LS</td><td>20. MK</td><td>21. MT</td></tr> <tr><td>22. P</td><td>23. PS</td><td>24. Q</td></tr> <tr><td>25. R</td><td>26. RE</td><td>27. RT</td></tr> <tr><td>28. RV</td><td>29. S</td><td>30. T</td></tr> <tr><td>31. TC</td><td>32. TR</td><td>33. U</td></tr> <tr><td>34. V</td><td>35. VR</td><td>36. Y</td></tr> </table> <p>IEC-Standard</p> <table border="1"> <tr><td>1. B</td><td>2. G</td><td>3. C</td></tr> <tr><td>4. Q</td><td>5. F</td><td>6. V</td></tr> <tr><td>7. R</td><td>8. Z</td><td>9. L</td></tr> <tr><td>10. A</td><td>11. U</td><td>12. K</td></tr> <tr><td>13. S</td><td>14. T</td><td></td></tr> </table>	1. A	2. AR	3. AT	4. BT	5. C	6. CB	7. CR	8. D	9. DC	10. DS	11. E	12. F	13. FL	14. G	15. H	16. J	17. K	18. L	19. LS	20. MK	21. MT	22. P	23. PS	24. Q	25. R	26. RE	27. RT	28. RV	29. S	30. T	31. TC	32. TR	33. U	34. V	35. VR	36. Y	1. B	2. G	3. C	4. Q	5. F	6. V	7. R	8. Z	9. L	10. A	11. U	12. K	13. S	14. T	
1. A	2. AR	3. AT																																																		
4. BT	5. C	6. CB																																																		
7. CR	8. D	9. DC																																																		
10. DS	11. E	12. F																																																		
13. FL	14. G	15. H																																																		
16. J	17. K	18. L																																																		
19. LS	20. MK	21. MT																																																		
22. P	23. PS	24. Q																																																		
25. R	26. RE	27. RT																																																		
28. RV	29. S	30. T																																																		
31. TC	32. TR	33. U																																																		
34. V	35. VR	36. Y																																																		
1. B	2. G	3. C																																																		
4. Q	5. F	6. V																																																		
7. R	8. Z	9. L																																																		
10. A	11. U	12. K																																																		
13. S	14. T																																																			
list of enumerate values																																																				

PartReferenceDesignation captures the reference designator based on the ANSI standard or the IEC standard. If the reference designator is not defined by either of those standards, *Other* reference designator may be defined. If the symbol is made up of multiple fractures, and each *FractureReferenceDesignation* is the same, then that value can roll up to be the same for the *PartReferenceDesignation*.

4.8.1.1.3. Fracture – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array
diagram	<pre> classDiagram class FractureArrayType { <<NumberOffractures>> default 1 type xs:integer } class FractureArrayType { <<Fracture-Array>> type FractureArrayType } class FractureType { <<FractureType>> class ID { type xs:string } class SymbolName { type SymbolNameType } class FunctionSymbolVersion { type FunctionSymbolVersionType } class FunctionReferenceDesignation { type SymbolReferenceDesignationType } class GraphicalRepresentation { type SymbolFractureGraphicalRepresentationType } } class FractureType { <<Fracture>> type FractureType } Fracture-Array "1..1" --> Fracture Fracture "1..infinity" --> FractureType </pre>
type	Fracture-ArrayType , FractureType , SymbolNameType , FunctionSymbolVersionType , SymbolReferenceDesignationType , SymbolFractureGraphicalRepresentationType .

Fracture-Array contains fractures of symbols. When there are many terminals on a Part, it might be more practical to split the symbol into several Fractures. Terminal are then typically assigned to a *Fracture*, based on some logical organization of the terminals. The fracture name is a string that can be appended onto the symbol name or can remain as a standalone name. The list of terminals on all the fractures is the complete list of terminals for the part.

4.8.1.1.3.1. Symbol Name

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/SymbolName
diagram	<pre> classDiagram class SymbolNameType { <<SymbolNameType>> class SymbolName { type xs:string } class Function { type xs:string } } class SymbolNameType { <<SymbolName>> type SymbolNameType } SymbolNameType "1..1" --> Function </pre>
type	SymbolNameType .

4.8.1.1.3.1.1. Function Symbol Version

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/FunctionSymbolVersion
diagram	<pre> classDiagram class FunctionSymbolVersion { <<FunctionSymbolVersion>> <<FunctionSymbolVersionType>> } class FunctionSymbolVersionType { <<FunctionSymbolVersionType>> VersionNumber : xs:string Timestamp-of-Change : xs:dateTime ChangeDescription : xs:string Reason-for-Change : xs:dateTime } FunctionSymbolVersion "1..*" --> "1..*" FunctionSymbolVersionType </pre>
type	FunctionSymbolVersionType .

Each Fracture can have its own symbol. This structure provides the ability to track the version of the symbol, the date of change, the description and the reason for the change, in their respective elements.

4.8.1.1.3.2. Function Reference Designation

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/FunctionReferenceDesignation
diagram	<pre> classDiagram class FunctionReferenceDesignation { <<FunctionReferenceDesignation>> <<SymbolReferenceDesignationType>> } class SymbolReferenceDesignationType { <<SymbolReferenceDesignationType>> ANSI-Standard : Symbol-ANSI-RefDesType IEC-Standard : Symbol-IEC-RefDesType Other : xs:string } FunctionReferenceDesignation "1..*" --> "1..*" SymbolReferenceDesignationType </pre>
type	SymbolReferenceDesignationType .

The [FunctionReferenceDesignation](#) follows the same concept as the [PartReferenceDesignation](#). When compiling the schematic to be forward annotated with the layout, the [PartReferenceDesignation](#) will over-write the [FunctionReferenceDesignation](#).

4.8.1.1.3.3. Graphical Representation

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation
diagram	<pre> classDiagram class GraphicalRepresentation { <<SymbolFractureGraphicalRepresentationType>> } class Body { <<FractureSymbolBodyType>> } class GraphicalFormat { <<GraphicalFormat-ArrayType>> } class Attribute { <<SymbolAttribute-ArrayType>> } class TextFormat { <<TextFormat-ArrayType>> } class Terminal { <<TerminalGraphicalRepresentation-ArrayType>> } GraphicalRepresentation "1..∞" *-- "*" Body GraphicalRepresentation "1..∞" *-- "*" GraphicalFormat GraphicalRepresentation "1..∞" *-- "*" Attribute GraphicalRepresentation "1..∞" *-- "*" TextFormat GraphicalRepresentation "1..∞" *-- "*" Terminal </pre>
type	SymbolFractureGraphicalRepresentationType , FractureSymbolBodyType , GraphicalFormat-ArrayType , SymbolAttribute-ArrayType , TextFormat-ArrayType , TerminalGraphicalRepresentation-ArrayType .

The [GraphicalRepresentation](#) is principally made up of 3 sections, namely the [Body](#) structure, the [Attribute-Array](#) structure that is applied to the Part, and the [Terminal-Array](#), with two supporting structures called [GraphicalFormat-Array](#) and [TextFormat-Array](#).

4.8.1.1.3.3.1. Body

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body			
diagram	<pre> classDiagram class Body { type FractureSymbolBodyType } class Shape-Array { type Shape-ArrayType } class Shape { type ShapeType } class SVG-Shape { type SVG-ShapeType } class Image { type ImageType } class ShapeOrder { type ShapeOrderType default Back-to-Front } class ShapeText { type SymbolBodyAnnotationType } Body "3" -- "3" Shape-Array Shape-Array "*" -- "*" Shape Shape-Array "*" -- "*" SVG-Shape Shape-Array "*" -- "*" Image Shape-Array "*" -- "*" ShapeOrder Shape-Array "*" -- "*" ShapeText </pre>			
type	FractureSymbolBodyType, Shape-ArrayType, ShapeType, SVG-ShapeType, ImageType, SymbolBodyAnnotationType.			
list of enumerate values	<p>ShapeOrder</p> <table border="1"> <tr> <td>1. Back-to-Front</td> <td>2. Front-to-Back</td> <td></td> </tr> </table>	1. Back-to-Front	2. Front-to-Back	
1. Back-to-Front	2. Front-to-Back			

The Body shape can be constructed from a series of *Shapes*, a single *SVG-Shape*, or a series of *Images*. The *ShapeOrder* is set for all the shapes as “*Back-to-Front*” as default but can be changed to “*Front-to-Back*”. The order of the sequence is captured within the relevant shape array. The *ShapeText* captures the structure for any text that is assigned to the shape.

4.8.1.1.3.3.1.1. Shape

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape
diagram	<p>The diagram shows a UML Class Diagram for the ShapeType. It includes the following classes and associations:</p> <ul style="list-style-type: none"> ShapeType (superclass) with associations to Vertex-Array, Line, Arc, EllipticalArc, and Primitive-Shape. Shape (sub-class) with a multiplicity of 1..∞ and an association to ShapeType. Vertex-Array, Line, Arc, EllipticalArc, and Primitive-Shape each have their own type definitions (Vertex-ArrayType, LineType, ArcType, EllipticalArcType, and Primitive-ShapeType). GraphicalFormatID and GraphicalFormat are nested within Shape, with GraphicalFormatID having type xs:string and GraphicalFormat having type GraphicalFormatType. ShapeOrderSequence is also associated with Shape.
type	ShapeType , Vertex-ArrayType , LineType , ArcType , EllipticalArcType , Primitive-ShapeType , GraphicalFormatType .

The Shape can be created from a choice of Vertices, Lines, Arcs, Elliptical Arcs, or primitive shapes.

GraphicalFormatID serves as a reference ID for the **GraphicalFormat**, that is defined under the **GraphicalFormat-Array/GraphicalFormats** where a set of graphical formats can be defined that can be standardized across multiple **Shape** entries. The addition of the **GraphicalFormat** under **Shape** enables unique modification of a references Graphical Format for applying to this specific **Shape** instance.

The **ShapeOrderSequence** can be used to define the order of shapes for a complex symbol that may consist of various shapes. It works in tandem with the **ShapeOrder** that is defined under **Body/Shape-Array**.

4.8.1.1.3.3.1.1.1. Vertex – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Vertex-Array
diagram	<pre> classDiagram class Vertex-ArrayType { <<Vertex>> <<Edge>> } class Vertex { <<VertexType>> } class Edge { <<EdgeVertexControlType>> } Vertex-ArrayType "1..∞" --> "1..∞" Vertex Vertex-ArrayType "1..∞" --> "1..∞" Edge </pre>
type	Vertex-ArrayType, VertexType, EdgeVertexControlType.

4.8.1.1.3.3.1.1.2. Vertex

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Vertex-Array/Vertex
diagram	<pre> classDiagram class VertexType { <<x>> <<xs:decimal>> <<y>> <<xs:decimal>> } class ArcVertexType { <<Radius>> <<xs:decimal>> } class JEP30-D10:PointXYType { <<x>> <<xs:decimal>> <<y>> <<xs:decimal>> } class SplineType { <<ControlPoint>> <<JEP30-D10:PointXYType>> } VertexType "1..∞" --> "1..∞" Vertex VertexType "1..∞" --> "1..∞" Arc Arc "1..∞" --> "1..∞" Center Arc "1..∞" --> "1..∞" Radius Center "1..∞" --> "1..∞" JEP30-D10:PointXYType ControlPoint "1..∞" --> "1..∞" Spline </pre>
type	VertexType, ArcVertexType, SplineType, JEP30-D10:PointXYType.

4.8.1.1.3.3.1.1.3. Edge

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Vertex-Array/Edge
diagram	<pre> classDiagram class Edge { <<EdgeVertexControlType>> } class EdgeVertexControlType class Line { <<LineVertexControlType>> } class LineVertexControlType class Triangle { <<TriangleVertexControlType>> } class TriangleVertexControlType class Lines { <<JEP30-D10:EmptyType>> } class Strip { <<JEP30-D10:EmptyType>> } class Loop { <<JEP30-D10:EmptyType>> } class Triangles { <<JEP30-D10:EmptyType>> } class Strip { <<JEP30-D10:EmptyType>> } class Fan { <<JEP30-D10:EmptyType>> } Edge < -- EdgeVertexControlType Edge < -- Line Edge < -- Triangle Line < -- LineVertexControlType LineVertexControlType < -- Lines LineVertexControlType < -- Strip LineVertexControlType < -- Loop Triangle < -- TriangleVertexControlType TriangleVertexControlType < -- Triangles TriangleVertexControlType < -- Strip TriangleVertexControlType < -- Fan </pre> <p>The diagram illustrates the inheritance path for the Edge class. It starts with Edge (type: EdgeVertexControlType) which can be either a Line or a Triangle. If it's a Line, it further inherits from LineVertexControlType, which defines Lines, Strip, and Loop. If it's a Triangle, it further inherits from TriangleVertexControlType, which defines Triangles, Strip, and Fan. A note indicates that if the Edge element is omitted, the default should be assumed to be Line/Strip.</p>
type	EdgeVertexControlType , LineVertexControlType , TriangleVertexControlType , JEP30-D10:EmptyType .

The default is assumed to be **Line/Strip** if the **Edge** element is omitted. The **Edge** structure governs how each of the vertices in the unbounded **Vertex** element is to be processed.

Line/Lines represent pairs of vertices interpreted as individual line segments. Vertices 1 and 2 make one line segment whereas vertices 3 and 4 makes the 2nd line segment. Vertices 2 and 3 are not connected with a line segment.

Line/Strip represent a series of connected line segments. Vertices 1 and 2 make one line segment whereas vertices 2 and 3 makes the 2nd line segment, and vertices 3 and 4 makes the 3rd line segment.

Lines/Loop is the same as strip, with a segment added between last and first vertices. From the previous example, vertices 4 and 1 makes the final line segment, creation a closed loop.

Triangle/Triangles represent each triple set of vertices to be interpreted as an individual triangle. Vertices 1, 2, and 3 make one traingle whereas vertices 4, 5, and 6 makes the 2nd triangle.

4.8.1.1.3.3.1.1.3. Edge (cont'd)

Triangle/Strip represent a linking strip of triangles, where vertices 1, 2, and 3 make up the first triangle and vertices 2, 3 and 4 make up the next triangle.

Triangle/Fan represent a linking strip of triangles the same as the strip where the last triangle is defined by vertices n-1, n and 1. This in effect creates a fan of triangles

4.8.1.1.3.3.1.1.4. Line

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Line
diagram	<pre> classDiagram class Line { <<LineType>> <<Line>> type LineType } class StartingPoint { <<JEP30-D10:PointXYType>> type JEP30-D10:PointXYType } class EndPoint { <<JEP30-D10:PointXYType>> type JEP30-D10:PointXYType } Line "1" -- "2" StartingPoint Line "1" -- "2" EndPoint </pre>
type	LineType, JEP30-D10:PointXYType.

4.8.1.1.3.3.1.1.5. Arc

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Arc			
diagram	<pre> classDiagram class Arc { <<ArcType>> } class StartingPoint { <<JEP30-D10:PointXYType>> } class EndPoint { <<JEP30-D10:PointXYType>> } class InnerPoint { <<JEP30-D10:PointXYType>> } class CenterPoint { <<JEP30-D10:PointXYType>> } class Radius { <<xs:decimal>> } class Center { <<JEP30-D10:PointXYType>> } class StartAngle { <<xs:integer>> } class AngleToFill { <<xs:integer>> } class Direction { <<ArcDirectionType>> } Arc "1" -- "*" StartingPoint : Arc "1" -- "*" EndPoint : Arc "1" -- "*" InnerPoint : Arc "1" -- "1" CenterPoint : Arc "1" -- "1" Radius : Arc "1" -- "1" Center : Arc "1" -- "1" StartAngle : Arc "1" -- "1" AngleToFill : Arc "1" --> "1" Direction </pre>			
type	ArcType , ArcDirectionType , JEP30-D10:PointXYType .			
list of enumerate values	<p>Direction</p> <table border="1"> <tr> <td>1. Clockwise</td> <td>2. Anti-clockwise</td> <td></td> </tr> </table>	1. Clockwise	2. Anti-clockwise	
1. Clockwise	2. Anti-clockwise			

4.8.1.1.3.3.1.1.6. Elliptical Arc

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/EllipticalArc
diagram	<pre> classDiagram class EllipticalArcType { StartingPoint EndPoint InnerPoint CenterPoint Direction X-Radius Y-Radius Center StartAngle AngleToFill } class EllipticalArc { <> EllipticalArcType } EllipticalArc < -- EllipticalArcType </pre> <p>The diagram illustrates the structure of the <code>EllipticalArcType</code> class. It contains the following attributes:</p> <ul style="list-style-type: none"> <code>StartingPoint</code>: type <code>JEP30-D10:PointXYType</code> <code>EndPoint</code>: type <code>JEP30-D10:PointXYType</code> <code>InnerPoint</code>: type <code>JEP30-D10:PointXYType</code> <code>CenterPoint</code>: type <code>JEP30-D10:PointXYType</code> <code>Direction</code>: type <code>ArcDirectionType</code> <code>X-Radius</code>: type <code>xs:decimal</code> <code>Y-Radius</code>: type <code>xs:decimal</code> <code>Center</code>: type <code>JEP30-D10:PointXYType</code> <code>StartAngle</code>: type <code>xs:integer</code> <code>AngleToFill</code>: type <code>xs:integer</code> <p>The <code>EllipticalArc</code> class is shown to inherit from <code>EllipticalArcType</code>.</p>
type	<code>EllipticalArcType, ArcDirectionType, JEP30-D10:PointXYType.</code>

4.8.1.1.3.3.1.1.7. Primitive-Shape

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Primitive-Shape
diagram	<pre> classDiagram class Primitive-ShapeType { <<Primative-ShapeType>> <<RectangleShapeType>> <<CircleShapeType>> <<EllipseShapeType>> <<ShapeOriginType>> <<ShapeTransformationType>> } class Rectangle { <<RectangleShapeType>> } class Circle { <<CircleShapeType>> } class Ellipse { <<EllipseShapeType>> } class DimensionX { <<xs:decimal>> } class DimensionY { <<xs:decimal>> } class Radius { <<xs:decimal>> } class X-Radius { <<xs:decimal>> } class Y-Radius { <<xs:decimal>> } Primitive-ShapeType < -- Rectangle Primitive-ShapeType < -- Circle Primitive-ShapeType < -- Ellipse Primitive-ShapeType < -- ShapeOriginType Primitive-ShapeType < -- ShapeTransformationType Rectangle < -- RectangleShapeType Circle < -- CircleShapeType Ellipse < -- EllipseShapeType RectangleShapeType < -- DimensionX RectangleShapeType < -- DimensionY CircleShapeType < -- Radius EllipseShapeType < -- X-Radius EllipseShapeType < -- Y-Radius </pre> <p>The diagram illustrates the UML Class Diagram for the <code>Primative-ShapeType</code>. It defines several classes and their associations:</p> <ul style="list-style-type: none"> <code>Primative-ShapeType</code> is the base class, which has four subclasses: <code>Rectangle</code>, <code>Circle</code>, <code>Ellipse</code>, and <code>ShapeOriginType</code>. <code>ShapeOriginType</code> and <code>ShapeTransformationType</code> are also associated with <code>Primative-ShapeType</code>. <code>Rectangle</code> is associated with <code>RectangleShapeType</code>. <code>Circle</code> is associated with <code>CircleShapeType</code>. <code>Ellipse</code> is associated with <code>EllipseShapeType</code>. <code>RectangleShapeType</code> has two attributes: <code>DimensionX</code> and <code>DimensionY</code>, both of type <code>xs:decimal</code>. <code>CircleShapeType</code> has one attribute: <code>Radius</code>, of type <code>xs:decimal</code>. <code>EllipseShapeType</code> has two attributes: <code>X-Radius</code> and <code>Y-Radius</code>, both of type <code>xs:decimal</code>.
type	<code>Primative-ShapeType</code> , <code>RectangleShapeType</code> , <code>CircleShapeType</code> , <code>EllipseShapeType</code> , <code>ShapeOriginType</code> , <code>ShapeTransformationType</code> .

4.8.1.1.3.3.1.1.8. Shape Origin

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Primitive-Shape/ShapeOrigin
diagram	<pre> classDiagram class ShapeOriginType class Horizontal class Vertical class Left class Center class Right class Top class Bottom class Coordinate class JEP30-D10:PointXYType ShapeOriginType "1" -- "*" Horizontal ShapeOriginType "1" -- "*" Vertical Horizontal "1" -- "*" Left Horizontal "1" -- "*" Center Horizontal "1" -- "*" Right Vertical "1" -- "*" Top Vertical "1" -- "*" Center Vertical "1" -- "*" Bottom Coordinate "1" -- "*" JEP30-D10:PointXYType JEP30-D10:PointXYType "1" -- "*" x JEP30-D10:PointXYType "1" -- "*" y </pre> <p>Default should be assumed to be Horizontal and Vertical Center if this section is omitted.</p>
type	ShapeOriginType , HorizontalShapeOriginType , VerticalShapeOriginType , JEP30-D10:PointXYType , JEP30-D10:EmptyType .

If the *ShapeOrigin* is omitted, the default should be assumed to be *Horizontal/Center* and *Vertical/Center*.

4.8.1.1.3.3.1.1.9. Transformation

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Primitive-Shape/Transformation
diagram	<pre> classDiagram class ShapeTransformationType { <<ShapeTransformation>> } class TranslateShapeType { <<Translate>> } class ScaleShapeType { <<Scale>> } class RotateShapeType { <<Rotate>> } class TransformationType { <<Transformation>> } ShapeTransformationType < -- TranslateShapeType ShapeTransformationType < -- ScaleShapeType ShapeTransformationType < -- RotateShapeType ShapeTransformationType < -- TransformationType TranslateShapeType < -- Translate Translate < -- To To < -- JEP30-D10:PointXYType JEP30-D10:PointXYType < -- x JEP30-D10:PointXYType < -- y ScaleShapeType < -- Scale Scale < -- About About < -- Factors Factors < -- ScaleFactorsType ScaleFactorsType < -- y ScaleFactorsType < -- x RotateShapeType < -- Rotate Rotate < -- About About < -- Angle TransformationType < -- Transformation Transformation < -- Transformation </pre> <p>The diagram illustrates the UML class structure for ShapeTransformationType, which is a base class for four specific transformation types: TranslateShapeType, ScaleShapeType, RotateShapeType, and TransformationType. The TranslateShapeType includes a Translate component with a To path leading to a PointXYType object containing x and y coordinates. The ScaleShapeType includes a Scale component with an About path leading to a Factors object, which is associated with a ScaleFactorsType object containing y and x scale factors. The RotateShapeType includes a Rotate component with an About path leading to an Angle object. The TransformationType is shown as a self-referencing association.</p>
type	ShapeTransformationType , TranslateShapeType , JEP30-D10:PointXYType , ScaleShapeType , ScaleAboutType , ScaleFactorsType , RotateShapeType , RotateShapeAboutType , JEP30-D10:EmptyType .

[Transformation](#) enables additional modification to be performed to a primitive shape. The [Translate](#) moves the shape from the location currently defined by the [ShapeOrigin](#) to the new x,y coordinates. The [Scale](#) enables the re-sizing of the shape and provides options for the scaling to be performed around the [Origin](#), [ShapeCenter](#) or any set of [Coordinates](#). The [Rotate](#) enables the rotation to be performed around a similar set of points such as the [Origin](#), [ShapeCenter](#) or any set of [Coordinates](#). The [Angle](#) is specified in degrees whereas a positive number is considered as anti-clockwise and a negative number is considered clockwise (eg., if 3 o'clock is 0 degrees, then 12 o'clock is 90 degrees and 6 o'clock is 270 degrees or -90 degrees).

4.8.1.1.3.3.1.2. SVG-Shape

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/SVG-Shape									
diagram	<pre> classDiagram class SVG-ShapeType { <![CDATA[Insert SVG File here]]> BoundingBoxType FitStyleType JEP30-D10:PointXYType } class SVG-ShapeData { type xs:string } class BoundingBox { type BoundingBoxType } class DimensionX { type xs:decimal } class DimensionY { type xs:decimal } class Origin-of-SVG-Shape { type SymbolTerminalLabelOriginType default 5 - Center } class SVG-ShapeLocation { type JEP30-D10:PointXYType } class X { type xs:decimal } class Y { type xs:decimal } SVG-ShapeData "1" -- "2" : BoundingBox BoundingBox "*" -- "3" : DimensionX BoundingBox "*" -- "3" : DimensionY BoundingBox "*" -- "4" : Origin-of-SVG-Shape BoundingBox "*" -- "5" : SVG-ShapeLocation SVG-ShapeLocation "*" -- "6" : X SVG-ShapeLocation "*" -- "6" : Y </pre> <p>The diagram illustrates the UML Class Diagram for the SVG-ShapeType. It consists of several classes and their associations:</p> <ul style="list-style-type: none"> SVG-ShapeType is the main container class. SVG-ShapeData is associated with SVG-ShapeType via a multiplicity of 1. BoundingBox is associated with SVG-ShapeType via a multiplicity of *. DimensionX and DimensionY are associated with BoundingBox via a multiplicity of *. Origin-of-SVG-Shape is associated with BoundingBox via a multiplicity of *. SVG-ShapeLocation is associated with BoundingBox via a multiplicity of *. X and Y are associated with SVG-ShapeLocation via a multiplicity of *. 									
type	SVG-ShapeType , BoundingBoxType , FitStyleType , SymbolTerminalLabelOriginType , JEP30-D10:PointXYType .									
list of enumerate values	FitStyleType <table border="1"> <tr> <td>1. Scale to Fit</td> <td>2. Stretch to Fit</td> <td>3. Crop</td> </tr> </table>	1. Scale to Fit	2. Stretch to Fit	3. Crop						
1. Scale to Fit	2. Stretch to Fit	3. Crop								
list of enumerate values	SymbolTerminalLabelOriginType <table border="1"> <tr> <td>1. Northwest</td> <td>2. Leftcenter</td> <td>3. Southwest</td> </tr> <tr> <td>4. Uppercenter</td> <td>5. Center</td> <td>6. Backcenter</td> </tr> <tr> <td>7. Northeast</td> <td>8. Rightcenter</td> <td>9. Southeast</td> </tr> </table>	1. Northwest	2. Leftcenter	3. Southwest	4. Uppercenter	5. Center	6. Backcenter	7. Northeast	8. Rightcenter	9. Southeast
1. Northwest	2. Leftcenter	3. Southwest								
4. Uppercenter	5. Center	6. Backcenter								
7. Northeast	8. Rightcenter	9. Southeast								

4.8.1.1.3.3.1.3. Image

path	<code>PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Image</code>									
diagram	<pre> classDiagram class ImageType { ImageData BoundingBox OriginOfImage ImageLocation } class ImageData { type ImageDataType Data GraphicalFileFormat } class Data { type xs.base64Binary } class GraphicalFileFormat { type GraphicalFileType } class BoundingBox { type BoundingBoxType } class OriginOfImage { type SymbolTerminalLabelOriginType default 5 - Center } class ImageLocation { type JEP30-D10:PointXYType } Image < -- ImageType Image < -- ImageData </pre> <p>The diagram illustrates the structure of the <code>ImageType</code> element. It contains four main components: <code>ImageData</code>, <code>BoundingBox</code>, <code>Origin-of-Image</code>, and <code>ImageLocation</code>. The <code>ImageData</code> component is associated with <code>Data</code> (type <code>xs.base64Binary</code>) and <code>GraphicalFileFormat</code> (type <code>GraphicalFileType</code>). The <code>Image</code> element is associated with <code>ImageType</code> via a multiplicity of <code>1..∞</code>.</p>									
type	<code>ImageType, ImageDataType , GraphicalFileType , BoundingBoxType, FitStyleType, SymbolTerminalLabelOriginType, JEP30-D10:PointXYType.</code>									
list of enumerate values	<table border="1"> <tr> <td><code>GraphicalFileType</code></td> <td></td> <td></td> </tr> <tr> <td>1. <code>.jpg</code></td> <td>2. <code>.png</code></td> <td>3. <code>.gif</code></td> </tr> <tr> <td>4. <code>.bmp</code></td> <td></td> <td></td> </tr> </table>	<code>GraphicalFileType</code>			1. <code>.jpg</code>	2. <code>.png</code>	3. <code>.gif</code>	4. <code>.bmp</code>		
<code>GraphicalFileType</code>										
1. <code>.jpg</code>	2. <code>.png</code>	3. <code>.gif</code>								
4. <code>.bmp</code>										

When inserting an `Image` into the xml file, the information should be processed as a string and not as xml. Therefore, insert the image file name within the string `<![CDATA[Insert Image (.png, .jpg, .gif, .bmp) File here]]>`

4.8.1.1.3.3.1.4. Shape Text

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/ShapeText												
diagram	<pre> classDiagram class ShapeText { <<SymbolBodyAnnotationType>> <<0..>> } class SymbolBodyAnnotationType { <<Detail>> <<TextFormatID>> <<TextFormat>> <<Orientation>> <<Location>> <<Origin>> } ShapeText "0..>" -- "1" SymbolBodyAnnotationType ShapeText "*" -- "1" TextFormatID ShapeText "*" -- "1" TextFormat ShapeText "*" -- "1" Orientation ShapeText "*" -- "1" Location ShapeText "*" -- "1" Origin </pre>												
type	SymbolBodyAnnotationType, TextLabelFormatType, SymbolTextOrientationType, JEP30-D10:PointXYType, SymbolTerminalLabelOriginType.												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="3">SymbolTextOrientationType</th> </tr> </thead> <tbody> <tr> <td>0 - Normal orientation</td> <td>1 - 90 Degree rotation</td> <td>2 - 180 Degree rotation</td> </tr> <tr> <td>3 - 270 Degree rotation</td> <td></td> <td></td> </tr> </tbody> </table>	SymbolTextOrientationType			0 - Normal orientation	1 - 90 Degree rotation	2 - 180 Degree rotation	3 - 270 Degree rotation					
SymbolTextOrientationType													
0 - Normal orientation	1 - 90 Degree rotation	2 - 180 Degree rotation											
3 - 270 Degree rotation													
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="3">SymbolTerminalLabelOriginType</th> </tr> </thead> <tbody> <tr> <td>1. Northwest</td> <td>2. Leftcenter</td> <td>3. Southwest</td> </tr> <tr> <td>4. Uppercenter</td> <td>5. Center</td> <td>6. Backcenter</td> </tr> <tr> <td>7. Northeast</td> <td>8. Rightcenter</td> <td>9. Southeast</td> </tr> </tbody> </table>	SymbolTerminalLabelOriginType			1. Northwest	2. Leftcenter	3. Southwest	4. Uppercenter	5. Center	6. Backcenter	7. Northeast	8. Rightcenter	9. Southeast
SymbolTerminalLabelOriginType													
1. Northwest	2. Leftcenter	3. Southwest											
4. Uppercenter	5. Center	6. Backcenter											
7. Northeast	8. Rightcenter	9. Southeast											

TextFormatID serves as a reference ID for the *TextFormat*, that is defined under the *GraphicalRepresentation/TextFormat-Array/TextFormats* where a set of text formats can be defined that can be standardized in the *Body/Shape-Array* entry. The addition of the *TextFormat* under *ShapeText* enables unique modification of a references Text Format for applying to this specific *ShapeText* instance. When the *TextFormatID* is specified, then all the elements under *GraphicalRepresentation/TextFormat-Array/TextFormats/TextFormat* will be applied to the *ShapeText*, unless elements under the *ShapeText/TextFormat* are populated, in which case the latter *ShapeText/TextFormat* elements will override those elements as referenced by the *TextFormatID*.

FontStyle for *TextFormat* is restricted to any combination of *Bold*, *Italics* and *Underline*

4.8.1.1.3.3.2. Graphical Format – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/GraphicalFormat-Array
diagram	<pre> classDiagram class GraphicalFormat { type GraphicalFormatType } class GraphicalFormats { type GraphicalFormatsType } class GraphicalFormatArray { type GraphicalFormatArrayType } class DefaultGraphicalFormat { type DefaultGraphicalFormatType } GraphicalFormatArray "1..>" GraphicalFormats GraphicalFormats "1..>" GraphicalFormat GraphicalFormat "1..>" GraphicalFormats GraphicalFormatArray --> DefaultGraphicalFormat </pre>
type	GraphicalFormat-ArrayType , GraphicalFormatsType , GraphicalFormatType , DefaultGraphicalFormatType .

GraphicalFormat-Array contains a structure that enables the supplier to provide a series of additional graphical formats including a default graphical format, that can be referenced by the shapes under the *Body/Shape-Array*.

4.8.1.1.3.3.2.1. Graphical Format

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/GraphicalFormat-Array/GraphicalFormats/GraphicalFormat			
diagram	<pre> classDiagram class GraphicalFormatType { <<GraphicalFormat>> <<GraphicalFormatType>> <<StrokeWidth>> <<StrokeWidthType>> <<minIncl/maxIncl>> 1 <<StrokeOpacity>> <<StrokeOpacityType>> <<minIncl/maxIncl>> 0 100 <<StrokeLineCap>> <<StrokeLineCapType>> <<StrokeDash-ArrayType>> <<StrokeDash-Array>> <<type>> StrokeDash-ArrayType <<Dash>> <<type>> xs:integer <<minIncl/maxIncl>> 1..∞ <<ColorType>> <<Name>> <<type>> ColorNameType <<Hex>> <<pattern>> #[A-Fa-f0-9]{6} <<type>> ColorHexType <<R>> <<type>> Color-RType <<minIncl/maxIncl>> 0 255 <<G>> <<type>> Color-GType <<minIncl/maxIncl>> 0 255 <> <<type>> Color-BType <<minIncl/maxIncl>> 0 255 <<FillColor>> <<type>> ColorType <<FillOpacity>> <<type>> FillOpacityType <<minIncl/maxIncl>> 0 100 } </pre>			
type	GraphicalFormatType, StrokeWidthType, StrokeOpacityType, StrokeLineCapType, StrokeDash-ArrayType, ColorType, ColorNameType, ColorHexType, Color-RType, Color-GType, Color-BType, FillOpacityType.			
list of enumerate values	<p>StrokeLineCapType</p> <table border="1"> <tr> <td>1. Butt</td> <td>2. Round</td> <td>3. Square</td> </tr> </table>	1. Butt	2. Round	3. Square
1. Butt	2. Round	3. Square		

[ColorNameType](#) as defined in the [StrokeColor/Name](#) element list the names of the Scalable Vector Graphics (SVG) Colors as defined by the Scalable Vector Graphics (SVG) Specification.

[ColorHexType](#) as defined in the [StrokeColor/Hex](#) element uses a regular expression pattern to recognize the six-digit hexadecimal representation of the Scalable Vector Graphics (SVG) Colors.

[Color-RType](#), [Color-GType](#), [Color-BType](#) allows the construction of all the colors from the combination of the red, green, and blue colors in the RGB Color Space. The red, green, and blue use 8 bits each, which have integer values from 0 to 255.

4.8.1.1.3.3.2.2. Default Graphical Format

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/GraphicalFormat-Array/DefaultGraphicalFormat
diagram	<pre> classDiagram class DefaultGraphicalFormatType { <<DefaultGraphicalFormat>> <<DefaultStrokeWidthType>> <<DefaultStrokeOpacityType>> <<DefaultStrokeLineCapType>> <<DefaultColorType>> <<DefaultColorNameType>> <<DefaultColorHexType>> <<DefaultColorRType>> <<DefaultColorGType>> <<DefaultColorBType>> <<DefaultFillOpacityType>> } class DefaultGraphicalFormat { <<DefaultGraphicalFormatType>> } class StrokeWidth { <<DefaultStrokeWidthType>> } class StrokeOpacity { <<DefaultStrokeOpacityType>> } class StrokeLineCap { <<DefaultStrokeLineCapType>> } class StrokeColor { <<DefaultColorType>> } class FillColor { <<DefaultColorType>> } class Name { <<DefaultColorNameType>> } class Hex { <<DefaultColorHexType>> } class R { <<DefaultColorRType>> } class G { <<DefaultColorGType>> } class B { <<DefaultColorBType>> } class FillOpacity { <<DefaultFillOpacityType>> } DefaultGraphicalFormat "1" *-- "1" StrokeWidth DefaultGraphicalFormat "1" *-- "1" StrokeColor DefaultGraphicalFormat "1" *-- "1" FillColor StrokeColor "1" *-- "1" Name StrokeColor "1" *-- "1" Hex StrokeColor "1" *-- "1" R StrokeColor "1" *-- "1" G StrokeColor "1" *-- "1" B StrokeColor "1" *-- "1" FillOpacity </pre>
type	<p>DefaultGraphicalFormatType, DefaultStrokeWidthType, DefaultStrokeOpacityType, DefaultStrokeLineCapType, DefaultColorType, DefaultColorNameType, DefaultColorHexType, DefaultColorRType, DefaultColorGType, DefaultColorBType, DefaultFillOpacityType.</p>

If no independent *GraphicalFormat* are defined, then the *DefaultGraphicalFormat* will define the default values as follows,

- a. *StrokeWidth* is defaulted to the value 1
- b. *StrokeLineCap* is defaulted to value “Butt”
- c. *StrokeColor/Name* is defaulted to value “Black”
- d. *StrokeColor/Hex* is defaulted to value “#000000”
- e. *StrokeColor/R*, *G* and *B* are defaulted to the value 0
- f. *StrokeOpacity* and *FillOpacity* are defaulted to 100

4.8.1.1.3.3.3. Attribute – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Attribute-Array				
diagram	<pre> classDiagram class SymbolAttributeArrayType { <<SymbolAttribute-ArrayType>> } class SymbolAttributeType { <<SymbolAttributeType>> } class AttributeDetailsType { <<AttributeDetailsType>> } class AttributeDetail { <<AttributeDetail>> } class Name { <<Name>> } class Value { <<Value>> } class TextFormatID { <<TextFormatID>> } class TextFormat { <<TextFormat>> } class Visibility { <<Visibility>> } class TextVariantforOneoftheViewsID { <<TextVariant-for-One-of-the-ViewsID>> } class TextVariantforOneoftheViewsArrayType { <<TextVariant-for-One-of-the-Views-Array>> } SymbolAttributeType "1..>" AttributeDetailsType AttributeDetailsType "1..>" AttributeDetail AttributeDetail "1..>" Name AttributeDetail "1..>" Value AttributeDetailsType "1..>" TextFormatID AttributeDetailsType "1..>" TextFormat AttributeDetailsType "1..>" Visibility AttributeDetailsType "1..>" TextVariantforOneoftheViewsArrayType TextVariantforOneoftheViewsArrayType "0..7" TextVariantforOneoftheViewsID </pre> <p>constraints</p>				
type	SymbolAttribute-ArrayType , SymbolAttributeType , AttributeDetailsType , TextLabelFormatType , VisibilityType , SymbolAttributeTextVariant-for-One-of-the-Views-ArrayType .				
list of enumerate values	<table border="1"> <tr> <td>Visibility</td> <td></td> </tr> <tr> <td>1. Invisible</td> <td>2. Visible</td> </tr> </table>	Visibility		1. Invisible	2. Visible
Visibility					
1. Invisible	2. Visible				

Attributes are text that can be assigned to the symbol. When symbols are rotated, then the attributes will also need to be rotated and its rotation is not the same as the rotation of the symbol. The attribute may also have to be re-located on the symbol.

The structure under the element [TextVariant-for-One-of-the-Views-Array](#) as described in the next section provides that ability to re-position the attribute text and to independently rotate the text to the rotation of the symbol.

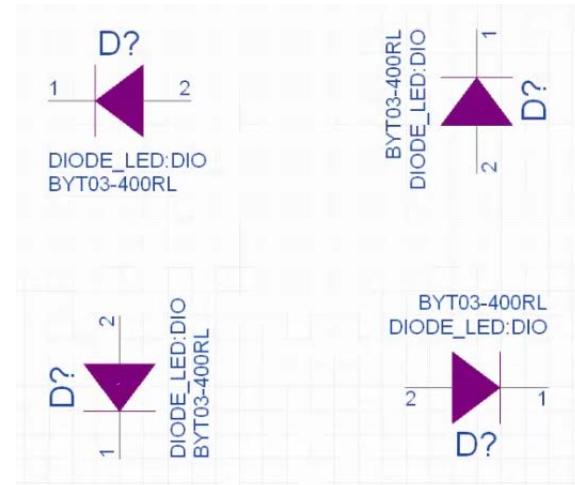


Figure 46 – Text Views for Alternative Symbol Rotation

4.8.1.1.3.3.3.1. Text Variant for One of the Views – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Attribute-Array/Attribute/TextVariant-for-One-of-the-Views-Array												
diagram	<pre> classDiagram class TextVariant-for-One-of-the-Views { <<SymbolAttributeTextVariant-for-One-of-the-Views-Type>> } class ID { <<xs:string>> } class SymbolOrientation { <<SymbolOrientationType>> } class OrientationTextVariant { <<SymbolTextOrientationType>> default 0 - Normal Rotation } class AttributeLocation { <<JEP30-D10:PointXYType>> } class OriginOfAttribute { <<SymbolTerminalLabelOriginType>> default 5 - Center } TextVariant-for-One-of-the-Views "1..8" --> ID TextVariant-for-One-of-the-Views "1..8" --> SymbolOrientation TextVariant-for-One-of-the-Views "1..8" --> OrientationTextVariant TextVariant-for-One-of-the-Views "1..8" --> AttributeLocation TextVariant-for-One-of-the-Views "1..8" --> OriginOfAttribute </pre> <p>The diagram illustrates the UML class structure for the <code>SymbolAttributeTextVariant-for-One-of-the-Views-ArrayType</code>. It features a central <code>TextVariant-for-One-of-the-Views</code> class associated with eight instances of <code>ID</code>, <code>SymbolOrientation</code>, <code>OrientationTextVariant</code>, <code>AttributeLocation</code>, and <code>OriginOfAttribute</code>. A constraint box labeled <code>constraints</code> is also present.</p>												
type	<code>SymbolAttributeTextVariant-for-One-of-the-Views-ArrayType</code> , <code>SymbolAttributeTextVariant-for-One-of-the-ViewsType</code> , <code>SymbolOrientationType</code> , <code>SymbolAttributeLocationType</code> , <code>JEP30-D10:PointXYType</code> , <code>SymbolTerminalLabelOriginType</code> .												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="3"><code>SymbolOrientationType</code></th> </tr> </thead> <tbody> <tr> <td>0 - Normal orientation</td> <td>1 - 90 Degree rotation</td> <td>2- 180 Degree rotation</td> </tr> <tr> <td>3 - 270 Degree rotation</td> <td>4 - Mirror about x axis</td> <td>5 - Mirror about x axis and 90 Degree rotation</td> </tr> <tr> <td>6 - Mirror about x axis and 180 Degree rotation</td> <td>7 - Mirror about x axis and 270 Degree rotation</td> <td></td> </tr> </tbody> </table>	<code>SymbolOrientationType</code>			0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation	3 - 270 Degree rotation	4 - Mirror about x axis	5 - Mirror about x axis and 90 Degree rotation	6 - Mirror about x axis and 180 Degree rotation	7 - Mirror about x axis and 270 Degree rotation	
<code>SymbolOrientationType</code>													
0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation											
3 - 270 Degree rotation	4 - Mirror about x axis	5 - Mirror about x axis and 90 Degree rotation											
6 - Mirror about x axis and 180 Degree rotation	7 - Mirror about x axis and 270 Degree rotation												
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="3"><code>SymbolTextOrientationType</code></th> </tr> </thead> <tbody> <tr> <td>0 - Normal orientation</td> <td>1 - 90 Degree rotation</td> <td>2- 180 Degree rotation</td> </tr> <tr> <td>3 - 270 Degree rotation</td> <td></td> <td></td> </tr> </tbody> </table>	<code>SymbolTextOrientationType</code>			0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation	3 - 270 Degree rotation					
<code>SymbolTextOrientationType</code>													
0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation											
3 - 270 Degree rotation													
list of enumerate values	<table border="1"> <thead> <tr> <th colspan="3"><code>SymbolTerminalLabelOriginType</code></th> </tr> </thead> <tbody> <tr> <td>1. Northwest</td> <td>2. Leftcenter</td> <td>3. Southwest</td> </tr> <tr> <td>4. Uppercenter</td> <td>5. Center</td> <td>6. Backcenter</td> </tr> <tr> <td>7. Northeast</td> <td>8. Rightcenter</td> <td>9. Southeast</td> </tr> </tbody> </table>	<code>SymbolTerminalLabelOriginType</code>			1. Northwest	2. Leftcenter	3. Southwest	4. Uppercenter	5. Center	6. Backcenter	7. Northeast	8. Rightcenter	9. Southeast
<code>SymbolTerminalLabelOriginType</code>													
1. Northwest	2. Leftcenter	3. Southwest											
4. Uppercenter	5. Center	6. Backcenter											
7. Northeast	8. Rightcenter	9. Southeast											

4.8.1.1.3.3.4. Text Format – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/TextFormat-Array
diagram	<pre> classDiagram class TextFormat-ArrayType { <<TextFormat-ArrayType>> <<TextFormat-ArrayType>> } class TextFormats { <<TextFormatsType>> <<TextFormat-ArrayType>> } class ID { <<ID>> type xs:string } class TextFormat { <<TextLabelFormatType>> } class DefaultTextFormat { <<DefaultTextFormatType>> } TextFormat-ArrayType "1..∞" --> TextFormats : <<TextFormat-ArrayType>> TextFormat-ArrayType --> ID : <<ID>> TextFormat-ArrayType --> TextFormat : <<TextLabelFormatType>> TextFormat-ArrayType --> DefaultTextFormat : <<DefaultTextFormatType>> </pre>
type	TextFormat-ArrayType , TextFormatsType , TextLabelFormatType , DefaultTextFormatType .

TextFormat-Array contains a structure that enables the supplier to provide a series of additional text formats including a default text format, that can be referenced by the *ShapeText* under the *Body/Shape-Array*, or the *Attribute* under the *Attribute-Array*.

4.8.1.1.3.3.4.1. Text Format

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/TextFormat-Array/TextFormats/TextFormat
diagram	<pre> classDiagram class TextFormat { <<TextFormat>> <<TextLabelFormatType>> } class TextLabelFormatType { <<TextLabelFormatType>> TextSize Color } class SymbolFontStyleType { <<SymbolFontStyleType>> Bold Italic Underline } class FontIdentifier { <<FontIdentifier>> type xs:string } TextFormat --> TextLabelFormatType : <<TextLabelFormatType>> TextFormat --> SymbolFontStyleType : <<SymbolFontStyleType>> </pre>
type	TextLabelFormatType , ColorType , SymbolFontStyleType , JEP30-D10:EmptyType .

4.8.1.1.3.3.4.2. Default Text Format

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/TextFormat-Array/DefaultTextFormat
diagram	<pre> classDiagram class DefaultTextFormat { type DefaultTextFormatType } class DefaultTextFormatType { <<Composite Structure>> TextSize type xs:integer default 16 Color type DefaultColorType FontIdentifier type xs:string default Arial } DefaultTextFormat "1" --> "1" DefaultTextFormatType </pre>
type	DefaultTextFormatType , DefaultColorType .

The [DefaultTextFormat](#) does not have [FontStyle](#) since the default font style is normal (i.e., Not Bold, nor Italics nor Underline). The default values as follows,

- TextSize is defaulted to value “16”
- Color has the same default as previously defined for shapes.
- FontIdentifier is defaulted to value “Arial”

4.8.1.1.3.3.5. Terminal – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Terminal-Array
diagram	
type	TerminalGraphicalRepresentation-ArrayType , TerminalGraphicalRepresentationType , TerminalGroupsGraphicRepresentationType , SymbolTerminalLocationType , SymbolTerminalLabelType , TerminalFunctionGraphicsType , DefaultLabelSettingsType , TextLabelFormatType .

The **TerminalMapID** has a KeyRef called **SymbolTerminalMapKeyRef** which references a Key called **TerminalMapKey**. This **TerminalMapKey** is assigned to [ElectricalSection/Mapping-Array/Mapping/PackageTerminalMap/TerminalMap/ID](#). It therefore links in here the **TerminalName** and **TerminalNumber** for each terminal on the device. For standard based symbols for which no package has yet been defined, then it will just bring in the **TerminalName**. This is therefore suitable for Standards based Interfaces and Functions that are defined by Standards Bodies, while the assignment of the Standards based functionality into the package body can be left to the component manufacturer.

The **TerminalGroupGraphicsID** has a KeyRef called **TerminalGroupsGraphicKeyRef** which references a Key called **TerminalGroupsGraphicKey**. This **TerminalGroupsGraphicKey** is assigned to [Terminal-Array/TerminalGroupsGraphic/ID](#), as shown above in the diagram. It therefore links in here the **TerminalName** and **TerminalNumber** for each terminal on the device.

4.8.1.1.3.3.5. Terminal - Array (cont'd)

The *Logic-GroupID* has a KeyRef called *SymbolLogicalGroupKeyRef* which references a Key called *LogicalGroupKey*. This *LogicalGroupKey* is assigned to *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/LogicalGroup-Array/LogicalGroup/ID*. It therefore links in here the *TerminalName* to the respective logical group that it is assigned too.

All terminals that belong to a group as defined by the *Logic-GroupID* should be co-located on the symbol graphics, so that hierarchical symbol representation can be applied to the group as opposed to duplicating that same symbol representation for each terminal. Once the logical grouping of the terminals is applied to the symbol, then the following group symbols can be applied.

The *GroupDirectionSymbol* has a KeyRef called *SymbolTerminalGroupDirectionKeyRef* which references a Key called *LogicalGroupElectricalPropertyKey*. This *LogicalGroupElectricalPropertyKey* is assigned to *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/LogicalGroup-Array/LogicalGroup/PropertyID*. It therefore links the logical group of terminals as defined by the *Logic-GroupID* to the *Direction* in the *PropertiesType* via its *PropertyID*.

The *DifferentialPairID* has a KeyRef called *SymbolDifferentialPairKeyRef* which references a Key called *DifferentialPairKey*. This *DifferentialPairKey* is assigned to *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/DifferentialPair-Array/DifferentialPair/ID*. It therefore links pair of *TerminalName* and the *DifferentialPair* symbol for each respective pair of terminal on the device.

The *FunctionID* has a KeyRef called *SymbolTerminalGroupFunctionSymbolKeyRef* which references a Key called *LogicalGroupTerminalFunctionKey*. This Key is assigned to *PartModel/ElectricalSection /ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/LogicalGroup-Array/LogicalGroup /TerminalFunctionID*. It therefore links the logical group of terminals as defined by the *Logic-GroupID* to the Signal in the *SignalClassificationType* via the *TerminalFunctionID*.

4.8.1.1.3.3.5.1. Location

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Terminal-Array/Terminal/Location						
diagram	<pre> classDiagram class SymbolTerminalLocationType { <<SymbolTerminalLocationType>> } class Location { <<SymbolTerminalLocationType>> } class Position { <<SymbolTerminalPositionLocationType>> } class SymbolSide { <<SymbolSideType>> } class Boundary { <<JEP30-D10:PointXYType>> } class Interior { <<JEP30-D10:PointXYType>> } class JEP30-D10:PointXYType { <<x:xs:decimal>> <<y:xs:decimal>> } Location "1" -- "*" Position : Position "1" -- "*" Boundary : Position "1" -- "*" Interior : SymbolSide "1" -- "*" SymbolTerminalOrder : SymbolTerminalOrder "1" -- "*" JEP30-D10:PointXYType : JEP30-D10:PointXYType "1" -- "*" JEP30-D10:PointXYType : JEP30-D10:PointXYType "1" -- "*" JEP30-D10:PointXYType : </pre>						
type	SymbolTerminalLocationType, SymbolSideType, SymbolTerminalPositionLocationType, JEP30-D10:PointXYType.						
list of enumerate values	<p>SymbolSideType</p> <table border="1"> <tr> <td>1. Left</td> <td>2. Right</td> <td>3. Top</td> </tr> <tr> <td>4. Bottom</td> <td></td> <td></td> </tr> </table>	1. Left	2. Right	3. Top	4. Bottom		
1. Left	2. Right	3. Top					
4. Bottom							

The *SymbolTerminalOrder* is the sequence of the placement of the terminal on the Symbol from the top to the bottom when the terminal is placed on either the left or right side of the symbol. The *SymbolTerminalOrder* is from left to right when the terminal is placed on either the top or bottom side of the symbol.

Alternatively, for non-standard shaped symbol bodies, the terminals can be located via the *Position* branch in which the user can specify *Boundary* (i.e., outer point of the terminal line) and the *Interior* (i.e., inner point of the terminal line that is adjacent to the Symbol body)

4.8.1.1.3.3.5.2. Label

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Terminal-Array/Terminal/Label									
diagram	<pre> classDiagram class SymbolTerminalLabelType { Value type xs:string TextFormatID type xs:string TextFormat type TextLabelFormatType Visibility type SymbolTerminalLabelVisibilityType Orientation type SymbolTextOrientationType OriginOfTextString type SymbolTerminalLabelOriginType Label type SymbolTerminalLabelType *--> "1..*" LabelLocation type JEP30-D10:PointXYType } </pre>									
type	SymbolTerminalLabelType, TextLabelFormatType, SymbolTerminalLabelVisibilityType, SymbolTextOrientationType, SymbolTerminalLabelOriginType, JEP30-D10:PointXYType.									
list of enumerate values	Visibility <table border="1"> <tr> <td>Invisible</td> <td>Visible</td> <td></td> </tr> </table>	Invisible	Visible							
Invisible	Visible									
list of enumerate values	Orientation <table border="1"> <tr> <td>0 - Normal orientation</td> <td>1 - 90 Degree rotation</td> <td>2- 180 Degree rotation</td> </tr> <tr> <td>3 - 270 Degree rotation</td> <td></td> <td></td> </tr> </table>	0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation	3 - 270 Degree rotation					
0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation								
3 - 270 Degree rotation										
list of enumerate values	OriginOfTextString <table border="1"> <tr> <td>1. Northwest</td> <td>2. Leftcenter</td> <td>3. Southwest</td> </tr> <tr> <td>4. Uppercenter</td> <td>5. Center</td> <td>6. Backcenter</td> </tr> <tr> <td>7. Northeast</td> <td>8. Rightcenter</td> <td>9. Southeast</td> </tr> </table>	1. Northwest	2. Leftcenter	3. Southwest	4. Uppercenter	5. Center	6. Backcenter	7. Northeast	8. Rightcenter	9. Southeast
1. Northwest	2. Leftcenter	3. Southwest								
4. Uppercenter	5. Center	6. Backcenter								
7. Northeast	8. Rightcenter	9. Southeast								

4.8.1.1.3.3.5.3. Function Graphics

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Terminal-Array/Terminal/FunctionGraphics
diagram	<pre> classDiagram class TerminalFunctionGraphicsType { SignalID : xs:string ElectricalPropertyID : xs:string } class FunctionGraphics { <<TerminalFunctionGraphicsType>> } FunctionGraphics --> TerminalFunctionGraphicsType </pre>
type	TerminalFunctionGraphicsType .

The [SignalID](#) has a KeyRef called [SignalKeyRef](#) which references a Key called [SignalKey](#). This [SignalKey](#) is assigned to [PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalFunction-Array/TerminalFunction/DigitalFunction/Signal/ID](#). It therefore links in here the [Signal](#) value for this specific [TerminalName](#).

The [ElectricalPropertyID](#) has a KeyRef called [ElectricalPropertyKeyRef](#) which references a Key called [ElectricalPropertyKey](#). This [ElectricalPropertyKey](#) is assigned to [PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/ID](#). It therefore links in here the [Properties](#) value for this specific [TerminalName](#).

4.8.2. Required Circuitry - Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array
diagram	<pre> classDiagram class RequiredCircuitry { <<RequiredCircuitryType>> } class RequiredCircuitryType { attributes : 1..infinity ID : xs:string ElectricalSpecificationID : xs:string * 0..infinity ds:Signature : ds:SignatureType } class Net-Array { <<Net-ArrayType>> } RequiredCircuitry --> attributes RequiredCircuitry --> ID RequiredCircuitry --> ElectricalSpecificationID RequiredCircuitry --> ds:Signature </pre>
type	RequiredCircuitry-ArrayType , RequiredCircuitryType , Net-ArrayType , ds:SignatureType .

This section captures the necessary circuitry surrounding the device for the device to operate correctly. This [RequiredCircuitry](#) can be more complex than that defined via 4.7.1.6.

4.8.2.1. Net - Array

path	<ol style="list-style-type: none"> 1. PartModel/ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array/RequiredCircuitry/Net-Array 2. PartModel/ElectricalSection/ReferenceDesign-Array/ReferenceDesign/Net-Array
diagram	<pre> classDiagram class NetArrayType { NetType } class NetType { Name } class NetConnectionType { OrderablePartNumberID PartDetailsID ReferencePartDetailsID ManufacturerPartNumber ManufacturerName TerminalName TerminalNumber ReferenceDesignator } class Net { type NetType } class Net-Array { type Net-ArrayType } class NetConnection { type NetConnectionType } Net-Array "1..>" Net Net "1..>" NetType Net "1..>" NetConnection NetType "1..>" Name NetConnection "1..>" OrderablePartNumberID NetConnection "1..>" PartDetailsID NetConnection "1..>" ReferencePartDetailsID NetConnection "1..>" ManufacturerPartNumber NetConnection "1..>" ManufacturerName NetConnection "1..>" TerminalName NetConnection "1..>" TerminalNumber NetConnection "1..>" ReferenceDesignator </pre> <p>The diagram illustrates the UML Class Diagram for the Net-ArrayType, NetType, and NetConnectionType. It shows the relationships between these classes and their attributes.</p> <ul style="list-style-type: none"> Net-ArrayType contains the NetType class. NetType contains the Name attribute. NetConnectionType contains the following attributes: OrderablePartNumberID, PartDetailsID, ReferencePartDetailsID, ManufacturerPartNumber, ManufacturerName, TerminalName, TerminalNumber, and ReferenceDesignator. Net has a multiplicity of 1..> for both Net-Array and NetType. Net-Array has a multiplicity of 1..> for Net. NetType has a multiplicity of 1..> for Name. NetConnection has a multiplicity of 1..> for all attributes in NetConnectionType.
type	Net-ArrayType , NetType , NetConnectionType .

4.8.2.1. Net - Array (cont'd)

This section captures the list of *NetConnections* (otherwise known as the Netlist) between devices that makes up the required circuitry for the primary device being defined in the XML file. Since many parts can be defined within this xml structure, each part is identified via an ID or *ManufacturerPartNumber* and *ManufacturerName* combination. The details of use are defined as follows.

1. *OrderablePartNumberID*, or
 - a. Specify the exact Part from a specific Manufacturer. This part may be described via several Part Details ID's that are connected together under an orderable part number ID. This is a fully identifiable part.
2. *PartDetailsID*, or
 - a. Specify a limited set of specifications via a partial part number from a specified manufacturer. This is a partially anonymous part.
 - b. The limited set of specifications are critical to be met to satisfy the requirements of the circuit for the successful operation of the primary device.
3. *ReferencePartDetailsID*, or
 - a. Specify a limited set of specifications without having to specify any variation of the part number (partial or full), and without having to specify any manufacturer. This is a fully anonymous part, as the only detail may be 10K Resistor or any resistor between the value of 10K and 100K.
 - b. The limited set of specifications are critical to be met to satisfy the requirements of the circuit for the successful operation of the primary device.
4. Alternatively, for parts that are not defined within this xml, the provision of a *ManufacturerPartNumber* and *ManufacturerName*,
 - a. This branch is for Parts that are not specified in this XML File

The net is connected to either the *TerminalName* or to the *TerminalNumber* of the part identified under the above 4 choices.

The *ReferenceDesignator* is an optional element that is required in the event that the circuit being captured here contains two or more of the exact same devices, or device functions in the same circuit.

4.9. Mapping - Array

path	PartModel/ElectricalSection/Mapping-Array
diagram	<p>Mapping-ArrayType</p> <p>MappingType</p> <p>+ attributes</p> <p>ID type xs:string</p> <p>ElectricalMap type ElectricalMapType 0..∞</p> <p>TerminalMap type TerminalMapType 0..∞</p> <p>Use this Terminal Map structure which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. The Package and Die Terminal Map structures below will go obsolete in first release in 2028.</p> <p>PackageTerminalMap type PackageTerminalMapType 0..∞</p> <p>Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Package Terminal Map will go obsolete in first release in 2028.</p> <p>DieTerminalMap type DieTerminalMapType 0..∞</p> <p>Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Die Terminal Map will go obsolete in first release in 2028.</p> <p>SimulationMap type SimulationMapType 0..∞</p> <p>ds:Signature type ds:SignatureType 0..∞</p> <p>+ constraints</p>
type	Mapping-Array, MappingType, PackageTerminalMapType, DieTerminalMapType, SimulationMapType, ds:SignatureType.
NOTE	Use Terminal Map structure which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. The Package and Die Terminal Map structures below will go obsolete in first release in 2028.

4.9.1. Electrical Map – Array

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap
diagram	<pre> classDiagram class ElectricalMapType { ID OperationalMode Terminal DifferentialPairID Logical-GroupID SuperInterfaceID FunctionID RequiredCircuitryID ReferenceDesignID PropertyID TerminalFunctionID ParameterSetID ParameterID ElectricalSpecificationID ESD-ID } class OperationalModeType class TerminalElectricalMapType class DifferentialPairIDType class LogicalGroupIDType class SuperInterfaceIDType class FunctionIDType class RequiredCircuitryIDType class ReferenceDesignIDType class PropertyIDType class TerminalFunctionIDType class ParameterSetIDType class ParameterIDType class ElectricalSpecificationIDType class ESDIDType ElectricalMapType < -- OperationalModeType ElectricalMapType < -- TerminalElectricalMapType ElectricalMapType "0..∞" --> DifferentialPairIDType ElectricalMapType "0..∞" --> LogicalGroupIDType ElectricalMapType "0..∞" --> SuperInterfaceIDType ElectricalMapType "0..∞" --> FunctionIDType ElectricalMapType "0..∞" --> RequiredCircuitryIDType ElectricalMapType "0..∞" --> ReferenceDesignIDType ElectricalMapType "0..∞" --> PropertyIDType ElectricalMapType "0..∞" --> TerminalFunctionIDType ElectricalMapType "0..∞" --> ParameterSetIDType ElectricalMapType "0..∞" --> ParameterIDType ElectricalMapType "0..∞" --> ElectricalSpecificationIDType ElectricalMapType "0..∞" --> ESDIDType </pre>
type	ElectricalMapType, OperationalModeType, TerminalElectricalMapType.

4.9.1 Electrical Map – Array (cont'd)

The *ElectricalMap* is an array that contains a mapping of between

- a set of terminals as defined by the *TerminalMapID*, or
- a pre-defined group of terminals as defined the *DifferentialPairID* or *Logical-GroupID*, or
- a Function as defined by the *FunctionID*, or
- a required circuitry for the operation of the device as defined by the *RequiredCircuitryID*, or
- a reference design as defined by the *ReferenceDesignID*.

over to their properties, terminal functions and/or their electrical specifications as defined by their connections via *PropertyID*, *TerminalFunctionID*, and/or *ElectricalSpecificationID*.

Figure 47 — Sample NAND Gate Device shows a sample device that contains four NAND gates. Each gate is similar in terms of properties, function and electrical specification; therefore, we can associate several terminal mappings to these reference ID's.

A *TerminalMap-Array* sample is shown below where we just label the signals as A, B and C. This is suitable for parts with duplicate functions.

```
<Mapping-Array>
  <Mapping>
    <ElectricalMap>
      < Terminal>
        <TerminalMapID>Terminal Map ID 1</TerminalMapID>
      </Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 2</TerminalMapID>
      </Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 4</TerminalMapID>
      </Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 5</TerminalMapID>
      </Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 9</TerminalMapID>
      </Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 10</TerminalMapID>
      </Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 12</TerminalMapID>
      </Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 13</TerminalMapID>
      </Terminal>
    <TerminalGroupDescription>Input Logic Signal</TerminalGroupDescription>
    <PropertyID>Terminal Property ID 1</PropertyID>
    <TerminalFunctionID>Terminal Function ID 1</TerminalFunctionID>
    <ElectricalSpecificationID>Terminal Electrical Specification ID 1</ElectricalSpecificationID>
```

4.9.1 Electrical Map – Array (cont'd)

```
</ElectricalMap>
<ElectricalMap>
  < Terminal>
    <TerminalMapID>Terminal Map ID 3</TerminalMapID>
  </ Terminal>
  < Terminal>
    <TerminalMapID>Terminal Map ID 6</TerminalMapID>
  </ Terminal>
  < Terminal>
    <TerminalMapID>Terminal Map ID 8</TerminalMapID>
  </ Terminal>
  < Terminal>
    <TerminalMapID>Terminal Map ID 11</TerminalMapID>
  </ Terminal>
  <TerminalGroupDescription>Output Logic Signal</TerminalGroupDescription>
  <PropertyID>Terminal Property ID 2</PropertyID>
  <TerminalFunctionID>Terminal Function ID 2</TerminalFunctionID>
  <ElectricalSpecificationID>Terminal Electrical Specification ID 2</ElectricalSpecificationID>
</ElectricalMap>
<ElectricalMap>
  < Terminal>
    <TerminalMapID>Terminal Map ID 7</TerminalMapID>
    <TerminalDescription>Ground Terminal</TerminalDescription>
  </ Terminal>
  <PropertyID>Terminal Property ID 3</PropertyID>
  <TerminalFunctionID>Terminal Function ID 3</TerminalFunctionID>
  <ElectricalSpecificationID>Terminal Electrical Specification ID 3</ElectricalSpecificationID>
</ElectricalMap>
<ElectricalMap>
  < Terminal>
    <TerminalMapID>Terminal Map ID 14</TerminalMapID>
    <TerminalDescription> Power Terminal</TerminalDescription>
  </ Terminal>
  <PropertyID>Terminal Property ID 4</PropertyID>
  <TerminalFunctionID>Terminal Function ID 4</TerminalFunctionID>
  <ElectricalSpecificationID>Terminal Electrical Specification ID 4</ElectricalSpecificationID>
</ElectricalMap>
</Mapping>
</Mapping-Array>
```

4.9.1.1. Operational Mode

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode
diagram	<pre>classDiagram class OperationalModeType { FunctionalOperationalMode PowerOperationalMode LoadOperationalMode TestOperationalMode Other } class OperationalMode { type OperationalModeType } OperationalMode < -- OperationalModeType</pre> The diagram illustrates the operational mode hierarchy. At the top is a dashed-line box labeled 'OperationalModeType'. Inside this box are five sub-classes: 'FunctionalOperationalMode' (type 'FunctionalOperationalModeType'), 'PowerOperationalMode' (type 'PowerOperationalModeType'), 'LoadOperationalMode' (type 'LoadOperationalModeType'), 'TestOperationalMode' (type 'TestOperationalModeType'), and 'Other' (type 'xs:string'). Below this box is another dashed-line box labeled 'OperationalMode'. Inside this box is a single attribute 'type OperationalModeType'. A directed association line connects the 'OperationalMode' class to the 'OperationalModeType' class, with the multiplicity '*' at the 'OperationalMode' end and '1' at the 'OperationalModeType' end.
type	OperationalModeType

4.9.1.1.1. Functional Operational Mode

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode/FunctionalOperationalMode
diagram	<p>The diagram shows a class hierarchy for FunctionalOperationalModeType. At the top is FunctionalOperationalModeType, which contains several subclasses: Active, Burst, ContinuousConversion, One-Shot, ClockWake, Functional, PulseSkipping, Reset, Sleep, Sniff, Suspend, Standby, Trigger, Unprogrammed, OtherFunctional, and FunctionalModelInstance. There are two association points between FunctionalOperationalModeType and each of the subclasses. Below this class hierarchy is a separate box containing FunctionalOperationalMode, which has an association point to the FunctionalOperationalModeType class.</p> <pre> classDiagram FunctionalOperationalModeType { Active Burst ContinuousConversion OneShot ClockWake Functional PulseSkipping Reset Sleep Sniff Suspend Standby Trigger Unprogrammed OtherFunctional FunctionalModelInstance } FunctionalOperationalMode { <--> FunctionalOperationalModeType } </pre>
type	FunctionalOperationalModeType

4.9.1.1.2. Power Operational Mode

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode/PowerOperationalMode
diagram	<pre> classDiagram class PowerOperationalModeType { <<PowerOperationalModeType>> <<JEP30-D10:EmptyType>> <<PowerUpSequence>> <<JEP30-D10:EmptyType>> <<LowPower>> <<JEP30-D10:EmptyType>> <<HighPower>> <<JEP30-D10:EmptyType>> <<PowerDownMode>> <<JEP30-D10:EmptyType>> <<PowerOffSequence>> <<JEP30-D10:EmptyType>> } class PowerOperationalMode { <<PowerOperationalModeType>> } PowerOperationalMode "3" --> PowerOperationalModeType PowerOperationalModeType "3" --> PowerUpSequence PowerOperationalModeType "3" --> LowPower PowerOperationalModeType "3" --> HighPower PowerOperationalModeType "3" --> PowerDownMode PowerOperationalModeType "3" --> PowerOffSequence </pre>
type	PowerOperationalModeType

[PowerDownMode](#) differs from [PowerOffSequence](#), insofar that [PowerDownMode](#) brings the device to a Sleep or Hibernation mode, whereas [PowerOffSequence](#), completely shuts the device down.

4.9.1.1.3. Load Operational Mode

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode/LoadOperationalMode
diagram	<pre> classDiagram class LoadOperationalModeType { <<LoadOperationalModeType>> <<JEP30-D10:EmptyType>> <<ConstantCurrent>> <<JEP30-D10:EmptyType>> <<ConstantPower>> <<JEP30-D10:EmptyType>> <<ConstantResistance>> <<JEP30-D10:EmptyType>> <<ConstantVoltage>> <<JEP30-D10:EmptyType>> } class LoadOperationalMode { <<LoadOperationalModeType>> } LoadOperationalMode "3" --> LoadOperationalModeType LoadOperationalModeType "3" --> ConstantCurrent LoadOperationalModeType "3" --> ConstantPower LoadOperationalModeType "3" --> ConstantResistance LoadOperationalModeType "3" --> ConstantVoltage </pre>
type	LoadOperationalModeType

4.9.1.1.4. Test Operational Mode

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode/TestOperationalMode
diagram	<pre> classDiagram class TestOperationalMode { type TestOperationalModeType } class SerialWireDebug { type JEP30-D10:EmptyType } class JTAG { type JEP30-D10:EmptyType } class OtherTestMode { type xs:string } class TestInstance { type xs:string } TestOperationalMode < --> SerialWireDebug TestOperationalMode < --> JTAG TestOperationalMode < --> OtherTestMode TestOperationalMode < --> TestInstance </pre>
type	TestOperationalModeType

4.9.2. Terminal Map

path	PartModel/ElectricalSection/Mapping-Array/TerminalMap
diagram	<pre> classDiagram class TerminalMap { type TerminalMapType } class PackageID { type xs:string multiplicity 0..∞ } class DieID { type xs:string multiplicity 0..∞ } class Map { type MapType } class ID { type xs:string } class TerminalName { type StructuredNamingType } class RecommendedNetlistName { type StructuredNamingType } class InternalNodeName { type StructuredNamingType } class StandardTerminalName { type StandardTerminalNameMapType multiplicity 0..∞ } class CompanionStandardTerminalMap { type CompanionStandardTerminalMapType } TerminalMap < --> PackageID TerminalMap < --> DieID TerminalMap < --> Map Map < --> ID Map < --> TerminalName Map < --> RecommendedNetlistName Map < --> InternalNodeName Map < --> StandardTerminalName Map < --> CompanionStandardTerminalMap </pre> <p>Use this Terminal Map structure which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. The Package and Die Terminal Map structures below will go obsolete in first release in 2026.</p>
type	TerminalMapType , MapType , StructuredNamingType , StandardTerminalNameMapType , CompanionStandardTerminalMapType

4.9.2.1. Terminal Name

path	<code>PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName</code>
diagram	<pre> classDiagram class TerminalName { type StructuredNamingType } class StructuredNamingType { class NamingGroup class Name { type xs:string } class TerminalSelection { type JEP30-D10:TerminalSelectionType } } class NamingField-ArrayType { class NamingField { type NamingFieldType } } TerminalName "1" --> "1..∞" StructuredNamingType StructuredNamingType "1" --> "1" Name StructuredNamingType "1" --> "1" TerminalSelection StructuredNamingType "1" --> "1" NamingField-ArrayType NamingField-ArrayType "1..∞" --> "1" NamingField </pre>
type	<code>StructuredNamingType, JEP30-D10:TerminalSelectionType, NamingField-ArrayType, NamingFieldType.</code>
Group	<code>NamingGroup</code>

4.9.2.1.1.1.1. Terminal Selection

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName/TerminalSelection
diagram	<pre> classDiagram class TerminalSelection { <<JEP30-D10:TerminalSelectionType>> } class TerminalPatternID class PatternGroupID class TerminalGroupID class RegionID class RowTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class FromRowTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class ToRowTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class ColumnTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class FromColumnTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class ToColumnTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class PolarTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class FromPolarTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class ToPolarTerminalIndex { type MinIntegerOfOneType minIncl/maxIncl 1 } class TerminalNumber { type xs:string } class FromTerminalNumber { type xs:string } class ToTerminalNumber { type xs:string } TerminalSelection "0..>" TerminalPatternID TerminalSelection "0..>" PatternGroupID TerminalSelection "0..>" TerminalGroupID TerminalSelection "0..>" RegionID TerminalSelection "1..>" RowTerminalIndex RowTerminalIndex "*" FromRowTerminalIndex RowTerminalIndex "*" ToRowTerminalIndex RowTerminalIndex "*" ColumnTerminalIndex ColumnTerminalIndex "*" FromColumnTerminalIndex ColumnTerminalIndex "*" ToColumnTerminalIndex ColumnTerminalIndex "*" PolarTerminalIndex PolarTerminalIndex "*" FromPolarTerminalIndex PolarTerminalIndex "*" ToPolarTerminalIndex PolarTerminalIndex "*" TerminalNumber TerminalNumber "*" FromTerminalNumber TerminalNumber "*" ToTerminalNumber </pre> <p>The diagram illustrates the UML class structure for <code>JEP30-D10:TerminalSelectionType</code>. It includes the following classes and their associations:</p> <ul style="list-style-type: none"> <code>TerminalSelection</code> (type: <code>JEP30-D10:TerminalSelectionType</code>) is associated with <code>TerminalPatternID</code>, <code>PatternGroupID</code>, <code>TerminalGroupID</code>, and <code>RegionID</code>. <code>RowTerminalIndex</code> (type: <code>MinIntegerOfOneType</code>, minIncl/maxIncl: 1) is associated with <code>FromRowTerminalIndex</code> and <code>ToRowTerminalIndex</code>. <code>ColumnTerminalIndex</code> (type: <code>MinIntegerOfOneType</code>, minIncl/maxIncl: 1) is associated with <code>FromColumnTerminalIndex</code> and <code>ToColumnTerminalIndex</code>. <code>PolarTerminalIndex</code> (type: <code>MinIntegerOfOneType</code>, minIncl/maxIncl: 1) is associated with <code>FromPolarTerminalIndex</code> and <code>ToPolarTerminalIndex</code>. <code>TerminalNumber</code> (type: <code>xs:string</code>) is associated with <code>FromTerminalNumber</code> and <code>ToTerminalNumber</code>.
type	<code>JEP30-D10:TerminalSelectionType, MinIntegerOfOneType.</code>

4.9.2.1.2. Naming Field

path	<p>PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName/NamingField-Array/NamingField</p> <p>PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/RecommendedNetlistName/NamingField-Array/NamingField</p> <p>PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/InternalNodeName/NamingField-Array/NamingField</p> <p>PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/CompanionStandardTerminalMap/NamingField-Array/NamingField</p>
diagram	<pre> classDiagram class NamingFieldType { <<xs:string>> FieldName <<xs:string>> FieldDescription <<xs:string>> Code <<xs:string>> TerminalNumberPatternID <<xs:string>> TerminalNumber <<xs:string>> FromTerminalNumber <<xs:string>> ToTerminalNumber <<xs:string>> TerminalPatternID <<xs:string>> PatternGroupID <<xs:string>> TerminalGroupID <<xs:string>> RegionID <<NamingFieldDuplicateSuffixArrayType>> DuplicateSuffix-Array <<NamingFieldTerminalSelectionArrayType>> TerminalSelection-Array } class NamingField { <<NamingFieldType>> } class constraints </pre> <p>The diagram illustrates the UML Class Diagram for the NamingFieldType. It features a central NamingFieldType class containing attributes for FieldName, FieldDescription, and Code. A multiplicity of 1..∞ connects NamingFieldType to TerminalNumberPatternID. TerminalNumberPatternID has a multiplicity of 0..∞ connecting to TerminalNumber, which in turn has a multiplicity of 1..∞ connecting to FromTerminalNumber and ToTerminalNumber. There are also associations from NamingFieldType to TerminalPatternID (1..∞), PatternGroupID (0..∞), TerminalGroupID (0..∞), and RegionID (0..∞). Additionally, NamingFieldType contains two composite regions: DuplicateSuffix-Array and TerminalSelection-Array.</p>
type	NamingFieldType , NamingFieldDuplicateSuffixArrayType , NamingFieldTerminalSelectionArrayType

4.9.2.1.2.1. Duplicate Suffix - Array

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName/NamingField-Array/NamingField/DuplicateSuffix-Array
diagram	
type	NamingFieldDuplicateSuffix-ArrayType , NamingFieldDuplicateSuffixType , SubPatternGroupReferenceType , SubPatternStartPositionRelative-to-ParentPatternGroupType , JEP30-D10:SequentialTerminalNomenclatureOrderingType

4.9.2.1.2.1.1. Sub Pattern Group Reference

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName/NamingField-Array/NamingField/DuplicateSuffix-Array/DuplicateSuffix/SubPatternGroupReference
diagram	
type	SubPatternGroupReferenceType ,

4.9.2.1.2.1.2. Start Position

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName/NamingField-Array/NamingField/DuplicateSuffix-Array/DuplicateSuffix/StartPosition
diagram	
type	SubPatternStartPositionRelative-to-ParentPatternGroupType , SubPatternStartPositionRelative-to-ParentPatternGroupCodeType , SubPatternStartPositionRelative-to-ParentPatternGroupDescriptionType .

TABLE 5 – SUB PATTERN START POSITION RELATIVE-TO-PARENT PATTERN GROUP

Code	Description
SW	Southwest
SE	Southeast
NE	Northeast
NW	Northwest
L	Left
B	Back
R	Right
F	Front
BL	Back-Left
BR	Back-Right
FL	Front-Left
FR	Front-Right
LB	Left-Bottom
LT	Left-Top
RB	Right-Bottom
RT	Right-Top

4.9.2.1.2.1.3. Sequence

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName/NamingField-Array/NamingField/DuplicateSuffix-Array/DuplicateSuffix/Sequence
diagram 1 of 3	<pre> classDiagram class JEP30-D10::SequentialTerminalNomenclatureOrderingType { Code type TerminalNomenclatureOrderingCodeType Description type TerminalNomenclatureOrderingDescriptionType NumericalSequence type NumericalSequenceType AlphabeticalSequence type AlphabeticalSequenceType } Sequence < -- JEP30-D10::SequentialTerminalNomenclatureOrderingType </pre>
diagram 2 of 3	<pre> classDiagram class NumericalSequenceType { Prefix type xs:string Start type xs:string Numerical type xs:integer 0..∞ ExcludedNumerical type xs:integer 0..∞ Suffix type xs:string } NumericalSequence < -- NumericalSequenceType </pre>
diagram 3 of 3	<pre> classDiagram class AlphabeticalSequenceType { Prefix type xs:string Start type xs:string Character type xs:string 0..∞ ExcludedCharacter type xs:string 0..∞ Suffix type xs:string } AlphabeticalSequence < -- AlphabeticalSequenceType </pre>
type	SequentialTerminalNomenclatureOrderingType, TerminalNomenclatureOrderingCodeType, TerminalNomenclatureOrderingDescriptionType, NumericalSequenceType, AlphabeticalSequenceType.

4.9.2.1.1.3 Sequence (cont'd)**TABLE 6 – NAMING NOMENCLATURE ORDERING CODE & DESCRIPTION**

Code	Description
CW	Clockwise
CCW	Counter-Clockwise
L2R	Left-to-Right
B2F	Back-to-Front
R2L	Right-to-Left
F2B	Front-to-Back
S-H	Snake Horizontal
S-V	Snake Vertical
ZZ-H	Zig-Zag Horizontal
ZZ-V	Zig-Zag Vertical

Typically, the default sequence is a running number starting with the digit 1 and incrementing by 1. If this is the case, then the *NumericalSequence* or the *AlphabeticalSequence* do not have to be populated. However, in some cases, the Numerical or alphabetical sequence can override the default, have its own *Start* value in which case the sequence of applied numbers or applied characters is applied in the same order as the sequence defined in the xml.

4.9.2.1.1.3 Sequence (cont'd)

TABLE 7 - NAMING FIELD PATTERNS

Pattern	Naming Field Pattern	Sequential Code	Start Position	Towards Point 2	Towards Point 3	Towards Point 4	Towards Point 5	Last Point
1		CW	User Defined	Start + 1 CW	Start + 2 CW	Start + 3 CW	Start + 4 CW	Start - 1 CW
2		CCW	User Defined	Start + 1 CCW	Start + 2 CCW	Start + 3 CCW	Start + 4 CCW	Start - 1 CCW
3		ZZ-V	BL	TL	BL+1	TL+1	BL+2	TR
4		ZZ-V	BR	TR	BR-1	TR-1	BR-2	TL
5		ZZ-V	TR	BR	TR-1	BR-1	TR-2	BL
6		ZZ-V	TL	BL	TL+1	BL+1	TL+2	BR
7		ZZ-H	TL	TR	T-1L	T-1R	T-2L	BR
8		ZZ-H	BL	BR	B+1L	B+1R	B+2L	TR
9		ZZ-H	BR	BL	B+1R	B+1L	B+2R	TL
10		ZZ-H	TR	TL	T-1R	T-1L	T-2R	BL
11		S-V	TL	BL	BL+1	TL+1	TL+2	IF(Col # = odd, BR, TR)
12		S-V	BL	TL	TL+1	BL+1	BL+2	IF(Col # = odd, TR, BR)
13		S-V	TR	BR	BR-1	TR-1	TR-2	IF(Col # = odd, BL, TL)
14		S-V	BR	TR	TR-1	BR-1	BR-2	IF(Col # = odd, TL, BL)
15		S-H	TL	TR	T-1R	T-1L	T-2L	IF(Col # = odd, BR, BL)
16		S-H	BL	BR	B+1R	B+1L	B+2L	IF(Col # = odd, TR, TL)
17		S-H	TR	TL	R-1L	T-1R	T-2R	IF(Col # = odd, BL, BR)

4.9.2.1.1.3 Sequence (cont'd)

TABLE 7 - NAMING FIELD PATTERNS (cont'd)

Pattern	Naming Field Pattern	Sequential Code	Start Position	Towards Point 2	Towards Point 3	Towards Point 4	Towards Point 5	Last Point
18		S-H	BR	BL	B+1L	B+1R	B+2R	IF(Col # = odd, TL, TR)
19		L2R	L	L+1	L+2	L+3	L+4	R
20		R2L	R	R-1	R-2	R-3	R-4	L
21		B2F	B	B+1	B+2	B+3	B+4	T
22		F2B	T	T-1	T-2	T-3	T-4	B

By combining the start position of the naming field nomenclature via the *SubPatternStartPositionRelative-to-ParentPatternGroupType* with the *Sequence* pattern, we can achieve the naming field patterns outlined in Table 7 - Naming Field Patterns for naming field nomenclature.

For positions of BL, TL, BR, TR, the following is also valid for naming fields for *SubPatternGroupReference* positioned in non-matrix patterns within the *ParentPatternGroupReference*, namely,

1. The position of BL (Bottom Left) corresponds to
 - (a) For *SubPatternGroupReference* with a position of Left-to-Right within the *ParentPatternGroupReference*, then Left-bottom (LB),
 - (b) For *SubPatternGroupReference* with a position of Back-to-Front within the *ParentPatternGroupReference*, then Back-left (BL),
 - (c) For *SubPatternGroupReference* with a position of Diagonal within the *ParentPatternGroupReference*, then Southwest (SW),
2. The position of TL (Top Left) corresponds to
 - (a) For *SubPatternGroupReference* with a position of Left-to-Right within the *ParentPatternGroupReference*, then Left-top (LT),
 - (b) For *SubPatternGroupReference* with a position of Back-to-Front within the *ParentPatternGroupReference*, then Front-left (FL),
 - (c) For *SubPatternGroupReference* with a position of Diagonal within the *ParentPatternGroupReference*, then Northwest (NW),

4.9.2.1.1.1.3 Sequence (cont'd)

3. The position of BR (Bottom Right) corresponds to
 - (a) For *SubPatternGroupReference* with a position of Left-to-Right within the *ParentPatternGroupReference*, then Right-bottom (RB),
 - (b) For *SubPatternGroupReference* with a position of Back-to-Front within the *ParentPatternGroupReference*, then Back-right (BR),
 - (c) For *SubPatternGroupReference* with a position of Diagonal within the *ParentPatternGroupReference*, then Southeast (SE),
4. The position of TR (Top Right) corresponds to
 - (a) For *SubPatternGroupReference* with a position of Left-to-Right within the *ParentPatternGroupReference*, then Right-top (RT),
 - (b) For *SubPatternGroupReference* with a position of Back-to-Front within the *ParentPatternGroupReference*, then Front-right (FR),
 - (c) For *SubPatternGroupReference* with a position of Diagonal within the *ParentPatternGroupReference*, then Northeast (NE).

TL	TL+1	TL+2	:	TR-2	TR-1	TR
T-1L	T-1L+1	T-1L+2	:	T-1R-2	T-1R-1	T-1R
T-2L	T-2L+1	T-2L+2	:	T-2R-2	T-2R-1	T-2R
:	:	:	:	:	:	:
B+2L	B+2L+1	B+2L+2	:	B+2R-2	B+2R-1	B+2R
B+1L	B+1L+1	B+1L+2	:	B+1R-2	B+1R-1	B+1R
BL	BL+1	BL+2	:	BR-2	BR-1	BR

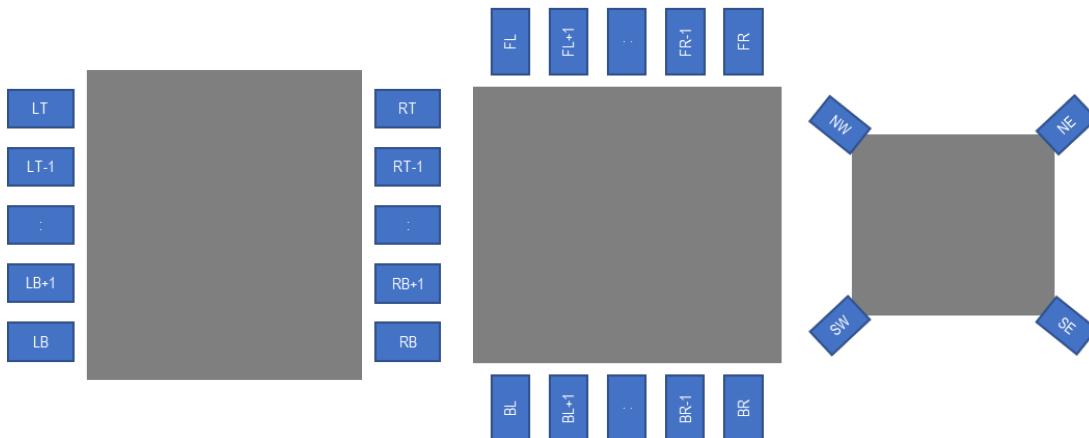


FIGURE 47 – TERMINAL NAMING FIELD PATTERN SAMPLES

4.9.2.1.2.2. Terminal Selection - Array

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName/NamingField-Array/NamingField/TerminalSelection-Array
diagram	<pre> classDiagram class TerminalSelection { ID Select Deselect } class Sequence { type JEP30-D10:SequentialTerminalNomenclatureOrderingType } TerminalSelection "1..∞" --> ID TerminalSelection "*" --> Select TerminalSelection "*" --> Deselect Select "*" --> Sequence </pre>
type	NamingFieldTerminalSelection-ArrayType , NamingFieldTerminalSelectionType , JEP30-D10:TerminalSelectionType , JEP30-D10:SequentialTerminalNomenclatureOrderingType

4.9.2.2. Standard Terminal Name

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/StandardTerminalName
diagram	<pre> classDiagram class StandardTerminalName { FunctionID } class FunctionID { type xs:string } class Name { type xs:string } class BaseName { type xs:string } StandardTerminalName "0..∞" --> FunctionID FunctionID "1..∞" --> Name FunctionID "1..∞" --> BaseName </pre>
type	StandardTerminalNameMapType .

4.9.2.3. Companion Standard Terminal Map

path	PartModel/ElectricalSection/Mapping-Array/Mapping/TerminalMap/Map/TerminalName
diagram	<pre> classDiagram class CompanionStandardTerminalMap { <<CompanionStandardTerminalMap>> } class CompanionStandardTerminalMapType { <<CompanionStandardTerminalMapType>> FunctionID : xs:string NamingFieldGroup } class NamingFieldGroup class NamingField_Array { <<NamingField-Array>> type NamingFieldArrayType } CompanionStandardTerminalMap "0..>" CompanionStandardTerminalMapType CompanionStandardTerminalMapType "0..>" FunctionID CompanionStandardTerminalMapType "0..>" NamingFieldGroup NamingFieldGroup "0..>" NamingField_Array </pre>
type	CompanionStandardTerminalMapType , NamingField-ArrayType .
Group	NamingFieldGroup

4.9.3. Package Terminal Map

path	PartModel/ElectricalSection/Mapping-Array/PackageTerminalMap
diagram	<pre> classDiagram class PackageTerminalMap { <<PackageTerminalMap>> } class PackageTerminalMapType { <<PackageTerminalMapType>> ID : xs:string PackageID : xs:string } class TerminalMap { <<TerminalMap>> type PackageDieTerminalMapType } class PackageDieTerminalMapType { RecommendedNetlistName : xs:string TerminalName : xs:string TerminalNumber : xs:string InternalNodeName : xs:string FunctionID : xs:string StandardTerminalName : xs:string CompanionTerminal } PackageTerminalMap "0..>" PackageTerminalMapType PackageTerminalMapType "1..>" TerminalMap TerminalMap "1..>" PackageDieTerminalMapType </pre> <p>Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Package Terminal Map will go obsolete in first release in 2028.</p>
type	PackageTerminalMapType , PackageDieTerminalMapType .
NOTE	Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Package Terminal Map will go obsolete in first release in 2028.

4.9.3 Package Terminal Map (cont'd)

The *TerminalMap* is an array that contains a mapping of *TerminalNames* to *TerminalNumbers*, or *InternalNodeName*. When Terminal Numbers is populated, it is assumed that the Package section of the PartModel is also populated so that the consumer knows which package that these terminal numbers belong to as defined by the *PackageID*.

Figure 47 shows a sample device that contains four NAND gates. The *TerminalNumber* is an unbounded element that then captures the Terminal Numbers that is specific to a *TerminalName*.

A *TerminalMap-Array* sample is shown below.

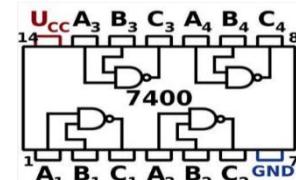


Figure 48 — Sample NAND Gate Device

```

<Mapping-Array>
  <Mapping>
    <ID>Mapping ID 1</ID>
    <PackageTerminalMap>
      <ID>Package Terminal Map ID 1</ID>
      <PackageID>Package ID 1</PackageID> <!--SOIC-->
      <TerminalMap>
        <ID>Terminal Map ID 1</ID>
        <TerminalName>A1</TerminalName>
        <TerminalNumber>1</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 2</ID>
        <TerminalName>B1</TerminalName>
        <TerminalNumber>1</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 3</ID>
        <TerminalName>C1</TerminalName>
        <TerminalNumber>1</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 4</ID>
        <TerminalName>A2</TerminalName>
        <TerminalNumber>1</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 5</ID>
        <TerminalName>B2</TerminalName>
        <TerminalNumber>1</TerminalNumber>
      </TerminalMap>
      <TerminalMap>
        <ID>Terminal Map ID 6</ID>
        <TerminalName>C2</TerminalName>
        <TerminalNumber>1</TerminalNumber>
      </TerminalMap>
    </PackageTerminalMap>
  </Mapping>
</Mapping-Array>

```

4.9.3 Package Terminal Map (cont'd)

```

<TerminalMap>
  <ID>Terminal Map ID 7</ID>
  <TerminalName>GND</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
<TerminalMap>
  <ID>Terminal Map ID 8</ID>
  <TerminalName>C4</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
<TerminalMap>
  <ID>Terminal Map ID 9</ID>
  <TerminalName>B4</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
<TerminalMap>
  <ID>Terminal Map ID 10</ID>
  <TerminalName>A4</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
<TerminalMap>
  <ID>Terminal Map ID 11</ID>
  <TerminalName>C3</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
<TerminalMap>
  <ID>Terminal Map ID 12</ID>
  <TerminalName>B3</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
<TerminalMap>
  <ID>Terminal Map ID 13</ID>
  <TerminalName>A3</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
<TerminalMap>
  <ID>Terminal Map ID 14</ID>
  <TerminalName>VCC</TerminalName>
  <TerminalNumber>1</TerminalNumber>
</TerminalMap>
</PackageTerminalMap>
</Mapping>
</ Mapping-Array>

```

The identification of an *InternalNode* (A shown in Figure 48 — Sample Transistor Circuit) becomes particularly important when trying to describe the internal circuit of a device that has Nodes that are not connected to any external terminal of the part.

These nodes can be optionally connected to a Terminal Name and Terminal Number.

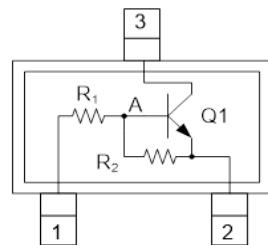


Figure 49 — Sample Transistor

4.9.4. Die Terminal Map

path	PartModel/ElectricalSection/Mapping-Array/DieTerminalMap
diagram	<p>Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Die Terminal Map will go obsolete in first release in 2028.</p>
type	DieTerminalMapType, PackageDieTerminalMapType.
NOTE	Use Terminal Map which combines Package and Die terminal Map, plus enhances a structured Terminal Name Mapping that avoids mapping out every terminal. This Die Terminal Map will go obsolete in first release in 2028.

4.9.5. Simulation Map

path	PartModel/ElectricalSection/Mapping-Array/SimulationMap
diagram	
type	SimulationMapType.

4.10. Simulation Model - Array

path	PartModel/ElectricalSection/SimulationModel-Array
diagram	<p>The diagram illustrates the UML class structure for the SimulationModel-ArrayType. It features a main class SimulationModel-ArrayType with a dashed boundary. Inside, there is a nested class SimulationModelType containing an attribute ID (type: xs:string). A multiplicity of 1..∞ connects SimulationModel-ArrayType to SimulationModel (type: SimulationModelType). The SimulationModel class has a dashed boundary and contains a nested class SPICE-Functional (type: SPICE-FunctionalType) and another nested class SPICE-SignalIntegrity (type: SPICE-SignalIntegrityType). Outside the SimulationModel boundary, there is a large vertical stack of simulation models: VHDL-AMS, VHDL, Verilog, Verilog-AMS, SystemC, OtherLanguageType, Behavioral, RTL, Structural, Other, IBIS, and OtherModelType. Below these is a Model class (type: xs:string) with a nested class ModelDescription (type: xs:string) and a reference ds:Signature (type: ds:SignatureType) with multiplicity 0..∞. A dashed boundary labeled constraints encloses the bottom part of the diagram.</p>
type	SimulationModel-ArrayType , SimulationModelType , SPICE-FunctionalType , SPICE-SignalIntegrityType , ds:SignatureType .

Electronic circuit simulation uses mathematical models to replicate the behavior of an actual electronic device or circuit. Simulation software allows for modeling of circuit operation prior to building the product. There are various types of simulation software which require various **SimulationModel** libraries to perform the function of analyzing the device within the intended product design.

4.10 Simulation Model - Array (cont'd)

SPICE (Simulation Program with Integrated Circuit Emphasis) is a general-purpose, open source analog electronic circuit simulator. It is a program used in integrated circuit and board-level design to check the integrity of circuit designs and to predict circuit behavior. The Spice models can be either SPICE-Functional or SPICE-SignalIntegrity.

VHDL (VHSIC Hardware Description Language) is a hardware description language used in electronic design automation to describe digital and mixed-signal systems such as field-programmable gate arrays and integrated circuits.

Verilog, standardized as IEEE 1364, is a hardware description language (HDL) used to model electronic systems. It is most commonly used in the design and verification of digital circuits at the register-transfer level of abstraction. It is also used in the verification of analog circuits and mixed-signal circuits, as well as in the design of genetic circuits

Input/output Buffer Information Specification or **IBIS** is a specification of a method for integrated circuit vendors to provide information about the input/output buffers of their product to their prospective customers without revealing the intellectual property of their implementation and without requiring proprietary encryption keys.

Model captures the URL link for the appropriate simulation model, or in the event that the simulation model is provided with the XML File, then the Model file name.

4.11. Reference Design - Array

path	PartModel/ElectricalSection/ReferenceDesign-Array
diagram	
type	ReferenceDesign-ArrayType, ReferenceDesignType, Net-ArrayType, ds:SignatureType.

This section has the same structure as section 4.6.2 Required Circuitry - Array above, however it is to capture **ReferenceDesigns** as opposed to **RequiredCircuitry**. Component Manufacturers can provide reference designs via this section, to enable and support their users with various Product designs, in which their parts will be used.

4.12. Software Interface Description - Array

path	PartModel/ElectricalSection/SoftwareInterfaceDescription-Array
diagram	<pre> classDiagram class SoftwareInterfaceDescription-Array { <<SoftwareInterfaceDescription>> <<SoftwareInterfaceDescription-Array>> } class SoftwareInterfaceDescription { <<SoftwareInterfaceDescription>> <<SoftwareInterfaceDescriptionType>> } class SoftwareInterfaceDescriptionType { <<SoftwareInterfaceDescriptionType>> attributes ID : xs:string IP-XACT : IP-XACT-Type SystemRDL : SystemRDLType ds:Signature : ds:SignatureType } SoftwareInterfaceDescription-Array "1..∞" --> "1..∞" SoftwareInterfaceDescription SoftwareInterfaceDescription-Array "1..∞" --> SoftwareInterfaceDescriptionType </pre>
type	SoftwareInterfaceDescription-ArrayType, SoftwareInterfaceDescriptionType, IP-XACT-Type, SystemRDLType, ds:SignatureType.

Component Manufacturers are now able to include their *SoftwareInterfaceDescription* in the form of either *IP-XACT* or a *SystemRDL* along with other part related technical content in one submission to their customers.

4.12.1. IP-XACT

path	PartModel/ElectricalSection/SoftwareInterfaceDescription-Array/SoftwareInterfaceDescription/IPXact
diagram	<pre> classDiagram class IP-XACT { <<IP-XACT>> <<IP-XACT-Type>> } class MemoryMap { <<ipxact:memoryMapType>> ipxact:memoryMapType } class File { <<xs:string>> } class IPXactFile { <<JEP30-D10:EmptyType>> } IP-XACT "1..∞" --> "1..∞" MemoryMap MemoryMap "1..∞" --> File File --> IPXactFile </pre>
type	IP-XACT-Type, ipxact:memoryMapType, JEP30-D10:EmptyType

IP-XACT is an XML format that defines and describes individual, re-usable electronic circuit designs to facilitate their use in creating integrated circuits. *IP-XACT* describes the meta-data of IP designs and flows and the interconnection of IP interfaces in a standard specification. A sub-

section of the *IP-XACT* specification, memoryMap, describes the registers, memories and address maps used in the circuit designs.

4.12.1 IP-XACT (cont'd)

The component manufacturer has the option to provide the data as an XML structured data (as per memoryMap.xsd) or as a file reference.

For more information on IP-XACT, refer to <https://www.accellera.org/downloads/standards/ip-xact>.

4.12.2 System RDL

path	PartModel/ElectricalSection/SoftwareInterfaceDescription-Array/SoftwareInterfaceDescription/SystemRDL
diagram	<pre> classDiagram class SystemRDL { <<SystemRDLType>> } class File { <<xs:string>> } class SystemRDLFile { <<JEP30-D10:EmptyType>> } SystemRDL "*" -- "*" File File "*" -- "*" SystemRDLFile </pre>
type	SystemRDLType, JEP30-D10:EmptyType

SystemRDL is a language for the design and delivery of intellectual property (IP) products used in complex digital systems. SystemRDL semantics supports the description of a system of registers and memory. SystemRDL can also be used to automatically generate and synchronize the register specification in hardware design, software development, verification, and documentation of the complex digital system.

For more information on SystemRDL, refer to <https://www.accellera.org/activities/working-groups/systemrdl>.

5 Quick TeX reference

The notation used for representing electrical symbols is based on the TeX language. TeX is a system for typesetting high quality technical documents. It provides a text-based language that describes complex mathematical formulas. TeX commands commonly start with a backslash and are grouped with curly braces. These commands include all kinds of technical symbols, and modifiers such as subscripts and superscripts. TeX is a widely used formatting system in both academia and industry, and it is in the public domain. Reference the "Short Math Guide by the American Mathematical Society.

5.1. Syntax

Table 8 — LaTeX Syntax Sample

Syntax	Description
<code>_abc</code>	Subscript a_b_c
<code>^abc</code>	Superscript a^{b_c}
<code>\overline{abc}</code>	Overline $\bar{a}b\bar{c}$
<code>\bar{a}</code>	\bar{a}
<code>\hat{a}</code>	\hat{a}
<code>\tilde{a}</code>	\tilde{a}
<code>\vec{a}</code>	\vec{a} with arrow on top
<code>\sqrt{a}</code>	Square root of a

5.2. Symbols

Table 9 — LaTeX Symbols and Descriptions Sample

Symbol	Description
<code>\alpha, \beta, \gamma, ... , \omega</code>	$\alpha, \beta, \gamma, \dots, \omega$ (lowercase greek letters)
<code>\Gamma, \Delta, \Theta, ..., \Omega</code>	$\Gamma, \Delta, \Theta, \dots, \Omega$ (non-latin-looking uppercase greek letters)
<code>_, \^, \#, \%, \&, \backslash</code>	$_, \^, \#, \%, \&, \backslash$ (symbols with special meaning)
<code>\infty</code>	∞
<code>\circ</code>	\circ
<code>\pm</code>	\pm
<code>\leq, \geq, \neq</code>	\leq, \geq, \neq
<code>\backslash</code>	\backslash
<code>\sim</code>	\sim
<code>\approx</code>	\approx
<code>\leftarrow, \rightarrow</code>	\leftarrow, \rightarrow
NOTE < and > are special symbols in XML, so they have to be escaped in the document as < and >. XML tools may or may not take care of this.	

6 Rule Syntax

The rule syntax within the PartModel is used to create string definitions in the XML element *Rule*. The *Rule* element occurs in the *ElectricalSpecification* section where there is a *TestCondition/Rule* and also under the *ParameterSet*. However the is accessed via its ID from several places such as:-

1. *PartModel/ManufacturerPartNumber-Array/PartDetails/ElectricalSpecificationID*,
2. *PartModel/ReferenceManufacturerPartNumber-Array/PartDetails/ElectricalSpecificationID*,
3. *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalMap-Array/TerminalMap/ElectricalSpecificationID*,
4. *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/InternalNode-Array/InternalNode/ElectricalSpecificationID*,
5. *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/DifferentialPair-Array/DifferentialPair/ElectricalSpecificationID*,
6. *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/Logical-Group-Array/Logical-Group/ElectricalSpecificationID*,
7. *PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/ElectricalSpecificationID*,
8. *PartModel/ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array/RequiredCircuitry/ElectricalSpecificationID*,
9. *PartModel/ElectricalSection/ReferenceDesign-Array/ReferenceDesign/ElectricalSpecificationID*.

In some cases such as defining the electrical relationship between terminals the rule is defining a single electrical parameter of one or more terminals as a function of other terminals.

6.1. Rule applied to a Logical-Group

The *Rule* elements within the PartModel schema allow the electrical parameter of a terminal or group of terminals to be described as a function of another terminal or group of terminals. As an example a part defines as a limit parameter that a group of terminals has a maximum total current, I_{OHmax} and I_{OLmax} , for all combined outputs, which should not exceed ± 48 mA. If the output terminals included *TerminalNumbers* 13 through *TerminalNumbers* 18, then a formula can be written which describes the sum of the current for each of the *TerminalNumbers* 13 – 18 must be greater than -48 mA and less than 48 mA. A PartModel XML that defines this electrical characteristic would use the symbol $I_{\{OHmax\}}$ as the symbol name. The rule will be:

$$I(13)+I(14)+I(15)+I(16)+I(17)+I(18) < 48mA$$

$I(13)$ references the steady-state current of terminal 13. The '+' sign represents the summing the operands on each side of the symbol. The less than sign expresses that what is to the left of the less than sign must be less than what is to the right. The "48mA" is a constant and its unit is mA. To complete the example a second rule would be written for the negative side and applied to the symbol $I_{\{OLmax\}}$. This rule is:

$$-48mA < I(13)+I(14)+I(15)+I(16)+I(17)+I(18)$$

These 2 rules would be represented in the xml file under the ParameterSet as follows:

```
<ParameterSet>
  <Parameter>
    <LaTeX-Symbol>I_{\{OHmax\}}</LaTeX-Symbol>
    <SymbolDescription>Capacitance change</SymbolDescription>
    <Rule>
      <LaTeX-Rule>I(13)+I(14)+I(15)+I(16)+I(17)+I(18)&lt;48mA</LaTeX-Rule>
    </Rule>
  </Parameter>
  <Parameter>
    <LaTeX-Symbol>I_{\{OLmax\}}</LaTeX-Symbol>
    <Rule>
      <LaTeX-Rule>-48mA&lt;I(13)+I(14)+I(15)+I(16)+I(17)+I(18)</LaTeX-Rule>
    </Rule>
  </Parameter>
</ParameterSet>
```

A rule such as this would be place in the *ElectricalSpecification* section but referenced from the *TerminalDetails/TerminalGrouping/Logical-Group-Array/Logical-Group/ElectricalSpecificationID* as shown in 4.7.1.5.6 Logical-Group - Array.

6.2. Rule applied to a Terminal Map

It is also possible to define an electrical parameter for a single terminal as a function of another terminal's electrical parameter. As an example a terminal's Low-level input voltage, V_{IL} , is defined as the value of Ground plus 0.6V. *TerminalNumber* 4 of the part is connected to Ground. A PartModel XML that defines the electrical characteristic would use the symbol $V_{\{IL\}}$ and Volts would be the unit of the electrical specification. The rule for the parameter value will be:

$$=V(4)+0.6V$$

These 2 rules would be represented in the xml file under the ParameterSet as follows:-

```
<ParameterSet>
  <Parameter>
    <LaTeX-Symbol>V_{\{IL\}}</LaTeX-Symbol>
    <Rule>
      <LaTeX-Rule>V(4)+0.6V</LaTeX-Rule>
    </Rule>
  </Parameter>
</ParameterSet>
```

$V(4)$ represents the voltage of terminal 4. "0.6V" is a constant whose unit is Volts. If the name of the terminal 4 is V_{SS} the same rule can be defined as.

$$=V(V_{\{SS\}})+0.6V$$

These 2 rules would be represented in the xml file under the ParameterSet as follows:-

```
<ParameterSet>
  <Parameter>
    <LaTeX-Symbol>V_{\{IL\}}</LaTeX-Symbol>
    <Rule>
      <LaTeX-Rule>=V(V_{\{SS\}})+0.6V</LaTeX-Rule>
    </Rule>
  </Parameter>
</ParameterSet>
```

A rule such as this would be place in the *ElectricalSpecification* section but referenced from the *TerminalDetails/TerminalMap-Array/TerminalMap/ElectricalSpecificationID* as shown in 4.7.1.1 Terminal Map - Array.

6.3. Rule Functions

Rules can be defined in multiple ways through the use of arithmetic operators, comparison operators, constant values, references to electrical characteristics of other terminals, and functions.

6.3.1. Arithmetic Operations

Table 10 — Arithmetic Operations

Operator	Description
+	Addition of right and left operand
-	Subtraction of right operand from left operand.
/	Division of left operand by right operand.
*	Multiplication of left and right operand
^	Raise the left hand operand to the power of the value of the right hand operand.

Arithmetic operations follow the basic order of precedence. * and / take precedence over + and -.

6.3.2. Modifying Precedence

Parenthesis allow you to define the proper precedence of your operations. Take the following as an example:

$$2 + 3 * 4$$

The calculation would be interpreted as follows:

$$2 + 3 * 4$$

$$2 + 12$$

$$14$$

Adding parenthesis will change the calculation and the result as follows.

$$(2 + 3) * 4$$

$$(5) * 4$$

$$20$$

Use parentheses where necessary to ensure you explicitly and correctly define rules.

6.3.3. Function to Reference Other Terminals

Functions enable rules to reference electrical parameters of other terminals. The function has a specific form. The name of the function is any valid symbol as defined by JEDEC. The parameter, inside the parentheses, is any valid *TerminalName* or *TerminalNumber* which must be defined in the *TerminalMap* section of the XML file.

6.3.3 Function to Reference Other Terminals (cont'd)

Table 11 — Rule Functions

Function	Description
I(T)	Steady-state current of terminal T (Positive value returned for flowing out, Negative value returned for current flowing in)
V(T)	Steady-state voltage of terminal T
P(T)	Power of terminal T

6.3.3.1. Functions

Additional mathematical operations that can be performed in the rule.

Table 12 — Mathematical Functions

Operation	Description
ABS(x)	Absolute value of expression x.
SQRT(X)	Square Root of expression x.
AVG(X, Y, ...)	Average of expression X and expression Y. Must have at least 2 parameters. Allows any number of parameters.
MAX(X, Y, ...)	Returns the maximum value out of X and Y. Must have at least 2 parameters. Allows any number of parameters.
MIN(X, Y, ...)	Returns the minimum value out of X and Y. Must have at least 2 parameters. Allows any number of parameters.
EXP(X)	Returns e to the power of X.
LOG(X)	Returns the base 10 logarithm of X.
LOG(X, [Y])	Returns the base Y logarithm of X
LN(X)	Returns the natural log of X.
POWER(X, Y)	

6.3.4. Constants

Any value that begins with a number is considered a constant value. The constant may be defined as an integer or a decimal. A '.' is used as the separator for the decimal portion of the constant. If the constant has a unit, then that unit must follow the value. Examples include:

1. 10 mA
2. 10.5 V
3. 0.6 W

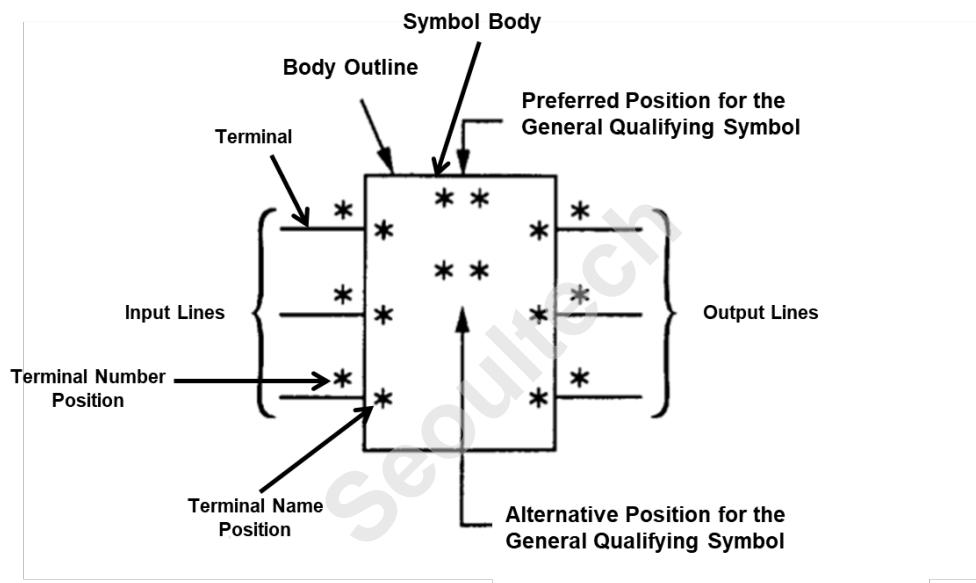
The valid unit abbreviations can be found in Table 3 — UOM Enumerated Lists.

Annex A (informative) Graphic Symbol Definition

The following recommendations are based on the IEEE Standard/ American National Standard/ Canadian Standard “Graphic Symbols for Electrical and Electronics Diagrams” IEEE Std 315-1975 (Reaffirmed 1993), ANSI Y32.2-1975 (Reaffirmed 1989), CSA Z99-1975.

While Drafting Standards define many “terminal naming requirements”, it is recommended that Terminal names used in the Symbol are the same as that specified by the component manufacturer.

A.1. General Recommended Graphic Sizing



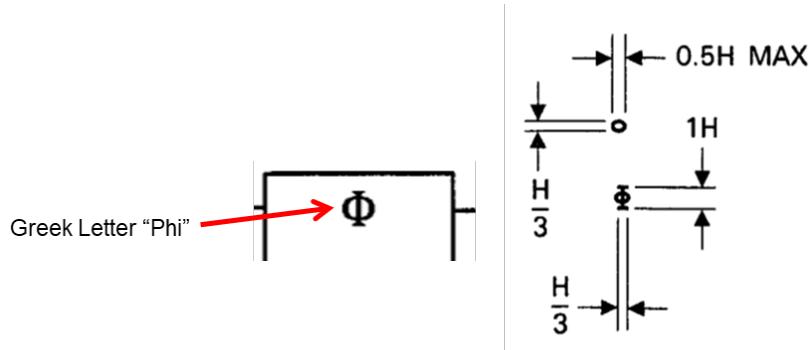
Typical symbol properties include Symbol Name and Reference Designator. Terminal Names are placed inside the symbol body. As the text grows within the body outline, the body outline should grow to prevent terminal names on the left from overlapping the terminal names on the right.

Body outline size and placements of terminals should be placed on a grid. Grid sizes should be defined in terms of grid spacing, so that customers who want to see their graphics on a metric grid versus an imperial grid can simply define the grid size to either a metric value or an imperial value.

The normal conventions are that Input terminals are on the left with output terminals are on the right. Positive power terminals are on the top, while ground and negative power terminals are on the bottom of the body outline. However, violating these conventions does not violate the drafting standards.

A.1 General Recommended Graphic Sizing (cont'd)

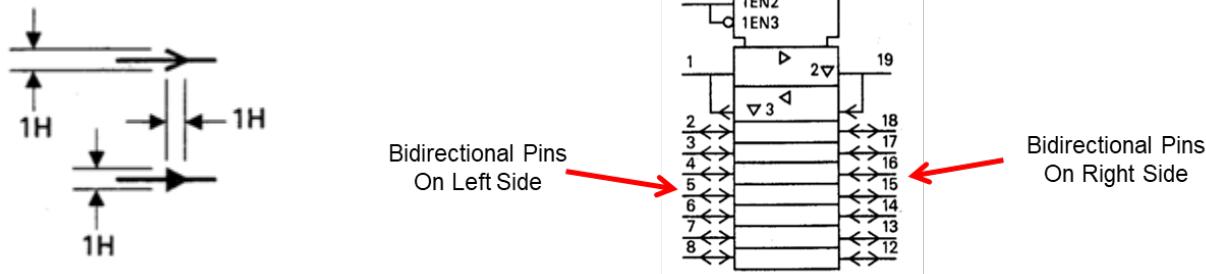
The "General Qualifying Symbol" is the Greek Letter "Phi"

**A.1.1 Input and Output Terminal Graphics.**

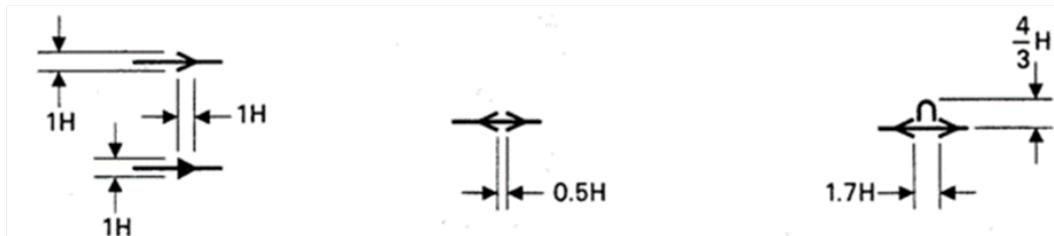
When an Input terminal is on the Left side it does not require a terminal direction graphical indicator. When the Input terminal is placed on any other side of the symbol the terminal Direction should be marked by a graphical indicator.

When an Output terminal is on the Right it does not require a terminal direction graphical indicator. When the Output terminal is placed on any other side of the symbol the terminal Direction should be marked by a graphical indicator.

Bi-directional terminals should always have a graphical indicator. The IEEE spec recognizes two separate forms of a bidirectional signal and has specified different graphical conventions: (IEEE 315 Pg. 32). The ASNI IEEE spec allows either "closed arrows" or "open arrows" to be used for indicating signal direction.



A.1.2. Bidirectional Signal – Only One Direction at a time



Note that when a bi-directional signal is an analog signal, the analog symbol indicator is placed between the arrows. The dimensions of the arrows follow the same dimensions as single directional arrows.

A.1.3. Bidirectional Signal – Both directions Simultaneously

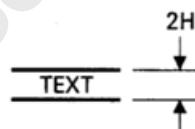
The arrow points are directed towards each other when the bidirectional signal is in both directions simultaneously.



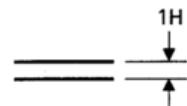
A.1.4. Text Height



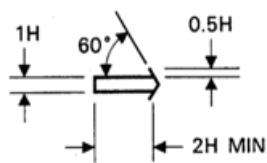
A.1.5. Line to Line with Text

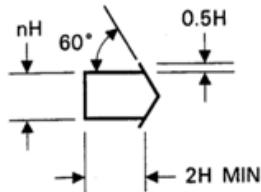


A.1.6. Line to Line without Text

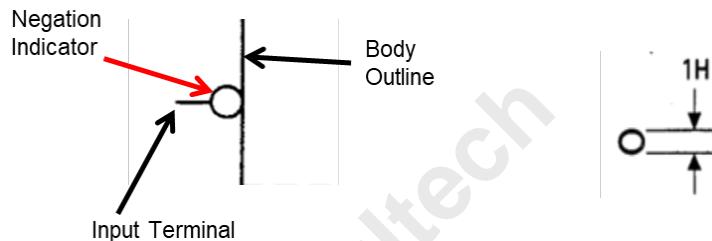
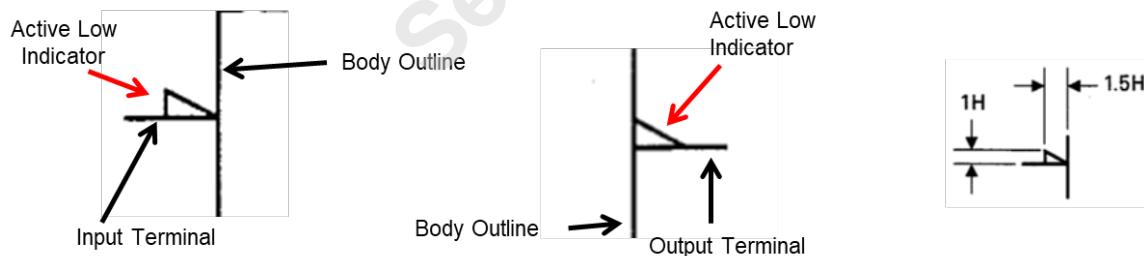


A.1.7. Graphic Sizing for Bus without Text



A.1.8. Graphic Sizing for Bus with Text**A.2. Recommended Graphic Symbol Representation of Signals, Properties, Functions**

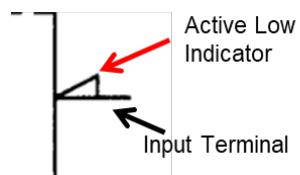
The following graphics are recommended on the symbol graphics to visibly show the function of the symbol or part.

A.2.1. Negation**A.2.2. Active Low & Active High**

"Active Low" terminals has a graphical indicator, that can represent

1. "A low signal activated this input"
2. "A low signal implies this output has been activated."

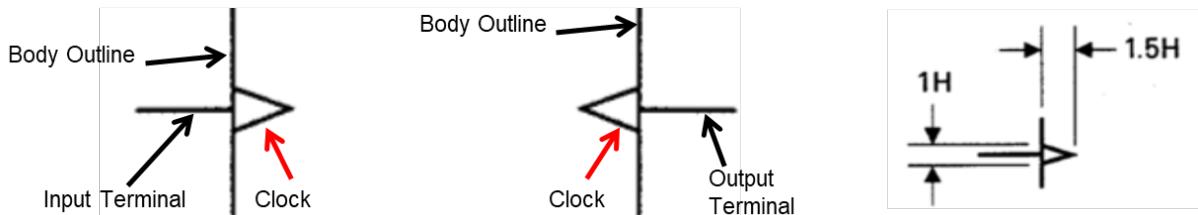
Level indicators are oriented along the direction of the signal flow. If the input terminal is on the right side of the symbol, the active low indicator should be drawn as shown in the image to the right.



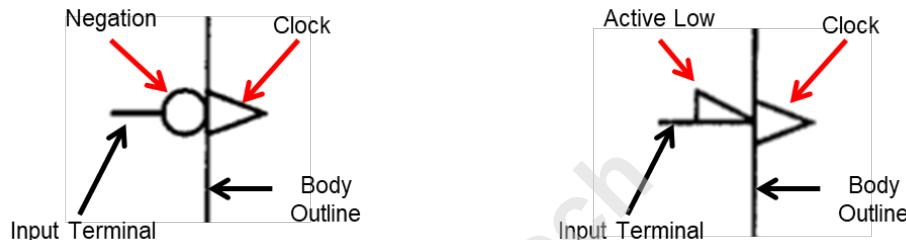
Since Active High is the assumed behavior, graphical indicators for "active high" are omitted from the symbol.

A.2.3. Clock

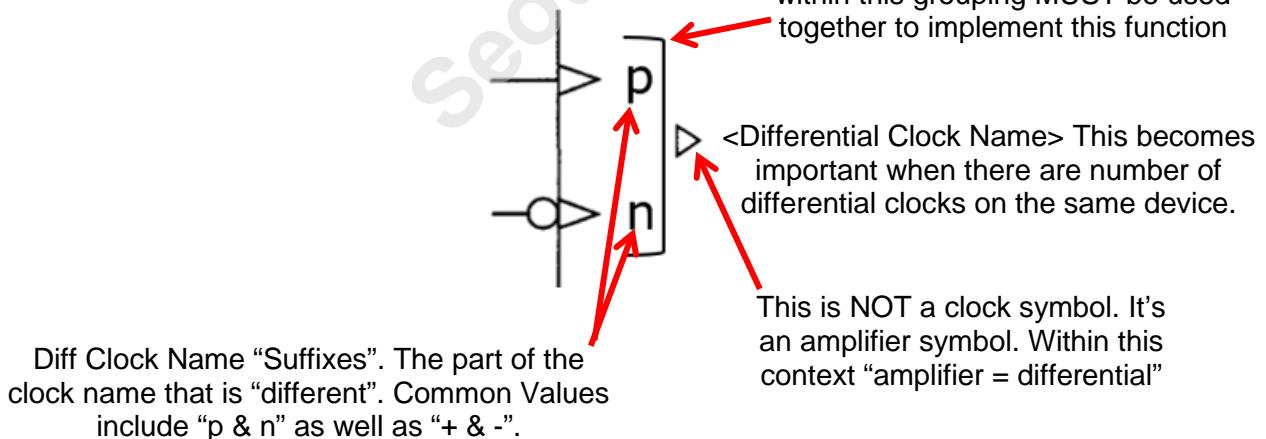
The Clock signal should be represented on the input terminal and the output terminal as shown below



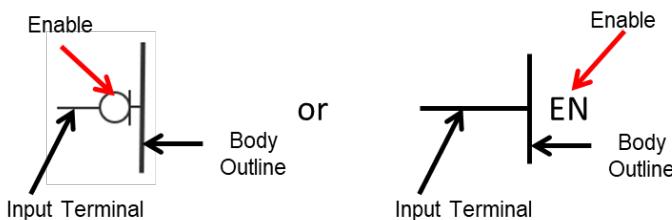
When combining Clocks with polarity indicators, the symbols look like the image below.



A.2.4. Differential Clock

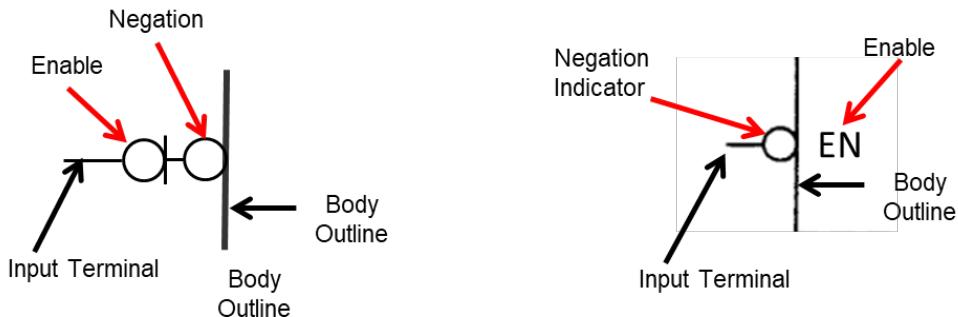


A.2.5. Enable

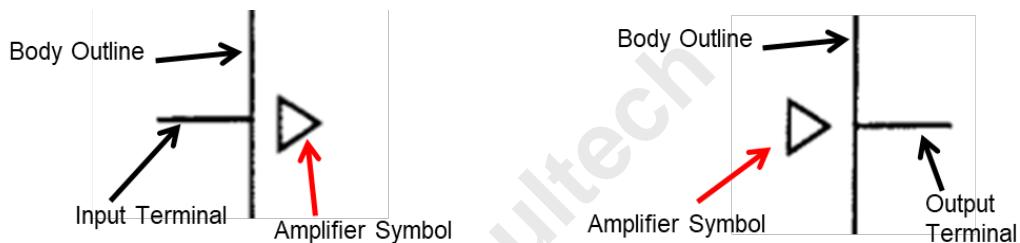


A.2.5 Enable (cont'd)

When combining Enable with a Negation indicators, the symbols look like the image below.

**A.2.6 Amplifier**

The amplifier should be represented on the input and the output terminal as shown below

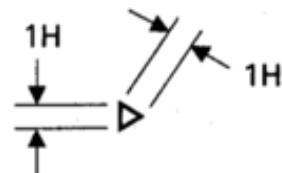
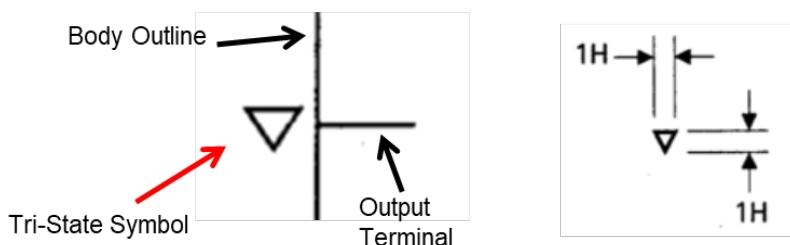


The Amplifier symbol applies only to inputs and outputs. It does not apply to Bi-directional signals. It may be used in combination with other terminal function symbols.

Output Combination Restrictions: If this symbol is used with Open Circuit Output, Open Circuit Output H-Type, Open Circuit Output L-Type, Passive-pulldown, Passive-pullup or Tri-State symbols, those symbols are placed between the amplification symbol and the edge of the Body outline.

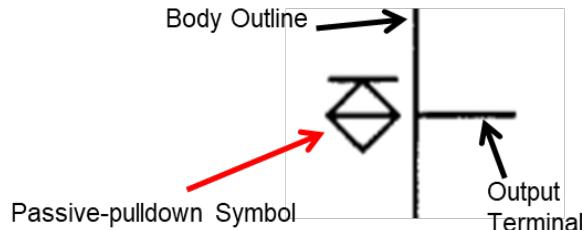
Input Combination Notes: For use with the hysteresis symbol, the line-grouping symbol, or dependency notation

The recommended graphic sizing for the Amplifier symbol is shown here.

**A.2.7 Tri-State Output**

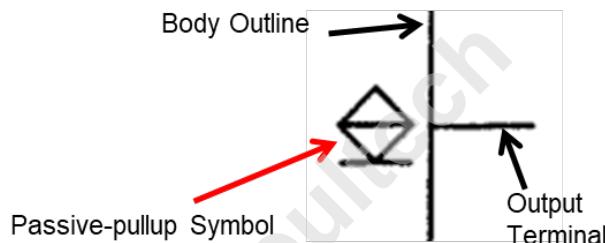
A.2.8. Passive Pulldown

Supports simple wired logic without the need for an external component (Resistor). The passive pulldown should be represented on the output terminal as shown below.



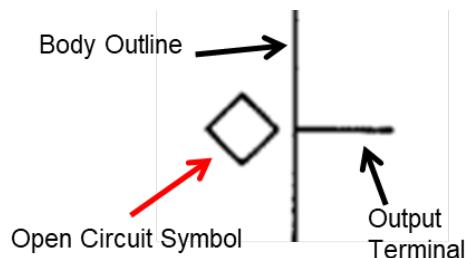
A.2.9. Passive Pullup

Supports simple wired logic without the need for an external component (Resistor). The passive pullup should be represented on the output terminal as shown below.



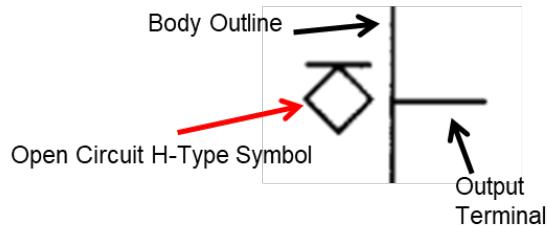
A.2.10. Open Circuit Output

Examples of open circuit output include open-emitter, open-collector, open-source, or open-drain output. Open Circuit Output should be represented on the output terminal as shown below.

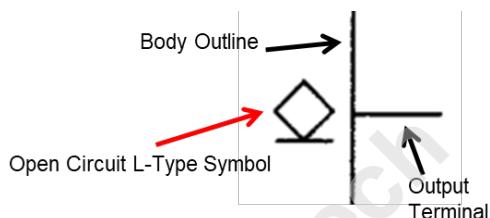


A.2.11. Open Circuit Output (H-Type)

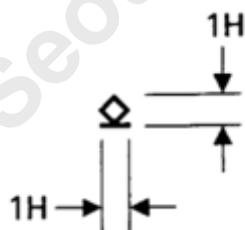
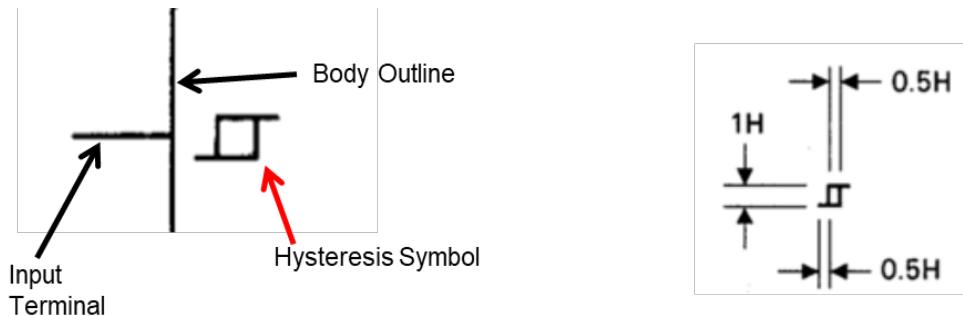
Examples of open circuit output High type include PNP Open-collector, NPN open-emitter, P-Channel open-drain, or N-Channel open Source. Open Circuit Output H-Type should be represented on the output terminal as shown below.

A.2.11 Open Circuit Output (H-Type) (cont'd)**A.2.12. Open Circuit Output (L-Type)**

Examples of open circuit output Low type include NPN Open-collector, PNP open-emitter, N-Channel Open-drain, or P-channel open-source. Open Circuit Output L-Type should be represented on the output terminal as shown below.



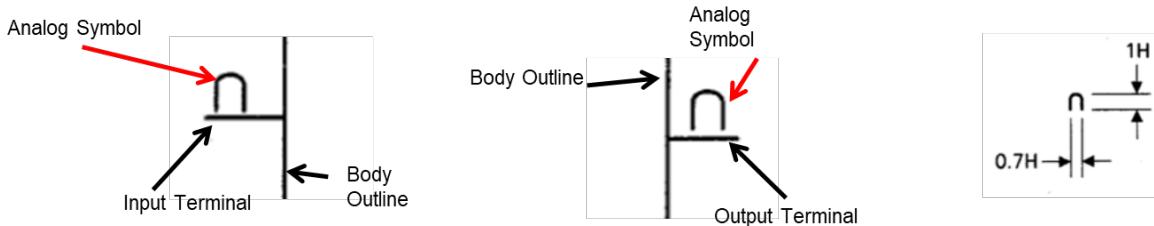
The recommended graphic sizing for an Open Circuit (L Type) symbol is shown below.

**A.2.13. Schmitt Trigger**

The Hysteresis symbol may be used in combinations with other terminal function symbols. When used with a dynamic input symbol (clock symbol), the hysteresis symbol shall be shown following the dynamic input symbol even though its effect occurs first. For use with the special amplification symbol, the line-grouping symbol, or dependency notation

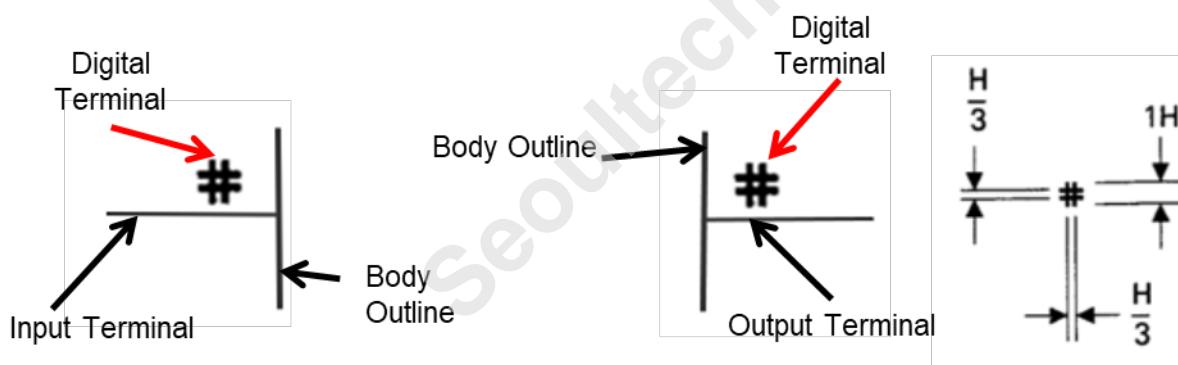
A.2.14. Analog Terminal

The Analog terminal symbol should be used on the input and output terminals only when it is necessary to distinguish analog signals on parts that contain both analog and digital signals.



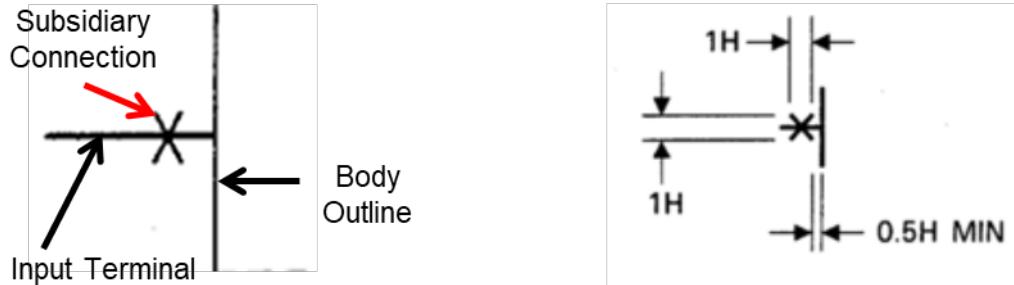
A.2.15. Digital Terminal

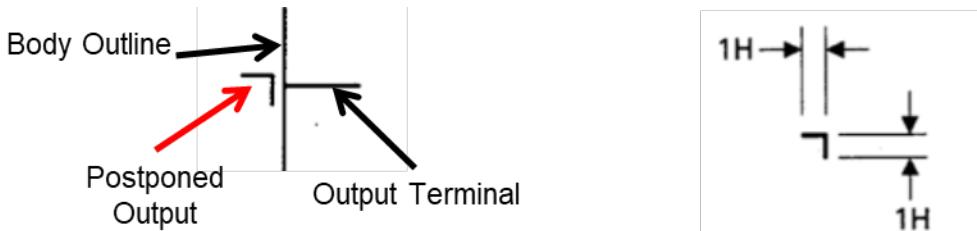
Signals are assumed to be digital so in most situations, the use of the digital terminal symbol is not used. When the display of the symbol would be simplified or clarified the digital terminal symbol may be used. The Digital terminal symbol should be represented on the input, output, or Bi-directional signals as shown below.



A.2.18. Subsidiary Connection

Subsidiary connection symbol may be used to designate an input supplying power to the device or a connection, the knowledge of whose level is not important to understand the function of the element and the circuit (such as a connection to an external supplementary resistor or capacitor).



A.2.19. Postponed Output**A.3. Recommended Rules**

This section defines the recommended rules for symbol ordering, terminal grouping and their graphical sizing. There are Two Types of terminal symbols

- Symbols on the pin stub
- Symbols within the symbol body

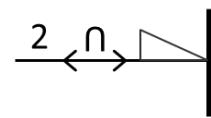
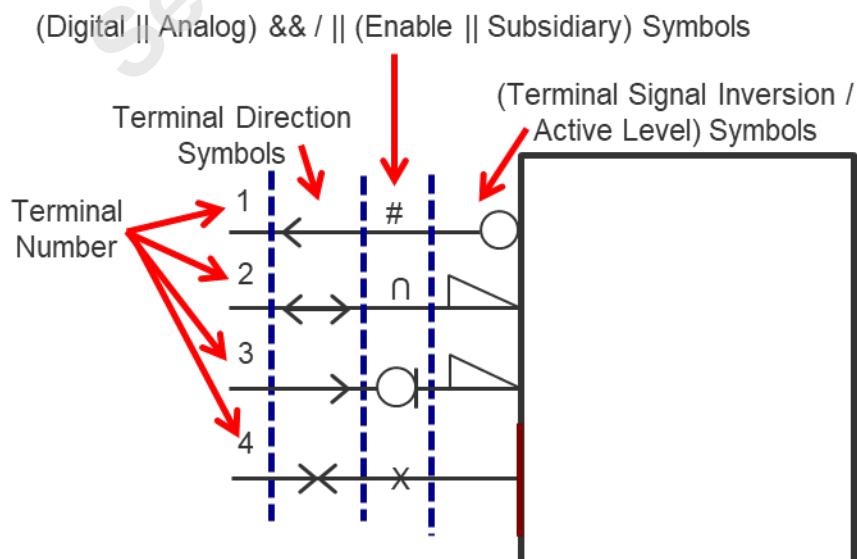
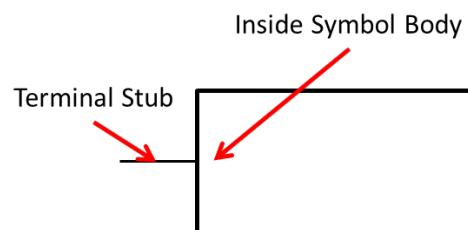
Both Types of terminal symbols may be present in combinations. The Terminal Stub: Composed of four "sections"

- Terminal Number
- Terminal Signal Direction
- Analog or Digital and/or Enable or Subsidiary marker
- Inversion or Level activation market

It is good practice that all Terminals on any one side of the symbol must have the same length. The minimum stub length is determined by the terminal requiring the most terminal symbol "decorations".

The minimum stub length may be reduced by stacking symbols as in the case shown above of an analog symbol on a bidirectional symbol.

A simple symbol pattern consisting of terminal functional graphics internal to the symbol body plus located on the terminal stub is shown below.

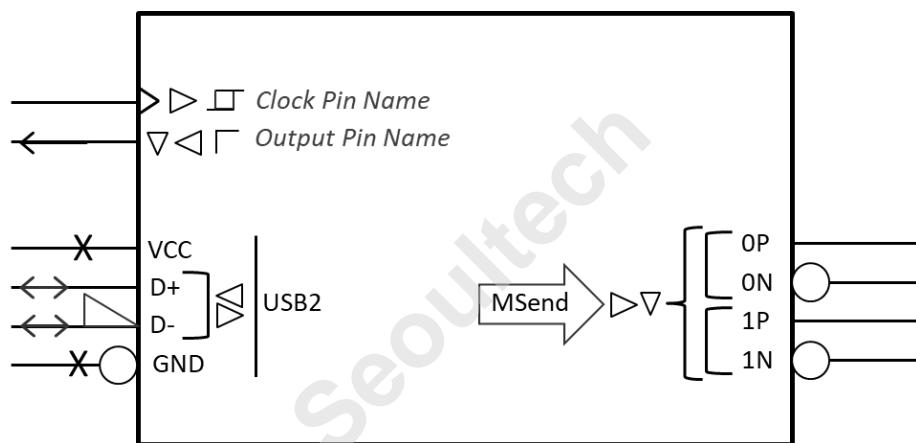


A.3 Recommended Rules (cont'd)

Inside the Symbol Body

- a. Terminal Specific Functional Symbol may be comprised of up to 3 symbols
- b. Terminal Name (or Terminal Name Suffix / Bit)
- c. Grouping Symbol
- d. Group Functional Symbol may be composed of up to 2 symbols
- e. Group Name (Terminal Base Name), may be wrapped with "Group Direction Symbol"
- f. Groups symbols may be "nested" or stacked.

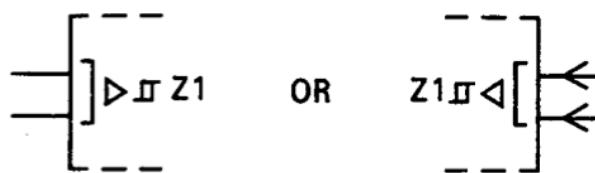
Where Possible move the function symbols to the highest common level {furthest from Terminal}. This will minimize symbol duplication.



A.3.1. Symbol Ordering Rule

If more than one symbol is required for one terminal the symbol order is defined by the Drafting Standard

- a. Terminal Grouping
- b. Amplification
- c. Hysteresis



A.3.2. Terminal Grouping

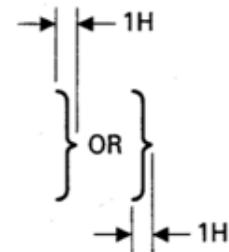
Three Forms of Terminal Grouping are defined by the ANSI IEEE spec:

- a. Bit Grouping
- b. Function Grouping, and
- c. Common Name Grouping.

A.3.2.1 Bit Grouping

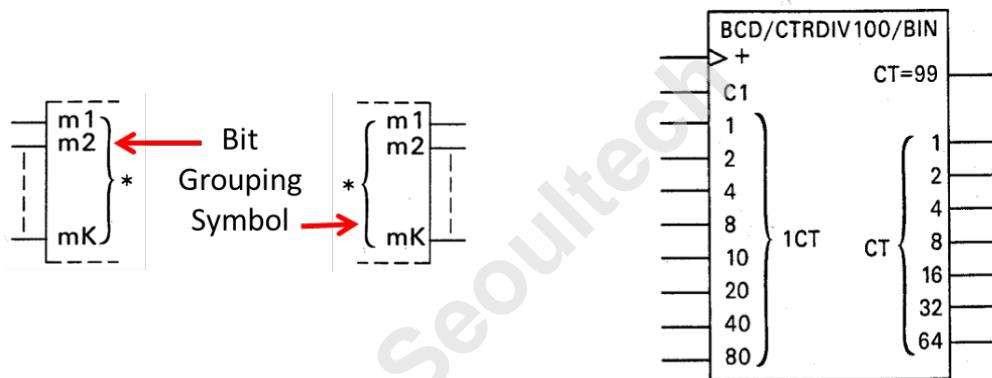
Bit-Grouping is denoted using Curly Braces "{}" & "{}"

- a. Inputs & Outputs {No Bidirectional Terminals}
- b. Does NOT support mixing inputs & outputs in the same group
- c. "Outputs grouped by this symbol represent a value that is the sum of the individual weights of the outputs standing at their internal 1-states. The individual outputs shall be shown in ascending or descending order by weight"



The recommended graphic sizing for Bit Grouping is shown below. The precise shape of the curly braces is unimportant.

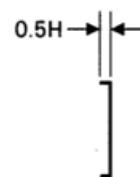
A few examples of Bit Grouping are shown below:



A.3.2.2 Function Grouping

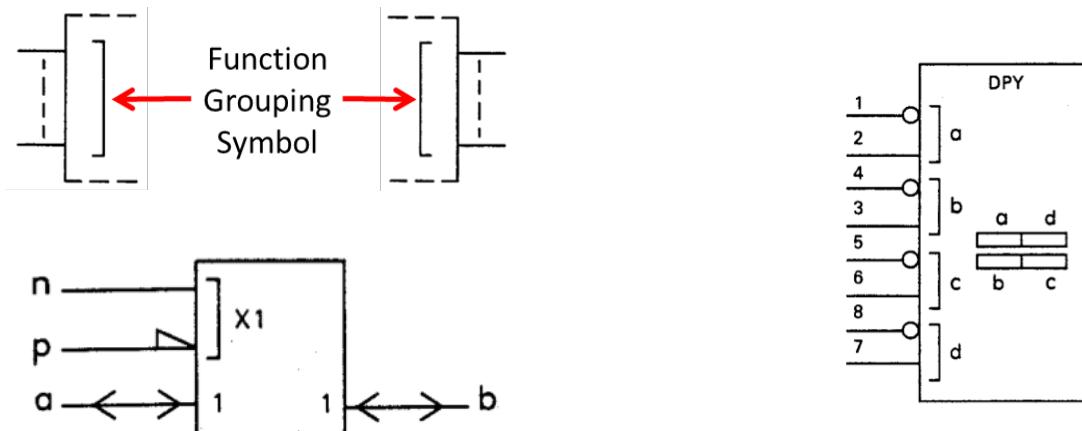
Function Grouping is denoted by using Square Bracket "[" & "]"

- a. Inputs & Outputs {No Bidirectional Terminals}
- b. Does NOT support mixing inputs & outputs in the same group
- c. "This symbol indicates that two or more terminals are needed to implement a single logic input."



A few examples of Function Grouping are shown below:

A.3.2.2 Function Grouping (cont'd)

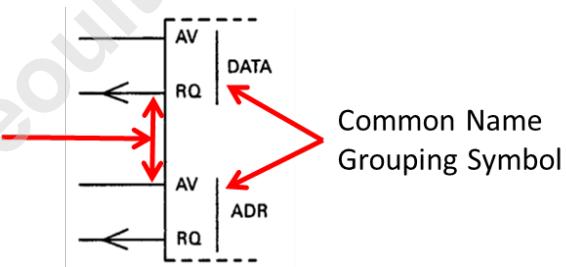


A.3.2.3 Common Name Grouping

- a. Inputs, Outputs & Bidirectional Terminals.
- b. Supports Mixing inputs, outputs & Bidirectional terminals in the same group.

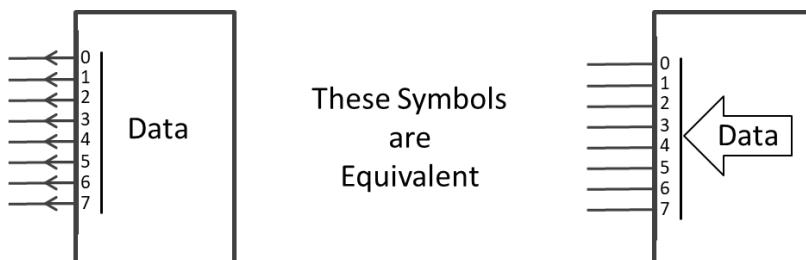
A few examples of Common Name Grouping are shown below:

Terminal to Terminal Spacing
may be adjusted (increased)
at Terminal group boundaries



A.3.3 Group Direction Symbols

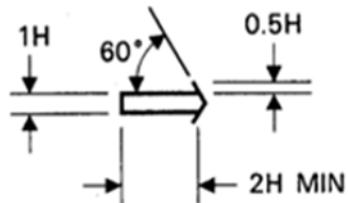
Group Direction Graphics may be used with label names to indicate signal direction. If group direction graphics are used direction symbols on individual terminals are not required.



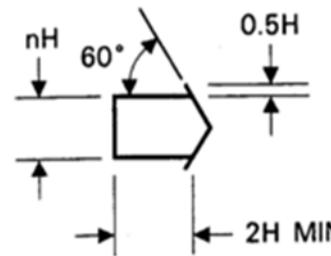
A.3.3 Group Direction Symbols (cont'd)



Group Name is enclosed within the Group Direction Symbol



Bus directional arrow without the Group Name



Bus with Group Name within the directional Arrow

Annex B (informative) Differences between JEP30-E100 and its predecessors

This table briefly describes most of the changes made to entries that appear in this standard, JEP30-E100, compared to its predecessor; Punctuation changes may or may not be included.

Initial Issue: N/A	Date: APRIL 2018	JC11 Item Number: 11.2-938
--------------------	------------------	----------------------------

Change Record History

Issue: A	Date: March 2023	Item Number: 839S

Description of changes

Section 2 Applicable Documents: Added in references to additional standards that are applicable to this Standard

Section 4.3 Linking the Manufacturing Part Number to a specific Electrical Data set: Revised this section.

Section 4.4.x Electrical Section: Restructured this section to facilitate adding Digital Signatures.

Section 4.5.1 Part Classification Array: Added in new sub-classifications for *CableAndWiring*. Added in section for *CompanionPart*. Classification types are added or updated to add a *Property-Array* structure for every classification and sub-classification branch. Classification hierarchy is limited to 3 levels under each section, with additional levels transferred to properties.

Section 4.5.1.1.x Cable and Wiring Classification Type: Added in a classification structure with Properties for *CableAndWiring*

Section 4.5.1.2.x Connector Classification Type: Added in a classification structure with Properties for *Connector*.

Section 4.5.1.3.1.x Amplifier Classification: Added in 2 new sub-classifications “*Audio*” and “*Video*”.

Section 4.5.1.3.3.x Battery Classification: Moved Material sub-classification into *Material* property.

Section 4.5.1.3.4.x Capacitor Classification: Added *Silicon* as a new Sub-classification to Fixed Capacitor. Added *Class*, *Dielectric*, *Electrolyte* and *Material* properties throughout the Capacitor Classification structure.

Section 4.5.1.3.5 Circuit Protection Classification: Revised this section using *OverCurrent*, *OverVoltage* and *OverTemperature* Protection classifications.

Section 4.5.1.3.7 Diode Classification: Added *LED*, *PIN*, *SiliconCarbide*, and *ESD* as new sub-classifications

Section 4.5.1.3.8 Filter Classification: Added *Ceramic* and *SAW* as new sub-classifications and moved Transfer Function to a Property

Annex B (cont'd)

Section 4.5.1.3.10 IC Classification: Added <i>DataAcquisition</i> , <i>DigitalSignalProcessing</i> , <i>Interface</i> , <i>Power Management</i> and <i>Timing</i> as new sub-classifications
Section 4.5.1.3.11.x Inductor Classification: Revised this section.
Section 4.5.1.3.12.2 NonVolatile Memory Classification: Added <i>EPROM</i> , <i>MASKROM</i> and <i>PROM</i> as new sub-classifications.
Section 4.5.1.3.13.x Optoelectronics Classification: Revised this section.
Section 4.5.1.3.14.x Regulator Classification: Revised this section.
Section 4.5.1.3.15.x Relay Classification: Changed <i>Coil</i> to <i>ElectroMechanical</i> . Moved Contact Form classifications to property. Added Functional classification property.
Section 4.5.1.3.16.x Resistor Classification: Reduced number of classification levels and moved Material and Temperature coefficient to Properties.
Section 4.5.1.3.19 Switch Classification: Moved Contact Form to Property. Transferred extended Package Outline from JEP30-P101 to sub-classifications in this section.
Section 4.5.1.3.20.x Thyristor Classification: Added “ <i>DIAC</i> ”, “ <i>SCR</i> ”, “ <i>SIDAC</i> ” and “ <i>TRIAC</i> ” as new sub-classifications. Moved Direction and Conductance classifications to Properties
Section 4.5.1.3.21.x Transformer Classification: Changed sub-classification <i>Power</i> to <i>Pulse-or-Power</i>
Section 4.5.1.3.22.x Transistor Classification: Reduced number of classification levels and moved Material and Transistor types to Properties.
Section 4.5.1.4.x Hardware Classification Type: Added in a classification structure with Properties for <i>Hardware</i> .
Section 4.5.1.5.x Optics Classification Type: Added in a classification structure with Properties for <i>Optics</i> .
Section 4.7.2.3.2 Terminal Swap Array: Added in <i>TerminalName</i> into <i>TerminalSwap</i> group
Section 4.7.2.3.3 Function Swap Array: Added in <i>TerminalNameOrderedList</i> into <i>FunctionSwap</i> group
Section 4.7.3.9.x Interface Function: Grouped liked Interfaces into a hierarchy – <i>DDR3</i> , <i>DDR4</i> , <i>DDR5</i> , <i>DDR6</i> , <i>HBM</i> , <i>HDMI</i> , <i>MultiMediaCard</i> , <i>MII</i> , <i>PCIe</i> , <i>CablingPCIe</i> , <i>C-PHY</i> , <i>D-PHY</i> , <i>Universal Flash Storage</i> and <i>USB</i> . Added in several new Interfaces into this section - <i>DDR4-X72</i> , <i>LPDDR4</i> - Single Channel, <i>LPDDR4</i> - Dual Channel, <i>DDR4DB02</i> , <i>DDR5-x4</i> , <i>DDR5-x8</i> , <i>DDR5-x16</i> , <i>LPDDR5</i> , <i>GDDR5</i> , <i>GDDR5X</i> , <i>GDDR6</i> , <i>FC-PI-6</i> , <i>HBM1</i> , <i>HBM2</i> , <i>HBM2E</i> , <i>HBM3</i> , <i>eMMC</i> , <i>MMC Mode</i> , <i>SPI Mode</i> , <i>OIF-CEI-04.0</i> , <i>A-PHY</i> , <i>M-PHY</i> , <i>UniPro</i> , <i>UFS</i> , and <i>UFSHCl</i>
Section 4.5.4.1.1 Units. Added additional Units-of-Measure to Units and updated Table 3 with their enumerated values
Section 4.5.4.3.x Parameter Graph: Enhanced the Graph section to enabling formatting of the graph data.

Annex B (cont'd)

Section 4.5.6 ESD: Changed reference to the “ANSI/ESDA/JEDEC JS-001-2014” to “JS-001-2017” and updated the enumerated list for the HBM-Classification Type to align with JS-001-2017. Changed reference to the “ANSI/ESDA/JEDEC JS-002-2014” to “JS-002-2018”.

Revised Table of Contents

Issue: B	Date: August 2023	Item Number: 400.07
Description of changes		
Section 4.5.1.3.14.4. Linear Voltage Regulator Property-Array: Changed Max occurrence under Protection from 4 to 3.		
Section 4.5.3.1 Super Interface – Array: Added new section for Super Interfaces.		
Section 4.5.3.10.3: Added new Interface for the Compute Express Link		
Section 4.5.3.10.13: Added new Interface for the Embedded Display Port Interface		
Section 4.6.1 Symbol – Array: Added optional digital signature capability to Symbols		
Section 4.6.1.1.3.3.5 Terminal – Array: Make element “Terminal/TerminalGroupGraphicsID” unbounded.		
Section 4.6.2 Required Circuitry – Array: Removed Simulation Model and added Required Circuitry Signature.		
Section 4.7.1 Electrical Map – Array: Added Operational Mode		
Section 4.7.2. Package Terminal Map: Added in Recommended Netlist Name and Companion Terminal to Terminal Map Type.		

Issue: C	Date: November 2023	Item Number: 400.08
Description of changes		
Section 2.18, Added new reference to Accellera.org		
Section 4.3, Update diagrams to include Software Interface Description Association		
Section 4.3.18, Added new section to show the linking of the Manufacturing Part Number to the Software Interface Description		
Section 4.10: Added new section for Software Interface Description Array		

Annex B (cont'd)

Issue: D	Date: February 2024	Item Number: 400.09
Description of changes		
Section 4.7, Update diagrams to include Die Terminal Map		
Section 4.7.3, Add new section for Die Terminal Map		

Issue: E	Date: August 2024	Item Number: 400.10
Description of changes		
Section 2 Applicable Documents: Added new reference for MathML		
Section 4.1, and section 4.2: Update sections to align with modifications performed at the JEP30 parent structure.		
Section 4.5.1 Part Classification – Array: Update entire section to include OverflowCategories group to enhance the existing Other element with Parts and Accessories.		
Section 4.5.1.2.5.1 CardEdge-to-Cable Interface Connector Classification and section 4.5.1.2.5.2 CardEdge-to-Cable Interface Connector Classification: Added missing enumerated list for CableStyle.		
Section 4.5.1.3.4 Capacitor Classification: Remove the Fixed Level and moved all sub-categories up one level.		
Section 4.5.1.3.4.1 Ceramic Capacitor Classification and Property-Array: Added MultiLayer element.		
Section 4.5.1.3.4.2 Electrolytic, 4.5.1.3.4.3 Film, 4.5.1.3.4.4 Silicon Capacitor Classifications: Added Array element.		
Section 4.5.1.3.6 Data Converter Classification: Changed “Current-to-Voltage” to “Current-Voltage”, and “Frequency-to-Voltage” to “Frequency-Voltage”. Added “Digital Potentiometer” and “Mechanical-Signal”		
Section 4.5.2.3.7 Terminal-to-Terminal Signal Path – Array: Updated image to fix spelling error.		
Section 4.5.3.1 Super-Interface – Array: Updated image		
Section 4.5.4.1 Test Condition: Added Footnote and updated Value and Rule structure		
Section 4.5.4.1.1 Units: Added several new UOMs and updated Table 3 with enumerated lists		
Section 4.5.4.1.1.1 ComplexUOM: Added comments to further explain ComplexUOM		
Section 4.5.4.2.1 Parameter: Added Footnote and updated Value and Rule structure		
Section 4.5.4.2.1.1 Values: Changed type from Values Set Type to ParametersValuesType.		

Annex B (cont'd)

Section 4.5.4.2.1.2: Added new section for Context
Section 4.5.4.3 Parameter Graph: Added Footnote and updated Graph Formula structure
Section 4.5.4.3.1.1. Linking the Data-Array to the Appropriate Parameter Definition: Updated image
Section 4.6.1.1.1 Part Symbol Version: Correct type for "Reason-for-Change".
Section 4.7.2 Package Terminal Map: Updated image with correction for Recommended netlist Name

Issue: F	Date: December 2024	Item Number: 400.11
Description of changes		
Section 4.5.1 Part Classification: Update image for Overflow Categories Group to reflect that Property-Array is optional		
Section 4.5.1.2 Connector Classification: Merge Board-to-Board and Cable-to-Board to Board. Rename Cable-to-Cable to Cable. Merge CardEdge-to-Board and CardEdge-to-Cable to CardEdge. Then re-order based on priority of assignment.		
Section 4.5.1.2.1 CardEdge Classification: Consolidate level 3 sub classifications that was in CardEdge-to-Board and CardEdge-to-Cable prior classifications into this new CardEdge classification.		
Section 4.5.1.2.3 Socket Classification: Replace specific socket types with two generic types of PCB-Mountable and Non-PCB-Mountable classifications.		
Section 4.5.1.2.4 Board Classification: Consolidate level 3 sub classifications that was in Board-to-Board and Cable-to-Board to Board into this new Board classification.		
Section 4.5.1.2.5 Cable Classification: Rename Cable-to-Cable to Cable throughout this section.		

Annex B (cont'd)

Issue: G	Date: February 2025	Item Number: 400.12
Description of changes		
Section 4.5.1.3 Electrical Classification: Changed Drivers to ElectromechanicalDrivers		
Section 4.5.1.3.10 IC Classification: Revised the IC sub-classifications.		
Section 4.5.1.3.18 Sensor Classification: Added Position and Proximity Sensor classifications.		
Section 4.5.2.1.3 Signal Direction: Update image to make Bidirectional choice optional.		
Section 4.5.4.1 Test Condition: Update structure to add Groups enforcing consistency across other schemas. Changed Value to Nonimal. Also added Footnote ID's and Test Method.		
Section 4.5.4.1.1 Units: Added several units and updated Complex UOM ... UOM type		
Section 4.5.4.2.1 Parameter: Update structure to add Groups enforcing consistency across other schemas. Also added Footnote ID's.		
Section 4.5.4.2.1.1 Values: Update structure to add Groups enforcing consistency across other schemas. Also added Footnote ID's.		
Section 4.5.4.2.1.2 Rule Context: Changed "Context" to "Rule Context"		
Section 4.5.4.3.1.1 Linking the Data-Array to the Appropriate Parameter Definition: Updated image due to insertion of groups.		
Section 4.5.4.3.2. Data-Array: Changed "TestCondition" to "PlotTestCondition" to distinguish the different types of Test conditions, and make unbounded. Updated image and added Footnote ID to Paramater Value and PlotTestCondition.		
Section 4.5.4.3.3 Graph Formula: Moved structure to a group and place under an element called GraphFormula.		
Section 4.5.5 Truth Table: Correct list of types in document.		

Issue: H	Date: May 2025	Item Number: 400.13
Description of changes		
Section 4.5.4.1.1 Units: Added additional units of measure to diagram and table		
Section 4.5.4.3.1 Formatting: Updates the X and Y axis Range from xs:int to xs:decimal.		

Annex B (cont'd)

Issue: I	Date: September 2025	Item Number: 400.14
Description of changes		
Section 2.1 Applicable Documents/JEDEC: Added new document for HBM4		
Section 4.1 PartModel - SupplyChain Section: Updated images to match changes in the JEP30 parent schema		
Section 4.2 Manufacturer Part Number-Array: Updated images to match changes in the JEP30 parent schema		
Section 4.3 Standards Identifier Array: Added new section to connect Standards Identifiers to the Electrical content.		
Section 4.4 Process Technology Identifier Array: Added new section to connect Process Technology Identifiers to the Electrical content.		
All Sub-sections of 4.5 Linking the Manufacturing Part Number to to a specific Electrical Data set: Added SuperInterfaceID to association diagram. Added TerminalDetails-Array and Functiongroup-Array at the Electrical Parameters Array level.		
Section 4.7.1 Part Classification Array: Added throughout all classifications ParameterID and ParameterSetID and made Property key value pair type optional under a Searchable Properties Group.		
Section 4.7.2.1.6.3 Programmable: Updated Programmable Output Circuit with image pointing to new units.		
Section 4.7.3.10.17 HBM Interface Function: Added HBM4. Added reference to " Standard Terminal Base Name" throughout schema in multiple sections alongside each instance of "Standard Terminal Name".		
Sections 4.7.4.x Electrical Specifications sub-sections: Add StandardsAuthorityBodyID with PropertyID to the ParameterIdentityGroup. Also changed Symbol to LaTeX-Symbol and added MathML-Symbol. Moved Units out from under Values and made it available to both Values and Rules. Change choice of Values or Rules to a Sequence and made Values optional, to enable a combination of Rules and Values to be defined for Min, Max and Nominal. Also added annotation for ValueText. Moved FootnoteID up one level. Moved ParameterRuleType to dictionary. Added choice for Ascii Units and Units. Added note of discontinuance in future release for Ascii Units. Added new sub section for Units including table of enumerated values, that comply with IEC definition.		
Section 4.7.3.14.2 Potentiometer Function: Correct image due to typo and added unique constraint.		
Multiple Sections: Update all examples in which Symbol was changed to LaTeX-Symbol and where Units was changed to AsciiUnits with a choice of Units.		
Section 4.7.4.2 Parameter Set: Added Title and Sub-Title to help correlate the data in the PartModel xml file to the tables in the datasheet. Also added Footnote Array.		

Annex B (cont'd)

Section 4.7.4.2.1.1 Values: Removed the branches for Standard and Limits because of duplication with the flatten structure of ValueSetgroup. Added annotation for ValueText.
Section 4.7.4.3 Parameter Graph: Added Footnote Array.
Section 4.7.4.3.1 Formatting: Updates the X and Y axis Range from xs:int to xs:decimal.
Section 4.7.4.3.1.1 Linking the Data-Array to the Appropriate Parameter Definition: Updated image to reflect the choice of Units.
Section 4.7.4.3.2 Data-Array: Update PlotTestCondition with the new Parameter identity Group and Units structure.
Section 4.7.6.1 HBM – Human Body Model: Documented the enumerated values of UOM.
Section 4.9 Mapping Array: Added new structure for Terminal Map to replace the Package Terminal Map and the Die Terminal Map which will be removed in 2028.
Section 4.9.1 Electrical Map Array: Update the Electrical Map to add additional reference a ParameterSetID, ParameterID, and ESD-ID.
Section 4.9.2 Terminal Map: Added new structure for Terminal Map. This new structure also enhances a structured Terminal Name Mapping that avoids mapping out every terminal name. The new “Terminal Map” structure will replace the existing “Package Terminal Map” and the “Die Terminal Map” which will be removed in future release.
Sections 4.9.5 Simulation Map: Added SuperInterfaceID to the Simulaion Map.

This page intentionally left blank.



Standard Improvement Form**JEDEC Standard No. JEP30-E100I**

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC
Attn: Publications Department
3103 10th Street North
Suite 240S
Arlington, VA 22201

Email: angies@jedec.org

1. I recommend changes to the following:

Requirement, clause number _____

Test method number _____ Clause number _____

The referenced clause number has proven to be:

Unclear Too Rigid In Error

Other _____

2. Recommendations for correction:

3. Other suggestions for document improvement:

Submitted by

Name: _____

Phone: _____

Company: _____

E-mail: _____

Address: _____

Date _____

City/State/Zip: _____

